

UDP10G-IP Demo Instruction

Rev1.1 8-Mar-19

Contents

1	Overview	2
Par	rt A UDP10G-IP demo by using FPGA and PC	3
2	Environment Setup	3
3	PC Setup	5
3	3.1 IP Setting	5
3	3.2 Speed and Frame Setting	6
3	3.3 Power Option Setting	9
3	3.4 Firewall Setting	10
4	FPGA board setup	11
5	Main menu	15
5	5.1 Display current parameter	15
5	5.2 Reset UDP10G-IP	16
5	5.3 Send Data Test	18
5	5.4 Receive Data Test	20
5	5.5 Full-duplex Test	22
_		
Pa	rt B UDP10G-IP demo by using two FPGAs	24
6	Environment Setup	24
/	FPGA board setup	
8	Main menu	
8	3.1 Display current parameter	29
8	3.2 Reset UDP10G-IP	
8	3.3 Send and Receive data Test (Half-duplex test)	
8	3.4 ⊢uii-aupiex iest	
9	Revision history	

D•G

dg_udp10gip_instruction_xilinx_en.doc

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.





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 <u>http://www.intel.com/content/www/us/en/network-adapters/converged-network-adapters/</u> <u>ethernet-x520-server-adapters-brief.html</u>
- [2] a) 10-Gigabit SFP+ DAC cable <u>http://www.netgear.com/business/products/switches/modules-accessories/axc761.aspx</u>
 b) 10-Gigabit SFP+ transceiver + optical cable SFP+ transceiver (850nm) <u>http://www.fit-foxconn.com/Product/ProductDetail?topClassID=Electronic%20Module&&</u> <u>PN=AFBR-709SMZ</u> 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 2-2 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

Connect using: 10-Gb LAN connection Intel(R) Ethemet Server Adapter X520-2	You can get IP settings assigned this capability. Otherwise, you ne for the appropriate IP settings.	automatically if your network support eed to ask your network administrator
This connection uses the following items:	Obtain an IP address autom Obtain an IP address autom	atically
Client for Microsoft Networks	ID address:	197 168 7 75
Reg QoS Packet Scheduler Reg Rile and Printer Sharing for Microsoft Networks	Subnet mask:	255.255.255.0
Internet Protocol Version 6 (TCP/IPV6) Internet Protocol Version 4 (TCP/IPV4)	Default gateway:	
 Link-Layer Topology Discovery Mapper I/O Driver Link-Layer Topology Discovery Responder 	 Obtain DNS server address Use the following DNS server 	automatically er addresses:
Install Uninstall Properties	Preferred DNS server:	
Description	Alternate DNS server:	
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.	Validate settings upon exit	Advanced

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

🌵 Local Area Connection 2 Properties	Local Area Connection 2
Networking Sharing Connect using: Intel(R) Ethemet Server Adapter X520-2 Configure	You have made changes to the properties of this connection. If you proceed your changes will be lost. Do you wish to proceed?
	<u>Yes</u> <u>No</u> Intel(R) Ethernet Server Adapter X52Q-2 boerties
	VLANs Boot Options Driver Details
	General Link Speed Advanced Teaming
OK Cancel	Profile: Standard Server Settings: Value: Interrupt Moderation 9014 Bytes Jumbo Packet Disabled Large Send Offload V2 (IPv6) Disabled Large Send Offload V2 (IPv6) Disabled Locally Administered Address 014 Bytes Log Link State Event Performance Options Priority & VLAN Use Default Jumbo Packet Enables, Jumbo Packet capability for TCP/IP packets. In situations
	Enables Johns Packets make up the majority of traffic and additional latency can be tolerated, Jumbo Packets can reduce CPU utilization and improve wire efficiency. Jumbo Packets are larger than standard Ethernet frames, which are approximately 1.5k in size. NOTE: Changing this setting may cause a momentary loss of connectivity. OK Cancel

- 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.

