COMPUTER SCIENCE & INFORMATION TECHNOLOGY

Digital Logic

Comprehensive Theory
with Solved Examples *and* Practice Questions

MADE EASY PUBLICATIONS

www.madeeasypublications.org

MADE EASY Publications Pvt. Ltd.
Corporate Office: 44-A/4, Kalu Sarai (Near Hauz Khas Metro Station), New Delhi-110016 | **Ph. :** 9021300500
Email : infomep@madeeasy.in | **Web :** www.madeeasypublications.org

Digital Logic

EDITIONS

First Edition : 2015
Second Edition : 2016
Third Edition : 2017
Fourth Edition : 2018
Fifth Edition : 2019
Sixth Edition : 2020
Seventh Edition : 2021
Eighth Edition : 2022
Ninth Edition : 2023
Tenth Edition : 2024
Eleventh Edition : 2025

Twelfth Edition : 2026

CONTENTS

Digital Logic

Digital Logic

GOAL OF THE SUBJECT

Digital logic design is a system in electrical and computer engineering that uses simple number values to produce input and output operations. As a digital design engineer, you may assist in developing cell phones, computers, and related personal electronic devices. Digital logic is the representation of signals and sequences of a digital circuit through numbers. It is the basis for digital computing and provides a fundamental understanding on how circuits and hardware communicate within a computer. Digital logic is typically embedded into most electronic devices, including calculators, computers, video games, and watches. This field is utilized by many careers that work with computers and technology, such as engineers and repair technicians.

More specifically, DLD provides following things:

- It dictates how the number can be represented in computers and it's conversion in various bases.
- It includes various gates which are helpful in designing of circuits.
- It also allows us to minimize the functions using Karnaugh map technique which is widely popular in digital world.
- Moreover, DLD also defines flip flop which helps in recording the count of values and can be used to store values in registers which are very fast to access.

INTRODUCTION

Digital logic design deals with electronics that operate on digital signals. Digital techniques are helpful because it is much easier to get an electronic device to switch into one of a number of unknown states than to accurately reproduce a continuous range of values.

Chapter 1: Basics of Digital Logic: In this chapter, we discuss digital number systems, codes, arithmetic operations on signed number representation and its overflow concept.

Chapter 2: Boolean Algebra and Minimization Techniques: In this chapter, we discuss about boolean algebra, its laws and postulates, minimization of logic functions using K-map.

Chapter 3: Logic Gates and Switching Circuits: In this chapter, we study basic gates and its properties. Moreover, universal gates and its properties have also been discussed.

Chapter 4: Combinational Logic Circuits: In combinational circuits, we discuss full adder/half adder, subtractors with different propers. Hazards mainly static 1 has also been introduced.

Chapter 5: Sequential Logic Circuits In this chapter we get to know about latches, flip-flops using NAND/NOR gates. All kinds of flip-flops are defined in this very particular chapter.

Chapter 6: Registers and Counters: In this chapter we discuss application of flip-flops which includes registers and counters. Some standard counters like ring and Jhonson are also discussed.

■■■■

CHAPTER

1

Basics of Digital Logic

1.1 INTRODUCTION

Electronic systems are of two types:

(*i*) Analog systems

(*ii*) Digital systems

Analog systems are those systems in which voltage and current variations are continuous through the given range and they can take any value within the given specified range, whereas a digital system is one in which the voltage level assumes finite number of distinct values. In all modern digital circuits there are just two discrete voltage level.

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. Digital circuits are also called logic circuits, because every digital circuit obeys a certain set of logical rules.

Digital systems are extensively used in control systems, communication and measurement, computation and data processing, digital audio and video equipments, etc.

1.1.1 Advantages of Digital Systems

Digital systems have number of advantages over analog systems which are summarized below:

I. Ease of Design

The digital circuits having two voltage levels, OFF and ON or LOW and HIGH, are easier to design in comparison with analog circuits in which signals have numerical significance ; so their design is more complicated.

II. Greater Accuracy and Precision

Digital systems are more accurate and precise than analog systems because they can be easily expanded to handle more digits by adding more switching circuits.

III. Information Storage is Easy

There are different types of semiconductor memories having large capacity, which can store digital data.

IV. Digital Systems are More Versatile

It is easy to design digital systems whose operation is controlled by a set of stored instructions called program. However in analog systems, the available options for programming is limited.

