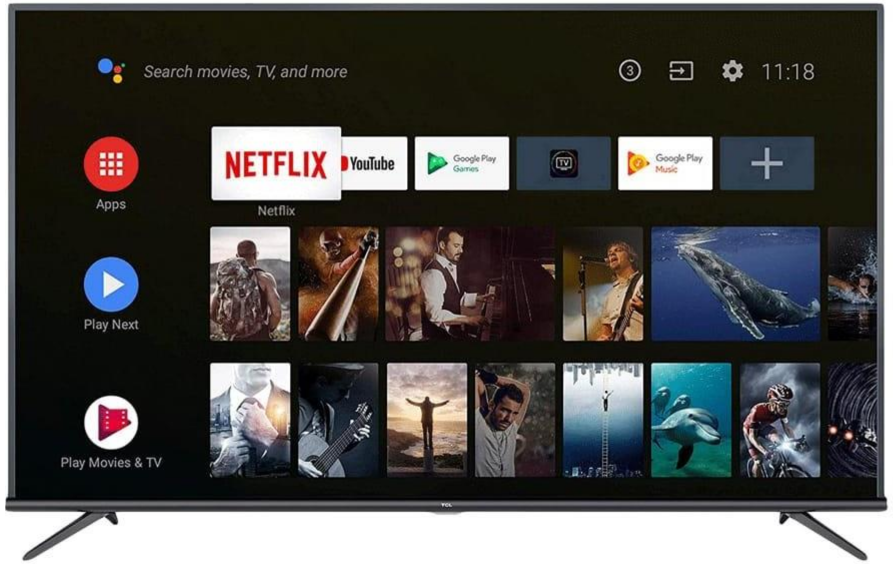
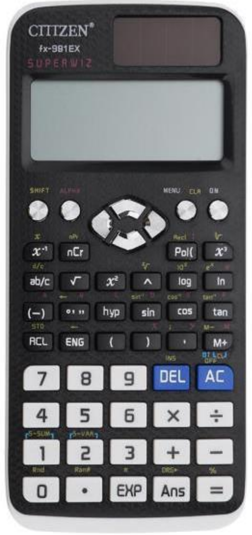
Outline

- ❑ Analog and Digital representation
- ❑ Binary number system
- ❑ Decimal to binary conversion
- ❑ Signed binary numbers (2's complement system)
- ❑ Binary addition and subtraction
- ❑ Octal and Hexadecimal number system
- ❑ Solved Examples and Exercises

DIGITAL TECHNOLOGIES





Digital devices connected via the Internet

NUMERICAL REPRESENTATIONS

There are two ways of representing the numerical value of quantities

1. Analog

- Quantities are represented by a **continuously variable**, proportional indicator
- Can vary over a **continuous range** of values (e.g. real numbers)

2. Digital

- Quantities are represented not by continuously variable indicators but by **symbols** called **digits**
- Changes in **discrete steps** (e.g. integer values)

ANALOG AND DIGITAL REPRESENTATIONS

Speedometer with
Analog Representation

Speedometer with
Digital Representation

Analog and Digital Representations

Voltmeter with
Analog Representation

Voltmeter with
Digital Representation

ANALOG AND DIGITAL REPRESENTATIONS

Thermometer with
Analog Representation

Thermometer with
Digital Representation

ANALOG AND DIGITAL REPRESENTATIONS

Clock with
Analog Representation

Clock with
Digital Representation

ANALOG AND DIGITAL REPRESENTATIONS

Weighing Scale with
Analog Representation

Weighing Scale with
Digital Representation

ADVANTAGES OF DIGITAL TECHNOLOGY

- **Digital** systems are generally **easier to design**
  - Exact values of voltage or current are not important
- **Information storage** is **easy**
  - Mass storage techniques that can store billions of bits of information in a relatively small physical space e.g. flash drives
  - Analog storage capabilities are, by contrast, extremely limited
- **Accuracy** and **precision** are **easier to maintain** throughout the system
  - Noise doesn't affect the information too much as it is being processed, easier to keep track of integers than exact real numbers

