Logic

High (logic 1)
low (logic 0)

→ positive logic : logic 0 (low) = 0 V
logic 1 (high) = +5 V

→ Negative logic : logic 0 = +5 V
logic 1 = 0 V

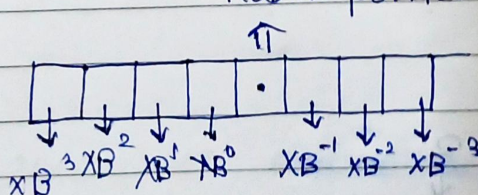
⇒ largest digit = (Base - 1)

	Number system	Base
1	Decimal	10
2	Binary	2
3	Octal	8
4	Hexadecimal	16

Number system

1. Positional
2. Non-positional
   - → Grey code
   - → Excess-3 code (XS-3)

Radix point.



Conversion related to decimal system

4 Conversion from any N.S to decimal system

① $(1011.01)_2$

1×2^3, 0×2^2, 1×2^1, 1×2^0, 0×2^{-1}, 1×2^{-2}

$8+2+1.25$

11.25

② $(314)_8$

3×8^2, 1×8^1, 4×8^0

$192+8+4$

$(204)_{10}$

③ $(365.24)_8$

3 6 5 . 2 4

3×8^2 6×8^1 5×8^0 2×8^{-1} 4×8^{-2}

$192+48+5.$ $0.\cancel{25}\,3125$

245.3125

④ $(4C8.2)_{16}$

4 C 8 . 2

4×16^2 12×16^1 8×16^0 2×16^{-1}

$1024+192+8.125 = (1224.125)_{10}$

* conversion of decimal to any other.

① $(105)_{10}$ into binary.

2	105	
2	52	1
2	26	0
2	13	0
2	6	1
2	3	0
	1.	1
		1

⇒ $(1101001)_2$

Steps.

① Divide integer part of given decimal system by the base of desired no. system and write remainder.

② Keep dividing the quotient by base until nothing is left.

③ Write down the remainder in reverse order

④ An it will give you equivalent number.

② $(204)_{10}$

$$\begin{array}{r|l l}
8 & 204 & \\
\hline
8 & 25 & 4 \\
\hline
 & 3 & 1 \\
 & & 3
\end{array} \uparrow \rightarrow (314)_8$$

③ $(259)_{10}$

$$\begin{array}{r|l l}
16 & 259 & \\
\hline
16 & 16 & 3 \\
\hline
 & 1 & 0 \\
 & & 1
\end{array} \uparrow \quad (103)_{16}$$

- Successive Multiplication of fractional part.

1. Multiply the given fractional decimal number by the base of desired number system.
2. Record the carry as MSB
3. Multiply only fractional part of result obtained from step 2 by base & record the carry as next bit to MSB
4. Repeat steps 2 & 3 till the end & last carry will correspond to LSB

- Usually the conversion for fractional part is carried out only upto 5 digits

① $(0.42)_{10}$ into binary.

		carry	
0.42×2	$= \boxed{0}.84$	0	MSB
0.84×2	$= \boxed{1}.68$	1	
0.68×2	$= \boxed{1}.36$	1	
0.36×2	$= \boxed{0}.72$	0	
0.72×2	$= \boxed{1}.44$	1	LSB

Ans $= (0.01101)_2$

② $(0.8)_{10}$ into binary

0.8×2	$= \boxed{1}.6$	1
0.6×2	$= \boxed{1}.2$	1
0.2×2	$= \boxed{0}.4$	0
0.4×2	$= \boxed{0}.8$	0
0.8×2	$= \boxed{1}.6$	1

Ans: $(0.11001)_2$

③ $(0.6234)_{10} \rightarrow$ octal

0.6234×8	$= \boxed{4}.9872$	4
0.9872×8	$= \boxed{7}.8976$	7
0.8976×8	$= \boxed{7}.1808$	7
0.1808×8	$= \boxed{1}.4464$	1
0.4464×8	$= \boxed{3}.5712$	3

Ans $= (0.47713)_8$

9 10 11 12 1

Page No.

Date: / /

4 $(0.122)_{10}$

$\rightarrow$

$0.122 \times 16 = \boxed{1}.952$ 1

$0.952 \times 16 = \boxed{15}.232$ 15

$0.232 \times 16 = \boxed{3}.712$ 3

$0.712 \times 16 = \boxed{11}.392$ 11

$0.392 \times 16 = \boxed{6}.272$ 6

Ans $= (0.1F3B6)_{16}$

5 $(85.63)_{10} \rightarrow$ binary.

