

FPGA Homework4 第八組

F44071128 李其佑

C14074021 張柏彥

F14051041 陳 祺

Problem1. Block Ram Utilize

一、 Verilog Template 說明

在 top module “MEM” 底下引用 RAMB36E1 的 template，將所需的訊號設為 top module 的 I/O。並依照題目規格修改 template 的 Attributes。

```
module MEM(  
    input CLK_A, CLK_B, ENA, ENB,  
    input [31:0] AIN, BIN,  
    input [11:0] BRAM_PORTA, BRAM_PORTB,  
    input [3:0] WEA, WEB,  
    output [31:0] AOUT, BOUT  
);
```

MEM			
Signal Name	Direction	Width	Description
CLK_A	Input	1	PORT A 之時脈訊號
CLK_B	Input	1	PORT B 之時脈訊號
ENA	Input	1	PORT A 致能訊號
ENB	Input	1	PORT B 致能訊號
AIN	Input	32	Port A 寫入資料
BIN	Input	32	Port B 寫入資料
BRAM_PORTA	Input	12	Port A 地址
BRAM_PORTB	Input	12	Port B 地址
WEA	Input	4	PORT A 寫入致能訊號
WEB	Input	4	PORT B 寫入致能訊號
AOUT	Output	32	PORT A 輸出資料
BOUT	Output	32	PORT B 輸出資料

1. Data Width : 32bit

將 Attributes 中 READ_WIDTH/WRITE_WIDTH 設為 36

```
.READ_WIDTH_A(36), // 0-72
.READ_WIDTH_B(36), // 0-36
.WRITE_WIDTH_A(36), // 0-36
.WRITE_WIDTH_B(36), // 0-72
```

2. Memory Size : 32Kbit

只引用一個 RAMB36E1，自動滿足 32Kbit 之需求

3. RAM Mode : TDP

將 Attributes 中 RAM_MODE 設為”TDP”

```
.RAM_MODE("TDP"),
```

4. Initial Contents

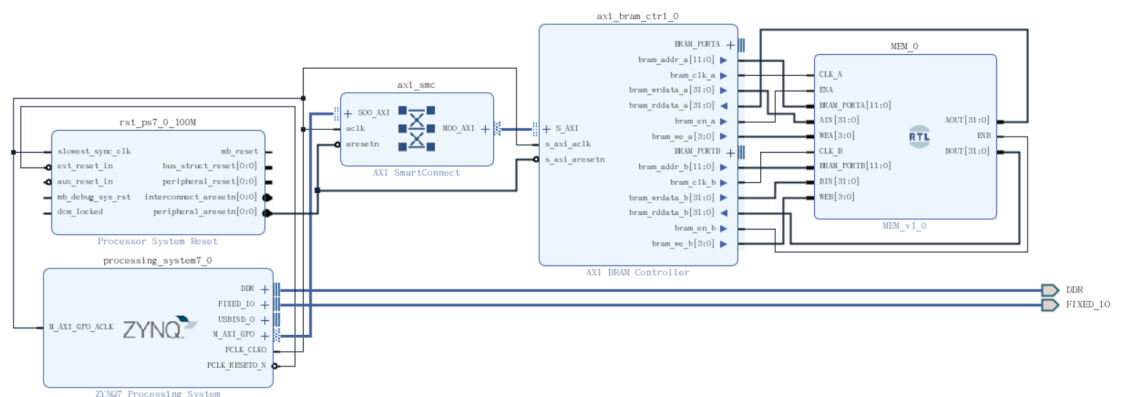
```
.INIT_00(256'h00002379_00000000_00000000_00000000_00000000_00000000_00002454_00002330),
.INIT_01(256'h00000000_00000000_00000000_00000000_00000000_00000000_00000000_00000000),
.INIT_02(256'h00000000_00000000_00000000_00000000_00000000_00000000_00000000_00003034),
```

5. top module I/O 與 RAMB36E1_inst 連接

```
RAMB36E1 #(
    ....
)
RAMB36E1_inst (
    ....
    // Port A Data: 32-bit (each) output: Port A data
    .DOADO(AOUT), // 32-bit output: A port data/LSB data
    // Port B Data: 32-bit (each) output: Port B data
    .DOBDO(BOUT), // 32-bit output: B port data/MSB data
    .ADDRARDADDR( {1'b0, BRAM_PORTA[11:0], 3'b000} ), // 16-bit input: A port address/Read address
    .CLKARDCLK(CLK_A), // 1-bit input: A port clock/Read clock
    .ENARDEN( ENA ), // 1-bit input: A port enable/Read enable
    .REGCEAREGCE( ENA ), // 1-bit input: A port register enable/Register enable
    .WEA( WEA ), // 4-bit input: A port write enable
    // Port A Data: 32-bit (each) input: Port A data
    .DIADI( AIN ), // 32-bit input: A port data/LSB data
    .ADDRBRWADDR( {1'b0, BRAM_PORTB[11:0], 3'b000} ), // 16-bit input: B port address/Write address
    .CLKBWRCLK( CLK_B ), // 1-bit input: B port clock/Write clock
    .ENBWREN( ENB ), // 1-bit input: B port enable/Write enable
    .REGCEB( ENB ), // 1-bit input: B port register enable

    .WEBWE( {4'd0, WEB} ), // 8-bit input: B port write enable/Write enable
    // Port B Data: 32-bit (each) input: Port B data
    .DIBDI( BIN ), // 32-bit input: B port data/MSB data
);
```

