FMC-GbE-RJ45 User Manual - Avnet ZedBoard -

Version 0 Revision 0

Oct. 20, 2018 (Oct. 10, 2018)

Future Design Systems
www.future-ds.com / contact@future-ds.com

Abstract

This manual describes FMC-GbE-RJ45 and HSR components with FMC-GbE-RJ45 using Avnet ZedBoard. FMC-GbE-RJ45 is a three-port Gigabit Ethernet board with FMC interface and FMC-GbE-RJ45 can build HSR components along with off-the-shelf FPGA board. The HSR components include RedBox and DANH.

Table of Contents

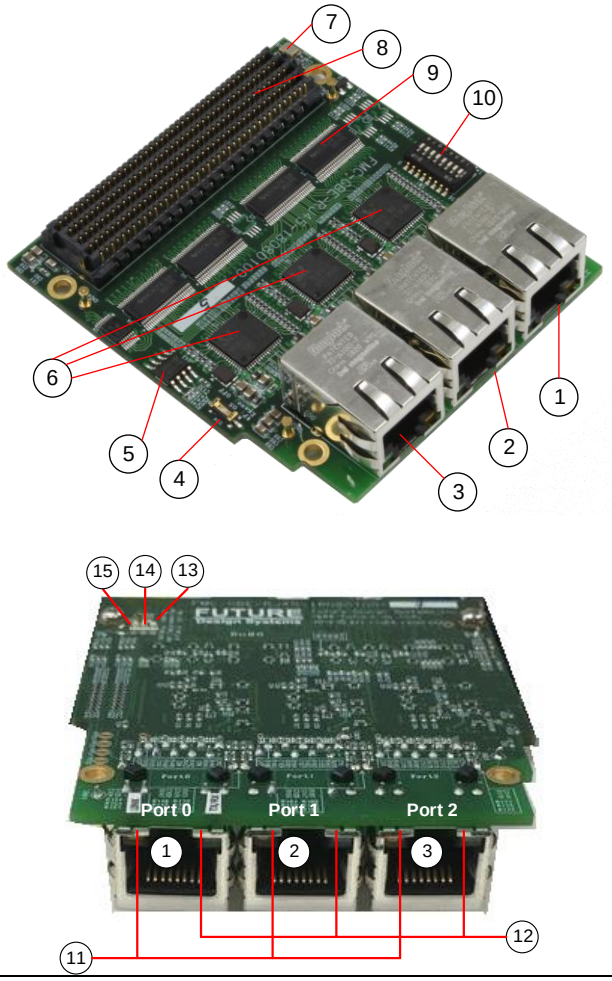
Copyright © 2018 Future Design Systems, Inc. 1
Abstract 1
Table of Contents 1
1 FMC-GbE-RJ45 3
    1.1 FPGA carrier board 4
    1.2 Other matters 4
        1.2.1 Power 4
        1.2.2 Clock 4
        1.2.3 MDIO 5
        1.2.4 GMII 5
        1.2.5 I2C EEPROM 5
2 RedBox: Redundancy Box 5
3 DANH: Double Attached Node with HSR 7
    3.1 Monitor 9
4 An example system 10
    4.1 Testing HSR between RedBoxes using packet generator 10
    4.2 Testing HSR between DANHs 11
    4.3 Testing HSR using PING 12
    4.4 Testing HSR using internet web browsing 12
    4.5 System setup example 13
5 Related document 13

6 Vivado XDC 13
6.1 RedBox 13
6.2 DANH 17
Revision history 20

1 FMC-GbE-RJ45

FMC-GbE-RJ45 is three Gigabit Ethernet board in FMC (FPGA Mezzanine Card) card format complying with ANSI/VITA 57.1-2008 standard.

Figure 1 shows FMC-GbE-RJ45 PBA (Printed Board Assembly) Version 18080100.



① GbE port 0 for upstream
② GbE port 1 for HSR Link-A
③ GbE port 2 for HSR Link-B
④ EEPROM Write Protection DIP switch
⑤ EEPROM for IPM/FRU with I2C
⑥ GbE PHY 0, 1, and 2
⑦ 125Mhz clock
⑧ FMC male connector
⑨ Level shift
⑩ 8-bit DIP switch
⑪ Link LED (Orange)
⑫ Tx/Rx LED (Green)
⑬ FMC 3.3V LED
⑭ FMC 3.3VAUX LED
⑮ FMC VADJ LED

Figure 1: FMC-GbE-RJ45 PBA board

Figure 2 shows block diagram of CON-FMC. This board uses LPC pins for minimum GMII signals and it can be connected to LPC to use all three Gigabit Ethernet ports.

- GMII GTX_CLK
- GMII TXD[7:0]
- GMII TXEN
- GMII TXER
- GMII RXC

- GMII RXD[7:0]
- GMII RXDV
- GMII RXER

With HPC supporting FPGA carrier board, all GMII signals can be utilized, which include TXC, CRC and COR.

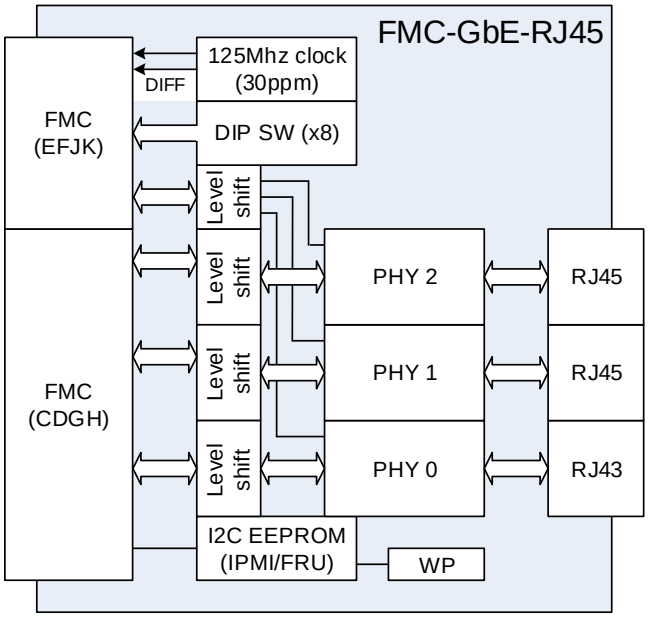


Figure 2: CON-FMC block diagram

1.1 FPGA carrier board

FMC-GbE-RJ45 can be used with FMC supporting FPGA board, which includes LPC, HPC, and HPC+. Refer to 'Section 6 Vivado XDC' for more details,

- Avnet Zedboard (AES-Z7EV-7Z020-G) with Xilinx Zynq-7000 SoC XC7Z020-CLG484-1
- Xilinx Zynq boards including ZC702, ZC706
- Xilinx Zynq UltraScale+ boards including XCU102
- Any FPGA board with FMC connector compliant with ANSI/VITA 57.1

1.2 Other matters

1.2.1 Power

FMC-GbE-RJ45 only uses power supplied through FMC, which is 3.3V. There are voltage level shift between PHY and FMC in order to support 1.2V to 3.6V with 4.6V tolerant.

1.2.2 Clock

FMC-GbE-RJ45 provides 125Mhz clock through FMC and it is LVDS signal.

1.2.3 MDIO

FMC-GbE-RJ45 provides MDIO (Media Dependent Input/Output) interface to control PHY. Each PHY has following PHY Address. MDC and MDIO signals are pulled-up. PHY ADDR pins are shared with status LEDs.

- GbE PHY for Port 0: 0x00
- GbE PHY for Port 0: 0x01
- GbE PHY for Port 0: 0x02

By default, all PHY is set to be Gigabit Ethernet mode and it requires PHY reset signal

1.2.4 GMII

FMC-GbE-RJ45 uses minimal signals of GMII (Gigabit Media Independent Interface) for Gigabit Ethernet PHY through FMC LPC pins. Other optional GMII signals are assigned at the FMC HPC pins.

1.2.5 I2C EEPROM

FMC-GbE-RJ45 has I2C EEPROM that stores IPMI/FRU information. The content should not be changed without consent from Future Design Systems.

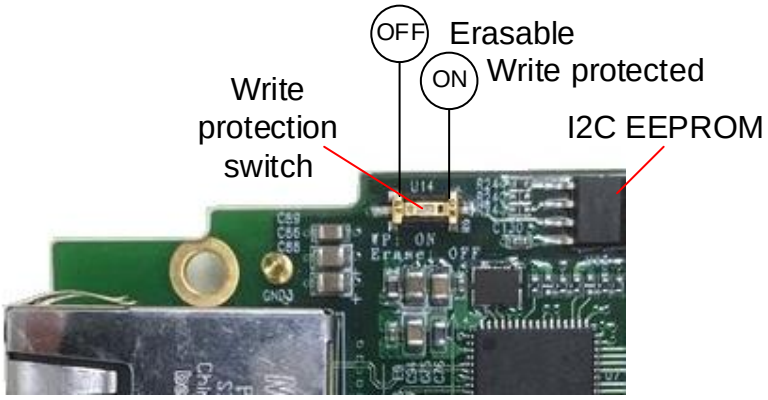


Figure 3: I2C EEPROM form IPMI

2 RedBox: Redundancy Box

Figure 4 shows an implementation of RedBox with Avnet ZedBoard. More details of HSR design (gig_eth_hsr) can be found from reference[1] [2].

