

UDP10G-IP Demo Instruction

Rev1.1 8-Mar-19

Contents

1	Overview	2
Part A UDP10G-IP demo by using FPGA and PC..... 3		
2	Environment Setup.....	3
3	PC Setup.....	5
3.1	IP Setting	5
3.2	Speed and Frame Setting	6
3.3	Power Option Setting.....	9
3.4	Firewall Setting	10
4	FPGA board setup.....	11
5	Main menu	15
5.1	Display current parameter.....	15
5.2	Reset UDP10G-IP	16
5.3	Send Data Test	18
5.4	Receive Data Test.....	20
5.5	Full-duplex Test.....	22
Part B UDP10G-IP demo by using two FPGAs..... 24		
6	Environment Setup.....	24
7	FPGA board setup.....	26
8	Main menu	29
8.1	Display current parameter.....	29
8.2	Reset UDP10G-IP	30
8.3	Send and Receive data Test (Half-duplex test).....	32
8.4	Full-duplex Test.....	34
9	Revision History	36

1 Overview

The demo is designed to run UDP10G-IP for transferring 10 Gb Ethernet data by using UDP/IP protocol. Two test environments can be setup for the demo, as shown in Figure 1-1. First test environment (Test Env#A) uses one FPGA board transferring data with TestPC. More details to run the demo by using FPGA and TestPC are described in PartA.

Second test environment (Test Env#B) uses two FPGA boards to transfer data from the 1st FPGA to the 2nd FPGA. More details to run the demo by using two FPGAs are described in PartB.

User interface to set test parameters on FPGA and monitor hardware status is Serial console.

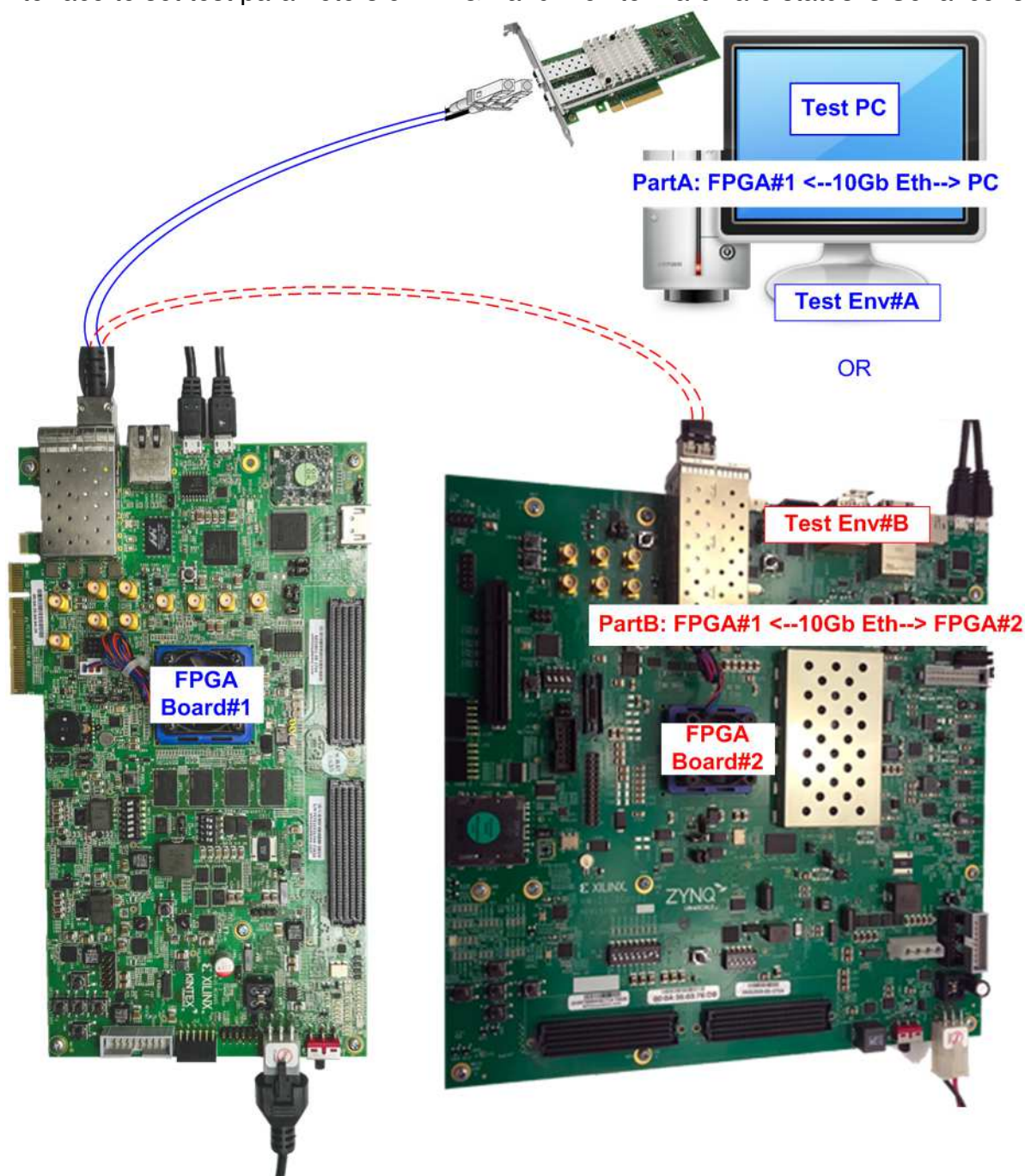


Figure 1-1 Two test environments for the demo

Part A UDP10G-IP demo by using FPGA and PC

“udpdatatest” is test application to run on TestPC for transferring data with FPGA by using UDP/IP protocol. The input parameters of “udpdatatest” by user are transfer direction (transmit or receive data), transfer size, and network parameters. User could run half-duplex test with UDP10G-IP by running “udpdatatest” on PC.

In addition, the demo supports to run full-duplex test with UDP10G-IP by running two “udpdatatest” on two Windows Command prompts. The first “udpdatatest” is run to receive data from UDP10G-IP while the second “udpdatatest” is run to send data to UDP10G-IP. In full-duplex test, two “udpdatatest” must use different port number to communicate with UDP10G-IP. More details of the demo are described as follows.

2 Environment Setup

To operate UDP10G-IP demo by using FPGA and PC, please prepare following test environment.

- 1) FPGA development boards (ZCU102/KCU105)
- 2) PC with 10 Gigabit Ethernet support or 10 Gigabit Ethernet card
- 3) 10 Gb SFP+ copper cable (DAC) or 2x10 Gb SFP+ transceiver (10G BASE-R) with optical cable for network connection between FPGA board and PC
- 4) micro USB cable for programming FPGA, connecting between FPGA board and PC
- 5) micro USB cable for Serial console, connecting between FPGA board and PC
- 6) “udpdatatest.exe” which is test application provided by Design Gateway, installed on PC
- 7) Serial console software such as HyperTerminal (Baudrate=115,200 Data=8 bit Non-parity Stop=1), installed on PC
- 8) Vivado tool for programming FPGA, installed on PC

Note: Test result in this document is captured by using following test environment.

[1] 10G Network Adapter: Intel X520-DA2

<http://www.intel.com/content/www/us/en/network-adapters/converged-network-adapters/ethernet-x520-server-adapters-brief.html>

[2] a) 10-Gigabit SFP+ DAC cable

<http://www.netgear.com/business/products/switches/modules-accessories/axc761.aspx>

b) 10-Gigabit SFP+ transceiver + optical cable

SFP+ transceiver (850nm)

<http://www.fit-foxconn.com/Product/ProductDetail?topClassID=Electronic%20Module&&PN=AFBR-709SMZ>

Optical cable 2105027-3 (LC to LC 1.8mm OM3 DPX LSZH&OFNR 3M)

[3] PC: Motherboard ASUS Z170-K, 32 GB RAM, 64-bit Windows7 OS

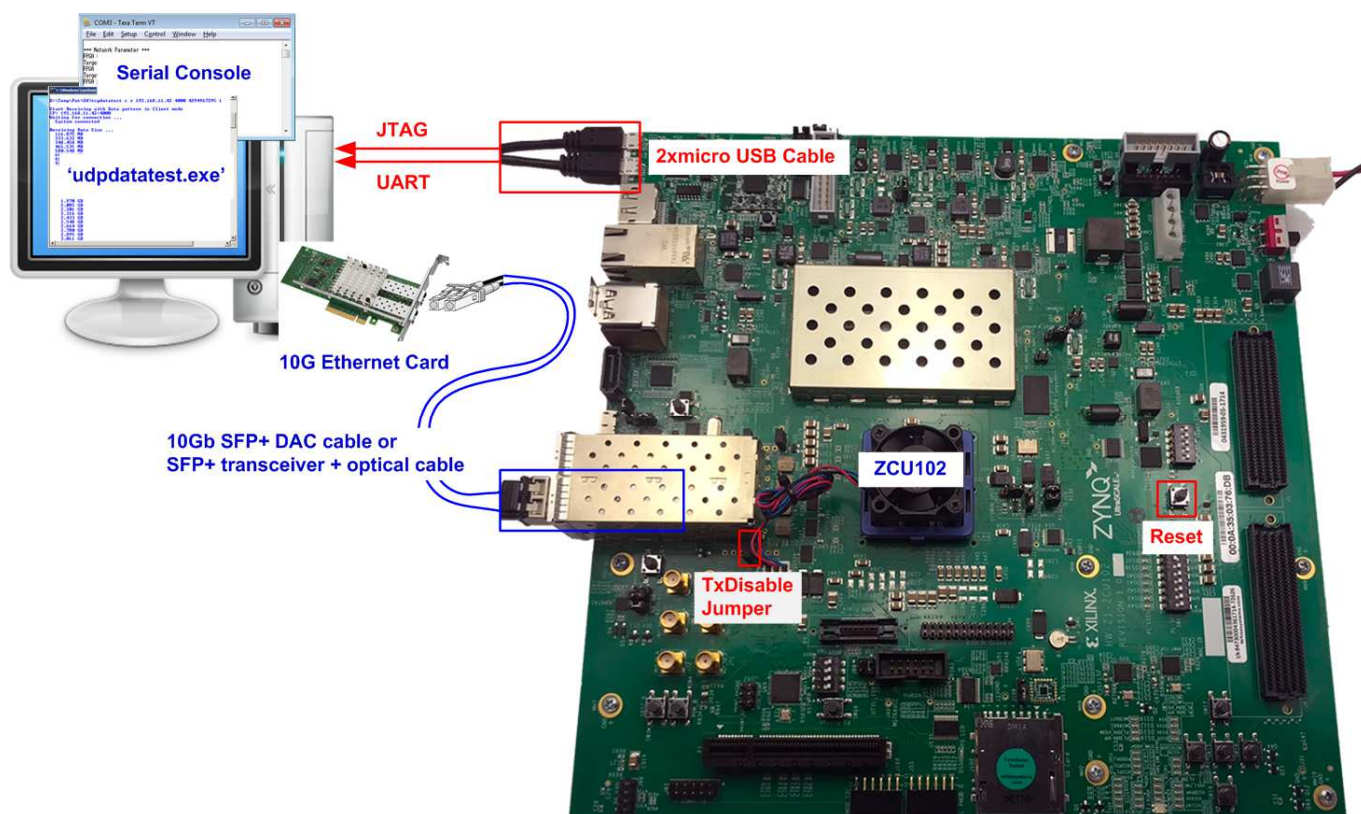


Figure 2-1 UDP10G-IP with CPU demo (FPGA <-> PC) on ZCU102

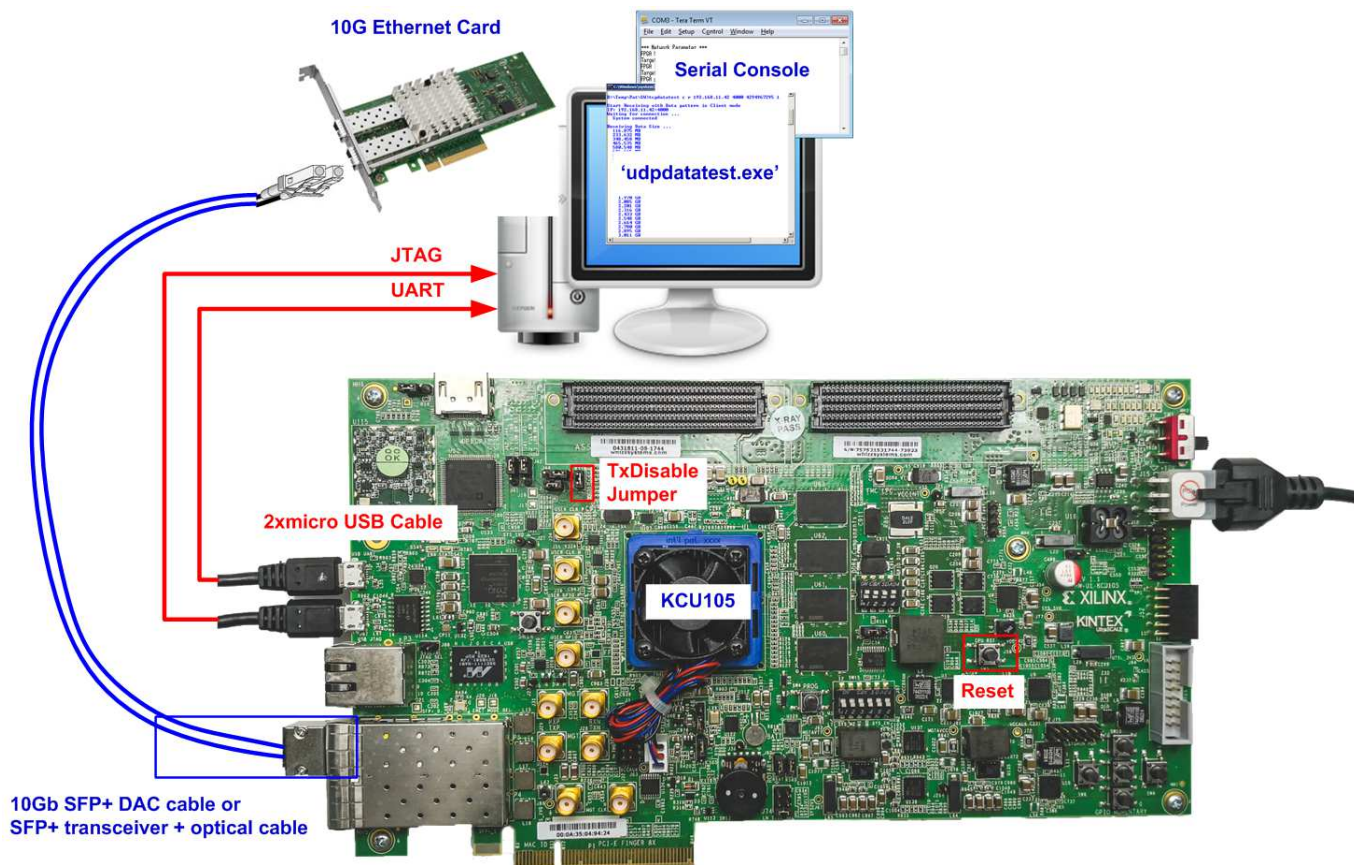


Figure 2-2 TOE10G-IP with CPU demo (FPGA <-> PC) on KCU105

3 PC Setup

Before running demo, network setting on PC is required. The example to set the network is described as follows.

