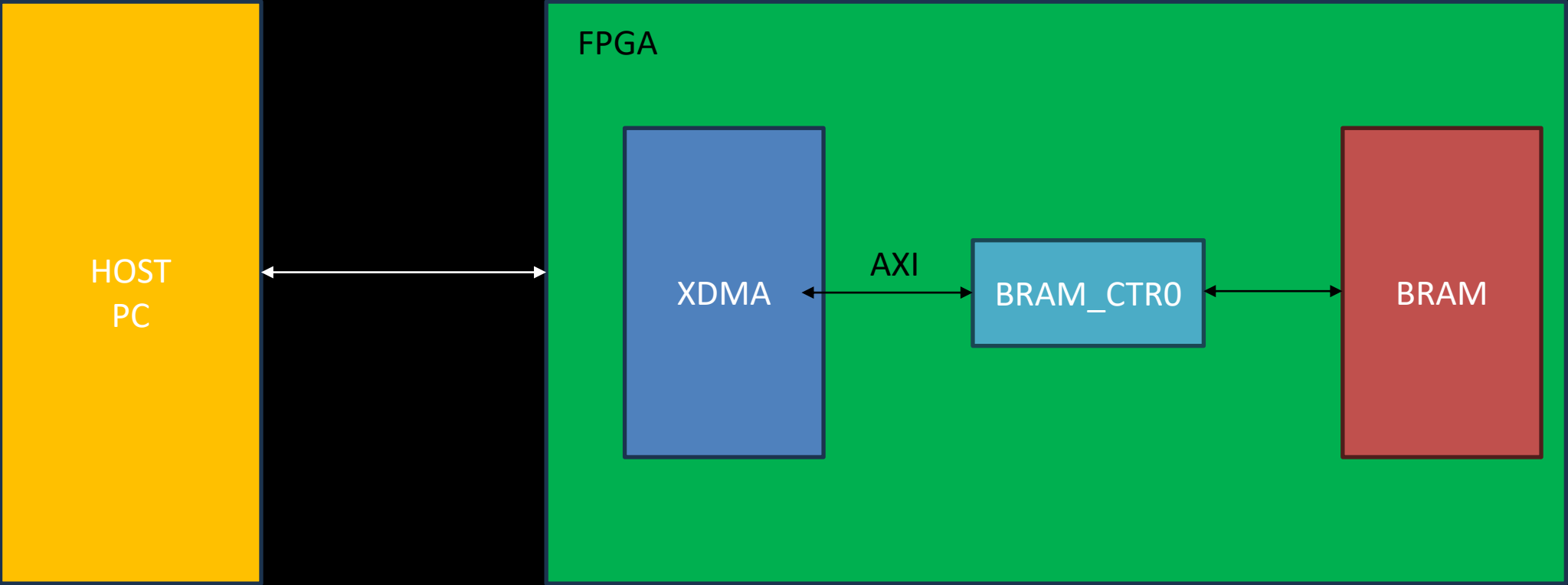




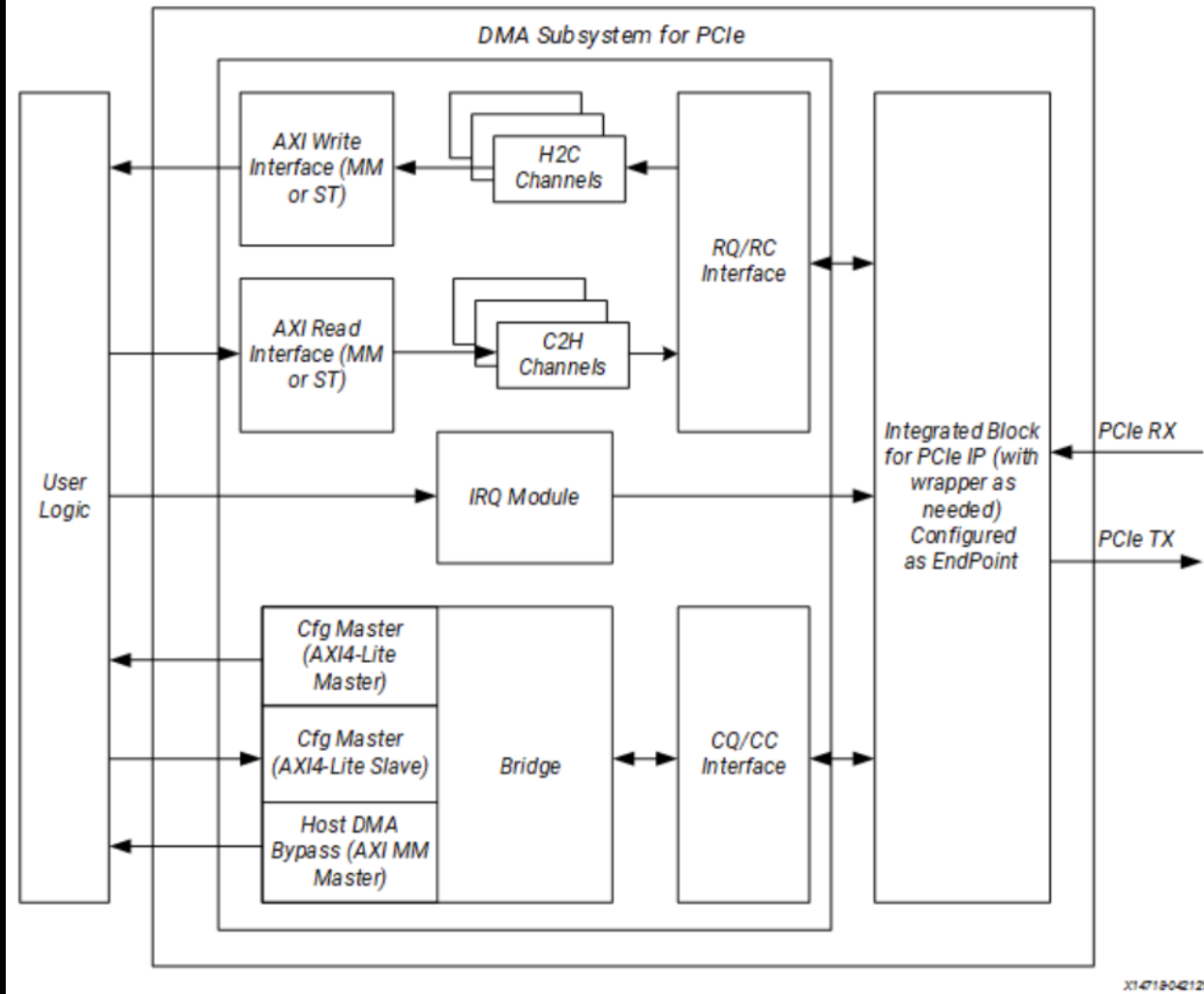
Peripheral Component Interconnect Express(Pcie) with Xilinx DMA