[1] This technical document is available under NDA agreement with Future Design Systems and contact at 'contact@future-ds.com' for more details.

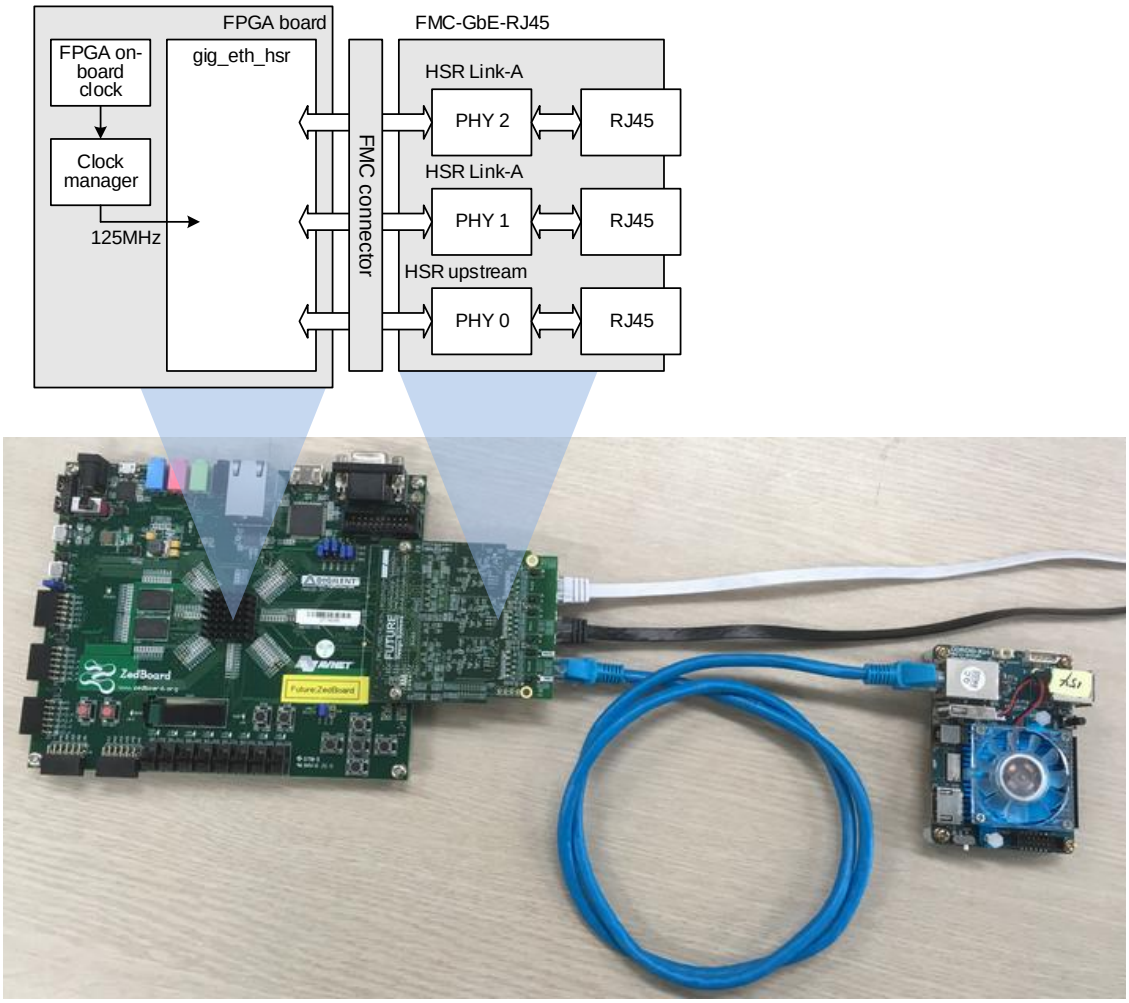


Figure 4: RedBox implementation with ZedBoard

RedBox bit-stream for ZedBoard will be delivered as follows.

- BOOT.bit: This file can be stored in the SD-Card for the ZedBoard and the board will be configured automatically when power is turn on.
- fpga.bit: This file can be downloaded to the FPGA board through JTAG using Vivado Hardware Manager.

The mode jumps should be set as shown in the Figure 5 in order to the board is bring up with the bit-stream in the SD Card.

VADJ selection jumps should be set as 2.5V.

As shown in Figure 5, The LEDs on bottom side should be lit.

- LED0 (Right-most LED): HSR is ready when lit. (Note that this LED will not let when FMC-GbE-RJ45 board is not installed)
- LED1: This board is RedBox when lit.

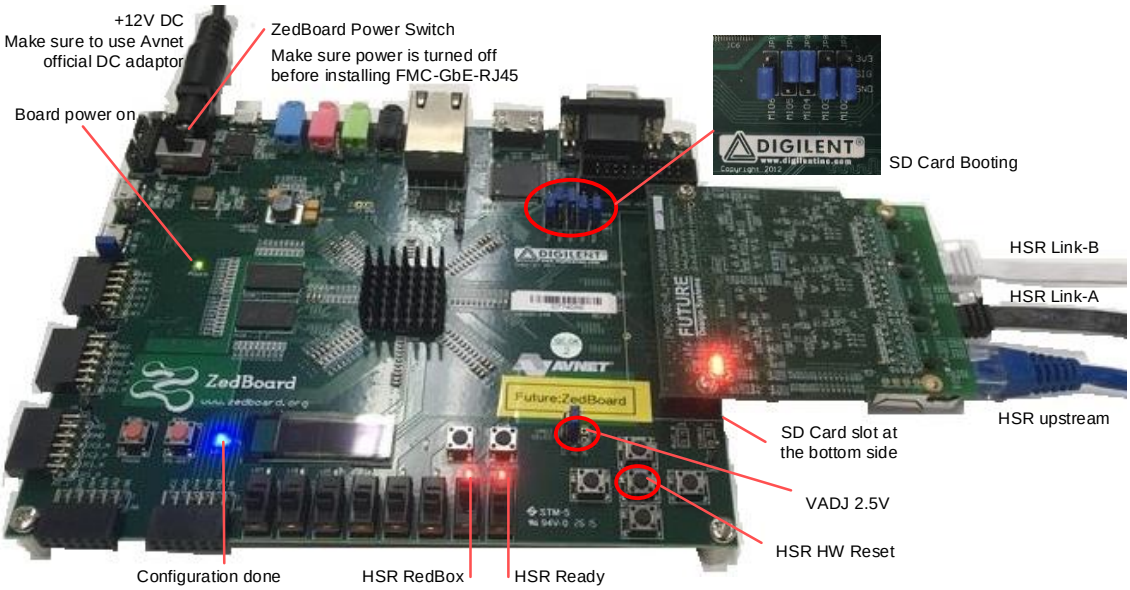


Figure 5: RedBox in operation

HSR upstream for SAN (Single Attached Node) should use GbE Port 0 and HSR Link-A & B should use GbE Port 1 & 2.

3 DANH: Double Attached Node with HSR

Figure 6 shows an implementation of DANH with Avnet ZedBoard. More details of MAC (gig_eth_mac) and HSR design (gig_eth_hsr) can be found from references[2] [1] and [2], respectively.

ARM processor in the PS region can access blocks in the PL reassign through following addresses.

- S0: 0x4000_0000 (BRAM)
- S1: 0x4100_0000 (TX BRAM)
- S2: 0x4200_0000 (RX BRAM)
- S3: 0x4300_0000 (MAC)
- S4: 0x4C00_0000 (APB)
- P0: 0x4C00_0000 (MDIO)
- P1: 0x4C01_0000 (HSR)
- P2: 0x4C02_0000 (GPIO)

[2] This technical document is available under NDA agreement with Future Design Systems and contact at 'contact@future-ds.com' for more details.