3.1 IP Setting

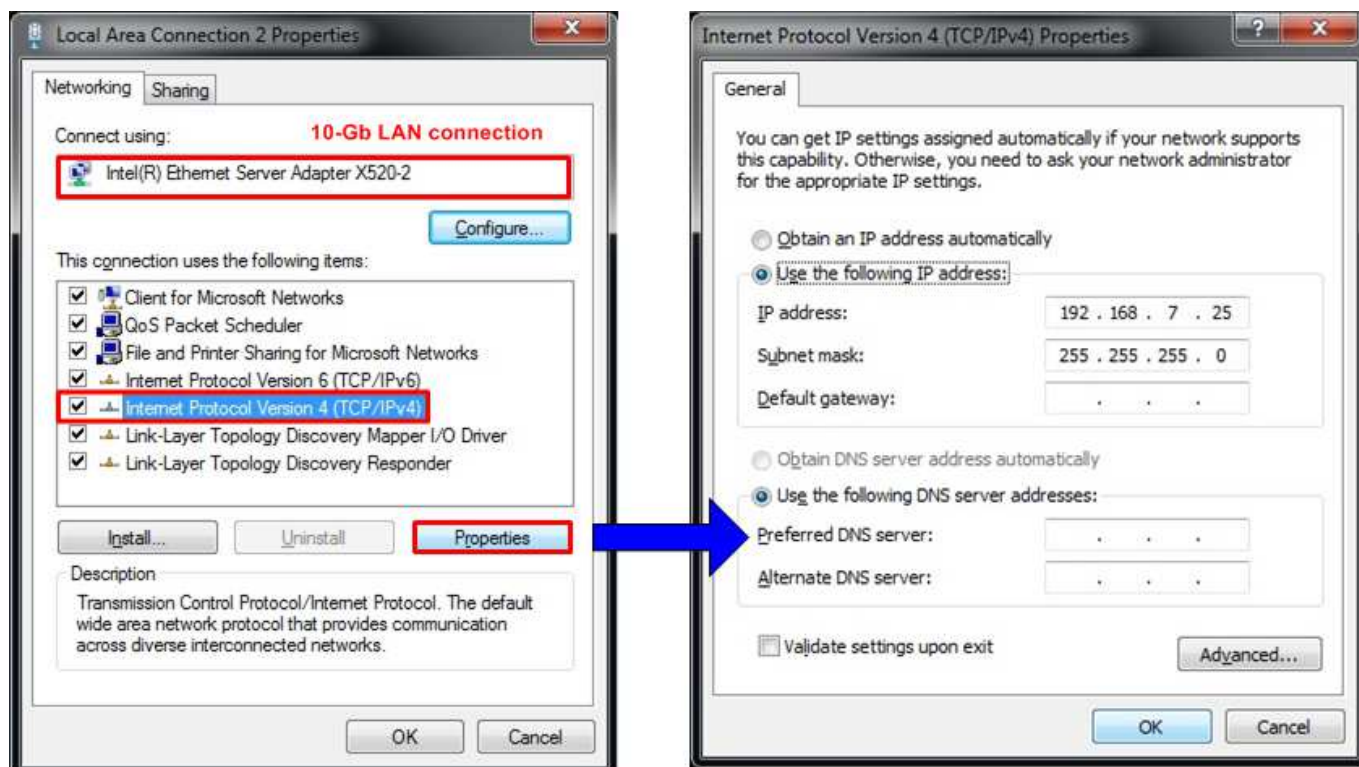


Figure 3-1 Setting IP address for PC

- 1) Open Local Area Connection Properties of 10-Gb connection, as shown in the left window of Figure 3-1.
- 2) Select “TCP/IPv4” and then click Properties.
- 3) Set IP address = 192.168.7.25, and Subnet mask = 255.255.255.0, as shown in the right window of Figure 3-1.

3.2 Speed and Frame Setting

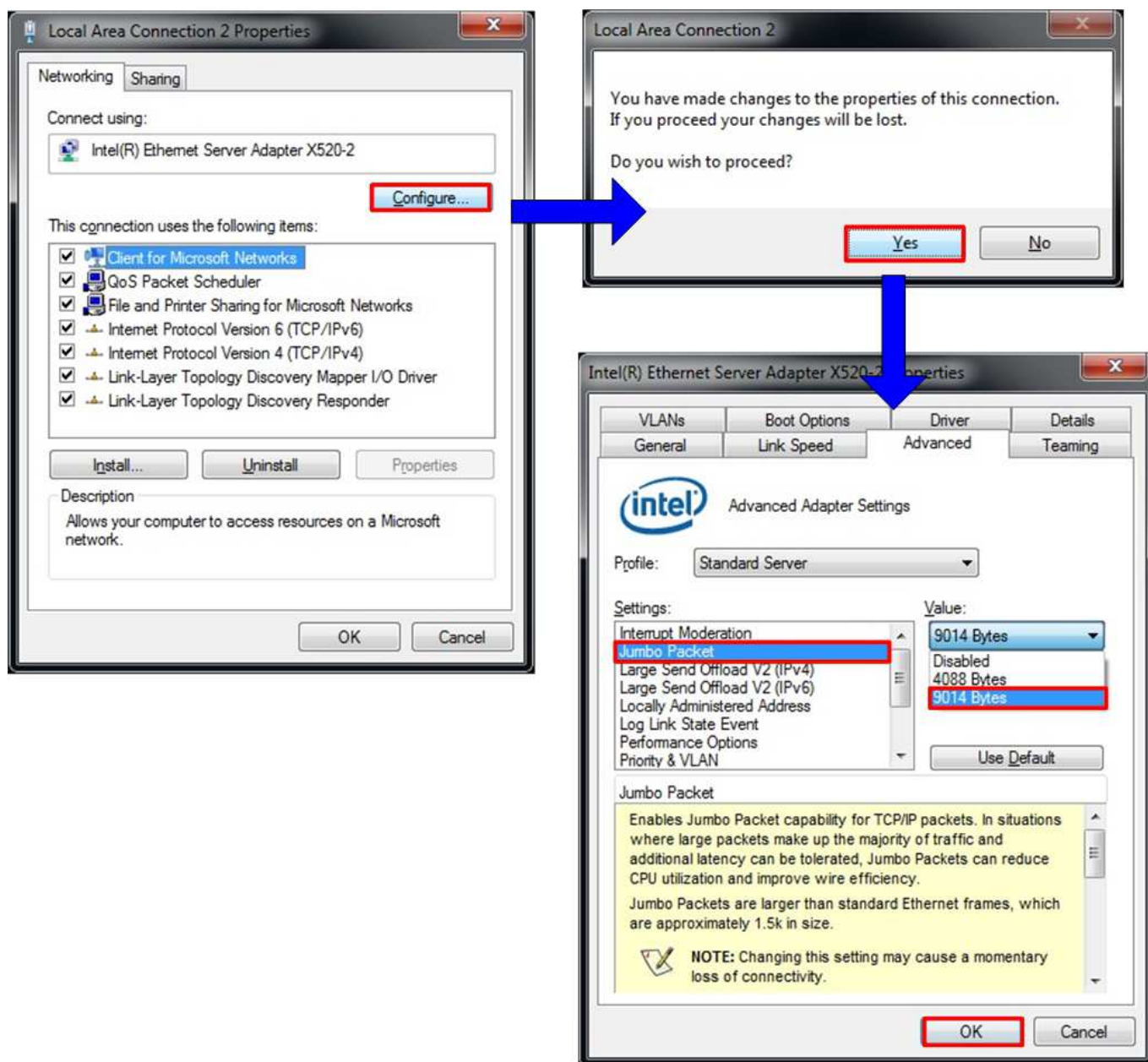


Figure 3-2 Set frame size = jumbo frame

- 1) On Local Area Connection Properties window, click “Configure” as shown in Figure 3-2.
- 2) On Advanced Tab, select “Jumbo Packet”. Set Value to “9014 Bytes” for Jumbo Frame support or set value to “Disabled” for non-Jumbo Frame support, as shown in bottom window of Figure 3-2.

- 3) On Link Speed, select “10 Gbps Full Duplex” for running 10-Gigabit transfer test, as shown in Figure 3-3.

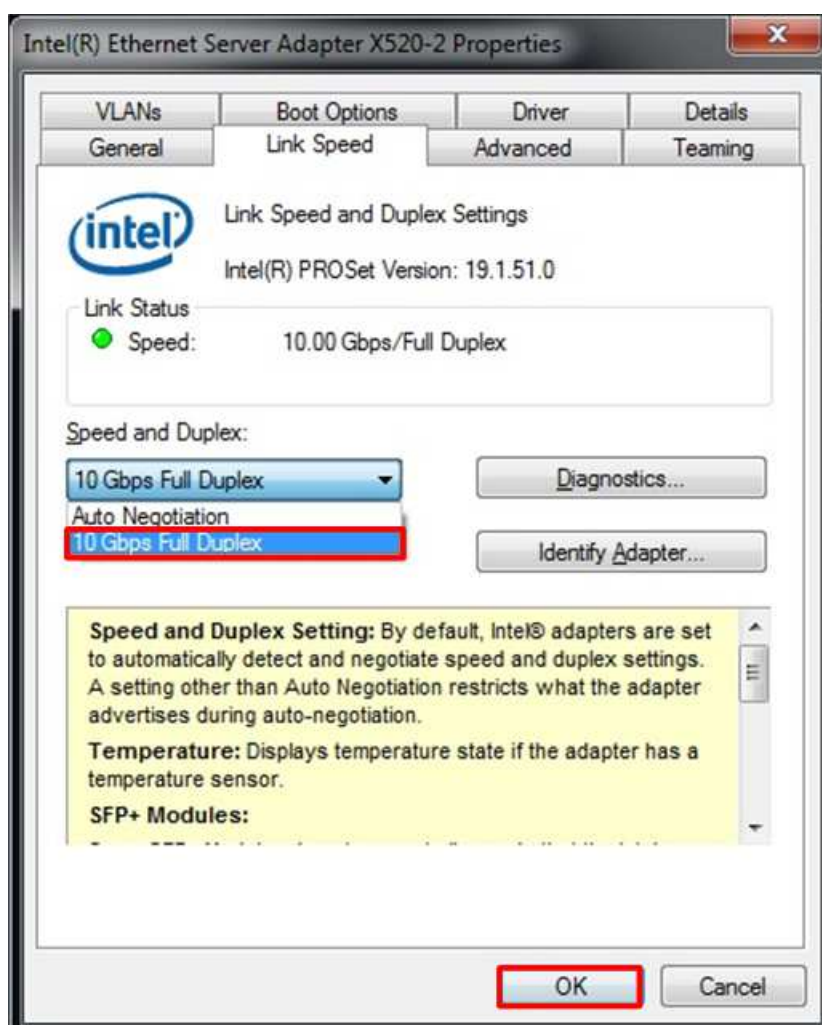


Figure 3-3 Set link speed = 10 Gbps

- 4) On Advanced Tab, select “Performance Options” and click “Properties” button.
- 5) On “Performance Options” window, select “Low Latency Interrupts” and click “Properties” button.
- 6) On “Low Latency Interrupts” window, select “Use Low Latency Interrupts” and click “OK” button.
- 7) Click “OK” button to save and exit all setting windows.

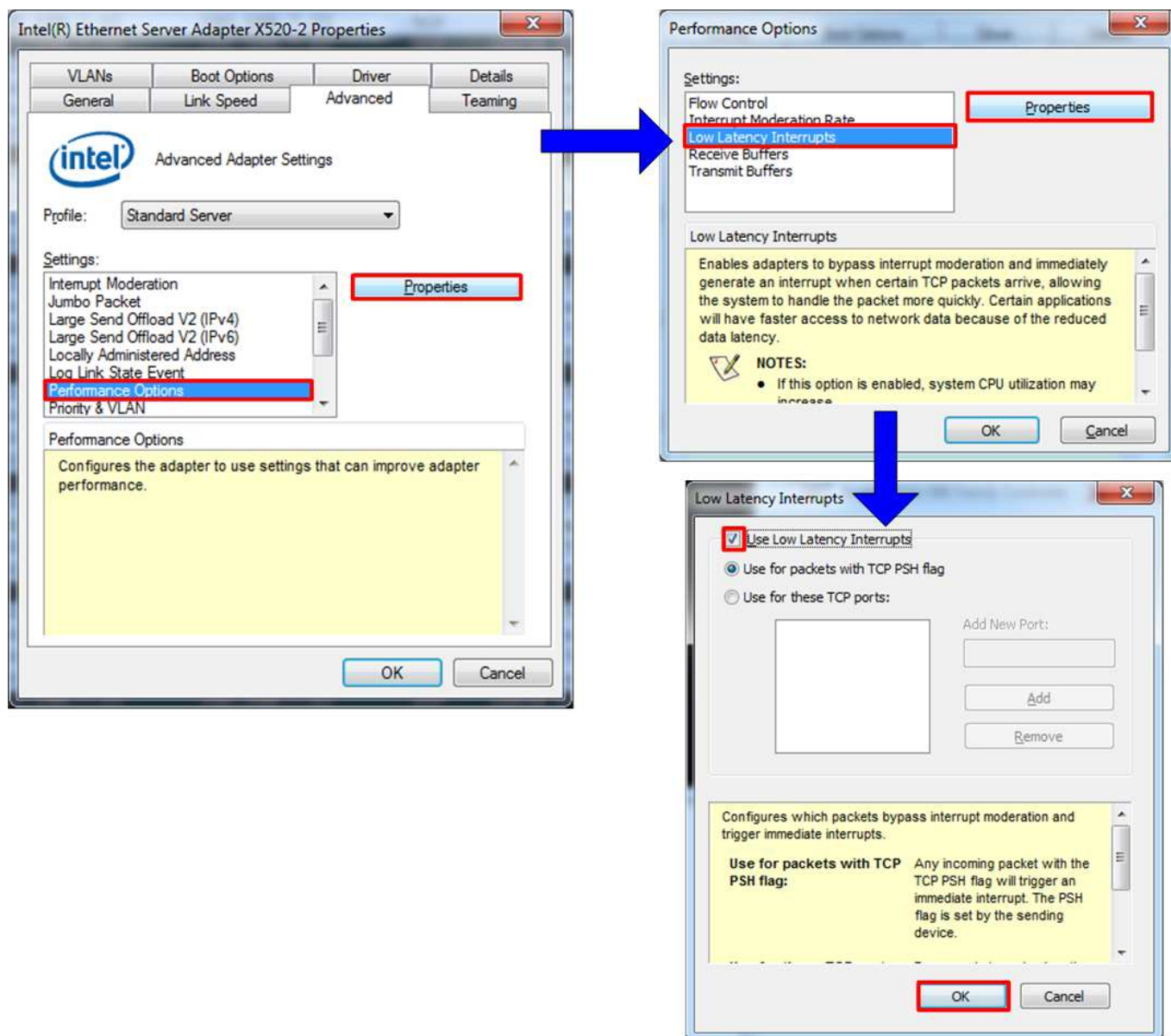


Figure 3-4 Performance option

3.3 Power Option Setting

- 1) Open Control Panel and select Power Options as shown in the left window of Figure 3-5.
- 2) Change setting to High Performance as shown in the right window of Figure 3-5.