$0.63 \times 2 = 1.26 \rightarrow 1$

$0.26 \times 2 = 0.52 \rightarrow 0$

$0.52 \times 2 = 1.04 \rightarrow 1$

$0.04 \times 2 = 0.08 \rightarrow 0$

$0.08 \times 2 = 0.16 \rightarrow 0$

2	85	
2	42	1
2	21	0
2	10	1
2	5	0
2	2	1
	1	0
		1

$(1010101.10100)_2$

② $(3000.45)_{10}$ to Octal.

0.45	× 8	→ 3.6	3	MSB
0.6	× 8	→ 4.8	4	
0.8	× 8	→ 6.4	6	
0.4	× 8	→ 3.2	3	
0.2	× 8	→ 1.6	1	LSB

8	3000	
8	375	0
8	46	7
8	5	6
		5

Ans = $(5670.34631)_8$

③ $(2003.31)_{10}$ → hexa

0.31 × 16	→ 4.96	→ 4	MSD
0.96 × 16	→ 15.36	→ 15	
0.36 × 16	→ 5.76	→ 5	
0.76 × 16	→ 12.16	→ 12	
0.16 × 16	→ 2.56	→ 2	LSD

16	2003	
16	125	3
~~16~~	7	13
		7

$(7D3.4F5C2)_{16}$

Conversion from binary to other systems.

Binary to octal.

① $(11010010)_2$

0	0	0	→ 0
0	0	1	→ 1
0	1	0	→ 2
0	1	1	→ 3
1	0	0	→ 4
1	0	1	→ 5
1	1	0	→ 6
1	1	1	→ 7

→ $\underbrace{011}_{3}\ \underbrace{010}_{2}\ \underbrace{010}_{2}$

Ans = $(322)_8$

* Group the bits into group of 3 LSB.
* Always add extra zero's on MSB side and not on LSB side.

Binary to hex

② $(11010010)_2$

$\underbrace{1101}_{13}\ \underbrace{0010}_{2}$ → $(D2)_{16}$

$(\underbrace{1010}_{A}\underbrace{1111}_{F}\underbrace{1011}_{B}\underbrace{0010}_{2})_2$

$(AFB2)_{16}$

Page No.

Date: / /

conversion from any N.S to binary.

① octal to binary

$(364)_8$

$(011110100)_2$

② Hex to Binary

$(AFB2)_{16}$

$(1010111110110010)_2$

+ octal → hex

$(436)_8$

$00(0100011110)_2$

$(11E)_{16}$

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

→ Hex to octal

① $(46A)_{16}$

~~0100 0110 1010~~ 0100 1100 1010

~~2 1 5 2~~ $(2\ 3\ 1\ 2)_8$

~~$(2152)_8$~~

② $(0.12E)_{16}$

$0.\underbrace{000}\underbrace{100}\underbrace{101}\underbrace{110}$

0 4 5 6

$(0.0456)_8$

③ $(68.4B)_{16}$

0 will be added to MSB side (integer part)

0 added after LSB in frac. part

$\underbrace{001}\underbrace{101}\underbrace{000}.\underbrace{010}\underbrace{010}\underbrace{110}$

1 5 0 . 2 2 6

$(150.226)_8$

Date:

* convert following decimal no to binary, octal, hexadecimal.

① $(25.55)_{10}$
② $(456)_{10}$

* convert hex → decimal, binary & octal
① 48 ② 2E.

* convert following number into decimal
① $(10110.0101)_2$ ② $(16.5)_{16}$

* Represent decimal numbers in binary, octal and hexadecimal.
① $(8.96)_{10}$ ② $(4096)_{10}$

* convert following octal no to hexa, binary, decimal.
① $(76)_8$ ② $(0.7634)_8$ ③ $(1567)_8$ ④ $(65.04)_8$

Q.1)

① $(25.55)_{10}$

2	25	
2	12	1
2	6	0
2	3	0
	1	1
		1

$0.55 \times 2 = 1.1 = 1$
$0.1 \times 2 = 0.2 = 0$
$0.2 \times 2 = 0.4 = 0$
$0.4 \times 2 = 0.8 = 0$
$0.8 \times 2 = 1.6 = 1$

$(\underbrace{011}_{3}\,\underbrace{001}_{1}.\underbrace{100}_{4}\,\underbrace{010}_{2})_2$

$(31.42)_8$

$\underbrace{0001}\underbrace{1001}.\underbrace{1000}\underbrace{1000}$

1 9 . 8 8

$(19.88)_{16}$

② $(456)_{10}$

2	456	
2	228	0
2	114	0
2	57	0
2	28	1
2	14	0
2	7	0
2	3	1
	1	1
		1

$(\underbrace{111}\underbrace{001}\underbrace{000})_2$

$(710)_8$

$(1C8)_{16}$

Page No.
Date: / /

① 4 A

4 1 1

2	411	
2	205	1
2	102	1
2	51	0
2	25	1
2	12	1
2	6	0
2	3	0
	1	1
		1

$4 \times 16^1 + 11 \times 16^0$

7 5

$(75)_{10}$

$(\underbrace{1\,1\,0}_{6}\,\underbrace{0\,1\,1}_{3}\,\underbrace{0\,1\,1}_{3})_2$