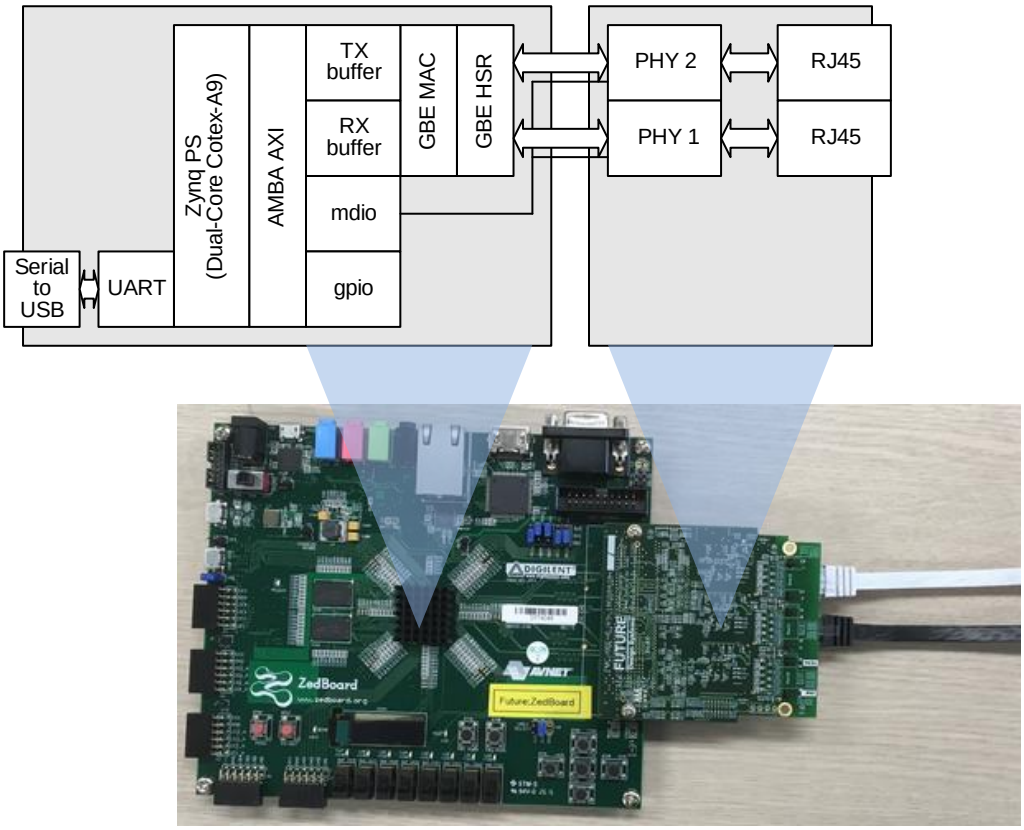


Figure 6: DANH implementation with ZedBoard

DANH bit-stream for ZedBoard will be delivered as follows.

- BOOT.bit: This file can be stored in the SD-Card for the ZedBoard and the board will be configured automatically when power is turn on.
- fpga.bit: This file can be downloaded to the FPGA board using SDK XSDB through JTAG.
- program.elf: This file can be downloaded to the FPGA board using SDK XSDB through JTAG.

The mode jumps should be set as shown in the Figure 5 in order to the board is bring up with the bit-stream in the SD Card.

VADJ selection jumps should be set as 2.5V.

DANH requires HW MAC address and 8 slide switch determines lower 8-bit of the MAC address at reset.

- Default HW MAC address: 0x20_12_34_56_78_{slide switch setting}

As shown in Figure 7, The LEDs on bottom side should be lit.

- LED0 (Right-most LED): HSR is ready when lit. (Note that this LED will not let when FMC-GbE-RJ45 board is not installed)
- LED1: It is not used for DANH.
- LED2: This board is DANH when lit.

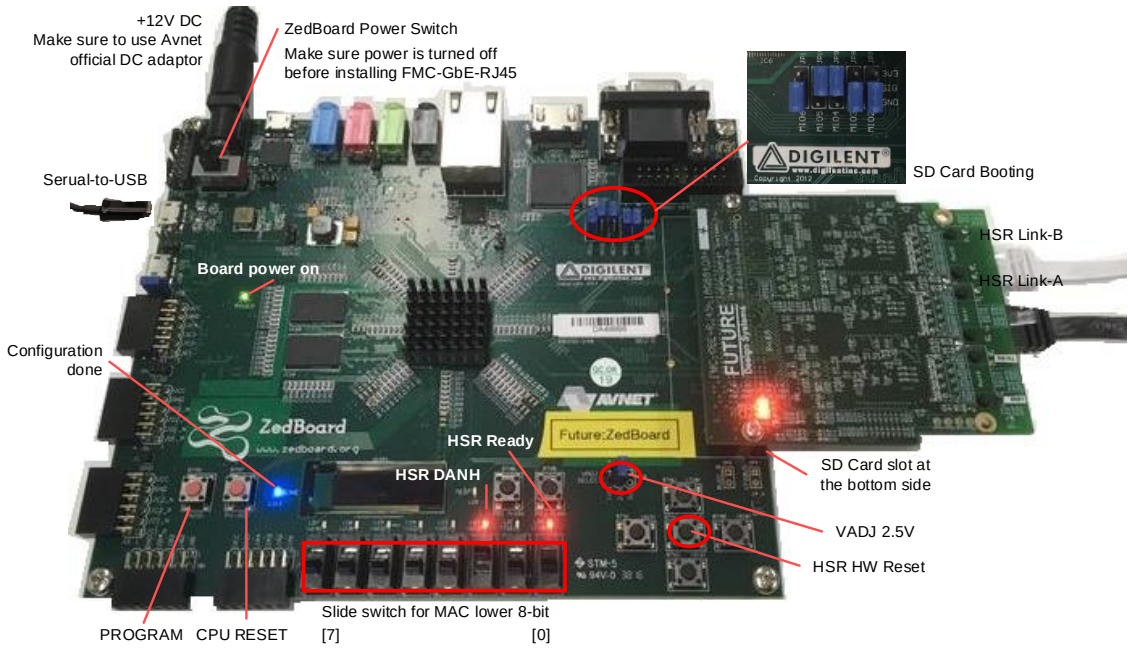


Figure 7: DANH in operation

HSR Link-A & B should use GbE Port 1 & 2.

3.1 Monitor

When power is turned on, the processor runs a simple monitor program, which provides interactive user interfaces. The monitor program uses serial-to-USB UART port with 115200 baud, 8-bit no-parity 1-stop bit none flow control.

```
monitor> help
FdsMON - 2018.10.22.
Copyright (c) 2018 by Future Design Systems
www.future-ds.com
help [-d] [-l]                          : print help message
verbose [level]                         : verbose level
mr [-w|s|b] <start_addr[:leng]>         : memory read - w for word
mw [-w|s|b] <cont> <start_addr[:leng]>  : memory write
mm [-w|s|b] <src_addr> <dst_addr> <num> : memory move
mc [-w|s|b] [-d] <addr1> <addr2> <num>  : memory compare
mt <start_addr> <end_addr> [level]      : memory test
phy_init [-a phy_addr]                  : initialize PHY (not yet)
mac_addr [-a mac_addr] [-r]             : set/get MAC address
mac_init [-a mac_addr]                  : initialize MAC
```

```
        [-b val]        //conf_tx_jumbo_en=0;
        [-c val]        //conf_tx_no_gen_crc=0;
        [-d val]        //conf_tx_bchunk=4*32;
        [-e val]        //conf_rx_jumbo_en=0;
        [-f val]       //conf_rx_no_chk_crc=0;
        [-g val]        //conf_rx_promiscuous=0;
        [-h val]        //conf_rx_bchunk=4*32;
        [-i start:size] // TX frame-buffer
        [-j start:size] // RX frame-buffer
        [-s]           //status
pkt_snd  [-a mac_src] [-b mac_dst]          : send packet
        [-n bstart[:bend]] //byte number start and end
        [-r]              //repeat
        [-t broad] [-t vlan] [-t hsr] //packet type
        [-x timeout]       //timeout (0 for blocking)
        [-v verbose]       //verbose level
pkt_rcv                                     : receive packet
        [-r]              //repeat
        [-x timeout]       //timeout (0 for blocking)
        [-v verbose]       //verbose level

monitor>
```

4 An example system

An example system can be built using two RedBoxes and two or more DANH nodes.

4.1 Testing HSR between RedBoxes using packet generator

As shown in Figure 8, packet generator attached to the one RedBox feeds in known packets of normal braodcating Ethernet frames and the other RedBox at the end of the HSR ring receives the packet.

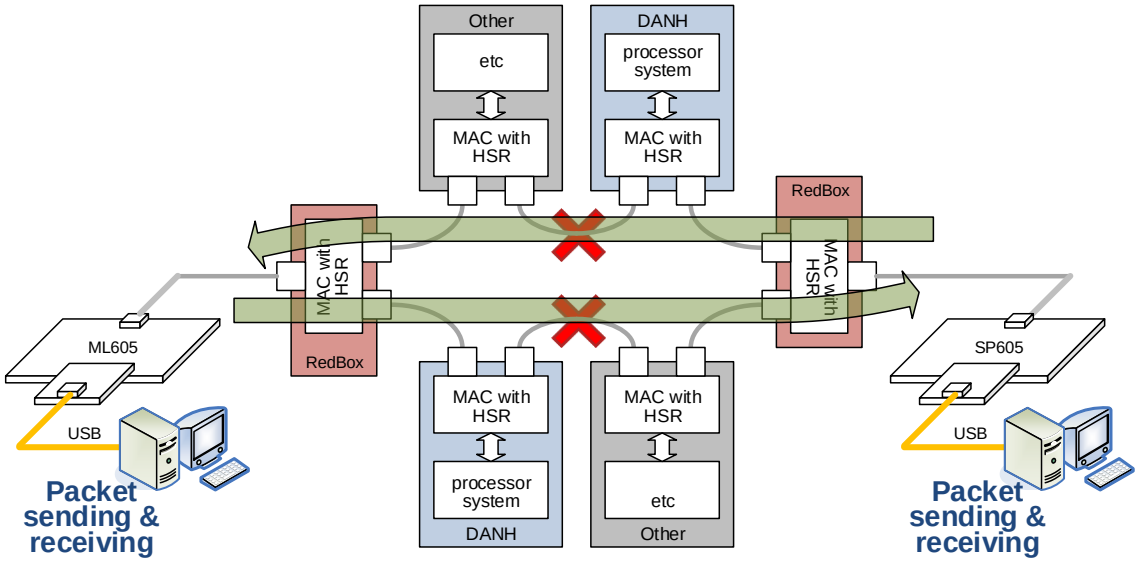


Figure 8: RedBox to RedBox testing using packet generator

During packets are transmitting, one or two links can be disconnected in order to see if HSR operation work.