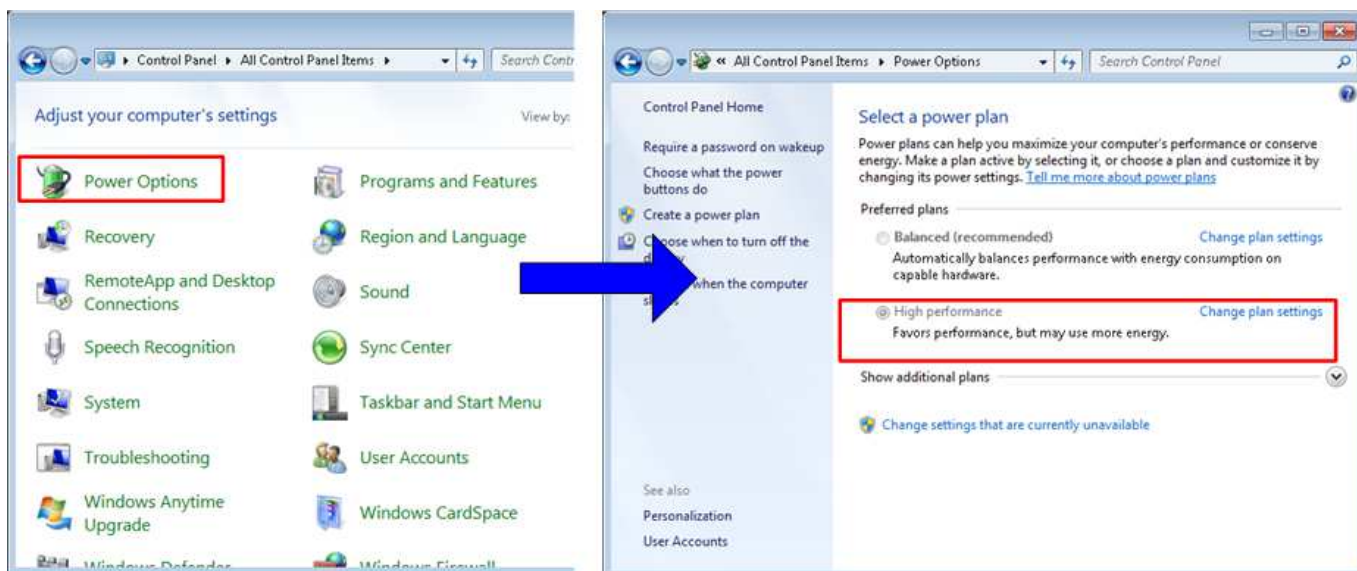


Figure 3-5 Power options

3.4 Firewall Setting

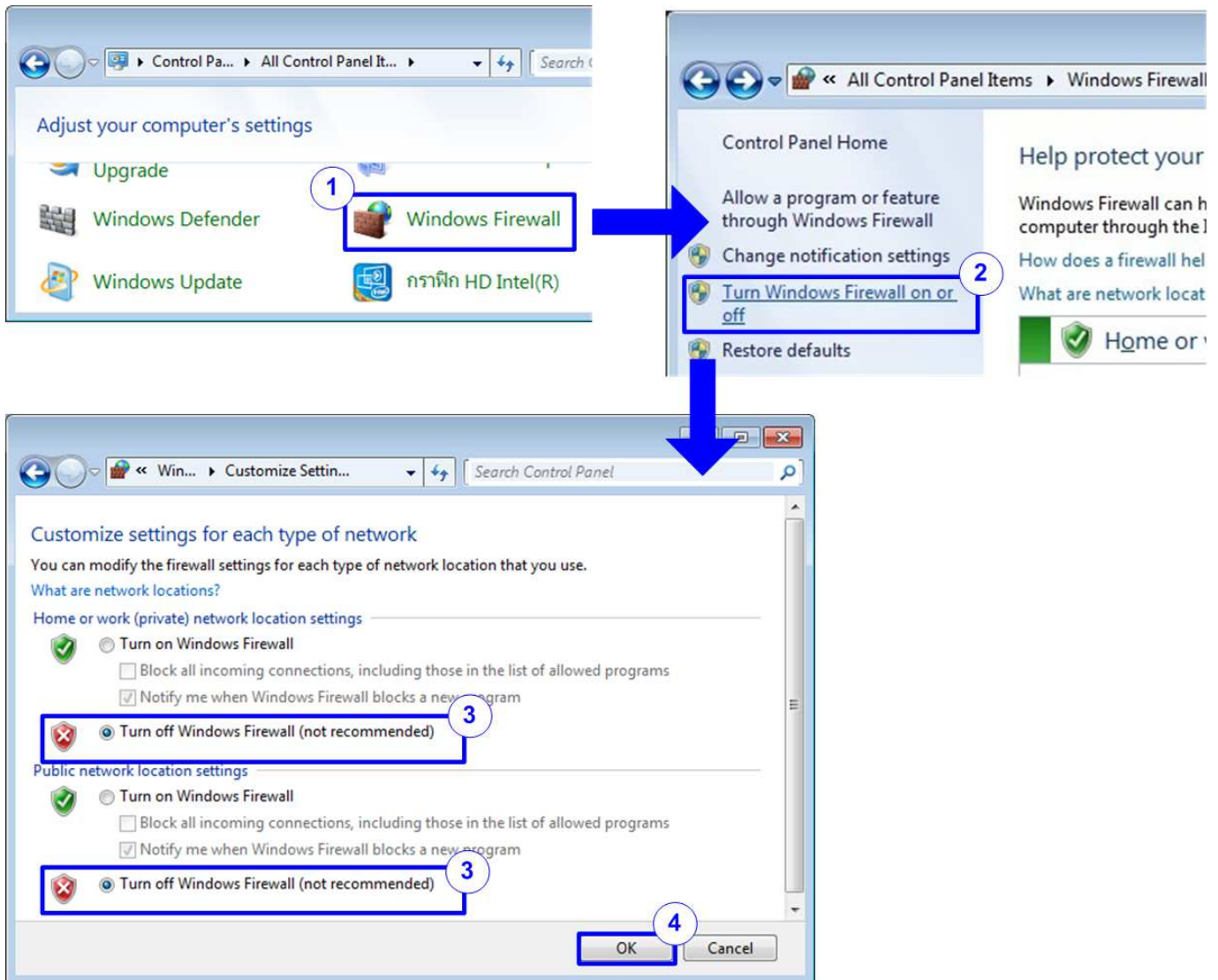


Figure 3-6 Firewall setting

- 1) Open Control Panel and select Windows Firewall.
- 2) Click “Turn Windows Firewall on or off”.
- 3) Select Turn off Firewall under Private and Public network settings.
- 4) Click OK button to confirm the setting.

4 FPGA board setup

- 1) Check DIPSW and jumper setting on FPGA board
 - a) Board setting on ZCU102 board is shown in Figure 4-1.
 - Insert jumper to J16 to enable Tx SFP+
 - Set SW6=all ONs to use USB-JTAG.

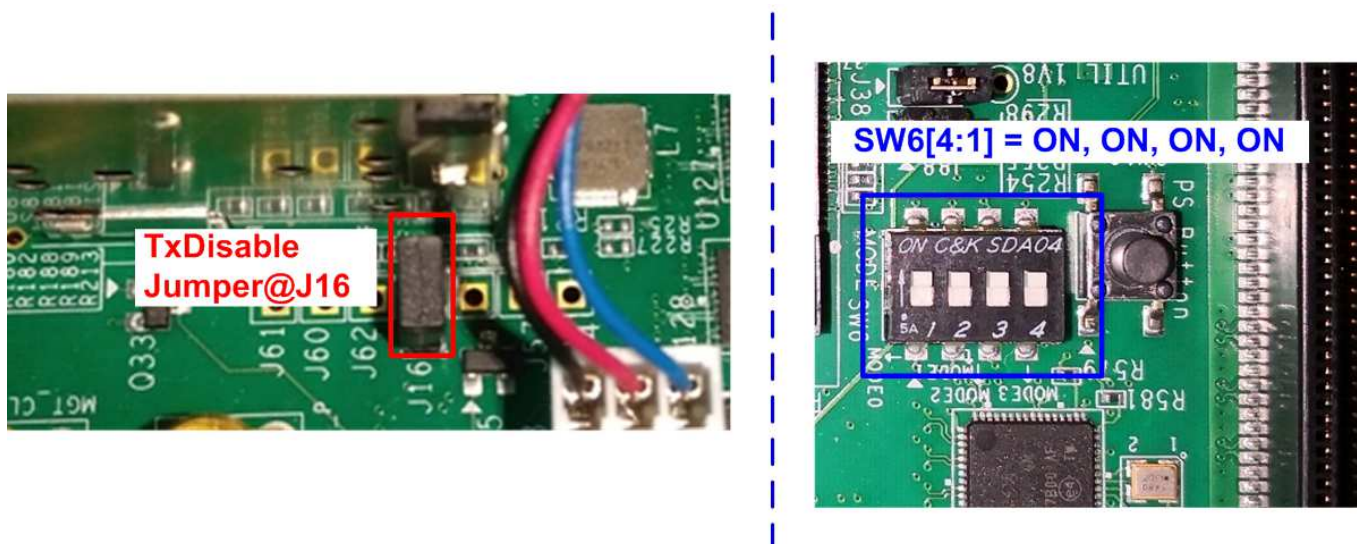


Figure 4-1 ZCU102 board setting

- b) Board setting on KCU105 board is shown in Figure 4-2. Insert jumper to J6 to enable Tx SFP+.

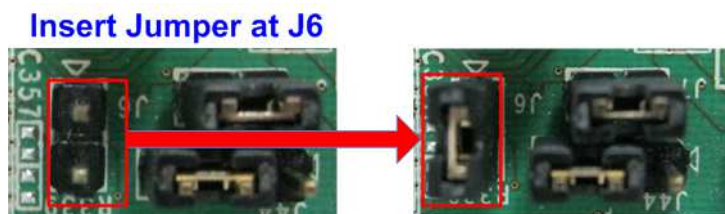


Figure 4-2 Insert jumper to enable SFP+ on KCU105

- 2) Connect micro USB cable from FPGA board to PC for JTAG programming.
- 3) Connect micro USB cable from FPGA board to PC for USB UART.
- 4) Connect power supply to FPGA development board.

- 5) Insert 10-Gigabit SFP+ DAC or SFP+ transceiver with optical cable between FPGA board and PC. Figure 4-3 shows active channel to run the demo on ZCU102 and KCU105.

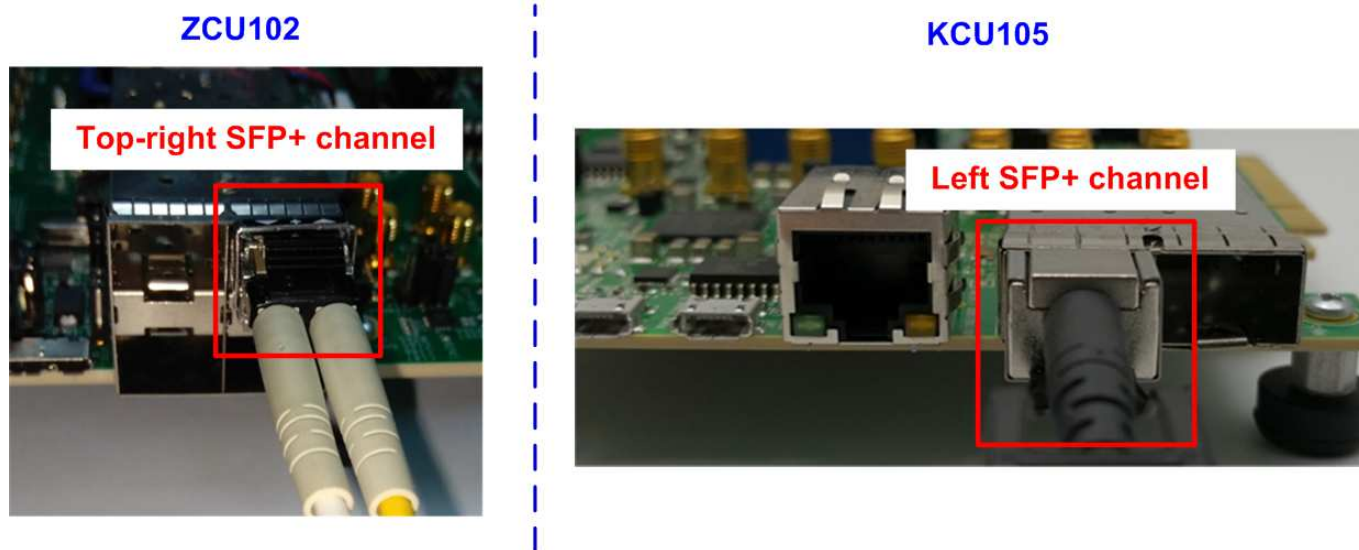


Figure 4-3 SFP+ channel using on ZCU102/KCU105 board

- 6) Power on FPGA board.
- 7) Open Serial console. When connecting FPGA board to PC, many COM ports from FPGA connection are detected and displayed on Device Manager. In case of KCU105, select Standard COM port. In case of ZCU102, select COM port of Interface0 (COM15 in right side of Figure 4-4) for Serial console. On Serial console, use following setting: Baud rate=115,200, Data=8 bit, Non-Parity, and Stop = 1.

