# Assignment 2

**2. 12. 1**

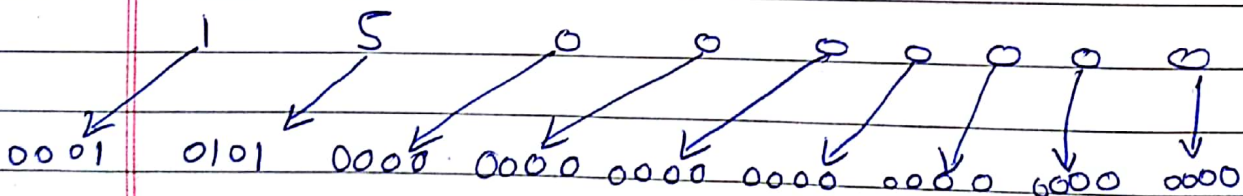$$\$s0 = 0x8000\,0000$$

$$\$s1 = 0xD000\,0000$$

add \$t0, \$s0, \$s1

$$\begin{array}{r} 0x \quad 8000\ 0000 \\ D000\ 0000 \\ + \hline \end{array}$$

\$t0 (Hex) = 1 5000 0000

Converting Hex → Bin

1 5 0 0 0 0 0 0 0

0001  0101  0000  0000  0000  0000  0000  0000  0000

**2.12. 2** As this is more than 32 bits it results in overflow.

## 2.12.3

sub $t0, $s0, $s1

$$0x \quad 8 \; 000 \quad 0000$$
$$0x \quad D \; 000 \quad 0000$$
$$-$$
$$\$t0 = \; -5 \; 000 \quad 0000$$

## 2.12.4

Converting from hex $\rightarrow$ Bin

-5    O    O    O    O    O    O    O

|101  0000  0000  0000  0000  0000  0000  0000
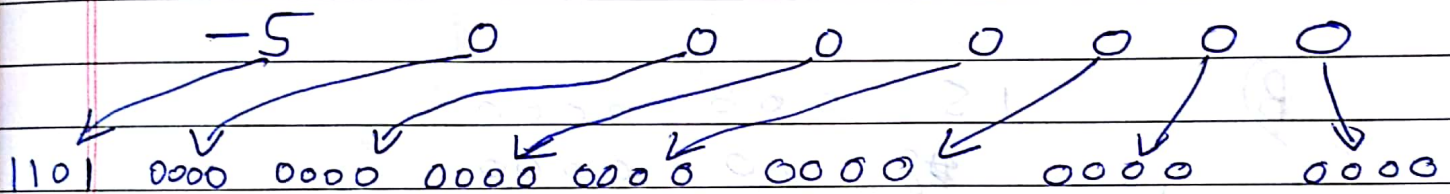
the (1)101 is the signed bit. & the above
is 2's complement

### NO OVER FLOW

2.12.5]

A)   add $t0, $s0, $s1
B)   add $t0, $t0, $s0

A)

$$0_x \quad 8 \quad 000 \quad 00 \; 00$$
$$0_x \quad D \quad 000 \quad 0000$$
$$+$$
$$\overline{\$t0 = \quad 15 \quad 000 \quad 0000}$$

B)

$$15 \quad 000 \quad 0000$$
$$+ \quad 8 \quad 000 \quad 0000$$
$$\overline{1 \; D \quad 000 \quad 0600}$$

Over flow condition.

2.12.6]   Converting from Hex to Bin.



1      D

0001   1101   0000  0000   0000  000 0000  0000  0000

$$\left[13 = 2^3 + 2^2 + 2^0\right]$$

OVER FLOW Condition

2. 14

```
0 000        0 010    0 001    0 000  100 0
                      0 000    00 10  0000 two
```

Instruction format.
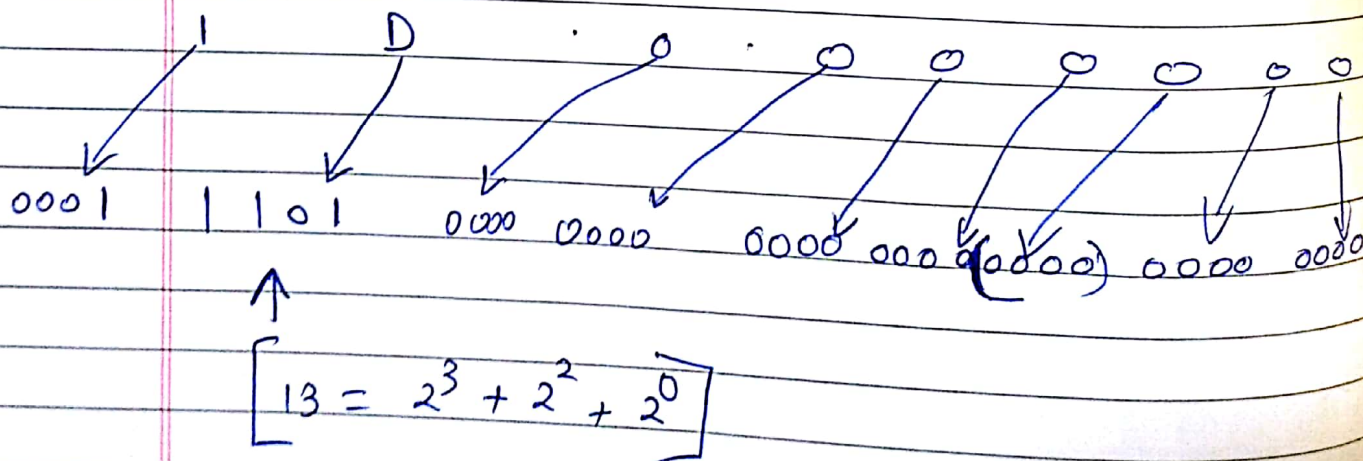
the above is R-type instruction.

