

- Last Lecture -

- Basic digital gates
- Combinational logic block
 - Multiplexer
 - Demultiplexer
 - Decoder
 - Encoder
- Multibyte Multibit logic
- Hardware descriptor Language (RTL)
- Test bench using (TSC) & Simulation

- Today's Lecture :-

- Objective - Designing an Arithmetic & Logic Unit
- Data representation
- Basic arithmetic operations
 - Addition
 - Subtraction
- Combinational Logic for arithmetic operation
 - Half adder
 - Full adder
 - n-bit adder & subtractor
- Arithmetic & Logic Unit

Data representation :-

The number system: ~~base~~

- Integer
- Fractional numbers

Integer :-
- Decimal number system (base - 10)
- Octal number system (base - 8)
- Hexadecimal system (base - 16)

A number can be represented as:

$$\begin{matrix} 4 & 3 & 2 & 1 & 0 \\ a_4 & a_3 & a_2 & a_1 & a_0 \end{matrix} = \cancel{+10^4 \times a_4} + \cancel{10^3 \times a_3} + \cancel{10^2 \times a_2} + 10^1 \times a_1 + 10^0 \times a_0$$

$$= 10^4 \times a_4 + 10^3 \times a_3 + 10^2 \times a_2 + 10^1 \times a_1 + 10^0 \times a_0$$

if $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

call it as decimal system
where base is 10

- $a_0 = 2$
- $a_1 = 1$
- $a_2 = 4$
- $a_3 = 0$
- $a_4 = 8$

$$a_4 \ a_3 \ a_2 \ a_1 \ a_0$$

8	0	4	1	2
---	---	---	---	---

$$= 10^4 \times 8 + 10^3 \times 0 + 10^2 \times 4 + 10^1 \times 1 + 10^0 \times 2$$

$$= b^4 \times a_4 + b^3 \times a_3 + b^2 \times a_2 + b^1 \times a_1 + b^0 \times a_0$$

$$= \sum_{i=0}^n b^i a_i \rightarrow \text{base (n+1) digit no.}$$

for binary system

$b = 2$ and $a_i \in \{0, 1\}$

+ve & +ve integer number in binary system:-

Example:- 3 bit number system

$a_2 a_1 a_0$	Mag	2's comp
0 0 0	0	0
0 0 1	1	+1
0 1 0	2	+2
0 1 1	3	+3
1 0 0	4	-4
1 0 1	5	-3
1 1 0	6	-2
1 1 1	7	-1
<u>1 0 0 0</u>	8	

$(+a) + (-a) = 0$

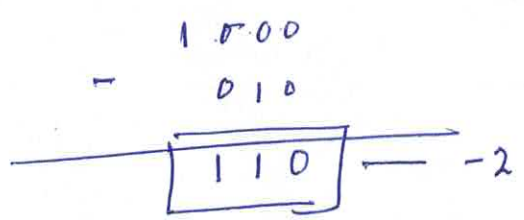
- 1 + 7 = 8
- 2 + 6 = 8
- 3 + 5 = 8
- 4 + 4 = 8

General form =

$2^n - a = -a$

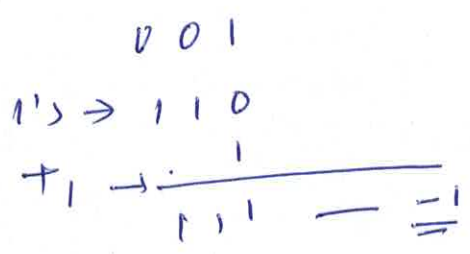
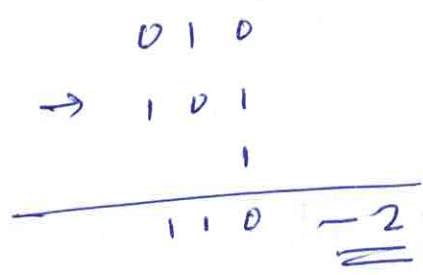
in 2's complement

How to generate 2's complement-number:-



Easy way

One's complement + 1



Overblow 2 largest-possible number :-

Example $\underline{3\text{-bit}} \rightarrow \boxed{-4 \text{ to } 3}$

Example $\underline{4\text{-bit}} \rightarrow \boxed{-8 \text{ to } 7}$

General $\underline{n\text{-bit}} \rightarrow \boxed{-2^{(n-1)} \text{ to } 2^{(n-1)} - 1}$

Arithmetic operation:-

- Addition (+)
- Subtraction (-)

Example (Adding two bits)

Example (Adding three bits)

Combinational circuit

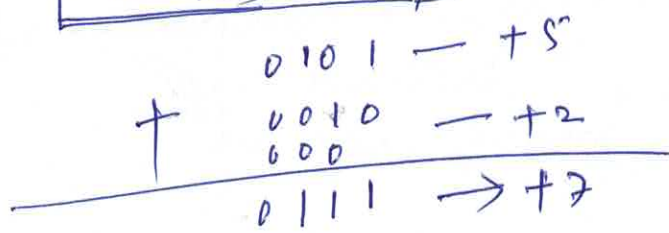
Half adder

Combinational circuit

- Full adder
- Using XOR gates
- Full adder using half adder

Example - n-bit number

$5 + 2 = 7$ → required 4 bit size



Designing a circuit:-

