2. Three enhancements with the following speedups are proposed for a new machine: Speedup(a) = 30, Speedup(b) =20, and Speedup(c) =15. Assume that for some set of programs, the fraction of use is 25% for enhancement (a), 30% for enhancement (b), and 45% for enhancement (c).

(2.1)If only one enhancement can be implemented, which should be chosen to maximize the speedup?Why?

(2mark)

$T = \frac{T_{affected}}{n} + T_{unaffected}$.

$T_A = \frac{25x}{30\times100} + \frac{75x}{100}$

$T_B = \frac{30x}{100\times20} + \frac{70}{60}x$

→ Speedup c is to be used.

~~$\frac{45}{100}\times15$~~ T_c is the largest speedup available

$T_c = \frac{45}{100\times15} + \frac{55}{100}x$.

T_c is the smallest ∴ we use T_c.

2)If two enhancements can be implemented, which should be chosen, to maximize the edup? why?

(2 mark)

(b) and (c) are to be chosen.

the combination of b and c offer maximum speedup i.e. lowest time

$T_B = \frac{71.5}{100}x$.

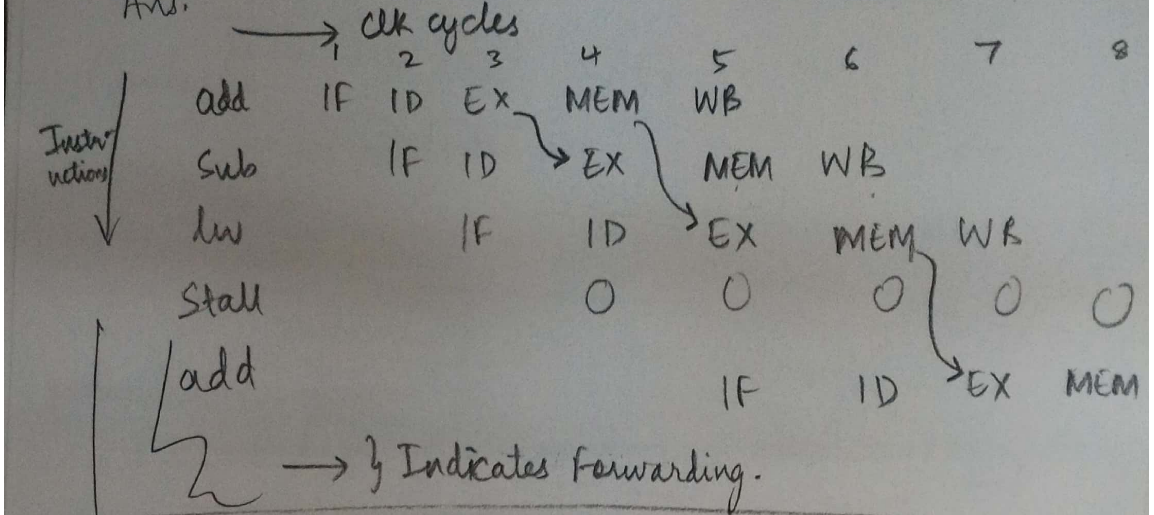
$T_C = \frac{58}{100}x$

9. Show the forwarding paths needed to execute the following four instructions:

```
add $s3, $t4, $t2
sub $t5, $s3, $t1
lw $t6, 4($s3)
add $t9, $s3, $t6
```