Block Diagram



XDMA IP CORE

Figure: DMA/Bridge Subsystem for PCI Express® Overview

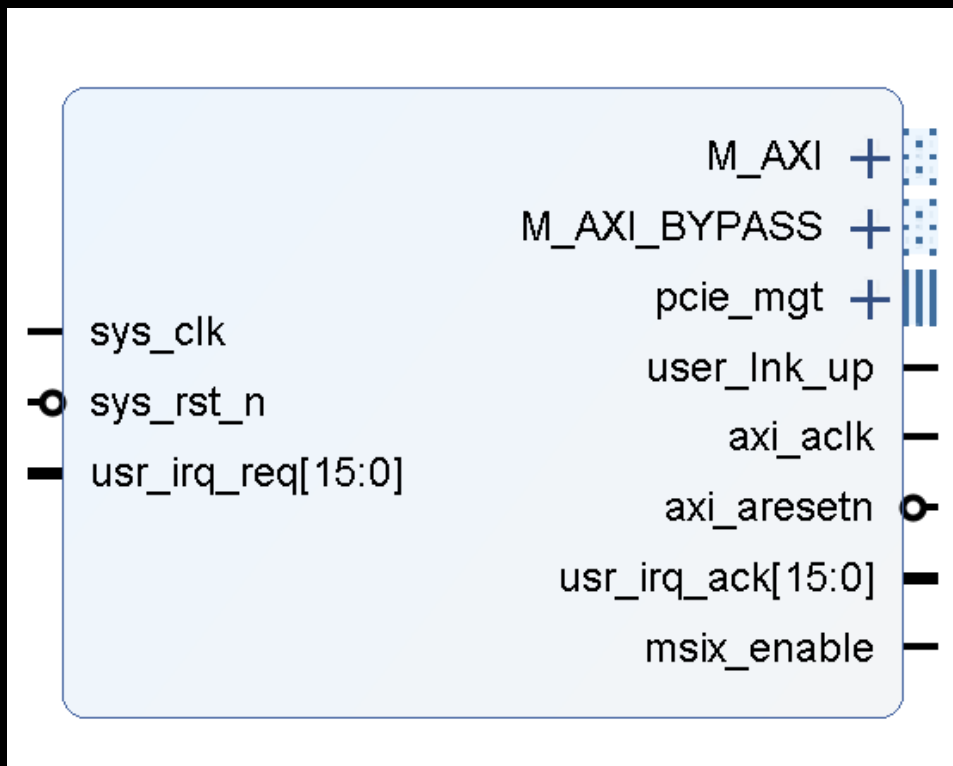


XDMA IP Setting

IP Port

驅動PCie通信的clk&Reset

中斷請求



與FPGA的User Logic走axi通信

PCIE總線

通信完成信號線

驅動AXI傳輸的CLK&Reset

中斷響應

XDMA IP Setting

Re-customize IP

DMA/Bridge Subsystem for PCI Express (4.1)

Documentation IP Location

☐ Show disabled ports

Component Name: xdma_0

Basic **PCIe ID** **PCIe : BARs** **PCIe : MISC** **PCIe : DMA**

Functional Mode: DMA

Mode: Basic

Device / Port Type: PCI Express Endpoint device

PCIe Block Location: X0Y1

PCIe Interface

Lane Width: X1

Maximum Link Speed: ☐ 2.5 GT/s ☒ 5.0 GT/s ☐ 8.0 GT/s

Reference Clock Frequency (MHz): 100 MHz

AXI Interface

AXI Address Width: 64 [32 - 64]

AXI Data Width: 64 bit

AXI Clock Frequency: ☒ 62.5 ☐ 125 ☐ 250

DMA Interface option

☒ AXI Memory Mapped ☐ AXI Stream

☐ AXI-Lite Slave Interface

☐ Enable PIPE Simulation

☐ Enable GT Channel DRP Ports

☐ Enable PCIe DRP Ports

☐ Additional Transceiver Control and Status Ports

sys_clk

sys_rst_n

usr_irq_req[15:0]

M_AXI

M_AXI_BYPASS

pcie_mgt

user_lnk_up

axi_aclk

axi_aresetn

usr_irq_ack[15:0]

msix_enable

PCie bus

傳輸速度

系統時鐘

AXI bus

AXI clk

OK Cancel

XDMA IP Setting

Pcie ID辨識 依據Device修改

Basic	PCle ID	PCle : BARs	PCle : MISC	PCle : DMA
ID Initial Values				
Vendor ID	10EE			
Device ID	7021			
Revision ID	00			
Subsystem Vendor ID	10EE			
Subsystem ID	0007			
<input type="checkbox"/> Enable PCIe-ID Interface				
Class Code Lookup Assistant				
<input type="checkbox"/> Use Class Code Lookup Assistant				
Base Class Menu	Simple communication controllers			
Base Class Value	05			Range: 00..FF
Sub Class Interface Menu	Generic XT compatible serial controller			
Sub Class Value	80			Range: 00..FF
Interface Value	00			Range: 00..FF
Class Code	058000			Range: 000000..FFFFFF