VLANS	Boot Options	Driver	Details
General	Link Speed	Advanced	Teaming
intal	Link Speed and Duple	x Settings	
inter	Intel(R) PROSet Versi	no: 19 1 51 0	
Link Status	inter(n) i no set versi	. 13.1.31.0	
Speed:	10.00 Gbps/Ful	Duplex	
0 Gbps Full Du	blex	Identify A	dapter
0 Gbps Full Du	olex	Identify A	ydapter
Speed and D	uplex Setting: By de	efault, Intel® adapter	s are set
to automaticall	y detect and negotiate	e speed and duplex	settings.
advertises dur	ing auto-negotiation.	n restricts what the	adapter
Temperature	e: Displays temperatu	re state if the adapt	er has a
temperature se	ensor.		
SFP+ Module	s:		



- 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.

VLANs	Boot Options	Driver	Details	Settinas:		
General	Link Speed	Advanced	Teaming	Flow Control		Properties
\frown				Interrupt Moderation P	Rate	
intel)	Advanced Adapter Se	ttings		Receive Buffers		
	122	1990		 Transmit Buffers		
file: St	andard Server	-				
				Low Latency Interrupt	ts	
emupt Mode mbo Packet rge Send Of rge Send Of cally Admini g Link State	ration fload V2 (IPv4) fload V2 (IPv6) stered Address Event potions		perties	Enables adapters to generate an interrup the system to handle will have faster acced data latency. NOTES: • If this op	bypass interrupt t when certain Ti the packet more ess to network d	moderation and immediately CP packets arrive, allowing e quickly. Certain applications lata because of the reduced system CPU utilization may
ority & VLAN) Intiana			increase		OK Can
erformance		gs that can improve	adapter	Low Latency Interrup	ncy Interrupts	
performance		gs that can improve	adapter	Low Latency Interrup	ncy Interrupts ts with TCP PSH fla TCP ports:	ag Add New Port:
performance		OK	adapter	Low Latency Interrup	packets bypass interrupts. s with TCP An TCP	ag Add New Port: Add Add Add Remove interrupt moderation and y incoming packet with the P PSH flag will trigger an mediate interrupt. The PSH g is set by the sending vice.

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.

	Control Panel + All Control	ol Panel It	ems > • 47 Search Contr.	G 😡 🗣 📽 All Control Panel I	Items Power Options
Adjust ye	our computer's settings		View by:	Control Panel Home Require a password on wakeup	Select a power plan Power plans can help you maximize your computer's performance or conserve energy. Make a plan active by selecting it, or choose a plan and customize it by
🍞 Po	ower Options	圓	Programs and Features	Choose what the power buttons do	changing its power settings. <u>Tell me more about power plans</u>
-0		0		😵 Create a power plan	Preferred plans
Re Re	ecovery	2	Region and Language	Croose when to turn off the	Balanced (recommended) Change plan settings
Re Co	emoteApp and Desktop onnections	0	Sound	d when the computer	Automatically balances performance with energy consumption on capable hardware.
0 sr	peech Recognition		Sync Center		Favors performance, but may use more energy.
			1		Show additional plans
S)	/stem	1	Taskbar and Start Menu		Change attend that are suggestively with the
Tr	oubleshooting	52	User Accounts		Change settings that are currently unavailable
💐 w	findows Anytime pgrade	1	Windows CardSpace	See also Personalization	
-	Cadava Defendas	-	and a min a	USER ACCOUNTS	



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.
- 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: Buad rate=115,200, Data=8 bit, Non-Parity, and Stop = 1.





- 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.



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

ivado 2017.4	HARDWARE MANAGER - unconnected	
Eile Flow Iools Window Help Q- Quick Access	1 No hardware target is open. Open tar	get
	Hardware	Auto Connect
Quick Start Create Project >	HARDWARE M AGER - localhostxil	Available Targets on Server Open New Target inx_tcf/Digiler 4. Click Program device am device Pefresh device
Open Project > Open Example Project >	Hardware	
Tasks Manage IP > 1. Click Open Hardware Manager	Name Liccalhost (1)	Status Connected 3. Select FPGA device to program bit file
Open Hardware Manager > Xilinx Tcl Store >	 	Not programm initor)
	Program vice Select a bitsby am programming file and do device. You can optio 5. Click "" the debug cores cont file (TOE10Cl	woload it to your hardware to select Programming PUTest.bit)
	Bitstream file: D:/Temp/UDP10GIPTe Debug probes file: Enable end of startup check 6. Click Program button f	est_KCU105.bit
Figure 4.6 Pro	aram EPGA by Vivado	



- 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.