V. Digital Systems are Less Affected by Noise

The effect of noise in analog system is more. Since in analog systems the exact values of voltages are important. In digital system noise is not critical because only the range of values is important.

VI. Digital Systems are More Reliable

As compared to analog systems, digital systems are more reliable.

Limitations of Digital System

(*i*) The real world is mainly analog.
(*ii*) Human does not understand the digital data.

1.2 RADIX NUMBER SYSTEMS

The numeric system we use daily is the decimal system, but this system is not convenient for machines since the information is handled codified in the shape of on or off bits, this way of codifying takes us to the necessity of knowing the positional calculation which will allow us to express a number is any base where we need it.

A base of a number system or radices defines the range of values that a digit may have.

1. In the binary system or base 2, there can be only two values for each digit of a number, either a "0" or a "1".
2. In the octal system or base 8, there can be eight choices for each digit of a number:
"0", "1", "2", "3", "4", "5", "6", "7"
3. In the decimal system or base 10, there are ten different values for each digit of a number:
"0", "1", "2", "3", "4", "5", "6", "7", "8", "9"
4. In the hexadecimal system, we allow 16 values for each digit of a number:
"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "A", "B", "C", "D" and "F"
where "A" stands for 10, "B" for 11 and so on.

In general, a positive number N can be written in positional notation as

$$N = (a_{n-1}\, a_{n-2} \ldots a_1\, a_0 \,.\, a_{-1}\, a_{-2} \ldots a_{-m})$$

where,

. = radix point separating the integer and fractional digits.
r = radix or base of the number system being used
n = number of integer digits to the left of the radix point
m = number of fractional digits to the right of the radix point
a_i = integer digits i when $n - 1 \geq i \geq 0$
a_j = fractional digits j when $-1 \geq i \geq -m$
a_{n-1} = most significant digit
a_{n-2} = least significant digit

A number system with base or radix 'r' will have r number of different digits from $0 \rightarrow (r - 1)$ thus, number system is represented by $(N)_b$

where, N = Number ; b = Base or radix

1.3 CONVERSION AMONG RADICES

1.3.1 Convert from Decimal to Any Base

Let's think about what you do to obtain each digit. As an example, let's start with a decimal number 1234 and convert it to decimal notation. To extract the last digit, you move the decimal point digit, you move the decimal point left by one digit, which means you divide the given number by its base 10.

$$1234/10 = 123 + 4/10$$

To remainder of 4 is the last digit. To extract the next last digit, you again move the decimal point left by one digit and see what drops out.

$$123/10 = 12 + 3/10$$

The remainder of 3 is the next last digit. You repeat this process until there is nothing left. Then you stop in summary, you do the following:

	Quotient	Remainder
1234/10	123	4
123/10	12	3
12/10	1	2
1/10	0	1

(Stop when quotient is 0)

1 2 3 4

Now, let's try a non-trivial example. Let's express a decimal number 1341 in binary notation. Note that the desired base is 2, so we repeatedly divide the given decimal number by 2.

	Quotient	Remainder
1341/2	670	1
670/2	335	0
335/2	167	1
167/2	83	1
83/2	41	1
41/2	20	1
20/2	10	0
10/2	5	0
5/2	2	1
2/2	1	0
1/2	0	1

(Stop when the quotient is 0) (BIN; Base 2)

1 0 1 0 0 1 1 1 1 0 1

Let's express the same decimal number 1341 in hexadecimal notation.

	Quotient	Remainder
1341/16	83	13
83/16	5	3
5/16	0	5

(Stop when the quotient is 0) (HEX; Base 16)

5 3 D

Conclusion:

In conclusion, the easiest way to convert fixed point numbers to any base is to convert each part separately. We begin by separating the number into its integer and fractional part. The integer part is converted using the remainder method, by using a successive division of the number in the base until a zero is obtained. At each division, the reminder is kept and then the new number in the base *r* is obtained by reading the remainder from the last to remainder upwards.

The conversion of the fractional part can be obtained by successively multiplying the fraction with the base. If we iterate this process on the remaining fraction, then we will obtain successive significant digit. This methods form the basis of the multiplication methods of converting fractions between bases.

Example 1.1 Convert $(13)_{10}$ to binary.