6 3 3

$(633)_8$

1 1 0 0 1 1 0 1 1

$2^8\ 2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

$(639)_{10}$

② 2 E

2	214	
2	107	0
2	53	1
2	26	1
2	13	0
2	6	1
2	3	0
	1	1

$(\underbrace{0\,1\,1}_{3}\,\underbrace{0\,1\,0}_{2}\,\underbrace{1\,1\,0}_{6})_2$

3 2 6

$(326)_8$

decimal

$2 \times 16^1 + 14 \times 16^0$

$(46)_{10}$

① $(101100101)_2$

→

1	0	1	1	0	0	1	0	1
2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

256 + 32 + 64 + 4 + 1

$(357)_{10}$

② $(16.5)_{16}$

1 6 . 5

1×16^1 6×16^0 5×16^{-1}

16 + 6. 3125

22.3125

Q. ① $(3.96)_{10}$

2	3	
	1	1
		1

$0.96 \times 2 = 1.92 \to 1$

$0.92 \times 2 = 1.84 \to 1$

$0.84 \times 2 = 1.68 \to 1$

$0.68 \times 2 = 1.36 \to 1$

$0.36 \times 2 = 0.72 \to 0$

Binary ~~(11)~~

$(11.11110)_2$

Octatal

$(3.74)_8$

hexa

$(3.F0)_{16}$

② $(4096)_{10}$

2	4096	
2	2048	0
2	1024	0
2	512	0
2	256	0
2	128	0
2	64	0
2	32	0
2	16	0
2	8	0
2	4	0
2	2	0
	1	0
		1

$(1000000000000)_{10}$

$(10000)_8$

$(1000)_{16}$

① $(76)_8$

$(111\,110)_2$

$(3E)_{16}$

$$\begin{matrix} 7 & 6 \\ \downarrow & \downarrow \\ 7\times8^1 & 6\times8^0 \end{matrix}$$

$\cancel{(54)_{10}}$

$(62)_{10}$

Page No.

Date: / /

② $(1567)_8$

$(001101110111)_2$

$(1567)_8$

$8^3 \quad 8^2 \quad 8^1 \quad 8^0$

$512 + 320 + 48 + 7 = (887)_{10}$

$(377)_{16}$

③ $(65.04)_8$

$(110101.000100)_2$

$(35.10)_{16}$

$6 \quad 5 \quad . \quad 0 \quad 4$

$6\times8^1 \quad 5\times8^0 \quad 4\times8^{-2}$

$48+5.0625$

$(53.0625)_{10}$

② $(0.7634)_8$

$(0.\underbrace{111}\underbrace{110}\ \underbrace{011}\ \underbrace{100})_2$

$(0.F9C)_{16}$

$(0.\ 7\ \ 6\ \ 3\ \ 4)$

$8^{-1}\ \ 8^{-2}\ \ 8^{-3}\ \ 8^{-4}$

~~0.~~ $0.875 + 0.09375 + 0.00585 + 0.000976$

$(\cancel{0.}\ \cancel{0.975})_{10}$

$(0.0975)_{10}$

→ Binary Addition.

	carry	sum
0 + 0	0	0
0 + 1	0	1
1 + 0	0	1
1 + 1	1	0

A $(10111)_2$ B = $(11001)_2$

```
    1 1 1 1
    1 0 1 1 1
  + 1 1 0 0 1
  -----------
 (1 1 0 0 0 0)2
```

Binary Multiplication

$0 \times 0 = 0$
$0 \times 1 = 0$
$1 \times 0 = 0$
$1 \times 1 = 1$

$(9)_{10} \times (8)_{10}$

$(1001) \times (1000)$

```
  1 0 0 1
x 1 0 0 0
---------
```

```
        1 0 0 1
      x 1 0 0 0
      ---------
        0 0 0 0
      0 0 0 0 x
    0 0 0 0 x
  1 0 0 1 x
  ---------------
  1 0 0 1 0 0 0
```

Page No.
Date: / /

Q. Binary subtraction:

		subtract	borrow
1	0 - 0	0	0
	0 - 1	1	1
	1 - 0	1	0
	1 - 1	0	0

$A = (11011)_2$ and $B = (10110)_2$

```
  1 1 0 1 1
- 1 0 1 1 0
-----------
  0 0 1 0 1
```