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.

<u>File Edit Setup Control Window Help</u>	
Press 'x' to skip parameter setting x -> Rese IP initialization complete default	t by using parameter
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>	-> Main Menu

<u>Note</u>: 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.

<u>File Edit Setup Control Window Help</u> +++ 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)	💆 COM9 - Tera Term VT	
<pre>+++ 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 parameters [1] : Reset UDP10GIP parameter [2] : Send Data Test (UDP10GIP -> Target) [3] : Receive Data Test (Target -> UDP10GIP) [4] : Full duplex Test (UDP10GIP <-> Target)</pre>	<u>File Edit Setup Control Window H</u> elp	
	+++ 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 -> UDP10GI [4] : Full duplex Test (UDP10GIP <-> Target	-> Current parameter

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.



🔟 COM9 - Tera Term VT	
<u>File Edit Setup Control Window Help</u>	Green : User Input Blue: Output to console
<pre>+++ 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 Press 'x' to skip parameter setting:by Input mode : [0] Client [1] Server => n Input Invalid input : Parameter not change Input FPGA MAC address : n Invalid input : Parameter not change Input FPGA port number : 50000 Input FPGA port number : 50000 Input Target IP address : n Invalid input : Parameter not change 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 Input Target port number (FPGA->Target) : n Invalid input : Parameter not change Input Target port number (FPGA->Target) : n Invalid input : Parameter not change</pre>	Current parameter before changing Input other key (not 'x') to set parameter It invalid value to same value
<pre>+++ Current Network Parameter +++ Mode = CLIENT FPGA MAC address = 0x000102030405 FPGA IP = 192.168.7.42 FPGA port number = 50000 New parameter Target IP = 192.168.7.25 Target port number (Target->FPGA) = 61000 Target port number (FPGA->Target) = 60000 WARNING: Please also change IP setting and po IP initialization complete Reset complete UDP10GIP menu [0] : Show UDP10GIP parameters [1] : Reset UDP10GIP parameters [1] : Reset UDP10GIP parameter [2] : Send Data Test (UDP10GIP -> Target) [3] : Receive Data Test (UDP10GIP <-> Target) [4] : Full duplex Test (UDP10GIP <-> Target)</pre>	rt number on Test app

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 0xFFFFFF8. 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.
 <u>Note</u>: 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).





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	×
<u>File Edit S</u> etup C <u>o</u> ntrol <u>W</u> indow <u>H</u> elp	
+++ UDP10GIP Send Mode +++ Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFF8 => 0x10000000 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)	
	Ŧ

Figure 5-4 Error from invalid transfer size

<u>2</u> C	0M9 - 1	Tera Terr	m VT								-	×
<u>F</u> ile	Edit	<u>S</u> etup	Control	Window	<u>H</u> elp							
++++ Ente Out- [0] [1] [2] [3] [4]	UDP10 of tra of -ra UDP10 : Sho : Res : Ser : Rec : Ful	DGIP S ansfer cket s ange i DGIP m DW UDP set UD nd Dat ceive 11 dup	end Mo ize (a nput 10GIP P10GIP a Test Data Te lex Te	de +++ (aligned ligned paramet paramet (UDP10 est (Tai st (UDP:	l to 64-k to 64-bit ter GIP -> Ta rget -> U LØGIP <->	bit) : t) : JDP10G > Targ	8 - 8 - IP> et>	ØxFFFFFF 8968	?8 => =>	Ø×FFFF 8969	FFFF8	•

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 0xFFFFFF8. 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.





Figure 5-7 Receive data test when enable data verification

[3] : Receive Data Test (Target -> UDP10GIP) [4] : Full duplex Test (UDP10GIP <-> Target)



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 "udpdatatest" 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 0xFFFFFF8. 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 "udpdatatest".
 - >> udpdatatest [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.



