**ELECTRICAL ENGINEERING 1 － Control Circuits of Memory Data Scan and Output**

【Purpose】

Design a simple memory control circuit.

1. Learning basic memory theory.
2. Use Quartus create memory module.
3. Using a thermistor to experience the results of analog-to-digital conversion.

【Experiment Background】

Verilog design experience.

【Principle and Description】

1. 一張含有 文字, 行, 圖表, 繪圖 的圖片

   自動產生的描述Memory theory

The memory module has 5 I/O signals that the Quartus megafunction generates : Input data(data), Write enable(ween), Adress(address), Clock(clock), Output data(q).

The module combines two parts : (1)Memory signal control circuit, (2)Memory unit.

**Memory signal control circuit**

Use three latches and clock control data, address, write enable. And send the signal to memory unit.

**Memory unit**

The memory unit actions based on control signals generated by memory signal control circuit.

1. If write enable signal start, write data to the address signal location.

2. Output data according to address signal.

**Memory module**

Write: When the write enable starts and waits for the clock signal, write data to the location specified by the address signal line.

Read: When clock signal trigger, output the data to output data signal specified by the address signal line.

一張含有 螢幕擷取畫面, 行, 平行, 數字 的圖片

自動產生的描述**Waveform simulation**

In mark 1, when the positive edge of the clock trigger and scan write enable signal(write\_en) is set to 1, output the data\_in value(AA) from location the address specified by the address signal(01).

Mark 2 same as mark 1, clock trigger and write enable signal is set to 1, output data(BB) to the address signal location.

In mark 3, when the positive edge of the clock trigger, and the write enable is set to 0. Send the data "AA" to the data\_out signal where the address signal is set to "01".

2. Use Quartus Create Memory

**Step1 :** Quartus [File] -> New project wizard … -> FPGA type : **Cyclone V - 5**   **CSEMA5F31C6** -> [File] -> New -> Block Diagram / Schematic File  
 (Key : File paths should not contain Chinese characters)

一張含有 文字, 圖形, 字型, 平面設計 的圖片

自動產生的描述**Step2 :** Use Mega Function create memory module  
 Library -> Basic Function -> On Chip Memory -> RAM1-PORT

一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

自動產生的描述

一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

自動產生的描述After double clicks RAM-PORT.

Memory name can not same as file name .

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述File type choose Verilog.

Set bus wide and size. We use 8 bits because the output need show 2 hexadecimal in this experiment.

一張含有 文字, 螢幕擷取畫面, 軟體, 陳列 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述Set output latch, we do not need it in this experiment.

At this page, click next.

一張含有 文字, 螢幕擷取畫面, 軟體, 網頁 的圖片

自動產生的描述Choose \*.mif file.(Provide by TA), The path cannot contain Chinese.

一張含有 文字, 電子產品, 螢幕擷取畫面, 軟體 的圖片

自動產生的描述At this page, click next.

一張含有 文字, 電子產品, 螢幕擷取畫面, 軟體 的圖片

自動產生的描述Tick to create symbol.

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述 If forgot tick the option, you can find the verilog file and right click it, choose "Create Symbol Files for Current File".

一張含有 文字, 螢幕擷取畫面, 陳列, 圖表 的圖片

自動產生的描述**Step3 :** Put the memory module in .bdf file, complete create.

3. Wiring

As the circuit design use more and more complex, it becomes necessary to convert Verilog code into symbol and put integrate them into Block Diagram / Schematic File for easier management.

一張含有 文字, 軟體, 網頁, 電腦圖示 的圖片

自動產生的描述**Step 1:** Compile the Verilog code and back to Files directory, right click and choose "Create Symbol Files for Current File".

**Step2 :** In .bdf file, Double-clicks on Block Diagram empty space, choose symbol and press OK.一張含有 文字, 螢幕擷取畫面, 軟體, 陳列 的圖片

自動產生的描述

一張含有 文字, 圖表, 螢幕擷取畫面, 行 的圖片

自動產生的描述**Wiring:** Create port to every component. EX: We use the tool inside the blue box connect inst1 and inst2.