二、 Block Diagram



三、 測試程式

先讀取有設定初始資料的位置確認是否有成功初始化記憶體，在寫入資料後進行讀取，確認記憶體讀寫是否正確。這裡我們寫入的資料和 offset 一樣，因此讀出結果如下圖所示，data 和 offset 是相同的，也就代表讀寫沒問題。

```
Hw4 Initial Contents Test Start.
offset = 0, Data = 2330
offset = 4, Data = 2454
offset = 28, Data = 2379
offset = 64, Data = 3034
Hw4 Initial Contents Test End.
Hw4 Write Test Start.
offset = 0, Data = 0
offset = 4, Data = 4
offset = 8, Data = 8
offset = 12, Data = c
offset = 16, Data = 10
offset = 20, Data = 14
offset = 24, Data = 18
offset = 28, Data = 1c
offset = 32, Data = 20
offset = 36, Data = 24
offset = 40, Data = 28
offset = 44, Data = 2c
offset = 48, Data = 30
offset = 52, Data = 34
offset = 56, Data = 38
offset = 60, Data = 3c
offset = 64, Data = 40
Hw4 Write Test End.
```

Problem

1. $36\text{Kbit} * 140 = 5040\text{Kbit} = 630\text{KB}$

ZYNQ XC7Z020-1CLG400C

- 650MHz dual-core Cortex-A9 processor
- DDR3 memory controller with 8 DMA channels and 4 High Performance AXI3 Slave ports
- High-bandwidth peripheral controllers: 1G Ethernet, USB 2.0, SDIO
- Low-bandwidth peripheral controller: SPI, UART, CAN, I2C
- Programmable from JTAG, Quad-SPI flash, and MicroSD card
- Programmable logic equivalent to Artix-7 FPGA
 - 13,300 logic slices, each with four 6-input LUTs and 8 flip-flops
 - 630 KB of fast block RAM
 - 4 clock management tiles, each with a phase locked loop (PLL) and mixed-mode clock manager (MMCM)
 - 220 DSP slices
 - On-chip analog-to-digital converter (XADC)

2. 140 個

BRAM	1	140	0.71
------	---	-----	------

3. 使用 RAMB36E1 組成 FIFO 之 Verilog Template 說明
(假設 data width 為 32bit，PORT A 讀取、PORT B 寫入)

```

RAMB36E1 #(
    .RDADDR_COLLISION_HWCONFIG("DELAYED_WRITE"),
    .SIM_COLLISION_CHECK("ALL"), //模擬警告設為ALL可避免記憶體衝突
    .DOA_REG(0),
    .DOB_REG(0),
    .EN_ECC_READ("FALSE"),
    .EN_ECC_WRITE("FALSE"),
    // INITP_00 to INITP_0F: Initial contents of the parity memory array 00 //FIFO不須初始值
    // INIT_00 to INIT_7F: Initial contents of the data memory array 00 //FIFO不須初始值
    .INIT_A(36'h00000000),
    .INIT_B(36'h00000000),
    .INIT_FILE("NONE"), //FIFO內部資料不須初始化
    .RAM_MODE("TDP"), //必須為TDP才能同時讀寫
    .RAM_EXTENSION_A("NONE"), //32bit的FIFO不需cascade
    .RAM_EXTENSION_B("NONE"), //32bit的FIFO不需cascade
    // READ_WIDTH_A/B, WRITE_WIDTH_A/B: Read/write width per port
    .READ_WIDTH_A(36), //PORT A可以讀取資料
    .READ_WIDTH_B(0), //PORT B無法讀取資料
    .WRITE_WIDTH_A(0), //PORT A無法寫入資料
    .WRITE_WIDTH_B(36), //PORT B可以寫入資料
    .RSTREG_PRIORITY_A("RSTREG"),
    .RSTREG_PRIORITY_B("RSTREG"),
    .SRVAL_A(36'h00000000), //reset時FIFO輸出為0
    .SRVAL_B(36'h00000000), //reset時FIFO輸出為0
    .SIM_DEVICE("7SERIES"),
    .WRITE_MODE_A("WRITE_FIRST"),
    .WRITE_MODE_B("WRITE_FIRST")
)

```

```

RAMB36E1_inst (
    ...
    // Port A Data: 32-bit (each) output: Port A data
    .DOADO( FIFO_OUT ), // 由PORT A讀取FIFO中的資料
    .ADDRARDADDR( rd_ptr ), //FIFO內部紀錄讀取位置之指針連接至PORT A之地址
    .CLKARDCLK(CLK_A), // 時脈訊號
    .ENARDEN( rd_en ), // FIFO內部控制訊號 當要讀取時須將PORT A設為有效
    .WEA( 4'b0000 ), // PORT A不需寫入資料

    .ADDRBWADDR( wr_ptr ), //FIFO內部紀錄寫入位置之指針連接至PORT B之地址
    .CLKBWRCLK(CLK_B), // 時脈訊號
    .ENBWREN( wr_en ), // FIFO內部控制訊號 當要寫入時須將PORT B設為有效

    .WEBWE( WEB ), //當要寫入資料時FIFO內部控制訊號需控制此訊號
    // Port B Data: 32-bit (each) input: Port B data
    .DIBDI( FIFO_IN ), // 由PORT B將資料寫入FIFO
);

```