

**ECE571: Advanced Microprocessor Design – Homework 5**  
Fall 2019

**Due: Thursday 10 October 2019, 2:00pm**

Create a document that contains the data and answers described in the sections below. A .pdf or .txt file is preferred but I can accept MS Office or Libreoffice format if necessary.

**1. Cache parameters**

A Haswell machine has a 44-bit physical address space, 32-kB L1 data cache, 8-way set associative, 64-bytes per line.

- (a) How many bits are used to calculate the offset?
- (b) How many bits are used to calculate the line?
- (c) How many bits are used for the tag?

**2. Cache problem**

This question assumes a 512-byte cache, 16-bytes per line, 2-way associative, 32-bit address size. (24 bits of tag, 4 bits for line, 4 bits for offset). The cache's current contents are as follows:

line	Way 0				Way 1			
	V	D	LRU	Tag	V	D	LRU	Tag
0	1	0	1	0000 00	1	0	0	0000 08
1	1	0	0	0000 00	1	1	1	0000 0a
2	0				0			
3	0				0			
4	0				0			
5	0				0			
...								
b	0				0			
c	0				0			
d	0				0			
e	0				0			
f	0				0			

For each of the following sequence of memory accesses state whether it is a cache hit or miss. If a line is evicted due to a miss, state whether the evicted data need to be written back to memory or not.

- (a) `ldrb r0, 0x0000080f`
- (b) `ldrb r0, 0xffffffff`
- (c) `strb r0, 0x00000810`
- (d) `strb r0, 0xffffffff`

### 3. Bzip2 cache behavior on the x86 Haswell-EP Machine

For this section, log into the Haswell-EP machine just like in previous homeworks. Run the bzip2 benchmark, recall you will use a command line something like this:

```
perf stat -e instructions:u,L1-icache-load-misses:u \  
/opt/ece571/401.bzip2/bzip2 -k -f ./input.source
```

- (a) Measure and report the L1 instruction cache miss rate.  
Use `instructions:u` and `L1-icache-load-misses:u` for the events.
- (b) Measure and report the L1 data cache load miss rate.  
Use `L1-dcache-loads:u` and `L1-dcache-load-misses:u`
- (c) Measure and report the L2 cache miss rate  
Use `l2_rqsts.references:u` and `l2_lines_in.all`
- (d) Measure and report the L3 cache miss rate  
Use `cache-references:u` and `cache-misses:u`

### 4. equake\_1 cache behavior on the x86 Haswell-EP Machine

Recall that running equake looks something like this:

```
perf stat -e instructions:u,L1-icache-load-misses:u \  
/opt/ece571/equake_1.speccomp/equake_1 < \  
/opt/ece571/equake_1.speccomp/inp.in
```

- (a) Measure and report the L1 instruction cache miss rate.  
Use `instructions:u` and `L1-icache-load-misses:u` for the events.
- (b) Measure and report the L1 data cache load miss rate.  
Use `L1-dcache-loads:u` and `L1-dcache-load-misses:u`
- (c) Measure and report the L2 cache miss rate  
Use `l2_rqsts.references:u` and `l2_lines_in.all`
- (d) Measure and report the L3 cache miss rate  
Use `cache-references:u` and `cache-misses:u`

### 5. Bzip2 cache behavior on the Jetson

Now run the bzip2 benchmark on the ARM64 jetson machine. (As with the last homework, just ssh from the haswell machine, `ssh jetson`).

You will note on the jetson machine that `perf list` shows a lot of events missing. For some reason the linux-kernel is missing support for them, but if you look at the ARM Cortex-A57 manual:

[http://web.eece.maine.edu/~vweaver/classes/ece571\\_2016s/DDI0488C\\_cortex\\_a57\\_mpcore\\_r1p0\\_trm.pdf](http://web.eece.maine.edu/~vweaver/classes/ece571_2016s/DDI0488C_cortex_a57_mpcore_r1p0_trm.pdf)  
in Chapter 11.8 you will see support is listed for them.

The perf built-in events for Cortex-A57 were not really updated until Linux 4.4 but we are running 3.10 on our system.

- (a) For icache rate try measuring `r14:u` (which is `L1_ICACHE`) and `r01:u` (which is `L1I_CACHE_REFILL`).
- (b) Measure and report the L1 data cache load miss rate.  
Use `r40:u` (which is `L1D_CACHE_LD`) and `r42:u` (which is `L1D_CACHE_REFILL_LD`).

- (c) Measure and report the L1 data cache store miss rate.  
Use `r41:u` (which is `L1D_CACHE_ST`) and `r43:u` (which is `L1D_CACHE_REFILL_ST`).
- (d) Measure and report the L2 data cache load miss rate.  
Use `r50:u` (which is `L2D_CACHE_LD`) and `r52:u` (which is `L2D_CACHE_REFILL_LD`).
- (e) Measure and report the L2 data cache store miss rate.  
Use `r51:u` (which is `L2D_CACHE_ST`) and `r53:u` (which is `L2D_CACHE_REFILL_ST`).

## 6. Short Answer Questions

To answer these questions, it might be useful to know the cache size parameters of the various machines. See if you can find the L1/L2/L3 cache sizes for the Haswell-EP (Xeon E5-2640 v3) and the NVidia Jetson TX-1 (ARM Cortex-A57).

One additional piece of information, you can use "top" while a program is running to see the memory working set size. For `bzip2` this is around 11MB, where `equake` is using 700MB.

- (a) How does `equake`'s cache behavior differ from `bzip2`'s on Haswell? What might be the reason for this?
- (b) How does `bzip2`'s cache behavior on Haswell differ from `bzip2`'s cache behavior on Jetson? What might be the reason for this?

## 7. Submitting your work.

- Create the document containing the data as well as answers to the questions asked.
- Please make sure your name appears in the document.
- e-mail the file to me by the homework deadline.