

Lecture - 1

Digital Circuit

- Number systems

- Logic gates

- Logic family

(~~with~~ Combinational & Sequential circuit)

→ Number system

1) Binary No. System

base = 2 or B

digits = 0, 1

ex. $(101)_B$ or $(101)_2$

2) Octal No. System

base = 8 or O

digits = 0, 1, 2 ... 7

ex. $(345)_8$

3) Decimal

base = 10 or D

digits = 0, 1, ..., 9

ex. $(189)_{10}$

4) Hexadecimal No. system

base - 16 or H

Digits - 0 ~ 9

Symbol	A	B	C	D	E	F
10, 11, 12, 13, 14, 15	10	11	12	13	14	15

5) Base - 4

Digits 0, 1, 2, 3

ex. $(12)_4$

0, 1, 2, 3

10, 11, 12, 13

20, 21, 22, 23

In

Hexadecimal number system

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, 12,

19, 1A, 1B, 1C, 1D, 1E, 1F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F

Base - 14

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, 12, 13

Base - 10 $(10)_10 \equiv (10)_H$

Base - 8 0, 1, 2, 3, 4, 5, 6, 7

Note: largest number in any number system is 1 less than its base.

Conversion:- ~~positive .04 (in binary)~~

Hence 21 - 3rd

① ~~Octal to Binary~~ $\rightarrow 0 - \text{dipib}$

Dividend 730321 \rightarrow ①

$(25)_{10} = (11001)_2$

Quotient	2	15	→ Remainder
divisor	2	12	1 → $(51)_{10}$
	2	6 0	
	2	3 0	
	2	1 1	
	2	0 1	

$$(25)_{10} = (11001)_2$$

$$\rightarrow (0.625)_{10} = (0.P)_2$$

② $0.625 \times 2 = 1.25 \rightarrow 1$
 $0.25 \times 2 = 0.5 \rightarrow 0$
 $0.5 \times 2 = 1.0 \rightarrow 1$

$$(0.625)_{10} = (0.101)_2$$

$$(25.625)_{10} = (11001.101)_2$$

Send off math 225 each the seas

* Octal to binary :-

$$(73)_8 = (1001)_2$$

* Decimal - octal. n = 9 1 5 8

$$(73)_{10} = (111)_8$$

$$\begin{array}{r} 8 | 73 \\ 8 | 9 \quad 1 \\ 8 | 1 \quad 1 \\ 0 \quad 1 \end{array}$$

$$(73)_{10} = (111)_8$$

$$*(73)_{10} = (111)_8$$

$$\begin{array}{r} 16 | 73 \\ 16 | 4 \quad 9 \\ 0 \quad 4 \end{array}$$

$$(73)_{10} = (49)_{16} = (111)_8$$

Note:- If base is greater, than number will be smaller; If base is lesser than number will be bigger.

Binary to Decimal.

MSB \rightarrow LSB - binary of 10100

$$(1001)_2 = ()_{10}$$
$$(1001)_2$$

3 2 1 0 = n - bits = binary

$$= 1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3$$
$$= 1 + 0 + 0 + 8$$
$$(1001)_2 = (9)_{10}$$

→ binary to decimal:

$$(10.101)_2 = ()_{10.8}$$

$$(10.101)_2$$

1 0 = n - 1 \times 2^{-1} \times 2^{-2} \times 2^{-3} \times 2^{-4} \times 2^{-5} \times 2^{-6} \times 2^{-7} \times 2^{-8}

$$= 0 \times 2^0 + 1 \times 2^1 + 1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3}$$
$$= 0 + 2 + 0.5 + 0 + 0.125$$

$$(10.101)_2 = (2.625)_{10}$$

✓ Binary to Octal (-1) \rightarrow octal (+)

say binary with 3 bits per 21 second 4F
so 2nd (101) $_2$ = 5 $_8$ (1 pm) 8 3 8 11 14

so 5 is 11 in binary so divide by 2
will reach the sea

$$\underbrace{100}_1 \quad \underbrace{101}_5 \quad \underbrace{010}_2 = (152)_8$$

Make group of 3 bit (for octal) (bcos using 3 bit you can represent any number in octal)