ADVANTAGES OF DIGITAL TECHNOLOGY

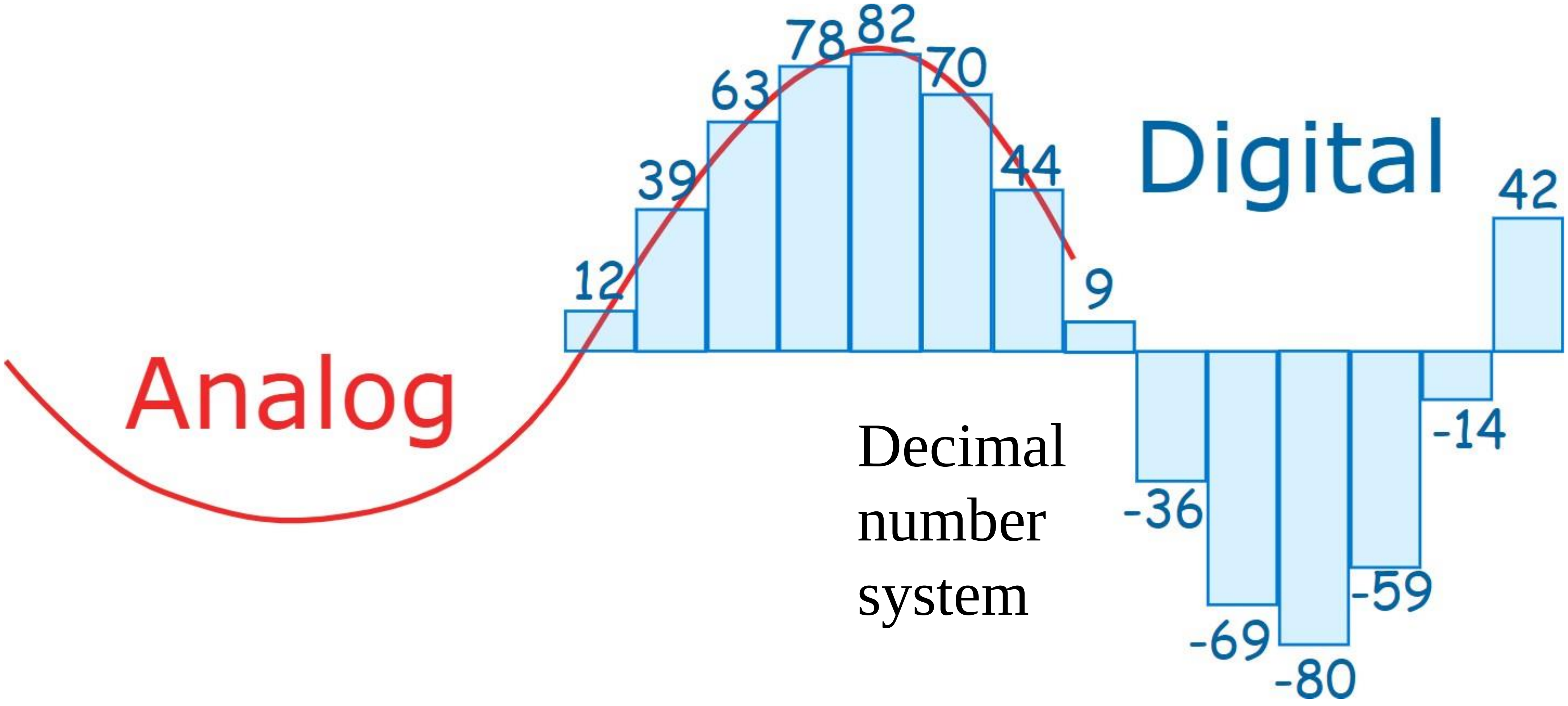
- **Operation**/computations can be **programmed**
  - Fairly easy to design digital systems whose operation is controlled by a set of stored instructions (integer manipulations) called a program
- Digital circuits are **less affected by noise**
  - Since the exact value of a voltage/current is not important ("round-off" error can be tolerated)
- More **digital circuitry** can be fabricated on **IC chips**
  - Analog systems have not achieved the same high degree of integration

LIMITATIONS OF DIGITAL TECHNOLOGY

Very few drawbacks when using digital techniques

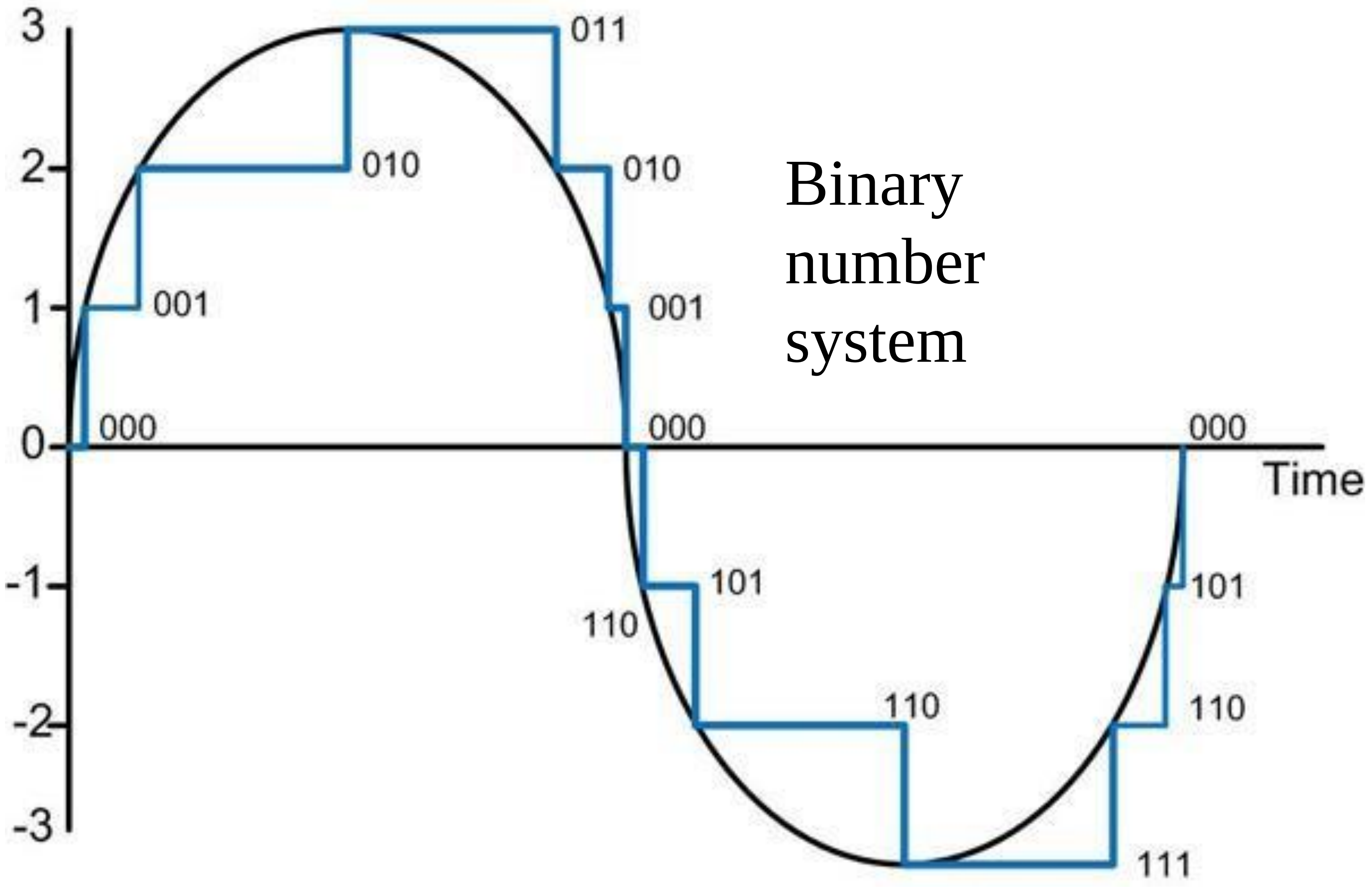
- The **real world** is **analog** (physical quantities have real number values)
  - The inputs have to be converted from analog to digital quantities using analog to digital converters (ADC)
  - The outputs have to be converted from digital to analog quantities using digital to analog converters (DAC)
- **Processing** digitized signals can thus take **more time**
- There are **specific situations**, where use of **analog** techniques is **faster**, **simpler** or **more economical**

