# University of Anbar <br> College of Engineering / Electrical Department 

## Digital Techniques Lectures

## By

Mohammed Khaleel

## Number Systems:

Many number systems are in use in digital technology. The most common are the decimal, binary, octal, and hexadecimal systems. The decimal system is clearly the most familiar to us because it is the tool that we use every day. Examining some of its characteristics will help us to better understand of other systems.

## Decimal System:

The decimal system is composed of 10 numerals or symbols. These symbols are $0,1,2,3,4,5,6,7,8,9$; using these symbols as digits of a number, we can express any quantity. The decimal system is also called the base-10 system. The decimal system is a positional-value system in which the value of a digit depends on its position. For example, consider the decimal number 453 . We know that the digit 4 actually represents 4 hundreds (400), the 5 represents 5 tens (50), and 3 represents 3 units.

In essence, the 4 carries the most weight of these digits; it's referred to us the most significant digit (MSD). The 3 carries the least weight and is called the least significant digit (LSD).

Example: consider the decimal number 27.35 ; this number is actually equal to: $2 \times 10+7 \times 1+3 \times 0.1+5 \times 0.01$

The decimal point is used to separate the integer and the fractional parts of the number.


## Binary System:

In the binary system there are only two symbols or possible digit values, 0 and 1 . Even so, this base- 2 system can be used of represent any quantity can be represented in decimal or other number systems. The binary number system of Table (1) is nothing more than a code. After some practice, it becomes almost as familiar as the decimal number system.

Table (1)

| Table (1) |  |  |
| :---: | :---: | :---: |
| Quantity | Binary No. | Decimal No. |
| None | 0 | 0 |
| . | 1 | 1 |
| .. | 10 | 2 |
| $\ldots$ | 11 | 3 |
| $\ldots$. | 100 | 4 |
| $\ldots .$. | 101 | 5 |
| $\ldots .$. | 110 | 6 |
| $\ldots \ldots$. | 111 | 7 |

## Binary-to-Decimal Conversion:

The binary number system is a positional system where each binary digit (Bit) carries a certain weight based on its position relative to the LSB (Least Significant Bit). Any binary number can be converted to its decimal equivalent simply by summing together the weights of the various positions in the binary number which contain a 1 . To illustrate:

$$
\begin{array}{rlrll}
\mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\
\mathbf{2}^{\mathbf{4}}+\mathbf{2}^{\mathbf{3}}+\mathbf{2}^{\mathbf{2}}+\mathbf{2}^{\mathbf{1}}+\mathbf{2}^{\mathbf{0}} & =16+8+2+1 \\
& & & & \\
& & \\
\text { (Dinary) }
\end{array}
$$

Let's try another example with greater number of bits.

$$
\begin{array}{llll}
\mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{1}
\end{array} \mathbf{0} \quad \mathbf{1} \quad \mathbf{0} \quad \mathbf{1}, \quad \text { (Binary) }
$$



Example: find the decimal equivalent of the 0.1101 ?
As far as mixed numbers are concerned (number that have an integer and fractional part), the weights for a mixed number are:


Hence: $0.1101_{(2)}=0.8125_{(0)}$
Example: convert binary 110.001 to a decimal number

$$
\begin{array}{ccccccccc}
\mathbf{1} & \mathbf{1} & \mathbf{0} & . & \mathbf{0} & \mathbf{0} & \mathbf{1} & & \\
\mathbf{2}^{2} & \mathbf{2}^{1} & \mathbf{2}^{\mathbf{0}} & . & \mathbf{2}^{-1} & 2^{-2} & \mathbf{2}^{-3} & \longrightarrow & \\
6.125
\end{array}
$$

## Decimal-to-Binary Conversion:

One way to convert a decimal number into its binary equivalent is to reverse the process described in the binary to decimal conversion paragraph. For instance, suppose you want to convert decimal 9 into the corresponding binary number. All you need to do is express 9 as a sum of power of 2 , and then write 1 's and 0 's in the appropriate positions.

$$
\begin{aligned}
9=8+1 & =8+0+0+1 \\
& =2^{3}+2^{2}+2^{1}+2^{0}
\end{aligned}
$$

$$
\begin{array}{llllll}
=1 & 0 & 0 & 1 & & \text { (Binary })
\end{array}
$$



As another example,

$$
\begin{aligned}
& 25=16+8+1=\mathbf{1 6}+\mathbf{8}+\mathbf{0}+\mathbf{0}+\mathbf{1} \\
& 110001 \text { (Binary) }
\end{aligned}
$$

Amore popular way to convert decimal number to binary numbers is the (repeated division). This method requires repeatedly dividing the decimal number by 2 and writing down the reminders after each division until a quotient of 0 is obtained. Note that the binary result is obtained by writing the first reminder as the LSB and the last reminder as the MSB. Let us convert decimal 25 to its binary equivalent using the repeated division method.


As far as fractions are concerned, it is possible to multiply by 2 and record a carry in the integer position. As an example, convert 0.625 to binary fraction.

By taking the carries in forward order, we get ( 0.101 ) which is the binary equivalent of 0.625



Example: convert $21.6_{(10)}$ to a binary number.
Split 21.6 into an integer of 21 and a fraction of 0.6 , and apply repeated division to each part.

hence: $21.6_{(10)}=\left(\begin{array}{llllll}1 & 0 & 1 & 01.10011\end{array}\right)_{2}$

## Octal Number System:

The octal number system is very important in digital computer work. the octal number system has a base of 8 , meaning it has eight possible digits: $0,1,2,3,4,5,6,7$. Thus, each digit of an octal number can have any value from 0 to 7 . The digital positions in an octal number have weights as follows:



## Octal-to-Decimal Conversion:

An octal number can be easily converted to its decimal equivalent by multiplying each octal digit by its positional weight. For example:

$$
\begin{aligned}
372_{(8)} & =3 \times 8^{2}+7 \times 8^{1}+2 \times 8^{0} \\
& =3 \times 64+7 \times 8+2 \times 1=250_{(10)}
\end{aligned}
$$

Example: convert $24.6_{(8)}$ to its decimal equivalent.

$$
\begin{aligned}
24.6_{(8)} & =2 \times 8^{1}+4 \times 8^{0}+6 \times 8^{-1} \\
& =2 \times 8+4 \times 1+6 \times 0.125=20.75_{(10)}
\end{aligned}
$$

## Decimal-to-Octal Conversion:

A decimal integer can be converted to octal by using the same repeated division method that have been used in the decimal-to-binary conversion, but with a division factor of 8 instead of 2 . An example is shown below:


For decimal fractions, multiplying instead of dividing, writing the carry into the integers position. An example of this is to convert 0.23 into an octal fraction.