Solution :

	Quotient	Remainder	
13 ÷ 2	6	1	LSB ↑
6 ÷ 2	3	0	
3 ÷ 2	1	1	
1 ÷ 2	0	1	MSB

∴ $(13)_{10} \Rightarrow (1101)_2$

Example 1.2 Convert $(0.65625)_{10}$ to an equivalent base-2 number.

Solution :

$0.65625 \times 2 = 1.31250 \rightarrow 1$

$0.31250 \times 2 = 0.62500 \rightarrow 0$

$0.62500 \times 2 = 1.25000 \rightarrow 1$

$0.25000 \times 2 = 0.50000 \rightarrow 0$

$0.50000 \times 2 = 1.00000 \rightarrow 1$

Thus, $(0.65625)_{10} = (0.10101)_2$

Example 1.3 Convert $(3287.5100098)_{10}$ into octal.

Solution :

For integral part:

	Quotient	Remainder
3287 ÷ 8	410	7 ↑
410 ÷ 8	51	2
51 ÷ 8	6	3
6 ÷ 8	0	6

∴ $(3287)_{10} = (6327)_8$

Now for fractional part:

$0.5100098 \times 8 = 4.0800784 \rightarrow 4$

$0.0800784 \times 8 = 0.6406272 \rightarrow 0$

$0.6406272 \times 8 = 5.1250176 \rightarrow 5$

$0.1250176 \times 8 = 1.0001408 \rightarrow 1$

∴ $(0.5100098)_{10} = (0.4051)_8$

Finally, $(3287.5100098)_{10} = (6327.4051)_8$

Example 1.4 Convert $(675.625)_{10}$ into Hexadecimal.

Solution :

For Integral Part:

	Quotient	Remainder
675 ÷ 16	42	3
42 ÷ 16	2	10 = A
2 ÷ 16	0	2

$\therefore$ $(675)_{10} = (2A3)_{16}$

For Fractional Part:

$625 \times 16 = 10 = A$

$\therefore$ $(0.625)_{10} = (0.A)_{16}$

Finally, $(675.625)_{10} = (2A3.A)_{16}$

1.3.2 Convert from Any Base to Decimal

Let's think more carefully what a decimal number means. For example, 1234 means that there are four boxes (digits); and there are 4 one's in the right most box (least significant digit), 3 ten's in the next box, 2 hundred's in the next box, and finally 1 thousand's in the left-most box (most significant digit). The total is 1235:

Original number:	1		2		3		4	
	↓		↓		↓		↓	
How many tokens:	1		2		3		4	
Digit/Token value:	1000		100		10		1	
Value:	1000	+	200	+	30	+	4	= 1234

or simply, $1 \times 1000 + 2 \times 100 + 3 \times 10 + 4 \times 1 = 1234$

Thus each digit has a value. $10^0 = 1$ for the least significant digit, increasing to $10^1 = 10$, $10^2 = 100$, $10^3 = 1000$ and so forth.

Likewise, the least significant digit in a hexadecimal number has a value of $16^0 = 1$ for the least significant digit, increasing to $16^1 = 16$ for the next digit, $16^2 = 256$ for the next, $16^3 = 4096$ for the next, and so forth. Thus, 1234 means that there are four boxes (digits); and there are 4 one's in the right-most box (least significant digit), 3 sixteen's in the next box, 2 256's in the next, and 1 4096's in the left-most box (most significant digit). The total is:

$$1 \times 4096 + 2 \times 256 + 3 \times 16 + 4 \times 1 = 4660$$

In summary, the conversion from any base to base 10 can be obtained from the formulae $x_{10} = \sum_{i=-m}^{n-1} d_i b^i$.

Where b is the base, d_i the digit at position i, m the number of digit after the decimal point, n the number of digits of the integer part and x_{10} is the obtained number in decimal. This form the basic of the polynomial method of converting numbers from any base to decimal.

Example 1.5 Convert $(3A.2F)_{16}$ into decimal system.

Solution :

$$(3A.2F)_{16} = 3 \times 16^1 + 10 \times 16^0 + 2 \times 16^{-1} + 15 \times 16^{-2}$$

$$= 48 + 10 + \frac{2}{16} + \frac{15}{16^2} = (58.1836)_{10}$$

Example 1.6 Convert $(6327.4051)_8$ into its equivalent decimal number.