Q 38 - 29

38 = 100110

29 = 11101

38 - 29

```
    1 0 0 1 1 0
-     1 1 1 0 1
---------------
  0 0 1 0 0 1
```

$2^4 \; 2^3 \; 2^2 \; 2^1 \; 2^0$

2	38	
2	19	0
2	9	1
2	4	1
2	2	0
	1	0
		1

2	29	
2	14	1
2	7	0
2	3	1
	1	1
		1

Page No.
Date: / /

* Binary division

$(110)_2 \div (10)_2 = (11)_2$

$6 \div 2 = 3$

- Unsigned binary number:

smallest 8 bit unsigned no.	0000 0000	$(00)_H$	$(00)_{10}$
largest 8 bit unsigned no	1111 1111	$(FF)_H$	$(255)_{10}$

for 16 bit

$(0)_{10}$
$(65535)_{10}$

- Signed binary number:

for 8 bit
+ve signed no = 0
-ve signed no = 1

sign bit

$\boxed{0}\,1111111 \quad +(127)_{10}$

$\underbrace{1}_{\text{sign bit}}\,\underbrace{1111111}_{\text{magnitude}} \quad -(127)_{10}$

for 16 bit
$-(32767)_{10}$ to $(32767)_{10}$

Complements.

1. Radix complemented form. (r's complement form)
2. Diminished radix complemented form (r-1)'s complemented form.

- Binary	1's	2's
- hexadecimal	15's	16's
- octal	8's	7's
- Decimal	10's	9's

1's complement.

- One's complement of +ve no have same representation as signed no +5 ⇒ 0101
- One's complement of -ve no by inverting +ve no i.e -5 ⇒ 1010

Range.

$+(\cancel{2^m}-1)$ to $\cancel{-(2^m-1)}$ for 4-bit $+(2^{m-1}-1) \to -(2^{m-1}-1)$

Dec. No	+ve Binary eq.	-ve 1's complement	
+7	0111	1000	-7
+6	0110	1001	-6
+5	0101	1010	-5
+4	0100	1011	-4
+3	0011	1100	-3
+2	0010	1101	-2
+1	0001	1110	-1
+0	0000	1111	-0

drawback of 1's c it have +0 & -0

For +ve no signed magnitude form, one's complement 2's complement are all the same.

2's C = 1's C + (LSB)

ex- -6

+6 = 0110

1's C = 1001 (-6)

+ $\boxed{1}$

1010 - 2's C [-6]

	~~signed magnitude~~ 1's c.	2's c	signed magni.
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
+4	0100	0100	0100
+3	0011	0011	0011
+2	0010	0010	0010
+1	0001	0001	0001
+0	0000	0000	0000
-1	1110	1111	1001
-2	1101	1110	1010
-3	1100	1101	1011
-4	1011	1100	~~1011~~ 1011
-5	1010	1011	~~101~~ 1100
-6	1001	1010	1101
-7	1000	1001	1110
-8 - 8 - 8	0111 ~~1111~~	1000 ~~0000~~	1111

8 = 1000 -18

$+5$ $\boxed{0101}$

-5 1010

$+1$

(-5) 1011

- Binary addition.

Case-I ~~sam~~ Smaller no is -ve

$A = -7$

$B = 12 = 00001100$

$+7 = 00000111$

$-7 = 11111000$

$+ \qquad 1$

$11111001 \rightarrow 2$'s C $+7$

$+ \; 00001100$

$\therefore$ $12-7$.

$\boxed{\cancel{1}}\,00000101 = 5$

- Binary subtraction by 2's complement

$A - B \Rightarrow A + (-B)$

1. Add A to 2's complement of B
2. If carry is generated then result is positive in true form. If not generated then result is negative & in 2's complement form

(1) $(9)_{10} - (5)_{10}$

1001 - (101)

0101 → 1's = 1010

```
  1001
+ 1010
------
[1]0011
+     1
------
  0100
```

= +4 = $(0100)_2$

```
2 | 9
2 | 4   1
2 | 2   0
    1   0
        1
```

```
2 | 5
2 | 2   1
    1   0
        1
```

(2) $(-48)_{10}$ and $(-23)_{10}$ using 8 bit representation

48 = 00110000   11001111

1's 48 = ~~11000111 + 2's =~~

23 = 00010111

1's 23 = 11101000

```
2 | 48
2 | 24   0
2 | 12   0
2 |  6   0
2 |  3   1
2 |  1   1
         1
```

```
  000111          01000
+      1        +     1
--------        -------
  001000          01001
```

```
  001000
+ 001001
--------
  010001
```

```
2 | 23
2 | 11   1
2 |  5   1
2 |  2   1
     1   0
         1
```

```
  11001111        11101000
+        1      +        1
----------      ----------
  11010000   +    11101001.
```

Page No.