🧧 COM11 - Tera Term VT	Call udpdatatest with recommended parameters to receive data
Eile Edit Setu Input '4' to run full duplex test	Administrator: C:\windows\system32\cma.exe 3
+++ UDP10GIP Full-duplex Mode +++ Input parameter for full duplex Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Input parameter for full duplex Enter packet size (aligned to 64-bit) : 8 - 8968 => 8968 Enable data verification : [0] Disable [1] Enable => 1 Run two test application on PC by following command	Start Receiving with Data pattern FPGA IP: 192.168.7.42:4000 PC Port: 60000 Timeout: 100 mSec 1.240 GB 2.482 GB 4
1) PC receive data test (UDP10GIP -> PC) udpdatatest r 192.168.7.42 4000 60000 4294967288	3.723 GB
2) PC send data test (PC -> UDP10GIP) udpdatatest t 192.168.7.42 4000 61000 4294967288	Receiving Data Rate: 1239.89 MByte(s)/Sec 5
Press any key to start data transfer Display recommended Start data transferring Send 1240 MByte Recv 977 MByte Send 2356 MByte Recv 1713 MByte Send 3472 MByte Recv 2228 MByte Send 4294 MByte Recv 2859 MByte Send 4294 MByte Recv 3741 MByte Transfer data complete	Call udpdatatest with recommended parameters to send data C. WINDOWS System 32 (CINALEXE) D:\Share udpdatatest t 192.168.7.42 4000 61000 4294967288 Start Sending with Data pattern FPGA IP: 192.168.7.42:4000
Total tx transfer size = 536870911 QHord (64-bit) Total = 4294(HB) , Time = 5300[ms] , Transfer speed = 810[HB/s]	PC Port: 61000 Timeout: 100 mSec 815.604 MB Display transfer size every second
Total rx transfer size = 536870911 QHord (64-bit) Total = 4294[HB] , Time = 5300[ms] , Transfer speed = 810[HB/s]	2.103 GB 2.780 GB 2.780 GB 2.780 CP Complete to transfer all data
UDP1DGIP menu [0] : Show UDP1DGIP parameters [1] : Reset UDP1DGIP parameter [2] : Send Data Test (UDP1DGIP -> Target) [3] : Receive Data Test (Target -> UDP1DGIP) [4] : Full duplex Test (UDP1DGIP <-> Target)	Spend 5.54 Second(s) for Sending 5 Sending Data Rate: 775.68 MByte(s)/Sec

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

	(
🚾 COM11 - Tera Term VT	Call udpdatatest with recommended parameters to receive data
Eile Edit Setu Input '4' to run full duplex test	D:\Share>udpdatatest r 192.168.7.42 4000 60000 4294967288
File Edit Setu Input 4 to run duplex test +++ UDP10GIP Full-duplex Mode +++ Input parameter for full duplex Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Enable data verification : [0] Disable [1] Enable => 0 Run tuo test application on PC by following command 1) PC receive data test (UDP10GIP -> PC) udpdatatest r 192.168.7.42 4000 60000 4294967288 2) PC send data test (PC -> UDP10GIP) udpdatatest t 192.168.7.42 4000 61000 4294967288 Press any key to start data transfer Start data transferring Send 2356 MByte Recv 2665 MByte Send 3472 MByte Recv 3709 MByte Transfer data complete Ital tx transfer size = 536870911 QHord (64-bit) Total tx transfer size = 536870911 QHord (64-bit) Total = 4294(MB) , Time = 3600[ns] , Transfer speed = 1193(MB/s] UDP10GIP nenu (0) : Shou UDP10GIP parameters (1) : Reset UDP10GIP nenu (2) : Send Data Test (UDP10GIP -> Target) (3) : Receive Data Test (UDP10GIP <-> Target)	D:\Share>udpdatatest 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 1.232 GB Software is stopped by timeout condition. Total received data is less than set value. Ist Data drop 0x0049ff26, Last data 0x3ceab012 Total Data Drop 25908552 Byte(s) Spend 3.56 Second(s) for Receiving 4.269 GByte(s) 5 Receiving Data Rate: 1197.49 MByte(s)/Sec Start Sending with Data pattern FPGA IP: 192.168.7.42:4000 PC Port: 61000 Timeout: 100 mSec 1.168 GB 2.331 GB 3.493 GB Spend 3.69 Second(s) for Sending 4.295 GByte(s) Spend 3.69 Second(s) for Sending 4.295 GByte(s) Sending Data Rate: 1164.26 MByte(s)/Sec
1	

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.