Rtype is encoded as:

| B 31-26 | B 25-21 | B 20-16 | B 15-11 | B 10-6 | B 5-0 |
|---|---|---|---|---|---|
| ↓ | ↓ | ↓ | ↓ | ↓ | ↓ |
| Opcode | Reg. S | Reg t | Reg d | Shift amt | func |

| 000000 | 1 0000 | 1 0000 | 1 0 000 | 0 00 00 | 1 00000 |
|---|---|---|---|---|---|
| Op = 0 | Rs = 16 | Rt = 16 | Rd = 16 | shift = 0 amt | func = 0x20 |
| (add) | | | | | |

Thus     add  $s0, $s0, $s0

sw  $t1, 32($t2)

The above is  $\underline{I}$ - type of instruction

sw =  0x 2b

$t1 =  9  =  rt

$t2 =  10 =  rs

Immediate  = 32

| 1 0 1 0 | 1 1 0 1 | 0 1 0 0 | 1 0 0 1 | 0 0 0 0 0 0 0 0 | 0 0 1 0 |

0 0 0 0

0x  A    D    4    9    0    0    2    0
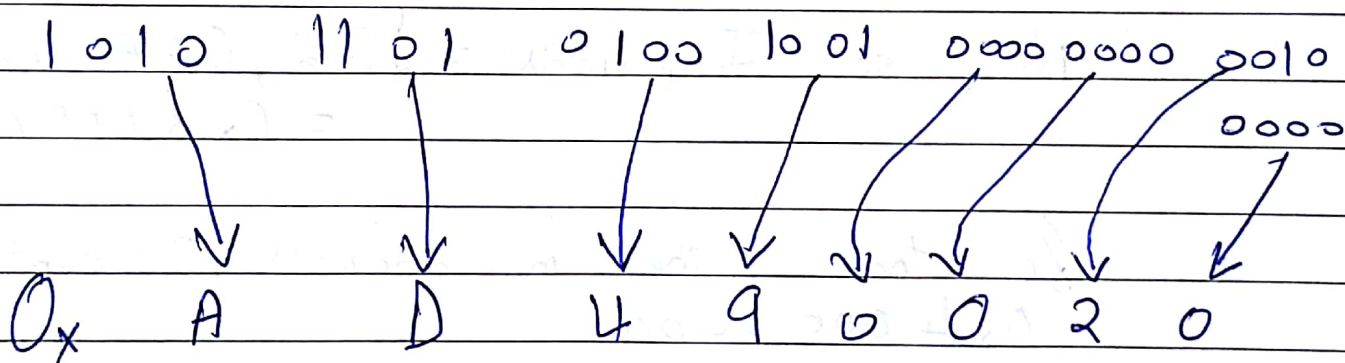
**2.24)**

**Ans)**

For Jump instruction

$$JumpAddr = \{ PC + 4[31:28], address[26], 2'b0 \}$$

PC = 0x 2000 0000

The max lower 26 bits of jump inst could be 0x 3FFF FFF

Shifted left by 2 bits to make it addressable

0x2 0x FFFFFFFC

Now so Jump addrs = { 0x2, 0xFFFFFFC }

= 0x 3 FFF FFFC

So that is the max address. It can't Reach 0x4 000 0000

For branch instruction

$$Branch Addr = PC + 4 + \{ immediate[16], 2'b0 \}$$

And the limit of branch distance is $2^{15}$ to $2^{15} - 1$

So it is not possible to use the beq to set OC to this address

**2.39)**

lui $t1, 0x2001

→ #t1  0010 0000  0000 0001 0000
(t1)          0000  0000  0000

ori $t1, 0x4924 # $t1= 0010 0000  0000
0001  0100  1001  0010 0100

**2.40)**

A single branch instruction cannot be used in to get the PC address in 2.39. as the range can go up to 0x 0 FFFFFFC