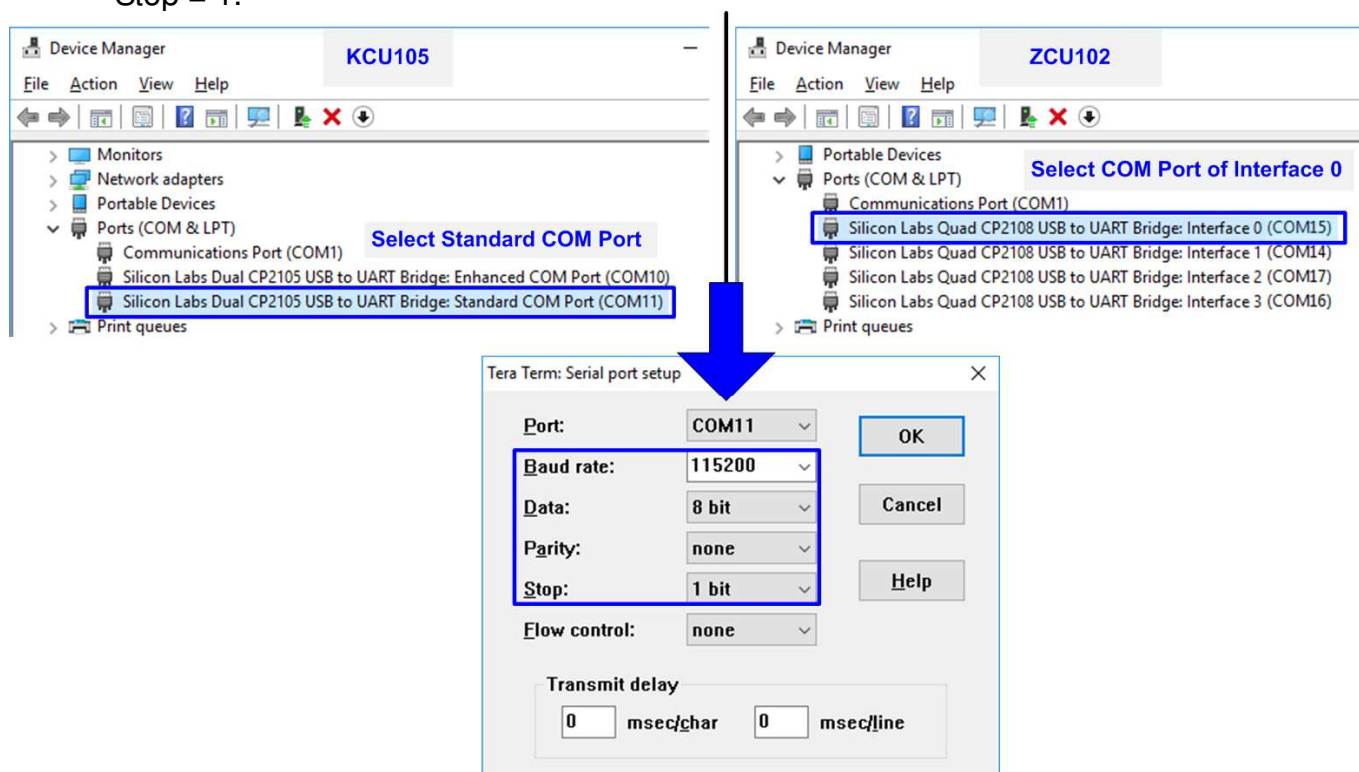


Figure 4-4 COM port number for Serial console

- 8) Download configuration file and firmware to FPGA board
 - a) For ZCU102 board, open Vivado TCL shell and change current directory to download folder which includes demo configuration file and script file (udp10gtest_zcu102.bat). Type “udp10gtest_zcu102.bat”, as shown in Figure 4-5.



Figure 4-5 Example command script for download to ZCU102 by Vivado tool

- b) For KCU105 board, use Vivado tool to program configuration file, as shown in Figure 4-6

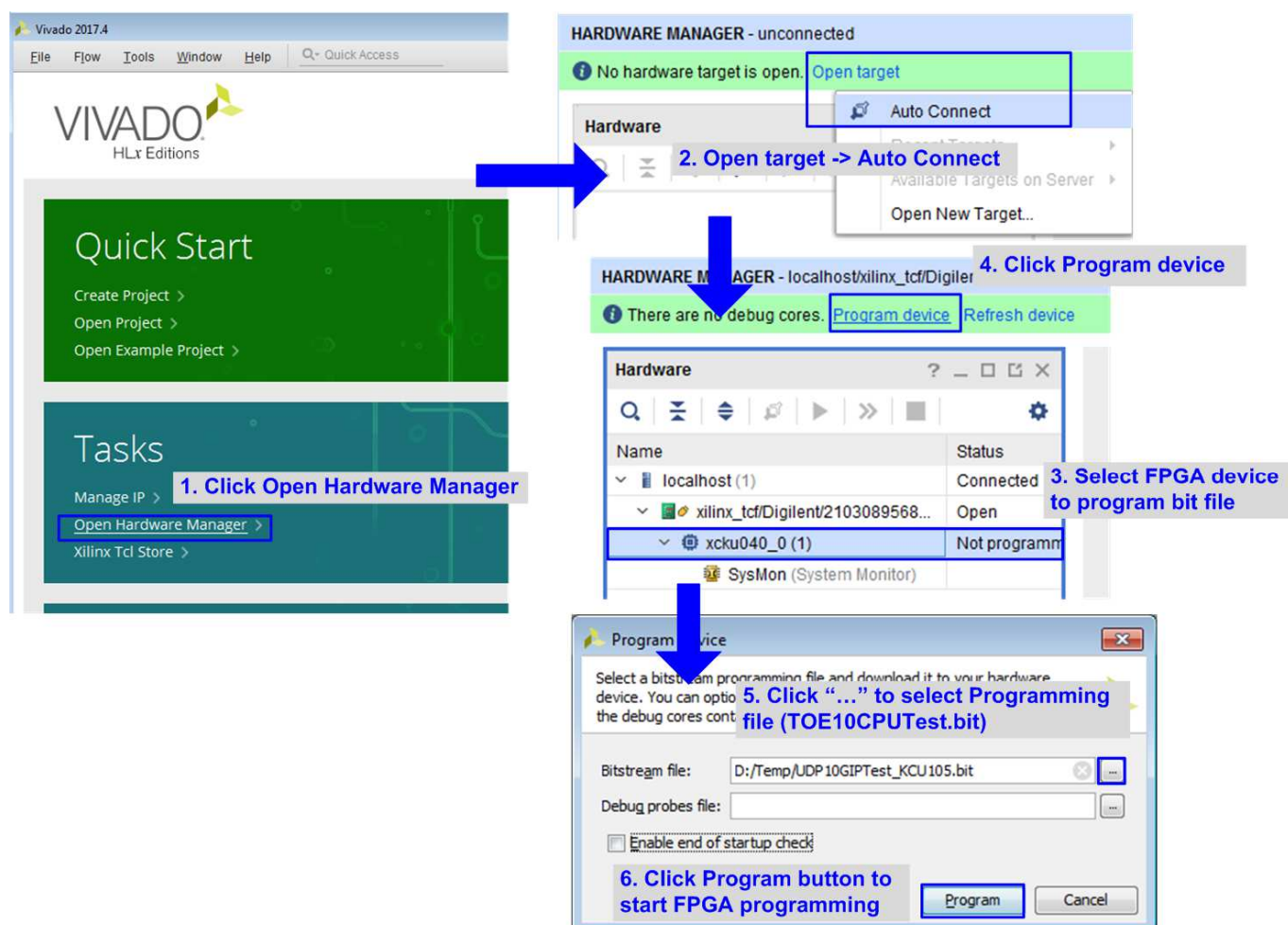


Figure 4-6 Program FPGA by Vivado

- 9) Input '0' to initialize UDP10G-IP in client mode (ask PC MAC address by sending ARP request).
- 10) Default parameter in client mode is displayed on the console.

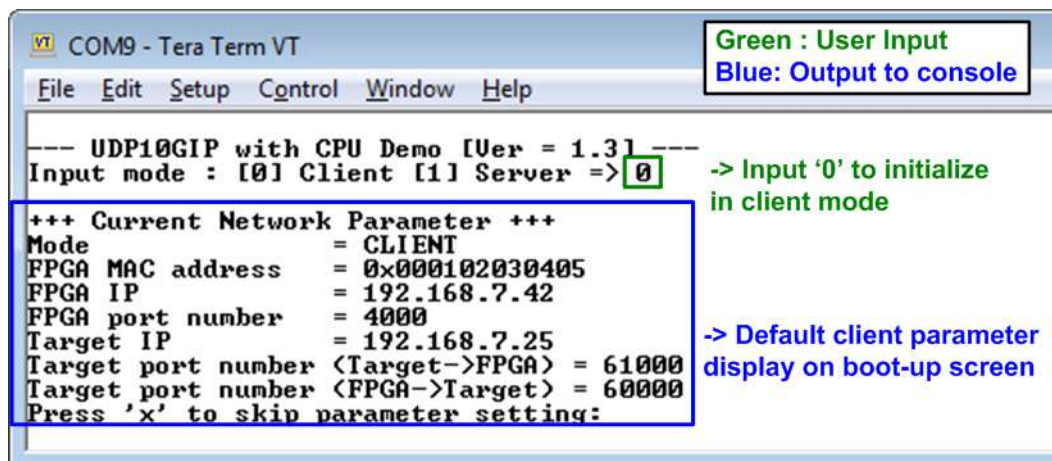


Figure 4-7 Message after system boot-up

- 11) User inputs 'x' to skip parameter setting and use default parameters for system initialization, as shown in Figure 4-8. If user inputs other keys, the menu to change parameter will be displayed. The example to change parameter is shown in topic 5.2.

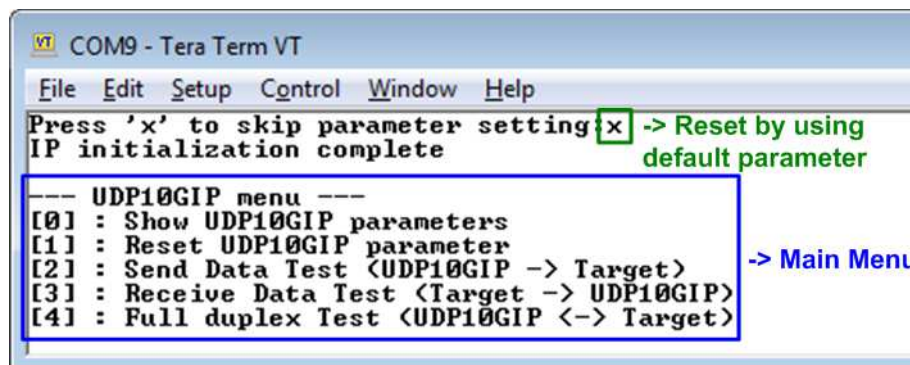


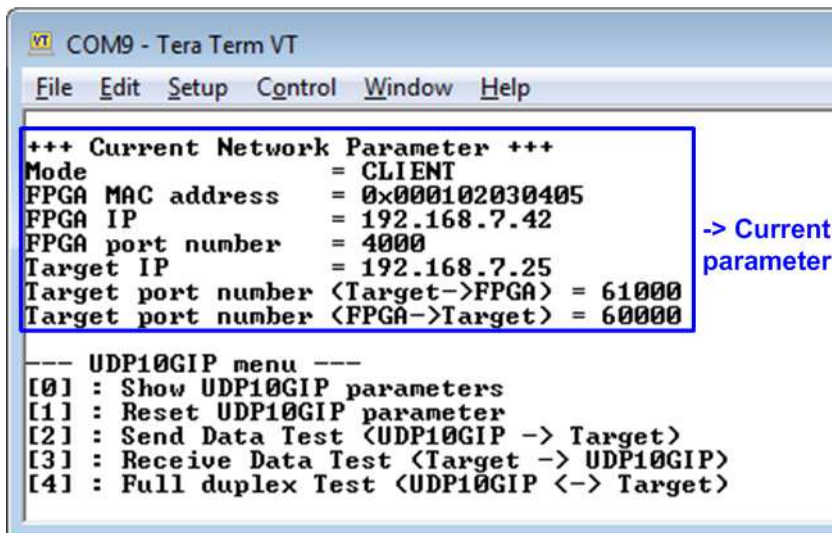
Figure 4-8 Initialization complete

Note: Transfer performance in the demo depends on Test PC resource in Test platform.

5 Main menu

5.1 Display current parameter

Select '0' to check current parameter in the demo. There are seven parameters displayed on the console.



```

COM9 - Tera Term VT
File Edit Setup Control Window Help

+++ Current Network Parameter +++
Mode = CLIENT
FPGA MAC address = 0x000102030405
FPGA IP = 192.168.7.42
FPGA port number = 4000
Target IP = 192.168.7.25
Target port number <Target->FPGA = 61000
Target port number <FPGA->Target = 60000

--- UDP10GIP menu ---
[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter
[2] : Send Data Test <UDP10GIP -> Target>
[3] : Receive Data Test <Target -> UDP10GIP>
[4] : Full duplex Test <UDP10GIP <-> Target>
  
```

Figure 5-1 Display current parameter result

- 1) Mode: Set mode to UDP10G-IP to run as server or client. To run with PC, please input '0' to initialize as client mode.
- 2) FPGA MAC address: 48-bit hex value to be MAC address of FPGA. Default value is 0x000102030405.
- 3) FPGA IP: IP address of FPGA. Default value is 192.168.7.42.
- 4) Target IP: IP address of destination device (10 Gb Ethernet on PC). Default value is 192.168.7.25.
- 5) FPGA port number: Port number of FPGA. Default value is 4000.
- 6) Target port number (Target->FPGA): Port number of the destination device to send data to FPGA. Default value is 61000.
- 7) Target port number (FPGA->Target): Port number of the destination device to receive data from FPGA. Default value is 60000.

To change some parameters, user can change by using menu [1] (Reset UDP10GIP parameter).

5.2 Reset UDP10G-IP

Select '1' to reset the IP or change IP parameters.

This menu is used to change IP parameters or send reset to UDP10G-IP. After user selects this menu, the current parameters are displayed on the console. User inputs 'x' to use same parameter set and send reset to UDP10G-IP. Other keys could be input to change some parameters and then reset UDP10G-IP.

There are seven parameters to set in this menu. After user inputs each parameter, CPU validates the value. The parameter is updated to UDP10G-IP when the input is valid. If the input is not valid, the new value will not be used. After user inputs all parameters, IP is reset. The description of each parameter is shown in topic 5.1 (Display current parameter) and the range of each parameter is described as follows.

- 1) Mode: Input '0' to initialize as client mode.
- 2) FPGA MAC address: Input 12-digit of hex value. Add "0x" as a prefix to input as hex value.
- 3) FPGA IP address: A set of four decimal digits is separated by ".". The valid range of each decimal digit is 0-255.
- 4) Target IP address: A set of four decimal digits like FPGA IP address. This value is IP address of Test PC.
- 5) FPGA port number: Valid range is 0-65535.
- 6) Target port number (Target->FPGA): Valid range is 0-65535.
- 7) Target port number (FPGA->Target): Valid range is 0-65535.

After complete to assign all parameters, new parameter set is displayed on Serial console. Next, reset signal is sent to the IP to load new parameter set. Finally, "IP initialization complete" is shown after IP completes initialization process, as shown in Figure 5-2.