Solution :

$$(6327.4051)_8 = 6 \times 8^3 + 3 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 + 4 \times 8^{-1} + 0 \times 8^{-2} + 5 \times 8^{-3} + 1 \times 8^{-4}$$

$$= 3072 + 192 + 16 + 7 + \frac{4}{8} + 0 + \frac{5}{512} + \frac{1}{4096}$$

$$= (3287.5100098)_{10}$$

Thus, $(6327.4051)_8 = (3287.5100098)_{10}$

Example 1.7 The decimal number representation of 101101.10101 is

Solution :

$$(101101.10101)_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 0 \times 2^{-4} + 1 \times 2^{-5}$$

$$= 32 + 0 + 8 + 4 + 0 + 1 + \frac{1}{2} + 0 + \frac{1}{8} + 0 + \frac{1}{32} = (45.65625)_{10}$$

Example 1.8 A particular number system having base B is given as $(\sqrt{41})_B = 5_{10}$. The value of 'B' is

(a) 5 (b) 6
(c) 7 (d) 8

Solution : (b)

Squaring both side, $(\sqrt{41})_B = (5)_{10}$

$$\left[\sqrt{(4B+1)_{10}}\right]^2 = [(5)_{10}]^2$$

$$(4B + 1)_{10} = (25)_{10}$$

$\Rightarrow$ $B = 6$

1.3.3 Relationship Between Binary-Octal and Binary-Hexadecimal

As demonstrated by the table below, there is a direct correspondence between the binary system and the octal system with three binary digits corresponding to one octal digit. Likewise, four binary digits translate directly into one hexadecimal digit.

With such relationship, in order to convert a binary number to octal, we partition the base 2 number into groups of three starting from the radix point, and pad the outermost groups with 0's as needed to form triples. Then, we convert each triple to the octal equivalent.

For conversion from base 2 to base 16, we use groups of four.

Consider converting 10110_2 to base 8:

$$10110_2 = 010_2\ 110_2 = 2_8 6_8 = 26_8$$

Notice that the leftmost two bits are pedaled with a 0 on the left in order to create a full triplet.

BIN	OCT	HEX	DEX
0000	00	0	0
0001	01	1	1
0010	02	2	2
0011	03	3	3
0100	04	4	4
0101	05	5	5
0110	06	6	6
0111	07	7	7
1000	10	8	8
1001	11	9	9
1010	12	A	10
1011	13	B	11
1100	14	C	12
1101	15	D	13
1110	16	E	14
1111	17	F	15

Example 1.9 Convert $(2F9A)_{16}$ to Binary System

Solution :

$$\begin{array}{cccc} 2 & F & 9 & A \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 0010 & 1111 & 1001 & 1010 \end{array}$$

$\therefore \quad (2F9A)_{16} = (0010\ 1111\ 1001\ 1010)_2$

Example 1.10 Convert $(10100110101111)_2$ to hexadecimal number system.

Solution :

$$\underbrace{00\,10}_{\downarrow\, 2}\ \underbrace{10\,01}_{\downarrow\, 9}\ \underbrace{10\,10}_{\downarrow\, A}\ \underbrace{1111}_{\downarrow\, F}$$

Here two 0's at MSB are added to make a complete group of 4 bits.

$\therefore \quad (10100110101111)_2 = (29AF)_{16}$

The number systems can also be classified as weighted binary number and unweighted binary number. Where weighted number system is a positional weighted system for example, Binary, Octal, Hexadecimal BCD, 2421 etc. The unweighted number systems are non-positional weightage system for example Gray code, Excess-3 code etc.

Example 1.11 Convert $(472)_8$ into binary

Solution :

$$\begin{array}{cccc} & 4 & 7 & 2 \\ & \Downarrow & \Downarrow & \Downarrow \\ (472)_8 = & (100 & 111 & 010)_2 \end{array}$$

Example 1.12 Convert $(1011011110.11001010011)_2$ into octal.

Solution :

For left-side of the radix point, we grouped the bits from LSB: $\underbrace{0\,0\,1}_{\downarrow\,1}\ \underbrace{0\,1\,1}_{\downarrow\,3}\ \underbrace{0\,1\,1}_{\downarrow\,3}\ \underbrace{1\,1\,0}_{\downarrow\,6}$

Here two 0's at MSB are added to make a complete group of 3 bits.

