Homework 7

Benny Chen

November 29, 2022

Problem 1

Cycle 5

IF		ID		EX		MEM		WB	
	add		or		and		sub		lw
Signal	Value	Signal	Value	Signal	Value	Signal	Value	Signal	Value
		RegWrite	1	RegWrite	1	RegWrite	1	RegWrite	1
		MemtoReg	0	MemtoReg	0	MemtoReg	0	MemtoReg	1
		Branch	0	Branch	0	Branch	0		
		MemRead	0	MemRead	0	MemRead	0		
		MemWrite	0	MemWrite	0	MemWrite	0		
		ALUop	0b10	ALUop	0b10				
		ALUSrc	0	ALUSrc	0				
PCSrc	0					PCSrc	0		
PC	0x00400040	PC	0x0040003c	PC	0x0040038				
Instruction	0x00940733	Instruction	0x007366b3	BranchTarget	X	BranchTarget	X		
		rs1	6	Zero	0	Zero	0		
		rs2	7	ALU Result	0x00000380	ALU Result	0x6fff6f7c	ALU Result	0x7fffefa4
		Read data 1	0x00000066	Read data 1	0x10007f80				
		Read data 2	0x00000077	Read data 2	0x000003ff	Read data 2	0x10008000		
		Imm	X	Imm	X				
		Instr[30,14-12]	0b0110	Instr[30,14-12]	0b0111				
		rd:Instr[11-7]	13	rd	12	rd	11	rd	10
						Read Data	X	Read Data	0x7fffef7c
		WriteRegister	10						
		WriteData	0x7fffef7c					WriteData	0x7fffef7c
		MEM_WB.RegWrite	1						

Cycle 6

IF		ID		EX		MEM		WB	
	beq		add		or		and		sub
Signal	Value	Signal	Value	Signal	Value	Signal	Value	Signal	Value
		RegWrite	1	RegWrite	1	RegWrite	1	RegWrite	1
		MemtoReg	0	MemtoReg	0	MemtoReg	0	MemtoReg	0
		Branch	0	Branch	0	Branch	0		
		MemRead	0	MemRead	0	MemRead	0		
		MemWrite	0	MemWrite	0	MemWrite	0		
		ALUop	0b10	ALUop	0b10				
		ALUSrc	0	ALUSrc	0				
PCSrc	0					PCSrc	0		
PC	0x00400044	PC	0x00400040	PC	0x0040003c				
Instruction	0x00008863	Instruction	0x00940733	BranchTarget	X	BranchTarget	X		
		rs1	8	Zero	0	Zero	0		
		rs2	9	ALU Result	0x00000077	ALU Result	0x00000380	ALU Result	0x6fff6f7c
		Read data 1	0x00000088	Read data 1	0x00000066				
		Read data 2	0x00000099	Read data 2	0x00000077	Read data 2	0x000003ff		
		Imm	X	Imm	X				
		Instr[30,14-12]	0b0000	Instr[30,14-12]	0b0110				
		rd:Instr[11-7]	14	rd	13	rd	12	rd	11
						Read Data	X	Read Data	X
		WriteRegister	11						
		WriteData	0x6fff6f7c					WriteData	0x6fff6f7c
		MEM_WB.RegWrite	1						

Problem 2

	Instructions	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Example	ADD x1, x2, x3	IF	ID	EX	MEM	WB										
	SUB x1, x1, x2		IF	ID	-	-	EX	MEM	WB							
a)	ADD x1, x2, x3	IF	ID	EX	MEM	WB										
	SUB x2, x1, x4		IF	ID	-	-	EX	MEM	WB							
	SW x1, 0(x2)			IF	-	-	ID	-	-	EX	MEM	WB				
b)	ADD x1, x2, x3	IF	ID	EX	MEM	WB										
	LW x2, 0(x1)		IF	ID	-	-	EX	MEM	WB							
	LW x3, 0(x1)			IF	-	-	ID	EX	MEM	WB						
	SUB x4, x2, x3						IF	ID	-	-	EX	MEM	WB			
	SW x4, 0(x3)							IF	-	-	ID	-	-	EX	MEM	WB
c)	LW x1, 0(x10)	IF	ID	EX	MEM	WB										
	SW x1, 8(x10)		IF	ID	-	-	EX	MEM	WB							
	LW x2, 0(x11)			IF	-	-	ID	EX	MEM	WB						
	ADD x3, x1, x2						IF	ID	-	-	EX	MEM	WB			
	SUB x4, x2, x2							IF	-	-	ID	EX	MEM	WB		