4.2 Testing HSR between DANHs

As shown in Figure 9, one DANH generates packets with a specific destination MAC address and let see whether the designated DANH receives the packet or not.

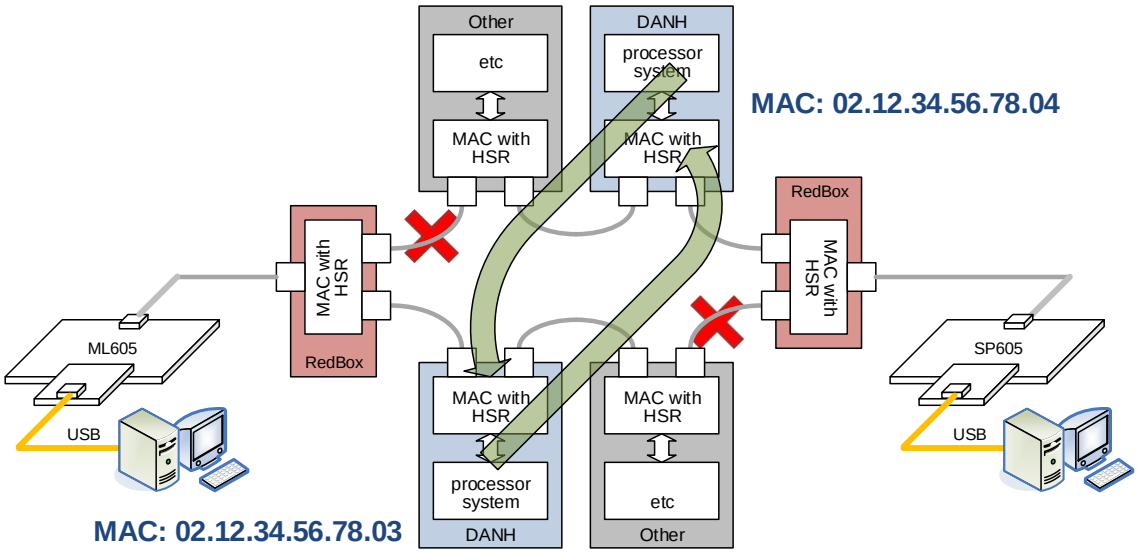


Figure 9: DANH to DANH testing using a simple monitor

During packets are transmitting, one or two links can be disconnected in order to see if HSR operation work.

4.3 Testing HSR using PING

As shown in Figure 10, two computers can be connected to the RedBoxes. One computer checks network operation by running PING, which is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network.

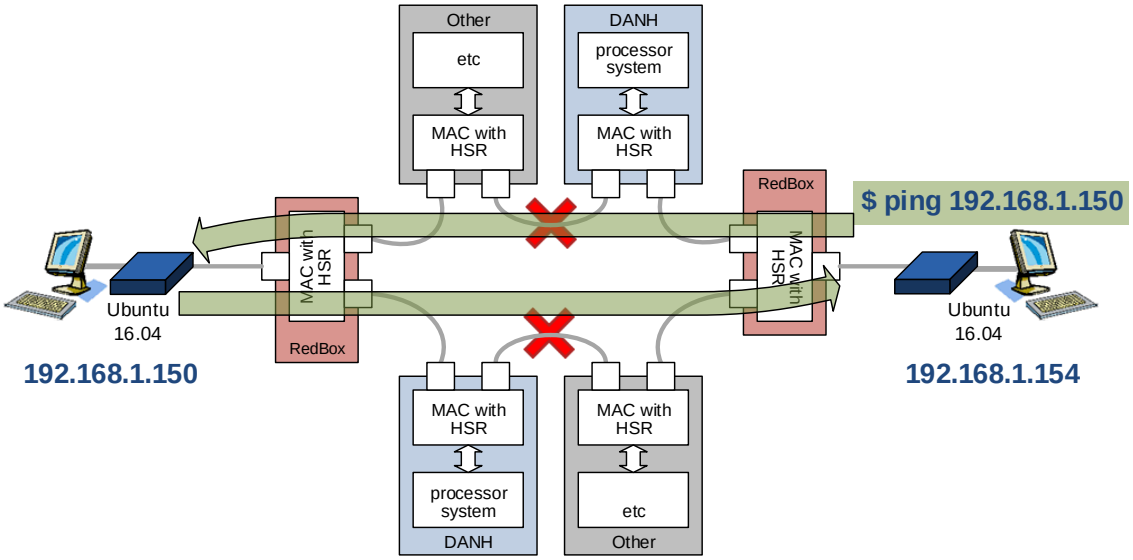


Figure 10: RedBox to RedBox testing using PING

4.4 Testing HSR using internet web browsing

As shown in Figure 11, external internet link can be connected to one RedBox and a computer connected to the other RedBox tries to access internet web sites such as Google or YouTube.

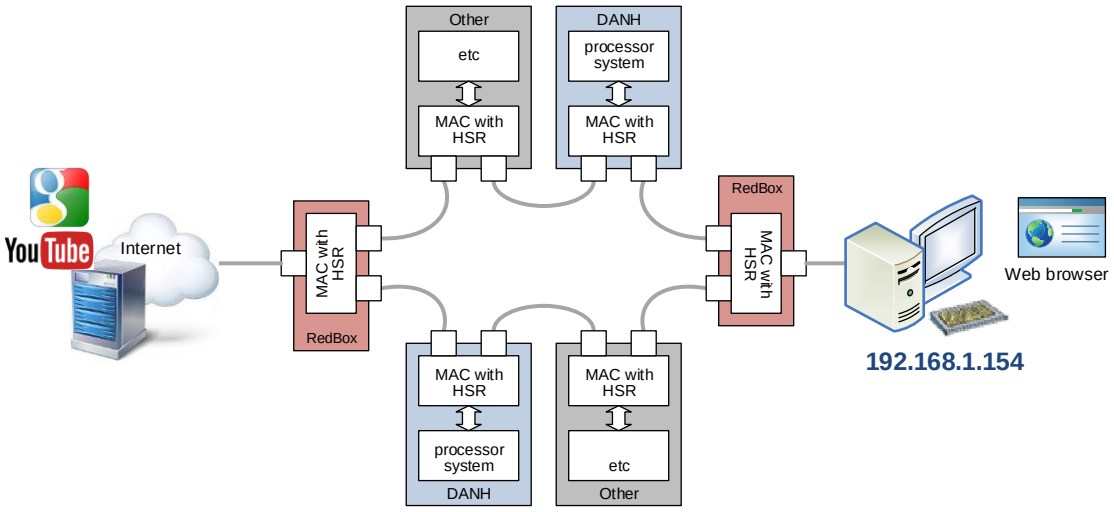


Figure 11: Internet access testing using web-browsing

4.5 System setup example

Figure 12 shows a conceptual schematic and actual setup of HSR system for testing.

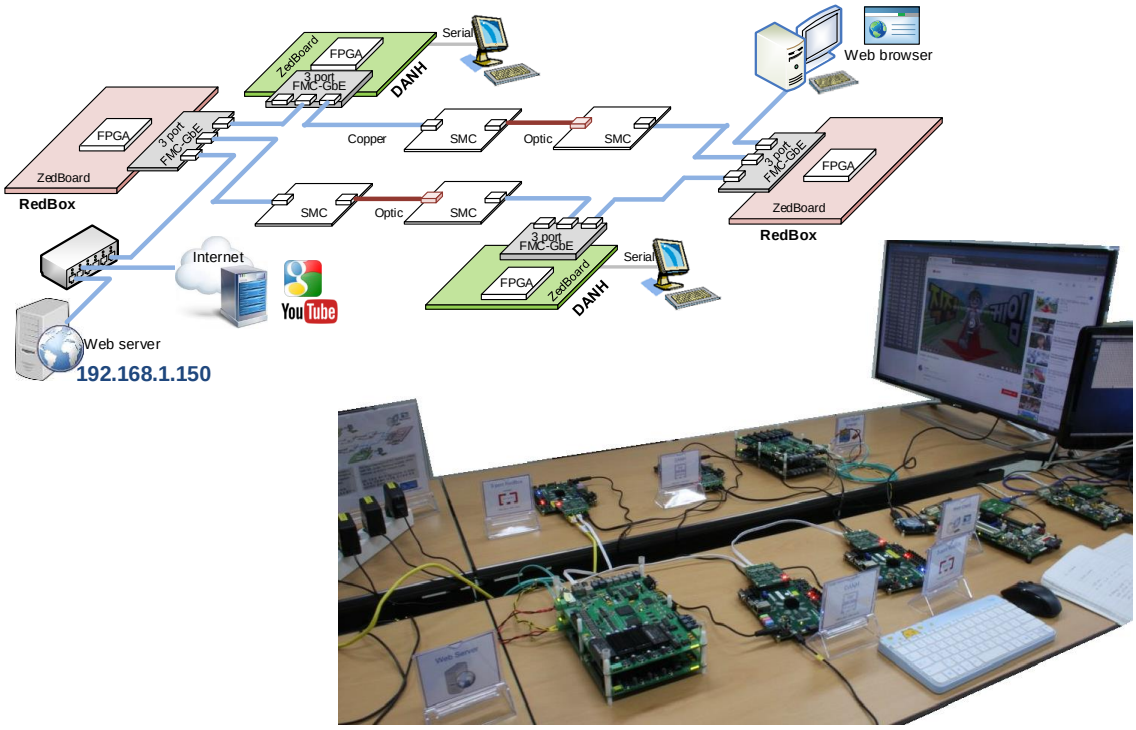


Figure 12: Actual system setup

5 Related document

[1] Future Design Systems, Gigabit Ethernet Media Access Controller, FDS-TD-2018-10-001, 2018. (Company confidential)

[2] Future Design Systems, High-avilability Seamless Redundancy Controller on Gigabit Ethernet, TD-2018-10-002, 2018. (Compay confidential)

[3] Realtek Semiconductor, RTL8211EG, Integrated 10/100/1000 Gigabit Ethernet Transceiver.