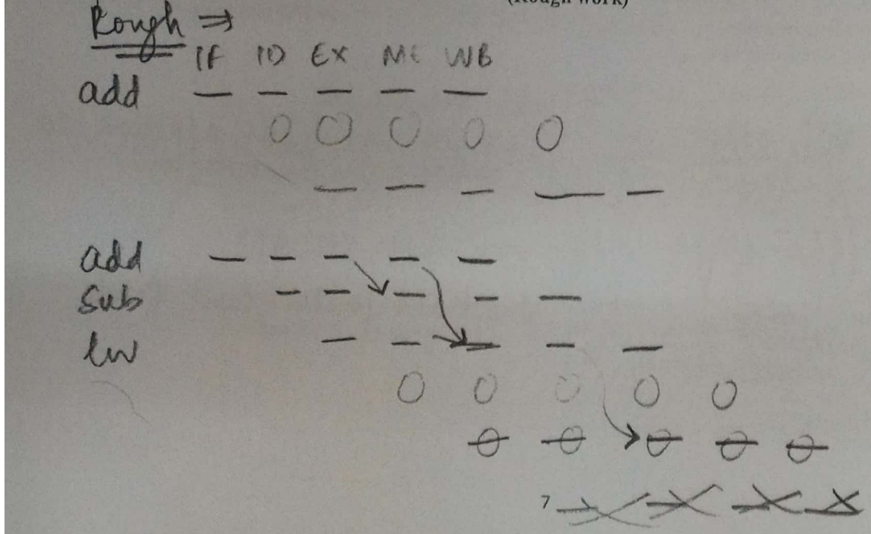
[1.5 marks]

Ans.



	1	2	3	4	5	6	7	8
add	IF	ID	EX	MEM	WB			
Sub		IF	ID	EX	MEM	WB		
lw			IF	ID	EX	MEM	WB	
Stall				O	O	O	O	O
add					IF	ID	EX	MEM

(Rough work)

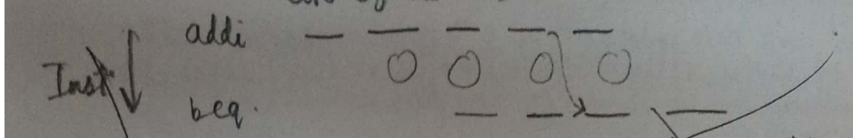


7. For the following combination of instructions, identify whether any hazard(s) exist. If there is a hazard, write the number of stall cycles for with and without data forwarding. Also justify your answer [0.5 *2 Marks]

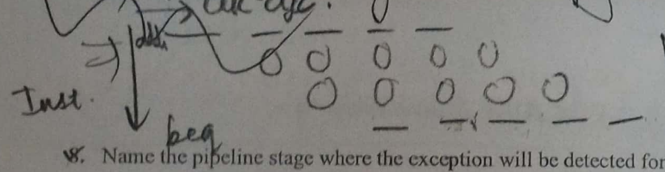
addi $s1,$s1,4
beq $s1, $s2,L1

→ There is a RAW (~~Write~~ Read after write dependency)

⇒ So, there exists a hazard (data).



⇒ One stall cycle with forwarding



Without forwarding, 2 stall cycles.

8. Name the pipeline stage where the exception will be detected for the following cases

a) Arithmetic overflow [0.5*2 marks]

b) Undefined instruction

(a) WB stage { Eg. add $t2, $t1, $t2
32 bit gets exceeded on addition, so, write register overflow.

(b) IF stage { Eg. xxx $t1, $t2, $t3
Undefined Instr. can't fetch
(in IF stage) Instruc. is fetched only.

c) Does support of the new instruction introduce any new hazards?

(a) Introduces Control hazard as PC incremented o in/after WB stage.

(b) No hazard of any kind

6. Suppose a program consists of five conditional branches. Below are the outcomes of each branch for one execution of the program core (T for taken and N for not taken).

a) Branch 1: T-T-T
b) Branch 2: N-N-N-N
c) Branch 3: T-N-T-N-T-N
d) Branch 4: T-T-T-N-T
e) Branch 5: T-T-N-T-T-N-T

Assume that the behavior of each branch remains the same for each program core execution. For dynamic branch prediction schemes, assume that each branch has its own prediction buffer and each buffer is initialized to the same state before each execution. List the predictions and the accuracies for the "Always not taken" prediction strategy.

[0.25*5+.75=2 marks]

Answer: Assuming buffer initial ~~value~~ = 0,

	Count when prediction strategy goes right	Count when prediction strategy goes wrong
a) Branch 1: T-T-T	0	~~0~~ 1
b) Branch 2: N-N-N-N	1	0 0
c) Branch 3: T-N-T-N-T-N	1	0
d) Branch 4: T-T-T-N-T	0	1
e) Branch 5: T-T-N-T-T-N-T	0	1

Accuracy = Always not taken accuracy < 50%.

