CSEE W3827

Fundamentals of Computer Systems

Homework Assignment 6

Solutions

Prof. Martha A. Kim and Stephen A. Edwards

Columbia University

Due April 30th, 2012 at 1:10 PM

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

1. (25 pts.) Examine the following MIPs program:

```
 i1:   ori  $t0, $0,  1000
 i2:   ori  $t1, $0,  2000
 i3:   addi $t2, $t0, 100
 i4:   lw   $t3, 0($t1)
 i5:   lw   $t4, 0($t0)
 i6:   add  $t3, $t3, $t4
 i7:   sra  $t3, $t3, 1
 i8:   sw   $t3, 0($t0)
 i9:   sw   $t3, 0($t1)
i10:   addi $t0, $t0, 4
i11:   addi $t1, $t1, 4
i12:   slt  $t3, $t0, $t2
i13:   bne  $0,  $t3, i4
```

(a) When this code fragment is run, how many instructions will be executed total?

$3 + 10 \times 25 = 253$ instructions

(b) Consider the five-stage MIPS pipeline with full bypass (i.e., M-to-E, W-to-E, and M-to-D bypasses) and stall logic discussed as in class and shown at the end of this assignment. List each hazard in the code and explain how the processor will resolve it, e.g., “stall two cycles,” “bypass W to E,” “bypass M to D.” The first is provided as an example. [1]

- $i1 \longrightarrow i3$, bypass W to E
- $i2 \longrightarrow i4$, bypass W to E
- $i5 \longrightarrow i6$, stall $i6$ one cycle then bypass W to E
- $i6 \longrightarrow i7$, bypass M to E
- $i7 \longrightarrow i8$, bypass M to E
- $i7 \longrightarrow i9$, bypass W to E
- $i10 \longrightarrow i12$, bypass W to E
- $i12 \longrightarrow i13$, stall $i13$ one cycle then bypass M to D
- $i13 \longrightarrow i4$, when branch taken, misprediction results in one cycle bubble

[1] For the handful of instructions not supported by the processor assume that: shift and setting operations for `sra` and `slt` are performed by the ALU in the Execute stage; that bne is just like beq and with branch resolution in the Decode stage.

(c) How many cycles will this code snippet take to execute on a piplined processor?

$3 + (10 + 3) \times 24 + (10 + 2) + 4$
$3 + 312 + 12 + 4 = 331$ cycles

2. (20 pts.) Assuming the same, fully-bypassed pipeline as in the previous problem.

(a) List all control and data dependencies which result in pipeline stalls (e.g., $i2 \rightarrow i3$). $i2 \rightarrow i3; i6 \rightarrow i7; i7 \rightarrow i1$

(b) Reorder the instructions to remove as many of the stalls as possible. You may not change the code other than reordering (i.e., no re-allocating registers). List the stalls that remain. only the misprediction stall remains: $i7 \rightarrow i1$

```
i1:     lw   $t0, 0($t1)
i2:     lw   $t4, 0($t2)
i3:     add  $t0, $t0, $t4
i4:     add  $t0, $t0, $t2
i5:     addi $t2, $t2, -8
i6:     addi $t1, $t1, -8
i7:     bne  $t1, $0, i1
```

```
i1:     lw   $t0, 0($t1)
i2:     lw   $t4, 0($t2)
i6:     addi $t1, $t1, -8
i3:     add  $t0, $t0, $t4
i4:     add  $t0, $t0, $t2
i5:     addi $t2, $t2, -8
i7:     bne  $t1, $0, i1
```

3. (25 pts.) Consider three **direct mapped** caches X, Y, and Z each interpreting an 8-bit address slightly differently according to the `{tag:setIdx:byteOffset}` format specified. For each address in the reference stream, indicate whether the access will hit (H) or miss (M) in each cache.

Address Formats →	**C1** **{2:2:4}**	**C2** **{2:3:3}**	**C3** **{2:4:2}**
`Block Size (Bytes):`	16	8	4
`Cache Size (Blocks):`	4	8	16

Address References
(in binary)

`00000010`	M	M	M
`00000100`	H	H	M
`00001000`	H	M	M
`00010000`	M	M	M
`00100000`	M	M	M
`01000000`	M	M	M

4. (30 pts.) Consider a three-level hierarchy consisting of an L1 cache backed by an L2 backed by main memory.