6 Vivado XDC

6.1 RedBox

```
set_property PACKAGE_PIN Y9 [get_ports CLK100] ;# 100Mhz
set_property IOSTANDARD LVCMOS33 [get_ports CLK100]
create_clock   -name CLK100 -period  10.0 [get_ports CLK100]
```

```
set_property PACKAGE_PIN P16      [get_ports BOARD_RST_SW] ;#BTNC
set_property IOSTANDARD  LVCMOS25 [get_ports BOARD_RST_SW]
set_input_delay 10 -clock [get_clocks CLK100] [get_ports BOARD_RST_SW]
set_false_path -from [get_ports BOARD_RST_SW]

set_property PACKAGE_PIN F22 [get_ports {BOARD_SLIDE_SW[0]}];  # "SW0"
set_property PACKAGE_PIN G22 [get_ports {BOARD_SLIDE_SW[1]}];  # "SW1"
set_property PACKAGE_PIN H22 [get_ports {BOARD_SLIDE_SW[2]}];  # "SW2"
set_property PACKAGE_PIN F21 [get_ports {BOARD_SLIDE_SW[3]}];  # "SW3"
set_property PACKAGE_PIN H19 [get_ports {BOARD_SLIDE_SW[4]}];  # "SW4"
set_property PACKAGE_PIN H18 [get_ports {BOARD_SLIDE_SW[5]}];  # "SW5"
set_property PACKAGE_PIN H17 [get_ports {BOARD_SLIDE_SW[6]}];  # "SW6"
set_property PACKAGE_PIN M15 [get_ports {BOARD_SLIDE_SW[7]}];  # "SW7"
set_property IOSTANDARD  LVCMOS25 [get_ports BOARD_SLIDE_SW*]
set_input_delay 10 -clock [get_clocks CLK100] [get_ports BOARD_SLIDE_SW*]
set_false_path -from [get_ports BOARD_SLIDE_SW*]

set_property PACKAGE_PIN T22 [get_ports {BOARD_LED[0]}];  # "LD0"
set_property PACKAGE_PIN T21 [get_ports {BOARD_LED[1]}];  # "LD1"
set_property PACKAGE_PIN U22 [get_ports {BOARD_LED[2]}];  # "LD2"
set_property PACKAGE_PIN U21 [get_ports {BOARD_LED[3]}];  # "LD3"
set_property PACKAGE_PIN V22 [get_ports {BOARD_LED[4]}];  # "LD4"
set_property PACKAGE_PIN W22 [get_ports {BOARD_LED[5]}];  # "LD5"
set_property PACKAGE_PIN U19 [get_ports {BOARD_LED[6]}];  # "LD6"
set_property PACKAGE_PIN U14 [get_ports {BOARD_LED[7]}];  # "LD7"
set_property IOSTANDARD  LVCMOS33 [get_ports BOARD_LED*]
set_output_delay 10 -clock [get_clocks CLK100] [get_ports BOARD_LED*]
set_false_path -to [get_ports BOARD_LED*]

set_property PACKAGE_PIN B22 [get_ports GBEU_PHY_RESET_N]   ;# "FMC_LA33_N" , G37
set_property PACKAGE_PIN J16  [get_ports GBE_MDC     ]   ;# "FMC_LA15_P"     , H19
set_property PACKAGE_PIN J17  [get_ports GBE_MDIO    ]   ;# "FMC_LA15_N"     , H20
set_property IOSTANDARD LVCMOS25          [get_ports {GBE_*}  ]

### Port0

set_property PACKAGE_PIN M19  [get_ports GBEU_RXC     ]   ;# "FMC_LA00_CC_P"  , G6
set_property PACKAGE_PIN M20  [get_ports GBEU_RXD[0]  ]   ;# "FMC_LA00_CC_N"  , G7
set_property PACKAGE_PIN P17  [get_ports GBEU_RXD[1]  ]   ;# "FMC_LA02_P"     , H7
set_property PACKAGE_PIN N19  [get_ports GBEU_RXD[2]  ]   ;# "FMC_LA01_CC_P"  , D8
set_property PACKAGE_PIN P18  [get_ports GBEU_RXD[3]  ]   ;# "FMC_LA02_N"     , H8
set_property PACKAGE_PIN N20  [get_ports GBEU_RXD[4]  ]   ;# "FMC_LA01_CC_N"  , D9
set_property PACKAGE_PIN L21  [get_ports GBEU_RXD[6]  ]   ;# "FMC_LA06_P"     , C10
set_property PACKAGE_PIN N22  [get_ports GBEU_RXD[5]  ]   ;# "FMC_LA03_P"     , G9
set_property PACKAGE_PIN P22  [get_ports GBEU_RXD[7]  ]   ;# "FMC_LA03_N"     , G10
set_property PACKAGE_PIN L19  [get_ports GBEU_RXDV    ]   ;# "FMC_CLK0_N"     , H5
set_property PACKAGE_PIN M21  [get_ports GBEU_RXER    ]   ;# "FMC_LA04_P"     , H10

set_property PACKAGE_PIN T19  [get_ports GBEU_GTXC    ]   ;# "FMC_LA10_N"     , C15
set_property PACKAGE_PIN P20  [get_ports GBEU_TXD[0]  ]   ;# "FMC_LA12_P"     , G15
set_property PACKAGE_PIN P21  [get_ports GBEU_TXD[1]  ]   ;# "FMC_LA12_N"     , G16
set_property PACKAGE_PIN N17  [get_ports GBEU_TXD[2]  ]   ;# "FMC_LA11_P"     , H16
set_property PACKAGE_PIN L17  [get_ports GBEU_TXD[3]  ]   ;# "FMC_LA13_P"     , D17
set_property PACKAGE_PIN N18  [get_ports GBEU_TXD[4]  ]   ;# "FMC_LA11_N"     , H17
set_property PACKAGE_PIN K19  [get_ports GBEU_TXD[5]  ]   ;# "FMC_LA14_P"     , C18
set_property PACKAGE_PIN M17  [get_ports GBEU_TXD[6]  ]   ;# "FMC_LA13_N"     , D18
set_property PACKAGE_PIN J20  [get_ports GBEU_TXD[7]  ]   ;# "FMC_LA16_P"     , G18
```

```
set_property PACKAGE_PIN R21  [get_ports GBEU_TXEN    ]   ;# "FMC_LA09_N"     , D15
set_property PACKAGE_PIN K20  [get_ports GBEU_TXER    ]   ;# "FMC_LA14_N"     , C19

set_property IOSTANDARD LVCMOS25          [get_ports {GBEU_*}  ]
set_property DRIVE 12                     [get_ports {GBEU_TX*}]
set_property SLEW  FAST                   [get_ports {GBEU_TX*}]
create_clock -name GBEU_RXC  -period  8.0  [get_ports {GBEU_RXC}]
create_clock -name GBEU_GTXC -period  8.0  [get_ports {GBEU_GTXC}]
set_output_delay 1 -clock [get_clocks {ZYNQ_CLKMGRA.SYS_CLK_CLKOUT2}] [get_ports
GBEU_GTXC]

set_output_delay 10 -clock [get_clocks GBEU_GTXC] [get_ports GBEU_PHY_RESET_N]
set_false_path -to   [get_ports GBEU_PHY_RESET_N]

set_input_delay  -clock [get_clocks GBEU_RXC]  -min 2 [get_ports {GBEU_RXD*}]; #hold
set_input_delay  -clock [get_clocks GBEU_RXC]  -max 4 [get_ports {GBEU_RXD*}]; #setup
set_input_delay  -clock [get_clocks GBEU_RXC]  -max 4 [get_ports {GBEU_RXDV}]; #setup
set_input_delay  -clock [get_clocks GBEU_RXC]  -min 2 [get_ports {GBEU_RXER}]; #hold
set_input_delay  -clock [get_clocks GBEU_RXC]  -max 4 [get_ports {GBEU_RXER}]; #setup

set_output_delay -clock [get_clocks GBEU_GTXC] -min 2 [get_ports {GBEU_TXD*}]; #hold
set_output_delay -clock [get_clocks GBEU_GTXC] -max 4 [get_ports {GBEU_TXD*}]; #setup
set_output_delay -clock [get_clocks GBEU_GTXC] -min 2 [get_ports {GBEU_TXEN}]; #hold
set_output_delay -clock [get_clocks GBEU_GTXC] -max 4 [get_ports {GBEU_TXEN}]; #setup
set_output_delay -clock [get_clocks GBEU_GTXC] -min 2 [get_ports {GBEU_TXER}]; #hold
set_output_delay -clock [get_clocks GBEU_GTXC] -max 4 [get_ports {GBEU_TXER}]; #setup

set_property IOB TRUE  [get_cells {u_hsr/u_host/BLK_REDBOX.u_gmii/gmii_rxd_int_reg*}]
set_property IOB TRUE  [get_cells {u_hsr/u_host/BLK_REDBOX.u_gmii/gmii_rxdv_int_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_host/BLK_REDBOX.u_gmii/gmii_rxer_int_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_host/BLK_REDBOX.u_gmii/gmii_txd_reg*}]
set_property IOB TRUE  [get_cells {u_hsr/u_host/BLK_REDBOX.u_gmii/gmii_txen_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_host/BLK_REDBOX.u_gmii/gmii_txer_reg}]

### Port1
set_property PACKAGE_PIN A21 [get_ports GBEA_PHY_RESET_N]   ;# "FMC_LA32_P"   , H37

set_property PACKAGE_PIN L18  [get_ports GBEA_RXC     ]   ;# "FMC_CLK0_P"     , H4
set_property PACKAGE_PIN J18  [get_ports GBEA_RXD[0]  ]   ;# "FMC_LA05_P"     , D11
set_property PACKAGE_PIN M22  [get_ports GBEA_RXD[1]  ]   ;# "FMC_LA04_N"     , H11
set_property PACKAGE_PIN K18  [get_ports GBEA_RXD[2]  ]   ;# "FMC_LA05_N"     , D12
set_property PACKAGE_PIN J21  [get_ports GBEA_RXD[3]  ]   ;# "FMC_LA08_P"     , G12
set_property PACKAGE_PIN J22  [get_ports GBEA_RXD[4]  ]   ;# "FMC_LA08_N"     , G13
set_property PACKAGE_PIN T16  [get_ports GBEA_RXD[5]  ]   ;# "FMC_LA07_P"     , H13
set_property PACKAGE_PIN R19  [get_ports GBEA_RXD[6]  ]   ;# "FMC_LA10_P"     , C14
set_property PACKAGE_PIN R20  [get_ports GBEA_RXD[7]  ]   ;# "FMC_LA09_P"     , D14
