

**DEPARTMENT OF ELECTRICAL ENGINEERING
GOVERNMENT COLLEGE OF ENGINEERING
KALAHANDI
BHAWANIPATNA**



Lecture Notes on Basic Electronics

By

Dr. Deepa Das

Assistant Professor

Department of Electrical Engineering

Disclaimer

This document does not claim any originality and cannot be used as a substitute for prescribed textbooks. The information presented here is merely a collection by the committee members for their respective teaching assignments. Various sources such as text books, reference books as well as freely available materials from internet were consulted for preparing this document. The ownership of the information lies with the respective author or the institution. Further, this document is not intended to be used for commercial purpose.

Digital Systems

Electronics circuits and systems are of two kinds

1. Analog
2. Digital

→ Analog circuits are those in which voltages and currents vary continuously through the given range.

→ A digital circuit is one in which the voltage levels assume a finite number of distinct values. All modern digital circuits have only two discrete voltage levels.

→ Digital circuits are often called switching circuits, because the voltage levels in a digital circuit are assumed to be switched from one value to another instantaneously, that is, the transition time is assumed to be zero.

→ Digital circuits are also called logic circuits, because each type of digital circuit obeys a certain set of logic rules. The manner in which a logic circuit responds to an input is referred to as the circuit's logic.

→ Logic design involves determining how to interconnect basic logic building blocks to perform a specific function.

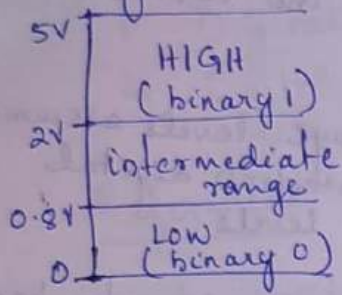
→ In digital systems such as computers, combinations of the two states are used to represent numbers, symbols, alphabetic characters and other type of information.

→ The two states are two binary digits 1 and 0 which are represented by two voltage levels.

• Positive logic system $\left\{ \begin{array}{l} 1 \text{ is represented by HIGH voltage level} \\ 0 \text{ is represented by LOW voltage level} \end{array} \right.$

• Negative logic system $\left\{ \begin{array}{l} 0 \text{ is represented by HIGH voltage level} \\ 1 \text{ is represented by LOW voltage level} \end{array} \right.$

- The voltages used to represent a 1 and a 0 are called logic levels.
- Normally, the binary 0 and 1 are represented by the logic voltage levels 0V and +5V.
- In practical digital circuit, because of circuit variations, the 0 and 1 would be represented by voltage ranges instead of particular voltage levels.

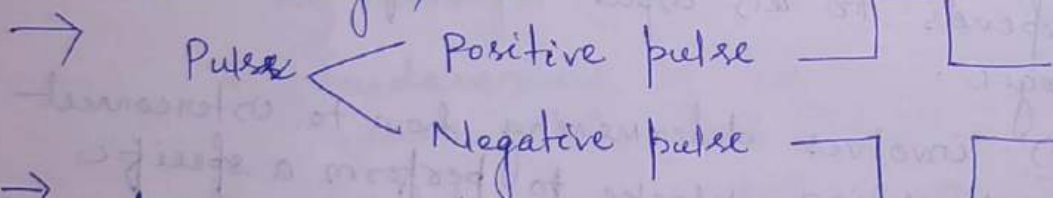


→ Usually, any voltage between 0V and 0.8V represents the logic 0 and any voltage between 2V and 5V represents logic 1.

(Logic level ranges of voltage for a logic circuit)