(a) For each address referenced, indicate whether it will be a hit (H) or a miss (M) in each cache. You should indicate no access with a slash. For each hit or miss, also indicate the set in which it occurred. For example, $M \rightarrow A$, means the access missed in set A.

	L1	**L2**
Sets	16	16
Bytes/Block	16	256
Ways	2	2

0xABCDE	$M \rightarrow D$	$M \rightarrow C$
0xABCD0	$H \rightarrow D$	–
0xABDD0	$M \rightarrow D$	$M \rightarrow D$
0xABCE0	$M \rightarrow E$	$H \rightarrow C$
0xABCEF	$H \rightarrow E$	–
0xABCDE	$H \rightarrow D$	–

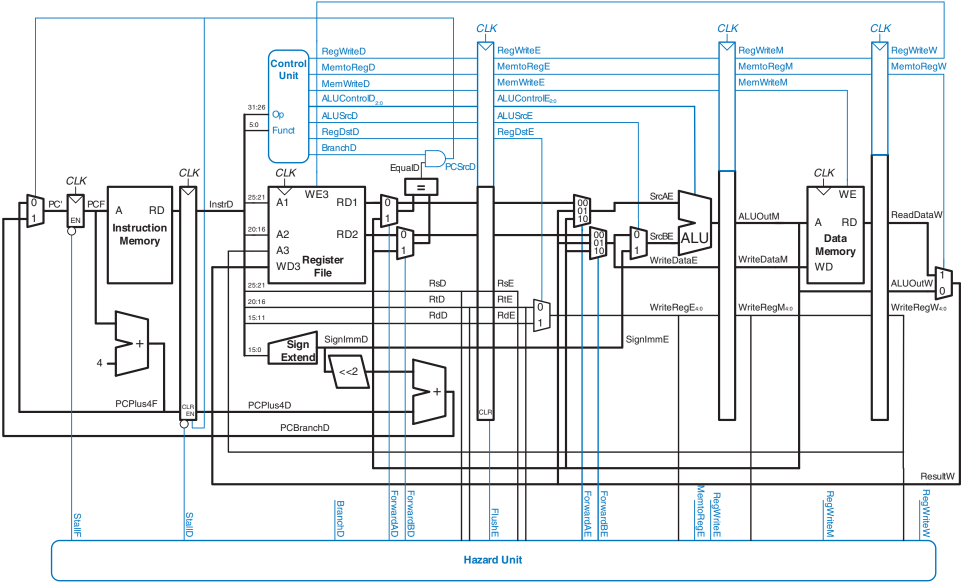
(b) What is the miss rate for L1 and L2?

$M_{L1} = 50\%; M_{L2} = 66.6\%$

(c) Let the L1 access time be 5 cycles, L2 50 cycles, and Memory (which backs L2) 1000 cycles. What is the expected access time for the entire hierarchy, assuming the reference pattern from the first part of the problem?

$T = T_{L1} + M_{L1} \times (T_{L2} + M_{L2} \times T_{Mem}) = 5 + 0.5 \times (50 + 0.666 \times 1000) = 363$

For reference, fully pipelined MIPS datapath:


CLK
Control Unit
Op
Funct
31:26
5:0
RegWriteD
MemtoRegD
MemWriteD
ALUControlD2:0
ALUSrcD
RegDstD
BranchD
RegWriteE
MemtoRegE
MemWriteE
ALUControlE2:0
ALUSrcE
RegDstE
RegWriteM
MemtoRegM
MemWriteM
RegWriteW
MemtoRegW
EqualD
PCSrcD
=
PC'
PCF
EN
A
RD
Instruction Memory
InstrD
25:21
20:16
A1
A2
A3
WD3
WE3
RD1
RD2
Register File
SrcAE
SrcBE
ALU
WriteDataE
ALUOutM
WriteDataM
WE
Data Memory
WD
ReadDataW
ALUOutW
WriteRegE4:0
WriteRegM4:0
WriteRegW4:0
ResultW
RsD
RtD
RdD
RsE
RtE
RdE
15:11
15:0
Sign Extend
SignImmD
SignImmE
<<2
+
4
PCPlus4F
PCPlus4D
PCBranchD
CLR
00
01
10
0
1
StallF
StallD
BranchD
ForwardAD
ForwardBD
FlushE
ForwardAE
ForwardBE
MemtoRegE
RegWriteE
RegWriteM
RegWriteW
Hazard Unit