* Binary \rightarrow Hexadecimal

$$(10101011)_2 = (?)_{16}$$

Make group of 4 bit for hex-decimal (bcos using 4 bit you can represent any number in hexadecimal number system)

$$\underbrace{11010}_A \quad \underbrace{1011}_B = (AB)_{16}$$

* Convert $(4F)_{16} = (?)_8$

$$= 100 . 100 . 111 . 010 . 010$$

$$= (0100 \quad 1111)_2$$

$$= \underbrace{001}_1 \quad \underbrace{100}_1 \quad \underbrace{111}_7 = (117)_8$$

$$(6A)_{16} = ()_{10}$$

000 101 100

$$(6A)_{16} = ()_{10}$$

word (100 not up tide + word 100
upcongan not up tide 100

(100 not up tide 100

$$= A \times 16^0 + 6 \times 16^1$$

$$= 10 + 96 = (106)_{10}$$

* gate question

Q The octal equivalent of binary

$(AB \cdot CD)_{16}$

$$\therefore (A \cdot B \cdot C \cdot D)_{16}$$

$$(1010 \ 1011 \cdot 1100 \ 1101)_2$$

\leftrightarrow (1010 1011 1100 1101)

010 101 011 . 110 011 010

2 5 3 6 3 2

$$(253.632)_8$$

will reach the sea.

(Q) The binary representation of decimal number (1.375) is

$$(1)_{10} = ()_2$$

$$01 \text{ (23)}$$

$$01 \text{ (22)}$$

2	1		
	0	1	1

↓ E, round

$$= (1)_2$$

additive form of binary

$$01 \times 8 +$$

↓ add

$$(0.375)_{10} = ()_2$$

$$\begin{aligned} 0.375 \times 2 &= 0.75 && \xrightarrow{\text{top } 0.75} 01 < 1 \\ 0.75 \times 2 &= 1.5 && \xrightarrow{\text{top } 1.5} 1 \\ 0.5 \times 2 &= 1.0 && \xrightarrow{\text{top } 1.0} 01 \end{aligned}$$

$$(0.375)_{10} = (0.011)_2$$

$$\checkmark (1.375)_{10} = (1.011)_2$$

double verifying

$$(1.011)_2 = ()_{10}$$

$$(1.011)_2 = 1 \times 2^0 + 0 \times 2^1 + 1 \times 2^{-2}$$

$$0 \text{ (1-2-3)} + 1 \times 2^{-3}$$

$$= 1 + 0 + 0.25 + 0.125$$

$$= 1 + 0.375 = (1.375)_{10}$$

Arithmetic operations

Addition part 1

$$\begin{array}{r}
 (23)_{10} \\
 + (55)_{10} \\
 \hline
 78
 \end{array}
 \quad
 \begin{array}{r}
 21(282.1)_{53} \text{ decimal} \\
 + (1)_{53} \\
 \hline
 21(283)_{53}
 \end{array}$$

answer ↴

if $(8)_{10}$ = Valid decimal number
base No carry.

$$\begin{array}{r}
 11 \\
 68 \\
 + 46 \\
 \hline
 114
 \end{array}
 \quad
 \begin{array}{r}
 11(282.1)_{53} \\
 + (2)_{53} \\
 \hline
 11(283)_{53}
 \end{array}$$

14 > 10 It's not

base single digit valid decimal

Number so it written as

$$\begin{array}{r}
 14 = 10 \times 1 + 4 \\
 \text{Base } \xrightarrow{\uparrow} \text{Carry } \xrightarrow{\uparrow} \text{Result}
 \end{array}$$

Base x Carry + Result

$$11 = 10 \times 1 + 1$$

11 = 10 + 1

$$\begin{array}{r}
 11.0 + 282.1 \\
 \hline
 293.1
 \end{array}$$

Addition using Binary

$$+ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}_2 \xrightarrow{\text{add 1st row to 2nd}} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}_2$$

$\xrightarrow{\text{add 2nd row to 3rd}}$

2 < base (2)

2 is not valid binary ~~num~~-digit

$$2 = 2 \underset{\text{Base}}{\overset{I}{\times}} 1 + \underset{\text{Carry}}{\overset{0}{\downarrow}} \underset{\text{Result}}{\overset{I}{+}}$$

Octal

$$\begin{array}{r}
 67 \\
 + 2 \\
 \hline
 71
 \end{array}
 \quad
 \begin{array}{r}
 7+2=9 \\
 \times \text{Valid}
 \end{array}$$

