24/25

Fall 2023 CSE340 Online Midterm Exam

Full name : Rejwan Shafi

ID : 23241108 (21301155)

Section : 09

Question Set: A

Answer to the question no 1

(a)

Elapsed time: Elapsed time and CPU time are the same time. Elapsed time is the total time taken by a computer to execute a program,

CPU time:
CPU time is the total time taken by a computer to execute a program.

(b)

The power trend equation states that,

$$P = \text{Capacitive load} \times \text{Voltage}^2 \times \text{Frequency}$$

$$\therefore \frac{P_{new}}{P_{old}} = \frac{0.88C \times (0.79V)^2 \times 0.79F}{C \times V^2 \times F}$$

$$= 0.43387432.$$

~~$\therefore (1 - 0.43387432) = 56.612$~~

$\therefore (1 - 0.43387432) \times 100\% = 56.612\%$ can be reduced in the new system compared to the old System

(c)

Given that

Total time, $T = 120$ s

$\therefore T_{affected} = 120 \times (100-17)\%$ s

$= 120 \times 83\%$ s

$= 99.6$ s

$\therefore T_{unaffected} = 20.4$.

we want to improve the performance by 2.56 times

$\therefore T_{improved} = \frac{T}{2.56} = \frac{120}{2.56} = 46.875$

improving factor, $n = ?$

According to Amphdal's law

$$T_{improved} = \frac{T_{affected}}{n} + T_{unaffected}$$

$$\Rightarrow 46.875 = \frac{99.6}{n} + 20.4$$

$$\Rightarrow \frac{99.6}{n} = 46.875 - 20.4$$

$$\Rightarrow \frac{99.6}{n} = 26.475$$

$$\Rightarrow n = 3.76203.$$

$\therefore$ So to improve our overall performance by 2.56 times; we have to improve our Taffected time by 3.762 times. ~~Then, our~~ To achieve our goal our time taken by parallel operation should be 26.475 seconds.

Answer to the question no 2

(a)

Instruction format for j type instruction is as follows

Opcode	offset or Address
6 bits	26 bits

There is no need of ALU or adder in case of jump address calculation. Because, while we calculate the jump address we first 2 bit left shift our 26 bit offset or address. Then we get 28 bit address representation of our offset address. Still 4 bits are left for 32 bit address. and for that left out 4 bits we ~~cor~~ use our PC's first 4 MSB bit. There fore, we ~~can~~ don't need ALU or adder in jump address calculation.

Our given j type instruction is,

JAL 120

and pc's address is $= (0 \times 00000000)_{16}$

$\therefore$ PC's address in binary would be

$= (0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000)_2$

$\therefore$ offset is $(120)_{10}$

$\therefore$ offset in 26 bit binary ~~= 00 0000~~

$= 00\ 0000\ 0000\ 0000\ 0000\ 0111\ 1000$

$\therefore$ offset after 2 bit left shift; so 28 bit representation of offset

$= 00\ 0000\ 0000\ 0000\ 0000\ 0111\ 1000\ 00$

$\therefore$ Jump address = (PC's 4 MSB bits): (28 bit rep. of offset)

$= 0000 : 00\ 0000\ 0000\ 0000\ 0000\ 0111\ 1000\ 00$

$= 0000.\ 00 \cdot 0000 \cdot 0000\ 0000\ 0000\ 0111\ 1000\ 00$

$= (000001E0)_{16}$

$\therefore$ Jump address 0x 00000 1E0.

j type instruction format for the given mips code would be

xxxxxxx	0000000000 0 0 0 0 0 0 0 0 0 0 0111 1000

(b)

Assuming that pc holds the address

add $5, $6, $9 // $55 = 12 + 4 = 16

sll $5 , $5 , 3 // $55 = 128 ,

srl $5 , $5, 2 // $55 = 32 .

~~sw $55, 40 ($56)~~ // $55 = 172

sw $55, 40 ($56) # 55 = 32 ; address = 12 + (40x4)

= 172 .

(c)

```
lui $4, 1234
Ori $4, $4, 5678
```

(d)

Given, MIPs code is,

$$A[3] = 12 \times X - 7 * A[5] + 33 * Y - 70;$$

Base address of A = $s0,
X = $s1.
Y = $s2

MIPs Code:

```
lw $t0, 20($s0)

sll $t1, $t0, 3
sub $t1, $t1, $t0      # 7*A[5]

sll $t0, $s1, 3        # 8X
add $t0, $s1, $t0
add $t0, $s1, $t0
add $t0, $s1, $t0
add $t0, $s1, $t0      # 12X
Sub $t0, $t0, $t1      # (12X - 7A[5])
```

sll $t1, $s2, 5

add $t1, $s2, $t1, # 33Y

add $t0, $t0, $t1, # (12X - 7A[5] + 33Y)

addi $t0, $t0, -70

sw $t0, 12 ($s0)

(e)

addi $9, $18, -1

10 0011	10010	01001	1111 1111 1111 1111

∴ Machine Code's Corresponding Hex would be

= (1000 11 1 0010 0 1001 1111 1111 1111 1111)

= 8E49 FFFF