UDP10G-IP with CPU Demo Instruction

UDP10G-IP with CPU Demo Instruction

Rev2.1 12-Jul-23

1 Overview

This document provides an example of running the UDP10G-IP demo using two different test environments. The first environment involves one FPGA board that transfers UDP data to a PC running a test application for transferring UDP data via 10G Ethernet. However, the test performance in this test environment is limited by the resource of the PC.

In contrast, the second environment involves two FPGA boards that work together to transfer 10G Ethernet data. This configuration achieves the best performance for transferring UDP data via 10G Ethernet using UDP10G-IP.

The document is divided into several topics. Topic 2 explains how to set up the 10G Ethernet card on the PC to obtain optimal performance for data transfer when running the test in the first environment with the FPGA and PC. Topic 3 describes the example console and test results when running the test in the first environment. Finally, topic 4 provides an example console when running the test in the second environment, using tow FPGAs.

For more details on each topic, please refer to the following sections.

2 PC Setup

Before running demo, please check the network setting on PC. Below is an example of how to set up the 10G Ethernet card.

2.1 IP Setting

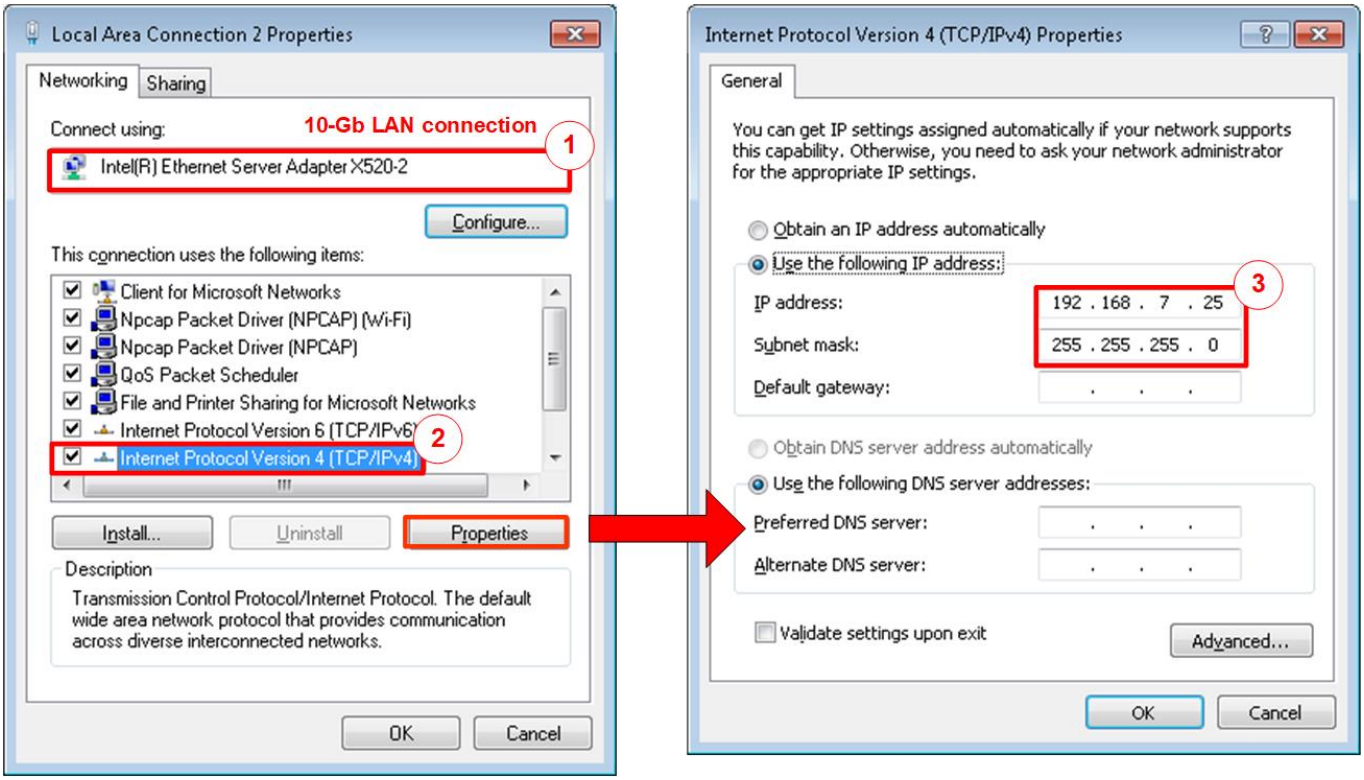

Figure 2-1 Setting IP address for PC

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

2.2 Speed and Frame Setting

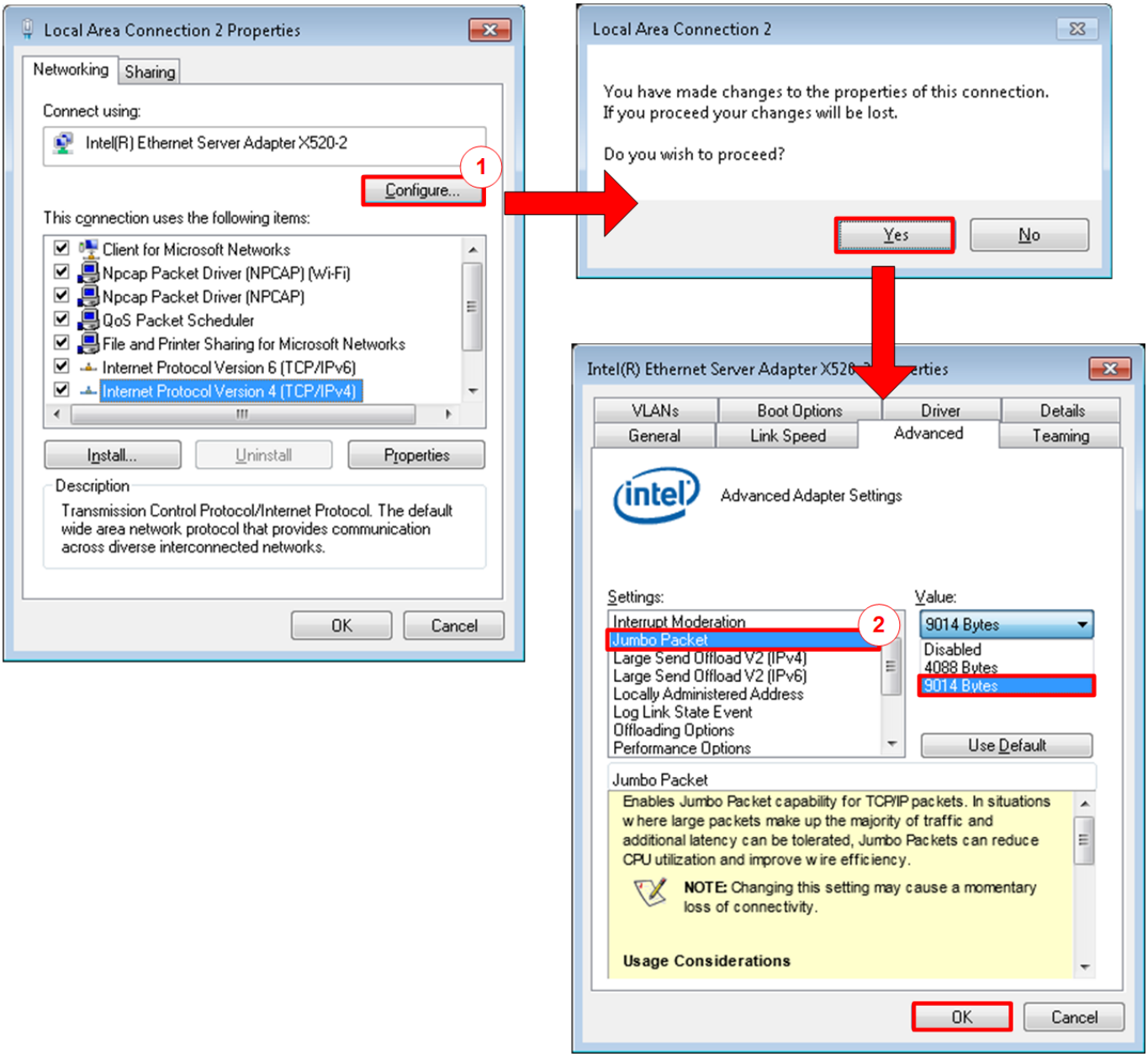

Figure 2-2 Set frame size = jumbo frame

1) On Local Area Connection Properties window, click “Configure” as shown in Figure 2-2.
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 on the bottom window of Figure 2-2.

3) On Link Speed, select “10 Gbps Full Duplex” for running 10-Gigabit transfer test, as shown in Figure 2-3.

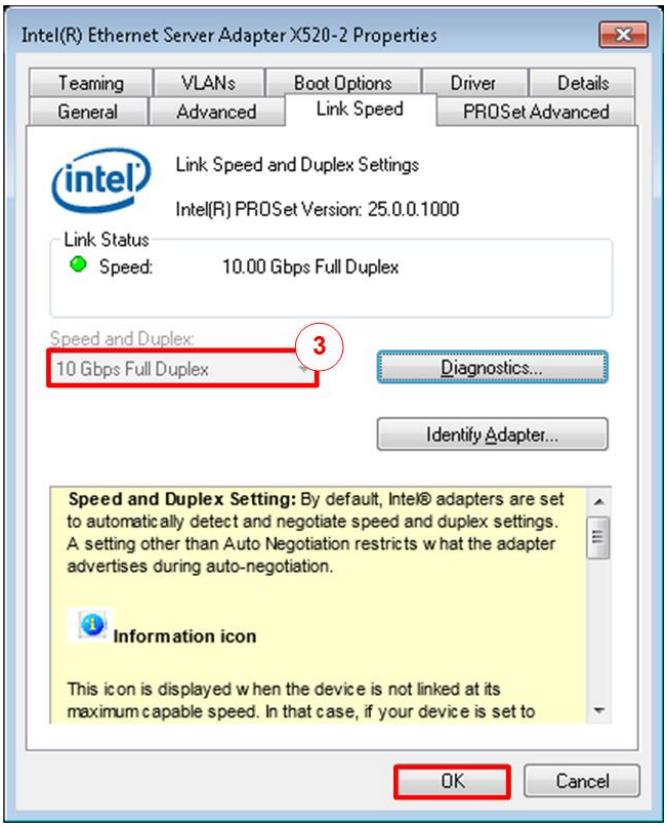

Figure 2-3 Set link speed = 10 Gbps

4) On PROSet Advanced Tab, select “Performance Options” and click “Properties” button.
5) Set “Interrupt Moderation Rate” = OFF.