Problem 3

	Instructions	**Forward rs1**	**Forward rs2**	1	2	3	4	5	6	7	8	9	10
Example	ADD x1, x2, x3			IF	ID	EX	MEM	WB					
	SUB x1, x1, x2	EX/MEM->EX			IF	ID	EX	MEM	WB				
a)	ADD x1, x2, x3			IF	ID	EX	MEM	WB					
	SUB x2, x1, x4	EX/MEM->EX			IF	ID	EX	MEM	WB				
	SW x1, 0(x2)	EX/MEM->EX	MEM/WB->EX			IF	ID	EX	MEM	WB			
b)	ADD x1, x2, x3			IF	ID	EX	MEM	WB					
	LW x2, 0(x1)	EX/MEM->EX			IF	ID	EX	MEM	WB				
	LW x3, 0(x1)	MEM/WB->EX				IF	ID	EX	MEM	WB			
	SUB x4, x2, x3		MEM/WB->EX				IF	ID	-	EX	MEM	WB	
	SW x4, 0(x3)		EX/MEM->EX					IF	-	ID	EX	MEM	WB
c)	LW x1, 0(x10)			IF	ID	EX	MEM	WB					
	SW x1, 8(x10)		MEM/WB->MEM		IF	ID	EX	MEM	WB				
	LW x2, 0(x11)					IF	ID	EX	MEM	WB			
	ADD x3, x1, x2		MEM/WB->EX				IF	ID	-	EX	MEM	WB	
	SUB x4, x2, x2							IF	-	ID	EX	MEM	WB

Problem 4

a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
I1: lw t0, 0(s0)	IF	ID	EX	MEM	WB														
I2: lw t1, 0(s1)		IF	ID	EX	MEM	WB													
I3: add t0, t0, t1			IF	ID	-	-	EX	MEM	WB										
I4: sw t0, 0(s0)				IF	-	-	ID	-	-	EX	MEM	WB							
I5: addi s0, s0, 4							IF	-	-	ID	EX	MEM	WB						
I6: addi s1, s1, 4										IF	ID	EX	MEM	WB					
I7: addi s2, s2, 1											IF	ID	EX	MEM	WB				
I8: bne s2, s3, I1												IF	ID	-	-	EX	MEM	WB	
I9:													IF	-	-	ID	EX	X	
I1: lw t0, 0(s0)																		IF	

b)

There are 17 cycles per iteration. From that, there are 6 stall cycles due to data hazards and 3 stall cycles due to control hazards.

c)

We can schedule the instructions by moving Instructions 6 and 7 to the position after Instruction 2. We would do this as Instruction 3 would be stalling for 2 cycles so instead of stalling we can put Instructions 6 and 7 to be executed first so after it is done we can execute Instruction 3 immediately. This would also remove the data hazard in Instruction 8 leading to only 3 stall cycles due to control hazards and 2 stall cycles due to data hazards. In total this would lead to 13 cycles per iteration.

d)

The speedup would be $17/13 = 1.31$.

Problem 5

a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
I1: lw t0, 0(s0)	IF	ID	EX	MEM	WB												
I2: lw t1, 0(s1)		IF	ID	EX	MEM	WB											
I3: add t0, t0, t1			IF	ID	-	EX	MEM	WB									
I4: sw t0, 0(s0)				IF	-	ID	EX	MEM	WB								
I5: addi s0, s0, 4						IF	ID	EX	MEM	WB							
I6: addi s1, s1, 4							IF	ID	EX	MEM	WB						
I7: addi s2, s2, 1								IF	ID	EX	MEM	WB					
I8: bne s2, s3, I1									IF	ID	EX	MEM	WB				
I9:										IF	ID	EX	X				
I1: lw t0, 0(s0)													IF	ID	EX	MEM	WB

b)

There are 12 cycles per iteration of the loop. There are a total of 1 stall cycle due to data hazard and 3 stall cycles due to control hazard.

c)

We can schedule the instructions so there is no hazard for Instruction 3. We can do this by putting in Instruction 7 after Instruction 2. By putting in Instruction 7 to run in the meantime, we can avoid the data hazard for Instruction 3. In total there are 11 cycles per iteration of the loop.

d)

The speedup would be 12/11 = 1.09.

Problem 6

We study the performance of an application on processors B. Suppose the instructions executed in the application break down into the following categories.

Type	Percentage
Arithmetic and Logic	30%
Load	30%
Store	15%
Branch	25%

The ideal CPI is 1. However, the actual CPI is higher because of hazards. 10% of the load instructions cause 1 cycle stall because of the load-use hazards. 60% of the executed branches are taken.

1. What is the CPI overhead from data hazards?
   .3 * .1 * 1 = .03
2. What is the CPI overhead from control hazards?
   .25 * .6 * 3 = .45
3. What is the average CPI of this application on Processor B?
   1 + .03 + .45 = 1.48

Problem 7

We study the performance of an operation Processor B. The two instructions below perform the **operation**: increment register x11 by 1 if and only if x10 is non-zero (not necessarily 1).

```
        beq   x10, x0, skip
        addi  x11, x11, 1
skip:
```

1. How many cycles does the operation take if x10 is 0?
   There would be 4 cycles as 1 cycle for the branch and 3 cycles for the stall.

2. How many cycles does the operation take if x10 is not 0?
   Since there is no control hazard, there would be 2 cycles, one for each instruction.

3. If x10 is 0 80% of the times when the operation is performed, how many cycles does the operation take on average?
   .8 * 4 + .2 * 2 = 3.6