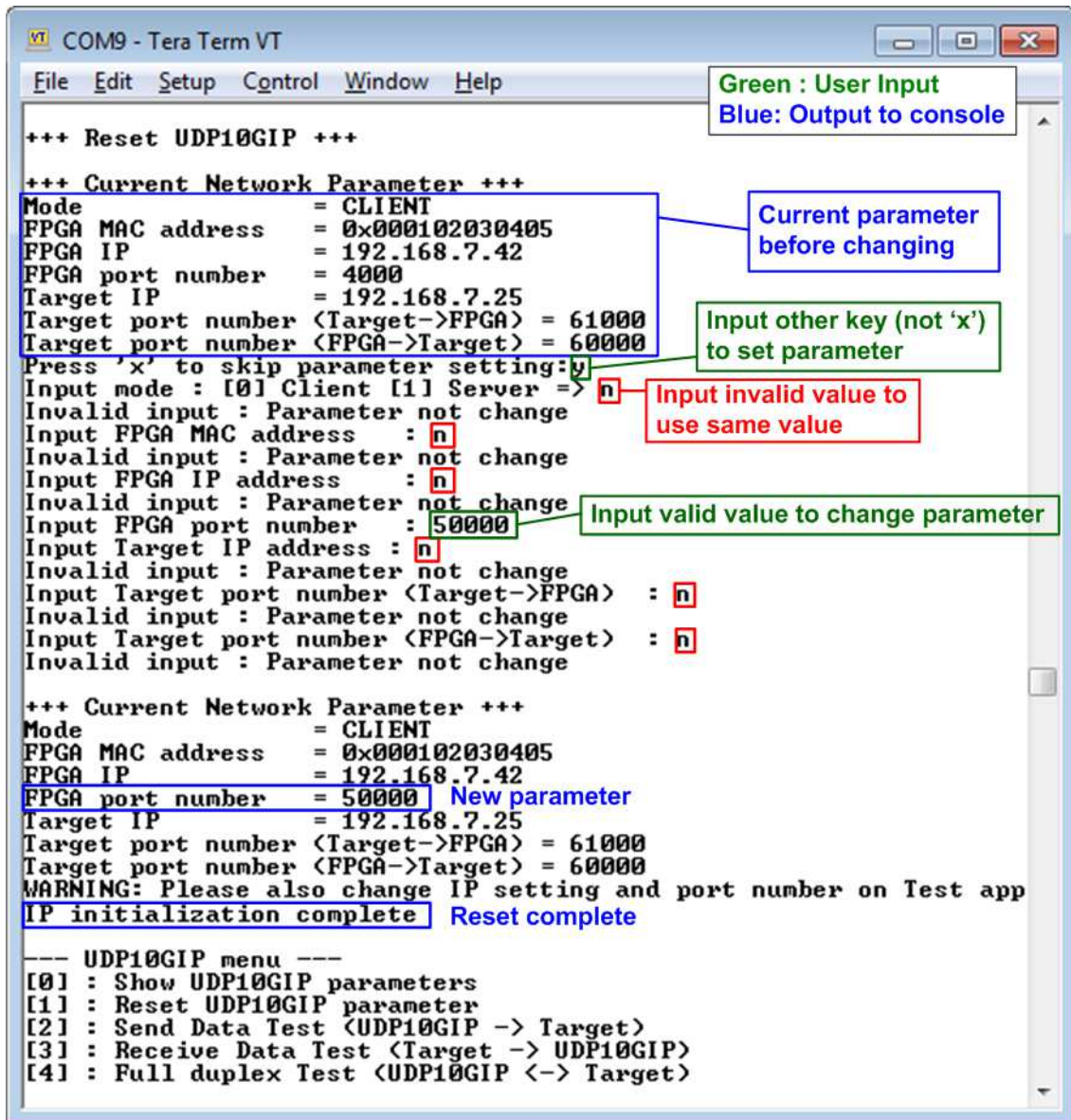


Figure 5-2 Change IP parameter result

5.3 Send Data Test

To transfer data from FPGA to PC, select '2' to run send data test on FPGA and prepare "udpdatatest.exe" on PC to receive data. User inputs test parameters on FPGA for sending data through Serial console. On PC, user inputs test parameters of "udpdatatest" to receive data through Command prompt. The sequence to run the test is shown as follows.

- 1) On Serial console, input two parameters under send data test menu.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be a prefix when input is hexadecimal unit.
 - b) Input packet size: Unit of packet size is byte. Valid value is 8 – 8968. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be a prefix when input is hexadecimal unit.
Note: If packet size is more than 1472, the packet output from UDP10G-IP will be jumbo frame. In this case, Test PC must support jumbo frame.
- 2) If inputs are valid, recommended parameters to run test application on PC will be displayed. Next, "Press any key to start data sending ..." is displayed to wait user input to start operation.
- 3) On Command prompt, input parameters following the recommended value. There are six parameters for "udpdatatest".


```
>> udpdatatest [Dir] [FPGA IP] [FPGA Port] [PC Port] [Bytelen] <Timeout>
```

 - a) Dir: Input 'r' to receive and verify test data from FPGA
 - b) FPGA IP: Input same value as FPGA IP address
 - c) FPGA port: Input same value as FPGA port number
 - d) PC port: Input same value as target port number (FPGA->Target)
 - e) Bytelen: Input same value as "Input transfer size" of step 1a)
 - f) Timeout: Timeout in msec unit. Valid value is 100-65536. To use default value (100 msec), this input is not necessary.
- 4) After running test application, summary of setting parameter is shown before starting to receive data from FPGA. PC waits received data from FPGA.
- 5) On Serial console, user inputs any key(s) to start sending data. During transferring data, current transfer size is displayed on Serial console (transmit size) and Command prompt (received size) every second.
- 6) "Send data complete" is displayed on Serial console after all data are sent. On PC, test application is completed when total data are received or timeout is found. Timeout message with the 1st error position is displayed when test application is completed by timeout. Finally, total transfer size and performance are displayed on Serial console (transmit performance) and Command prompt (received performance).

```

COM11 - Tera Term VT
File Input '2' to run send data test Help
+++ UDP10GIP Send Mode +++
Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFFF8 => 0xFFFFFFFF8
Enter packet size (aligned to 64-bit) : 8 - 8968 => 8968

Run test application on PC by following command
udpdatest r 192.168.7.42 4000 60000 4294967288

Press any key to start data sending ...
Start data sending
Send 1240 MByte Recv 0 Byte
Send 2480 MByte Recv 0 Byte
Send 3720 MByte Recv 0 Byte
Send data complete

Total tx transfer size = 536870911 QWord (64-bit)
Total = 4294[MB], Time = 3465[ms], Transfer speed = 1239[MB/s]

Total rx transfer size = 0 QWord (64-bit)
Total = 0[B], Time = 3465[ms], Transfer speed = 0[MB/s]

--- UDP10GIP menu ---
[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter
[2] : Send Data Test (UDP10GIP -> Target)
[3] : Receive Data Test (Target -> UDP10GIP)
[4] : Full duplex Test (UDP10GIP <-> Target)
    
```

Green : User Input
Blue: Output to console

Call udpdatest.exe with recommended parameters

```

Administrator: C:\Windows\system32\cmd.exe
D:\Share\udpdatest r 192.168.7.42 4000 60000 4294967288

Start Receiving with Data pattern
FPGA IP: 192.168.7.42:4000
PC Port: 60000
Timeout: 100 mSec

1.241 GB
2.483 GB
3.724 GB

Spend 3.46 Second(s) for Receiving 4.295 GByte(s)
Receiving Data Rate: 1240.24 MByte(s)/Sec
    
```

Figure 5-3 Send data test by using jumbo frame

If the input is invalid, “Out-of-range input”/“Invalid input” will be displayed and the operation will be cancelled, as shown in Figure 5-4 - Figure 5-5.

```

COM9 - Tera Term VT
File Edit Setup Control Window Help

+++ UDP10GIP Send Mode +++
Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFFF8 => 0x100000000
Out-of-range input

--- UDP10GIP menu ---
[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter
[2] : Send Data Test <UDP10GIP -> Target>
[3] : Receive Data Test <Target -> UDP10GIP>
[4] : Full duplex Test <UDP10GIP <-> Target>
    
```

Figure 5-4 Error from invalid transfer size

```

COM9 - Tera Term VT
File Edit Setup Control Window Help

+++ UDP10GIP Send Mode +++
Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFFF8 => 0xFFFFFFFF8
Enter packet size (aligned to 64-bit) : 8 - 8968 => 8969
Out-of-range input

--- UDP10GIP menu ---
[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter
[2] : Send Data Test <UDP10GIP -> Target>
[3] : Receive Data Test <Target -> UDP10GIP>
[4] : Full duplex Test <UDP10GIP <-> Target>
    
```

Figure 5-5 Error from invalid packet size

5.4 Receive Data Test

To transfer data from PC to FPGA, select '3' to run receive data test on FPGA and prepare "udpdatatest" on PC to send data. User inputs test parameters on FPGA for receiving data through Serial console. On PC, user inputs parameters of "udpdatatest" to send data through Command prompt. The sequence to run the test is shown as follows.

- 1) On Serial console, input two parameters under receive data test menu.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - b) Input data verification mode: Set '0' to disable data verification or '1' to enable data verification sent from PC.
- 2) If inputs are valid, recommended parameters to run test application on PC will be displayed. Then, "Wait data from Target ..." is displayed to show that IP now is ready to receive data from PC.
- 3) On Command prompt, input parameters following the recommended value. There are six parameters for "udpdatatest".


```
>> udpdatatest [Dir] [FPGA IP] [FPGA Port] [PC Port] [Bytelen] <Timeout>
```

 - a) Dir: Input 't' to send test data from FPGA
 - b) FPGA IP: Input same value as FPGA IP address
 - c) FPGA port: Input same value as FPGA port number
 - d) PC port: Input same value as target port number (Target->FPGA)
 - e) Bytelen: Input same value as "Input transfer size" of step 1a)
 - f) Timeout: Timeout in msec unit. Valid value is 100-65536. This input is not used in transmit mode.
- 4) After running test application, test application starts to send data out to FPGA. During transferring data between FPGA and PC, current transfer size is displayed on both Serial console (received size) and Command prompt (transmit size) every second.
- 5) "Receive data completed" is displayed on Serial console after FPGA receives all data or timeout is found. Finally, total transfer size and performance are displayed on Serial console (received performance) and Command prompt (transmit performance).

Figure 5-6 shows the example of receive data test when disable data verification mode on FPGA. The left window is test result on Serial console while the right window is test result on Command prompt.

Figure 5-7 shows the example of receive data test when enable data verification mode on FPGA. PC sends increment data to FPGA. The left window is test result on Serial console while the right window is test result on Command prompt. If verification module is failed, error message will be displayed.

COM11 - Tera Term VT

File **Input '3' to run receive data test** W Help

```

+++ UDP10GIP Receive Mode +++
Enter transfer size (aligned to 64-bit): 8 - 0xFFFFFFFF8 => 0xFFFFFFFF8
Enable data verification      : [0] Disable [1] Enable => 0
  
```

Input parameter for client mode 1

Run test application on PC by following command
 udptest t 192.168.7.42 4000 61000 4294967288

Wait data from Target ...

```

Send 0 Byte Recv 1225 MByte
Send 0 Byte Recv 2323 MByte
Send 0 Byte Recv 3423 MByte
Send 0 Byte Recv 4294 MByte
  
```

Display recommended parameter and wait data from PC 2

Receive data completed

```

Total tx transfer size = 0 QWord (64-bit)
Total = 0[B], Time = 3702[ns], Transfer speed = 0[MB/s]

Total rx transfer size = 536870911 QWord (64-bit)
Total = 4294[MB], Time = 3702[ns], Transfer speed = 1160[MB/s]
  
```

Display transfer size every second 4

```

1.226 GB
2.450 GB
3.673 GB
  
```

Spend 3.51 Second(s) for Sending 4.295 GByte(s)
 Sending Data Rate: 1223.29 MByte(s)/Sec

Display performance after complete to transfer all data 5

--- UDP10GIP menu ---
 [0] : Show UDP10GIP parameters
 [1] : Reset UDP10GIP parameter
 [2] : Send Data Test (UDP10GIP -> Target)
 [3] : Receive Data Test (Target -> UDP10GIP)
 [4] : Full duplex Test (UDP10GIP <-> Target)

Figure 5-6 Receive data test without data verification

COM11 - Tera Term VT

File Edit Setup Control Window Help

```

+++ UDP10GIP Receive Mode +++
Enter transfer size (aligned to 64-bit): 8 - 0xFFFFFFFF8 => 0xFFFFFFFF8
Enable data verification      : [0] Disable [1] Enable => 1
  
```

Input parameter to enable data verification 1

Run test application on PC by following command
 udptest t 192.168.7.42 4000 61000 4294967288

Wait data from Target ...

```

Send 0 Byte Recv 1185 MByte
Send 0 Byte Recv 2257 MByte
Send 0 Byte Recv 3312 MByte
Send 0 Byte Recv 4294 MByte
  
```

Receive data completed

```

Total tx transfer size = 0 QWord (64-bit)
Total = 0[B], Time = 3800[ns], Transfer speed = 0[MB/s]

Total rx transfer size = 536870911 QWord (64-bit)
Total = 4294[MB], Time = 3800[ns], Transfer speed = 1130[MB/s]
  
```

Display performance after complete to transfer all data 5

--- UDP10GIP menu ---
 [0] : Show UDP10GIP parameters
 [1] : Reset UDP10GIP parameter
 [2] : Send Data Test (UDP10GIP -> Target)
 [3] : Receive Data Test (Target -> UDP10GIP)
 [4] : Full duplex Test (UDP10GIP <-> Target)

Administrator: C:\Windows\System32\cmd.exe

```

D:\Share>udptest t 192.168.7.42 4000 61000 4294967288

Start Sending with Data pattern
FPGA IP: 192.168.7.42:4000
PC Port: 61000
Timeout: 100 mSec
1.186 GB
2.378 GB
3.551 GB

Spend 3.63 Second(s) for Sending 4.295 GByte(s)
Sending Data Rate: 1181.88 MByte(s)/Sec
  
```

Display performance after complete to transfer all data 5

Figure 5-7 Receive data test when enable data verification

5.5 Full-duplex Test

Select '4' to run full-duplex test to transfer data between FPGA and PC in both directions at the same time. User inputs test parameters on FPGA through Serial console and inputs test parameters on PC through Command prompt. Two "udpdatabtest" are called through two Command prompts to send data and receive data by using different port number. The sequence to run the test is shown as below.

- 1) On Serial console, input three parameters under full-duplex test.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - b) Input packet size: Unit of packet size is byte. Valid value is 8 – 8968. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - c) Input data verification mode: Set '0' to disable data verification or '1' to enable data verification sent from PC.
- 2) If inputs are valid, two recommended parameter sets will be displayed. First parameter is for sending data test and another parameter is for receiving data test. Next, "Press any key to start data sending ..." is displayed to wait user input to start operation.
- 3) On PC, open two Command prompts and input the parameter following recommended value. There are six parameters for "udpdatabtest".


```
>> udpdatabtest [Dir] [FPGA IP] [FPGA Port] [PC Port] [Bytelen] <Timeout>
```

 - a) Dir: Input 'r' to the 1st Command prompt.
Input 't' to the 2nd Command prompt.
 - b) FPGA IP: Input same value as FPGA IP address
 - c) FPGA port: Input same value as FPGA port number
 - d) PC port: Input same value as target port number (FPGA->Target) for the 1st Command prompt. Input same value as target port number (Target->FPGA) for the 2nd Command prompt.
 - e) Bytelen: Input same value as "Input transfer size" of step 1a)
 - f) Timeout: Timeout in msec unit. Valid value is 100-65536. To use default value (100 msec), this input is not necessary.
- 4) On Serial console, user inputs any key(s) to start sending data to PC. Current transfer size in both directions is displayed on Serial console and two Command prompts every second.
- 5) "Transfer data complete" is displayed on Serial console after UDP10G-IP completes to send and receive all data. Finally, total transfer size and performance are displayed on Serial console and Command prompt.

As shown in Figure 5-8, transfer performance when running full-duplex with data verification is displayed. The left window is the test result on Serial console while the right window is the test result on Command prompt (upper window is receive performance and lower window is transmit performance).

Figure 5-9 shows the example result when lost data is found on PC. In this condition, the received software is stopped by timeout condition (no new received data until timeout value). The 1st lost position and total lost data are displayed as error message on the software.