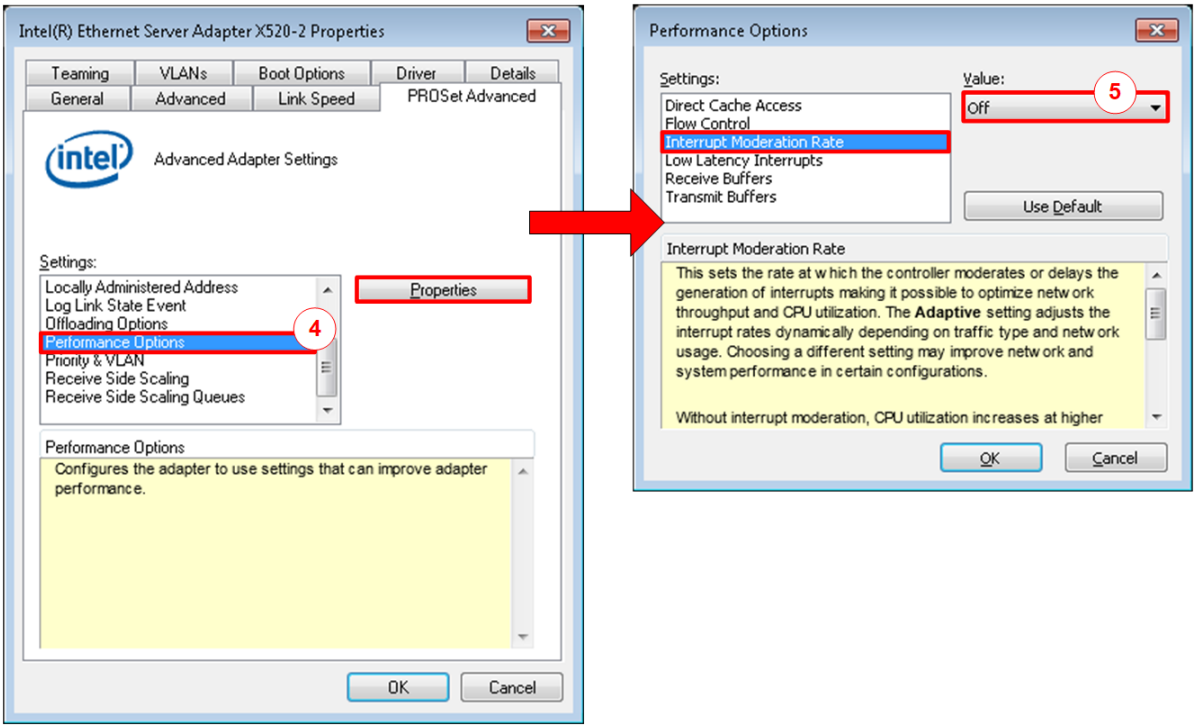

Figure 2-4 Interrupt Moderation Rate

6) Select “Low Latency Interrupts” and click “Properties” button.
7) On “Low Latency Interrupts” window, select “Use Low Latency Interrupts” and click “OK” button.
8) Click “OK” button to save and exit all setting windows.

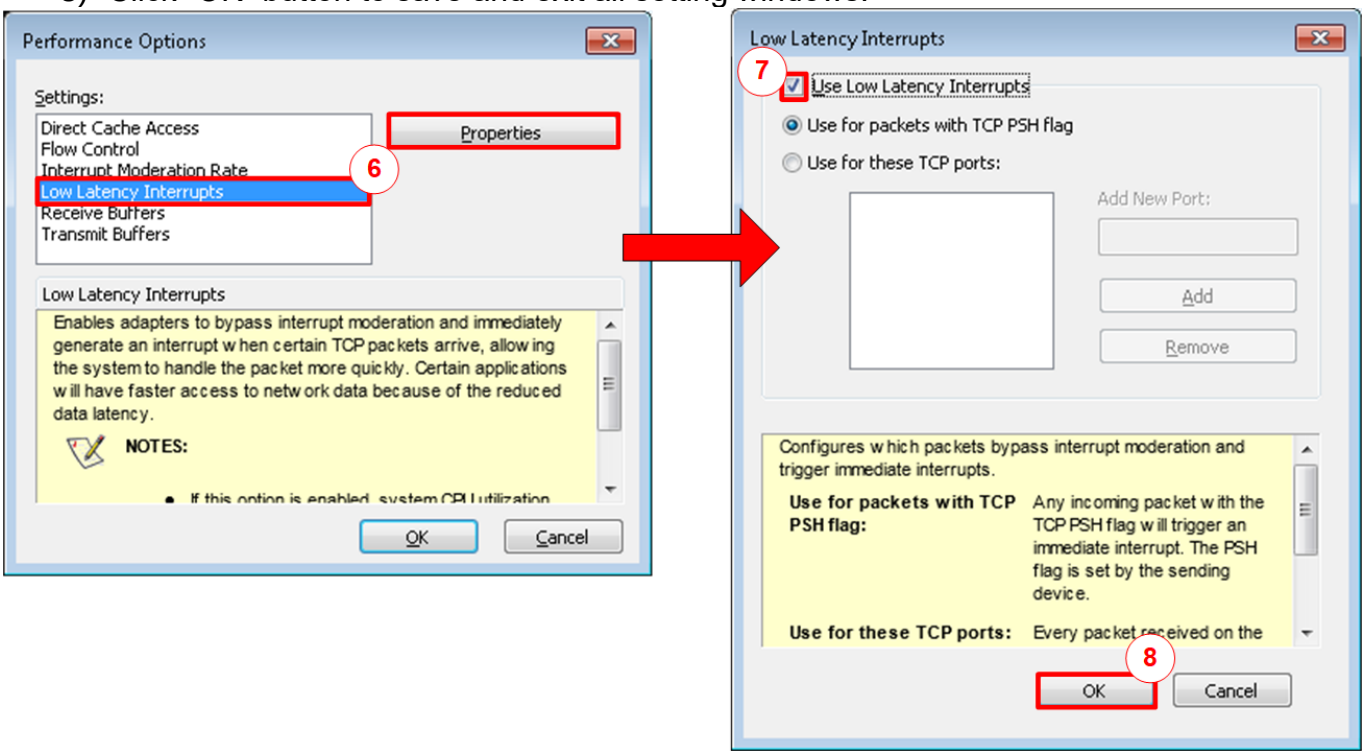

Figure 2-5 Use Low Latency Interrupts

2.3 Power Option Setting

1) Open Control Panel and select Power Options as shown on the left window of Figure 2-6.
2) Change setting to High Performance as shown on the right window of Figure 2-6.