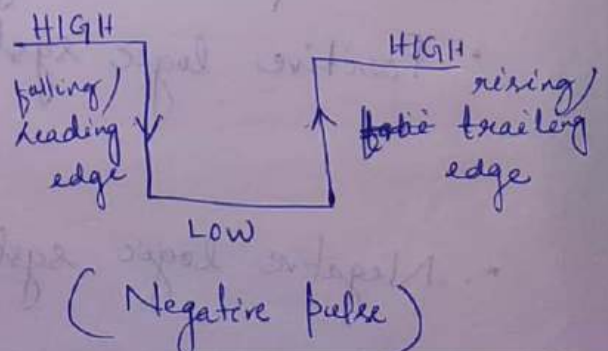
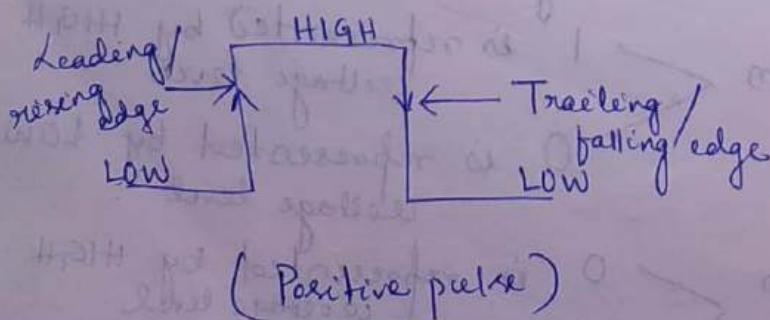
→ The range between 0.8V and 2V is called the intermediate range.

→ In digital circuits and systems, the voltage levels are normally changing back and forth between the HIGH and LOW stages/level.



→ A single positive-going pulse is generated when the voltage goes from its normally LOW level to its HIGH level and then back to its LOW level.

→ The negative-going pulse is generated when the voltage goes from its normally HIGH level to its LOW level and back to its HIGH level.



→ Most waveforms encountered in digital systems are composed of series of pulses, called pulse trains.

Pulse train $\begin{cases} \text{Periodic} \\ \text{Non-periodic} \end{cases}$

→ An important characteristic of a periodic digital waveform is its duty cycle, which is the ratio of the pulse width (t_w) to the period (T).

$$\boxed{\text{Duty cycle} = \left(\frac{t_w}{T}\right) 100\%}$$

Number Systems

1. Decimal Number System
2. Binary " "
3. Octal " "
4. Hexadecimal " "

Decimal Number System

→ The Decimal system contains 10 unique symbols 0-9. The base / radix is 10. Each symbol in the number is called digit.

→ A number with a decimal point is represented by a series of coefficients.

$$a_5 a_4 a_3 a_2 a_1 a_0 . a_{-1} a_{-2} a_{-3}$$

a_j = coefficient — any value between (0-9).

j = gives the place value, the power of 10 by which the coefficient must be multiplied.

→ The value of a digit is determined by its position in the number and has assigned weight.

$$10^3 \ 10^2 \ 10^1 \ 10^0 . 10^{-1} \ 10^{-2} \ 10^{-3}$$

→ The leftmost digit in any number representation has the greatest positional weight called most significant digit (MSD). The rightmost digit has least

Positional weight, called least significant digit (LSI).

Binary Number Systems

- The coefficients of the binary number system have only two possible values 0 and 1. The base/radix is 2. The binary digit is called bit. Eg: - $(110)_2$
- The position of a 1 or 0 in a binary number indicates its weight. The weights in a binary number are based on powers of 2.

Eg $d_3 d_2 d_1 d_0 \cdot d_{-1} d_{-2} d_{-3}$

$$d_3 \times 2^3 + d_2 \times 2^2 + d_1 \times 2^1 + d_0 \times 2^0 + d_{-1} \times 2^{-1} + d_{-2} \times 2^{-2} + d_{-3} \times 2^{-3}$$

- The right most bit is the least significant bit (LSB) and the left most bit is called the most significant bit (MSB).

* Computer capacity is usually given in bytes.

1 byte = 8 bits

1 Nibble = 4 bits

1 KB = 2^{10} bytes = 1024 bytes

1 MB = 2^{20} bytes

1 GB = 2^{30} bytes

Octal Number System

- The octal number system has a base of 8. It has eight digits (0-7). Eg: $(123.7)_8$

Base 8 = 2^3 — Every 3 bit group of binary can be represented by an octal digit.

Hexadecimal Number System

- The base/radix is 16. It has 16 independent symbols: (0-15). The 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Eg: $(F10.2C)_{16}$

Base 16 = 2^4 — Every 4 bit group of binary can be represented by an hexadecimal digit.

Number - Base Conversion

1. Conversion of $()_n$ to decimal number :-

→ The conversion of a number in base n to decimal equivalent is done by multiplying each coefficient to its position weight and the product terms are added.

