Everett Salley

EEL 4713 Assignment #1

Section 1: Done

Section 2: ch1 -

1.1.1) Servers
2) petabyte
3.) Super computer
4.) virtual worlds
5.) RAM (12)
6.) CPU (13)
7.) data centers
8.) multicore processors
9.) low-end servers
10.) embedded computer
11.) VHDL
12.) desktop computers
13.) compiler
14.) assembler
15.) Cobol
16.) machine language
17.) instruction
18.) Fortran
19.) assembly language
20.) Operating System
21.) application software
22.) bit
23.) systems software
24.) C
25) high-level language
26.) Terabyte

1.7 → 1.) Ratios

clock	1.28	1.5625	2.64	3.03	10	1.8	0.74
power	1.2424	1.195	2.06	2.88	2.587	1.3678	0.9223

Geometric means: $\boxed{\begin{array}{l}\text{clock} \to 2.15107 \\ \text{power} \to 1.61573\end{array}}$

2.) largest clock change ⇒ 10 (Pentium Pro → Pentium 4 Williamette)
largest power change ⇒ 2.88 (Pentium to Pentium Pro)

3) clock ratio → $\frac{2667}{12.5} = 213.36$

power ratio → $\frac{95}{3.3} = 28.7879$

4) Power = (Capacitive load)$\cdot$(Voltage)^2$\cdot$(Frequency) $C = \frac{P}{(V^2 F)}$

	Capacitive load
80286	10.56 nF
80386	10.25 nF
80486	7.84 nF
Pentium	6.1212 nF
Pentium Pro	13.36 nF
Pentium 4 Williamette	12.2939 nF
Pentium 4 Prescott	18.311 nF
Core 2 Kersfield	29.4385 nF

5) Voltage Ratios

$5 \to 3.3 = 0.66$

$3.3 \to 1.75 = 0.53$

$1.75 \to 1.25 = 0.7143$

$1.25 \to 1.1 = 0.88$

largest change = 0.53
Pentium Pro to Pentium 4 W

6.) geometric mean = 0.684866

$\$t_1 =$ some kind of count value

EEL 4713 Assignment #1

Section 3: Lab1.s

a) let $a0 = x = 6$
$a1 = y = 4$
$a2 = z = 8$

does $\$t0 = 6 \cdot 4$

32 $\$t2 = 8 + (6 \cdot 4) \longrightarrow$

16 $\$t3 = (6 \cdot 4) - 8 \longrightarrow$

$\$v0 = 0$

performs $\$v0 = (8 + (6 \cdot 4)) \cdot \frac{((6 \cdot 4) - 8)}{2}$

$= 256$

```
main: addi $a0, $0, 6
      addi $a1, $0, 4
      addi $a2, $0, 8
      addi $t0, $0, 0
      addi $t1, $a1, 0      (t1 = a1)

loop1: add $t0, $t0, $a0    (t0 = t0 + a0)
       addi $t1, $t1, $-1   (
       bne $t1, $0, loop1

       add $t2, $t0, $a2
       sub $t3, $t0, $a2
       add $v0, $0, $0
loop2:
       add $v0, $v0, $t2
       addi $t3, $t3, -2
       bne $t3, $0, loop2
```

so let $\$a0 = x$, $\$a1 = y$, $\$a2 = q$, $\$v0 = r$

this code performs the operation:

$$r = \frac{[q + (x \cdot y)]\,[(x \cdot y) - q]}{2}$$

for $x = 6$, $y = 4$, $q = 8$
$r = 256$

b) which registers are used to hold operands & results?

arguements

I'm assuming for example add $\underbrace{\$t2}_{\text{result}}, \underbrace{\$t0, \$a2}_{\text{operands}}$

(also immediate values)

i) for the **overall** process $a0, $a1, & $a2 hold the arguements to the function and **$v0** stores the result

therefore, overall

operands: $a0, $a1, $a2, -1, -2
results: $v0

ii) however, some registers hold intermediate values that are both results and operands for later instructions

- $t0 holds the result for a multiply but is used as an arguement for addition & subtraction instructions later
- $t1 & $t3 act as count values for the multiply operations and are use as results & operands
- $t2 stores the result of an addition, but is also the operand in a multiply later on

when including intermediate steps:

operands: $a0, $a1, $a2, -1, -2
results: $v0
both: $t0, $t1, $t2, $t3