For right-side of the radix point, we grouped the bits from MSB: $\underset{\uparrow \text{ radix point}}{\bullet}\ \underbrace{1\,1\,0}_{\downarrow\,6}\ \underbrace{0\,1\,0}_{\downarrow\,2}\ \underbrace{1\,0\,0}_{\downarrow\,4}\ \underbrace{1\,1\,0}_{\downarrow\,6}$

Here a '0' at LSB is added to make a complete group of 3 bits.

Finally, $(1011011110.11001010011)_2 = (1336.6246)_8$

Remember

To convert Fractional decimal into binary, Multiply the number by '2'. After first multiplication integer digit of the product is the first digit after binary point. Later only fraction part of the first product is multiplied by 2. The integer digit of second multiplication is second digit after binary point, and so on. The multiplication by 2 only on the fraction will continue like this based on conversion accuracy or until fractional part becomes zero.

1.4 CODES

When numbers, letters or words are represented by a special group of symbols, we say that they are being encoded, and the group of symbols is called "CODE".

CODES

Non-Binary

Ex: Morse code

⇒ In Morse code series of dots and dashes represents the letters of the alphabet.

Binary

Ex: Gray code

BCD, Excess-3 Code, 8-4-2-1 code
Error detecting and correcting codes

Alphanumeric

Ex: Boudot's code (5 bit code)

Internal code (6 bit)
ASCII code (7 bit + 1bit parity)
EBCDIC (8 bit)

1.4.1 Binary Coded Decimal Code (BCD)

- In this code, each digit of a decimal number is represented by binary equivalent.
- It is a 4-bit binary code.
- It is also known as "8-4-2-1 code" or simply "BCD Code".
- It is very useful and convenient code for input and output operations in digital circuits.
- Also, it is a "weighted code system".

Example:

$$(943)_{decimal} \longrightarrow (........)_{BCD}$$

$$\Rightarrow \quad \begin{array}{ccc} 9 & 4 & 3 \\ \downarrow & \downarrow & \downarrow \\ 1001 & 0100 & 0011 \end{array}$$

$$\therefore \quad (943)_{10} = (100101000011)_2$$

Advantages of BCD Code

- The main advantage of the BCD code is relative ease of converting to and from decimal.
- Only 4-bit code groups for the decimal digits "0 through 9" need to be remembered.
- This case of conversion is especially important from the hardware standpoint.

⇒ In 4-bit binary formats, total number of possible representation = 2^4 = 16

Then, Valid BCD codes = 10

Invalid BCD codes = 6

⇒ In 8-bit binary formats,

Valid BCD codes = 100

Invalid BCD codes = 256 – 100 = 156

1.4.2 Excess-3 Code

- It is a 4-bit code.
- It can be derived from BCD code by adding "3" to each coded number.
- It is an "unweighted code".
- It is a "self-complementing code" i.e. the 1's complement of an excess-3 number is the excess-3 code for the 9's complement of corresponding decimal number.
- This code is used in arithmetic circuits because of its property of self complementing.

- Most of the digital systems perform subtraction by using 2's complement method or by using 1's complement method. The advantage of performing subtraction by using complement method is reduction in hardware i.e. instead of using separate digital circuits for addition and subtraction, only adding circuit is needed.
- To perform substraction using 2's (or 1's) complement method, represent both the subtrahend and the minuend by the same number of bits.
- Take the 2's (1's) complement of the subtrahend including the sign bit. Keep the minuend in its original form, and add the 2's (or 1's) complement of the subtrahend to it.
- In 2's complement subtraction if carry is generated ignore it. If the MSB of the sum is '0' the result is in its true binary form, however if the MSB is '1' (whether there is a carry or not), the result is negative and the magnitude is in 2's complement form.

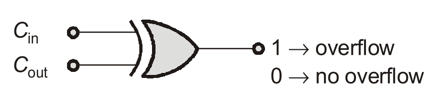
1.8 OVER FLOW CONCEPT

- If we add two same signed numbers and in the result if sign bit is opposite then it indicates "OVERFLOW".
- When "OVERFLOW" occurs, then number of bits being increased by "1-bit in MSB".

