計算機結構 作業五

B03901026 許凱傑

I In this homework, I implement two different architectures of direct-mapped cache. The only difference between them is whether input and output are registers. However, proc_reset and clk are not register type in both cases. In the below discussions, I will use cache denoting cache with I/O registers and cacheNoblock denoting cache without I/O registers.

II Report

1 General specification of the cache unit

A cache

a. Direct-mapped, 8 blocks with 4 words
b. Write back policy: only write the data in the block back to memory when the block is dirty and is going to be replaced.
c. Since this design is direct-mapped, there is no choice for replacement policy
d. Cell area: 529582 μm^2
e. Shortest Simulation Cycle: 5.8 ns
f. Total Simulation Time: 93563 ns
g. Total Power: 8.9340 mW

B cacheNoblock

a. Direct-mapped, 8 blocks with 4 words
b. Write back policy: only write the data in the block back to memory when the block is dirty and is going to be replaced.
c. Since this design is direct-mapped, there is no choice for replacement policy
d. Cell area: 460661 μm^2
e. Shortest Simulation Cycle: 5.0 ns
f. Total Simulation Time: 50763 ns
g. Total Power: 7.2501 mW

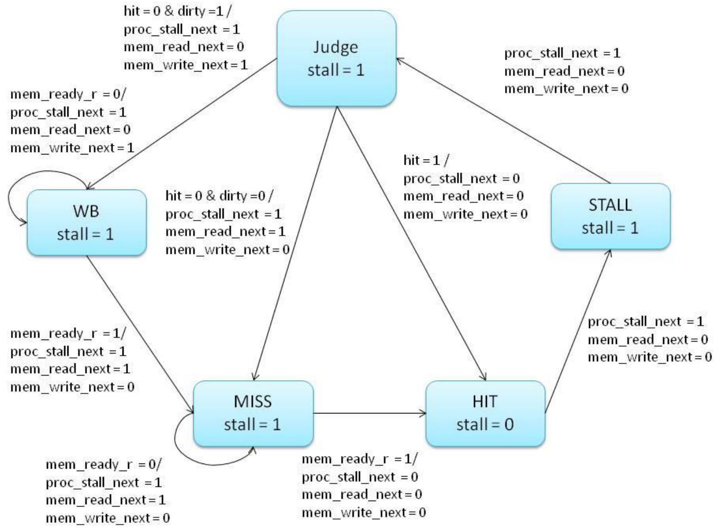
2 Design Architecture

A cache

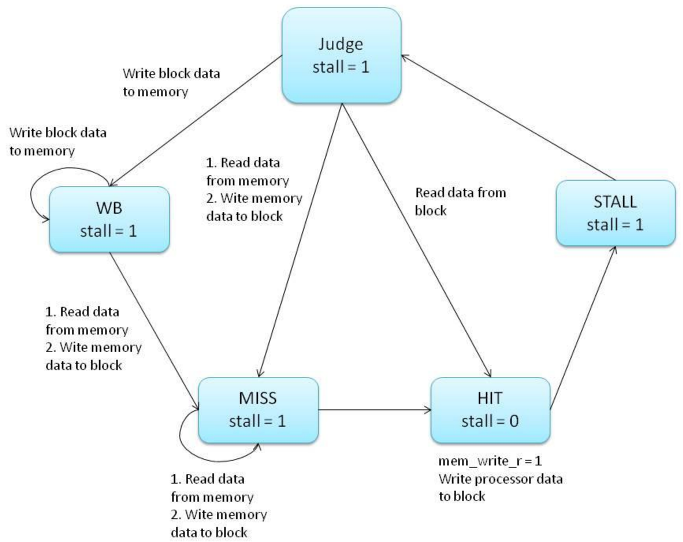
a. Finite State Machine

計算機結構 作業五

B03901026 許凱傑



b. Executions

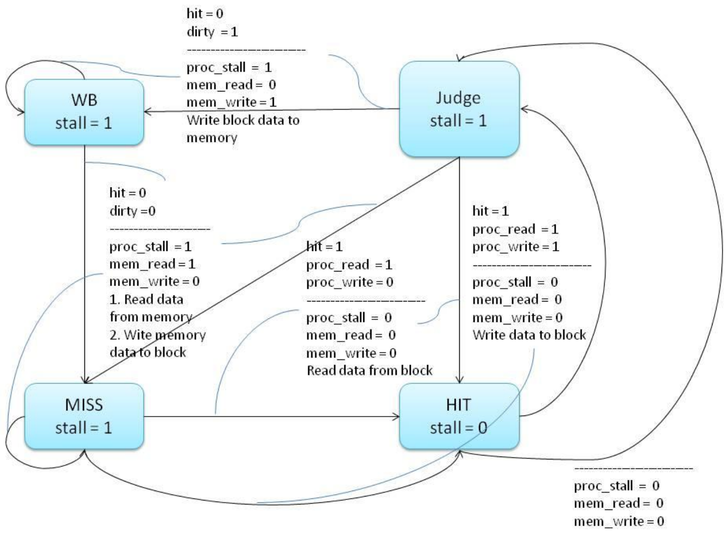


B cacheNoblock

a. Executions

計算機結構 作業五

B03901026 許凱傑



Because this architecture doesn't have flip-flops for input, output and state, there are no states. The state name inside the blue block is only for discussion.