- Two FPGA development boards (ZCU102/KCU105) <u>Note</u>: 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







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.

Image: COM14 - Tera Term VT Blue: Output to console File Edit Setup Control	
UDP10GIP with CPU Demo [Ver = 1.3] Input mode : [0] Client [1] Server => 1	-
+++ 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	Default parameters for server mode
Press 'x' to skip parameter setting X Use	default parameter

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.



<u>Note:</u> 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.

🚾 COM14 - Tera Term VT	COM9 - Tera Term VT
<u>File</u> Edit Set Current parameter on server	<u>Eile Edit Setu</u> Current parameter on client
+++ Current Network Parameter +++	+++ Current Network Parameter +++
Mode = SERUER	Mode = CLIENT
FPGA MAC address = 0x001122334455	FPGA MAC address = 0×000102030405
FPGA IP = 192.168.7.25	FPGA IP = 192.168.7.42
FPGA port number = 60000	FPGA port number = 4000
Target IP = 192.168.7.42	Target IP = 192.168.7.25
Target port number (Target->FPGA) = 4000	Target port number (Target->FPGA) = 60000
Target port number (FPGA->Target) = 4000	Target port number (FPGA->Target) = 60000
UDP10GIP menu	UDP10GIP menu
[0] : Show UDP10GIP parameters	[0] : Show UDP10GIP parameters
[1] : Reset UDP10GIP parameter	[1] : Reset UDP10GIP parameter
[2] : Send Data Test (UDP10GIP -> Target)	[2] : Send Data Test (UDP10GIP -> Target)
[3] : Receive Data Test (Target -> UDP10GIP)	[3] : Receive Data Test (Target -> UDP10GIP)
[4] : Full duplex Test (UDP10GIP <-> Target)	[4] : Full duplex Test (UDP10GIP <-> Target)

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.

<u>Note:</u>

- 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.





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 0xFFFFFF8. 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 0xFFFFFF8. 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).

