pr5 Simulator - Part 4 (Optimizations)

Sandeep Chandran

Deadline: 27-Nov 2024, 23:59 hrs

Lab Assignment 7 (Graded)

In this lab, you are required to implement caches into the RISCV-V processor simulator developed so far. The detailed specification is given below.

- You have to implement all three cache mapping policies:
  1. **`DM`** – A direct-mapped cache.
  2. **`FA`** – A fully-associative cache.
  3. **`SA`** – A set associative cache.
- You have to implement both the write policies:
  1. **`WT`** – Write-through cache.
  2. **`WB`** – Write-back cache.
- You have to implement the following replacement policies:
  1. **`RD`** – Randomly select a block.
  2. **`FI`** – FIFO when selecting blocks.
  3. **`LR`** – Least Recently Used block is selected.
  4. **`LF`** – Least Frequently Used block is selected.
- Take three additional command-line arguments:
  1. **`--l1i`** – Specifies the design of the L1 Instruction Cache.
  2. **`--l1d`** – Specifies the design of the L1 Data Cache.
  3. **`--l2`** – Specifies the design of the Unified L2 Cache.

  The exact design of the cache is a 5-tuple entry in the format `AA:BB:CC:DD:EE` where,
  - `AA` indicates cache mapping (`DM`, `FA` or `SA`).
  - `BB` indicates write policy (`WT` or `WB`).
  - `CC` indicates replacement policy (`RD`, `FI`, `LR` or `LF`).
  - `DD` indicates the total size of the cache (in Bytes).
  - `EE` indicates the block size of the cache (in Bytes).
  - `FF` indicates the associativity of the cache. This field (if present) is ignored if the cache is direct mapped (`DM`) or fully-associative (`FA`).
- The stats file printed at the end of the simulation should now print the following statistics for each cache:

1. Total Number of XX Accesses: <count>
2. Total Number of XX Hits: <count>
3. Total Number of XX Misses: <count>

where XX is the cache name (`l1i`, `l1d` or `l2`).

- Consider the following example that demonstrates these specifications:

```
$ python3 src/main.py --l1i DM:WT:LR:256:32:1 \
                      --l1d DM:WT:LR:256:32:1 \
                      --l2 SA:WB:LR:4096:32:4 \
                      --stats programs/runs/1-even.stats \
                      programs/bins/1-even.r5o
```

The above command simulates a processor that has a 256-byte L1 instruction and data caches, each of which have a block size of 32 bytes and is a direct-mapped cache which follows the LRU replacement policy. It also has a 4096-byte L2 cache cache which is organized as a 4-way set-associative cache which follows a LFU policy. At the end of this run, the stats file produced will contain the additional lines shown below:

```
$ cat programs/runs/1-even.stats
python3 src/main.py --l1i DM:WT:LR:256:32:1 --l1d DM:WT:LR:256:32:1 ...

Total cycles: 2030400
Total instructions: 1000000

Total Number of l1i Accesses: 1000000
Total Number of l1i Hits: 999900
Total Number of l1i Misses: 100

Total Number of l1d Accesses: 330000
Total Number of l1d Hits: 280500
Total Number of l1d Misses: 49500

Total Number of l2 Accesses: 49600
Total Number of l2 Hits: 44640
Total Number of l2 Misses: 4960
```