set_property PACKAGE_PIN L22  [get_ports GBEA_RXDV    ]   ;# "FMC_LA06_N"     , C11
set_property PACKAGE_PIN T17  [get_ports GBEA_RXER    ]   ;# "FMC_LA07_N"     , H14

set_property PACKAGE_PIN B19  [get_ports GBEA_GTXC    ]   ;# "FMC_LA17_CC_P"  , D20
set_property PACKAGE_PIN G20  [get_ports GBEA_TXD[0]  ]   ;# "FMC_LA20_P"     , G21
set_property PACKAGE_PIN D20  [get_ports GBEA_TXD[1]  ]   ;# "FMC_LA18_CC_P"  , C22
set_property PACKAGE_PIN G21  [get_ports GBEA_TXD[2]  ]   ;# "FMC_LA20_N"     , G22
set_property PACKAGE_PIN G15  [get_ports GBEA_TXD[3]  ]   ;# "FMC_LA19_P"     , H22
set_property PACKAGE_PIN C20  [get_ports GBEA_TXD[4]  ]   ;# "FMC_LA18_CC_N"  , C23
set_property PACKAGE_PIN E15  [get_ports GBEA_TXD[5]  ]   ;# "FMC_LA23_P"     , D23
set_property PACKAGE_PIN G16  [get_ports GBEA_TXD[6]  ]   ;# "FMC_LA19_N"     , H23
```

```
set_property PACKAGE_PIN D15  [get_ports GBEA_TXD[7] ]   ;# "FMC_LA23_N"     , D24
set_property PACKAGE_PIN B20  [get_ports GBEA_TXEN   ]   ;# "FMC_LA17_CC_N"  , D21
set_property PACKAGE_PIN G19  [get_ports GBEA_TXER   ]   ;# "FMC_LA22_P"     , G24

set_property IOSTANDARD LVCMOS25 [get_ports {GBEA_*}]
set_property DRIVE 12     [get_ports {GBEA_TX*}]
set_property SLEW  FAST   [get_ports {GBEA_TX*}]
create_clock -name GBEA_RXC  -period  8.0  [get_ports {GBEA_RXC}]
create_clock -name GBEA_GTXC -period  8.0  [get_ports {GBEA_GTXC}]
set_output_delay 1 -clock [get_clocks {ZYNQ_CLKMGRA.SYS_CLK_CLKOUT2}] [get_ports
GBEA_GTXC]

set_output_delay 10 -clock [get_clocks GBEA_GTXC] [get_ports GBEA_PHY_RESET_N]
set_false_path -to   [get_ports GBEA_PHY_RESET_N]

set_input_delay  -clock [get_clocks GBEA_RXC]  -min 2 [get_ports {GBEA_RXD*}]; #hold
set_input_delay  -clock [get_clocks GBEA_RXC]  -max 4 [get_ports {GBEA_RXD*}]; #setup
set_input_delay  -clock [get_clocks GBEA_RXC]  -max 4 [get_ports {GBEA_RXDV}]; #setup
set_input_delay  -clock [get_clocks GBEA_RXC]  -min 2 [get_ports {GBEA_RXER}]; #hold
set_input_delay  -clock [get_clocks GBEA_RXC]  -max 4 [get_ports {GBEA_RXER}]; #setup

set_output_delay -clock [get_clocks GBEA_GTXC] -min 2 [get_ports {GBEA_TXD*}]; #hold
set_output_delay -clock [get_clocks GBEA_GTXC] -max 4 [get_ports {GBEA_TXD*}]; #setup
set_output_delay -clock [get_clocks GBEA_GTXC] -min 2 [get_ports {GBEA_TXEN}]; #hold
set_output_delay -clock [get_clocks GBEA_GTXC] -max 4 [get_ports {GBEA_TXEN}]; #setup
set_output_delay -clock [get_clocks GBEA_GTXC] -min 2 [get_ports {GBEA_TXER}]; #hold
set_output_delay -clock [get_clocks GBEA_GTXC] -max 4 [get_ports {GBEA_TXER}]; #setup

set_property IOB TRUE  [get_cells {u_hsr/u_net_A/u_gmii/gmii_rxd_int_reg*}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_A/u_gmii/gmii_rxdv_int_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_A/u_gmii/gmii_rxer_int_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_A/u_gmii/gmii_txd_reg*}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_A/u_gmii/gmii_txen_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_A/u_gmii/gmii_txer_reg}]

### Port 2
set_property PACKAGE_PIN A22        [get_ports GBEB_PHY_RESET_N]   ;# "FMC_LA32_N"     ,
H38

set_property PACKAGE_PIN D18  [get_ports GBEB_RXC     ]   ;# "FMC_CLK1_P"     , G2
set_property PACKAGE_PIN F19  [get_ports GBEB_RXD[0]  ]   ;# "FMC_LA22_N"     , G25
set_property PACKAGE_PIN E19  [get_ports GBEB_RXD[1]  ]   ;# "FMC_LA21_P"     , H25
set_property PACKAGE_PIN E21  [get_ports GBEB_RXD[2]  ]   ;# "FMC_LA27_P"     , C26
set_property PACKAGE_PIN F18  [get_ports GBEB_RXD[3]  ]   ;# "FMC_LA26_P"     , D26
set_property PACKAGE_PIN E20  [get_ports GBEB_RXD[4]  ]   ;# "FMC_LA21_N"     , H26
set_property PACKAGE_PIN D21  [get_ports GBEB_RXD[5]  ]   ;# "FMC_LA27_N"     , C27
set_property PACKAGE_PIN E18  [get_ports GBEB_RXD[6]  ]   ;# "FMC_LA26_N"     , D27
set_property PACKAGE_PIN D22  [get_ports GBEB_RXD[7]  ]   ;# "FMC_LA25_P"     , G27
set_property PACKAGE_PIN C19  [get_ports GBEB_RXDV    ]   ;# "FMC_CLK1_N"     , G3
set_property PACKAGE_PIN C22  [get_ports GBEB_RXER    ]   ;# "FMC_LA25_N"     , G28

set_property PACKAGE_PIN A18  [get_ports GBEB_GTXC    ]   ;# "FMC_LA24_P"     , H28
set_property PACKAGE_PIN C17  [get_ports GBEB_TXD[0]  ]   ;# "FMC_LA29_P"     , G30
set_property PACKAGE_PIN C18  [get_ports GBEB_TXD[1]  ]   ;# "FMC_LA29_N"     , G31
set_property PACKAGE_PIN A16  [get_ports GBEB_TXD[2]  ]   ;# "FMC_LA28_P"     , H31
set_property PACKAGE_PIN A17  [get_ports GBEB_TXD[3]  ]   ;# "FMC_LA28_N"     , H32
set_property PACKAGE_PIN B16  [get_ports GBEB_TXD[4]  ]   ;# "FMC_LA31_P"     , G33
```

```
set_property PACKAGE_PIN B17  [get_ports GBEB_TXD[5]  ]   ;# "FMC_LA31_N"     , G34
set_property PACKAGE_PIN C15  [get_ports GBEB_TXD[6]  ]   ;# "FMC_LA30_P"     , H34
set_property PACKAGE_PIN B15  [get_ports GBEB_TXD[7]  ]   ;# "FMC_LA30_N"     , H35
set_property PACKAGE_PIN A19  [get_ports GBEB_TXEN    ]   ;# "FMC_LA24_N"     , H29
set_property PACKAGE_PIN B21  [get_ports GBEB_TXER    ]   ;# "FMC_LA33_P"     , G36

set_property IOSTANDARD LVCMOS25 [get_ports {GBEB_*}]
set_property DRIVE 12     [get_ports {GBEB_TX*}]
set_property SLEW  FAST   [get_ports {GBEB_TX*}]
create_clock -name GBEB_RXC  -period  8.0  [get_ports {GBEB_RXC}]
create_clock -name GBEB_GTXC -period  8.0  [get_ports {GBEB_GTXC}]
set_output_delay 1 -clock [get_clocks {ZYNQ_CLKMGRA.SYS_CLK_CLKOUT2}] [get_ports
GBEB_GTXC]

set_output_delay 10 -clock [get_clocks GBEB_GTXC] [get_ports GBEB_PHY_RESET_N]
set_false_path -reset_path    -to   [get_ports GBEB_PHY_RESET_N]

set_input_delay  -clock [get_clocks GBEB_RXC]  -min 2 [get_ports {GBEB_RXD*}]; #hold
set_input_delay  -clock [get_clocks GBEB_RXC]  -max 4 [get_ports {GBEB_RXD*}]; #setup
set_input_delay  -clock [get_clocks GBEB_RXC]  -max 4 [get_ports {GBEB_RXDV}]; #setup
set_input_delay  -clock [get_clocks GBEB_RXC]  -min 2 [get_ports {GBEB_RXER}]; #hold
set_input_delay  -clock [get_clocks GBEB_RXC]  -max 4 [get_ports {GBEB_RXER}]; #setup

set_output_delay -clock [get_clocks GBEB_GTXC] -min 2 [get_ports {GBEB_TXD*}]; #hold
set_output_delay -clock [get_clocks GBEB_GTXC] -max 4 [get_ports {GBEB_TXD*}]; #setup
set_output_delay -clock [get_clocks GBEB_GTXC] -min 2 [get_ports {GBEB_TXEN}]; #hold
set_output_delay -clock [get_clocks GBEB_GTXC] -max 4 [get_ports {GBEB_TXEN}]; #setup
set_output_delay -clock [get_clocks GBEB_GTXC] -min 2 [get_ports {GBEB_TXER}]; #hold
set_output_delay -clock [get_clocks GBEB_GTXC] -max 4 [get_ports {GBEB_TXER}]; #setup

set_property IOB TRUE  [get_cells {u_hsr/u_net_B/u_gmii/gmii_rxd_int_reg*}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_B/u_gmii/gmii_rxdv_int_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_B/u_gmii/gmii_rxer_int_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_B/u_gmii/gmii_txd_reg*}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_B/u_gmii/gmii_txen_reg}]
set_property IOB TRUE  [get_cells {u_hsr/u_net_B/u_gmii/gmii_txer_reg}]
```