Input '3' to run receive data test	Input '2' to run send data test Blue: Output to console Green: User input
<u>File Edit Setup Control Window Help</u>	Eile Edit Setup Control Window Herp
Input receiver parameter Input receiver Input receiver	+++ UDP1DGIP Send Mode +++ Input sender parameter Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFFF8 Enter packet size (aligned to 64-bit) : 8 - 8968 => 1472
Run test application on PC by following command udpdatatest t 192.168.7.25 60000 4000 4294967288 Hait data from Target Send 0 Byte Recv 1192 HByte Send 0 Byte Recv 2265 HByte 5 Send 0 Byte Recv 3337 HByte 5 Send 0 Byte Recv 4294 HByte 6 Receive data completed 6 Total tx transfer size = 0 QHord (64-bit) Total = 0[B] , Time = 3800[ms] , Transfer speed = 0[HB/s] Total rx transfer size = 536870911 QHord (64-bit) Total = 4294[HB] , Time = 3800[ms] , Transfer speed = 1130[HB/s] UDP10GIP menu [0] : Show UDP10GIP parameters [1] : Reset UDP10GIP parameters [1] : Reset UDP10GIP parameter [2] : Send Data Test (UDP10GIP -> Target) [3] : Receive Data Test (Target -> UDP10GIP) [4] : Full duplex Test (UDP10GIP <-> Target)	Run test application on PC by following command udpdatatest r 192.168.7.42 4000 60000 4294967288 Press any key to start data sending 4 Start data sending Send 1191 MByte Recv 0 Byte Send 2383 MByte Recv 0 Byte Send 3575 MBute Recv 0 Byte Send data complete Total tx transfer size = 536870911 QHord (64-bit) Total = 4294[MB], Time = 3605[ms], Transfer speed = 1191[MB/s] Total = 0[B], Time = 3605[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 (UDP10GIP (-> Target) [4] : Full duplex Test (UDP10GIP (-> Target)
Figure 8-3 Send/receive data	test by using non-jumbo frame
Input '3' to run receive data test	Input '2' to run send data test
<u>File Edit Setup Control Window H</u> elp	<u>File Edit Setup Control Window H</u> elp
+++ UDP10GIP Receive Mode +++ Enter transfer size (aligned to 64-bit): 8 - DxFFFFFFF8 => DxFFFFFF8 Enable data verification : [0] Disable [1] Enable => 1	+++ UDP10GIP Send Mode +++ Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Enter packet size (aligned to 64-bit) : 8 - 8968 => 8968 3 Do to to to size (aligned to 64-bit) : 8 - 8968 => 8968 3
Run test application on PC by following command udpdatatest t 192.168.7.25 60000 4000 4294967288	udpdatatest r 192.168.7.42 4000 60000 4294967288
Hait data from Target Send D Byte Recv 1240 HByte Send D Byte Recv 2357 HByte Send D Byte Recv 3473 HByte Receive data completed Total tx transfer size = 0 QHord (64-bit) Total = 0[B], Time = 3600[ms], Transfer speed = 0[HB/s] Total rx transfer size = 536870911 0Hord (64-bit)	Press any key to start data sending Start data sending Send 1240 MByte Recv 0 Byte Send 3720 MByte Recv 0 Byte Send data complete Total tx transfer size = 536870911 QHord (64-bit) Total = 4294(MB), Time = 3465[ms], Transfer speed = 1239(MB/s]
Total = 4294(HB) , Time = 3600(ms) , Transfer speed = 1193(HB/s)	Total rx transfer size = 0 QWord (64-bit) Total = D[B] , Time = 3465[ms] , Transfer speed = D[MB/s]
UDP10GIP непи [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)	UDP10GIP nenu [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)



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 0xFFFFFF8. 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 0xFFFFFF8. 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.



🔟 COM11 - Tera Term VT	COM5 - Tera Term VT Blue: Output to console
<u>File Edit Setup Cor Server</u> ndow <u>H</u> elp	File Edit Setup Cor Client Idow He
+++ UDP10GIP Full-duplex Mode +++ Input server parameter Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Enter packet size (aligned to 64-bit) : 8 - 8968 => 1472 Enable data verification : [0] Disable [1] Enable => 1	+++ UDP10GIP Full-duplex Mode +++ Input client parameter Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Enter packet size (aligned to 64-bit) : 8 - 8968 Enter packet size (aligned to 64-bit) : 8 - 8968 Enable data verification Enable data verification
Run two test application on PC by following command	Run two test application on PC by following command
1) PC receive data test (UDP10GIP -> PC) udpdatatest r 192.168.7.25 60000 4000 4294967288	1) PC receive data test (UDP10GIP -> PC) udpdatatest r 192.168.7.42 4000 60000 4294967288
2) PC send data test (PC -> UDP10GIP) udpdatatest t 192.168.7.25 60000 4000 4294967288	2) PC send data test (PC -> UDP10GIP) udpdatatest t 192.168.7.42 4000 60000 4294967288
Hait data from Target 2 Wait data from Target Start data transferring 2 Send 1191 HByte Recv 1195 HByte Send 2264 HByte Recv 2267 HByte 5 Send 3263 HByte Recv 3340 HByte 5 Send 4294 HByte Recv 4294 HByte 6 Iransfer data complete 6	Press any key to start data transfer 4 Start data transferring Send 1191 HByte Recv 1190 HByte Send 2264 HByte Recv 2263 HByte Send 3336 HByte Recv 3335 HByte Send 4294 HByte Recv 4294 HByte Iransfer data complete
Total tx transfer size = 536870911 QHord (64-bit) Total = 4294[MB] , Time = 3800[ms] , Transfer speed = 1130[MB/s]	Total tx transfer size = 536870911 QHord (64-bit) Total = 4294[HB] , Time = 3800[ms] , Transfer speed = 1130[HB/s]
Total rx transfer size = 536870911 QHord (64-bit) Total = 4294[MB] , Time = 3800[ms] , Transfer speed = 1130[MB/s]	Total rx transfer size = 536870911 QWord (64-bit) Total = 4294[MB] , Time = 3800[ms] , Transfer speed = 1130[MB/s]
UDP10GIP menu [0] : Shou UDP10GIP parameters [1] : Reset UDP10GIP parameter [2] : Send Data Test (UDP10GIP -> Target) [3] : Receive Data Test (Target -> UDP10GIP) [4] : Full duplex Test (UDP10GIP <-> Target)	6 ID: Shou U0P10GIP parameters I1: Reset U0P10GIP parameters I2: Send Data Test (U0P10GIP -> Target) I3: Receive Data Test (Target -> U0P10GIP) I4: Full duplex Test (U0P10GIP <-> Target)