1.8.1 Overflow Condition

- If X and Y are the MSB's of two number and Z is the resultant MSB after adding two numbers then overflow condition is, $\boxed{\bar{X}\,\bar{Y}\,Z + X\,Y\,\bar{Z}}$
- In 2's complement arithmetic operation, if carry in and carry out from MSB bit position are different then it indicates overflow. $\boxed{C_{in} \oplus C_{out}}$

1.8.2 EX-OR Logic Diagram for overflow



[Since, $A \oplus A = 0$ and $A \oplus \bar{A} = 1$]

Remember

- BCD code is used in calculators, counters, digital voltmeters, digital clocks etc.
- BCD code has the advantage that it can be easily converted to and from the decimal code.
- Gray code is also known as "minimum distance code".
- A group of bits is called "word".
- "CHUNKING" is replacing a longer string by a shorter one.
- The largest number that can be represented by using *N-bits* is $(2^N - 1)_{10}$.
- In self complemented weighted code, the sum of each weightage is equal to 9.
- The weighted codes 2421, 3321, 4311 and 5211 are also self complemented code.
- Excess-3 code is unweighted self complementary code.
- Integer part will increase whenever there is conversion from higher base to lower base and vice-versa.
- Fraction part will decrease whenever there is a conversion from higher base to the lower base and vice-versa.

Student's Assignments

Q.1 If we convert a binary sequence, $(1100101.1011)_2$ into its octal equivalent as $(X)_8$, the value of 'X' will be
(a) (145.13) (b) (145.54)
(c) (624.54) (d) (624.13)

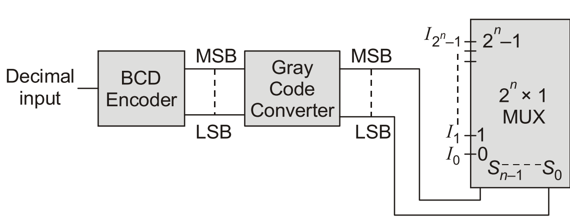
Q.2 A binary $(11011)_2$ may be represented by following ways:
1. $(33)_8$ **2.** $(27)_{10}$
3. $(10110)_{GRAY}$ **4.** $(1\,B)_H$
Which of these above is/are correct representation?
(a) 1, 2 and 3 (b) 2 and 4
(c) 1, 2, 3 and 4 (d) 2 only

Q.3 The decimal equivalent of hexadecimal number of '2A0F' is
(a) 17670 (b) 17607
(c) 17067 (d) 10767

Q.4 The sign-magnitude form and 2's complement form of a signed binary number $(10111)_2$ are:
(a) –23 and –25 (b) –23 and –9
(c) –7 and –23 (d) –7 and –9

Q.5 Given that $292_{10} = 1204$ in some number system. Which of the following represents the base of the that system?
(a) 5 (b) 6
(c) 7 (d) 8

Q.6 Consider the circuit given below:



If the decimal input is 92 then Y_{out} corresponds to I_m, then value of m is ______.

Q.7 Consider the addition of numbers with different bases $(X)_7 + (Y)_8 + (W)_{10} + (Z)_5 = (K)_9$
If $X = 36$, $Y = 67$, $W = 98$ and $K = 241$ then Z is ________.

Q.8 Which one of the following is the correct sequence of numbers represented in the series $(2)_3$, $(3)_4$, $(14)_5$, $(15)_6$?
(a) 2, 5, 10, 12 (b) 2, 3, 9, 11
(c) 3, 7, 10, 14 (d) 3, 8, 13, 17

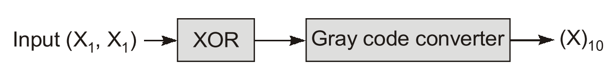
Q.9 Which of the following statement is **incorrect** for the range of n bits binary numbers?
(a) Range of unsigned numbers is 0 to $2^n - 1$.
(b) Range of signed numbers is $-2^{n-1} + 1$ to $2^{n-1} - 1$
(c) Range of signed 1's complement numbers is $-2^{n-1} + 1$ to 2^{n-1}
(d) Range of signed 2's complement numbers is -2^{n-1} to $2^{n-1} - 1$

Q.10 The base of the number system for the addition 13 + 24 = 42 to be true will be ________.