一張含有 文字, 螢幕擷取畫面, 圖表, 數字 的圖片

自動產生的描述**Name:** Click wire and name. The lines on the diagram don't need to be connected, this make the diagram cleaner.

**Input / Output Pin :** Used for external connections to the circuit

一張含有 文字, 螢幕擷取畫面, 陳列, 軟體 的圖片

自動產生的描述**Step 1:** Double-clicks on empty space to open libraries. There are 3 types Pins can choose, bidir, input and output. You can also search by entering the component's name.

**Search by name**

一張含有 螢幕擷取畫面, 行, 圖表, 文字 的圖片

自動產生的描述**Step 2 :** Connect Input / Output Pin and connect the circuit.

Use a single bold line represent multiple ports, we call "Bus". The format is： Name[MSB.. LSB]，EX: wire[2..0].

一張含有 螢幕擷取畫面, 行, 圖表, 繪圖 的圖片

自動產生的描述

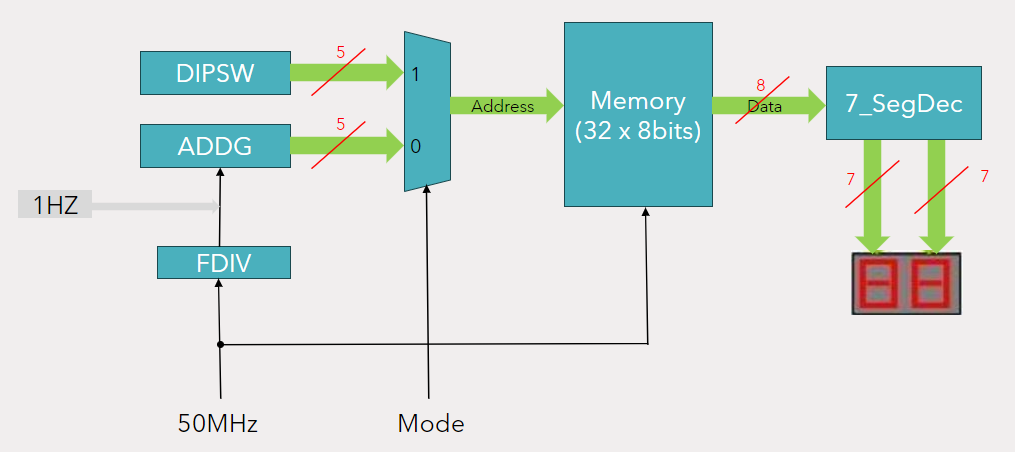
【Implementation】

Design a memory output control circuit and change memory address. Show output data on 7\_segment.

This implementation have **2 mode**:

1. Use Dip-Switch(DIPSW) as address input, show output data on 7\_segment.

2. Use Address Generator auto generate address input, show output data on 7\_segment.



The circuit has 6 module:

1. Frequency Divider(FDiv) : Convert the frequency of 50Hz to 1Hz.

2. Address Generator(ADDG) : Automatically generate memory address.

3. Memory(MEM) : Use megafunction generate memory module.

4. 7\_Segment decoder(7\_segDec) : Decode the memory output data and show on 7\_Segment.

5. 2-to-1 Multiplexer(2to1Mux) : Choose the memory address input source.

6. Dip Switch(DIPSW) : Use the FPGA board's dip switch to generate address.

**Implementation objectives**

1. Use Verilog to complete every module on the diagram.

2. Design a simple address generator that adds one in every positive edge clock, return to 0 when counting to 31.

**Memory** : The memory write enable(wren) and data input(data) just connect to the ground.

**Bonus** : Implement any creative or additional features. EX : Display output data in decimal or LED.

Report : Complete diagram, waveform simulation for each module(need explanation), every module's Verilog code and comments, Introduction to Bonus Features, implementation reflection.