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

🚾 COM11 - Tera Term VT	COM5 - Tera Term VT Blue: Output to console	
<u>Eile E</u> dit <u>S</u> etup Co Server (indow <u>H</u> elp	<u>File Edit Setup Co Client</u> indow H	
+++ UDP10GIP Full-duplex Mode +++ Input jumbo frame Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Enter packet size (aligned to 64-bit) : 8 - 8968 Enable data verification : [0] Disable [1] Enable => 1	+++ UDP10GIP Full-duplex Mode +++ Input jumbo frame Enter transfer size (aligned to 64-bit) : 8 - 0xFFFFFFF8 => 0xFFFFFF8 Enter packet size (aligned to 64-bit) : 8 - 8968 Enable data verification : [0] Disable [1] Enable => 1	
Run two test application on PC by following command	Run two test application on PC by following command	
1) PC receive data test (UDP10GIP -> PC)	1) PC receive data test (UDP10GIP -> PC)	
udpdatatest r 192.168.7.25 60000 4000 4294967288	udpdatatest r 192.168.7.42 4000 60000 4294967288	
2) PC send data test (PC -> UDP10GIP)	2) PC send data test (PC -> UDP1DGIP)	
udpdatatest t 192.168.7.25 60000 4000 4294967288	udpdatatest t 192.168.7.42 4000 60000 4294967288	
Hait data from Target	Press any key to start data transfer	
Start data transferring	Start data transferring	
Send 1240 HByte Recv 1243 HByte	Send 1240 HByte Recv 1239 HByte	
Send 2356 HByte Recv 2359 HByte	Send 2356 HByte Recv 2355 HByte	
<u>Send 3472 HByte Recv 3475 HByte</u> 6	Send 3472 HByte Recv 3471 HByte	
Transfer data complete	Transfer data complete	
Total tx transfer size = 536870911 QHord (64-bit)	Total tx transfer size = 536870911 QHord (64-bit)	
Total = 4294[HB] , Time = 3600[ms] , Transfer speed = 1193[HB/s]	Total = 4294(MB) , Time = 3600[ms] , Transfer speed = 1193[MB/s]	
Total rx transfer size = 536870911 QHord (64-bit)	Total rx transfer size = 536870911 QHord (64-bit)	
Total = 4294[HB] , Time = 3600[ns] , Transfer speed = 1193[HB/s]	Total = 4294[MB] , Time = 3600[ms] , Transfer speed = 1193[MB/s]	
UDP1DGIP menu	UDP1DGIP nenu	
[0] : Show UDP1DGIP parameters	[0] : Shoμ UDP10GIP parameters	
[1] : Reset UDP1DGIP parameter	[1] : Reset UDP10GIP parameter	
[2] : Send Data Test (UDP1DGIP -> Target)	[2] : Send Data Test (UDP10GIP -> Target)	
[3] : Receive Data Test (Target -> UDP1DGIP)	[3] : Receive Data Test (Target -> UDP10GIP)	
[4] : Full duplex Test (UDP1DGIP <-> Target)	[4] : Full duplex Test (UDP10GIP <-> Target)	

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