6.2 DANH

```
set_property PACKAGE_PIN Y9 [get_ports BOARD_CLK_IN] ;# 100Mhz
set_property IOSTANDARD LVCMOS33 [get_ports BOARD_CLK_IN]
create_clock   -name BOARD_CLK_IN -period  10.0 [get_ports BOARD_CLK_IN]

set_property PACKAGE_PIN P16      [get_ports BOARD_RST_SW] ;#BTNC
set_property IOSTANDARD  LVCMOS25 [get_ports BOARD_RST_SW]
set_input_delay 10 -clock [get_clocks BOARD_CLK_IN] [get_ports BOARD_RST_SW]
set_false_path -from [get_ports BOARD_RST_SW]

set_property PACKAGE_PIN F22 [get_ports {BOARD_SLIDE_SW[0]}];  # "SW0"
set_property PACKAGE_PIN G22 [get_ports {BOARD_SLIDE_SW[1]}];  # "SW1"
set_property PACKAGE_PIN H22 [get_ports {BOARD_SLIDE_SW[2]}];  # "SW2"
set_property PACKAGE_PIN F21 [get_ports {BOARD_SLIDE_SW[3]}];  # "SW3"
set_property PACKAGE_PIN H19 [get_ports {BOARD_SLIDE_SW[4]}];  # "SW4"
set_property PACKAGE_PIN H18 [get_ports {BOARD_SLIDE_SW[5]}];  # "SW5"
```

```
set_property PACKAGE_PIN H17 [get_ports {BOARD_SLIDE_SW[6]}];  # "SW6"
set_property PACKAGE_PIN M15 [get_ports {BOARD_SLIDE_SW[7]}];  # "SW7"
set_property IOSTANDARD  LVCMOS25 [get_ports BOARD_SLIDE_SW*]
set_input_delay 10 -clock [get_clocks BOARD_CLK_IN] [get_ports BOARD_SLIDE_SW*]
set_false_path -from [get_ports BOARD_SLIDE_SW*]

set_property PACKAGE_PIN T22 [get_ports {BOARD_LED[0]}];  # "LD0"
set_property PACKAGE_PIN T21 [get_ports {BOARD_LED[1]}];  # "LD1"
set_property PACKAGE_PIN U22 [get_ports {BOARD_LED[2]}];  # "LD2"
set_property PACKAGE_PIN U21 [get_ports {BOARD_LED[3]}];  # "LD3"
set_property PACKAGE_PIN V22 [get_ports {BOARD_LED[4]}];  # "LD4"
set_property PACKAGE_PIN W22 [get_ports {BOARD_LED[5]}];  # "LD5"
set_property PACKAGE_PIN U19 [get_ports {BOARD_LED[6]}];  # "LD6"
set_property PACKAGE_PIN U14 [get_ports {BOARD_LED[7]}];  # "LD7"
set_property IOSTANDARD  LVCMOS33 [get_ports BOARD_LED*]
set_output_delay 10 -clock [get_clocks BOARD_CLK_IN] [get_ports BOARD_LED*]
set_false_path -through [get_ports BOARD_LED*]

set_property PACKAGE_PIN J16  [get_ports GBE_MDC     ]   ;# "FMC_LA15_P"     , H19
set_property PACKAGE_PIN J17  [get_ports GBE_MDIO    ]   ;# "FMC_LA15_N"     , H20
set_property IOSTANDARD LVCMOS25          [get_ports {GBE_*}  ]

### Port1
set_property PACKAGE_PIN A21        [get_ports GBEA_PHY_RESET_N]   ;#
"FMC_LA32_P"     , H37

set_property PACKAGE_PIN L18  [get_ports GBEA_RXC     ]   ;# "FMC_CLK0_P"     , H4
set_property PACKAGE_PIN J18  [get_ports GBEA_RXD[0] ]   ;# "FMC_LA05_P"     , D11
set_property PACKAGE_PIN M22  [get_ports GBEA_RXD[1]  ]   ;# "FMC_LA04_N"     , H11
set_property PACKAGE_PIN K18  [get_ports GBEA_RXD[2]  ]   ;# "FMC_LA05_N"     , D12
set_property PACKAGE_PIN J21  [get_ports GBEA_RXD[3]  ]   ;# "FMC_LA08_P"     , G12
set_property PACKAGE_PIN J22  [get_ports GBEA_RXD[4]  ]   ;# "FMC_LA08_N"     , G13
set_property PACKAGE_PIN T16  [get_ports GBEA_RXD[5]  ]   ;# "FMC_LA07_P"     , H13
set_property PACKAGE_PIN R19  [get_ports GBEA_RXD[6]  ]   ;# "FMC_LA10_P"     , C14
set_property PACKAGE_PIN R20  [get_ports GBEA_RXD[7]  ]   ;# "FMC_LA09_P"     , D14
set_property PACKAGE_PIN L22  [get_ports GBEA_RXDV    ]   ;# "FMC_LA06_N"     , C11
set_property PACKAGE_PIN T17  [get_ports GBEA_RXER    ]   ;# "FMC_LA07_N"     , H14

set_property PACKAGE_PIN B19  [get_ports GBEA_GTXC    ]   ;# "FMC_LA17_CC_P"  , D20
set_property PACKAGE_PIN G20  [get_ports GBEA_TXD[0]  ]   ;# "FMC_LA20_P"     , G21
set_property PACKAGE_PIN D20  [get_ports GBEA_TXD[1]  ]   ;# "FMC_LA18_CC_P"  , C22
set_property PACKAGE_PIN G21  [get_ports GBEA_TXD[2]  ]   ;# "FMC_LA20_N"     , G22
set_property PACKAGE_PIN G15  [get_ports GBEA_TXD[3]  ]   ;# "FMC_LA19_P"     , H22
set_property PACKAGE_PIN C20  [get_ports GBEA_TXD[4]  ]   ;# "FMC_LA18_CC_N"  , C23
set_property PACKAGE_PIN E15  [get_ports GBEA_TXD[5]  ]   ;# "FMC_LA23_P"     , D23
set_property PACKAGE_PIN G16  [get_ports GBEA_TXD[6]  ]   ;# "FMC_LA19_N"     , H23
set_property PACKAGE_PIN D15  [get_ports GBEA_TXD[7]  ]   ;# "FMC_LA23_N"     , D24
set_property PACKAGE_PIN B20  [get_ports GBEA_TXEN    ]   ;# "FMC_LA17_CC_N"  , D21
set_property PACKAGE_PIN G19  [get_ports GBEA_TXER    ]   ;# "FMC_LA22_P"     , G24

set_property IOSTANDARD LVCMOS25 [get_ports {GBEA_*}]
set_property DRIVE 12     [get_ports {GBEA_TX*}]
set_property SLEW  FAST   [get_ports {GBEA_TX*}]
create_clock -name GBEA_RXC  -period  8.0  [get_ports {GBEA_RXC}]
create_clock -name GBEA_GTXC -period  8.0  [get_ports {GBEA_GTXC}]
#set_output_delay 1 -clock [get_clocks {ZYNQ_CLKMGRA.SYS_CLK_CLKOUT2}] [get_ports
GBEA_GTXC]
```

```
set_output_delay 10 -clock [get_clocks GBEA_GTXC] [get_ports GBEA_PHY_RESET_N]
set_false_path -to   [get_ports GBEA_PHY_RESET_N]

set_input_delay  -clock [get_clocks GBEA_RXC]  -min 2 [get_ports {GBEA_RXD*}]; #hold
set_input_delay  -clock [get_clocks GBEA_RXC]  -max 4 [get_ports {GBEA_RXD*}]; #setup
set_input_delay  -clock [get_clocks GBEA_RXC]  -max 4 [get_ports {GBEA_RXDV}]; #setup
set_input_delay  -clock [get_clocks GBEA_RXC]  -min 2 [get_ports {GBEA_RXER}]; #hold
set_input_delay  -clock [get_clocks GBEA_RXC]  -max 4 [get_ports {GBEA_RXER}]; #setup