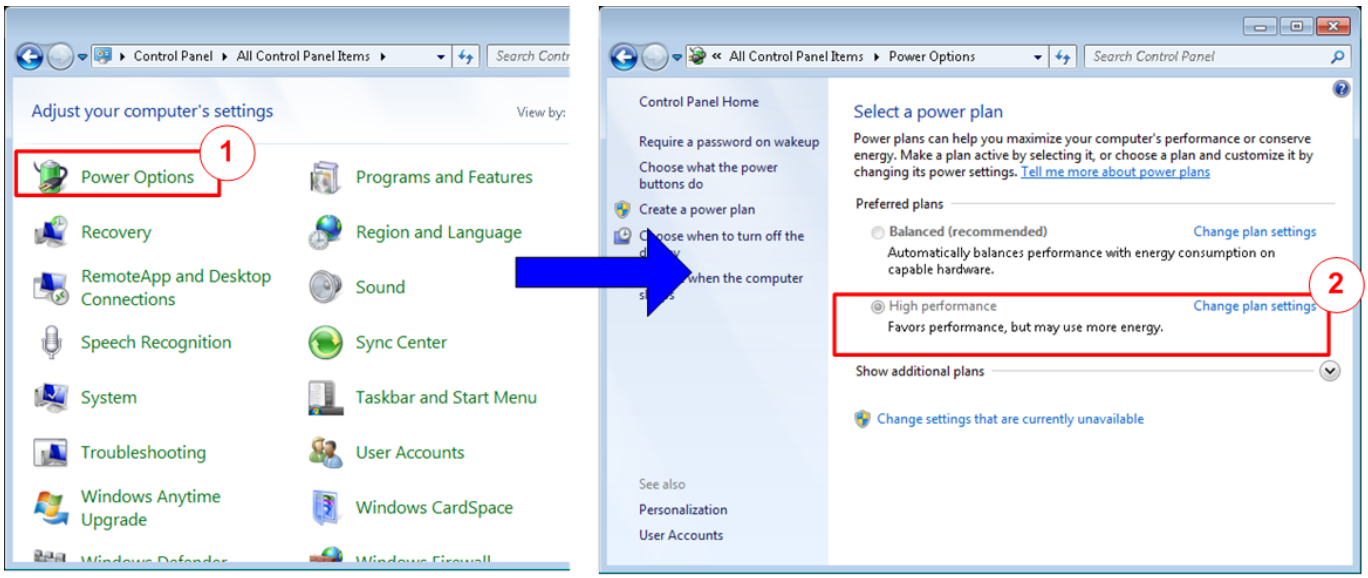

Figure 2-6 Power options

2.4 Firewall Setting

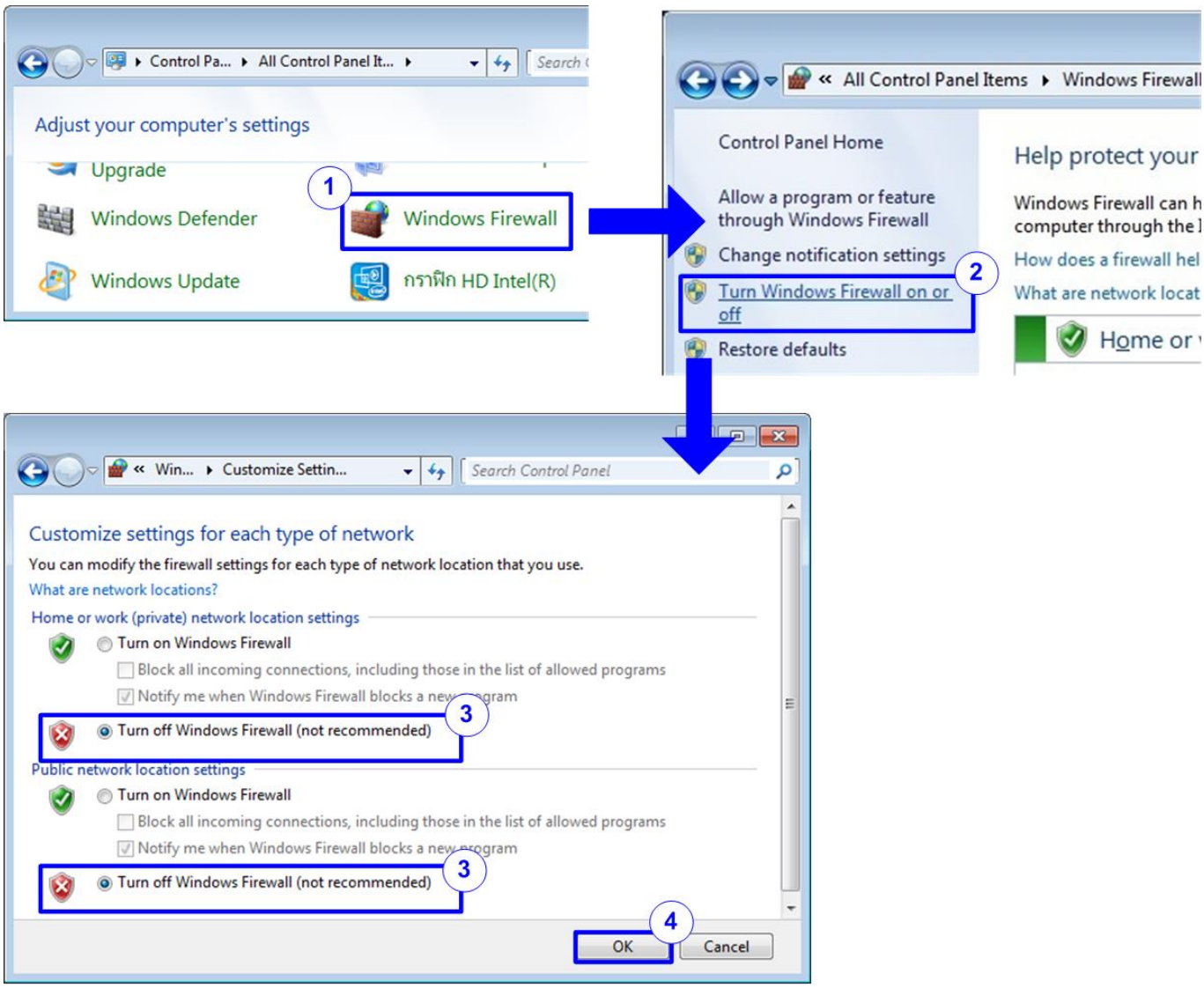

Figure 2-7 Firewall setting

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.

3 Test result using FPGA and PC

3.1 Display UDPIP parameters

Select '0' to check the current parameter in the demo. This will display seven parameters on the console.

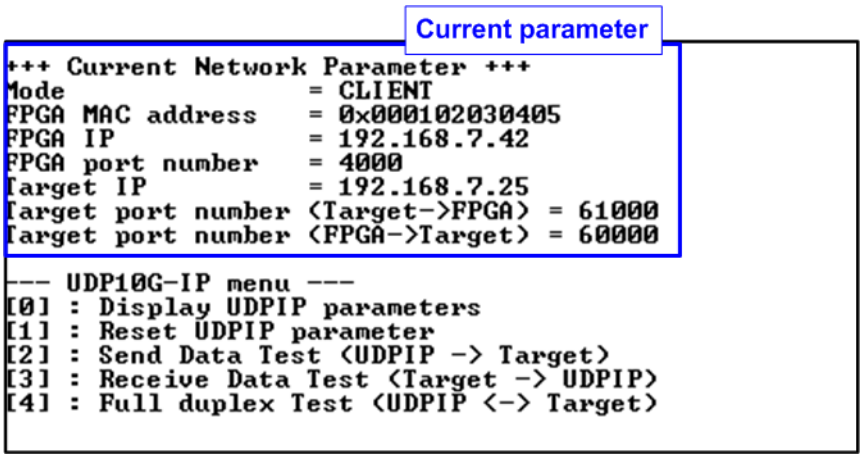

Figure 3-1 Display current parameter result

1) Mode: Set the initialization mode to UDP10G-IP to act as a Server or Client. To run with a PC, input '0' to initialize the IP in Client mode.
2) FPGA MAC address: This is a 48-bit hex value that serves as the MAC address of the FPGA. The default value is 0x000102030405.
3) FPGA IP: This is the IP address of the FPGA. The default value is 192.168.7.42.
   Note: This value is used to be the FPGA IP address, a parameter for the test application on the PC.
4) FPGA port number: This is the port number of the FPGA. The default value is 4000.
   Note: This value is used to be the FPGA port, a parameter for the test application on the PC.
5) Target IP: This is the IP address of the Target device, which is the 10G Ethernet on the PC. The default value is 192.168.7.25.
6) Target port number (Target->FPGA): This is the port number of the Target device to which UDP payload data will be sent from the PC to the FPGA. The default value is 61000.
   Note: This value is used to be the PC port, a parameter for test application on the PC in transmit mode
7) Target port number (FPGA->Target): This is the port number of the Target device from which UDP payload data will be sent from the FPGA to the PC. The default value is 60000.
   Note: This value is used to be the PC port, a parameter for the test application on the PC in receive mode

To change some parameters, the user can set them by using menu [1] (Reset UDPIP parameters).

3.2 Reset UDPIP parameters

Select ‘1’ to modify IP parameters or reset UDP10G-IP. Once selected, the current parameters are displayed, and the user can enter ‘x’ to keep the same parameters or modify individual parameters by entering a different value.

There are seven parameters that can be modified, as described in topic 3.1(Display UDPIP parameter). The range of each parameter is as follows.
Note: If the user sets the invalid input, the input will be rejected and the same value will be used for that parameter.

1) Mode : Input ‘0’ to initialize IP as Client mode.
2) FPGA MAC address : Input a 12-digit of hexadecimal value. Add “0x” as a prefix to indicate a hex value.
3) FPGA IP address : Input a set of four decimal digits separated by “.”. Each digit must be the range of 0-255.
4) FPGA port number : Input a value in the range of 0-65535.
5) Target IP address : Input a set of four decimal digits, similar to the FPGA IP address. This is the IP address of the PC.
6) Target port number (Target->FPGA) : Input a value in the range of 0-65535.
7) Target port number (FPGA->Target) : Input a value in the range of 0-65535.

Once all the parameters are entered, the new parameter set is displayed on the console. The reset signal is then asserted, and the IP begins initialization with the new parameters. Upon completion of the initialization process, the console displays “IP initialization complete”, as shown in Figure 3-2.

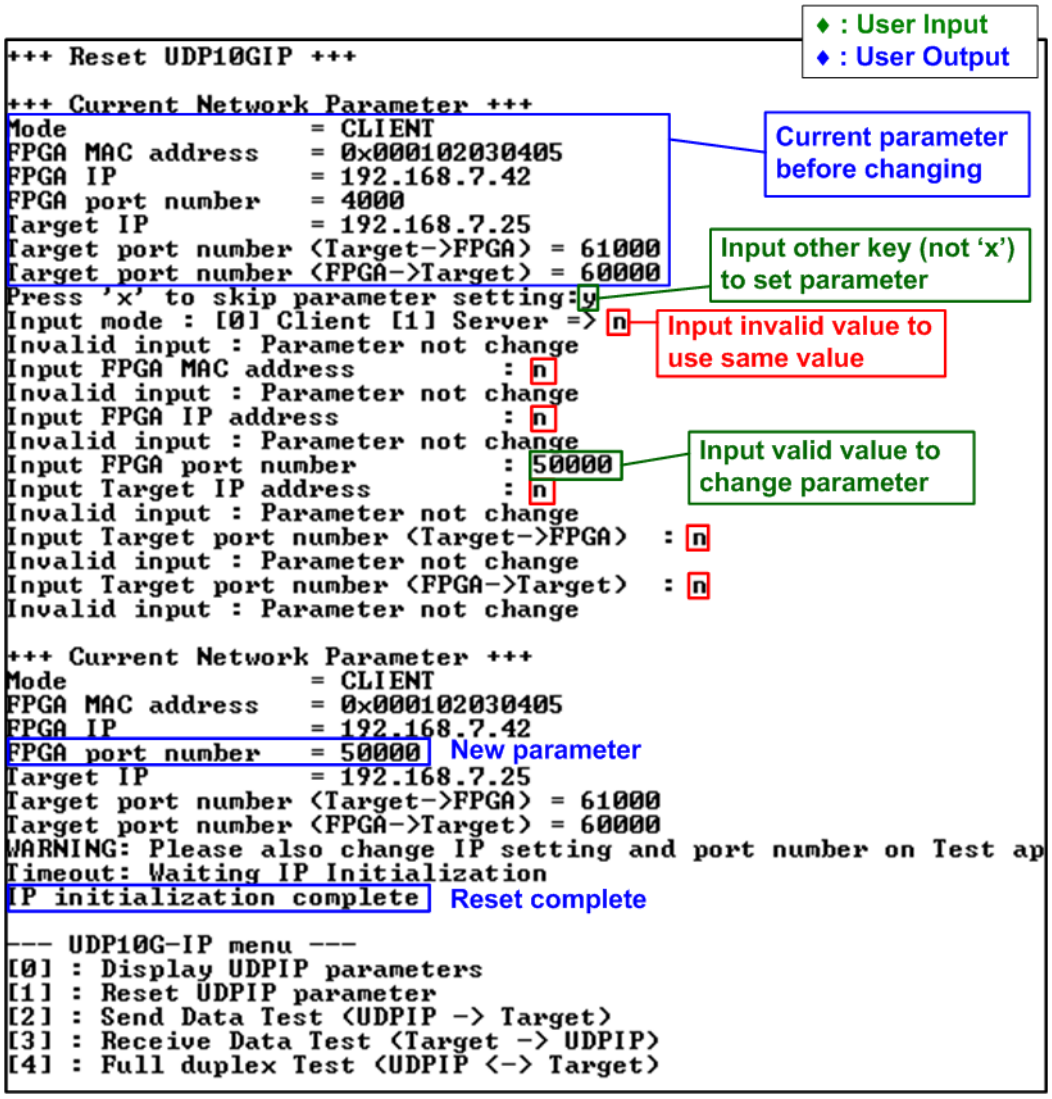

Figure 3-2 Change IP parameter result

3.3 Send Data Test

To transfer data from FPGA to PC, select '2' to run the Send data test on the FPGA console and run the "udpdatatest.exe" application to receive data via Command Prompt on the PC. The user inputs the test parameters for sending data on the FPGA console. The steps to run this menu are shown below.

