UDP10G-IP Introduction (Intel)

Super UDP Speed by hard-wired IP-Core

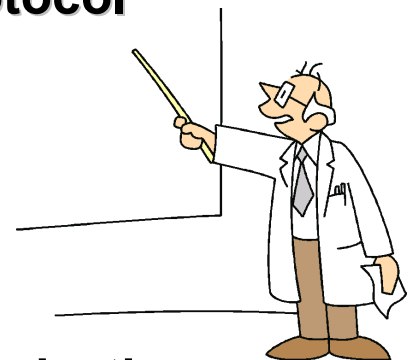---

Agenda

- **Merit and demerit of UDP protocol**
- **UDP10G-IP core overview**
- **UDP10G-IP core description**
    - **Initialization**
    - **High-speed transmit**
    - **High-speed reception**
- **User I/F, Buffer size parameterization**
- **Reference design**
- **Resource usage and real performance**
- **Application example**

Merit and demerit of UDP protocol

- ## Merit
  - **High-speed and low-latency by minimum overhead**
  - **Supports 1-to-N multicast and 1-to-All broadcast**
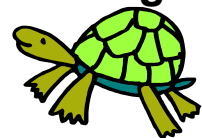 - **Suitable for real-time application such as VOD system**
- ## Demerit
  - **No ACK/retransmit, so data reliability is not guaranteed**
  - **If reliability is necessary, application layer must support it**

UDP implementation problem by CPU

- ## Problem in performance and latency
  - **CPU resource consumption by UDP packet building**
    - Check-sum calculation
    - Concatenate header and transmit data
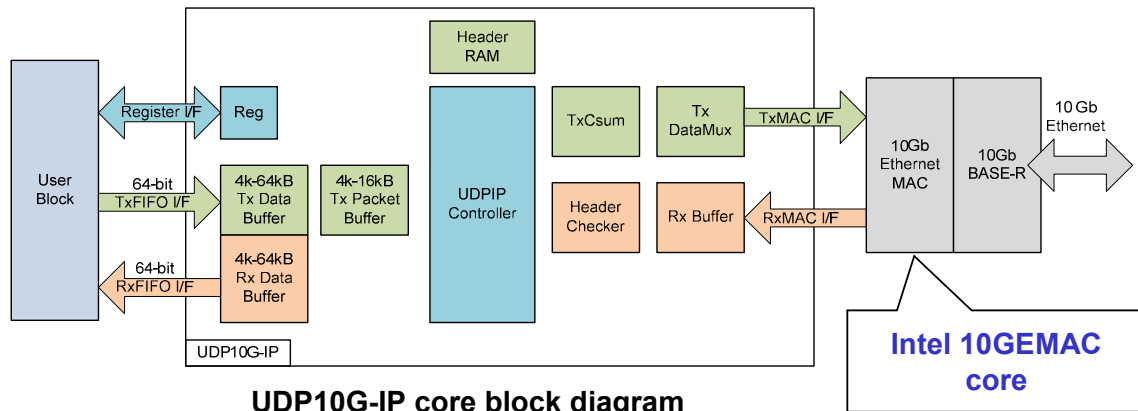 - **Bandwidth is not stable due to firmware process**
- ## The problem gets even worse with full duplex
  - **CPU needs to process time sharing between Tx&Rx**
  - **Bandwidth and latency further drops**
  - **Fatal problem for real time application**

  UDP10G-IP core can provide ideal solution!

UDP10G-IP core Overview

- **Fully hard-wired UDP control for both Tx and Rx**
- **Inserts between user logic and Intel 10GEMAC module**
- **Supports Full Duplex communication**

UDP10G-IP core block diagram

UDP10G-IP core Advantage 1

- **Fully hard-wired UDP protocol control**
  - Possible to build CPU-less network system
  - Zero load for CPU

- **Support all of Tx only, Rx only, and full-duplex**
  - More than 1200MByte/sec real performance

- **Can even keep some data reliability**
  - Tx: Calculate check sum and build header automatically
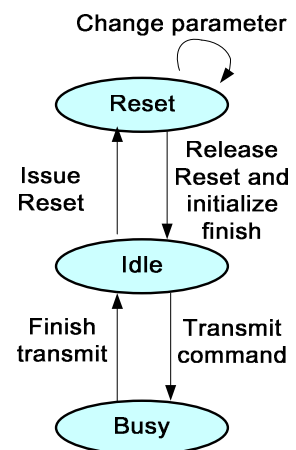 - Rx: Discard received Packet if check sum does not match