Eg $(x_3 x_2 x_1 x_0 . x_{-1} x_{-2})_n$

$$= x_3 \times n^3 + x_2 \times n^2 + x_1 \times n^1 + x_0 \times n^0 + x_{-1} \times n^{-1} + x_{-2} \times n^{-2}$$

a. Binary to decimal :-

Eg $(1101.11)_2 = ()_{10}$

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

$$= 8 + 4 + 1 + 0.5 + 0.25 = 13.75$$

b. Octal to decimal :-

Eg $(234.25)_8 = ()_{10}$

$$\Rightarrow 2 \times 8^2 + 3 \times 8^1 + 4 \times 8^0 + 2 \times 8^{-1} + 5 \times 8^{-2} =$$

c. Hexadecimal to decimal :-

Eg $(FAB.2C)_{16} = ()_{10}$

$$\Rightarrow F \times 16^2 + A \times 16^1 + B \times 16^0 + 2 \times 16^{-1} + C \times 16^{-2} =$$

d. $(321.22)_5$ to decimal

$$\Rightarrow 3 \times 5^2 + 2 \times 5^1 + 1 \times 5^0 + 2 \times 5^{-1} + 2 \times 5^{-2} =$$

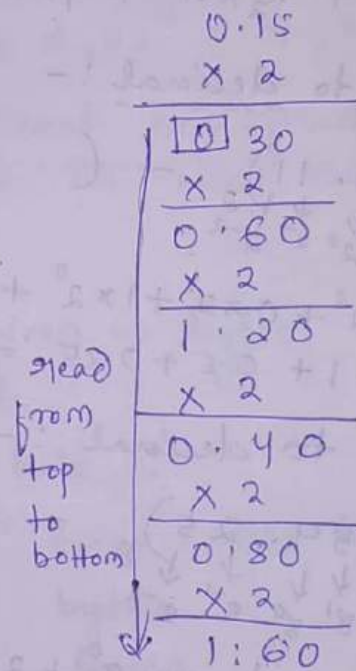
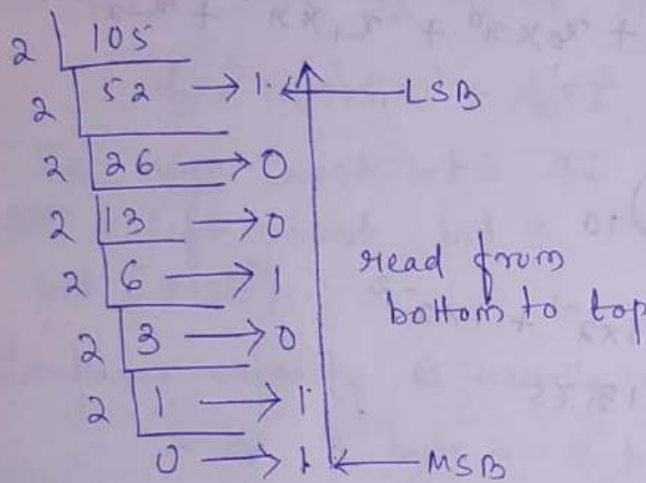
2. Conversion of decimal number to $()_2$

→ The decimal integer number is converted to the number in base 2 by successive division by 2, and the decimal fraction is converted to fraction number in base 2 by successive multiplication by 2.

— This method is also known as double-dabble method.

a. Decimal to Binary :- (Successive division by 2 method)

Eg $(105.15)_{10} = ()_2$



Ans $(105.15)_{10} = (1101001.001001)_2$

* 2nd method to convert decimal to binary
Sum-of-weights method :- (normally used for small numbers)

Eg $(75)_{10} = ()_2$

Ans The largest number, which is a power of 2, not exceeding 75 is 64 i.e. 2^6

$$64 = 2^6 = (1000000)_2$$

The remainder is

$$75 - 64 = 11$$

The largest number, which is a power of 2, not exceeding 11 is 8 i.e. 2^3

$$8 = 2^3 = 1000$$

The remainder is $11 - 8 = 3$

The largest number, which is a power of 2, not exceeding 3 is $2 = 2^1 = (10)_2$

The remainder is $3 - 2 = 1$.

