HW3 Cache Optimization

楊永琪, 109654020

***Abstract*—此為作業三報告, 嘗試分析 Aquila FIFO Cache Replacement hit, miss 狀況, 並嘗試其他 Cache Replacement 的機制。**

I. Introduction

此次作業為分析在 Aquila 上使用 2KB Dcache 的跑 CoreMark 的效能與 cache hit/miss 的行為。以下分為五個部分：第二部分會針對原先於 Aquila 上 FIFO 的 2K Dache 嘗試不同的 way set associated，並作分析。第三部分則是在 Aquila 上嘗試 Least Recently Used Cache Replacement 的實現與結果。第四部分則是 D cache FIFO、LRU、Direct Mapping 以及使用 TCM 的效能比較。

II. Performance Measurements using FIFO D$

A. Implementation

此報告分析中的 hit_count, miss_count 是計算當 Dcache 內部 (S == Analysis && ~ p_is_amo_i) 時是否有 cache hit 的數量。由於當一有外部 strobe 的訊號, state S 就會從 Idle 轉到 Analysis 並決定在下一個 cycle 是要從 DRAM 讀資料或是要將 dirty data 先寫回 DRAM, 故每次的 memory request 都只會到 Analysis state 一個 cycle。Write_dirty 與 read_dirty 是於 S == Analysis 時看 c_dirty_o[victim_sel] 的值計算。Latency 則是計算從 S == Analysis 到包含觸發 p_ready_o 的下個 cycle 之間的 cycle 數 (並以多出的 cycle 視為 p_strobe_i 當下的 cycle)，其所代表的意義為從 CPU 發出 memory request 到收到資料之間共經歷的 cycle 數。

B. Average Cache Latency for each memory request

TABLE I. Average Latency Among Different Scenarios

	total count	*latency*	*average latency(cycles)*
Hit	***90,808,452***	***181,616,904***	***2***
Miss Clean	***282***	***9,199***	***32.62***
Miss Dirty	***199,179***	***10,235,495***	***51.39***
Flush	***1***	***2435***	***2435***

Table I 的數據為使用 FIFO 4-way 2KB Dcache 取得, 計算範圍為從最一開始到 Coremark main function 結束。從由表得知 hit 的 latency 為 2 個 cycles, Read Miss 與 Write Miss 的 latency 為 32.6 個 cycles, 在 Read / Write Miss 並且 victim cell 為 dirty 的情況平均會花較多的時間, 約 51.4 個 cycles, 估計將一個 cache block寫回 DRAM 需約 18.8 個 cycles。

而如果有 fence 指令, 則會須把 cache 內 dirty 的資料全部寫入 DRAM 內, 共花了個 2435 個 cycles, 並因為 2KB 的 D cache 中有 128 cache block, 而 fence.i 指令是發生在將 .elf 檔 load 進 main memory 並初始化未用到的 memory 之後, 故 D$ 內部的 cache block 皆為 dirty, 平均下來每個 block 被寫回為 19.02 cycles 與所預估的相近。

C. Cache Miss/Hit rate

TABLE II. Hit / Miss Rate with FIFO 2KB D$

	Direct Mapping / FIFO D$ Hit-Miss Rate						
	total count	*Hit (%)*		*Miss(%)*		*Miss (Dirty %)*	
		Read	*Write*	*Read*	*Write*	*Read*	*Write*
direct	***90,979,579***	***79.001***	***20.422***	***0.478***	***0.099***	***99.995***	***99.859***
2-way	***90,979,577***	***79.241***	***20.489***	***0.238***	***0.032***	***99.989***	***99.575***
4-way	***90,979,577***	***79.297***	***20.486***	***0.182***	***0.035***	***99.979***	***99.632***
8-way	***90,979,577***	***79.301***	***20.482***	***0.178***	***0.039***	***99.977***	***99.675***