1) On the FPGA console, input two parameters under Send data test menu.
   i) Input transfer size: The unit of transfer size is byte. Valid values are 0x8 - 0xFFFF_FFF8 and the input must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add "0x" as a prefix.
   ii) Input packet size: The unit of packet size is byte. Valid values are 8 – 8968, and the value must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add "0x" as a prefix.
   Note: If packet size is over 1472, the packet output will be a jumbo frame, and the PC must support this.
2) If all inputs are valid, the recommended parameters for running the test application on the PC will be displayed. "Press any key to start data sending ..." is displayed to begin sending data when user enters any key(s).
3) On the Command prompt, input the test parameters following the recommended values. There are five mandatory parameters and two optional parameters for executing "udpdatatest" to receive data.
   >> udpdatatest [Dir] [FPGAIP] [FPGAPort] [PCPort] [ByteLen] <Pattern> <Timeout>

   Mandatory parameters
   i) Dir : Set 'r' to receive test data from FPGA
   ii) FPGA IP : Set the same value as FPGA IP address
   iii) FPGA port : Set the same value as FPGA port number
   iv) PC port : Set the same value as Target port number (FPGA->Target)
   v) Bytelen : Set the same value as "Input transfer size" of step 1)

   Optional parameters
   i) Pattern : '1'- enable data verification, '0'-disable data verification. The default value is '1' which is applied when there is no input.
   ii) Timeout : Timeout in msec unit. Valid value is 50-65536. It is recommended to set 100 for 10G Ethernet. The default value is 100 which is applied when there is no input.

4) After running the test application, a summary of the setting parameters will be displayed and the application will wait for received data from FPGA.
5) On the FPGA console, input any key(s) to start sending data. The current number of transfer data size will be displayed on both the FPGA console (transmitted data size) and the Command prompt (receive data size) every second.
6) Once all data has been sent, "Send data complete" will be displayed on the FPGA. The test application on the PC will finish when either the total number of received data equals the set value (indicating no lost data) or a timeout when no new data is received is reached, resulting in an error message being displayed on the Command prompt. If data verification is enabled, the first error position will also be displayed. Finally, the total number of receive data and performance will be displayed on both the FPGA console (transmit performance) and the Command prompt (receive performance).

Figure 3-3 shows an example of Send data test using jumbo frame size.

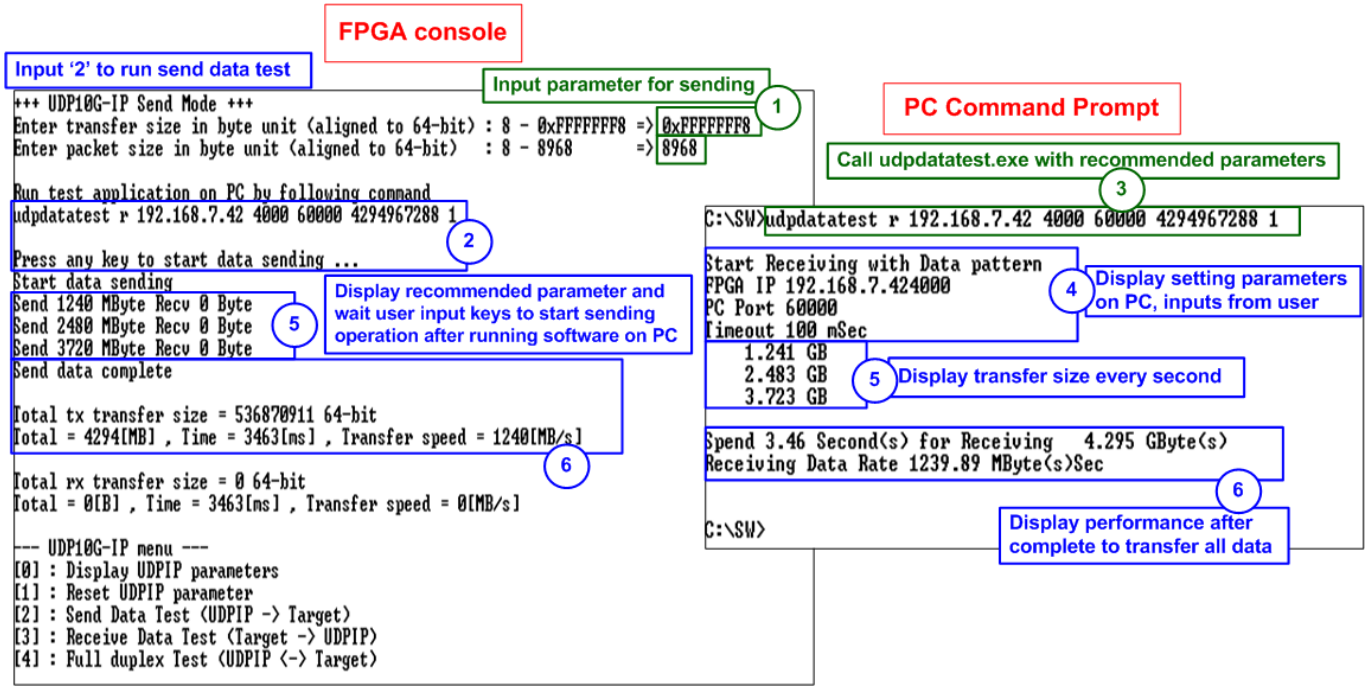

Figure 3-3 Send data test by using jumbo frame

If the input is invalid, "Out-of-range input"/"Invalid input" is displayed and then the operation is cancelled, as shown in Figure 3-4 - Figure 3-5.

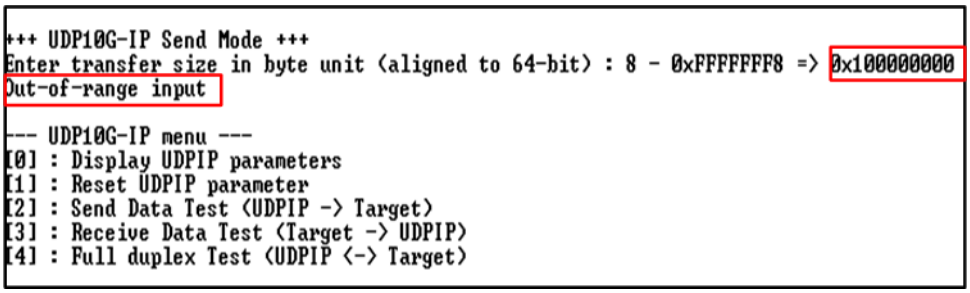

Figure 3-4 Error from invalid transfer size

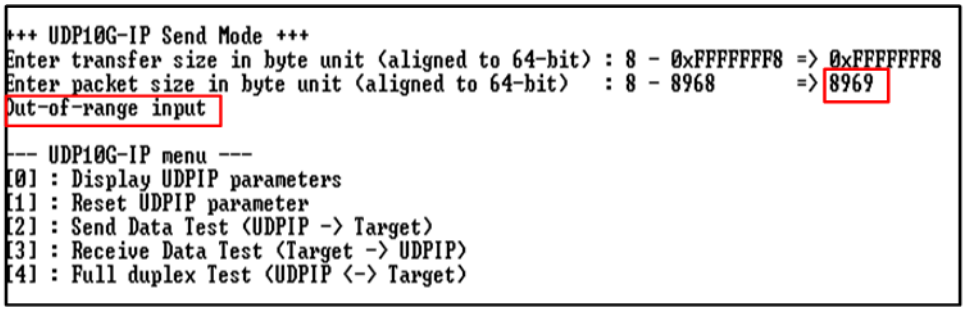

Figure 3-5 Error from invalid packet size

3.4 Receive Data Test

To received data on FPGA from PC, select ‘3’ to run the Receive data test on FPGA and input the required parameters for receiving data on the FPGA console. On the PC, run “udpdatatest.exe” to send data with setting the test parameters of “udpdatatest” on the Command prompt. The steps to run this menu are shown below.

1) On the FPGA console, input two parameters under Receive data test menu.
   i) Input transfer size: The unit of transfer size is byte. Valid values are 0x8 - 0xFFFF_FFF8 and the input must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   ii) Input data verification mode: Set ‘0’ to disable data verification or ‘1’ to enable data verification sent from the PC.
2) If inputs are valid, the recommended parameters for running the test application on the PC will be displayed. The message “Wait data from Target ...” is displayed to indicate that the FPGA is waiting for data to be received from the PC.
3) On the Command prompt, input the test parameters following the recommended values. There are five mandatory parameters and one optional parameter for executing “udpdatatest” to send data.
   >> udpdatatest [Dir] [FPGAIP] [FPGAPort] [PCPort] [ByteLen] <pattern>

   Mandatory parameters
   i) Dir : Set ‘t’ to send test data from PC
   ii) FPGA IP : Set the same value as FPGA IP address
   iii) FPGA port : Set the same value as FPGA port number
   iv) PC port : Set the same value as Target port number (Target->FPGA)
   v) Bytelen : Set the same value as “Input transfer size” of step 1)

   Optional parameters
   i) Pattern : ‘1’- Incremental pattern, ‘0’- dummy pattern. The default value is ‘1’ which is applied when there is no input.

4) Once the test application is initiated, the data is sent from the PC to the FPGA. The current transfer size is displayed on both the FPGA console (receive data size) and the Command prompt (transmit data size) every second.
5) Upon completion of the data transfer, the FPGA console displays “Receive data completed”. If no new data is received until timeout, the FPGA will also complete the operation by timeout condition. Finally, the total number of transfer data size and performance are displayed on both the FPGA console (receive performance) and the Command prompt (transmit performance).

Figure 3-6 shows an example of Receive data test when the data verification mode is disabled on the FPGA console. The left window displays the test results on the FPGA console, while the right window displays the test results on the Command prompt.

Figure 3-7 shows and example of Receive data test when the data verification mode is enabled on the FPGA console. The PC sends incremental data to the FPGA, and the left window displays the test result on the FPGA console, while the right window displays the test results on the Command prompt. If the verification module fails, an error message is displayed.