UDP10G-IP core Advantage 2

- ## Selectable data buffer size
  - Selectable buffer size of memory usage vs. performance

- ## Supports IP fragment packet reception
  - Receive IP fragment packet when packet order is correct

- ## Reference design on Intel evaluation board
  - Full QuartusII project for standard Intel board
  - Free sof-file for evaluation before purchase
  - All source code (except IP-core) in design project

- ## Can support multicast/broadcast transmission
  - Provided by IP-core customization service

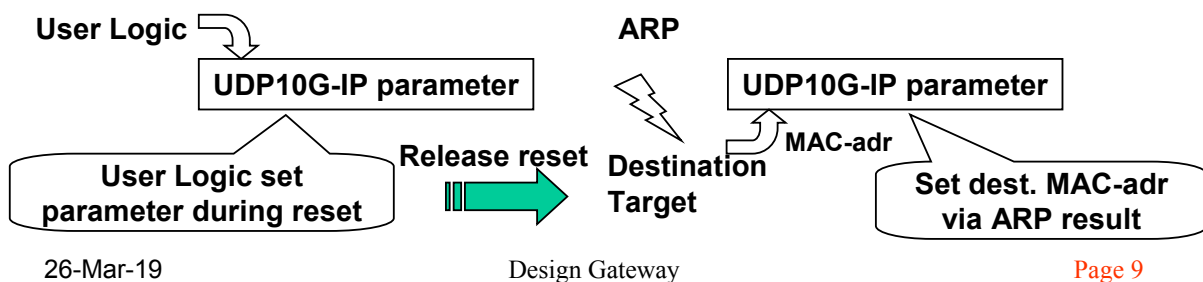
UDP10G-IP core Operation

- **Set parameter (IP-adr&MAC-adr, etc) during Reset**
- **Release Reset then initialize including ARP**
- **Idle state after initialization finish, wait command**
- **Tx operation starts by user command**
- **Rx operates at any time except Reset state
  (Accepts all Rx packet if parameter match)**
- **Tx and Rx operates individually (full duplex)**
- **If want change parameter, move to Reset state
  (transfer/packet length can change except Busy)**

Change parameter

Reset

Issue Reset   Release Reset and initialize finish

Idle

Finish transmit   Transmit command
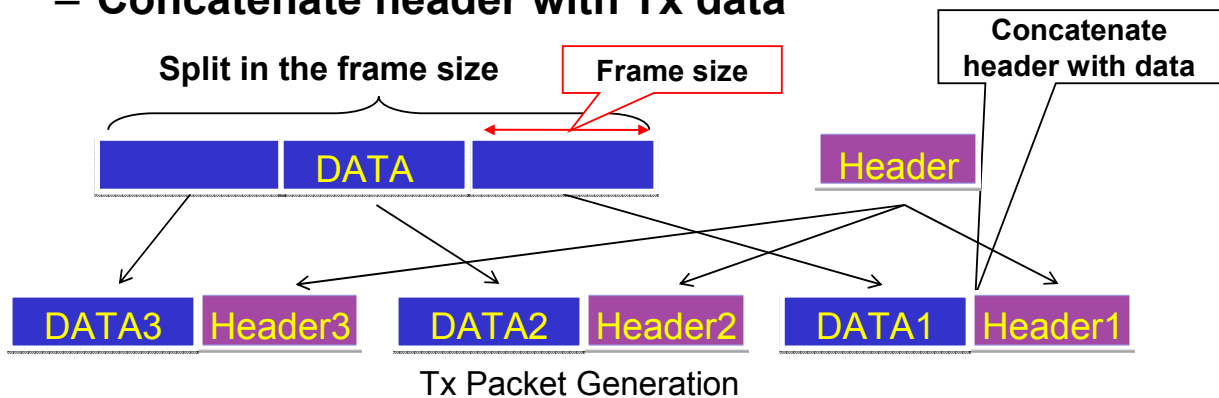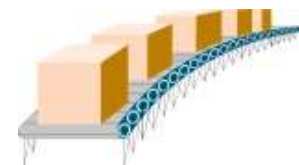
Busy