XDMA IP Setting

PCIE Address分配

Basic	PCie ID	PCie : BARs	PCie : MISC	PCie : DMA
<input type="checkbox"/> PCie to AXI Lite Master Interface				
<input type="checkbox"/> 64bit Enable <input type="checkbox"/> Prefetchable				
Size <input type="text" value="1"/> Scale <input type="text" value="Megabytes"/>				
Value <input type="text" value="FFF00000"/>				
PCie to AXI Translation <input type="text" value="0x00000000"/>				
<input checked="" type="checkbox"/> PCie to DMA Interface				
<input type="checkbox"/> 64bit Enable <input type="checkbox"/> Prefetchable				
<input checked="" type="checkbox"/> PCie to DMA Bypass Interface				
<input type="checkbox"/> 64bit Enable <input type="checkbox"/> Prefetchable				
Size <input type="text" value="1"/> Scale <input type="text" value="Megabytes"/>				
Value <input type="text" value="FFF00000"/>				
PCie to AXI Translation <input type="text" value="0x0000000000000000"/>				

XDMA IP Setting

PCIE 中斷設置 最高16bit

Basic	PCle ID	PCle : BARs	PCle : MISC	PCle : DMA
-------	---------	-------------	-------------	------------

User Interrupts

Number of User Interrupts Request (1-16) 16

Legacy Interrupt Settings

Legacy Interrupt Settings NONE

MSI Capabilities

☐ Enable MSI Capability Structure

Multiple Message Capability 1 vector

MSI-X Capabilities

☒ Enable MSI-X Capability Structure

Miscellaneous

☐ Extended Tag Field

☐ Configuration Extended Interface

☐ Add the PCIe XVC-VSEC to the Example Design

☐ Configuration Management Interface

Link Status Register

Selects whether the device reference clock is provided by the connector (Synchronous) or generated via an onboard PLL(Asynchronous)

☒ Enable Slot Clock Configuration

XDMA IP Setting

DMA設置 主要修改通道數量 以GEN3來說最高兩讀兩寫

Basic	PCIe ID	PCIe : BARs	PCIe : MISC	PCIe : DMA	
Number of DMA Read Channel (H2C)				1	▼
Number of DMA Write Channel (C2H)				1	▼
Number of Request IDs for Read channel (2,4,8,16,32,64)				32	⊗ [2 - 64]
Number of Request IDs for Write channel (2,4,8,16,32)				16	⊗ [2 - 32]
Descriptor Bypass for Read (H2C)				0000	▼
Descriptor Bypass for Write (C2H)				0000	▼
AXI ID Width				4	▼
<input type="checkbox"/> DMA Status Ports					

Bram_ctrl

接收xdma的axi data並轉給bram

Re-customize IP

AXI BRAM Controller (4.1)

Documentation

IP Location

Show disabled ports

+ S_AXI

s_axi_aclk

s_axi_aresetn

BRAM_PORTA

Component Name

axi_bram_ctrl_0

AXI Protocol

AXI4

Data Width

64

Memory Depth (Auto)

65536

ID Width (Auto)

4

AUTO

Support AXI Narrow Bursts

No

Read Latency

1

[1 - 128]

Read Command Optimization

No

BRAM Options

BRAM Instance (Auto)

External

Number of BRAM interfaces

1

ECC Options

Enable ECC

No

ECC TYPE

Hamming

Enable Fault Injection

No

ECC Reset Value

0

Bram inst

FPGA的bram例化

Re-customize IP

Block Memory Generator (8.4)

[Documentation](#) [IP Location](#)

IP Symbol

Power Estimation

☐ Show disabled ports

+ BRAM_PORTA rsta_busy

Component Name

axi_bram_ctrl_0_bram

Basic

Port A Options

Other Options

Summary

Mode

BRAM Controller

Generate address interface with 32 bits

Memory Type

Single Port RAM

Common Clock

ECC Options

ECC Type

No ECC

Error Injection Pins

Single Bit Error Injection

Write Enable

Byte Write Enable

Byte Size (bits)

8

Algorithm Options

Defines the algorithm used to concatenate the block RAM primitives.
Refer datasheet for more information.