ANALOG AND DIGITAL SIGNALS



Decimal number system

ANALOG AND DIGITAL SIGNALS



Binary
number
system

Digital Systems Design

Discrete Logic

✓ a circuit constructed using small-scale integrated (SSI) and medium-scale integrated (MSI) logic devices (NAND gates, decoders, multiplexers, etc.)

Programmable Logic Device (PLD)

✓ an integrated cirucit onto which a generic logic circuit can be programmed (an subsequently erased and re-programmed)

Field Programmable Gate Array (FPGA)

✓ an SRAM-based PLD that can be programmed in-circuit (no need to "erase" since SRAM-based)

VHDL and VERILOG

✓ advanced hardware simulation and description languages

Levels of abstraction in computer hardware

Processor level

- CPU, I/O Processor, Graphic processor, RAM, etc
- It is implemented by using LSI/VLSI technology
- the processor level handle large blocks of data at a time

Register level

- registers, simple combinational and sequential logic circuits, bit-sliced ALU, etc
- the register level handle few bytes of data at a time

Gate level

- logic gates, flip-flops, etc
- the gate level handle bits of a data at a time

Device level

- transistors and manufacturing technology such as bipolar, CMOS, and speed, power consumption, heat dissipation and other issues.
- the device level handle electrical signals and several of them are needed to handle a bit of data

What does a Computer Understand ?

* Computers do not understand natural human languages, nor programming languages
* They only understand the language of bits

Bit	0 or 1
Byte	8 bits
Word	4 bytes
kiloByte	1024 bytes
megaByte	10^6 bytes

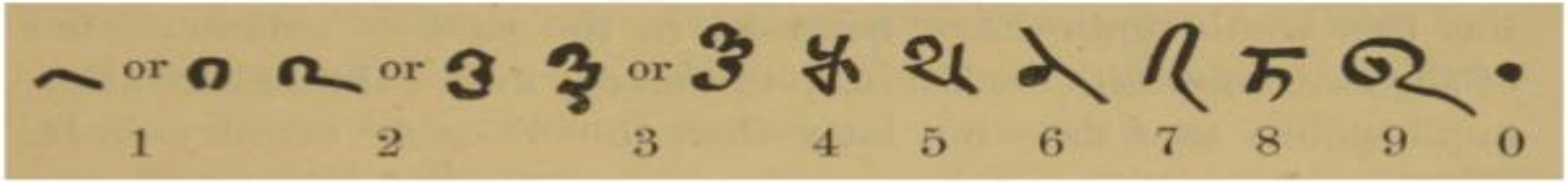
Representing Positive Integers

* Ancient Roman System

Symbol	I	V	X	L	C	D	M
Value	1	5	10	50	100	500	1000

* Issues :
  * There was no notion of 0
  * Very difficult to represent large numbers
  * Addition, and subtraction (very difficult)

