Solutions for Homework 4

CIS 501, Spring 1998
February 18, 1998

1. A machine is called "underpipelined" if additional levels of pipelining can be added without changing the pipeline-stall behaviour appreciably. Suppose that the DLX integer pipeline was changed to four stages by merging EX and MEM and lengthening the clock cycle by 50%. How much faster would the conventional DLX pipeline be versus the underpipelined DLX on integer code only? Assume that the number of stalls goes down from 9% to 5%.

$$\frac{Average\ instruction\ time\ underpipelined}{Average\ instruction\ time\ conventionally\ pipelined}$$

$$= \frac{CPI\ underpipelined \times Clock\ cycle\ underpipelined}{CPI\ pipelined \times Clock\ cycle\ pipelined}$$

$$= \frac{(1+0.05)\times 1.5}{(1+0.09)\times 1} = 1.44$$

2. Consider the following code fragment:

```
LW      R3, 0(R2)
ADD     R1, R2, R3
ADD     R4, R1, R1
SUB     R2, R3, R2
```

Assume a simple, pipelined DLX. Assume that each stage of the pipeline takes 1 clock cycle. Write out a diagram similar to that of Figure 3.14 showing the execution of these instructions for each of the following cases:

(a) Assume that any data hazads cause a stall in the pipeline and that there are no mecahnisms for forwrding results. Do assume that the register set is written in the first half of a cycle and read in the second half.

Let F, D, X, M, and W stand for the IF, ID, EX, MEM, and WB stages, respectively. Let - denote a stall.

Instruction	Clock Cycle											
	1	2	3	4	5	6	7	8	9	10	11	12
`LW R3,0(R2)`	F	D	X	M	W							
`ADD R1,R2,R3`		F	-	-	D	X	M	W				
`ADD R4,R1,R1`					F	-	-	D	X	M	W	
`SUB R2,R3,R2`								F	D	X	M	W

Total clock cycles 12.

(b) Assume now that forwarding of results is available.

Instruction	Clock Cycle								
	1	2	3	4	5	6	7	8	9
LW R3,0(R2)	F	D	X	M	W				
ADD R1,R2,R3		F	D	-	X	M	W		
ADD R4,R1,R1			F	-	D	X	M	W	
SUB R2,R3,R2					F	D	X	M	W

Total clock cycles 9.

(c) Assume that the memory path is shared by both instruction fetches as well as data accesses so that there is a structural hazard. Further assume that the forwarding of results is available.

Instruction	Clock Cycle								
	1	2	3	4	5	6	7	8	9
LW R3,0(R2)	F	D	X	M	W				
ADD R1,R2,R3		F	D	-	X	M	W		
ADD R4,R1,R1			F	-	D	X	M	W	
SUB R2,R3,R2					F	D	X	M	W

Total clock cycles 9.

3. Take the following code:

```
loop:   LW      R1, 0(R2)
        ADDI    R1, R1, \#1
        SW      R1, 0(R2)
        ADDI    R2, R2, \#4
        SUB     R4, R3, R2
        BNZ     R4,loop
```

Use the DLX pipeline. Assume that the branch causes the pipeline to be flushed, and that the register set is written in the first half of a cycle and read in the second.

The initial value of R3 is R2 + 396

(a) Show the timing of this instruction sequence for two loop cycle without any forwarding.

First iteration:

Instruction	Clock Cycle																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
LW R1,0(R2)	F	D	X	M	W													
ADDI R1,R1,#1		F	-	-	D	X	M	W										
SW R1,0(R2)					F	-	-	D	X	M	W							
ADDI R2,R2,#4								F	D	X	M	W						
SUB R4,R3,r2									F	-	-	D	X	M	W			
BNEZ R4,loop												F	-	-	D	X	M	W

Second iteration:

Instruction		Clock Cycle																	
		18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
`LW`	`R1,0(R2)`	F	D	X	M	W													
`ADDI`	`R1,R1,#1`		F	-	-	D	X	M	W										
`SW`	`R1,0(R2)`					F	-	-	D	X	M	W							
`ADDI`	`R2,R2,#4`								F	D	X	M	W						
`SUB`	`R4,R3,r2`									F	-	-	D	X	M	W			
`BNEZ`	`R4,loop`												F	-	-	D	X	M	W

There are 17 cycles in each iteration. And there are $17 \times \frac{396}{4} + 1 = 17 \times 99 + 1 = 1684$ cycles in the entire loop.

(b) Now assume that forwarding is available.

First iteration:

Instruction		Clock Cycle										
		1	2	3	4	5	6	7	8	9	10	11
`LW`	`R1,0(R2)`	F	D	X	M	W						
`ADDI`	`R1,R1,#1`		F	D	-	X	M	W				
`SW`	`R1,0(R2)`			F	-	D	X	M	W			
`ADDI`	`R2,R2,#4`					F	D	X	M	W		
`SUB`	`R4,R3,r2`						F	D	X	M	W	
`BNEZ`	`R4,loop`							F	D	X	M	W

Second iteration:

Instruction		Clock Cycle										
		11	12	13	14	15	16	17	18	19	20	21
`LW`	`R1,0(R2)`	F	D	X	M	W						
`ADDI`	`R1,R1,#1`		F	D	-	X	M	W				
`SW`	`R1,0(R2)`			F	-	D	X	M	W			
`ADDI`	`R2,R2,#4`					F	D	X	M	W		
`SUB`	`R4,R3,r2`						F	D	X	M	W	
`BNEZ`	`R4,loop`							F	D	X	M	W

There are 10 cycles in each iteration. And there are $10 \times 99 + 1 = 991$ cycles in the entire loop.