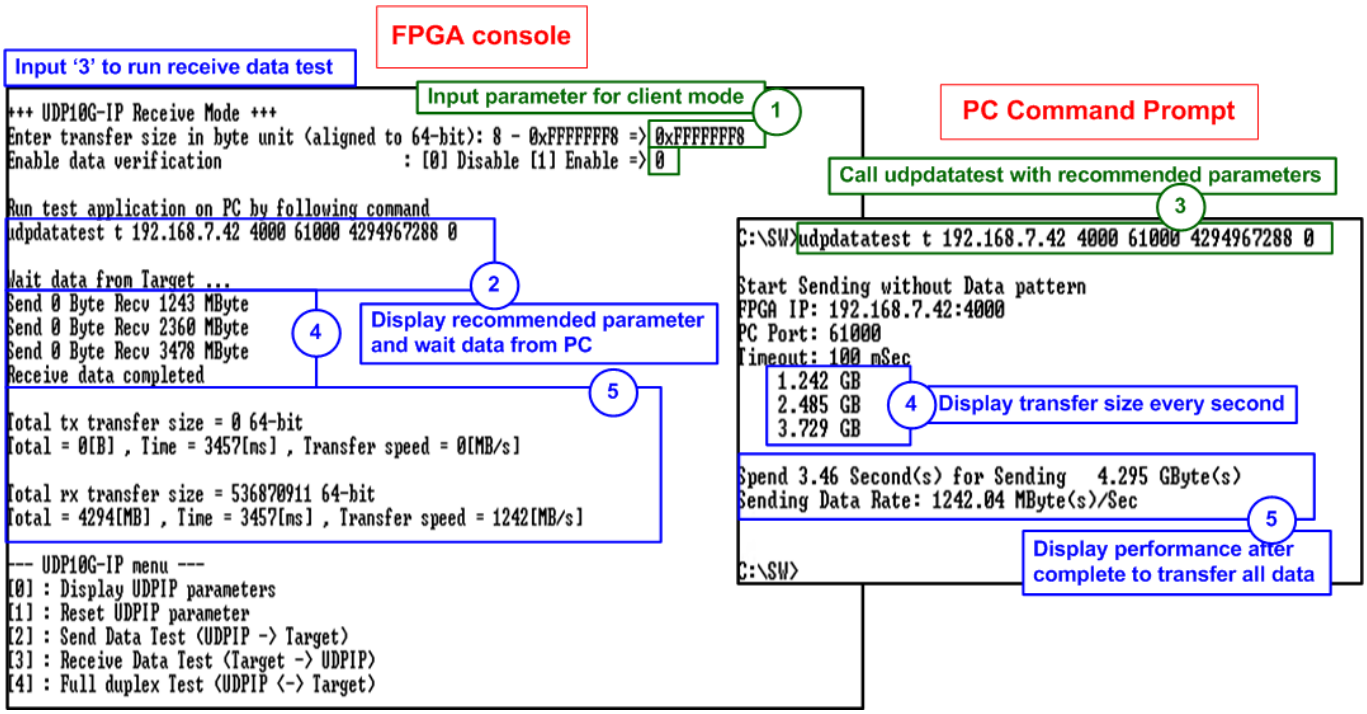

Figure 3-6 Receive data test without data verification

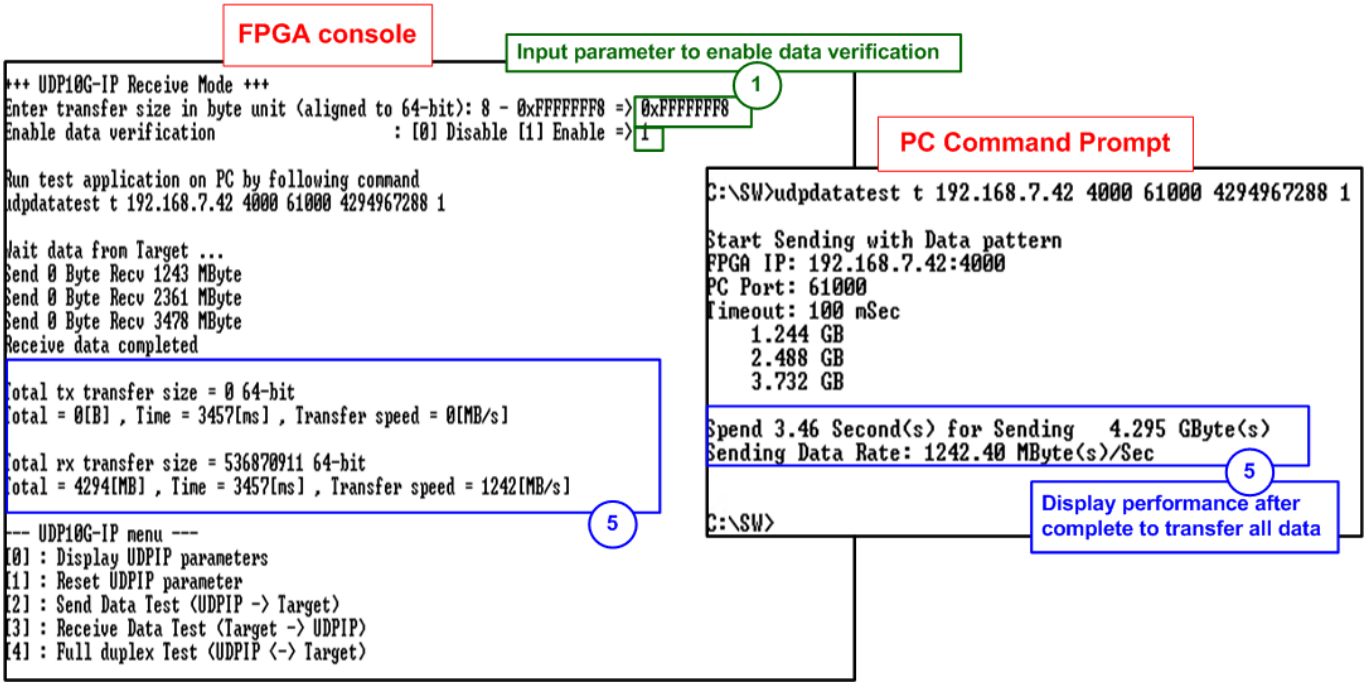

Figure 3-7 Receive data test when enable data verification

3.5 Full duplex Test

Select ‘4’ to run full duplex test which transfers data between FPGA and PC in both directions simultaneously. User inputs test parameters on the FPGA console and PC Command prompt. Two “udpdatatest” applications must be executed by the user on two separate Command prompts using different port number, one for sending data and another for receiving data. The steps to run the test are as follows.

1) On the FPGA console, input three parameters under Full duplex test.
   i) Input transfer size: Unit of transfer size is byte. Valid value is 0x8 - 0xFFFF_FFF8”. The input must be aligned to 8. The input is decimal unit when input only digit number. User adds “0x” to be a prefix for hexadecimal unit. This value must be equal to total transfer size, set on test application.
   ii) Input packet size: Unit of packet size is byte. Valid value is 8 – 8968. The input must be aligned to 8. The input is decimal unit when input only digit number. User adds “0x” to be a prefix for hexadecimal unit.
   iii) Input data verification mode: Set ‘0’ to disable data verification or ‘1’ to enable data verification sent from the PC.
2) If all inputs are valid, two recommended parameter sets to run two applications on the PC are displayed, sending data and receiving data. “Press any key to start data sending ...” is displayed to begin the operation when user enters any key(s).
3) Open two Command prompts to execute the applications using the recommended parameter values. Five mandatory parameters and one optional parameter for executing “udpdatatest” are displayed.

   Receive data (The 1st command prompt)
   >> udpdatatest r [FPGAIP] [FPGAPort] [PCPort] [ByteLen] <Pattern> <Timeout>

   Send data (The 2nd command prompt)
   >> udpdatatest t [FPGAIP] [FPGAPort] [PCPort] [ByteLen] <Pattern>

   Mandatory parameters
   i) Dir :
      The 1st command prompt : Set ‘r’ to receive data
      The 2nd command prompt : Set ‘1’ to send data
   ii) FPGA IP : Set the same value as FPGA IP address
   iii) FPGA port : Set the same value as FPGA port number
   iv) PC port :
      The 1st command prompt: Set the same value as Target port number (FPGA->Target)
      The 2nd command prompt: Set the same value as Target port number (Target->FPGA)
   v) ByteLen : Set the same value as “Input transfer size” of step 1)

   Optional parameters
   i) Pattern : ‘1’- enable data verification, ‘0’-disable data verification. The default value is ‘1’ which is applied when there is no input.
   ii) Timeout : Timeout in msec unit. Valid value is 50-65536. It is recommended to set 100 for 10G Ethernet. The default value is 100 which is applied when there is no input.

4) On the FPGA console, press any key(s) to initiate data transmission to the PC. The FPGA console and two Command prompts will display the current number of data transfers in both directions every second.
5) Once UDP10G-IP has sent and received all data, the FPGA console will display “Transfer data complete”. Finally, the total transfer size and performance will then be displayed on the FPGA console and the Command prompts.

Figure 3-8 displays the transfer performance during full-duplex operation with data verification. The left window is the test result on FPGA console while the right window is the test result on the Command prompts: the upper window shows receive performance and the lower window shows transmit performance.

Figure 3-9 shows an example result when lost data is detected on the PC. In this scenario, the receive software is halted due to a timeout condition, where no new data is received until timeout value is reached. The application displays an error message indicating the first lost position and the total amount of lost data.