set_output_delay -clock [get_clocks GBEA_GTXC] -min 2 [get_ports {GBEA_TXD*}]; #hold
set_output_delay -clock [get_clocks GBEA_GTXC] -max 4 [get_ports {GBEA_TXD*}]; #setup
set_output_delay -clock [get_clocks GBEA_GTXC] -min 2 [get_ports {GBEA_TXEN}]; #hold
set_output_delay -clock [get_clocks GBEA_GTXC] -max 4 [get_ports {GBEA_TXEN}]; #setup
set_output_delay -clock [get_clocks GBEA_GTXC] -min 2 [get_ports {GBEA_TXER}]; #hold
set_output_delay -clock [get_clocks GBEA_GTXC] -max 4 [get_ports {GBEA_TXER}]; #setup

set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_A/u_gmii/gmii_rxd_int_reg*}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_A/u_gmii/gmii_rxdv_int_reg}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_A/u_gmii/gmii_rxer_int_reg}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_A/u_gmii/gmii_txd_reg*}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_A/u_gmii/gmii_txen_reg}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_A/u_gmii/gmii_txer_reg}]

### Port 2
set_property PACKAGE_PIN A22        [get_ports GBEB_PHY_RESET_N]   ;#
"FMC_LA32_N"     , H38

set_property PACKAGE_PIN D18  [get_ports GBEB_RXC     ]   ;# "FMC_CLK1_P"     , G2
set_property PACKAGE_PIN F19  [get_ports GBEB_RXD[0]  ]   ;# "FMC_LA22_N"     , G25
set_property PACKAGE_PIN E19  [get_ports GBEB_RXD[1]  ]   ;# "FMC_LA21_P"     , H25
set_property PACKAGE_PIN E21  [get_ports GBEB_RXD[2]  ]   ;# "FMC_LA27_P"     , C26
set_property PACKAGE_PIN F18  [get_ports GBEB_RXD[3]  ]   ;# "FMC_LA26_P"     , D26
set_property PACKAGE_PIN E20  [get_ports GBEB_RXD[4]  ]   ;# "FMC_LA21_N"     , H26
set_property PACKAGE_PIN D21  [get_ports GBEB_RXD[5]  ]   ;# "FMC_LA27_N"     , C27
set_property PACKAGE_PIN E18  [get_ports GBEB_RXD[6]  ]   ;# "FMC_LA26_N"     , D27
set_property PACKAGE_PIN D22  [get_ports GBEB_RXD[7]  ]   ;# "FMC_LA25_P"     , G27
set_property PACKAGE_PIN C19  [get_ports GBEB_RXDV    ]   ;# "FMC_CLK1_N"     , G3
set_property PACKAGE_PIN C22  [get_ports GBEB_RXER    ]   ;# "FMC_LA25_N"     , G28

set_property PACKAGE_PIN A18  [get_ports GBEB_GTXC    ]   ;# "FMC_LA24_P"     , H28
set_property PACKAGE_PIN C17  [get_ports GBEB_TXD[0]  ]   ;# "FMC_LA29_P"     , G30
set_property PACKAGE_PIN C18  [get_ports GBEB_TXD[1]  ]   ;# "FMC_LA29_N"     , G31
set_property PACKAGE_PIN A16  [get_ports GBEB_TXD[2]  ]   ;# "FMC_LA28_P"     , H31
set_property PACKAGE_PIN A17  [get_ports GBEB_TXD[3]  ]   ;# "FMC_LA28_N"     , H32
set_property PACKAGE_PIN B16  [get_ports GBEB_TXD[4]  ]   ;# "FMC_LA31_P"     , G33
set_property PACKAGE_PIN B17  [get_ports GBEB_TXD[5]  ]   ;# "FMC_LA31_N"     , G34
set_property PACKAGE_PIN C15  [get_ports GBEB_TXD[6]  ]   ;# "FMC_LA30_P"     , H34
set_property PACKAGE_PIN B15  [get_ports GBEB_TXD[7]  ]   ;# "FMC_LA30_N"     , H35
set_property PACKAGE_PIN A19  [get_ports GBEB_TXEN    ]   ;# "FMC_LA24_N"     , H29
set_property PACKAGE_PIN B21  [get_ports GBEB_TXER    ]   ;# "FMC_LA33_P"     , G36
```

```
set_property IOSTANDARD LVCMOS25 [get_ports {GBEB_*}]
set_property DRIVE 12     [get_ports {GBEB_TX*}]
set_property SLEW  FAST   [get_ports {GBEB_TX*}]
create_clock -name GBEB_RXC  -period  8.0  [get_ports {GBEB_RXC}]
create_clock -name GBEB_GTXC -period  8.0  [get_ports {GBEB_GTXC}]
#set_output_delay 1 -clock [get_clocks {ZYNQ_CLKMGRA.SYS_CLK_CLKOUT2}] [get_ports
GBEB_GTXC]

set_output_delay 10 -clock [get_clocks GBEB_GTXC] [get_ports GBEB_PHY_RESET_N]
set_false_path -reset_path    -to   [get_ports GBEB_PHY_RESET_N]

set_input_delay  -clock [get_clocks GBEB_RXC]  -min 2 [get_ports {GBEB_RXD*}]; #hold
set_input_delay  -clock [get_clocks GBEB_RXC]  -max 4 [get_ports {GBEB_RXD*}]; #setup
set_input_delay  -clock [get_clocks GBEB_RXC]  -max 4 [get_ports {GBEB_RXDV}]; #setup
set_input_delay  -clock [get_clocks GBEB_RXC]  -min 2 [get_ports {GBEB_RXER}]; #hold
set_input_delay  -clock [get_clocks GBEB_RXC]  -max 4 [get_ports {GBEB_RXER}]; #setup

set_output_delay -clock [get_clocks GBEB_GTXC] -min 2 [get_ports {GBEB_TXD*}]; #hold
set_output_delay -clock [get_clocks GBEB_GTXC] -max 4 [get_ports {GBEB_TXD*}]; #setup
set_output_delay -clock [get_clocks GBEB_GTXC] -min 2 [get_ports {GBEB_TXEN}]; #hold
set_output_delay -clock [get_clocks GBEB_GTXC] -max 4 [get_ports {GBEB_TXEN}]; #setup
set_output_delay -clock [get_clocks GBEB_GTXC] -min 2 [get_ports {GBEB_TXER}]; #hold
set_output_delay -clock [get_clocks GBEB_GTXC] -max 4 [get_ports {GBEB_TXER}]; #setup

set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_B/u_gmii/gmii_rxd_int_reg*}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_B/u_gmii/gmii_rxdv_int_reg}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_B/u_gmii/gmii_rxer_int_reg}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_B/u_gmii/gmii_txd_reg*}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_B/u_gmii/gmii_txen_reg}]
set_property IOB TRUE  [get_cells
{zed_bd_i/hsr_danh_axi_0/inst/u_fpga/u_dut/u_hsr/u_net_B/u_gmii/gmii_txer_reg}]
```

Revision history

☐ 2018.10.01: Document started by Ando Ki (adki@future-ds.com)

– End of document –