表二為使用 FIFO 2KB D$ 的 Hit/Miss Rates 資訊, 計算範圍為 Coremark Main 開始與結束之間。其中 Hit Rate 佔比很高, 介於 99.42% ~ 99.81% 之間。而 Miss 中 Dirty 個數的佔比也很高, 約都在 99% 左右。

III. LRU Cache Replacement

A. Algorithm

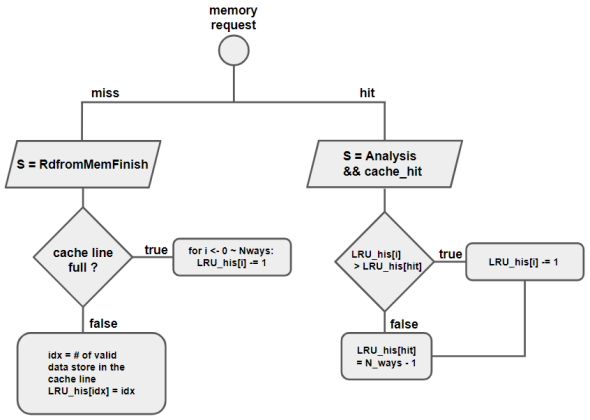

Fig I. LRU Implementation Flow Chart

在實現 LRU 時主要為利用一個 register table 紀錄每個 way 的 history, 每個 history 使用 lg(N_ways) 紀錄。其流程大致如上圖, victim_sel 的選擇為使用 LRU_his[idx] = 0 對應的 cell, 而當每次的 memory request 完成時則會更新 history 裡面的相對順序。

B. Results

TABLE III. LRU D$ HIT/MISS RATE

	total count	*Hit (%)*		*Miss (%)*		*Miss (Dirty) (%)*	
		Read	*Write*	*Read*	*Write*	*Read*	*Write*
2-way	*90,979,577*	*79.241*	*20.489*	*0.238*	*0.032*	*99.989*	*99.575*
4-way	*90,979,577*	*79.315*	*20.494*	*0.164*	*0.027*	*99.430*	*99.520*
8-way	*90,979,577*	*79.313*	*20.498*	*0.166*	*0.023*	*99.440*	*99.425*

表三為使用 LRU replacement policy 的 Hit/Miss Rates 結果, 雖然數據上有些許的不同, 普遍上 LRU 的 hit count 都比 FIFO 高一些, miss count 也都比較少一點, 但其統計出來的結果與 FIFO 相似。

IV. PERFORMANCE

TABLE IV. PERFORMANCE AMONG DIFFERENCE CACHE REPLACEMENT POLICY

	Iterations / Sec Results on CoreMark Test		
	2-way	*4-way*	*8-way*
FIFO	*24.758691*	*24.906594*	*24.905866*
LRU	*24.820225*	*24.949631*	*24.952513*
Direct	*24.333099*		
TCM	*25.264950*		

TABLE V. AVERAGE LATENCY (CYCLES) PER MISS

	2-way	*4-way*	*8-way*
FIFO	*51.30*	*51.39*	*51.30*
LRU	*51.39*	*51.28*	*51.19*
Direct	*51.41*		

由表四可看出使用 TCM on-chip memory 的效能最好。而 Direct Mapping 效果最差。在 FIFO replacement policy 下 4-way associated 的效果比較好, 8-way associated 的效果比較差一點;而 LRU 8-way 反而比 4-way 還要好一些, 整體來說 LRU 平均一秒的 iterations 次數會比 FIFO 高一點。

結合表二與表三一起看的話會發現使用 2KB 的 D$ Hit Rate 都很高, 但由於單次的 Miss Latency (Not Dirty) 為 hit 的 12 倍, Miss Dirty Latency 為 hit 的 26 倍左右, 並且在大部分 miss 的情況下 victim cell 都為 dirty, 因此從表五可以看出每個 miss 平均的 latency 皆約 51 cyles 左右。以表二 FIFO 2KB 4-way 的統計結果來看, miss count 只佔 0.21%, 卻佔約全部 latency 的 5.29%。