Chapter-1 Maths

2.

2. Calculate the execution time of the programs and SPECratio from the table below as well as the summary benchmark point using geometric mean.

Benchmark Program	Instruction Count $x10^9$	CPI	Clock Rate(GHz)	Reference Time (seconds)
Go	1970	1.10	3.2	10490
Hmmer	2910	0.6	3.2	9330

Go:

$$\text{Execution time} = \frac{IC \times CPI}{CR}$$

$$= \frac{1970 \times 10^9 \times 1.10}{3.2 \times 10^9}$$

$$= 677.1875$$

$$\text{Spec Ratio} = \frac{10490}{677.1875} = 15.4906$$

Hmmer:

$$\text{Execution time} = \frac{2910 \times 10^9 \times 0.6}{3.2 \times 10^9} = 545.625$$

$$\text{Spec Ratio} = \frac{9330}{545.625} = 17.09966$$

$$GM = \sqrt[2]{17.09966 \times 15.4906}$$

$$= 16.27526$$

$$SR = \frac{\text{Ref. Time}}{\text{Exe. Time}}$$

Quiz 1

Date - 10^{th} October

Syllabus - Chapter-1 (Theory + Math)

Duration - 30 minutes

Now, assume that a particular operation takes **2.5X%** of the total execution time. What improvement is required if we want **2.5** times speedup in that operation, where **X** is equal to 10.

$$ET = T_{affected} + T_{uneffected}$$

$$100\% = 2.5 \times 10\% + T_{uneffected}$$

$$T_{uneffected} = 75\%$$

$$T_{effected} = 25\%$$

$$\frac{100}{2.5} = \frac{25}{n} + 75$$

$$\Rightarrow 40 = \frac{25}{n} + 75$$

Time ↑	$\times$
Time ↓	$\div$

not possible for -ve and 0.

$$\Rightarrow -35 = \frac{25}{n} \text{ (Can't be done.)}$$

Suppose you are training a face recognition model, which is heavily dependent on a process **(80%)**. So, you installed a graphics card with to speed up that process. Now, you observe that it is taking only **3 days** to execute, as opposed to **6 days** before installing the card. What is the improvement?