Indian System



Bakshali numerals, 7th century AD

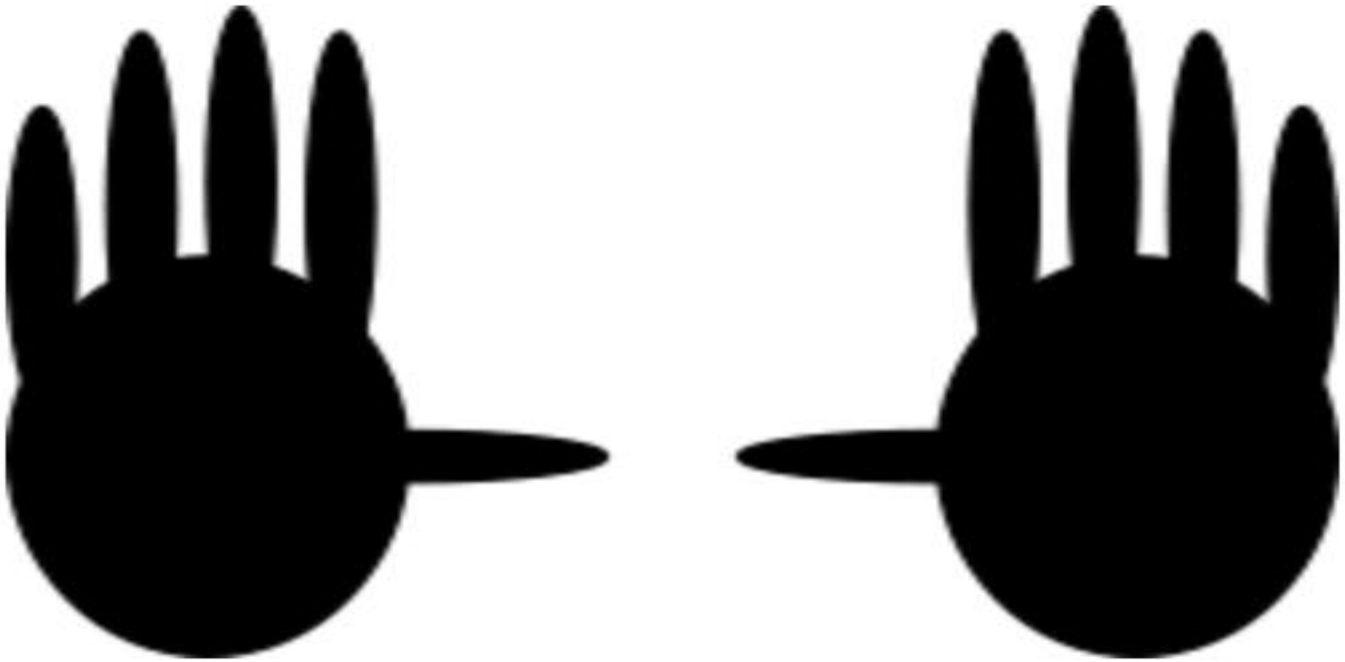
* Uses the place value system

$$5301 = 5 * 10^3 + 3 * 10^2 + 0 * 10^1 + 1*10^0$$

Example in base 10

Number Systems in Other Bases

* Why do we use base 10 ?
  * because ...



What if we had a world in which ...

- People had only two fingers.
  - They would use a number system with base 2.

Number in decimal	Number in binary
5	101
100	1100100
500	111110100
1024	10000000000

Base/ Radix

$\boxed{(\text{Number})_{\text{Base}}}$

For decimal number system, Number $\rightarrow$ 0 to 9, *i.e.* $(\text{Number})_{10}$
and Base $\rightarrow$ 10.
For binary number system, Number $\rightarrow$ 0 to 1, *i.e.* $(\text{Number})_2$
and Base $\rightarrow$ 2.

$$(138)_{10} = \underbrace{1 \times 10^2}_{\text{Hundreds}} + \underbrace{3 \times 10^1}_{\text{Tens}} + \underbrace{8 \times 10^0}_{\text{Ones}}$$

$$\begin{aligned}
(138)_{10} &= (10001010)_2 \\
&= 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\
&= 128 + 0 + 0 + 0 + 8 + 0 + 2 + 0 \\
&= 138
\end{aligned}$$

MCQ

The decimal system has a base of __________.

a. 0

b. 10

c. 100

d. 1000

Which digit represents "hundreds" in the number 8732?

a. 8

b. 7

c. 3

d. 2

MCQ

Which of the following is correct?

a. $25 = (2 \times 10^2) + (5 \times 10^1)$

b. $289 = (2 \times 10^3) + (8 \times 10^1) + (9 \times 10^0)$

c. $7523 = (7 \times 10^3) + (5 \times 10^2) + (2 \times 10^1) + (3 \times 10^0)$

d. $0.628 = (6 \times 10^{-3}) + (2 \times 10^{-2}) + (8 \times 10^{-1})$

In the number 3109, the 3 is referred to as the __________.

a. most significant digit [MSB]

b. least significant digit [LSB]

c. radix

d. base

MCQ

In the number 3109, the 9 is referred to as the __________.

a. most significant digit [MSB]

b. least significant digit [LSB]

c. radix

d. base

Numbers in the binary system are represented to the __________.

a. base 0

b. base 1

c. base 2

d. base 10

MCQ

Hexadecimal has a base of __________.

a. 2

b. 8

c. 10

d. 16

Another term for "base" is ___________.