Date: / /

$$
\begin{array}{r}
1\,1\,0\,0\,0\,0\,0\,0 \\
+\;1\,1\,1\,0\,1\,0\,0\,1 \\
\hline
\cancel{1\,0\,1\,1\,0\,0\,0\,1}
\end{array}
$$

$$
\begin{array}{r}
\boxed{1}\,1\,0\,1\,1\,1\,0\,0\,1 \\
-\qquad\qquad\quad 1 \\
\hline
1\,0\,1\,1\,1\,0\,0\,0
\end{array}
$$

→ 1's c. → 1's c

↓ inverting.

$\boxed{1}$ 1000111 → Final ans. $(-71)_{10}$

↓

-ve sign

$\boxed{1}$ To perform octal addition. ~~add~~

→

① Add octal numbers considering them as decimal numbers.

② If the sum is greater than or equal to 8, subtract 8 from it and generate carry of 1

ex- $(2)_8 + (4)_8$

$(2)_{10} + (4)_{10} = (6)_{10}$

$\underbrace{(7)_8 + (4)_8}_{\text{in decimal}} = (13)_8$

in decimal

(11) → greater than 8

$$
\begin{array}{r}
(11) \\
-\;8 \\
\hline
\boxed{1}\;3
\end{array}
$$

$\Rightarrow (7)_8 + (4)_8 = (13)_8$

↓

carry

Q. $(765)_8 + (365)_8 = (1352)_8$

$$
\begin{array}{cccc}
 & (7)_{10} & (6)_{10} & (5)_{10} \\
+ & (3)_{10} & (6)_{10} & (5)_{10} \\
 & {}^{1} & {}^{1} & \\
\hline
 & 11 & (13)_{10} & (10)_{10} \\
 & -8 & -8 & -8 \\
\hline
 & \boxed{1}\ 3 & \boxed{1}\ 5 & \boxed{1}\ 2
\end{array}
$$

$(1352)_8$

Q. $(354)_8 + (266)_8 + (123)_8$

$$
\begin{array}{ccc}
3 & 5 & 4 \\
2 & 6 & 6 \\
1 & 2 & 3 \\
{}^{1} & 14 & 13 \\
 & -8 & -8 \\
\hline
7 & \boxed{1}\ 6 & \boxed{1}\ 5
\end{array}
$$

$(765)_8$

- Octal subtraction using 7's complement.

1. Obtain 7's complement of number to be subtracted.
2. It will be added to first number
3. If carry is generated add it to the sum in step 2
4. If carry is not generated take 7's complement of the result

① $(536)_8 - (345)_8 = (171)_8$

7's complement of 345 is

$$\begin{array}{r} 7\ 7\ 7 \\ -\ 3\ 4\ 5 \\ \hline 4\ 3\ \cancel{5}2 \end{array}$$

$$\therefore \begin{array}{r} 4\ 3\ 5 \\ +\ 5\ 3\ 6 \\ \hline 7\ \ 11 \\ -8 \\ \hline \boxed{1}\ 3 \end{array}$$

$$\begin{array}{r} 4\ 3\ 2 \\ +\ 5\ 3\ 6 \\ \hline 9\ 6\ 8 \\ -\ 1\ 8 \\ \hline \boxed{1}\ 1\ 7\ 0 \\ +\ \ \ 1 \\ \hline 1\ 7\ 1 \end{array}$$

② $(161)_8 - (243)_8 = -(62)_8$

$$\begin{array}{r} 7\ 7\ 7 \\ -\ 2\ 4\ 3 \\ \hline 5\ 3\ 4 \\ +\ 1\ 6\ 1 \\ \hline 9\ 5 \\ -8 \\ \hline 7\ \boxed{1}\ 1\ 5 \end{array}$$

no carry

715

so take 7's complement

$$\begin{array}{r} 7\ 7\ 7 \\ -\ 7\ 1\ 5 \\ \hline -\ 6\ 2 \end{array}$$

Page No.
Date: / /

Octal subtraction using 8's complement.

① $(556)_8 - (345)_8$

$$\begin{array}{r} 7\;7\;7 \\ -\;3\;4\;5 \\ \hline 4\;3\;2 \\ +\;\;\;\;\;1 \\ \hline 4\;3\;3 \end{array} \rightarrow 8\text{'s c.}$$

$$\begin{array}{r} 4\;3\;3 \\ +\;5\;5\;6 \\ \hline 9\;7\;\;9 \\ -8\;\;\;\;-8 \\ \boxed{1}1\;7\;\boxed{1}1 \end{array}$$

carry so discard it

$\rightarrow \boxed{1}171$

$(556)_8 - (345)_8 = (171)_8$

① steps -

1. obtain 8's complemt of no. to be subtracted
2. Add it to 1'st no
3. if carry generated discard carry to get correct ans
4. if carry is not generated take 8's c of the result.