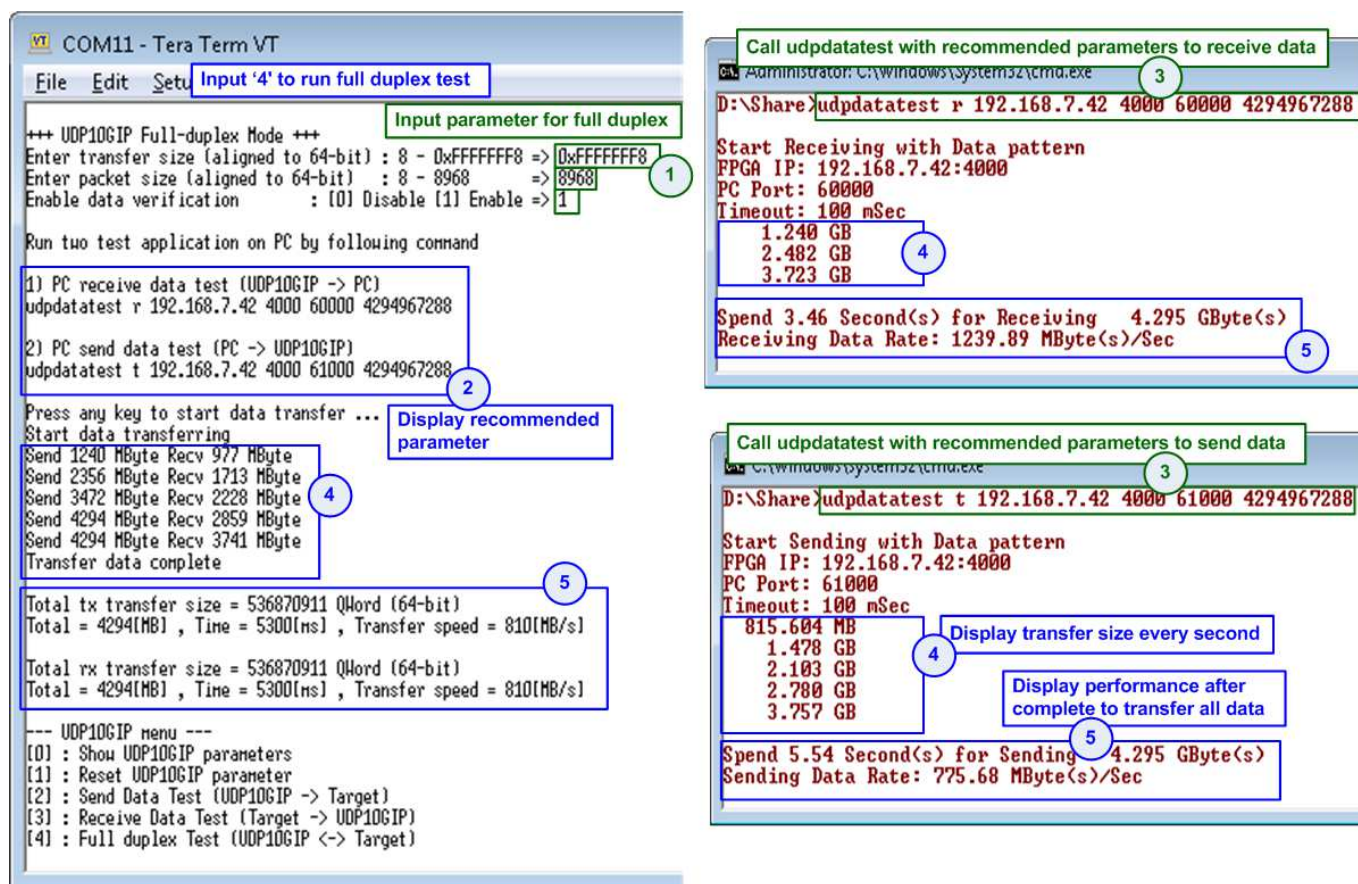


Figure 5-8 Full-duplex test when no lost data

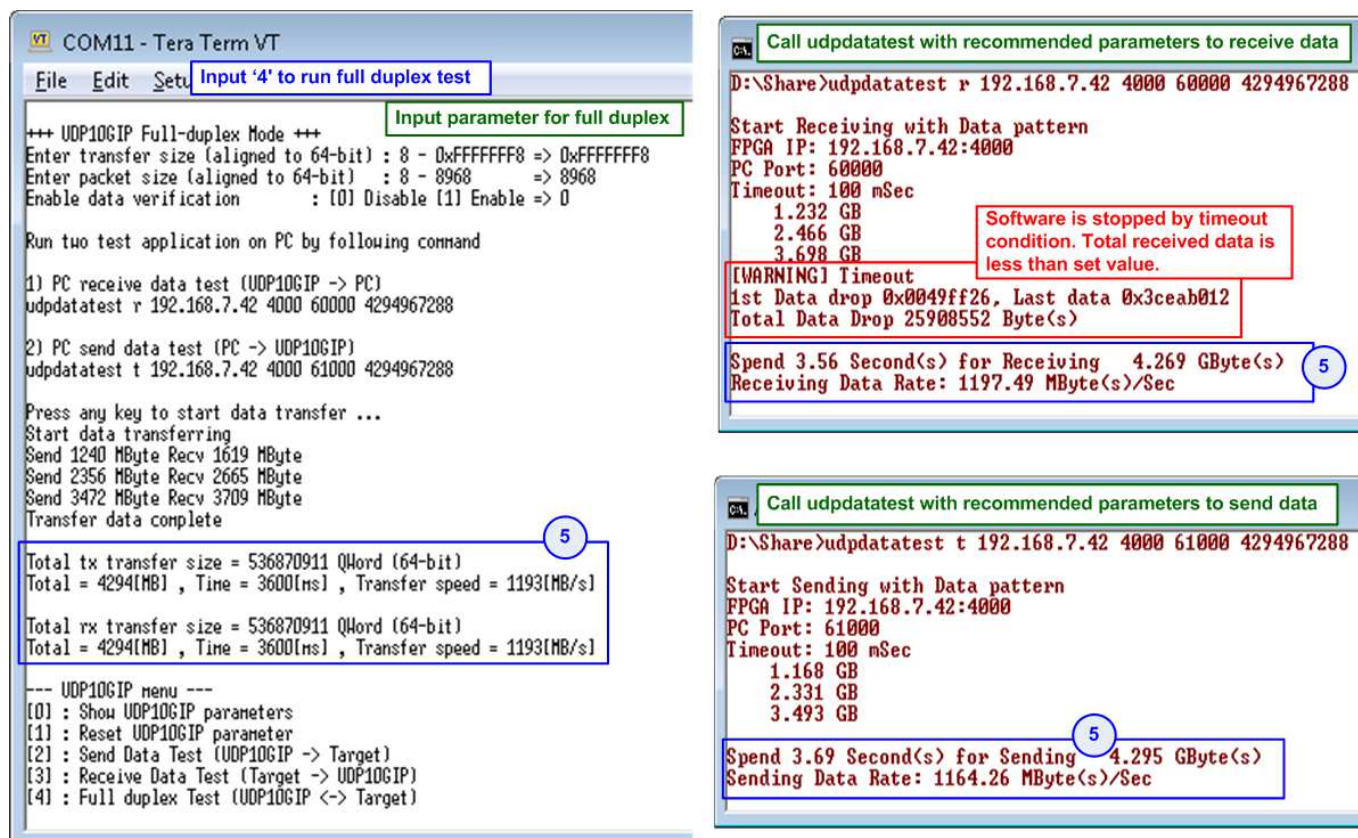


Figure 5-9 Full-duplex test when some lost data are found on PC

Part B UDP10G-IP demo by using two FPGAs

6 Environment Setup

To run UDP10G-IP by using two FPGAs as shown in Figure 6-1, please prepare following test environment.

- 1) Two FPGA development boards (ZCU102/KCU105)
Note: In test environment, two FPGA boards could be same or different board. Figure 6-1 shows the example demo by using ZCU102 and KCU105 board.
- 2) 10 Gb SFP+ copper cable (DAC) or 2x10 Gb SFP+ transceiver (10G BASE-R) with optical for network connection between two FPGA boards
- 3) micro USB cables for programming FPGA, connecting between FPGA board and PC
- 4) micro USB cable for Serial console, connecting between FPGA board and PC
- 5) Serial console software such as HyperTerminal (Baudrate=115,200 Data=8 bit Non-parity Stop=1), installed on PC
- 6) Vivado tool for programming FPGA, installed on PC

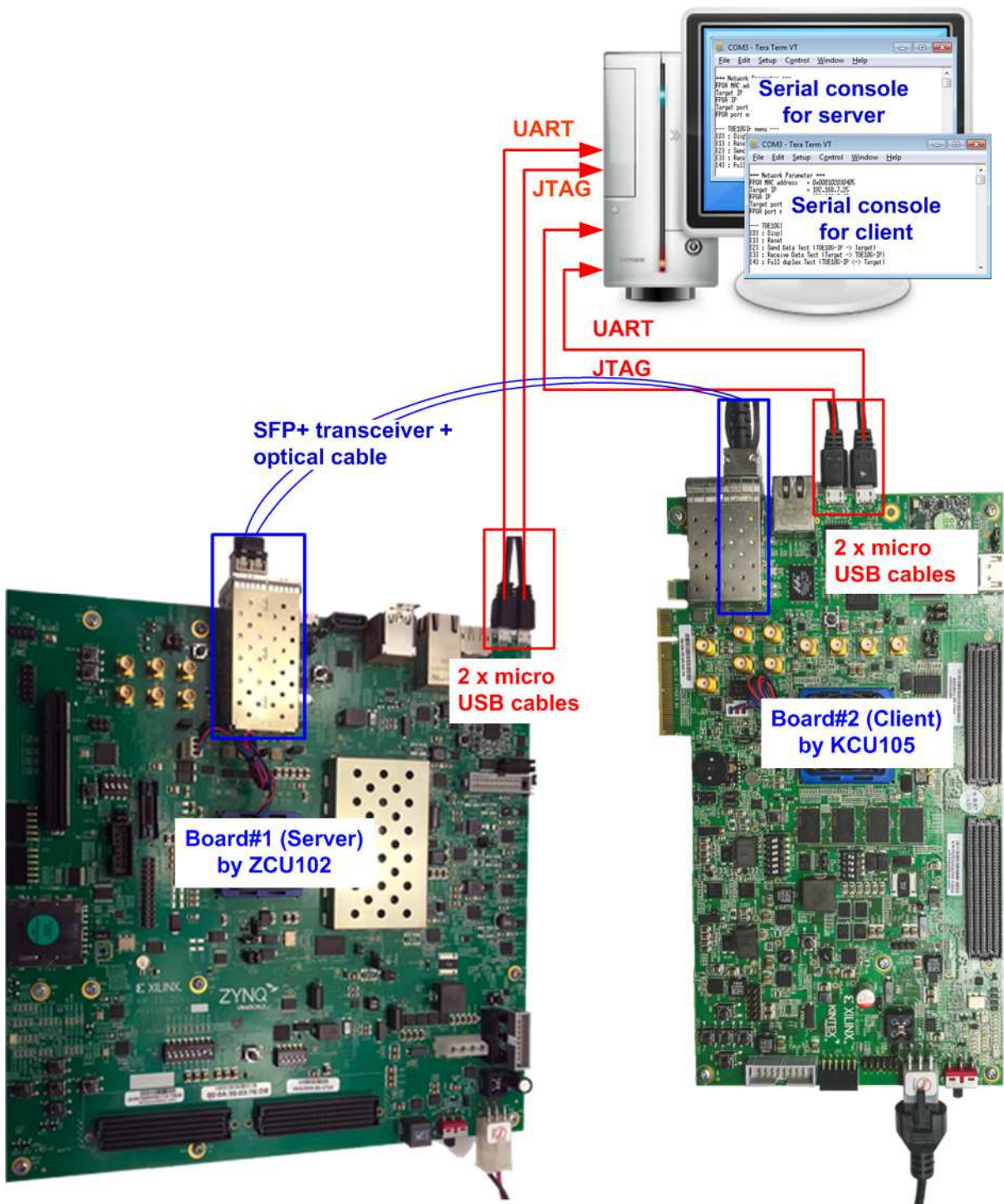


Figure 6-1 UDP10G-IP demo (FPGA<->FPGA) by ZCU102 and KCU105

7 FPGA board setup

Please follow topic 4 FPGA board setup to prepare FPGA board and SFP+ connection for running the demo. After two FPGA boards have been configured completely, Serial console displays the menu to select client mode or server mode. The step after FPGA configuration is described as follows.

- 1) Default parameters for server or client are displayed on Serial console, following setting mode. Please setup parameters of server mode before client mode.
 - a. For server mode, if user does not change default parameters, input 'x' to skip parameter setting, as shown in Figure 7-1.

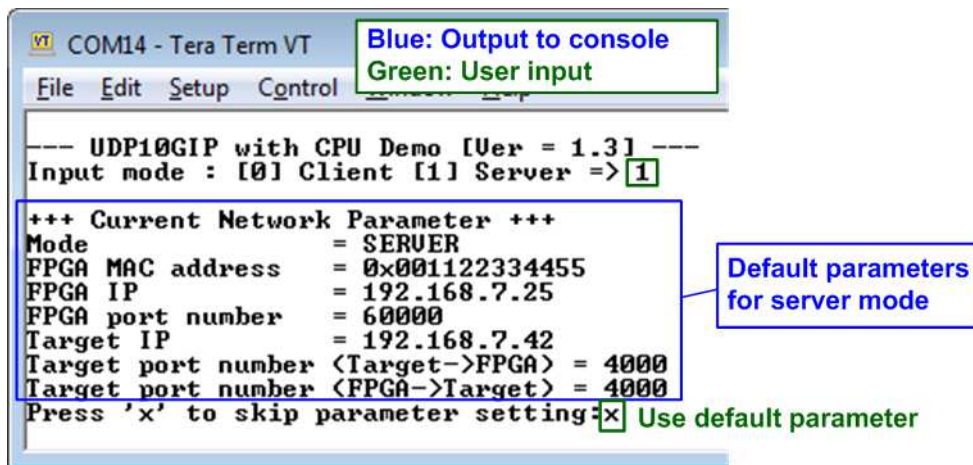


Figure 7-1 Display default parameter in server mode

- b. For client mode, user must change target port number (Target->FPGA) to use same value as target port number (FPGA->Target), as shown in Figure 7-2.