Hexadecimal

$$\begin{array}{r}
 2F \\
 + A3A \\
 \hline
 D2B
 \end{array}$$

$$F + 3 = 15 + 3 = 18 \quad \text{X} \quad \text{valid}$$

$$18 = 16x + 2$$

$$add (63)_7 + (56)_8$$

will give different → can't add directly

So we have to convert both into some number system other than base 10 (0, 1, 2, 3, 4, 5, 6, 7)

$$(56)_8 = b(1 \times 8^1) + (1 \times 8^0) = (1 \times 8^1) + (1 \times 8^0)$$

$$(56)_8 = 6 \times 8^0 + 1 \times 8^1 = 6 + 1 \times 8 = 14$$

$$6 \times 8^0 + 5 \times 8^1 =$$

$$= 6 + 40$$

$$= (46)_{10}$$

$$\begin{array}{r} 7 \\ \overline{)46} \\ 4 \\ \hline 6 \\ 4 \\ \hline 0 \\ 6 \end{array}$$

$$= (64)_7$$

$$\boxed{\begin{array}{r} 1 \\ \overline{)64} \\ 6 \\ \hline 4 \\ 4 \\ \hline 0 \end{array}}$$

$$\boxed{\text{Ans}} = (160)_7$$

$$\begin{array}{r} 1 \\ 7 \\ 1 \\ 4 \\ 8 \\ 8 \\ 0 \\ 1 \\ 3 \\ 7 \\ 1 \\ 0 \end{array}$$

$$7 = 7 \times 1 + 0$$

$$13 = 7 \times 1 + 6$$

Subtraction Examples

(13) ^{base}
10+2

(29) ^{base}
10+0

$$\begin{array}{r} 56 \\ - 23 \\ \hline 33 \end{array}$$

note: 52 ⁰ 4

$$\begin{array}{r} -123 \\ 29 \end{array}$$

positive 69.4 H

$$\begin{array}{r} -1.45 \\ 19.5 \end{array}$$

7 ^{base}
2+0

$$\begin{array}{r} 3 * \\ 23 \end{array}$$

Binary

110 00

$$\begin{array}{r} -101021 \\ 001021 \end{array}$$

Octal :-

$$\begin{array}{r} 8+2 \\ 62 \\ -45 \\ \hline 15 \end{array}$$

8

56

101

$$\begin{array}{r} -2x5 \\ 33 \end{array}$$

11 2

101
0101

11 1

Hexadecimal :-

$$\begin{array}{r} 1F + 0 \times 8 = 2 \\ 1C + 1 \times 8 = 21 \\ \hline 0B \end{array}$$

10+00

0F

12*

1F

X 0 + 1
2FF+

~~Q~~ $(15)_7 - (26)_7 + (33)_7$

Wrong explanations

* Multiplication No system

$$\begin{array}{r}
 \overset{28}{\cancel{2}} \\
 211 \\
 \times 6 \\
 \hline
 66 \\
 + 36 \\
 \hline
 30 \\
 + 150 \\
 \hline
 180 + 30 \\
 \hline
 180
 \end{array}$$

* Binary:-

$$\begin{array}{r}
 101 \\
 \times 11 \\
 \hline
 101 \\
 1010 \\
 \hline
 1111
 \end{array}$$

* Octal :-

$$\begin{array}{r}
 74 \\
 \times 21 \\
 \hline
 74 \\
 148 \\
 \hline
 170 \\
 \hline
 174
 \end{array}$$

$$8 = 8 \times 1 + 0$$

$$15 = 8 \times 1 + 7$$

Hexadecimal :- (1-Second)

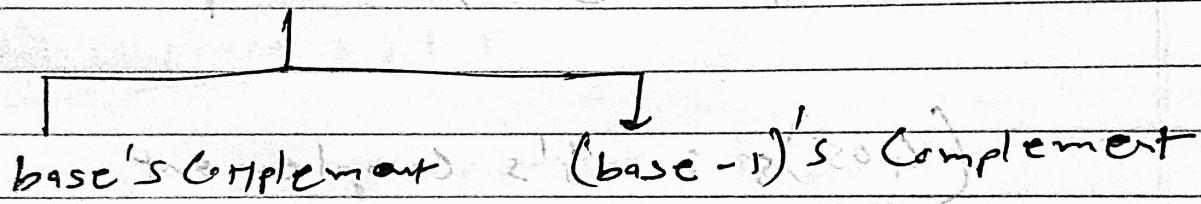
Hexagonal LFS \leftrightarrow (2nd)

$$\begin{array}{r} AB \\ \times 16 \\ \hline 22 F F F \\ \hline 156 \times 16 \\ \hline 16 B 6 \end{array}$$

Hexagonal 2nd (o)

Complements :- (1-Second)

Hexagonal L(1-Second) to Hexagon



ex. L + Hexagonal 2¹⁶ =

binary

= 2's complement 1's

Desired 9's complement

How to find complement :- (1-Second)

(1) (base-1)'s complement

Subtracting each digit from the largest digit in the number system.

