Computer Architecture Project 2 Report

楊傑森 Jason Yeo B07902101

OS 作業系統: Windows

Implementation of new modules & changes (比對 Project 1: Pipelined CPU without cache):

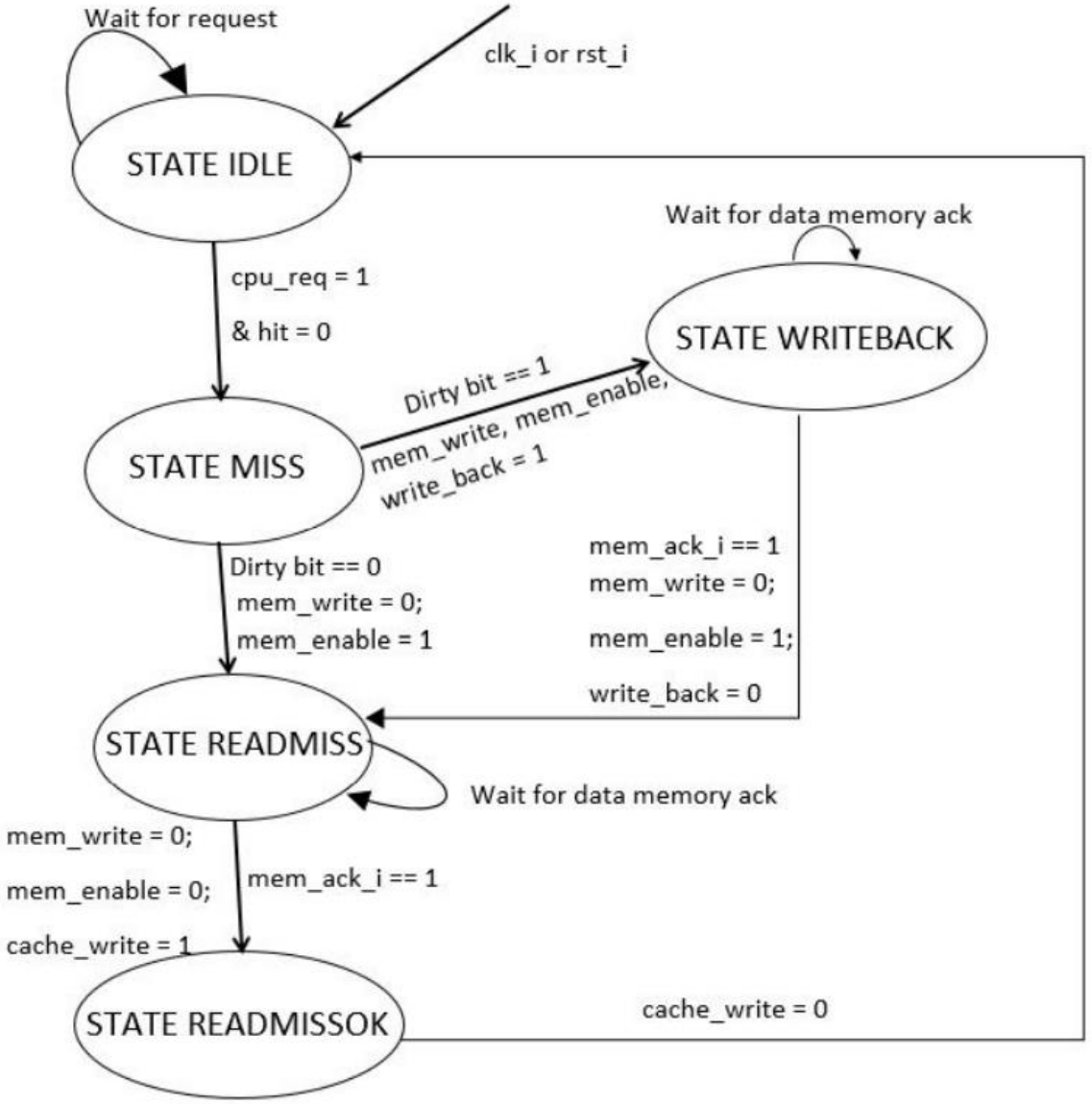
1) Dcache_controller (include dcache_sram)

(參考下面 dcache_sram): If hit = 1, r_hit_data is from cache.

Check offset to read from 256-bit data to 32-bit data.

Check offset to read from 32-bit data to 256-bit data.

Deal with cache miss. Details + Diagram provided as below



i) STATE_ IDLE is the initial state. It will go to State MISS when it receives a read/write request and cache miss.

ii) In STATE_MISS, if sram_dirty is set to 1, cache data block needs to write back to data memory. mem_write, mem_enable, write_back are all set to 1. Then go to STATE_WRITEBACK.

iii) If sram_dirty is set to 0, mem_write is set to 0, mem_enable is set to 1.

iv) In STATE_WRITEBACK, it waits until the data memory ack is set to 1. If ack = 1, mem_write is set to 0, mem_enable is set to 1, write_back is set to 0. Go to STATE_READMISS.

v) In STATE_READMISS, it waits until the data memory ack is set to 1. If ack = 1, mem_write is set to 0, mem_enable is set to 0, cache_write is set to 1. Go to STATE_READMISSOK.

vi) STATE_READMISSOK will go back to STATE_IDLE and cache_write is set to 0.

2) Dcache_sram

Handles tag & data together.

Reset signal = 1，lru, tag, data initialized to 0.

LRU 記錄上個使用的 sram,交替使用兩塊不同

因為考慮 2-way associative cache, hit 的 signal 由以下操作決定:

2 個 input 分為 hit0 和 hit1, 至少其中一個 input signal 是 1, hit 的 signal 就是 1。

hit0 和 hit1 由以下操作決定:

2 個 tag bits 都一樣 && valid bit 的值是 1，hit0 / hit1 的 signal 就是 1。

不滿足其中一個條件則 0。

data_o 和 tag_o 的值由 hit0 決定,

hit0 是 1, 選擇第一個 tag&data, 否則選第二個

3) Other modules (PC + 4 個 pipeline registers)

Stall when a new stall signal (MemStall) is received because of cache miss.

Difficulties while debugging & solutions 除錯時出現的問題和解決方案:

疑:

1) output.txt 在第 2 個測資&第 2s 個測資部分 registers 的值前面 2 個值應該是 ff, 不是 00。

2) 所有的測資建立的 output.txt 的 Data Memory（x200 和 x220）的值沒有更新。

解:

1) 修改 ALU 的 syntax, shift right arithmetic 的問題

2) 在 testbench 自行初始化