Computer Sciences Department
University of Wisconsin-Madison
CS/ECE 552 – Introduction to Computer Architecture
In-Class Exercise (04/14)

Answers to all questions should be uploaded on Canvas.

1. [1 point] From the reading (Challenge): Why does the MIPS R10000 have problems with synonyms?

Stall buffer

2. [9 points] Consider the following MIPS assembly program running on the single-issue R10K-like processor discussed in class (F, De, Di, S, X, C, R):

```
lw   $t0, 0($s2)
and  $s2, $t2, $t1
or   $s1, $s1, $t2
sub  $t2, $s0, $s2
lw   $t0, 4($t0)
lw   $s2, 0($s1)
sub  $t0, $t1, $s1
or   $s1, $t2, $t0
lw   $t2, 0($t0)
add  $t1, $t2, $s1
```

Assume that initially: the front-end is single issue (i.e., the instruction buffer and dispatch buffer are size 1), the reservation stations and reorder buffer are empty, and the contents of the map table and free list are as follows:

Map Table		
Architectural Reg	**Physical Reg**	**Ready (Y/N)**
$t0	P1	Y
$t1	P2	Y
$t2	P3	Y
$s0	P4	Y
$s1	P5	Y
$s2	P6	Y

Free List	P7, P8, P9, P10, P11, P12

Assume that memory operations (i.e., loads and stores) take 5 cycles to execute (i.e., 5-cycle X stage), while all other instructions take 1 cycle to execute. If the first instruction (i.e., lw $t0, 0($s2)) is in the F stage on cycle 1, fill in the contents below of the reservation stations, map table, free list and reorder buffer **at the beginning of cycle 9** (i.e., immediately after the clock edge between cycles 8 and 9). Pop registers from the left of the free list and push them onto the right of the free list. In the reorder buffer, list instructions from oldest (top) to youngest (bottom).

Instr \ cycle	1	2	3	4	5	6	7	8	9
`lw $t0, 0($s2)`	F	De	Di	S	X	X	X	X	C
`and $s2, $t2, $t1`		F	De	Di	S	X	C		
`or $s1, $s1, $t2`			F	De	Di	S	X	C	
`sub $t2, $s0, $s2`				F	De	Di	S	X	X
`lw $t0, 4($t0)`					F	De	Di	De	S
`lw $s2, 0($s1)`						F	De	Di	De
`sub $t0, $t1, $s1`							F	De	Di
`or $s1, $t2, $t0`								F	De
`lw $t2, 0($t0)`									F
`add $t1, $t2, $s1`									

Reservation Stations and Functional Units					
Functional Unit	**Instruction**	**Operand (rs)**		**Operand (rt)**	
		Physical Reg	**Ready (Y/N)**	**Physical Reg**	**Ready (Y/N)**
add/sub	**Sub P10, P4, P8**	**P4**	**y**	**P8**	**y**
and	**And P8, P3, P2**	**P3**	**y**	**P2**	**y**
or	**Or P9 p5 p3**	**P5**	**y**	**P3**	**y**
lw	**Lw p11 4(p7)**	**P11**	**n**	**P7**	**n**
sw					

Map Table		
Architectural Reg	**Physical Reg**	**Ready (Y/N)**
$t0	**P11**	**n**
$t1	**P2**	**y**
$t2	**P10**	**n**
$s0	**P4**	**y**
$s1	**P9**	**n**
$s2	**P12**	**y**

Free List	**P1**

Reorder Buffer			
Instruction		Dest Physical Reg (rd)	Old Physical Reg (rd)
Head (Oldest)	Lw p7 0(p6)	P7	P1
	And p8 p3 p2	P8	P6
	Or p9 p5 p3	P9	P5
	Sub p0 p4 p8	P8	P10
Tail (Youngest)			