3 *Performance evaluation of your cache design, including the miss rates of read/write operations, the execution cycles, the stalled cycles, and so on.*

A cache

a. Read/Write Hit/Miss

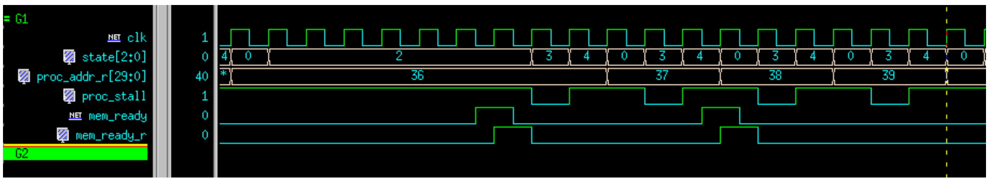
Read Hit	Read Miss	Write Hit	Write Miss
1536	512	768	256

Read miss rate : 25%

Write miss rate : 25%

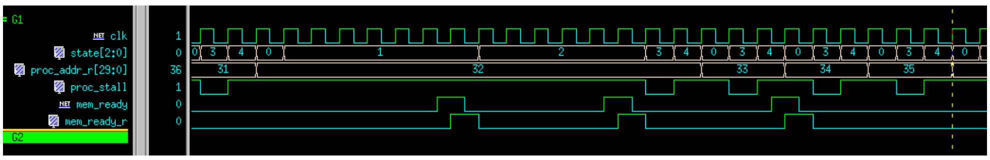
b. Execution /Stalled Cycles

i. Miss



Since inputs are flip-flops, the state 4 is needed to prevent from cache to use the information of last address. As the result, there are 15 stalled cycles and 4 execution cycles.

ii. Write Back + Miss



There are 22 stalled cycles and 4 execution cycles.

計算機結構 作業五

B03901026 許凱傑

iii. Summary

Stalled Cycles： $15 \times 256 \times 2 + 22 \times 256 = 13312$

Execution Cycles： $4 \times 256 \times 3 = 3072$

B　cacheNoblock

a.　Read/Write Hit/Miss

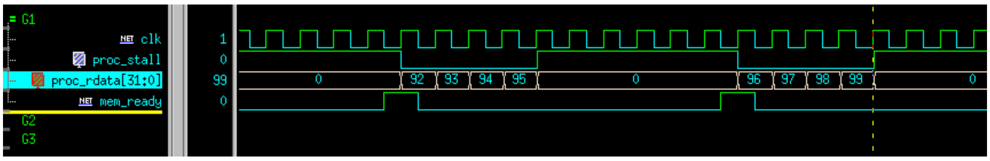
Read Hit	Read Miss	Write Hit	Write Miss
1536	512	768	256

Read miss rate : 25%

Write miss rate : 25%

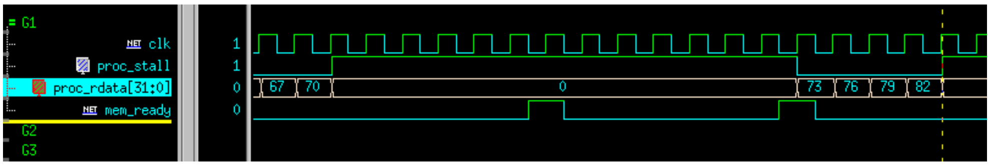
b.　Execution /Stalled Cycles

i. Miss



There are 6 stalled cycles and 4 execution cycles.

ii. Write Back + Miss



There are 13 stalled cycles and 4 execution cycles.

iii. Summary

Stalled Cycles： $6 \times 256 \times 2 + 13 \times 256 = 6400$

Execution Cycles： $4 \times 256 \times 3 = 3072$

III　Discussions

在這次作業中主要出現了幾個問題，第一個困惑最久的就是合成上的小失誤 +define+SDF 少寫一個 ”+”，導致不管怎樣都會有 timing violation，然後後來想想這好像剛好就是助教提到的沒有給 SDF information，模擬就會自動套用 tsmc13.v 的 default，所以難怪怎樣都不會過。

此外，原本以為 asynchronous reset 可以採用在 combinational 寫 reg_next = *，再把 reg <= reg_next 這種寫法，但發現後來合成完的跑去模擬會發現無法 reset，

計算機結構 作業五

B03901026 許凱傑

這裡不知道是我的邏輯有問題，還是就是這個語法軟體沒辦法接受，可能禮拜四要再和助教討論。

最後是拿去合成時，發現會有 LATCH 的產生，所以後來有把每個 register 的 input 都給定一個 default 值，這樣就可以避免 LATCH 的產生。還有合成完會有一些 warning，像是 high fanout net 或是 1 output port was not connected，這些其實不用太擔心，可能只是優化的時候有把它合併。

在這次作業中也有一些收穫，像是原本覺得 wire 不能拿來當 array 的 index，但後來發現軟體很聰明應該是會自動把它轉換成跟 case 的寫法會有一樣的結果。還有實作了擋 flip-flop 的版本，也做了 FSM，對於這種會有延遲 cycle 的設計，透過這次作業感覺有比較熟悉，而且透過在 miss 的最後一個 cycle 提前把 proc_rdata_next 準備好，便能少一個 stalled cycle，這種類似 forwarding 的機制，感覺對於 verilog 又有小小的進步所以蠻開心的。

不過這次兩個 architecture 的比較，就發現擋 flip-flop 如果不是搭配 pipelined 的系統，它的 performance 真的很差，除了 cycle 比較長，stalled cycle 數也很多，在這次的 test bench，更是快要兩倍的 simulation time，但優點可能就是很穩定吧!不管別的 module 最後有沒有擋 flip-flop 都可以有完整的 cycle time。