a. radix

b. integer

c. position

d. digit

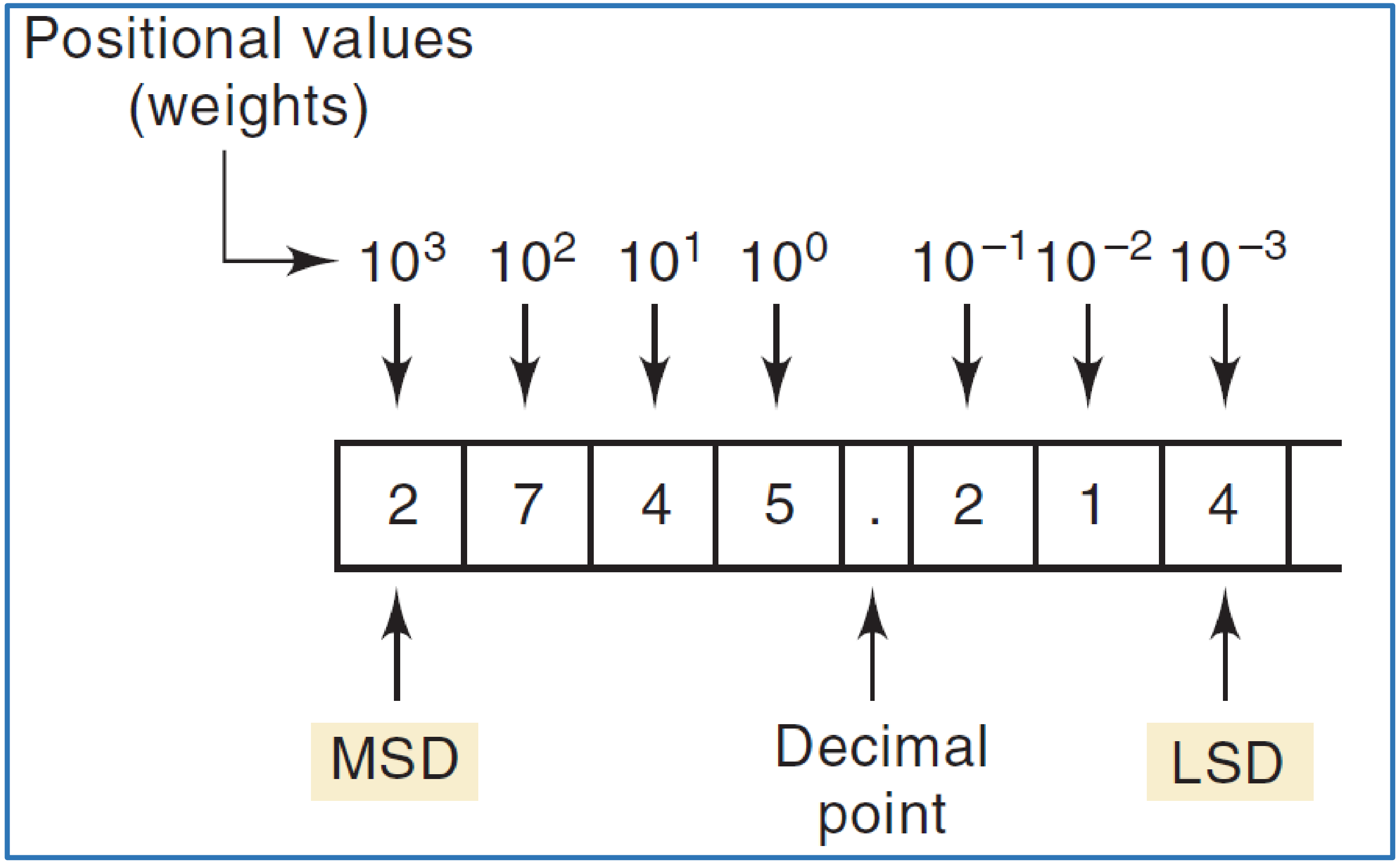
DIGITAL NUMBER SYSTEMS - DECIMAL SYSTEM

- The decimal system is composed of 10 numerals or symbols
- These 10 symbols are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Also called the **base-10** system because it has 10 digits
- Positional-value system in which the value of a digit depends on its position
- Various positions relative to the decimal point carry weights that can be expressed as **powers of 10**

DIGITAL NUMBER SYSTEMS - DECIMAL SYSTEM

- Consider the number **2745.214**; it is equal to

$$(2\times10^{+3})+(7\times10^{+2})+(4\times10^{+1})+(5\times10^{+0})+(2\times10^{-1})+(1\times10^{-2})+(4\times10^{-3})$$



- MSD is "most significant digit"
- LSD is "least significant digit"

DIGITAL NUMBER SYSTEMS - BINARY SYSTEM

- The **decimal** number system does **not** lend itself to **convenient** physical **implementation** in digital systems
- It is **very difficult** to design electronic equipment so that it can work with **10 different voltage levels**
- It is **very easy** to design simple, accurate electronic circuits that operate with only **two voltage levels** e.g. **transistor ON and OFF**
- In the **binary** system, there are only two symbols or possible digit values, **0 and 1**.