3. Compare the performance for single-cycle, multi-cycle, and pipelined design by the average instruction time. The instruction frequencies are given by: 25% for loads, 10% for stores, 10 % for branches, 5% for jumps, and 50% for ALU instructions. The functional unit times are 200 ps for memory access, 100 ps for ALU operation, and 50 ps for register file read or write. For the multi-cycle design, the number of clock cycles for each instruction depends on the instruction class or type. For the pipelined design, loads take 1 clock cycle when there is no load-use dependence and 2 when there is. Branches take 1 when predicted correctly and 2 when not. Average time for jumps is 2 clock cycles. Other instructions take 1 clock cycle. For pipelined execution, assume that half of the load instructions are immediately followed by an instruction that uses the result and that one-quarter of the branches are mispredicted. Ignore any other hazards.

[1+1+1=3 marks]

1. Single cycle implementation:
   Slowest instruction = load

lw : Mem-access ✓
ALU op ✓
Reg write ✓
$\Rightarrow (1)(50) + 100 + 200 = \underline{350}$ ps.

$\Rightarrow$ Execution time in this imp. = (350 ps)(No: of Clock cycles)

2. Multi cycle implementation:

3. Pipeline implementation:
   Value of 1 clk. cycle time = 200ps.

2. Consider an instruction pipeline with five stages without any branch prediction: Fetch Instruction (FI), Decode Instruction (DI), Fetch Operand (FO), Execute Instruction (EI) and Write Operand (WO). The stage delays for FI, DI, FO, EI and WO are 9 ns, 7 ns, 10 ns, 8 ns and 6 ns, respectively. There are intermediate storage buffers after FI,DI,FO,EI & WO stage and the delay of each buffer is 1 ns. A program consisting of 10 instructions (I1, I2, I3,...I10) is executed in this pipelined processor. I4 is the only branch instruction and its branch target is I9. If the branch is taken during the execution of this program, find out the time (in ns) needed to complete the program. [3 marks]

I1
I2
I3
I4 branch. label {beq --, L1)
I5
I6 } ⇒ These instructions
I7 } are not executed.
I8
I9 L1
I0.

It is a taken branch.

⇒ After I4, control jumps to I9.

Stages	FI	DI	FO	EI	WO
Delays	9	7	10	8	6

} All units in ns.

⇒ Max^m. ~~latency~~ = FO stage = 10 ns.

⇒ CCT = 10 ns.

143 ns

Clock cycles →
1 2 3 4 5 6 7 8 9 10 11 12 13

```
I1   FI DI FO EI WO
I2      FI DI FO EI WO
I3         FI DI FO EI WO
I4            FI DI FO EI WO
instruction.I4
I5 }
I6. } Not executed
I7 }
I8 }
I9               FI DI FO EI WO
I0.                 FI DI FO EI WO
```

∴ Time taken to complete the program is CCT × 10 clk cycle
Here CCT = 10 ns.

⇒ Also,
⇒ Time1 = CCT × 10
= 10 × 10
= 100 ns.

(Time 2) = Time due to buffer.
= 1(5) = 5 ns.

⇒ Time = 5 × 6 inst. = 30 ns

ANSWER ⇒ Time needed to complete the program = 100 ns + 30 ns = 130 ns

8 | 6+½

Roll No: B120735CS Name: AISHWARYA MARUVADA

National Institute of Technology Calicut

Department of Computer Science & Engineering

CS2004 – Second Mid Term Exam (Winter Semester 2013-14)

Max. Marks: 20 Time: 1 Hour

(Check whether the question paper consists of 9 questions)

1. Consider the following instructions:

```
add  $t0, $s4, $s5
sub  $t1, $s5, $s4
lw   $s3, 4($t0)
sub  $s1, $t1, $s3
```

4/4 ½

What is the average CPI for the following four ways of execution?

(a) single cycle design.