State Diagram

UDP10G-IP Initialization

- **Set parameter to UDP10G-IP**
  - **User logic can set parameter during UDP10G-IP reset**
  - **Set IP address, MAC address, and Port number**
  - **Release reset after parameter setting finish**
- **UDP10G-IP executes ARP after reset release**
  - **Issue ARP to destination target when Client mode**
  - **Wait ARP reception when Server mode**
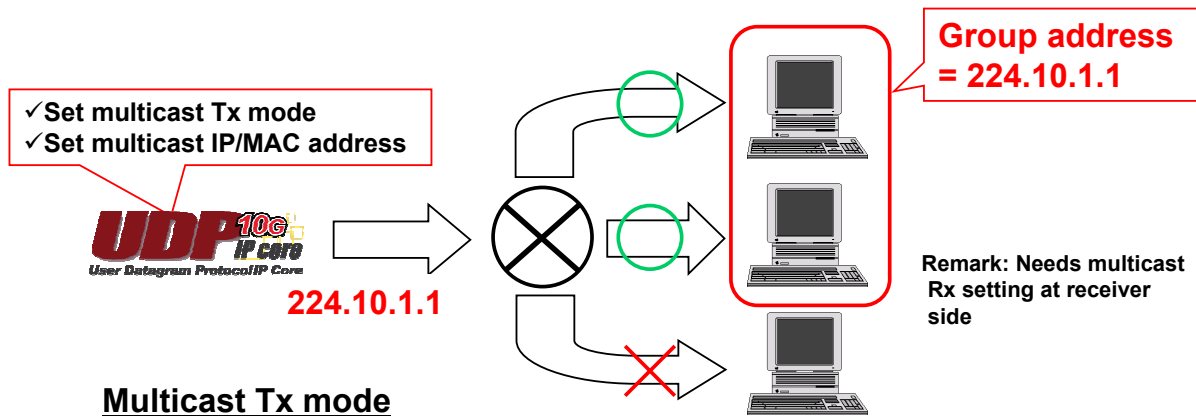
User Logic

UDP10G-IP parameter

ARP

UDP10G-IP parameter

MAC-adr

User Logic set parameter during reset

Release reset

Destination Target

Set dest. MAC-adr via ARP result

26-Mar-19   Design Gateway   Page 9

- **Multicast/broadcast Tx via customization**
  - **Suppress automatic ARP execution**
  - **Set multicast IP/MAC address from user logic**

Group address = 224.10.1.1

✓Set multicast Tx mode
✓Set multicast IP/MAC address

224.10.1.1

Remark: Needs multicast Rx setting at receiver side

Multicast Tx mode

26-Mar-19   Design Gateway   Page 11

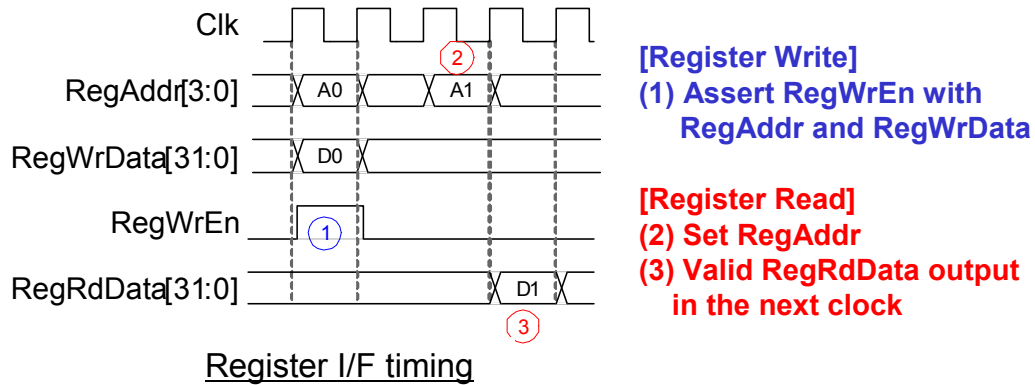# High-Speed Rx

- **Rx packet header check**
  - **Verify all of MAC, IP, and UDP header**
  - **Receive IP fragment packet when order is correct**
- **Check sum calculation and verification**
  - **Calculate check sum in received packet**
  - **Verify calculated value with header value**
  - **When mismatch, packet data is discarded**