② $(161)_8 - (243)_8 = -(62)_8$

$$\begin{array}{r} 7\;7\;7 \\ -\;2\;4\;3 \\ \hline 5\;3\;4 \\ +\;\;\;\;\;1 \\ \hline 5\;3\;5 \end{array}$$

$$\begin{array}{r} 5\;3\;5 \\ +\;1\;6\;1 \\ \hline 9\;\;6 \\ -8\;\;\;\; \\ 7\;\boxed{1}1\;\; \end{array}$$

$\therefore$ 716 → no carry so take 8'c

10 11 12 13 14
A B C D E

$$\begin{array}{r} 7\ 7\ 7 \\ -\ 7\ 1\ 6 \\ \hline 6\ 1 \\ +\ \ \ 1 \\ \hline (6\ 2) \end{array}$$

- Hexadecimal addition.

$(C2)_H + (3E)_H = (100)_H$

~~(12 2 + (3 14)~~

$$\begin{array}{r} 12\ \ 2 \\ +\ 3\ \ 14 \\ \hline 4\ 3\ 6 \end{array}$$

$$\begin{array}{rr} C & 2 \\ 3 & E \\ \hline 16 & 16 \\ -16 & -16 \\ \hline \boxed{1}0 & \boxed{1}0 \end{array} = 100$$

② $(DDCC)_H + (BBAA)_H$

$$\begin{array}{rrrr} D & D & C & C \\ B & B & A & A \\ \hline 26 & 25 & 23 & 22 \\ -16 & -16 & -16 & -16 \\ \hline 19 & \boxed{1}9 & \boxed{1}7 & \boxed{1}6 \end{array}$$

$(19976)_H$

Page No.
Date:

Hex subtraction using 15's complement.
Same as 7's c just find 15's c.

① $(\cancel{DDCC})_H$ ~~+~~ $-(\cancel{BBAA})_H$

B	B	A	A
11	11	10	10

$$\begin{array}{cccc} 15 & 15 & 15 & 15 \\ -\,11 & 11 & 10 & 10 \\ \hline 4 & 4 & 5 & 5 \end{array}$$

→ 15's c

① (D8A) - (426)

$$\begin{array}{rccc} & 15 & 15 & 15 \\ - & 4 & 2 & 6 \\ \hline & 11 & 13 & 9 \\ + & 13 & 8 & 10 \\ \hline & 25 & 22 & 19 \\ & -16 & -16 & -16 \\ \hline & \boxed{1}\,9 & \boxed{1}\,6 & \boxed{1}\,3 \\ + & & & 1 \\ \hline \end{array}$$

$(9\;6\;4)_H$

16's C

② $(426)_{16} - (D8A) = (964)_{16}$

	15	15	15
	13	8	10
	2	7	5
+	4	2	6
	6	9	11

	2	7	5
+			1
	2	7	6
+	4	2	6
$\boxed{}$	6	9	12

no carry so take 16's c

	15	15	15
−	6	9	12
	9	6	3
+			1
	9	6	4

→ add 1 to make 16's c.

③ $(587)_H - (4EB) = (9C)_H$

	15	15	15
−	4	14	11
	11	1	4
+			1
	11	1	5
+	5	8	7
	16	9	12

$\because$ 16

$\boxed{1}$, 00

$(0\ 9\ 12)_H$

$(9\ C)_H$

Page No.
Date: / /

Codes.

1. Weighted codes
2. Non-weighted codes
3. Error-detecting codes

1. Weighted codes — ① Binary code
② BCD [Binary coded decimal]
only valid for 0-9
10-15 → invalide.

BCD

0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

ex - BCD 10

$\underbrace{00010000}$ → 8 bit

while in binary only 4 bit i.e 1010

Disadv
① So BCD is less efficient than binary

Adv
② In BCD we only require to remember equivalent of 0-9 & represent them which is not in case of binary

BCD addition

1. sum ≤ 9, final carry = 0 ⇒ correct ans
2. sum ≤ 9, final carry = 1 ⇒ Incorrect [add 6 to it to correct it]
3. sum > 9, final carry = 0 ⇒ Incorrect [add 6 to it -11-]

① ex- $(2)_{10} + (6)_{10}$

$$\begin{array}{r} 0010 \\ +\ 0110 \\ \hline 1000 \end{array}$$ ⇒ correct

② $(3)_{10} + (7)_{10}$

$$\begin{array}{r} 0011 \\ +\ 0111 \\ \hline 1010 \end{array}$$ → Invalid as [10]

so case ③ so add 6 to it

$$\begin{array}{r} 1010 \\ +\ 0110 \\ \hline \underbrace{000\boxed{1}}_{1}\ \underbrace{0000}_{0} \end{array}$$ $= (10)_{10}$