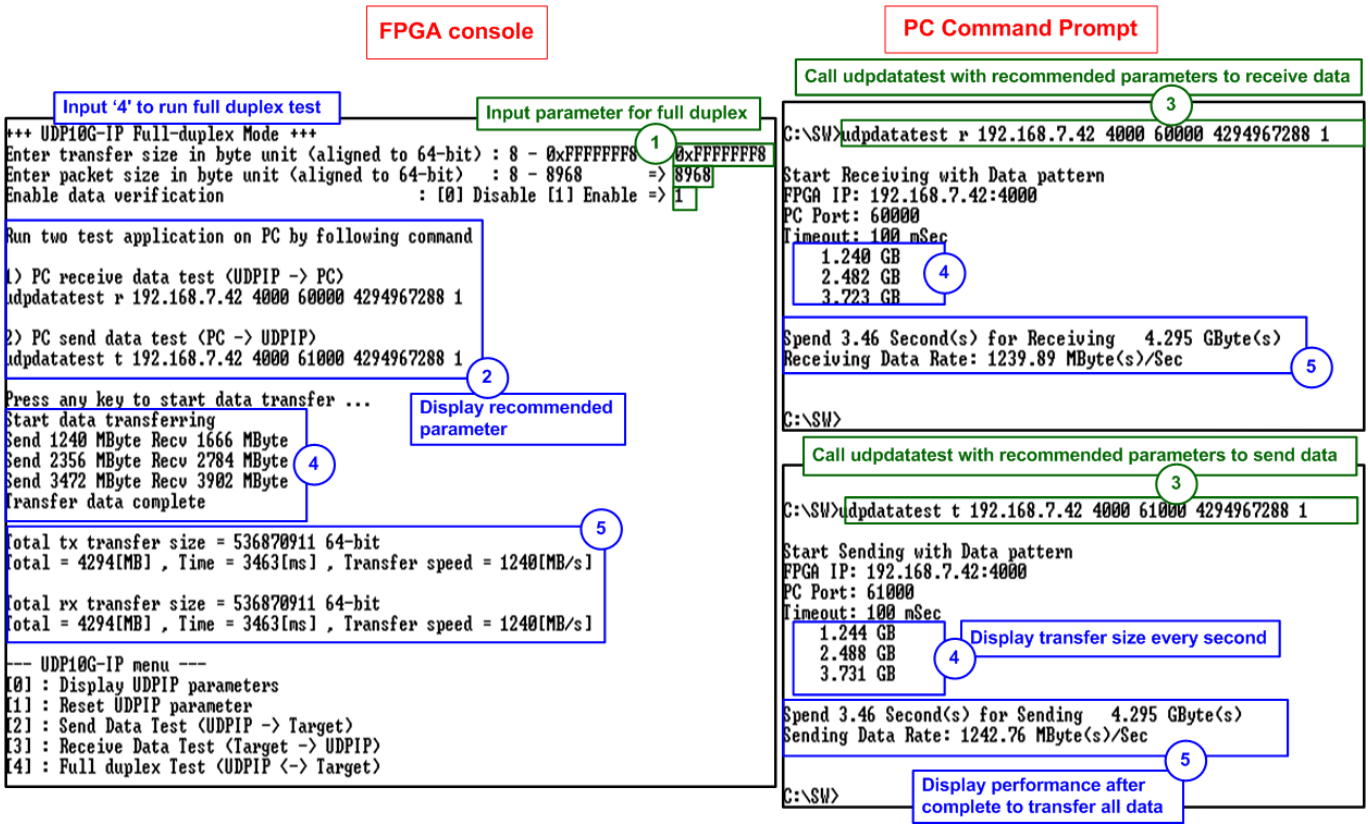

Figure 3-8 Full-duplex test when no lost data

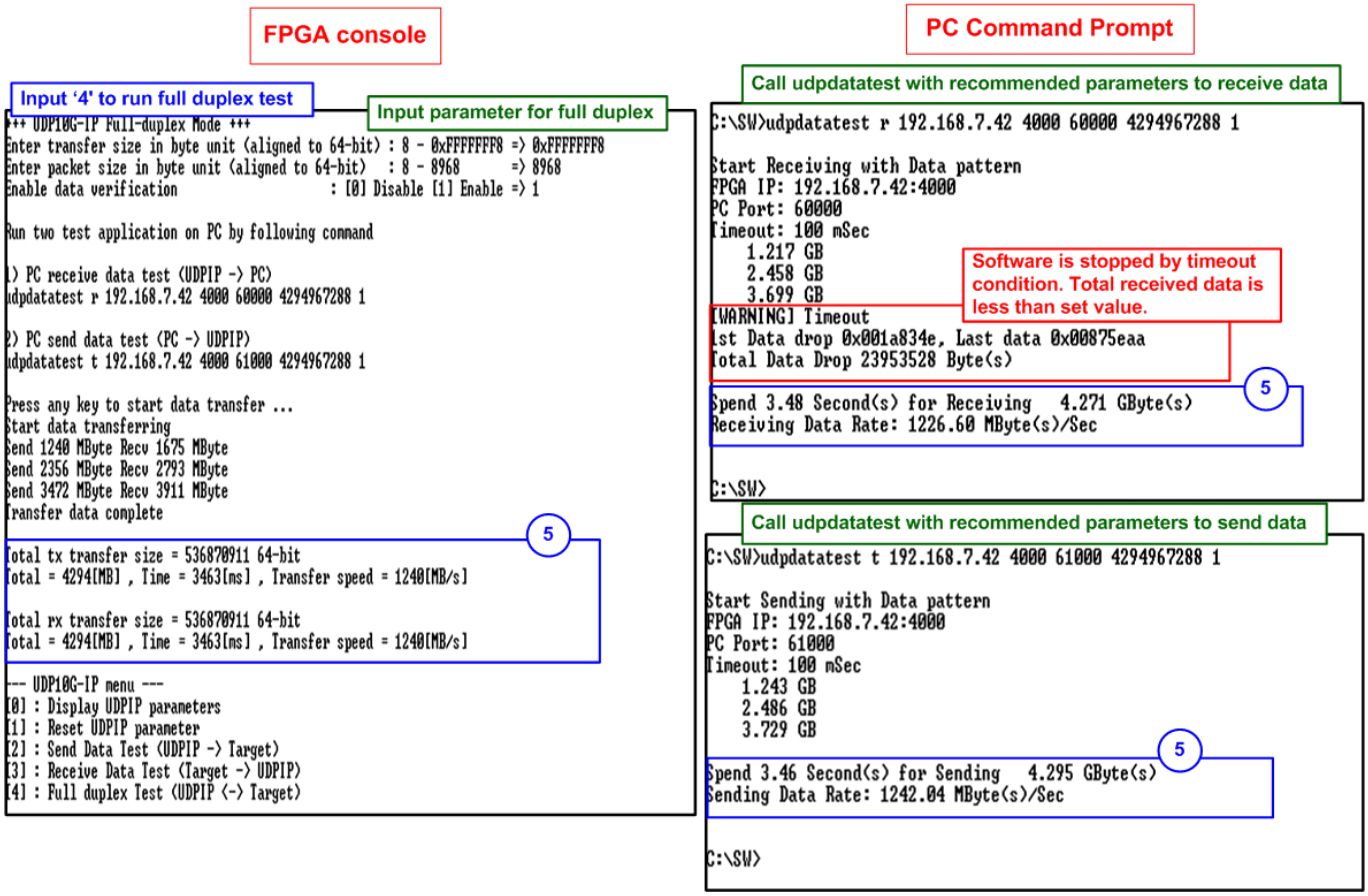

Figure 3-9 Full-duplex test when lost data is detected on PC

4 Test result when using two FPGAs

4.1 Display UDPIP parameter

Select '0' to check current parameter in the demo. This will display seven parameters on the console.

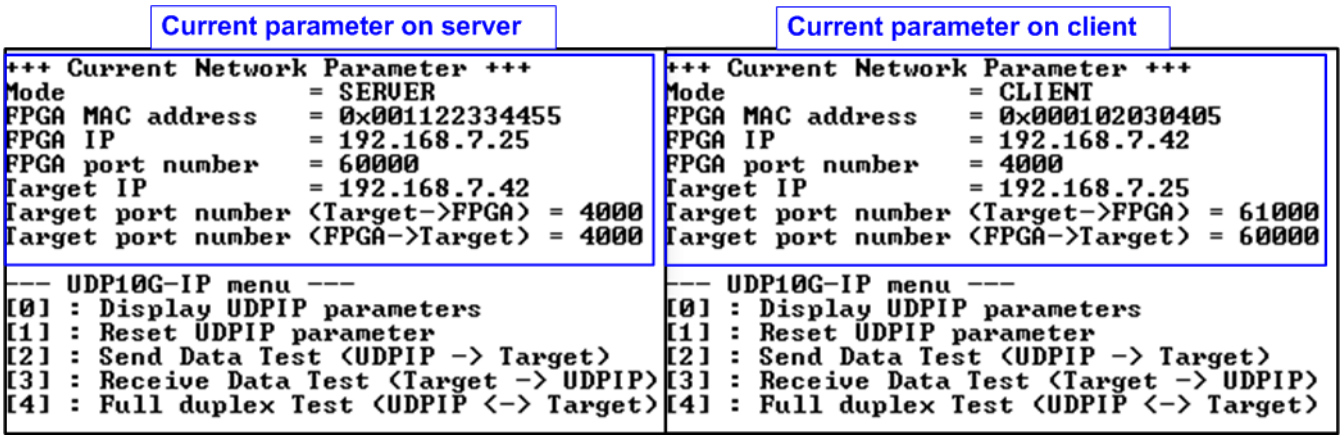

Figure 4-1 Display current parameter result

1) Mode: Set the initialization mode to UDP10G-IP to act as a Server or Client. Input '0' for Client and '1' for Server.
2) FPGA MAC address: This is a 48-bit hex value that serves as the MAC address of the FPGA. The default value is 0x000102030405 for Client and 0x001122334455 for Server.
3) FPGA IP: This is the IP address of the FPGA. The default value is 192.168.7.42 for Client and 192.168.7.25 for Server.
4) FPGA port number: This is the port number of the FPGA. The default value is 4000 for Client and 60000 for Server.
   Note: This value is used to be the FPGA port, a parameter for the test application on the PC.
5) Target IP: This is the IP address of the Target device. The default value is 192.168.7.25 for Client and 192.168.7.42 for Server.
6) Target port number (Target->FPGA): This is the port number of the Target device to receive data from the Target. The default value is 61000 for Client and 4000 for Server.
7) Target port number (FPGA->Target): This is the port number of the Target device to send data to the Target. The default value is 60000 for Client and 4000 for Server.

To change some parameters, the user can set them by using [1] (Reset UDPIP parameters).

Note: When running a test using two FPGA boards, it is important to ensure that the parameters of both FPGA boards are matched. The Target parameters of the first board must be equal to the FPGA parameters of the second board, and vice versa

4.2 Reset UDPIP parameters

Select ‘1’ to modify IP parameters or reset UDP10G-IP. Once selected, the current parameters will be displayed on the console. The user can enter ‘x’ to keep the same parameters or modify individual parameters by entering a different value.

There are seven parameters that can be modified, as described in topic 4.1(Display UDPIP parameter). The range of each parameter is as follows.

Note:

1. *If the user sets the invalid input, the input will be rejected and the same value will be used for that parameter.*
2. *When resetting parameters on the Server FPGA, the Client FPGA must also be reset. The Server should be reset first and wait for an ARP request sent from the Client.*
3. *The Target parameters of the first board must be equal to the FPGA parameters of the second board, and vice versa.*
4. *When using two FPGA board test, the Target port number for Target->FPGA and FPGA->Target must be the same value, which should be equal to the FPGA port number of the other board.*

1) Mode : Input ‘0’ for Client or ‘1’ for Server to determine FPGA initialization mode. It needs to set the different mode for two FPGA boards. One board is client and another board is server.
2) FPGA MAC address : Input a 12-digit of hexadecimal value. Add “0x” as a prefix to indicate a hex value.
3) FPGA IP address : Input a set of four decimal digits separated by “.”. Each digit must be the range of 0-255.
4) FPGA port number : Input a value in the range of 0-65535.
5) Target IP address : Input a set of four decimal digits. Use the same value as FPGA IP address on another board.
6) Target port number (Target->FPGA) : Input a value in the range of 0-65535.
7) Target port number (FPGA->Target) : Input a value in the range of 0-65535.

Once all the parameters are entered, the new parameter set is displayed on the console. The reset signal is then asserted, and the IP begins initialization with the new parameters. Upon completion of the initialization process, the console displays “IP initialization complete”, as shown in Figure 4-2.

