CSCE 513 Computer Architecture **Test 1** **October 2, 2013**

Name:_____ Email:_____

Instructions

- **No Calculators!!**
- Your notes on one side of an 8.5 x 11 sheet of paper should be signed and submitted with your test.
- Make sure your exam is complete.
- No Calculators, cell phones, or other electronic devices.
- All questions are equally weighted.
- Answer in the space provided if at all possible.
- If a question is unclear please ask early in the test.
- There is a Take Home question. It will be emailed today.
- Good Luck!

1. (a) What is Mem/WB.LMD ?

   (b) Why does it make sense to give reads priority over Writes? What is necessary to allow this? That is what is added to caches to allow this?

   (c) What is Moore's law?

   (d) Explain critical word first and early restart.

   (e) What is meant by way-prediction?

   (f) What is meant by statically scheduling code?

2. Classical 5 stage pipeline: Assuming the classical 5-stage pipeline with no forwarding except through the registers. Assume all of these instructions are integer and execute in 1 cycle. Given the code below:

```
loop1: LD      F6, 0(R1)
loop2: LD       F8, 0(R2)
       ADD     F6, F6, F8
      DADDIU  R2, R2, +8
      BNE     R2, R6, loop2
      DADDIU  R1, R1, +8
      BNE     R1, R4, loop1
```

   (a) (4 pts)Show how the first two iterations of the inner loop would proceed through the pipeline. Stop with the fetch of the first instruction of the inner loop on the third iteration or when you fill the table. Assume forwarding only through the registers and that you predict branch not taken. Also, state any assumptions you make about when the branch target address is computed.

Instruction		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

   (b) (2 points) How many cycles does the two iterations take.

   (c) (3 points) Identify a place where forwarding could help and explain in detail how the forwarding condition would be recognized and which pipeline register/field the data would be forwarded from.

3. (a) In the original pipeline approach to branches how was the branch target address calulated.

(b) Later we improved on this by adding what and to what stage?

(c) Draw the diagram of a 3-bit saturating counter branch predictor.

(d) If a loop executes 1000 times, has only the branch at the bottom of the loop and starts out in the most strongly not taken state how many branches are mispredicted?

4. Given an application that runs in 1 minute and executes 1G instructions.

   (a) If the applicaton is 80% parallelizable (both time and instructions) and you have 10 processors how fast will the application run?

   (b) If the CPI of the sequential portion of the program equals the CPI of the parallelizable part and we scaled the problem without changing the serial fraction. This means we worked a larger problem and took the same amount of time (1 minute) with the 100 processors. What number of instructions that could be executed with 100 processors in that 1 minute?

5. AMAT: In the system we are analyzing the memory has:

   - Separate L1 instruction and data caches, HitTime = Processor Cycle Time
   - 32KB L1 instruction cache with 1% miss rate, 64B blocks
   - 256KB L1 data cache with 10% miss rate, 16B blocks
   - 256K L2 unified cache with 64B blocks, local miss rate 20%, Hit Time = 6 cycles,
   - Main Memory Access time is 50 cycles for the first 64 bits and subsequent 64 bit chunks are available every 4 cycles.
   - Both L1 caches are direct mapped, L2 four-way associative.
   - Assume there are no misses to main memory.

   (a) Just for this subproblem assume the L1-data miss penalty is 14 cycles and ignore the information about Instruction cache, L2 and main memory. In this case what is the AMAT for data references?

   (b) What is the Miss Penalty for accesses to L2?

   (c) What is the average memory access time for instruction references?

   (d) Assume the only memory reference instructions are loads(30%) and stores(5%). What percentage of total memory references are data references?

   (e) What is the Average memory access time? Use the AMAT-data from part a.

6. Virtual Memory/TLB - Given

   - 40 bit virtual addresses
   - the page size is 4KB,
   - 32 bit physical addresses
   - 128KB cache, only one level unified, 8 way associativity, 256B lines
   - 128 entry TLB, 4 way associative
   - If the virtual address is 0x0F F123 4567 and if the physical page number is 0x322 with leading zeroes not shown

   (a) What is the page offset field?

   (b) What is the VPN?

   (c) How big is a PPN?

   (d) What is the physical address?

   (e) What is the the cache "block offset" field?

   (f) What is "set-index" field?

   (g) What is the cache "tag" field?

   (h) What happens on a TLB miss?

7. Given the code below and a direct mapped cache with 256 lines of 32B bytes.

```
float a[4096];
double sum = 0.0;
for(i=0; i < 4096; ++i)
      sum = sum + a[i];
```

Assume that the non-array variables are stored in registers and ignore instruction references.

(a) How many array elements fit in a block?

(b) What is the hit ratio?

(c) What is the hit ratio is we assume 128B blocks?

8. Forwarding

   (a) In full forwarding data is forwarded from a number of locations. What are they?

   (b) What kind of circuit does EX/MEM.IR[rd] == ID/EX.IR[rs]?

   (c) What other conditions in addition to the EX/MEM.IR[rd] == ID/EX.IR[rs] indicate that we need to do some forwarding?

   (d) Give an example of a code that would make this type of forwarding occur

9. Extra credit: What IEEE 754 float does 0 1000 0000 000 0000 0000 ... 0000 represent?