Q.11 Decimal equivalent of $(1000)_2 = -2^n$
Decimal equivalent of $(10000)_2 = -2^m$
So, $(n + m)_2$ would be
(a) 1 1 1 (b) 0 1 1
(c) 0 0 1 (d) 1 0 1

Q.12 When $(-89)_{10}$ is converted in binary, the sum of bits in binary will be ________.

Q.13 Consider the input X_1 = 10101010 and X_2 = 11111111 is feeded as input in the diagram:



Which of the following represent the value of X?
(a) +127 (b) –127
(c) –255 (d) +255

Q.14 The r's compliment of an n-digit decimal number N in base r is defined for all values of N except for $N = 0$. If the given number is $(247)_9$, then its 9's compliment will be equal to (_____ $)_9$.

Q.15 The maximum positive and negative decimal numbers that can be represented in two's complement using n-bits are
(a) $(2^{n-1} - 1)$ and $(2^{n-1} - 1)$
(b) $(2^{n-1} - 1)$ and -2^{n-1}
(c) 2^{n-1} and -2^{n-1}
(d) 2^{n-1} and $-(2^{n-1} - 1)$

Q.16 Consider the arithmetic operation performed in a particular number system whose radix is equal to 'r'.

$(23)_r + (44)_r + (14)_r + (32)_r = (223)_r$

The value of radix 'r' is equal to ________.

Q.17 A quadratic equation is formed in some number system with radix r as $x^2 - 11x + 22 = 0$. The roots of this equation are equal to $x = (3)_r$ and $x = (6)_r$ where r is the base of the number. Then, the value of r = ________.

Q.18 The representation of the value of 20 bit signed integer in 2's complement form is P = $(A72E5)_{16}$. Which of the following represents 16 × P in 1's complement representation?

(a) $(72E4F)_{16}$ (b) $(72E50)_{16}$
(c) $(72E4E)_{16}$ (d) None of the above

Q.19 The representation of the value of a 16-bit unsigned integer X in hexadecimal number system is A72E. The representation of the value of X in octal number system is

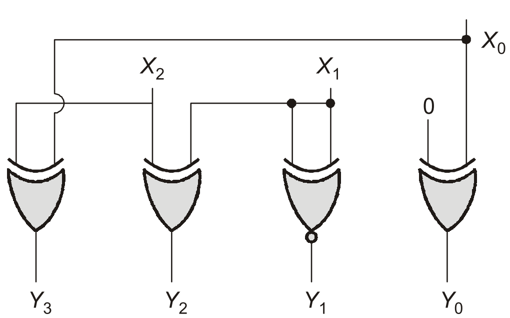
(a) 12346 (b) 123456
(c) 125756 (d) 10634

Q.20 If $(504)_X$ in base-X is equal to $(2320)_4$. Then what will be the value of base-X (in decimal) ________?

Q.21 Let A = 11111010 and B = 00001111 be two 8-bit 2's complement numbers. Their product in 2's complements is

(a) 01011010 (b) 10100110
(c) 10010010 (d) 11010101

Q.22 Consider the circuit shown in the figure below:



If the three bit input to the circuit is $(X_2X_1X_0)$ = 111 then the decimal equivalent of the corresponding output of the circuit $(Y_3Y_2Y_1Y_0)$ will be equal to ________.

Q.23 Which of the following is not true?

(a) The r's complement of a positive number N in base r is $(r^n - N)$.
(b) The $(r-1)$'s complement of a positive number N in base r is $(r^n - N - 1)$.
(c) The $(r-1)$'s complement of a positive number N having n digits and m digits in integer and fraction respectively in base r is $(r^n - r^{-m} - N)$.
(d) The $(r-1)$'s complement of a positive number N having n digit and m digits in integer and fraction part respectively in base r is $(r^n - r^m - N)$.

Q.24 How many minimum number of decimal digits is required to represent 19 bit of binary data. The number of decimal digit will be ________.

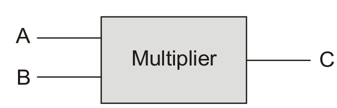
Q.25 The minimum decimal equivalent of the number $(1AC)_x$ is equal to ________.

Q.26 Consider the following arithmetic equation:

$$\frac{302}{20} = 12.1$$

The minimum possible non-zero base for the given system is ________.

Q.27 Consider a 3-bit number A and 2 bit number B are given to a multiplier. The output of multiplier is realized using AND gate and one bit full adders. If minimum number of AND gates required are X and one bit full adders required are Y, then $X + Y$ = ________.