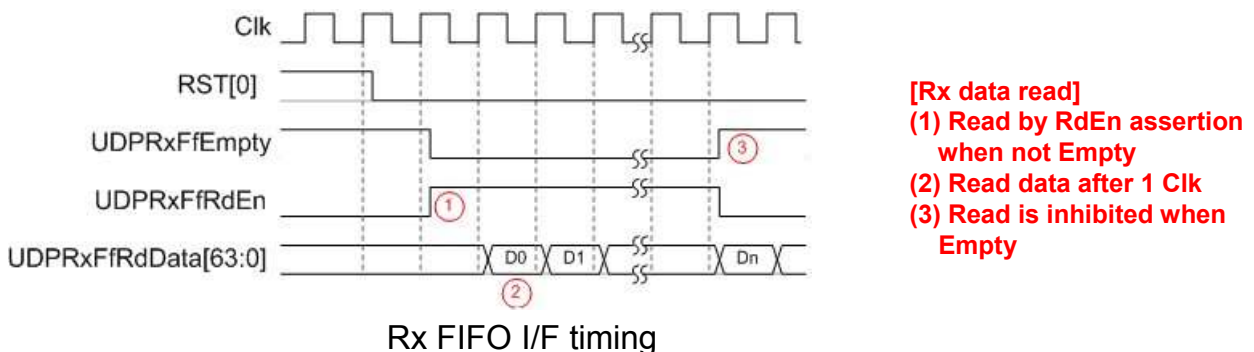26-Mar-19   Design Gateway   Page 12

User Interface (Control)

- **3 types of Register I/F, Tx FIFO I/F, and Rx FIFO I/F**
  - **Register I/F for initial parameter setting and Tx/Rx command**
  - **Tx FIFO I/F and Rx FIFO I/F is standard FIFO interface**

Register I/F timing

[Register Write]
(1) Assert RegWrEn with RegAddr and RegWrData

[Register Read]
(2) Set RegAddr
(3) Valid RegRdData output in the next clock

User Interface (Data)

Tx FIFO I/F timing

[Tx data write]
(1) Check FIFO is not full
(2) Write data with WrEn
(3) Suspend write within 4 Clk after Full assertion

Rx FIFO I/F timing

[Rx data read]
(1) Read by RdEn assertion when not Empty
(2) Read data after 1 Clk
(3) Read is inhibited when Empty

Buffer Capacity

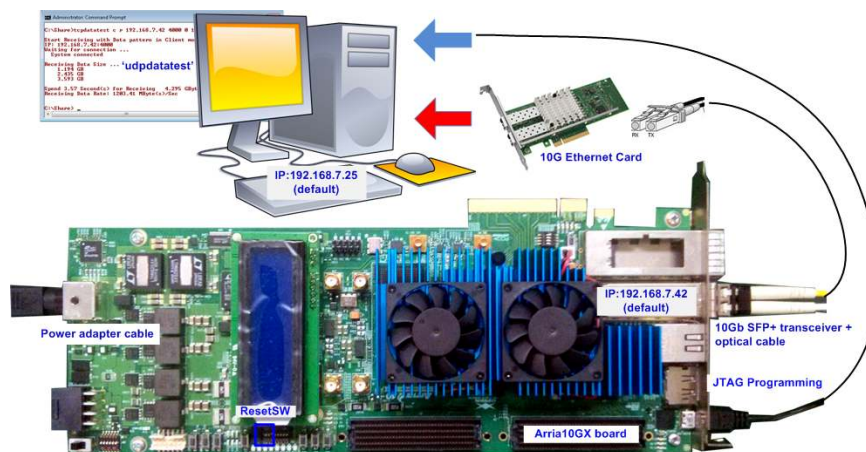- **Parameterized 3 types of data buffer**
    - **(1) Tx Data Buffer: 4KByte - 64KByte**
    - **(2) Tx Packet Buffer: 4KByte - 16KByte**
    - **(3) Rx Data Buffer: 4KByte - 64KByte**
- **User can optimize resource usage and performance**

Generic Name	Range	Description
TxBufBitWidth	**9-13**	**Set Tx data buffer size in address bit width** **When set to 9, size is 4KByte, when 13, 64KByte for example.**
TxPacBitWidth	**9-11**	**Set Tx packet buffer size in address bit width** **When set to 9, size is 4KByte, when 11, 16KByte for example**
RxBufBitWidth	**9-13**	**Set Rx data buffer size in address bit width** **When set to 9, size is 4KByte, when 13, 64KByte for example.**

