

CSEE W3827
Fundamentals of Computer Systems
Homework Assignment 6

Prof. Martha A. Kim
Columbia University

Due December 12, 2013 at **12 noon, in CSB 469.**

Write your name **and UNI** on your solutions

Show your work for each problem; we are more interested in how you get the answer than whether you get the right answer.

1. (20 pts.) Consider the execution of the program below on a fully bypassed (i.e., both W-E and M-E operand forwarding), 5-stage MIPS pipeline with early branch resolution (i.e., branches resolved in D stage).

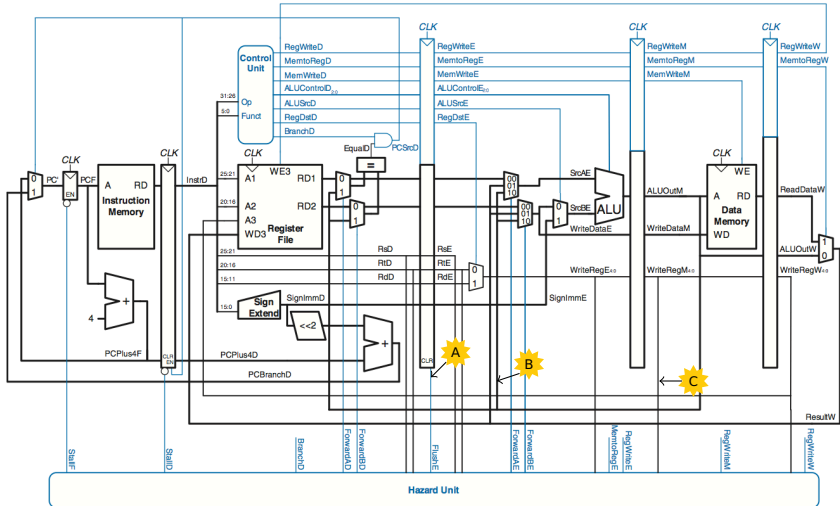
```
i1:    add    $v0, $0, $0
i2:    addi   $t0, $0, 100
i3:    lw     $t1, 0($a0)
i4:    add    $v0, $v0, $t1
i5:    addi   $a0, $a0, 4
i6:    addi   $t0, $t0, -1
i7:    beq   $t0, $0, end
i8:    beq   $0, $0, i3
end:
```

(a) Complete the list of operands that are forward when the instructions below are executed, e.g.,

- \$t1 i3 \rightarrow i4

(b) Reorder the instructions so that *no stalls are required*

2. (20 pts.) Give an example instruction sequence that would *not function* if each of the three indicated wires were cut. (The wire cuts are not cumulative, but individual.)



3. (20 pts.) Below is the program from the previous homework. Compute the CPI of this program on the *pipelined* processors listed. You may ignore the cycles required to fill/drain the pipe.

```
    addi  $s0, $0, 0           # i = 0
    add   $s1, $0, $0         # sum = 0
    addi  $t0, $0, 10        # $t0 = 10
loop:
    slt   $t1, $s0, $t0      # if (i < 10), $t1 == 1, else $t1 = 0
    beq   $t1, $0, done      # if (i >= 10), branch to done
    add   $s1, $s1, $s0      # sum += i
    addi  $s0, $s0, 1        # i++
    beq   $0, $0, loop
done:
```

(a) Early branch resolution, but no forwarding, can only stall to resolve hazards.

(b) Early branch resolution, fully bypassed.

4. (20 pts.) Consider a 1024B L1 direct-mapped cache with 8B lines.

(a) For the following ten references, indicate whether they will hit or miss in the cache (assuming it starts empty).

- 0x0000
- 0x0008
- 0x0000
- 0x0010
- 0x0002
- 0x0012
- 0x0080
- 0x0000
- 0x0800
- 0x0880

(b) Assuming this cache has an access time of 5 cycles and is backed by memory with 100 cycle access time, what miss rate is required so that the overall access time of the L1-Memory hierarchy is less than 10 cycles?

5. (20 pts.) For each of the four caches listed below, break the 32-bit addresses into *tag*, *set index*, and *byte offset* fields.

(a) 1024B capacity, direct mapped, 256B lines

00

(b) 4096B capacity, direct mapped, 64B lines

00

(c) 1024B capacity, 4-way, 128B lines

00