CSCI 4727 – Operating Systems

Project 1 – Memory Hierarchy Simulator

Goal

Your main goal is to develop a console-based simulation for individual elements of the memory hierarchy—a page table (PT), a data cache (DC), and a data translation look-aside buffer (DTLB)—and how a data reference (virtual and/or physical) traverses the hierarchy. You will read a trace of memory references from the standard input, simulate the effects of those memory references on the individual elements specified above for this memory hierarchy, and produce statistics about the trace to the standard output.

This is a ***group project***—students completing this project should organize into groups of 3-4.

Specifications

Trace: a memory reference has the following format:

```
<accesstype>:<hexaddress>
```

`<accesstype>` is one of the two following characters:

`R` – a read access
`W` – a write access

`<hexaddress>` is the starting address of the reference (expressed as a hexadecimal number) with a reference size ranging from 8 bits to 32 bits. See `trace.dat`, `real_tr.dat`, and the example section for examples of memory references.

Before processing the trace file, the memory hierarchy simulator will read a configuration file (named `trace.config`), located in the present working directory, that specifies the configuration for the individual modules. See `trace.config` and the example section for the required format and expected `key:value` pairs.

Configuration specifics:

- The DTLB may range from direct mapped to fully associative (and any set associative in-between)
- The DTLB has a maximum of 64 entries
- The L1 DC is direct mapped or set associative—with a maximum set associativity of 8
- The L1 DC has a maximum of 128 entries
- The L1 DC has a minimum line size of 8 bytes
- The maximum number of virtual pages is 8192
- The maximum number of physical pages is 2048
- A page has a maximum size of 4 KiB
- The number of sets, line size, and entries in the caches, the number of virtual pages, and the number of physical pages ***must*** be a power of 2

In addition, implement the following simulation configuration options in your configuration file:

- The caches shall use write-allocate, write-back or no write-allocate, write-through policies for data writes: If implementing both L1 and L2—assume an ***inclusive*** caching policy
    - Install misses in both levels of the cache; evict from both levels of the cache if L2 evicted (***only*** evict from L1 if L1 eviction)
- The simulator shall disable virtual to physical address translation if the associated option is disabled
- The simulator shall disable the DTLB if the associated option (or virtual memory) is disabled

Some additional simulation specifications:

- The simulator shall use an LRU replacement algorithm for the caches and page table
- The physical pages shall be initially allocated from 0 to *n*-1, where *n* is the number of physical pages
- On page replacement, associated lines in the various caches shall be invalidated

Output

In order:

1. The simulation shall display its configuration (see example output below for expected information)
2. The simulation shall display information about each reference
3. The simulation shall display the number of hits, the number of misses, and the hit ratio for each individual element present in the hierarchy (DTLB, PT, DC, and L2)
4. The simulation shall display some additional statistics:
    a. Number of read accesses
    b. Number of write accesses
    c. Ratio of accesses that were reads
    d. Total number of memory references
    e. Total number of accesses to the PT
    f. Total number of backing store references

Use the following C `printf` format specifier (or equivalent) to display each memory reference:

```
"%08x %6x %4x %6x %3x %4s %4s %4x %6x %3x %4s"
```

Each field corresponds to the following information:

- Virtual address
- Virtual page number
- Page offset
- DTLB tag
- DTLB index
- DTLB result
- PT result
- Physical page number
- DC tag
- DC index
- DC result

Deliverables

Prepare a presentation (15 minutes) that demonstrates your simulation implementation and discusses what works, what does not work, who did what, and team development practices/processes. A retrospective of the project (what went well, what didn't, how to improve for next time) would be appropriate here. Present on the day posted on D2L.

Above and Beyond

Additional layers of (larger) cache with different read/write policies; multicore simulation (per-core caching, per-system main memory); GUI simulation; web-based implementation; distributed implementation.

Examples

trace.config

```
Data TLB configuration
Number of sets: 2
Set size: 1

Page Table configuration
Number of virtual pages: 64
Number of physical pages: 4
Page size: 256

Data Cache configuration
Number of sets: 4
Set size: 1
Line size: 16
Write through/no write allocate: n

L2 Cache configuration
Number of sets: 16
Set size: 4
Line size: 16
Write through/no write allocate: n

Virtual addresses: y
TLB: y
L2 cache: y
```

trace.dat

```
R:c84
R:81c
R:14c
R:c84
R:400
R:148
R:144
R:c80
R:008
```

Output

```
Data TLB contains 2 sets.
Each set contains 1 entries.
Number of bits used for the index is 1.

Number of virtual pages is 64.
Number of physical pages is 4.
Each page contains 256 bytes.
Number of bits used for the page table index is 6.
Number of bits used for the page offset is 8.

D-cache contains 4 sets.
Each set contains 1 entries.
Each line is 16 bytes.
The cache uses a write-allocate and write-back policy.
Number of bits used for the index is 2.
Number of bits used for the offset is 4.

L2-cache contains 16 sets.
Each set contains 4 entries.
Each line is 16 bytes.
The cache uses a write-allocate and write-back policy.
Number of bits used for the index is 4.
Number of bits used for the offset is 4.

The addresses read in are virtual addresses.


Virtual  Virt.  Page TLB    TLB TLB  PT   Phys        DC  DC          L2  L2
Address  Page # Off  Tag    Ind Res. Res. Pg # DC Tag Ind Res. L2 Tag Ind Res.
-------- ------ ---- ------ --- ---- ---- ---- ------ --- ---- ------ --- ----
00000c84      c   84      6   0 miss miss    0      2   0 miss      0   8 miss
0000081c      8   1c      4   0 miss miss    1      4   1 miss      1   1 miss
0000014c      1   4c      0   1 miss miss    2      9   0 miss      2   4 miss
00000c84      c   84      6   0 miss  hit    0      2   0 miss      0   8  hit
00000400      4    0      2   0 miss miss    3      c   0 miss      3   0 miss
00000148      1   48      0   1  hit         2      9   0 miss      2   4  hit
00000144      1   44      0   1  hit         2      9   0  hit
00000c80      c   80      6   0 miss  hit    0      2   0 miss      0   8  hit
00000008      0    8      0   0 miss miss    1      4   0 miss      1   0 miss

Simulation statistics

dtlb hits : 2
dtlb misses : 7
dtlb hit ratio : 0.222222

pt hits : 2
pt faults : 5
pt hit ratio : 0.285714

dc hits : 1
dc misses : 8
dc hit ratio : 0.111111

L2 hits : 3
L2 misses : 5
L2 hit ratio : 0.375000

Total reads : 9
Total writes : 0
Ratio of reads : 1.000000

main memory refs : 5
page table refs : 7
disk refs : 5
```