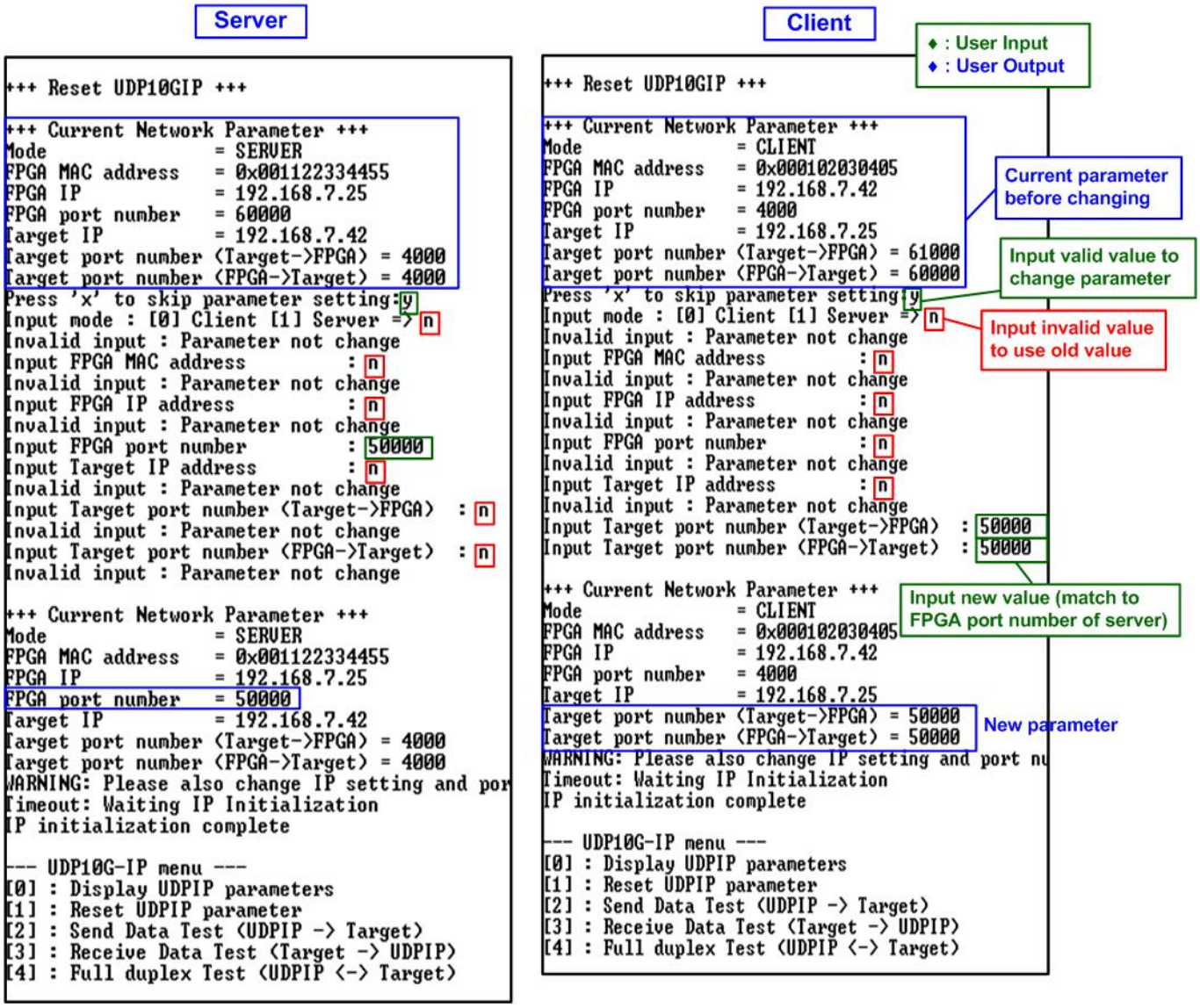

Figure 4-2 Change IP parameter result

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

To execute a half-duplex test with two FPGA boards, one board runs the Receive data test (menu 3), while another board runs the Send data test (menu 2). The user inputs the test parameters on the FPGA console. The steps to run the test are as follows.

1) Under menu 3 (Receive data test) on the FPGA console, input two parameters.
   i) Input transfer size: The unit of transfer size is byte. Valid values are 0x8 - 0xFFFF_FFF8 and the input must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   ii) 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, the message “Wait data from Target ...” is displayed, indicating that the FPGA is waiting for data to be received from another FPGA.
3) Under menu 2 (Send data test) on another FPGA console, input two parameters.
   i) Input transfer size: The unit of transfer size is byte. Valid values are 0x8 - 0xFFFF_FFF8 and the input must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   Note: This value must be equal to the transfer size set in step 1).
   ii) Input packet size: The unit of packet size is byte. Valid values are 8 – 8968, and the value must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   Note: If the packet size is more than 1472, the packet output from UDP10G-IP is a jumbo frame. The user needs to confirm that the network device supports jumbo frames when two FPGA boards are connected through the network device.
4) If all inputs are valid, the message “Press any key to start data sending ...” is displayed, indicating that the data transfer will begin when the user enters any key(s).
5) The user enters any key(s) to start data sending, and then the data starts transferring. During the transfer, the current number of transfer data size is displayed on both FPGA consoles every second.
6) When all data has been sent and received, “Send data complete” is displayed on the FPGA console that runs Send data test, and “Receive data completed” is displayed on the FPGA console that runs Receive data test. Finally, the total transfer size and performance are displayed on both FPGA consoles.

Figure 4-3 shows an example to data transfer between two FPGAs using non-jumbo frame size. The left window shows the FPGA console from FPGA running the Receive data test, and the right window shows the FPGA console from FPGA running the Send data test.

Figure 4-4 shows an example of data transfer between two FPGAs using jumbo frame size, which provides better performance, compared to non-jumbo frame size.

If the user input is invalid, the console displays an error message “Out-of-range input” or ”Invalid input”, and the operation is cancelled, as shown in Figure 3-4 - Figure 3-5, similar to FPGA<->PC test.

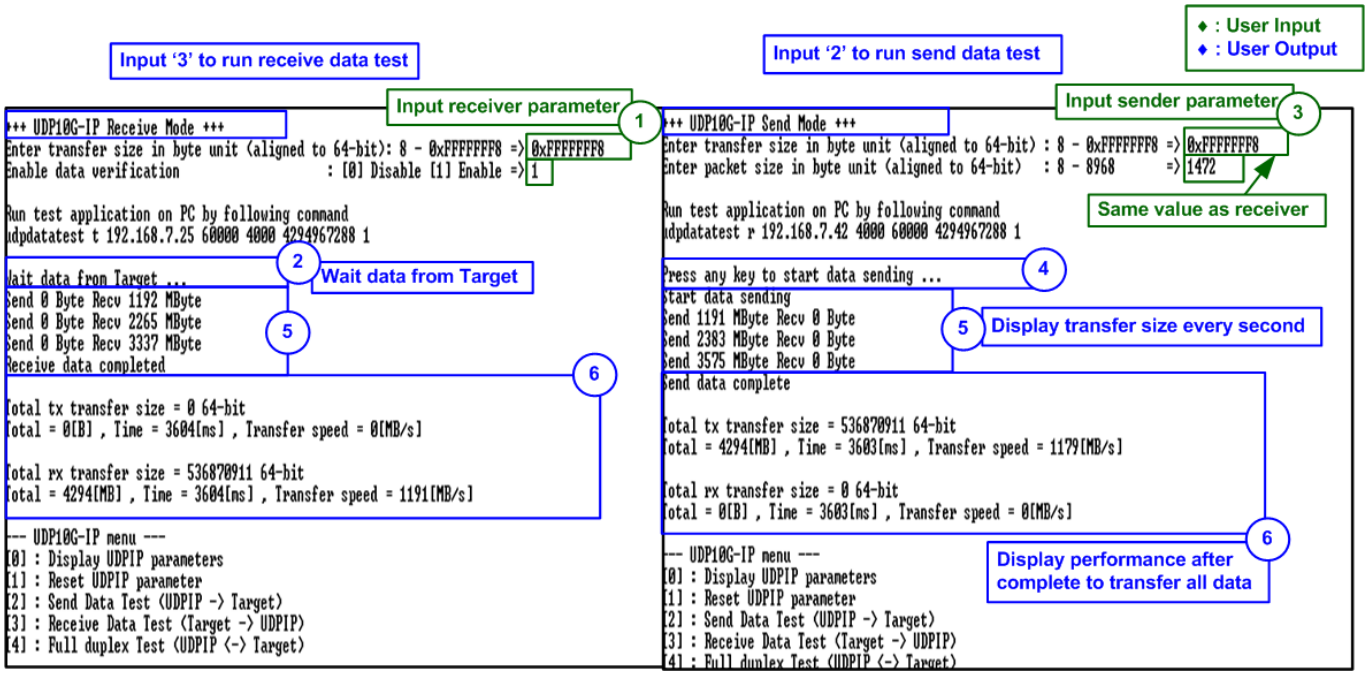

Figure 4-3 Send/receive data test by using non-jumbo frame size

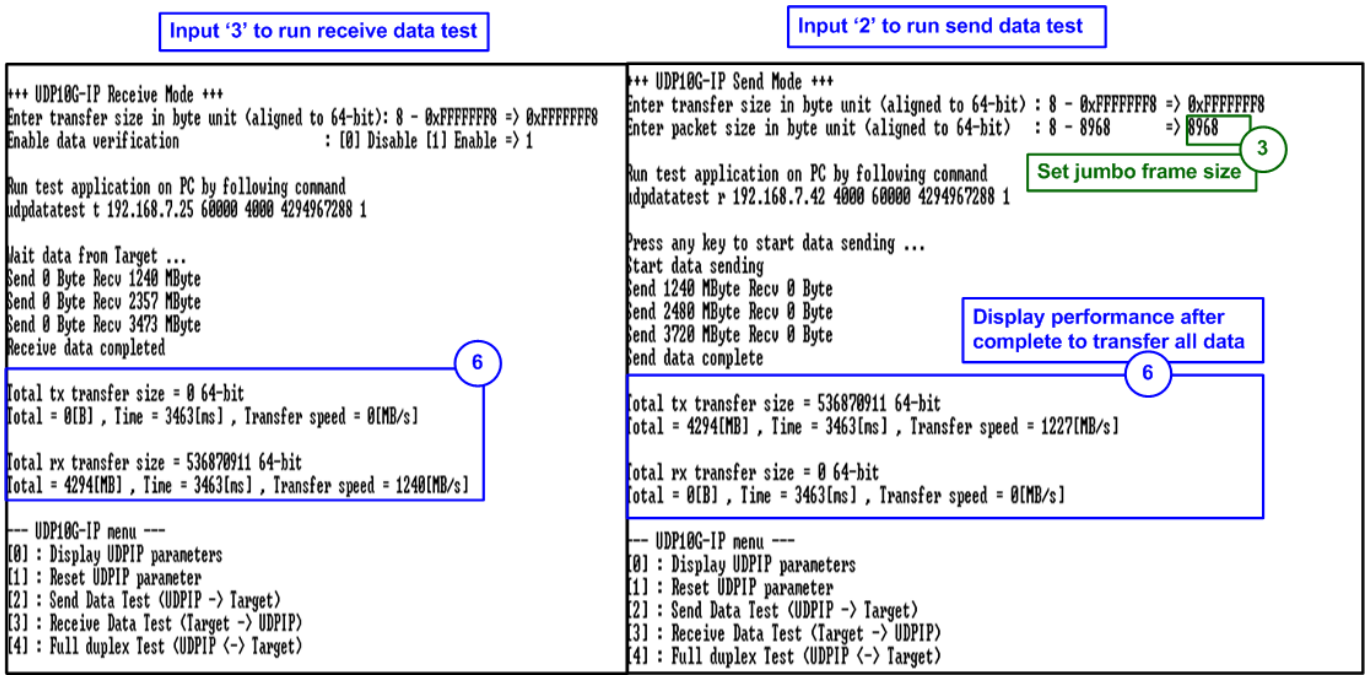

Figure 4-4 Send/receive data test by using jumbo frame size

4.4 Full-duplex Test

Select ‘4’ to run full-duplex test on two FPGA boards for transferring data in both directions simultaneously. The user inputs the test parameters on the FPGA console. The steps to run the test are as follows.

