

# CA pj2 report

## Members & Team Work

B04902003 董書博 33% 接線、debug、report

B04902017 李立譽 33% read cache、write cache、report

B04902043 謝宏祺 33% debug、solve problem、report

## How do you implement this project?

1. 將原先CPU.v中接在Data\_Memory的線改接到dcache\_top。
2. 在各個pipeline新增stall\_i。
3. 將dcache\_top的功能補齊。

## Cache Controller in detail

以下為這次project修改的檔案

### 1. dcache\_top.v

- 決定hit 的值以及data的內容

```
1 assign hit = (p1_tag == sram_tag && sram_valid) ? 1'b1 : 1'b0;
2 assign r_hit_data = (hit) ? sram_cache_data : 256'b0;
```

- 從cache讀出32bit的data
  - (a)為了將byte轉成bit，將要讀的offset往左移三位(\*8)
  - (b)將cache中的data右移後取值

```
1 wire [31:0] actual_offset;
2 reg [255:0] temp_read;
3 assign actual_offset = {{27{1'b0}}, p1_offset} << 3; // byte => bit
4 always@(p1_offset or r_hit_data or actual_offset) begin
5     //!!! add you code here! (p1_data=...?)
6     temp_read = r_hit_data[255:0] >> actual_offset;
7     p1_data[31:0] = temp_read[31:0];
8 end寫32bit的data至cache中
```

- 寫32bits的data至cache中
  - (a)將cache中，要write的區域全部變為0(利用bin\_window)
  - (b)將data寫入(因為前一步驟已歸零，因此直接or運算就完成了)

```
1 reg [255:0] temp_write;
2 reg [255:0] bin_window;
3 reg [255:0] bin_value;
4 always@(p1_offset or r_hit_data or p1_data_i or actual_offset) begin
5     //!!! add you code here! (w_hit_data=...?)
6     bin_window = ~({{224{1'b0}}, {32{1'b1}}} << actual_offset); //1111
7     bin_value = {{224{1'b0}}, p1_data_i} << actual_offset;
```

```
8     temp_write = r_hit_data[255:0] & bin_window[255:0];  
9     w_hit_data[255:0] = temp_write | bin_value;  
10 end
```

## Problems and solution of this project

- Problem1 : DataMemory中的ack一直沒有送出，一直是0。
  - Solution1 : 我們在 DataMemory.v 中發現 always裡頭少了一個(negedge rst\_i)，少了這個導致匯進不去if(~rst\_i)，進而導致count及state都無法初始化，然後data memory就出錯了。
- Problem2 : PC counter 一直是 0，不會動。
  - Solution2 : 我們在寫的時候剛好有兩個人用著不同版本的iverilog，結果我們發現在版本10.x的情況下PC.v 中的 write\_i 會一直是未定義。而另一位組員用的版本是 0.9.7 卻無此情況發生。在並不是很清楚iverilog 對於一些參數的初始化的情況下，我們決定使用 0.9.7 這個版本，而程式跑出來的final state也是正確的。