$$1 = 2^0$$

Hence $(75)_{10} = (1000000)_2 + (1000)_2 + (10)_2 + 1$

Counting in Binary

<u>Decimal number</u>	<u>Binary number</u>	<u>Decimal number</u>	<u>Binary number</u>
0	→ 0	9	→ 1001
1	→ 1	10	→ 1010
2	→ 10	11	→ 1011
3	→ 11	12	→ 1100
4	→ 100	13	→ 1101
5	→ 1001	14	→ 1110
6	→ 110	15	→ 1111
7	→ 111		
8	→ 1000		

* $(2^n)_{10} = (1 \text{ followed by } n \text{ zeros})_2$

* $(2^n - 1)_{10} = (n \text{ ones})_2$

b. Decimal to Octal :-

Eg $(378.93)_{10} = (\quad)_8 = (572.73241)_{10}$

8	378	→ 2	↑ LSB ↓ MSB
8	47	→ 7	
8	5	→ 5	
	0		

0.93	416
x 8	x 8
7.44	1.28
x 8	
3.52	
x 8	
4.16	

C. Convert $(2598.675)_{10} = (\quad)_{16}$

16	2598	
16	162	→ 6
16	10	→ 2
	0	→ A

0.675
× 16
10.8
× 16
12.8
× 16
12.8
× 16
12.8

Ans $(2598.675)_{10}$
 $= (A26.ACCC)_{16}$

D. Octal to Binary

→ Each octal digit is represented by a 3-bit binary number.

Eg

<u>Octal digit</u>	<u>Binary digit</u>
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Eg $(7526.766)_8 = (\quad)_2$

Ans $(111101010110.111110110)_2$

E. Binary to Octal

→ Start with the right-most group of three bits and moving from right to left, convert each 3-bit group to the equivalent octal digit.

Eg $(11011.1011)_2 = (\quad)_8$
 $(011011.101100)_2 = (33.54)_8$

E. Hexadecimal to Binary

→ Replace each hex digit by its 4-bit binary group.

<u>Hexadecimal digit</u>	<u>Binary digit</u>	<u>Hexadecimal digit</u>	<u>Binary digit</u>
0	0000	9	1001
1	0001	A	1010
2	0010	B	1011
3	0011	C	1100
4	0100	D	1101
5	0101	E	1110
6	0110	F	1111
7	0111		
8	1000		

Eg $(5A32.10C)_{16} = (\quad)_2$

$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0101 & 1010 & 1011 & 0010 & 0001 \end{matrix}$

Ans $(0101101010110010.000100001100)_2$

F. Binary to Hexadecimal :-

→ Make groups of 4 bits, and replace each 4-bit group by a hex digit.

Eg $(11011.11011)_2 = (\quad)_{16} = (1B.D8)_{16}$

$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow \\ 0001 & 1011 & 1101 & 1000 \\ \downarrow & & & \downarrow \\ 1 & B & D & 8 \end{matrix}$

G. Octal to Hexadecimal :-

Octal → binary → Hexadecimal

Eg $(730.12)_8 = (\quad)_{16}$

$(730.12)_8 \rightarrow (1D8.28)_{16}$ (Ans)

$\begin{matrix} \downarrow & & \downarrow & \downarrow & \downarrow \\ 111011000 & 001010 \\ 1 & 0 & 8 & 2 & 8 \end{matrix}$

H. Hexadecimal to octal

Hexadecimal \rightarrow binary \rightarrow octal

Eg $(BC2.F3)_{16} = (\quad)_{8}$

Ans $(BC2.F3)_{16}$
 \downarrow

$(\underbrace{101111000010}_{\text{BC2}} \cdot \underbrace{11110011}_{\text{F3}})_{2} \rightarrow (5702.7406)_{8}$

Questions

1. Find x

(a) $(55)_x = (30)_{10}$

(b) $(257)_x = (140)_{10}$

(c) $(431)_x = (116)_{10}$

(d) $\frac{14}{x} = 5$

(e) $(103)_8 - (45)_8 = (x)_5$ (f) $\frac{41}{3} = 13$

(g) $24 + 17 = 40$

(h) $\sqrt{(14)_x} = 3$

(i) $(125)_6 = (\quad)_5$

(j) The solutions to the quadratic equation $x^2 - 11x + 22 = 0$ are $x=3$ and $x=6$. What is the base of the number?