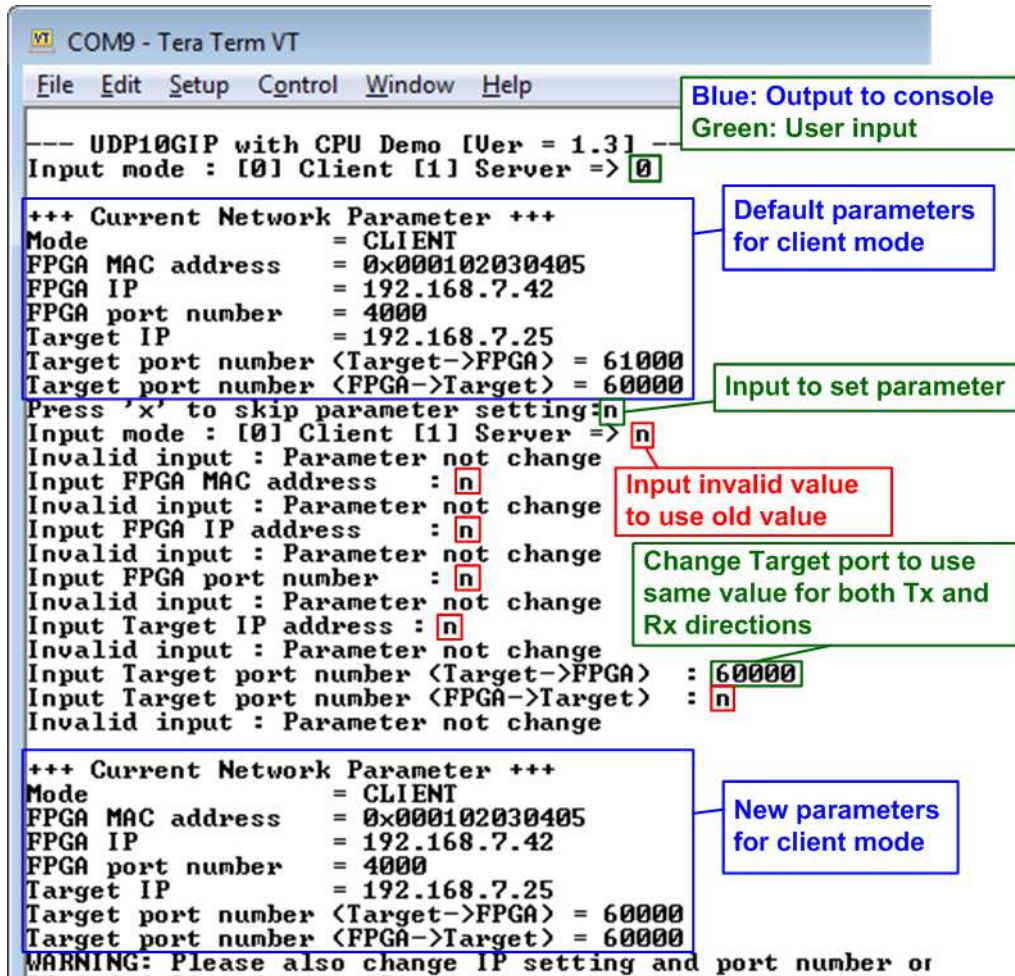


Figure 7-2 Modify default parameter in client mode

Note: Please complete to setup parameter and reset process on server before client. Server must be reset to start IP initialization by waiting ARP request from client.

- 2) After finish parameters setting, IP starts initialization process. "IP initialization complete" is displayed when finishing all initialization sequence. Finally, main menu is displayed on Serial console.

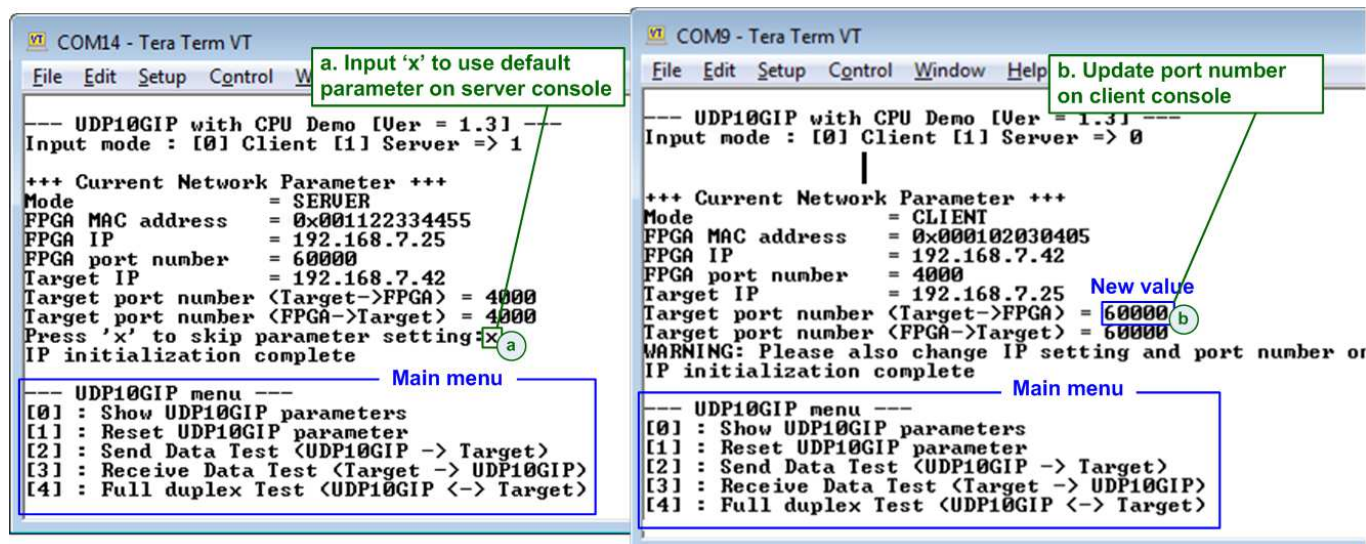


Figure 7-3 Main menu

8 Main menu

8.1 Display current parameter

Select '0' to check current parameter in the demo. There are seven parameters displayed on Serial console.

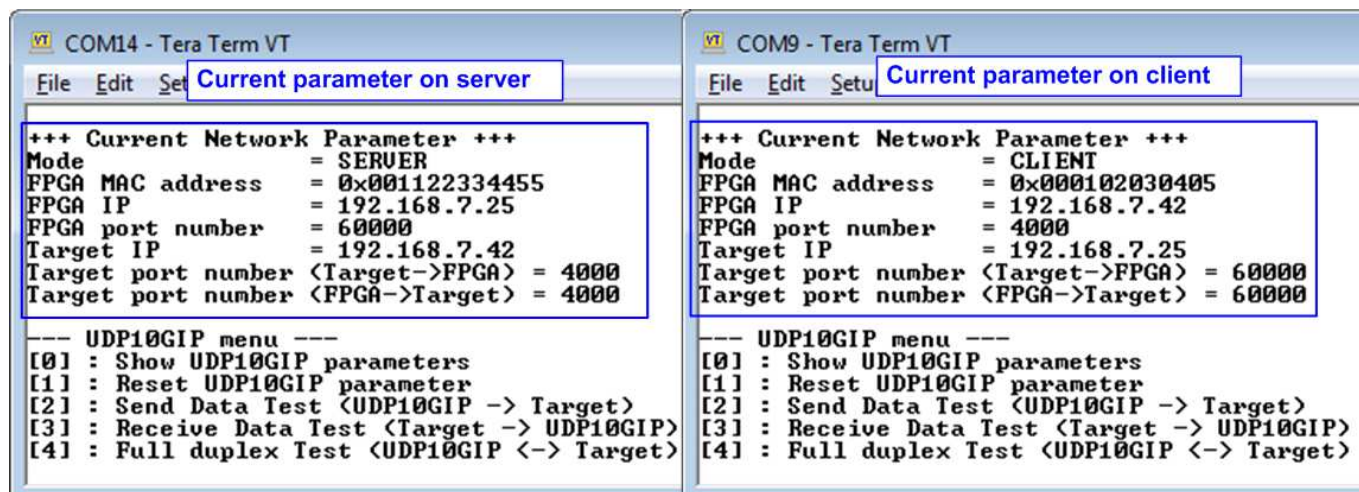


Figure 8-1 Display current parameter result

- 1) Mode: Set mode to UDP10G-IP to run as server or client. Input '0' for client and '1' for server.
- 2) FPGA MAC address: 48-bit hex value to be MAC address of FPGA. Default value is 0x000102030405 (client mode) or 0x001122334455 (server mode).
- 3) FPGA IP: IP address of FPGA. Default value is 192.168.7.42 (client mode) or 192.168.7.25 (server mode).
- 4) Target IP: IP address of destination device. Default value is 192.168.7.25 (client mode) or 192.168.7.42 (server mode).
- 5) FPGA port number: Port number of FPGA. Default value is 4000 (client mode) or 60000 (server mode).
- 6) Target port number (Target->FPGA): Port number of the destination device to send data to UDP10G-IP. Default value is 61000 (client mode) or 4000 (server mode). In client mode, this value must be changed from 61000 to 60000 (same as default value of Target port number for FPGA -> Target).
- 7) Target port number (FPGA->Target): Port number of the destination device to receive data from UDP10G-IP. Default value is 60000 (client mode) or 4000 (server mode).

To change some parameters, user can change by using menu [1] (Reset UDP10GIP parameter).

8.2 Reset UDP10G-IP

Select '1' to reset the IP and change IP parameters.

This menu is used to change IP parameters or send reset to UDP10G-IP. After user selects this menu, the current parameters are displayed on the console. User inputs 'x' to use same parameter set and send reset to UDP10G-IP. Other keys could be input to change some parameters and then reset UDP10G-IP.

There are seven parameters to set in the demo. After user inputs each parameter, CPU validates the value. The parameter is updated to UDP10G-IP when the input is valid. If the input is not valid, the new value will not be used. After user inputs all parameters, IP is reset. The description of each parameter is shown in topic 8.1 (Display current parameter) and the range of each parameter is described as follows.

Note:

1. *When user desires to reset parameters on server, the client FPGA must be also reset. Server must be reset before client to wait ARP request sent from client.*
2. *Target port number for sending and receiving data must be same value.*
3. *Parameter of client and server must be matched.*
 - a. *Target IP of server = FPGA IP of client*
 - b. *FPGA IP of server = Target IP of client*
 - c. *Target port number of server (both Tx and Rx direction) = FPGA port number of client*
 - d. *FPGA port number of server = Target port number of client (both Tx and Rx direction)*
- 1) Mode: Input '0' (client) or '1' (server) to determine FPGA initialization mode. It needs to set different mode for two FPGA boards (one board is client and another board is server).
- 2) FPGA MAC address: Input 12-digit of hex value. Add "0x" as a prefix to input as hex value.
- 3) FPGA IP address: A set of four decimal digits is separated by ".". The valid range of decimal digit is 0-255.
- 4) Target IP address: A set of four decimals like FPGA IP address.
- 5) FPGA port number: Valid range is 0-65535.
- 6) Target port number (Target->FPGA): Valid range is 0-65535.
- 7) Target port number (FPGA->Target): Valid range is 0-65535.

After complete to assign all parameters, new parameters are displayed on Serial console. Next, reset signal is sent to the IP to load new parameter set. Finally, "IP initialization complete" is shown after IP completes initialization process, as shown in Figure 8-2.

```

COM14 - Tera Term VT
File Edit Setup Control Server Help

+++ Reset UDP10GIP +++

+++ Current Network Parameter +++
Mode = SERVER
FPGA MAC address = 0x001122334455
FPGA IP = 192.168.7.25
FPGA port number = 60000
Target IP = 192.168.7.42
Target port number (Target->FPGA) = 4000
Target port number (FPGA->Target) = 4000
Press 'x' to skip parameter setting
Input mode : [0] Client [1] Server => n
Invalid input : Parameter not change
Input FPGA MAC address : n
Invalid input : Parameter not change
Input FPGA IP address : n
Invalid input : Parameter not change
Input FPGA port number : 50000
Input Target IP address : n
Invalid input : Parameter not change
Input Target port number (Target->FPGA) : n
Invalid input : Parameter not change
Input Target port number (FPGA->Target) : n
Invalid input : Parameter not change

+++ Current Network Parameter +++
Mode = SERVER
FPGA MAC address = 0x001122334455
FPGA IP = 192.168.7.25
FPGA port number = 50000
Target IP = 192.168.7.42
Target port number (Target->FPGA) = 4000
Target port number (FPGA->Target) = 4000
WARNING: Please also change IP setting and port num
IP initialization complete

--- UDP10GIP menu ---
[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter
[2] : Send Data Test (UDP10GIP -> Target)
[3] : Receive Data Test (Target -> UDP10GIP)
[4] : Full duplex Test (UDP10GIP <-> Target)

```

```

COM9 - Tera Term VT
File Edit Setup Control Client Help

+++ Reset UDP10GIP +++

+++ Current Network Parameter +++
Mode = CLIENT
FPGA MAC address = 0x000102030405
FPGA IP = 192.168.7.42
FPGA port number = 4000
Target IP = 192.168.7.25
Target port number (Target->FPGA) = 60000
Target port number (FPGA->Target) = 60000
Press 'x' to skip parameter setting
Input mode : [0] Client [1] Server => n
Invalid input : Parameter not change
Input FPGA MAC address : n
Invalid input : Parameter not change
Input FPGA IP address : n
Invalid input : Parameter not change
Input FPGA port number : n
Invalid input : Parameter not change
Input Target IP address : n
Invalid input : Parameter not change
Input Target port number (Target->FPGA) : 50000
Input Target port number (FPGA->Target) : 50000

+++ Current Network Parameter +++
Mode = CLIENT
FPGA MAC address = 0x000102030405
FPGA IP = 192.168.7.42
FPGA port number = 4000
Target IP = 192.168.7.25
Target port number (Target->FPGA) = 50000
Target port number (FPGA->Target) = 50000
WARNING: Please also change IP setting and port num
IP initialization complete

--- UDP10GIP menu ---
[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter
[2] : Send Data Test (UDP10GIP -> Target)
[3] : Receive Data Test (Target -> UDP10GIP)
[4] : Full duplex Test (UDP10GIP <-> Target)

```

Figure 8-2 Change IP parameter result

8.3 Send and Receive data Test (Half-duplex test)

The operation to send and receive data test is same for server and client mode. So, this topic describes the menu to transfer data between two FPGA boards. The 1st board runs receive data test (select menu '3') and another board runs send data test (select menu '2').

User inputs test parameters through Serial console. The sequence to run the test is shown as follows.

- 1) On Serial console under menu 3 (receive data test), input two parameters.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - b) Input data verification mode: Set '0' to disable data verification or '1' to enable data verification sent from another FPGA.
- 2) If inputs are valid, "Wait data from Target ..." will be displayed.
- 3) On Serial console under menu 2 (send data test), input two parameters.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit. This value must be same as transfer size input under receive data test (step 1a).
 - b) Input packet size: Unit of packet size is byte. Valid value is 8 – 8968. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
- 4) If all inputs are valid, "Press any key to start data sending ..." will be displayed. User inputs some key(s) to start sending data.
- 5) During transferring data, current transfer size is displayed on both Serial consoles every second.
- 6) "Send data complete" and "Receive data completed" are displayed on Serial console after complete to transfer all data. Finally, total transfer size and performance are displayed on both Serial consoles.

Figure 8-3 shows the example to transfer data between two FPGAs by using non-jumbo frame size. Left window is Serial console from FPGA running receive data test and right window is Serial console from FPGA running send data test.

Figure 8-4 shows the example of transfer data between two FPGAs by using jumbo frame size. When using jumbo frame size, performance is better than non-jumbo frame.

If the input is invalid, “Out-of-range input”/”Invalid input” will be displayed and the operation will be cancelled, as shown in Figure 5-4 - Figure 5-5 (same as FPGA<->PC test).