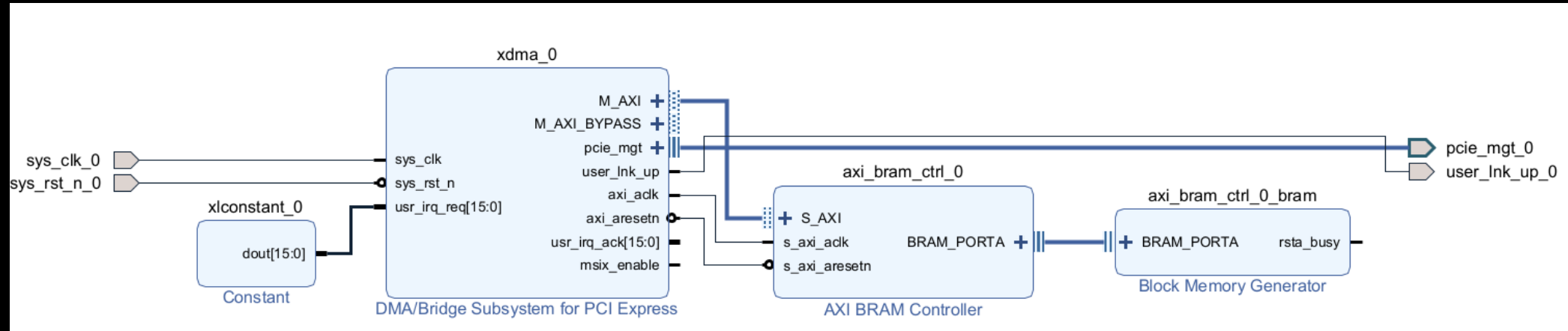
Algorithm

Minimum Area

Primitive

8kx2

Block Design



Address assign

Diagram x fpga_top.sv x xdma_bram_wrapper.v x **Address Editor** x

☒ Assigned (1) ☒ Unassigned (0) ☒ Excluded (0) ☒ Incomplete (1)

Name	Interface	Slave Segment	Master Base Address	Range	Master High Address
Network 0					
/xdma_0					
/xdma_0/M_AXI (64 address bits : 16E)					
/axi_bram_ctrl_0/S_AXI	S_AXI	Mem0	0x0	512K	0x7_FFFF
Network 1					
/xdma_0					
/xdma_0/M_AXI_BYPASS (64 address bits : 16E)					
Incomplete Paths (1)					
/xdma_0/M_AXI_BYPASS					

FPGA_Top

除了bd以外還需添加io & clock buffer，所以另外編寫一個頂層文件

```
module fpga_top (  
    output wire      o_led0,  
    input  wire      i_pcie_rstn,  
    input  wire      i_pcie_refclkp, i_pcie_refclkn,  
    input  wire [0:0] i_pcie_rxp, i_pcie_rxn,  
    output wire [0:0] o_pcie_txp, o_pcie_txn  
);  
wire pcie_rstn;  
wire pcie_refclk;
```

FPGA_Top

使用xilinx提供的原語

```
////////////////////////////////////  
// Ref clock input buffer  
////////////////////////////////////  
IBUFDS_GTE2 refclk_ibuf (  
    .CEB          ( 1'b0          ),  
    .I             ( i_pcie_refclkp ),  
    .IB           ( i_pcie_refclkn ),  
    .O            ( pcie_refclk    ),  
    .ODIV2        (                )  
);  
////////////////////////////////////  
// Reset input buffer  
////////////////////////////////////  
IBUF    sys_reset_n_ibuf (  
    .I          ( i_pcie_rstn      ),  
    .O          ( pcie_rstn        )  
);
```