1) On the Server console, input three parameters.
   i) Input transfer size: The unit of transfer size is byte. Valid values are 0x8 - 0xFFFF_FFF8 and the input must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   ii) Input packet size: The unit of packet size is byte. Valid values are 8 – 8968, and the value must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   iii) Input data verification mode: Set ‘0’ to disable data verification or ‘1’ to enable data verification sent from the Client FPGA.
2) If inputs are valid, the message “Wait data from Target …” is displayed, indicating that the FPGA is waiting for data to be received from another FPGA.
3) On the Client console, input three parameters.
   i) Input transfer size: The unit of transfer size is byte. Valid values are 0x8 - 0xFFFF_FFF8 and the input must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   Note: This input must be set by the same value set in step 1).
   ii) Input packet size: The unit of packet size is byte. Valid values are 8 – 8968, and the value must be aligned to 8. The input is in decimal unit when only digit number is entered. If entering a hexadecimal unit, add “0x” as a prefix.
   iii) Input data verification mode: Set ‘0’ to disable data verification or ‘1’ to enable data verification sent from the Server FPGA.
4) If inputs are valid, the message “Press any key to start data sending ...” is displayed, indicating that the data transfer will begin when the user enters any key(s). The user enters any keys to start full-duplex test.
5) The data starts transferring, and the current number of transferred data is displayed on both FPGA consoles every second.
6) Once all data has been transferred, the console displays “Transfer data complete” on both FPGAs. Finally, the total number of transferred data and performance are displayed on both FPGA consoles.

Figure 4-5 and Figure 4-6 shows the results when running full-duplex using non-jumbo frame size and jumbo frame size, respectively. The left window shows the Server console while the right window shows the Client console. Similar to the half-duplex test, using jumbo frame size shows better performance than using non-jumbo frame size

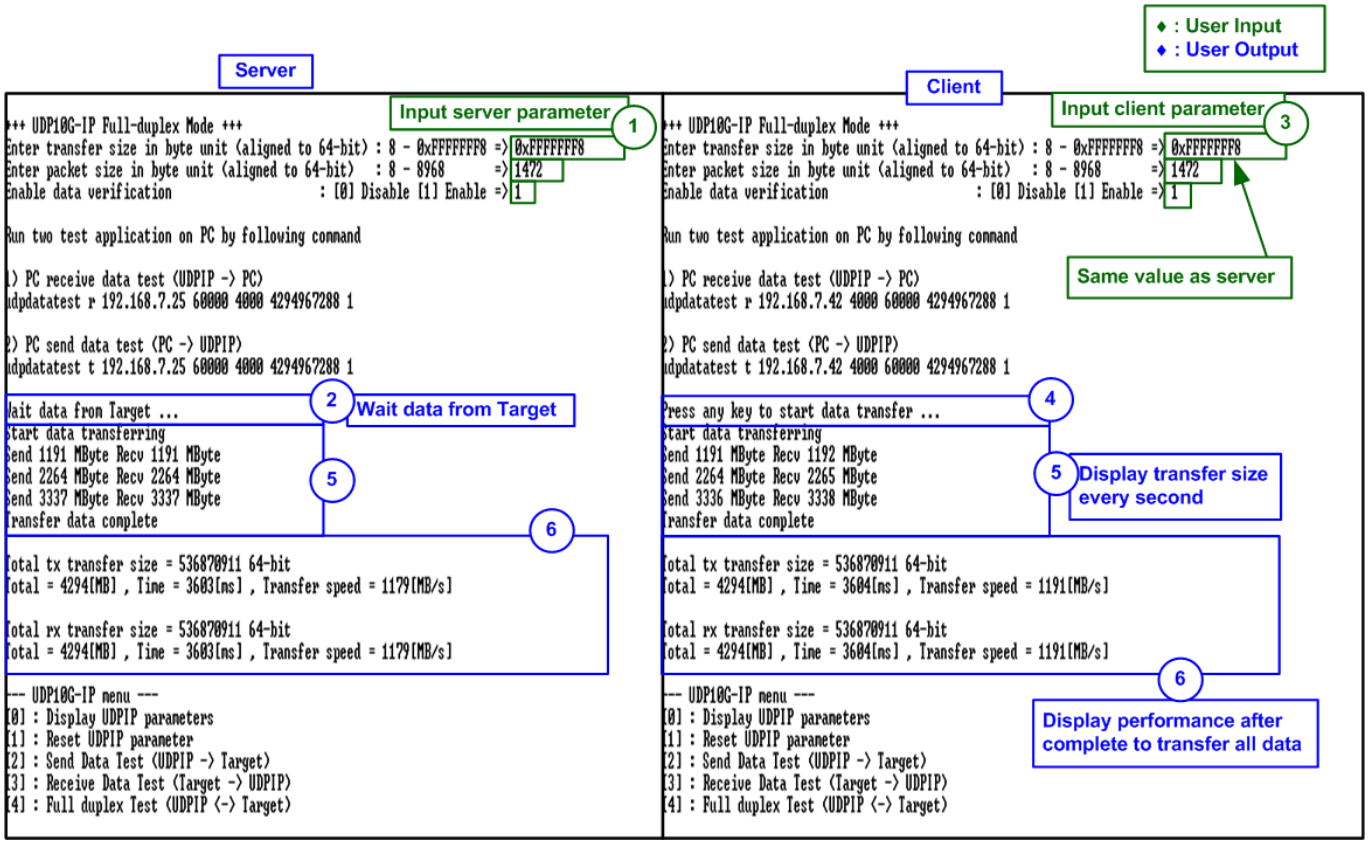

Figure 4-5 Full-duplex test when using non-jumbo frame size

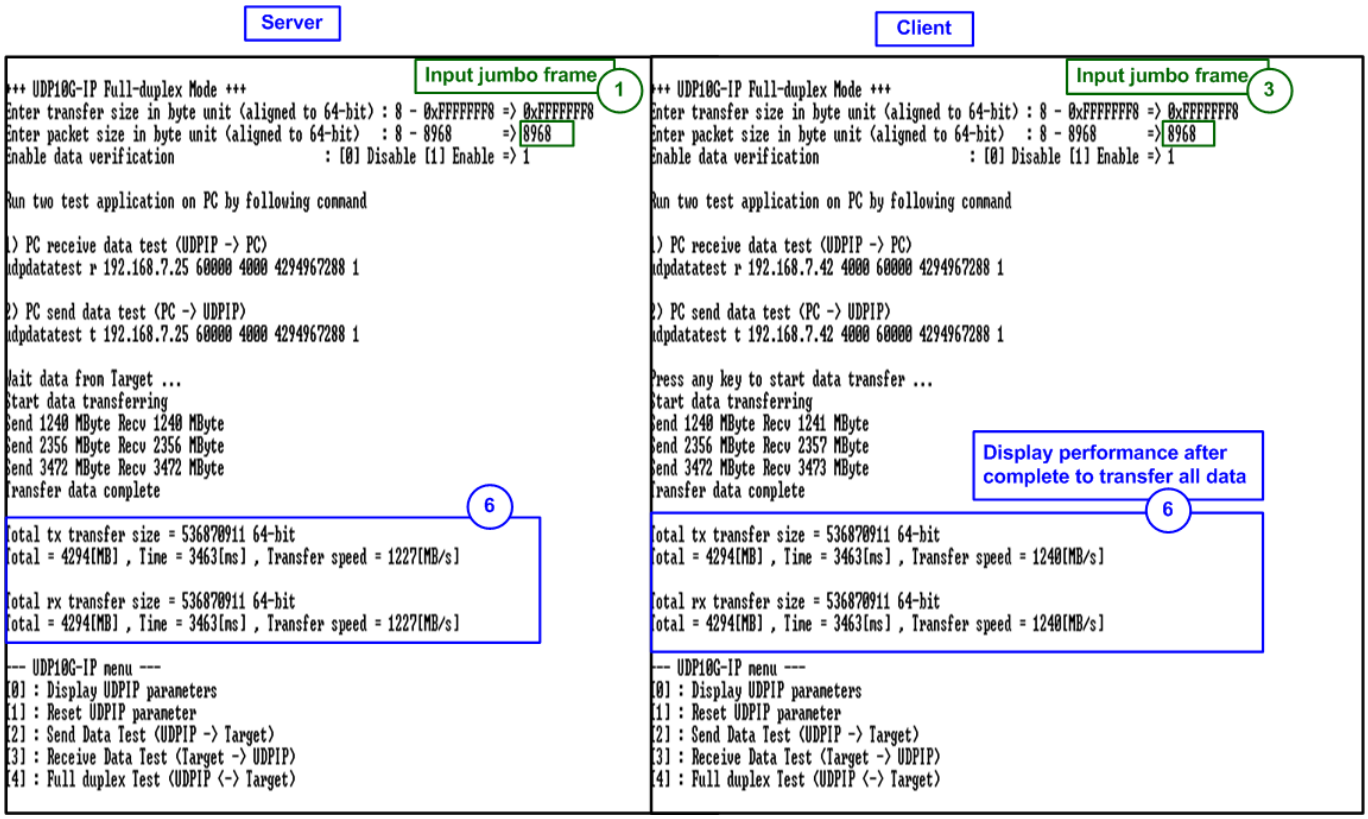

Figure 4-6 Full-duplex test when using jumbo frame size

5 Revision History

Revision	Date	Description
2.1	9-Mar-23	Update performance
2.0	21-Aug-20	Remove hardware setup from the document
1.1	8-Mar-19	Support FPGA <-> FPGA test and ZCU102
1.0	15-Sep-17	Initial version release