(base-1)'s Complement.

~~ex~~ $(405)_8 \rightarrow 7^1$'s Complement

$$\begin{array}{r}
 2+1 \times 2^1 \\
 2+1 \times 2^1 \quad 777\ldots \\
 \hline
 405 \\
 \hline
 372
 \end{array}$$

2) base's complement:-

by Simply adding 1 to the result of (base-1)'s complement

$(405)_8$ - 7^1's complement

$$= 7^1\text{'s complement} + 1$$

$$\begin{array}{r}
 2 = 372 \\
 + 1 \\
 \hline
 373
 \end{array}$$

* Calculate 3^1's complement for $(301)_4$

(Base-1)'s complement

$$\cancel{4^1\text{'s complement}} \quad 333$$

$$3^1\text{'s complement} \quad 301$$

$$\begin{array}{r}
 301 \\
 + 1 \\
 \hline
 032
 \end{array}$$

~~3^1 complement~~

* Calculate 2's complement for

$(1001)_2$ 1 0 0 1

$$\begin{array}{r} \text{1's complement} \\ \hline 1 1 1 1 \\ \hline 1 0 0 1 \\ \hline 0 1 1 0 \end{array}$$

↓ ↓ ↓
↑ ↑ ↑
1's complement

+ ← ←
1 0 1 0

$$\begin{array}{r} 0 1 1 0 \\ + \quad \quad \quad 1 \\ \hline 0 1 1 1 \end{array}$$

← ← ←
1's complement

direct 1's complement & 2's complement

1001 \Rightarrow Reverse all bit
1's complement \rightarrow 0110

1 0 0 1 \Rightarrow
↓
LSB

Start from LSB
Look for first "1"
keep that "1" as it is, ~~go~~ after first "1" reverse all bit

2's Complement \leftarrow ^{first} 1

1	0	1	0
↓	↓	↓	↓
0	1	1	0

↓ ↓ ↓ ↓
0 1 1 0
↓ ↓
out

~~ex~~

Find
2

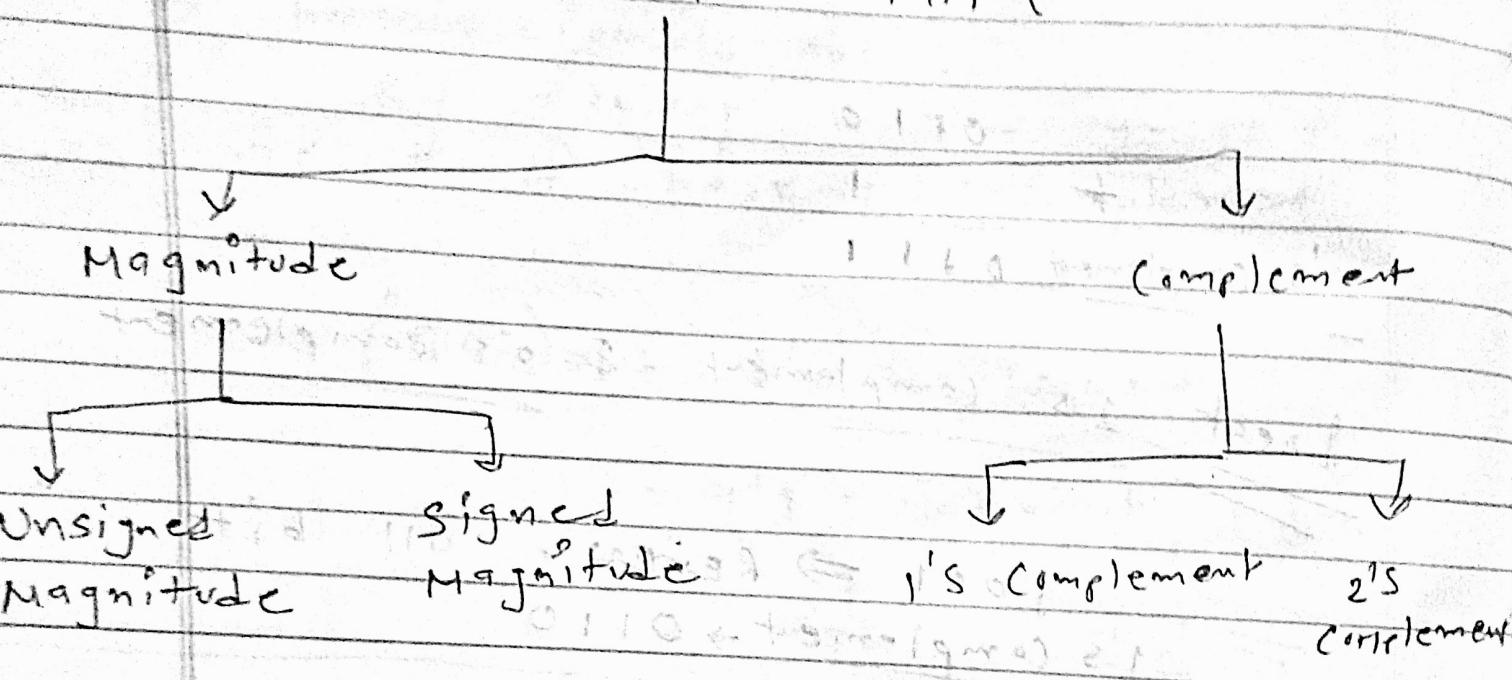
1 0 0 1 (1001)