Avg. CPI = $\frac{\cancel{4\times1+4\times1+}(5\times4)\cancel{+4\times1}}{\cancel{4}\ 4}$ = ~~4~~ 5 (slowest instru is lw)

(b) multi-cycle design without pipelining.

Avg. CPI = ~~$\frac{5\times4}{4}$ = 5 (Only due to lw)~~ $\frac{(4\times1)3+5}{4}$ =

(c) pipelined designed without data forwarding hardware.

Avg. CPI = ? CPI_{load} = 1, $\overset{CPI}{Add}$, Sub ⇒ CPI = 1.

$$= \frac{\sum CPI \times IC}{\sum IC} = \frac{2(1)+1(1)+1(1)}{4} = \underline{1}$$

(d) pipelined design with data forwarding hardware . [0.5+1+1+1=3.5 marks]

Avg. CPI =

b) Find all data dependence in the instruction sequence. [0.5 mark]

c) Rearrange your code for to achieve a better performance. [1 mark]

11. Consider an original machine is a 5 stage pipeline with 1 ns. The second machine is a 12 stage pipeline with 0.6ns clock cycle. The 5 stage pipeline experiences a stall due to a data hazard every 5 instructions where as the 12 stage pipeline experiences 3 stall every 8 instructions. What is the speed up of the 12 stage pipeline over the 5 stage pipeline taking data hazard into account? [1.5 marks]

consider 40 instructions.

~~$\frac{\text{Speedup}_{12}}{\text{Speedup}_5}$~~

12 stage

3 stall in 8 instruction,

15 stall in 40 instruction

51+ 15 = 66 clock cycl

5 stage

1 stall in 5 instructions => 8 stall in 40 instruct

51+8 = 59 clock cycles

$$\frac{\text{speedup}_{12}}{\text{speedup}_5} = \frac{66 \times 0.6ns}{59 \times 1}$$

$= 0.67119$.

c) Does support of the new instruction introduce any new hazards?

9. Assume the instructions executed by a pipelined processor are broken down as follows

Add	Beq	Lw	Sw
50%	25%	15%	10%

a) Assuming there are no stalls and the 60% of all conditional branches are taken, in what percentage of clock cycles does the branch adder in the EX stage generate a value that is actually used ? [1 marks]

$1 - \frac{60}{100} \times \frac{25}{100}$

b) How often (percentage of all cycles) do we use the data memory? [0.5 mark]

$\frac{2}{5} = \underline{40\%}$

10. a) Translate the below C code in to MIPS instructions [1 mark]

```
for( i=0; i!=j; i++)
    b[i]=a[i];
```

i & j will be assigned to $s1, $s2. And $s3, $s4 indicates the starting address of a & b.

```
add $s1, $zero, $zero
~~[illegible]~~slt $ $1, $s2
    sll $t1
    lw $t0,
```

c) If we can split one stage of the pipelined data path into new stages, each with half the latency of the original stage, which stage would you split and what is the new clock cycle time of the pipelined processor

MEM.

$New = \frac{400 \times 5}{5} = \frac{2000ps}{5} = 400ps.$

Consider the following fragments of code

```
lw $s5, -16 ($s5)
sw $s4, -16 ($s4)
lw $s3, -20 ($s6)
beq $s2, $zero, Lbl   ; Assume $s2!=$zero
add $s5, $s1, $s4
```

What is the total execution time of this instruction sequence in the 5 stage pipeline that only has one memory? [1.5 mark]

lw IF ID EX MEM WB

sw IF ID EX MEM WB

lw IF ID EX MEM WB

IF ID EX MEM WB

beq

add IF ID EX MEM W

12 x clockcycle time.

Consider the following new instruction [0.5 * 3 marks]

bezi (Rs), Label ; if mem[Rs]=0 then PC=PC + offset

a) What must be changed in the pipelined data path to add this instruction to the MIPS ISA?

b) Give a sequence of MIPS instruction that can replace this instruction?

$s0 ← Rs

beq $s0, $zero, label.