Buffer size is selectable by parameterization

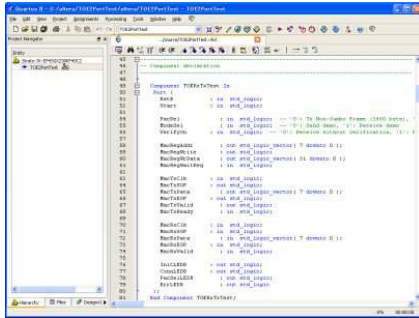# Free SOF File for Evaluation

- **SOF file for evaluation with Intel standard board**
    - **Support both Half-Duplex and Full-Duplex operation**
    - **Measure transfer speed performance and data reliability**
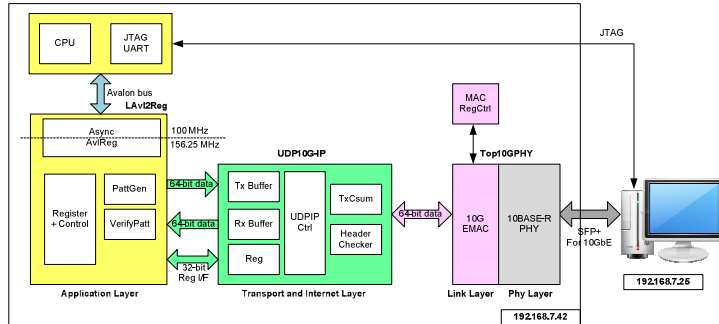

Evaluation environment for Intel board

Reference Design Overview

- ## Quartus design project for real operation
  - **Implemented into standard Intel board for each device family**
  - **IP-core deliverables include design of evaluation sof file**
  - **All source code (except IP-core) included in full project**

Quartus/Qsys project in package

Reference design block diagram

Effective Development on Ref. Design

- **Quartus project is attached to UDP10G-IP package**

- **Full source code (VHDL) except IP core**

- **Can save user system development duration**
  - **Confirm real board operation by original reference design.**
  - **Then modify a little to approach final user product.**
  - **Check real operation in each modification step.**

Short-term development is possible without big turn back

Resource Usage

- ## UDP10G-IP core standalone resource usage
  - ### Condition = Maximum buffer setting
    (TxDataBuf=RxDataBuf=64KB, TxPacketBuf=16KB)

Family	Example Device	Fmax (MHz)	ALMs	Registers	Block Memory bit
Arria10 GX	10AX115S2F45I1	156.25	1,327	1,979	1,179,648

UDP10G-IP core standalone compilation result

This result is based on maximum buffer size setting.
User can save memory resource by smaller buffer size setting

Transfer Performance

- ## Real performance in data Tx and data Rx

```
/cygdrive/d/altera/16.0

+++ UDP10G-IP Send Mode +++
Enter total tx size (aligned to 64-bit) :      0xFFFFFFF8 => 0xFFFFFFF8
Enter tx packet size : 8 - 8968 => 8968 (2)
Run udpdatatest application on PC by following command

udpdatatest r 192.168.7.42 4000 60000 4294967288
Press any key to start data sending
Send  1240.117 MB   Recv  0000.000 MB
Send  2480.185 MB   Recv  0000.000 MB
Send  3720.253 MB   Recv  0000.000 MB
Total = 4294[MB] , Time = 3463[ms] , Transfer speed = 1240[MB/s]
```
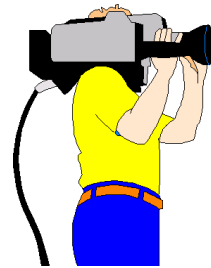
Tx (FPGA->PC) 1240MByte/sec

```
/cygdrive/d/altera/16.0

+++ UDP10G-IP Receive Mode +++
Input data verification mode : [0]-Disable [1]-
Wait data from PC
Run udpdatatest application on PC by following

udpdatatest t 192.168.7.42 4000 60001 4294967288
Send  0000.000 MB   Recv  1238.282 MB
Send  0000.000 MB   Recv  2353.038 MB
Send  0000.000 MB   Recv  3467.575 MB
Total = 4294[MB] , Time = 3600[ms] , Transfer speed = 1193[MB/s
```

Rx (PC->FPGA) 1193MByte/Sec (Rx performance depends on PC)

UDP10G-IP Application

- **Video-on-Demand via Broadcast**
  - Stream video transmission in real time
  - Requires minimum overhead and latency
  - UDP10G-IP provides best solution

- **Real time Online game**
  - Full duplex of game data download and user operation data upload
  - Very low latency required for realistic game
  - UDP10G-IP can cover full duplex within minimum latency

For more detail

- **Detailed documents available on the web site**
  - http://www.dgway.com/UDP10G-IP_A_E.html

- **Contact**
  - Design Gateway Co,. Ltd.
  - E-mail :
    ip-sales@design-gateway.com
  - FAX : +66-2-261-2290

Revision History

Rev.	Date	Description
1.0E	7-Dec-2017	English version initial release
1.1E	26-Mar-2019	Added multicast/broadcast Tx feature customization