29/8/22

1 Problem 2

Our favourite program runs in 10 seconds on computer which has a 2 GHz ~~so~~ clock, we are trying to help a Computer designer built a Computer B, which will run this program in 6 seconds. The designer has determined that a substantial increase in clock rate is possible, but this increase will affect the rest of CPU design, causing computer B to require 1.2 times as many clock cycles as Comp. A for this prog. What clock rate should we tell designer to target?

Soln

Execution time of A = 10 sec

Execution time of B = 5 sec

Clock rate of A = 2GHz

Clock rate of B = 1.2x clock rate of B

CPU time of A = CPU clock cycles / clock rate

$10 = \text{CPU clock cycles} / 2 \times 10^9$

CPU clock cycles of A = $10 \times 2 \times 10^9$

$= 20 \times 10^9$ cycles

CPU time of B = CPU clock cycles of B / clock rate

6 = CPU clock cycles of B / clock rate

= 1.2 x CPU clock cycles of A / clock rate

$6 = (1.2 \times 20 \times 10^9) / \text{clock rate}$

clock rate of B = $(1.2 \times 20 \times 10^9) / 6$

$= 4 \times 10^9$ cycles per seconds

= 4GHz

Clock rate of A = 2GHz

Clock rate of B = 4 GHz

To run this prog. in 6 sec. B must have time than A

Problem 3

Suppose we have two implementations of the same inst. set architecture, Computer A has a clock cycle time of 250 pico sec and a CPI of 2 for some prog. & Comp. B has a clock Cycle time of 500 ps & a CPI of 1.2 for the same prog., which Computer is faster for this prog. & by how much?

For Computer A

Clock Cycle time 250 ps

CPI 2

For B

clock cycle time 500 ps

CPI 1.2

CPU time = inst. count * CPI * clock cycle time

CPU time of A = I x 2 x 250 ps

= 500x I ps

CPU time of B = I x 1.2 * 500 ps

= 600I ps = 600 x I ps

$$n = \frac{\text{Execution time of B}}{\text{Execution time of A}}$$

$$= \frac{600 \times I \text{ ps}}{500 \times I \text{ ps}}$$

= 1.2

Computer A is 1.2 times as fast as Comp. B

Addressing Modes

Method to specify the operand of an instruction

immediate Addressing Mode

MOV A, #25H
↳ source
↳ source

25H aa A register la podanum

MOV R4 #62

Register addressing mode

MOV A, R0
MOV R2, A
A, R5
ADD

A, R0, R2, R5 all register
R0 la ulla etho value iruku athu A la podanum.

Direct Addressing Mode

MOV R0, 40H

MOV 56H, A

irutha value ilana add.

podana value iruntha athu main Memory la irukurathu

40H add la irukura value aa Register R0 la vaikanum

Register Indirect Addressing Mode

MOV A, @R0
MOV @R1, B

R0 ula oru add. iruku athu main Memory la irukura add. ula value aa A la podanum

B la value athu R1 ulla main Memory add la podanum

Indexed addressing mode

MOVC A, @A+DPTR
↳ Datapointer

Register A ula oru add 100
DPTR (2 for eg) 100+2 102
102 ula irukura value A ku Pod...

30/8/21
3

Unit 2 The Arithmetic

Fixed Point Arithmetic

Half adder
full

Booth's Multiplication

or otherwise called Recoding Multiplier
" " Radix-2 Multiplier

7 * 5
Multiplicant * Multiplier

$-(2^{n-1}-1)$ to $+(2^{n-1}-1)$

~~n=6~~ ~~0.5~~

$n=1 \rightarrow 0$ to 0

$n=2 \rightarrow -1$ to 1

$n=3 \rightarrow -3$ to 3

2^2 $n=4 \rightarrow -7$ to 7

$n=5 \rightarrow -15$ to 15

n=4 → 7 * 5 → 0101

0 1 1 1

Travel

0 → 0 = 0

0 → 1 = -1

1 → 0 = 1

1 → 1 = 0

100
011
100

1100
0011
0100

0101
1010
1011

add 0

7 → 0 1 1 1   0 ← 1 ← 0 ← 1 ← $\boxed{0}$

1 -1 +1 -1 ← +1 -1 +1 -1

1 0 0 1

$7 \times -1 = -7$

$\boxed{0111}$
1000
1
1001

7 aa 2^3 Complement with 4 bit bcoz $\boxed{n=4}$

$\boxed{-7}$ so remaining 4 bit fill with 1, result aa 8 bit akanum bcoz 4 bit represent if n=3 result 6 bit

1 1 1 1 0 0
1 1 1 1 1 0 0 1

0 0 0 0 1 1 1
0 0 0 0 0 1
1 1 1 0 0 1
0 0 1 1 1

7×-1
7×1

0 0 1 0 0 0 1 1 → $\boxed{35}$

3 place fill pananum
$7 \times 1 = 7$ so fill with 0