kv2154

(2) Given two register are X5 and X6, The best algorithm to swap 2 registers without using a 3rd is:

```
XOR  X5, X6
XOR  X6, X5
XOR  X5, X6
```

(3)

(3.1)

addi is an I-Type Instruction
Hence, it will take up read, two registers will read, and one register is written

∴ Energy Spent
= i mem + 2x register read + register write
= 140 pJ + 2x70 pJ + 60 pJ
= 340 pJ

(3.2)

ld is a R-type instruction
Hence, energy spent is similar to itype plus another 140 pJ

∴ Energy spent
= 140 pJ + 2x70 pJ + 60 pJ + 140 pJ
= 480 pJ

KV2154

3

beq will take up i-mem and 2 register

∴ Energy spent,
= 140 pJ + 2 × 70 pJ
= 280 pJ

4

~~[redacted]~~

4.1

Data forwarding alone does not suffice as by the time the first ALU instruction already completes the second step, the second ALU Instruction is already there in first ALU step. Thus, Its too late.

Hence, NOP between 2 Instructions as well as the forwarding between Second ALU step and decode step is required.

4.2

	0	1	2	3	4	5	6	7	8
add x1, x2, x3	IF	ID	ALU1	ALU2	MEM	WB			
NOP									
add x5, x4, x1			IF	ID	ALU1	~~DATA~~ ALU2	MEM	WB	

IKV2154

Q1

(i)

```
if( g > h)
    g = g + 1
else
    h = h - 1
```

```
BLT   X5, X6, CON
addi  X5, X5, 1
CON:  addi  X6, X6, -1
```

(ii)

```
if(g <= h)
    g = 0;
else
    h = 0;
```

```
BGE  X5, X6, CON
LD, X6, X0
CON: LD X5, 0
```

(5)

$4 + (3+4+3) \times 5 + 3 = 57$ clock cycles

Number of instructions executed =
$1 + (3 \times 5) + 1 = 17$

$CPI = 57/17 = 3.35$

For a 5 Stage pipeline $= 5 + (17-1)$
$= 21$

$\therefore CPI = 21/17 = 1.23$

(6)

XOR s1, s2, s3	IF	ID	EX	MEM	WB					
addi s0, s3, -6		IF	ID	EX	MEM	WB				
lw s3, 16(s7)			IF	ID	EX	MEM	WB			
sw s4, 20(s1)				IF	ID	EX	MEM	WB		
or t2, s0, s1					IF	ID	EX	MEM	WB	

↑ 5th cycle.

At Stage 5, XOR is in write back stage.
At Stage 5, addi is in memory stage.
At Stage 5, lw is in execute (ALU) stage
At stage 5, sw is in decode Stage
At Stage 5, or is in fetch Stage
Hence, only S1 is written in 5th Stage.

KV2154

7) or X13, X12, X11
ld X10, 0(X13)
ld X11, 8(X13)
add X12, X10, X11
Subi X13, X12, 16

7.1)

Data hazards and resolutions :-

→ Ex to 1st RAW hazard occurs when value of X13 is accessed by ld instruction when it is still in use by ex Stage of or Instruction. It can be resolved with 2 NOPs

→ EX to 2nd RAW hazard occurs when value of X11 is accessed by 2nd ld instruction. It can be resolved with 2 NOPs

→ MEM to 1st and MEM to 2nd occurs when add instruction tries to access X11 and X10 respectively. It can be resolved with 2 NOPs

→ Ex to 1st RAW hazard occurs when value of X12 is accessed by Subi instruction while it is still in use by EX of add instruction.

(8)

KV2154

```
OR x13, x12, x11
Nop
Nop
ld x10, 0(x13)
ld x11, 8(x13)
Nop
Nop
add x12, x10, x11
Nop
Nop
Subi x13, x12, 16
```

KV2154

7.2

	1	2	3	4	5	6	7	8	9	10
or	IF	ID	EX	MEM	WB					
ld		IF	ID	EX	MEM	WB				
ld			IF	ID	EX	MEM	WB			
NOP										
add					IF	ID	EX	MEM	WB	
subi						IF	ID	EX	WB	

8 A

Instruction / Program :-
It will decrease. The new Instruction will complete add in one Instruction which could have originally taken 2 Instruction

CPI :-
It has no effect.
It will complete 3 add in same Instruction. So no change in the CPI

Complexity :-
It will increase. Higher complexity of circuitry is required to handle one or more operand.