5. Write MIPS code to **Evaluate the Polynomial** $ax^2 + b.x + c$. Imagine that **a, b & c are real** numbers. [1.5 marks]

x ← $t0

mul $t1, $t0, $t0

Assume that **individual stages** of the data path have the following latencies

IF – 300 ps, ID – **400ps**, EX - 350ps, MEM – 500 ps, WB – 100ps [0.5 X 3 Marks]

a) What is the **clock cycle time** in a pipelined and non pipelined processor?

500

1650

Pipelined = $\frac{500 \times 5}{5} = \frac{2500}{5}$. 500 ps.

Non-pipelined = $\frac{300 + 400 + 350 + 500 + 100}{5}$

= $\frac{1650 ps}{5}$.

= 330ps

b) What is the **total latency of a *lw*** instruction in a pipelined processor?

500 × 5 = 2500 ps.

Consider data cache is 64KB, 32 bit physical address field and number of blocks 1024. Find the size of a block, figure out the cache address and find the total size of the cache. [3 marks]

64 kB data cache.

No.of blocks = 1024

size of a block = $\frac{64 \times 1024\ B}{1024}$ = 64 Bytes

= 16 words

4479

241221

16 bit	10bit	6 bit
Tag	Index	Byte offset

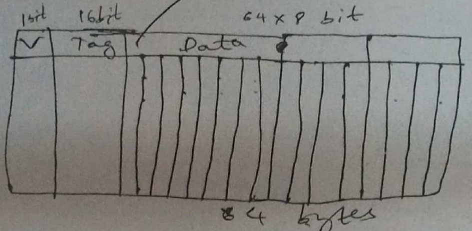
$2^6 = 64$

6 bit ~~byte~~ offset

1024 = 2^{10}

10 bit index

32 - 16 = 16 bit Tag



3

Total size of cache = (1 + 16 + 64×8) × 1024 bits

$\downarrow$ V $\quad \downarrow$ Tag $\quad \downarrow$ Data

= 541696 bit

= 67712 bytes

= 66.125 kB

4. multiply the following IEEE 754 single precision FP (0111 1111 1000 0000 0000 0000 0000 0000 0000)₂ and (1011 1110 1110 0000 0000 0000 0000 0000)₂ [1.5 marks]

2^8

0 1111 1111 0000000000000000000000000

$= +1.0 \times 2^{255-127}$ $= 1.0 \times 2^{126}$

1×2

1 01111101 1100000 ...

1.0×2

0.11 $2^{-1} + 2^{-2} = 0.75$

01111101

$-1.75 \times 2^{125-127}$ $= -1.75 \times 2^{-2}$

1×1.75 = 1.75 $2^{126-2} = 2^{124}$

-1.75×2^{124}.

$1+0\times2^{-1}+0\times2$

Roll No: B110153CS ~~ANJALI.K.J~~ Name ANJALI.K.J

National Institute of Technology Calicut

Department of Computer Science & Engineering

CS2004 – COMPUTER ORGANIZATION

Second Mid Term Exam (Winter Semester 2012-'13)

Max. Marks: 20 **Time: 1 Hour**

1. Represent $(-0.025)_{10}$ in IEEE 754 Single Precision binary FP format. [2 marks]

$(-1)^S(1+F)\times2^E$

$E=B+e$

$B=127$ for SP

$0.025\times2=0.05$

$0.05\times2=0.1$

$0.1\times2=0.2$

$0.2\times2=0.4$

$0.4\times2=0.8$

$0.8\times2=1.6$

$0.6\times2=1.2$

$=(0.00000\,110011)_2$

1.10011×2^{-6}

$E=127-6=121$

$1<121<254$

~~$(-1)^1\times1+0.00000110011)\times2$~~

$(-1)^1(1+0.10011)\times2^{121}$

1	121	10011

2. what would be the likely range of numbers if 16 bit IEEE 754 FP format with 5 Exponent bits? [2 marks]

1	5	10
S	E	F
1	2^5	2^{10}

1	5	10.
	2^5	2^{10}

127

$-2^{10}\times2^5$ to $2^{10}\cdot2^5$

-30 to 31 – E

-1024 to 1023 – F