Speed Up ↑ = Performance ↑

= Execution time ↓

$$ET_{BG} = T_{ABG} + T_{UBG}$$

$$\Rightarrow 6 = 6 \times 80\% + T_{UBG}$$

$T_{UBG} = 1.2$ days $T_{ABG} = 4.8$ days

$$3 = \frac{4.8}{n} + 1.2$$

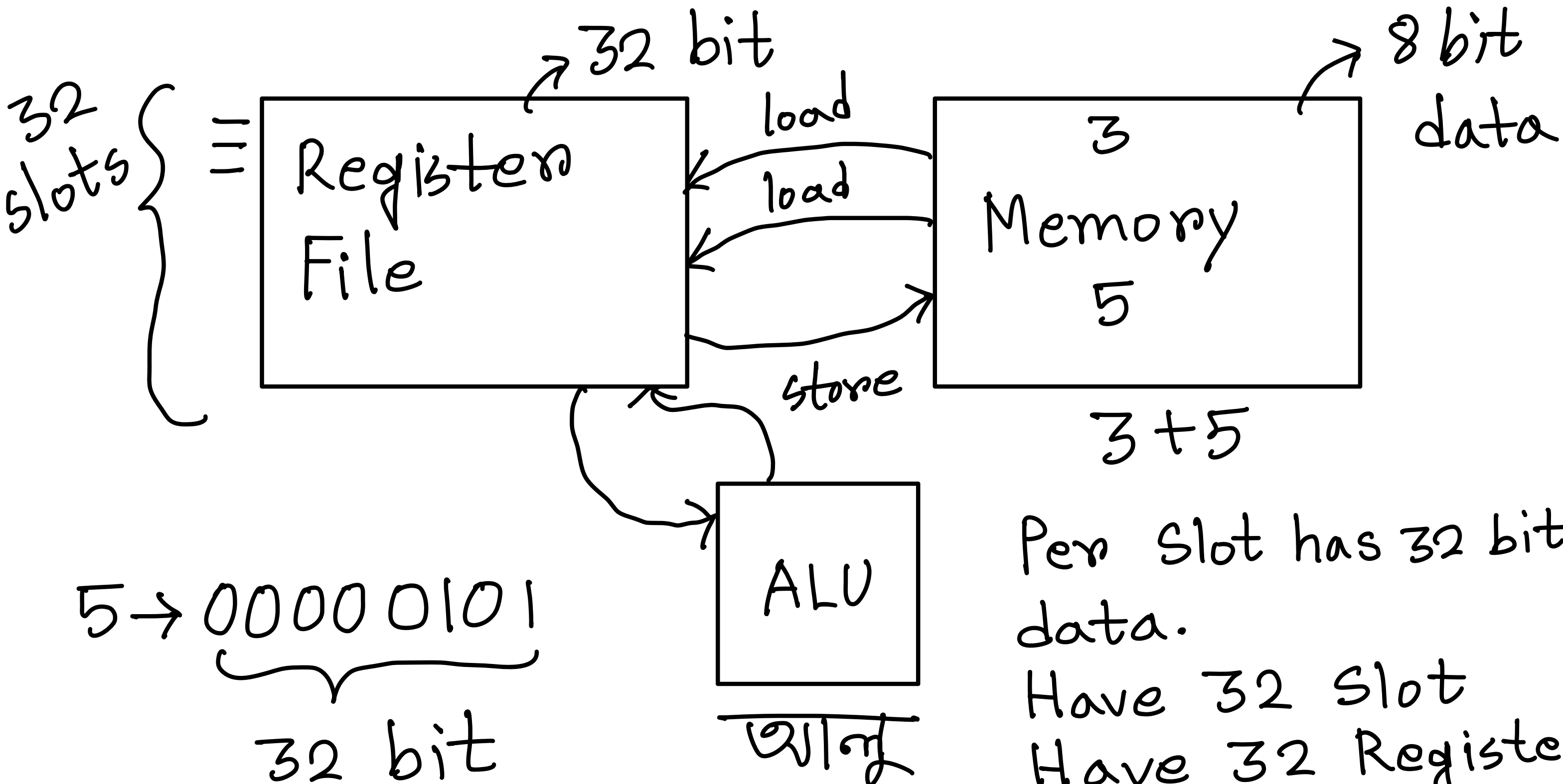
$$\Rightarrow 1.8 = \frac{4.8}{n}$$

$$\therefore n = 2.6667$$

Chapter-2

Language of the Computer

MIPS Architecture (32-bits)



5 → $\underbrace{00000101}_{32\ bit}$

Per Slot has 32 bit data.
Have 32 Slot
Have 32 Register

$2^5 = 32$

Sign/Unsign Value

00000
00001
00010
00011
00100
.....
11110
11111

একেকটা slot এর address বের করে then কত bit address লাগবে address represent করতে 5 bit লাগবে, since $2^5 = 32$