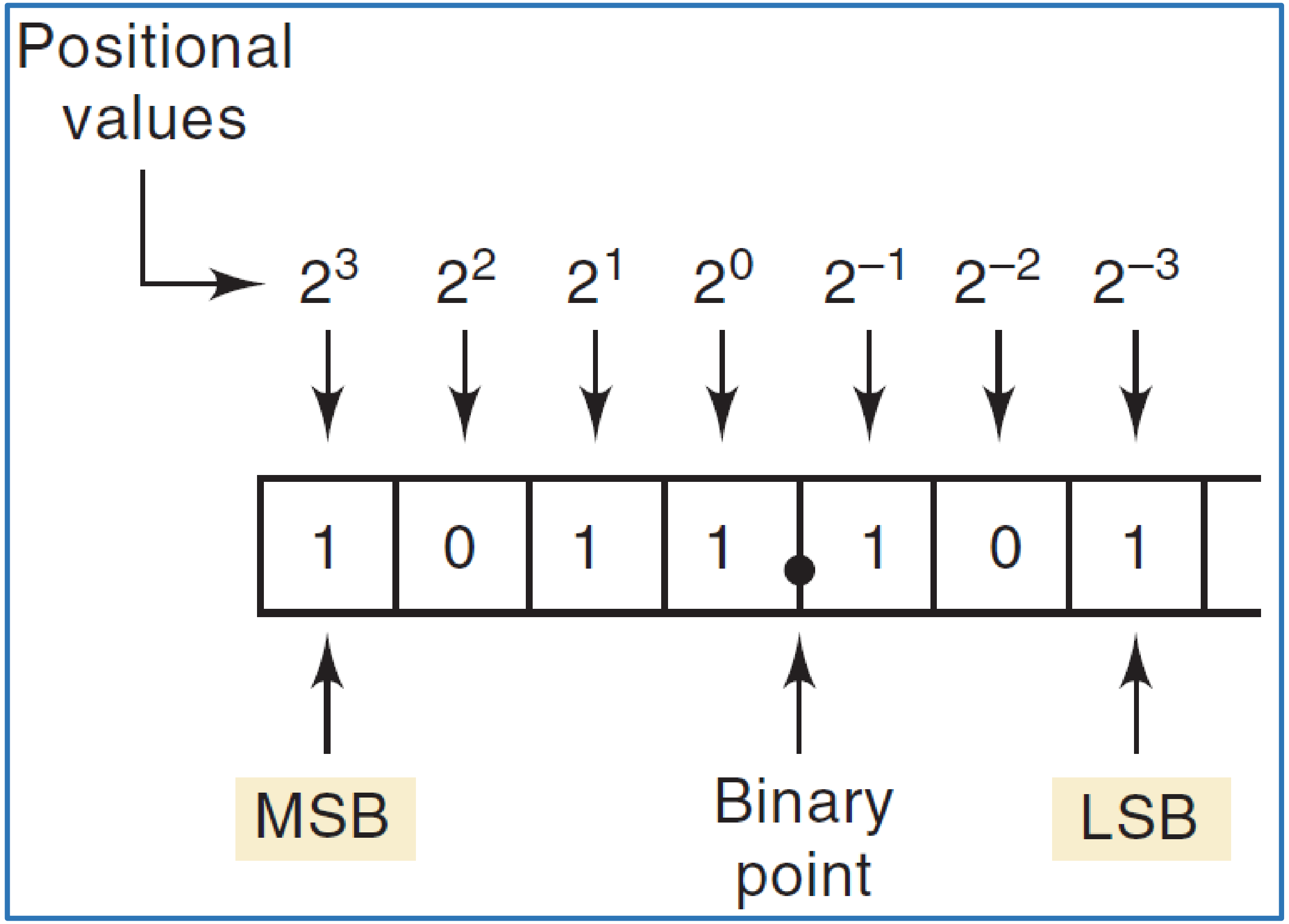
BINARY NUMBER SYSTEM

- The binary system is a positional value system
- Each binary digit has a positional value or **weight** expressed as a **power of 2**
- Places to the **left of the binary point** are **positive powers of 2**, and places to the **right** are **negative powers of 2**
- The term **binary digit** is often abbreviated to the term “**bit**”
- The most significant bit (**MSB**) is the **leftmost** bit (largest weight)
- The least significant bit (**LSB**) is the **rightmost** bit (smallest weight)

Binary Number System

- Consider the **binary** number **1011.101**; it's **decimal** equivalent is

$$1011.101_2 = (1\times 2^{+3}) + (0\times 2^{+2}) + (1\times 2^{+1}) + (1\times 2^{+0}) + (1\times 2^{-1}) + (0\times 2^{-2}) + (1\times 2^{-3}) = 11.625_{10}$$



BINARY TO DECIMAL CONVERSION

- Any binary number can be converted to its decimal equivalent simply by **summing** together the **weights of** the various **positions** in the binary number that **contain** the digit "**1**"

- e.g. To convert **10110101** from binary to decimal

$$\boxed{\begin{array}{l} 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1_2 = \\ 2^7 + 0 + 2^5 + 2^4 + 0 + 2^2 + 0 + 2^0 = 181_{10} \end{array}}$$

DECIMAL TO BINARY CONVERSION: METHOD 1

- The **decimal** number is simply expressed as a **sum of powers of 2**, and then 1s and 0s are written in the appropriate bit positions

- e.g. To convert 45 from decimal to binary

$$\begin{aligned} 45_{10} &= 32 + 8 + 4 + 1 = 2^5 + 0 + 2^3 + 2^2 + 0 + 2^0 \\ &= 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1_2 \end{aligned}$$

- Note that a 0 is placed in the 2^1 and 2^4 positions, since all positions must be accounted for

DECIMAL TO BINARY CONVERSION: METHOD 2

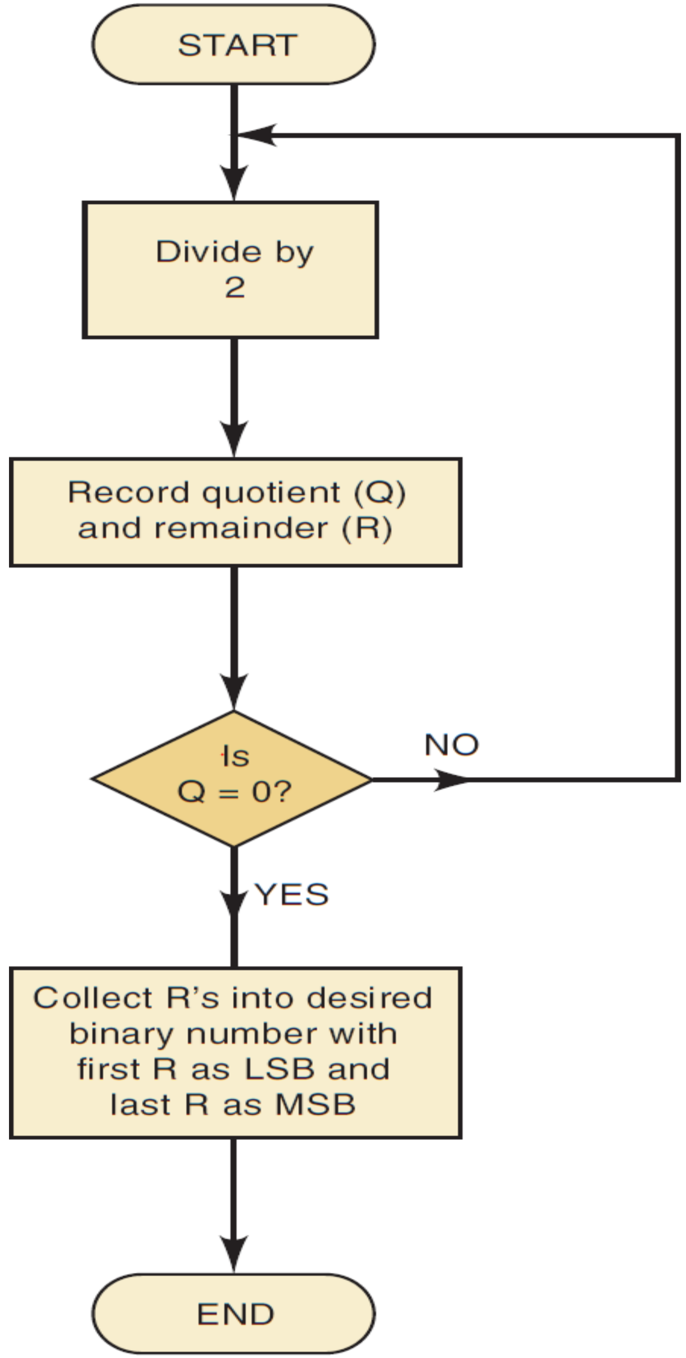
- The **repeated division** method requires repeatedly dividing the **decimal number by 2**, and writing down the **remainder after each division**, until a quotient of 0 is obtained