Solutions

1(a) $(55)_x = (30)_{10}$

\downarrow convert it into decimal

$$5x^1 + 5x^0 = 30 \Rightarrow 5x + 5 = 30$$

$$\Rightarrow x = \frac{25}{5} = 5 \Rightarrow \boxed{x=5}$$

(b) $(257)_x = (140)_{10}$

\downarrow convert it into decimal

$$2x^2 + 5x + 7 = 140$$

$$2x^2 + 5x - 133 = 0$$

$$x = \frac{-5 \pm \sqrt{5^2 - 4 \times 2 \times (-133)}}{4} =$$

(c) Same procedure

$$(d) \frac{14}{2} = 5$$

Let the base is x . Then

$$\frac{(14)_x}{(2)_x} = (5)_x$$

$$\frac{x+4}{2} = 5 \Rightarrow x+4 = 10 \Rightarrow \boxed{x=6}$$

$$(e) (103)_8 - (45)_8 = (x)_5$$

$$\Rightarrow \begin{array}{r} (103)_8 \\ - (45)_8 \\ \hline (36)_8 \end{array} \quad (36)_8 = ()_5$$

$(36)_8 = (3 \times 8 + 6) = (30)_{10}$
 \uparrow equivalent decimal number

Hence $(30)_{10} = (x)_5$

\downarrow convert it into decimal

$$\begin{array}{r} 5 \overline{) 30} \\ \underline{5 \times 6 = 30} \\ 0 \end{array}$$

$$\Rightarrow \cancel{x=30} \quad (30)_{10} = (110)_5$$

$$\Rightarrow \boxed{x=110}$$

$$\frac{41}{3} = 13$$

Let the base is x . Then

$$\frac{(15)_x}{(4)_x} = (13)_x$$

$$\frac{x+5}{4} = x+3 \Rightarrow x+5 = 4x+12$$

$$\Rightarrow \frac{(41)_x}{(3)_x} = (13)_x \Rightarrow \frac{4x+1}{3} = x+3$$

$$\Rightarrow 4x+1 = 3x+9 \Rightarrow \boxed{x=8}$$

Same as (e) and (f).

Binary Arithmetic

1. Binary Addition :-

The four basic rules for adding binary bits are

- $0 + 0 = 0$ — Sum 0, carry 0
- $0 + 1 = 1$ — Sum 1, carry 0
- $1 + 0 = 1$ — Sum 1, carry 0
- $1 + 1 = 10$ → Sum 0, carry 1

Eg 1. $(111.010)_2 + (101.111)_2 = (1101.001)_2$

Ans

$$\begin{array}{r}
 111111 \\
 111.010 \\
 + 101.111 \\
 \hline
 1101.001
 \end{array}$$

Eg 2 $(1111)_2 + (1111)_2 + (1111)_2 = ()$

Ans $(101101)_2$

4 = $(100)_2$ goes to next higher bit

$$\begin{array}{r}
 1111 \\
 + 1111 \\
 + 1111 \\
 \hline
 101101
 \end{array}$$

2. Binary Subtraction :-

The four basic rules for subtracting bits are as follows

- $0 - 0 = 0$ Difference 0, borrow 0
- $1 - 0 = 1$ Difference 1, borrow 0
- $1 - 1 = 0$ Difference 0, borrow 0
- $0 - 1 = 1$ Difference 1, borrow 1

Eg $(1010.01)_2 - (111.111)_2 = (\quad)_2$

Ans

$$\begin{array}{r} 1010.010 \\ - 111.111 \\ \hline 0011.010 \\ \hline \end{array}$$

Eg 2

$(10000)_2 - (1111)_2$

Ans

$$\begin{array}{r} 10000 \\ - 1111 \\ \hline 0001 \\ \hline \end{array}$$

3. Binary multiplication :

The four basic rules for multiplying bits are

$0 \times 0 = 0$

$0 \times 1 = 0$

$1 \times 0 = 0$

$1 \times 1 = 1$

Eg

$(110)_2 \times (111)_2 = (101010)_2$

$$\begin{array}{r} 110 \\ \times 111 \\ \hline 1110 \\ 1110 \\ 110 \\ \hline (101010)_2 \end{array}$$

4. Binary Division

Same as decimal division.