↓ ↓ ↓ ↓
0 1 1111
1001

2's complement



Number Representation



→ we can → Both +ve → Both → Both
 represent & -ve +ve & -ve +ve &
 only +ve -ve +ve &
 number

→ +6 → +6 → +6 → +6
 = 110 → ~~0110~~ → 0110 → 0110
 = '0' 110
 ↓
 +ve

representation of positive number are same in all case

In signed
magnitude

-6

X

Signed
magnitude

$\rightarrow +6$

'1110

F+

0-

1-

2-

3-

4-

5-

6-

7-

8-

9-

10-

11-

12-

13-

14-

15-

16-

17-

18-

19-

20-

21-

22-

23-

24-

25-

26-

27-

28-

29-

30-

31-

1's complement

$\rightarrow -6$

'0'110 $\Rightarrow +6$

\downarrow 1's complement

1 001 $\Rightarrow -6$

\downarrow 1's complement

0 110

$\rightarrow 6$ is
magnitude

This is representation
of -6 in 1's complement

complement form of -6 in 1's
complement

2's complement

$\rightarrow -6$

'0'110 $\Rightarrow +6$

\downarrow 2's complement

1 010 $\Rightarrow -6$

\downarrow 2's complement

0 110

$\rightarrow 6$ is
magnitude

This is negative
representation of -6 in 2's
complement

+6 in 6bit

'000110

+6 in 6bit

'000110

+6 in 6bit

.

-6 in 6bit

'111000

'100110

-6 in 6bit

'111000

-6 in 6bit

.

~~111000~~ $\Rightarrow +6$

\downarrow 1's complement

111000 $\Rightarrow +6$

\downarrow 1's complement

F-

2-

1-

0-

Copy MSB & -6
representation of
1's complement

Copy MSB & -6
representation of
2's complement

Copy MSB & -6
representation of
1's complement

Copy MSB & -6
representation of
2's complement

Copy MSB & -6
representation of
1's complement

Copy MSB & -6
representation of
2's complement

on.

Copy MSB & -6
representation of
1's complement

Copy MSB & -6
representation of
2's complement

Copy MSB & -6
representation of
1's complement

Copy MSB & -6
representation of
2's complement

below to 6bit

Signed Magnitude

0 0 0 0
0 0 0 1 ←
0 1 1 1 0 0 0 0
0 1 1 1 1 1 1 1

1 0 0 0 1 0 0 1

1 1 1 0 0 0

1 1 1 0 1 1

Decimal

+0
+1 ←
+6
+7
-0
-1
-6
-7

short form

disadvantages - it

→ 2 different representation for 0

→ Range

-2^{n-1} to 2^{n-1} (n bit)