FPGA_Top

Block design inst

```
////////////////////////////////////  
// block design  
////////////////////////////////////  
xdma_bram_wrapper xdma_bram_wrapper_i (  
    .pcie_mgt_0_rxn    ( i_pcie_rxn    ),  
    .pcie_mgt_0_rxp    ( i_pcie_rxp    ),  
    .pcie_mgt_0_txn    ( o_pcie_txn    ),  
    .pcie_mgt_0_txp    ( o_pcie_txp    ),  
    .sys_clk_0         ( pcie_refclk    ),  
    .sys_rst_n_0       ( pcie_rstn     ),  
    .user_lnk_up_0     ( o_led0        )  
);  
  
endmodule
```


FPGA_Top

Constraints

```
set_property -dict { PACKAGE_PIN AR22  IOSTANDARD LVCMOS15 } [get_ports { o_led0 }];  
#set_property -dict { PACKAGE_PIN AR23  IOSTANDARD LVCMOS15 } [get_ports { o_led1 }];
```

```
set_property -dict { PACKAGE_PIN AY35  IOSTANDARD LVCMOS18  PULLUP true } [get_ports i_pcie_rstn]  
set_property PACKAGE_PIN AB8  [get_ports { i_pcie_refclkp }];  
create_clock -name sys_clk -period 10 [get_ports i_pcie_refclkp]
```

```
set_property PACKAGE_PIN W2  [get_ports { o_pcie_txp[0] }];  
#set_property PACKAGE_PIN AA2  [get_ports { o_pcie_txp[1] }];  
#set_property PACKAGE_PIN AC2  [get_ports { o_pcie_txp[2] }];  
#set_property PACKAGE_PIN AE2  [get_ports { o_pcie_txp[3] }];  
#set_property PACKAGE_PIN AG2  [get_ports { o_pcie_txp[4] }];  
#set_property PACKAGE_PIN AH4  [get_ports { o_pcie_txp[5] }];  
#set_property PACKAGE_PIN AJ2  [get_ports { o_pcie_txp[6] }];  
#set_property PACKAGE_PIN AK4  [get_ports { o_pcie_txp[7] }];
```

```
set_property PACKAGE_PIN Y4  [get_ports { i_pcie_rxp[0] }];  
#set_property PACKAGE_PIN AA6  [get_ports { i_pcie_rxp[1] }];  
#set_property PACKAGE_PIN AB4  [get_ports { i_pcie_rxp[2] }];  
#set_property PACKAGE_PIN AC6  [get_ports { i_pcie_rxp[3] }];  
#set_property PACKAGE_PIN AD4  [get_ports { i_pcie_rxp[4] }];  
#set_property PACKAGE_PIN AE6  [get_ports { i_pcie_rxp[5] }];  
#set_property PACKAGE_PIN AF4  [get_ports { i_pcie_rxp[6] }];  
#set_property PACKAGE_PIN AG6  [get_ports { i_pcie_rxp[7] }];
```

Host PC Driver

上電燒錄後，須確保FPGA不斷電的情況下重啟Ubuntu主機

重啟後先尋找是否有辨識到Xilinx PCIe Device

```
$ lspci
...                                     # Other PCI Device
01:00.0 Memory controller: Xilinx Corporation Device 7021    # Xilinx PCIe-XDMA Device
...                                     # Other PCI Device
```

下載Xilinx提供的dma driver

https://github.com/Xilinx/dma_ip_drivers

cd至以下目錄中，並編譯xdma driver

```
$ cd xdma
$ sudo make install
$ cd tools
$ sudo mak
```

Load Kernal Driver

```
$ sudo modprobe xdma
```

使用xilinx提供的測試app來讀寫

```
$ ./load_driver.sh
$ ./run_test.sh
```