- The **binary** result is obtained by writing the **first remainder as the LSB** and the **last remainder as the MSB,** and all the other remainders in between in sequence

- Using **N bits**, we can represent all the **decimal numbers** from **0 to 2^N -1**, i.e. a total of 2^N different numbers

Decimal to Binary Conversion:

Method 2 (Repeated Division by 2)



DECIMAL TO BINARY CONVERSION: METHOD 2

Example:

Convert **25**

from decimal

to binary

representation.

$$\frac{25}{2} = \boxed{12} + \text{remainder of } 1 \quad \text{LSB}$$

$$\frac{\boxed{12}}{2} = 6 + \text{remainder of } 0$$

$$\frac{6}{2} = 3 + \text{remainder of } 0$$

$$\frac{3}{2} = 1 + \text{remainder of } 1$$

$$\frac{1}{2} = 0 + \text{remainder of } 1 \quad \text{MSB}$$

$$25_{10} = \boxed{1\ 1\ 0\ 0\ 1_2}$$

Examples

Convert $(320)_{10}$ into binary number system.

Solution $(320)_{10} = (?)_2$

2	320	—	× —
2	160	—	0 (LSB)
2	80	—	0
2	40	—	0
2	20	—	0
2	10	—	0
2	5	—	0
2	2	—	1
2	1	—	0
	0	—	1 (MSB)

(reading from bottom to top)

MSB

$$\boxed{(320)_{10} = (101000000)_2}$$

Digital Number Systems - Octal System

- As decimal numbers become larger, the corresponding binary number takes up more and more digits. So alternate number systems with base as some power of 2 are used
- e.g. The **octal system** has **eight** ($8 = 2^3$) **digits** from 0 to 7
- Each octal digit represents a power of 8
- Example: Convert **236** from octal to decimal

$$\boxed{236_8 = (2\times 8^{+2}) + (3\times 8^{+1}) + (6\times 8^{0}) = 158_{10}}$$

Examples

Convert $(12)_{10}$ into octal number

8	12		
8	1	—	4
	0	—	1

Octal equivalent $= 14$

$(12)_{10} = (14)_8$

Convert $(570)_{10}$ into octal number

Divisor	I.P.	Remainder
8	570	—
8	71	2
8	8	7
8	1	0
	0	1

LSB

Octal equivalent $= 1072$

MSB

$(570)_{10} = (1072)_8$

Convert $(1052)_8$ to decimal

$$\begin{aligned}(1052)_8 &= 1 \times 8^3 + 0 \times 8^2 + 5 \times 8^1 + 2 \times 8^0 \\ &= 512 + 0 + 40 + 2\end{aligned}$$

$\boxed{(1052)_8 = (554)_{10}}$ **Ans.**

DIGITAL NUMBER SYSTEMS - OCTAL SYSTEM

- To **convert octal** number **to binary** number, **replace** each **octal digit** by its **three digit binary** equivalent.
- Example: $415_8 = 100\ 001\ 101_2$

- To **convert binary** number **to octal** number, take **groups of three binary digits**, starting from the LSB, and write their **decimal equivalent**.
- Example: $\boxed{10111010_2 = \underline{010}\ 1\underline{11}\ \underline{010} = 272_8}$

DIGITAL NUMBER SYSTEMS - HEXADECIMAL SYSTEM

- The hexadecimal number system uses **base-16** (= 2^4).
  Thus, it has 16 possible digit symbols.
- It uses the digits **0 through 9** plus the letters **A, B, C, D, E, and F**.
- The digit positions are weighted as powers of 16.
- Digits A through F are equivalent to the decimal values 10 through 15
- A hex number can be converted to its decimal equivalent by using the fact that each hex digit position has a weight that is a power of 16.
- Example: $\boxed{2AF_{16} = (2\times16^{+2}) + (10\times16^{+1}) + (15\times16^{0}) = 687_{10}}$

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

$\ldots x_1$	x_2	x_3	x_4	$\bullet$	x_5	x_6	x_7	$x_8\ldots$
16^3	16^2	16^1	16^0	$\bullet$	16^{-1}	16^{-2}	16^{-3}	16^{-4}

Radix point

Examples

Convert $(235.25)_{10}$ into hexadecimal.