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

-7 to +7

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

1's complement

Decimal

0 0 0 0 0 0 0 0

+0 1 1 1 1 1 1 1

0 0 0 1 0 1 1 1

+1

0 1 1 0 1 0 0 0

+6

0 1 1 1 0 0 0 0

+7

1 0 0 0 0 0 0 0

-7

1 0 0 0 1 1 1 1

-6

1 1 1 1 0 0 0 0

-1

0 1 0 1 1 1 1 1

-0

disadvantages:

→ 2 different representations for 0

→ ambiguity

$0 = 0 + 0$

$0 = -(-2^{n-1} + 1)$

$0 = -2^{n-1} + 2^n$

00000000

00100000

46811

$+(-2^{n-1} - 1) \in \beta^-$

$-2^{n-1} + 2^n$

$n \text{ bits } -(-2^{n-1} - 1) \rightarrow +(-2^{n-1} - 1) \text{ max } 2^{\lfloor n/2 \rfloor}$

2's complement

$00000000 +$

00000001

0110

0111

1000

1001

1110

decimal = 2⁷

$+00000000$

$+10000000$

$+00000001$

$+7$

-8

-7

-2

01110000

min value = -2^{n-1} max value = $2^{\lfloor n/2 \rfloor}$

→ min = -2^{n-1} max = $+2^{\lfloor n/2 \rfloor}$ 4 bit

$00000000 \rightarrow -(-2^{n-1}) \rightarrow +(-2^{n-1} - 1)$

$n \text{ bits } -(-2^{n-1}) \rightarrow +(-2^{n-1} - 1)$

000100

100101

111001

111000

min value = 0

max value = $2^{\lfloor n/2 \rfloor}$

Arithmetic operation using Complement

Q) $5 - 4 = ?$

1's complement

$$+5 \Rightarrow 0101$$

$$+4 \Rightarrow 0100$$

$$-4 \Rightarrow (+0100) + (+1111)$$

$$\begin{array}{r} 0101 \\ + 1011 \\ \hline 10000 \end{array}$$

$$\begin{array}{r} 1111 \\ + 1011 \\ \hline 0000 \end{array}$$

$$\begin{array}{r} 0001 \\ \boxed{1} \\ \hline 0001 \end{array} \checkmark +1$$

8's complement ($+1$) + (-4) . This is in 1's complement form

Q) $5 - 4$

$$+5 = 010100$$

$$+4 = 0100$$

$$-4 = 1100$$

$$\begin{array}{r} 010100 \\ + 1100 \\ \hline 010000 \end{array} 8's$$

$$\begin{array}{r} 1100 \\ + 1100 \\ \hline 10000 \end{array} 4$$

$$\begin{array}{r} 10001 \\ \boxed{1} \\ \hline 0001 \end{array} \checkmark +1$$

↳ This result is in 2's complement form

discard it.

gate $x = 01110$ & $y = 11001$ are 2 bit binary number represented in 2's complement form. The sum of $x + y$ represented in 2's complement format using 6-bit is

④ 100111

⑤ 001000

⑥ 000111

⑦ 101001

$$X = 01110 \leftarrow +14$$

$$Y = 11001 \leftarrow -7$$

$$\begin{array}{r} 00110 \\ -11001 \\ \hline 11111 \end{array}$$

$$\begin{array}{r} 14 \\ -7 \\ \hline 7 \end{array}$$

in 6 bit, 2's complement

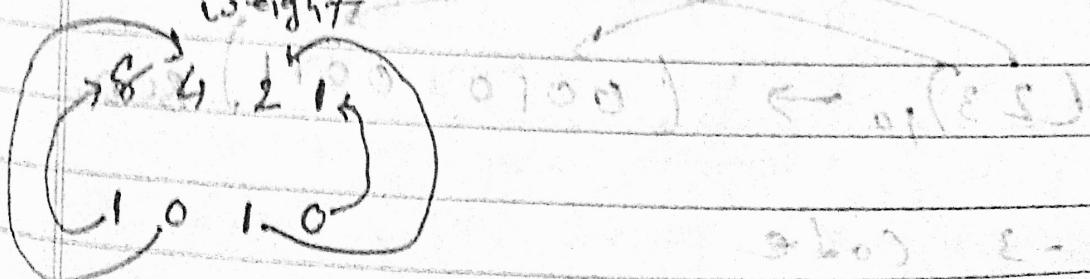
norm

100011

Lecture:-3

Weighted Codes

Weighted Codes (with weights) \rightarrow Non-weighted Code



\rightarrow BCD $8 + 0011 \leftarrow 8 \rightarrow$ Excess -3
 \rightarrow Gray Code