Eg $(101101)_2 \div (110)_2$

$$\begin{array}{r}
 110 \) \ 101101 \ (111.1 \\
 \underline{110} \\
 1010 \\
 \underline{110} \\
 1001 \\
 \underline{110} \\
 00110 \\
 \underline{110} \\
 0
 \end{array}$$

Eg $(110101.11)_2 \div (101)_2$

Complements of Numbers :-

→ Complements are used in digital computers to simplify the subtraction operation and for logical manipulation.
 → Simplifying operations leads to simpler, less expensive circuits to implement the operations.

There are two types of complements for each base-r system.

1. The radix/base complement (r 's complement)
2. The diminished radix complement ($(r-1)$'s complement)

Diminished Radix Complement :-

Given a number N in base r having n digits, the $(r-1)$'s complement of N , its diminished radix complement is defined as $(r^n - 1) - N$.

Eg (a) Decimal number:

$r = 10$
 Let $n = 4$. $10^4 = 10000$
 $(10^4 - 1) = 9999$

$r = 10$
 $(r^n - 1) = n \text{ } r$'s

Hence, 9's complement of a decimal number is obtained by subtracting each digit from 9.

(b) Binary number:

$r = 2$
 Let $n = 4$
 $2^n = 2^4$ in binary is $(10000)_2$
 $(2^4 - 1) = (10000)_2 - 1 = (1111)_2 = 4 \text{ ones } 1$'s

Thus, the 1's complement of a binary number is obtained by subtracting each digit from 1.

→ 1's complement of a binary number is formed by changing 1's to 0's and 0's to 1's.

(c) For octal number:

$$r = 8$$

$$\text{let } n = 4$$

8^4 in octal is represented as $(10000)_8$

$$(8^4 - 1) = (10000)_8 - (1)_8 = (7777)_8$$

Hence, 7's complement of octal number system is obtained by subtracting each digit from 7.

(d) Hexadecimal number:

$$r = 16$$

$$\text{let } n = 4$$

16^4 in hexadecimal is represented as $(10000)_{16}$

$$(16^4 - 1) = (10000)_{16} - 1 = (FFFF)_{16}$$

Hence, $(r-1)$'s complement or 15's complement is obtained by subtracting each digit from F.

Radix complement

The r 's complement of a n digit number N in base r is defined as $r^n - N$ for $N \neq 0$ and 0 for $N = 0$.

Eg base 2 = $r = 2$'s complement
 $(r-1) = 1$'s complement

base 8 = $r = 8$'s complement
 $(r-1) = 7$'s complement

base 16 = $r = 16$'s complement
 $(r-1) = 15$'s complement

base 10 = $r = 10$'s complement
 $(r-1) = 9$'s complement

Subtraction of unsigned number using 9's and (n-1)'s complement:

Let $A - B$

No. of digits in $A =$ No. of digits in B

If No. of digits in $A \neq$ No. of digits in B

Equal them by putting 0 in the left side of the number which has less digits.

9's complement :-

→ Obtain the 9's complement of B and add it to the A .
If there is a carry, ignore it. The remaining sum value will be the answer for $A - B$.

* The presence of carry indicates that the answer is +ve or $A > B$.

→ If there is no carry, it indicates that the answer is negative or $A < B$. Obtain the 9's complement of the result and place a -ve sign.

(n-1)'s complement :-

→ Obtain the (n-1)'s complement of B and add it to A .

→ If there is a carry, it indicates that the answer is +ve and add the carry to the LSB of the result to get the answer. This is called end around carry.

→ If there is no carry, it indicates that the answer is -ve or $A < B$ and the intermediate result is its (n-1)'s complement.

→ Take the (n-1)'s complement of this result and place a negative sign in front to get the answer.

2's complement - (signed number)

In binary number system, if the MSB is 0, the result is +ve and is in true binary form. If the MSB is 1, the result is -ve and is its 2's complement (whether there is a carry or no carry at all)