Solution Integral part

16	235	Remainder
	14	11 (B in Hexadecimal)
	0	14 (E in Hexadecimal)

$\uparrow$ Hexadecimal equivalent

$$\therefore \quad (235)_{10} = (EB)_{16}$$

Fractional part

$$0.25 \times 16 = 0 \quad \text{with a carry } 4 \downarrow$$

$$\therefore \quad (0.25)_{10} = (0.4)_{16}$$

$$\therefore \quad (235.25)_{10} = (EB.4)_{16}$$

$\ldots x_1$	x_2	x_3	x_4	$\bullet$	x_5	x_6	x_7	$x_8\ldots$
16^3	16^2	16^1	16^0	$\bullet$	16^{-1}	16^{-2}	16^{-3}	16^{-4}

Radix point

HEXADECIMAL NUMBER SYSTEM

- To convert a number from **hexadecimal to binary**, each **hex digit** is converted to its **four-bit binary** equivalent

- Example: Convert **(9F2)**$_{16}$ to binary

$$
\begin{array}{lcccc}
9F2_{16} = & 9 & & F & & 2 \\
 & \downarrow & & \downarrow & & \downarrow \\
 & \overline{1\quad 0\quad 0\quad 1} & & \overline{1\quad 1\quad 1\quad 1} & & \overline{0\quad 0\quad 1\quad 0} \\
\end{array}
$$

$$= 1\;0\;0\;1 \quad 1\;1\;1\;1 \quad 0\;0\;1\;0$$

$$= 100111110010_2$$

HEXADECIMAL NUMBER SYSTEM

- To convert a number from **binary to hexadecimal**, the binary number is grouped into **groups of four bits** starting from the right-most (LSB), and each group is converted to its **equivalent hex digit**
- Zeros (shown shaded) are added, as needed, to complete a four-bit group

$$1\,1\,1\,0\,1\,0\,0\,1\,1\,0_2 = \underbrace{0\;0\;1\;1}_{3}\;\underbrace{1\;0\;1\;0}_{A}\;\underbrace{0\;1\;1\;0}_{6}$$

$$= 3A6_{16}$$

HEXADECIMAL NUMBER SYSTEM

$$
\begin{aligned}
(37)_8 &= 011 \quad 111 \\
&= \underbrace{0001}_{1}\ \underbrace{1111}_{F} = (1F)_{16} \quad \text{(Two zeros added in MSB)}
\end{aligned}
$$

Thus the rule is
Hexa $\rightleftarrows$ (4-bit binary group) $\rightleftarrows$ (3-bit binary group) $\rightleftarrows$ octal.

S.N.	Decimal	Binary	Octal	Hexa-decimal
1	0	00	0	0
2	1	01	1	1
3	2	$(10)_2 = (2)_{10}$	2	2
4	3	11	3	3
5	4	$(100)_2 = (4)_{10}$	4	4
6	5	101	5	5
7	6	110	6	6
8	7	111	7	7
9	8	$(1000)_2 = (8)_{10}$	10	8
10	9	1001	11	9
11	10	1010	12	A
12	11	1011	13	B
13	12	1100	14	C
14	13	1101	15	D
15	14	1110	16	E
16	15	1111	17	F
17	16	$(10000)_2 = (16)_{10}$	20	$(10)_{16} = (16)_{10}$
18	17	10001	21	11
19	18	10010	22	12

EXERCISES

Convert $(235)_{10}$ into hexadecimal.

16	256	Remainder
	15	13 (D in Hexadecimal)
	0	15 (F in Hexadecimal)

$$\therefore \quad (253)_{10} = (FD)_{16}$$

Convert $(A13B)_{16}$ into decimal.

$$\overset{3}{(A}\ \overset{2}{1}\ \overset{1}{3}\ \overset{0}{B})_{16} = A\times16^3 + 1\times16^2 + 3\times16^1 + B\times16^0$$

$$= 10\times16^3 + 256 + 48 + 11\times1$$

$$= 40960 + 256 + 48 + 11$$

$$= (41275)_{10}$$

Convert $(EB4A)_{16}$ into decimal.

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

$$(EB4A)_{16} = E \times 16^3 + B \times 16^2 + 4 \times 16^1 + A \times 16^0$$

$$= 14 \times 16^3 + 11 \times 16^2 + 64 + 10 \times 1$$

$$= 57{,}344 + 2816 + 64 + 10$$

$$= (60234)_{10}$$

Solved Examples

1. How many bits are there in one byte?

Ans. There are 8 bits in one byte.

2. Convert the binary number 101.011_2 to decimal.

Ans. $101.011_2 = (1\times2^2) + (0\times2^1) + (1\times2^0) + (0\times2^{-1}) + (1\times2^{-2}) + (1\times2^{-3})$

$= 4 + 1 + 0.25 + 0.125 = 5.375_{10}$

3. Convert 101011110010 from binary to hexadecimal?

Ans. $101011110010_2 = \underline{1010}\ \ \underline{1111}\ \ \underline{0010} = AF2$

EXERCISES

1. What is the decimal equivalent of 1101001011_2? Ans. 843

2. What is the binary equivalent of 485_{10}? Ans. 111100101

3. What is the largest decimal number that can be represented using ten binary bits? Ans. 1023

4. How many bits are needed to represent decimal values ranging from 0 to 14,500? Ans. 14

EXERCISES

5. Convert the octal number 250_8 to decimal. Ans. 168

6. Convert 451_8 to binary. Ans. 100101001

7. Convert 11010110_2 to octal. Ans. 326

8. Convert the hexadecimal number $AD2B_{16}$ to decimal. Ans. 44331

9. Convert $FA2_{16}$ to binary. Ans. 111110100010

10. Convert 10111011110_2 to hexadecimal. Ans. 5DE