③ $(8)_{10} + (9)_{10}$

$$\begin{array}{r} 1000 \\ +\ 1001 \\ \hline \boxed{1}0001 \end{array}$$ → case -②

$$\begin{array}{r} \boxed{1}0001 \\ +\ 0110 \\ \hline \underbrace{0001}_{1}\ \underbrace{0111}_{7} \end{array}$$ $= (17)_{10}$

(4) $(57)_{10} + (26)_{10}$

```
  0101 0111
+ 0010 0110
-----------
  0111 1101   → case (3)
+      0110
-----------
  1000 0011   = (83)10
   8    3
```

(2) $(3)_{10} + (4)_{10}$

```
  0011
  0100
------
  0111   = (7)10
```

(3) $(7)_{10} + (9)_{10}$

```
   0111
 + 1000
 ------
   1111   → case (3)
 + 0110
 ------
 000[1] 0101
    1    6    = (16)10
```

(4) $(83)_{10} + (34)_{10} = (117)_{10}$

```
       1000 0011
       0011 0100
       ---- ----
 >9 ← 1011 0111  → ≤9
 +         0110
 ---------------
       1011 1001
```

```
    1 0 1 1   0 1 1 1
  + 0 1 1 0
  -----------------
000[1] 0 0 0 1   0 1 1 1
    1   1        7
```

- Non-weighted code

Excess-3 code [XS-3]

Decimal → BCD $\xrightarrow{+(3)_{10}}$ XS-3 code

Decimal	BCD	XS-3 (self complimenting)
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

→ complement

XS-3 code is sequential code as we add 1 to it it will result next no

ex

```
    0 0 0 0 → 0
  +       1
  ---------
    0 0 0 1 → 1
  +       1        } adding 1 to it
  ---------
    0 0 1 0 → 2
```

① $(428)_{10}$

	0100	0010	1000	BCD
				$\downarrow +(3)_{10}$
+	0011	0011	0011	
	0111	0101	1011	→ XS-3 code

XS-3 addition

case-

① When final carry = 0, subtract 3

② When final carry = 1, add 3

ex-① $(2)_{10} + (5)_{10} = (7)_{10} \xrightarrow{+(3)_{10}} (10)_{10}$

$$\begin{array}{r} 0010 \\ +\ 0101 \\ \hline 0111 \\ +\ 0011 \\ \hline 1010 \end{array} \qquad \begin{array}{r} 0101 \\ 1000 \\ \hline 1101 \\ -\ 0011 \\ \hline 1010 \end{array}$$

→ carry not generate so subtract 3

② $(27)_{10} + (39)_{10} = (99)_{10}$

$$\begin{array}{r} 0100\ 1010 \\ 0110\ 1100 \\ \hline 1000\ 0110 \end{array}$$

$$\begin{array}{rr} 0101 & 1010 \\ 0110 & 1100 \\ \hline 1100 & 0110 \\ -\ 0011 & +\ 0011 \\ \hline \underbrace{1001}_{9} & \underbrace{1001}_{9} \end{array}$$

ii] Grey code.

Frank Grey

Reflected binary code (RBC)

Unit distance code

- 2 successive values differ in only 1 bit
- Binary no is converted to grey code to reduce switching operation.

Conversion from binary to grey

1. Record MSB as it is
2. Add MSB with next bit, compute sum & ignore carry (XOR operation)
3. Repeat the process.

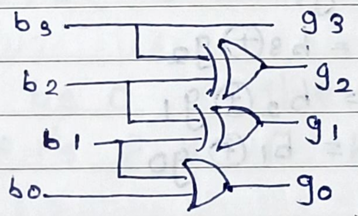
$b_3\ b_2\ b_1\ b_0$

$0\ \ 0\ \ 1\ \ 1$

$\oplus\ \ \oplus\ \ \oplus$

g_3 0 0 1 0 g_0

g_2 g_1



$g_3 = b_3$

$g_2 = b_3 \oplus b_2$

$g_1 = b_2 \oplus b_1$

$g_0 = b_1 \oplus b_0$

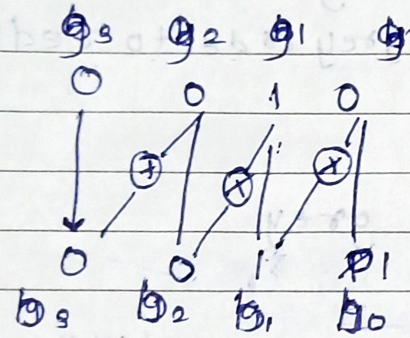
Decimal	Binary	Grey
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Unit distance code bcoz when we changing from 1 binary to other in grey code only 1 bit is changing (i.e 1→2→3→4)

[from 8 they are mirror images like 7-8, 6-9, 5-10

Gray to binary.