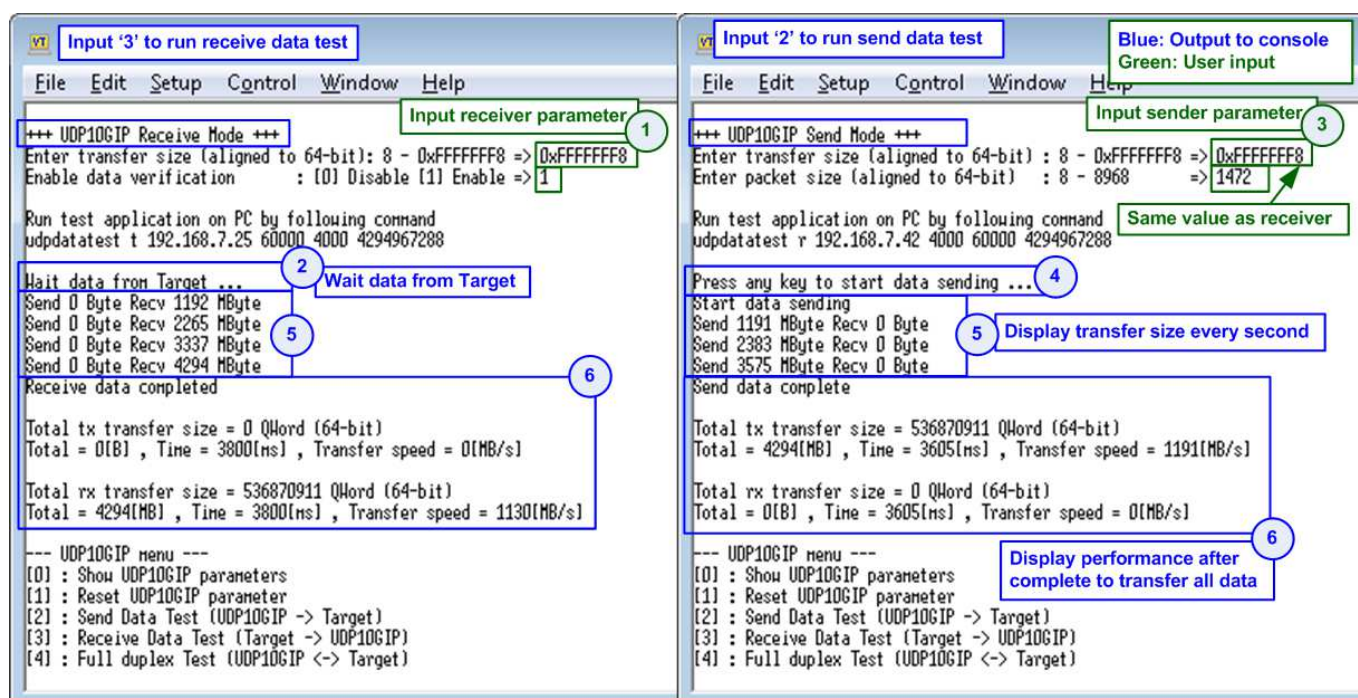


Figure 8-3 Send/receive data test by using non-jumbo frame

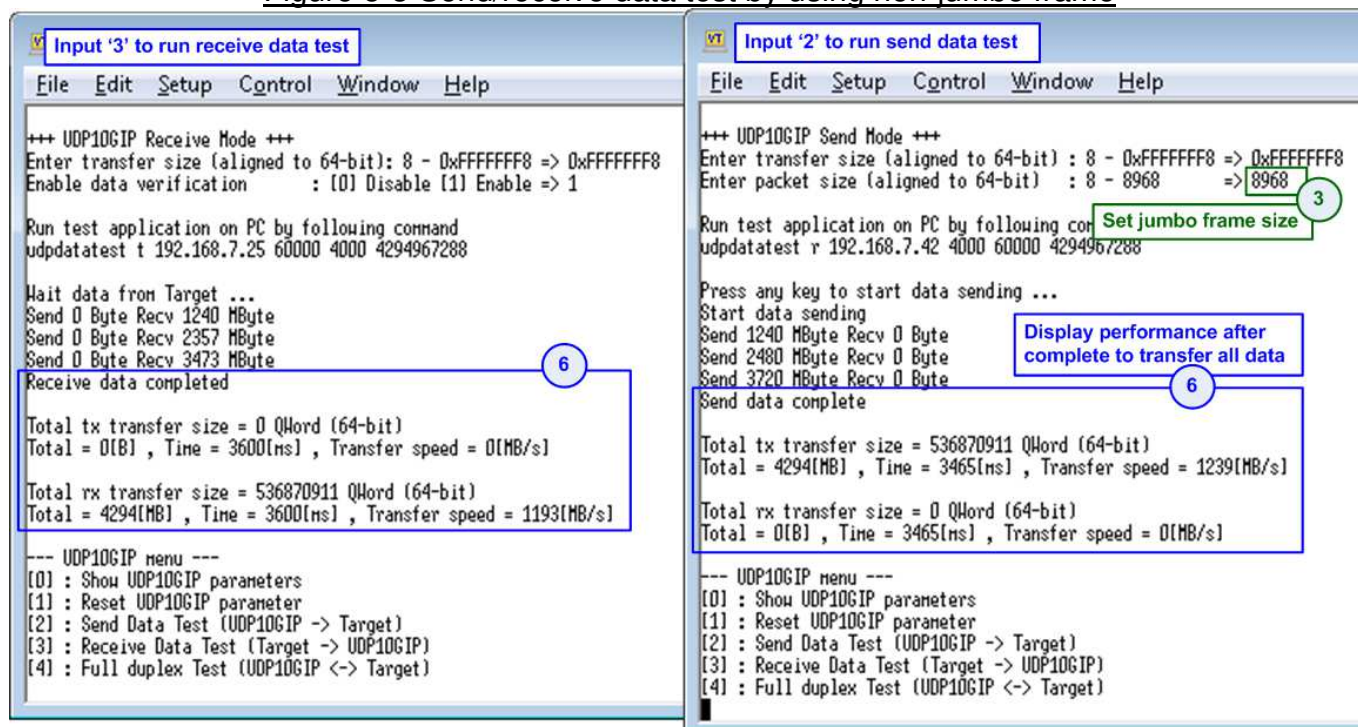


Figure 8-4 Send/receive data test by using jumbo frame

8.4 Full-duplex Test

Select '4' to run full-duplex test on two FPGAs to transfer data in both directions at the same time and same port number. User inputs test parameters through Serial console. Firmware in the demo is designed to start full-duplex by server FPGA before client FPGA. The sequence to run the test is shown as follows.

- 1) On server Serial console, input three parameters under full-duplex test.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - b) Input packet size: Unit of packet size is byte. Valid value is 8 – 8968. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - a) Input data verification mode: Set '0' to disable data verification or '1' to enable data verification sent from client FPGA.
- 2) If inputs are valid, "Wait data from Target ..." will be displayed.
- 3) On client Serial console, input three parameters under full-duplex test.
 - a) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFFFFFF8. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit. This input must be same as transfer size input of server FPGA (step 1a).
 - b) Input packet size: Unit of packet size is byte. Valid value is 8 – 8968. The value must be aligned to 8. The input is decimal unit when input only digit number. User can add "0x" to be prefix when input is hexadecimal unit.
 - c) Input data verification mode: Set '0' to disable data verification or '1' to enable data verification sent from server FPGA.
- 4) If inputs are valid, "Press any key to start data transfer .." will be displayed. User inputs some keys to start full-duplex test.
- 5) During transferring data, current transfer size is displayed on both Serial consoles every second.
- 6) "Transfer data complete" is displayed on both Serial consoles. Finally, total transfer size and performance are displayed on both Serial consoles.

Figure 8-5 and Figure 8-6 shows full-duplex test when running by using non-jumbo frame and jumbo frame size. Left window is Serial console from server FPGA and right window is Serial console from client FPGA.

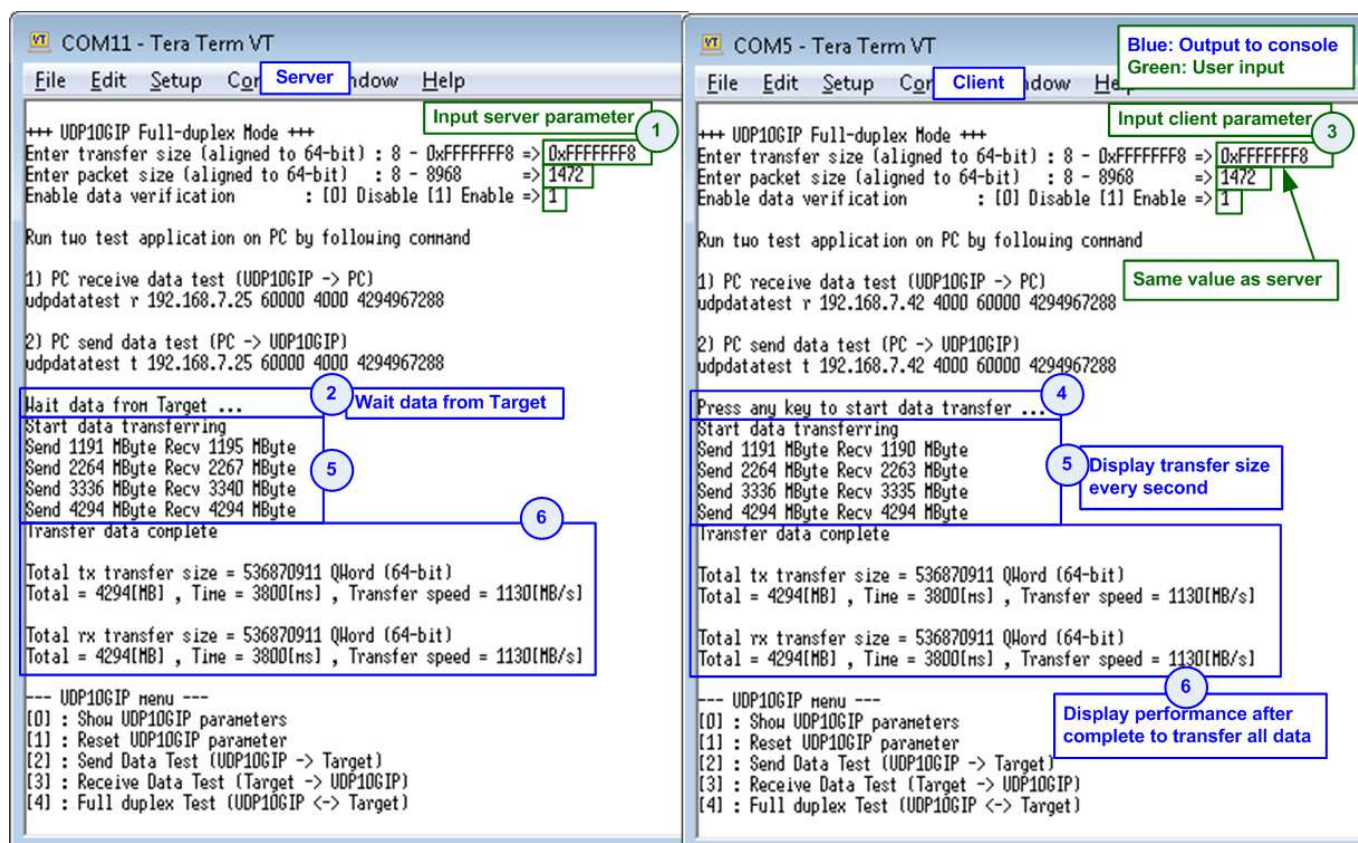


Figure 8-5 Full-duplex test when using non-jumbo frame

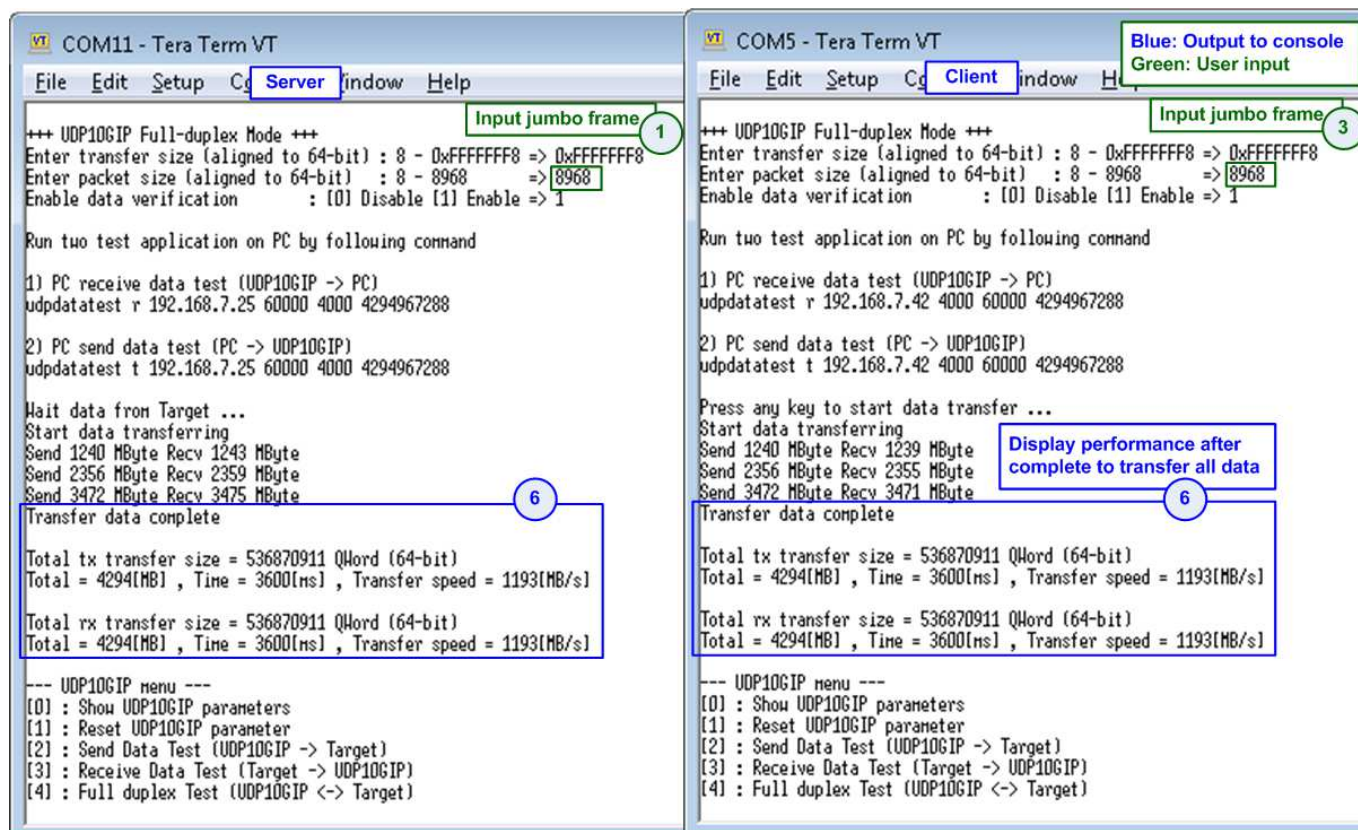


Figure 8-6 Full-duplex test when using jumbo frame

9 Revision History

Revision	Date	Description
1.0	15-Sep-17	Initial version release
1.1	8-Mar-19	Support FPGA <-> FPGA test and ZCU102