Answer Key :

1. (b)	**2.** (c)	**3.** (d)	**4.** (d)	**5.** (b)
6. (219)	**7.** (34)	**8.** (b)	**9.** (c)	**10.** (5)
11. (a)	**12.** (5)	**13.** (a)	**14.** (642)	**15.** (b)
16. (5)	**17.** (8)	**18.** (a)	**19.** (b)	**20.** (6)
21. (b)	**22.** (3)	**23.** (d)	**24.** (6)	**25.** (311)
26. (4)	**27.** (9)			

Student's Assignments Explanations

1. (b)

We have, binary sequence $(1100101.1011)_2$

In order to convert binary number into octal equivalent we need to group the bits into triplets.

$\underline{001}\ \underline{100}\ \underline{101}.\underline{101}\ \underline{100} = (145.54)_8$

2. (c)

- In decimal $= 2^0 \times 1 + 2^1 \times 1 + 2^2 \times 0 + 2^3 \times 1 + 2^4 \times 1 = 1 + 2 + 8 + 16 = (27)_{10}$
- In octal $= \underbrace{011}_{3}\ \underbrace{011}_{3} = (33)_8$
- In hexadecimal $= \underbrace{0001}_{(1B)_{16}}\ \underbrace{1011}_{3}$
- Gray code: 1 1 0 1 1 → $(1\ 0\ 1\ 1\ 0)_{GRAY}$

Hence, all the options are true.

3. (d)

We have, hexadecimal number 2A0F to convert it into decimal number, we can do:

2	A	0	F
↓	↓	↓	↓
2	10	0	15

$16^3 \times 2 + 16^2 \times 10 + 16^1 \times 0 + 16^0 \times 15$

Which is equals to 10767.

4. (d)

In sign-magnitude from 10111 can be defined as $\underbrace{1}_{\text{sign}}\ \underbrace{0111}_{\text{magnitude}}$ i.e. –7.

In 2's complement from 10111 can be defined as – 9.

5. (b)

Let the base be x, then

$$292_{10} = 1204\,x$$
$$= 1 \times x^3 + 2 \times x^2 + 0 \times x^1 + 4 \times x^0$$
$$= 292_{10} = x^3 + 2x^2 + 4$$
$$= 6 \text{ (By substitution)}$$

6. (219)

Decimal input = 92

BCD = 10010010

Output of Gray code converter = 11011011

Y_0 corresponds to I_m with $(S_n ..... S_0)$ is

$= (11011011)_2$

$m = 219$

7. (34)

$$(36)_7 = (27)_{10}$$
$$(67)_8 = (55)_{10}$$
$$(98)_{10} = (98)_{10}$$
$$(Z)_5 = (Z)_5$$
$$(241)_9 = (199)_{10}$$
$$\therefore\ (Z)_5 = (199)_{10} - (27)_{10} - (55)_{10} - (98)_{10}$$
$$(Z)_5 = (19)_{10}$$

Converting $(19)_{10} = (34)_5$

$\therefore\ Z = 34$

8. (b)

Converting into decimal,

$$(2)_3 = 2 \times 3^\circ = 2$$
$$(3)_4 = 3 \times 4^\circ = 3$$
$$(14)_5 = 1 \times 5^1 + 4 \times 5^\circ = 9$$
$$(15)_6 = 1 \times 6^1 + 5 \times 6^\circ = 11$$

9. (c)

Range of signed 1's complement number is $-2^{n-1} + 1$ to $2^{n-1} - 1$.

10. (5)

Let base be x, then

$$(13)\,x + (24)\,x = (42)\,x$$
$$(1x^1 + 3x^0) + (2x^1 + 4x^0) = 4x^1 + 2x^0$$
$$3x^1 + 7x^0 = 4x^1 + 2x^0$$
$$x = 5$$

11. (a)

Decimal equivalent of $(1000)_2 = -2^3$

$\Rightarrow\ n = 3$

Decimal equivalent of $(10000)_2 = -2^4$

$\Rightarrow\ m = 4$

So, $(n + m)_2 = (3 + 4)_2 = (7)_2 = 111$

12. (5)

Binary representation of $(89)_{10} = (01011001)$

$(-89)_{10}$ = 2's compliment of (01011001)