BCD is always 4 bit

BCD

decimal	B ₃	B ₂	B ₁	B ₀	BCD
0	0	0	0	0	0000
1	0	0	0	1	0001
2	0	0	1	0	0010
3	0	0	1	1	0011
4	0	1	0	1	0101
5	0	1	1	0	0110
6	0	1	1	1	0111
7	0	1	1	1	0111
8	1	0	0	0	1000
9	1	0	0	1	1001
10					
11					
12					
13					
14					
15					

Invalid BCD code

to represent greater numbers

$$(23)_{10} \rightarrow (0010\ 0011)_{BCD}$$

Excess - 3 code

$$\text{Excess - 3} = \text{BCD} + 3$$

Digit	B ₃	B ₂	B ₁	B ₀	E ₃	E ₂	E ₁	E ₀
0	0	0	0	0	0	0	1	1
1	0	0	1	0	0	1	0	0
2	0	1	0	0	0	0	1	0
3	0	1	0	1	1	0	0	0
4	0	0	1	0	0	1	1	1
5	0	0	1	1	1	1	0	0
6	0	1	1	0	1	1	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	1	0	0	1	1	0	0

It's also called self Complementing code.

→ Gray Code :- (Cyclic code all unit distance code)

	B ₃	B ₂	B ₁	B ₀	G ₃	G ₂	G ₁	G ₀
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	0
4	0	0	0	0	0	0	1	0
5	0	0	0	1	0	1	0	0
6	0	1	1	0	0	1	0	1
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	0	0	0
9	1	0	0	1	1	0	0	1
10	1	0	1	0	1	1	0	1
11	1	0	1	1	1	1	1	0
12	1	1	0	0	1	0	1	0
13	1	1	0	1	1	0	1	1
14	1	1	1	0	1	0	0	1
15	1	0	1	1	1	0	0	0

BCD Addition

23	00000	000110
+15	00011	010100
39	00110	101010

$$\begin{array}{r} 0 \quad 0 \quad 0 \quad | \quad 1 \quad 1 \quad 0 \quad | \quad 1 \\ 1 \quad 0 \quad 0 \quad | \quad 3 \quad 0 \quad 1 \quad | \quad 19 \quad 0 \end{array}$$

0 9 9 1 8 . - 1 1 1 1 1 3

100100011000011

$$110 \Rightarrow 01100011$$

5 0 | 0 | 0 | 0

~~Both
Invalid
BCD Code~~

~~121~~ 45018-19

to make it's too good to be true as
valid odds 0.0 10 10 10 10

$$10110 = +6 \quad 10110 - 10110 = 0$$

each Invalid 0001 0 0 1 0 0001

BCD number: ↓ ↓ ↓ ↓

BCD Number: ↓ ↓ ↓ ↓

1 9 1

1 1 1 2 1

00	-	-	-	-	-	-	-	-
----	---	---	---	---	---	---	---	---

— 0 — 0 — 0 — 0 — 0 — 0 — 0 — 0 —

— 3 —

1 2 3 4 5 6 7 8 9 10

Convert binary to gray code

(e)

$$\begin{array}{ccccccc} & \oplus & \oplus & \oplus & & & \\ 1 & 0 & 1 & 1 & 0 & & \\ \downarrow & & & & & & \\ (1011)_2 = (1110)_G \end{array}$$

(e)

$$\begin{array}{ccccc} & \oplus & \oplus & \oplus & \\ 1 & 1 & 1 & 1 & 0 \\ \downarrow & & & & \\ 1 & 0 & 0 & 0 & \\ & & & & \end{array} \quad (1111)_2 = (1000)_G$$

first ~~do~~ MSB as it is
then ~~do~~ do X-or operation of 1st
and 2nd bit (from MSB side)
result will be your 2nd bit
of gray code .. same process

a

$$\begin{array}{ccccc} & \oplus & \oplus & \oplus & \\ 0 & 1 & 0 & 0 & \\ \downarrow & & & & \\ 0 & 1 & 0 & 0 & \end{array}$$

$$(0100)_2 \rightarrow (0110)_G$$