Memory তে Data $\underbrace{\text{Load}}$ আর $\underbrace{\text{Store}}$ রাখে

Load → { retrive করা, memory থেকে নিচ্ছি।

Store → Write Operation

ALU → Access Data from register

Load operation = read operation
store operation = write operation

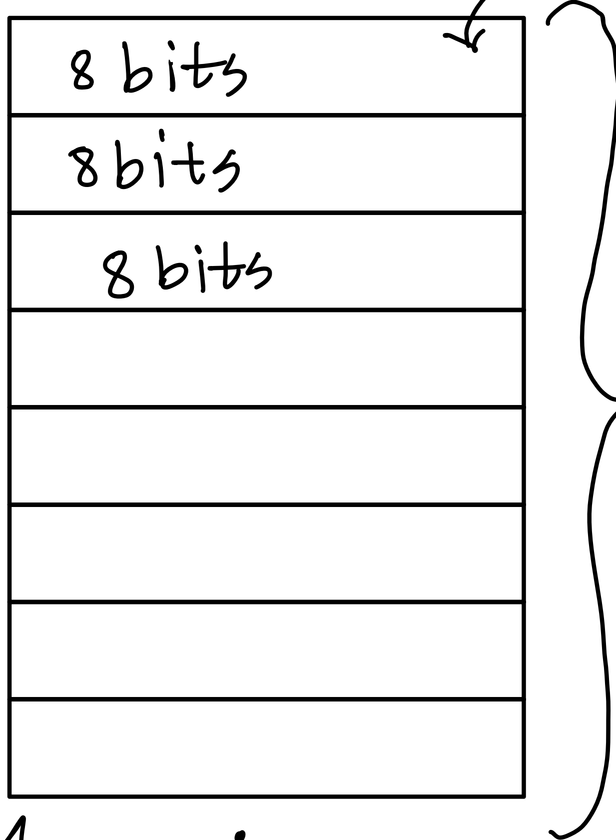
memory → register

32 bit ~~নিয়ে কাজ করছি~~ (0-3) তে

Memory

8 bit

PC → 0 0 | 8 bits
PC+4 → 4 1 | 8 bits
PC+8 → 8 2 | 8 bits
⋮ 12 3
16 4
20 5
⋮ 6
7



(0-3) → data 1
(4-7) → data 2
32 bits

1 slot = 8 bit data
4 slots = 4×8 bits = 32 bits data

0 → 0 + 1 = 1
1 + 1 = 2

8 bit = 1 byte
32 bit = 4 byte = 1 word

(0-3) তে Data 1
(4-7) তে Data 2
Every slot - 32 bits data { প্রত্যেকটা slot য়ে 32 bit data রাখবো। }

Need 32 bits (4 slots) to represent a data

কি Data রাখছি তা 8 bit data এর একেকটা slot যে address n bit দ্বারা Represent করলে 2^n সংখ্যক Combination থাকতে পারে। যেমন:

7 bit দিয়ে Represent করলে 2^7 টা Location থাকে। প্রত্যেকটা Location 8 bit data hold করে।

32 bit Arc. PC তে 4 করে increment করে।

7 bits

$2^7 \rightarrow$ memory size

0 000000
0000001

1 slots = 8 bits

4 slots = 8×4 bits

= 32 bits

↓

data

MIPS is Big Endian

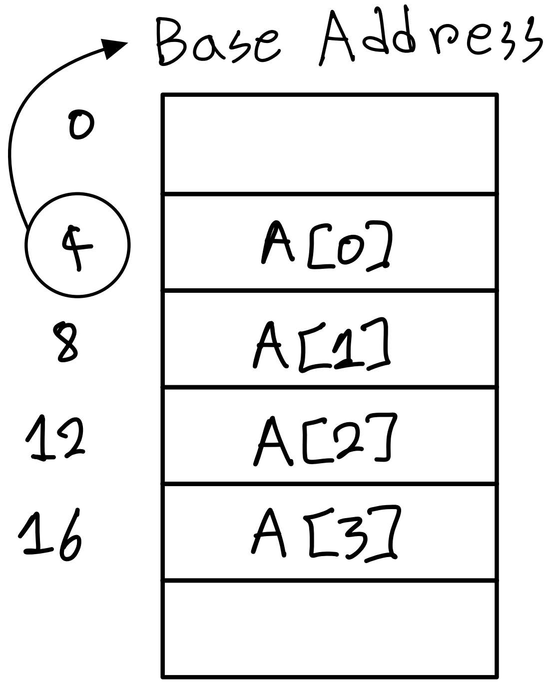
78925

↓ MSB ↓ LSB

LSB − higher address store

Address বের করো:

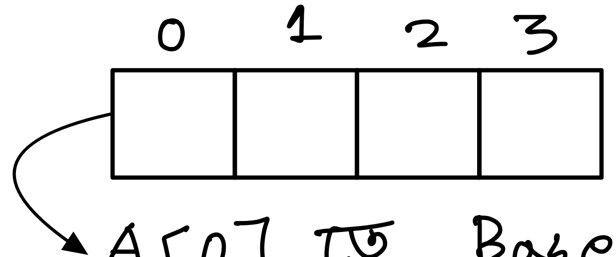
$g = h + A[3]$



MSB – Lower address store

Array, Structure, dynamic data, stack memory তে রাখতে পারি।

Stack এর মতো
Lifo



A[0] তে Base Address
↳ Start

Array টা যেখান থেকে start হচ্ছে।

$3 \times 4 = 12$ slots

কয়টা লেভেল cross করতে হবে।

$4 + 12 = 16$

4 → {initial address}

16 → data {destination address}