Figure 2.1 Positional values in decimal numbers.

### 2.2.2 Binary Number System

Unfortunately, the decimal number system does not lend itself to convenient implementation in digital systems. For example, it is very difficult to design electronic equipment so that it can work with 10 different voltage levels (each one representing one decimal character, 0 through 9). On the other hand, it is very easy to design simple, accurate electronic circuits that operate with only two voltage levels. For this reason, almost every digital system uses the binary number system (base 2) as the basic number system of its operations, although other systems are often used in conjunction with binary.

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 to represent any quantity that can be represented in decimal or other number systems.

The binary system is also a positional-value system, wherein each binary digit has its own value or weight expressed as a power of 2. This is illustrated in Fig. 2.2.

Here, places to the left of the binary point (counterpart of the decimal point) are positive powers of 2 and places to the right are negative powers of 2. The number 1010.0101 is shown represented in the figure.

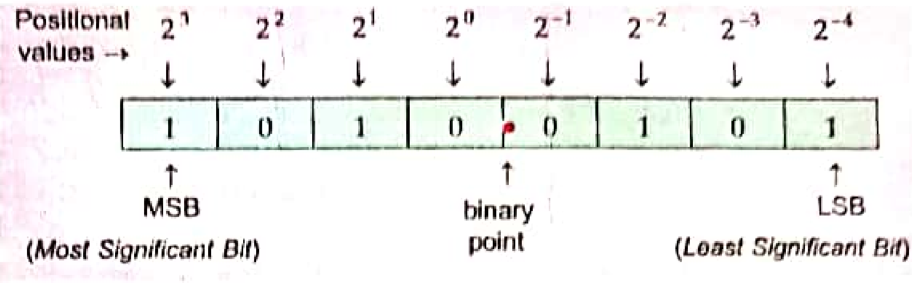


Figure 2.2 Positional values in binary numbers.

To find the decimal equivalent of above shown binary number, we simply take the sum of the products of each digit value (0 or 1) and its positional value :

$$1010.0101_2 = (1\times 2^3)+(0\times 2^2)+(1\times 2^1)+(0\times 2^0)+(0\times 2^{-1})+(1\times 2^{-2})+(0\times 2^{-3})+(1\times 2^{-4})$$
$$= 8+0+2+0+0+0.25+0+0.0625 = 10.3125_{10}$$

Notice in the preceding operation that subscripts (2 and 10) were used to indicate the base in which the particular number is expressed. This convention is used to avoid confusion whenever more than one number system is being employed.

In the binary system, the term *Binary digit* is often abbreviated to the term *bit*, which we'll use henceforth. As you see in Fig. 2.2, there are 4 bits to the left of the binary point, representing the integer part of the number, and 4 bits to the right of the binary point, representing the fractional part. The leftmost bit carries the largest weight and hence, is called the **most significant bit (MSB).** The rightmost bit carries the smallest weight, and hence called **least significant bit (LSB).**

The sequence of binary numbers goes as 00, 01, 10, 11, 100, 101, 110, 111, 1000, - - - - - -. The binary counting sequence has an important characteristic. The unit's bit (LSB) changes either from 0 to 1 or 1 to 0 with each count. The second bit (two's ($2^1$) position) stays at 0 for two counts, then at 1 for two counts, then at 0 for two counts, and so on. The third bit (four's ($2^2$) position) stays at 0 for four counts, then at 1 for four counts, and so on. The fourth bit (eight's ($2^3$) position) stays at 0 for eight counts, then at 1 for eight counts. If we wanted to count further we would add more places, and this pattern would continue with 0s and 1s alternating in groups of $2^{N-1}$.