1. Record MSB as it is
2. Add MSB ~~bit~~ bit with next bit
3. Repeat the process.

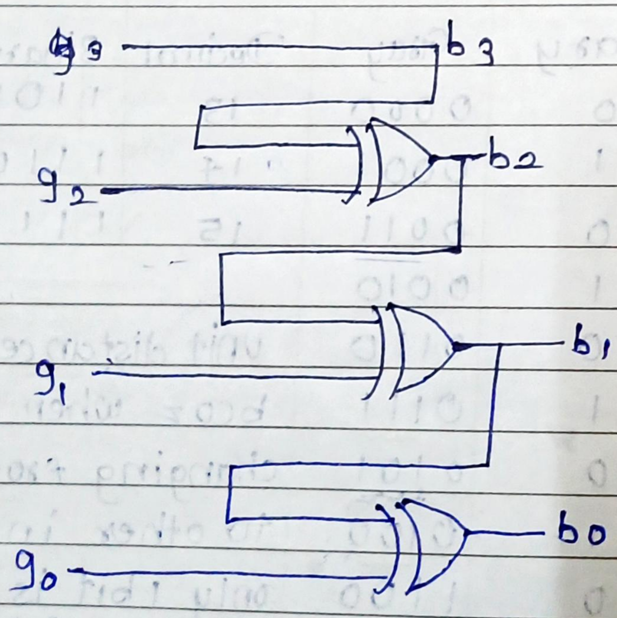


$g_3 = b_3$
$g_2 = b_3 \oplus b_2$
$g_1 = b_2 \oplus b_1$
$g_0 = b$

$\cancel{b_3} = g_3$
$\cancel{b_2} =$.

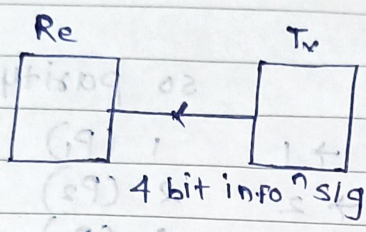
$g_3 = b_3$
$g_3 \oplus b_2 = g_2$
$g_2 \oplus b_1 = g_1$
$g_1 \oplus b_0 = g_0$

$b_3 = g_3$
$b_2 = b_3 \oplus g_2$
$b_1 = b_2 \oplus g_1$
$b_0 = b_1 \oplus g_0$



Error detecting code and error correcting code

parity
- To detect error
- single error will be ~~detect~~ detected



Types
1. odd parity
2. Even parity = Total no. of 1's are even.

- Even parity ⇒ parity will be 0 if there are even no. of 1's & 1 if there are odd no. of 1's

- odd parity → If total no. of 1's are odd. [parity '0' if odd no. of 1's and Parity is '1' if even no. of 1's]

ex→

	Parity
0 1 0 0	0
0 1 1 0	1

- Hamming code [R-W hamming code]

- 7 bit hamming code

7	6	5	4	3	2	1
D_7	D_6	D_5	P_4	D_3	P_2	P_1

Data bit → 4
parity bit → 3

which has position 1

D_1	1	0	0	$\boxed{1}$
D_2	2	0	1	0
D_3	3	0	1	$\boxed{1}$
D_4	4	1	0	0
D_5	5	1	0	$\boxed{1}$
D_6	6	1	1	0
D_7	7	1	1	$\boxed{1}$

$P_1 \to D_3\ D_5\ D_7$

$P_2 \to D_3\ D_6\ D_7$

$P_4 \to D_5\ D_6\ D_7$

parity 2^n i.e $2^0 \to 1$

$2^1 \to 2$

$2^2 \to 4$

so parity position

1 (P_1)

2 (P_2)

4 (P_4)

Q.1) perform binary subtraction.
1. $(26)_{10} - (15)_{10}$
2. $(-31)_{10} - (6)_{10}$
3. $(144)_{10} - (156)_{10}$

Q.2) Perform octal addition
1. $(777)_8 + (77)_8$

Q.3) perform octal subtraction using 7's c and 8's c
$(161)_8 - (243)_8$

Q.4] Perform Hex addition
1. $(658)_{16} + (975)_{16}$
2. $(888)_{16} + (999)_{16}$

Q.5) perform hex subtraction using 15's c + ~~16's~~ c
$(426)_{16} - (D8A)_{16}$
16's c = $(578)_{16} - (7EB)_{16}$

Q If 7 bit hamming code received by reciever is 1011011 assuming even parity. state whether recieved code word is correct or incorrect if incorrect, locate the bit having error.

D_7	D_6	D_5	P_4	D_3	P_2	P_1
1	0	1	1	0	1	1
7	6	5	4	3	2	1

P_1	D_3	D_5	D_7	
1	0	1	1	⇒ odd 1

P_2	D_3	D_6	D_7	
1	0	0	1	⇒ even 0

P_4	D_5	D_6	D_7	⇒ odd 1
1	1	0	1	