### 2.3.1 Conversion of Decimal to Binary

The method, which is used for the conversion of decimal into binary, is often called as the remainder method. This method involves the following steps:

1. Begin by dividing the decimal number by 2 (the base of binary number system).
2. Note the remainder separately as the rightmost digit of the binary equivalent.
3. Continually repeat the process of dividing by 2 until the quotient is zero and keep writing the remainders after each step of division (these remainders will either be 1 or 0).
4. Finally, when no more division can occur, write down the remainders in reverse order (last remainder written first).

**Example 1:** Determine the binary equivalent of $(36)_{10}$.

| | | | |
|---|---|---|---|
| 2 | 36 | **Remainder** | |
| 2 | 18 | 0 | **Least Significant Bit (LSB)** ↑ |
| 2 | 9 | 0 | |
| 2 | 4 | 1 | |
| 2 | 2 | 0 | |
| 2 | 1 | 0 | |
| | 0 | 1 | **Most Significant Bit (MSB)** |

Taking remainders in reverse order, we have 100100. Thus, the binary equivalent of $(36)_{10}$ is $(100100)_2$.

**Example 2:** Determine the binary equivalent of $(671)_{10}$.

| | | | |
|---|---|---|---|
| 2 | 671 | **Remainder** | |
| 2 | 335 | 1 | **Least Significant Bit (LSB)** ↑ |
| 2 | 167 | 1 | |
| 2 | 83 | 1 | |
| 2 | 41 | 1 | |
| 2 | 20 | 1 | |
| 2 | 10 | 0 | |
| 2 | 5 | 0 | |
| 2 | 2 | 1 | |
| 2 | 1 | 0 | |
| | 0 | 1 | **Most Significant Bit (MSB)** |

Taking remainders in reverse order, we have 1010011111. Thus, the binary equivalent of $(671)_{10}$ is $(1010011111)_2$.

In every number system, we will number each bit as follows:

- The first bit from the right in a binary number system is bit position zero.
- Each bit to the left is given as the next successive bit number.

Here, bit at position zero is usually referred to as the LSB (least significant bit). The first bit from the left is typically called the MSB (most significant bit). In the above examples 1 and 2, the LSB and the MSB are indicated. The intermediate bits are referred by their respective bit numbers.

## 2.3.3 Conversion of Binary to Decimal

In the binary to decimal conversion, each digit of the binary number is multiplied by its weighted position, and each of the weighted values is added together to get the decimal number. Consider the following examples:

**Example 1:** Determine the decimal equivalent of $(11010)_2$.

| Binary Number | 1 | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|
| Weight of Each Bit | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| Weighted Value | $2^4 \times 1$ | $2^3 \times 1$ | $2^2 \times 0$ | $2^1 \times 1$ | $2^0 \times 0$ |
| Solved Multiplication | 16 | 8 | 0 | 2 | 0 |

Sum of weight of all bits $= 16 + 8 + 0 + 2 + 0$
$= 26$

Thus, the decimal equivalent of $(11010)_2$ is $(26)_{10}$.

**Example 2:** Determine the decimal equivalent of $(10110011)_2$.

| Binary Number | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|
| Weight of Each Bit | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| Weighted Value | $2^7 \times 1$ | $2^6 \times 0$ | $2^5 \times 1$ | $2^4 \times 1$ | $2^3 \times 0$ | $2^2 \times 0$ | $2^1 \times 1$ | $2^0 \times 1$ |
| Solved Multiplication | 128 | 0 | 32 | 16 | 0 | 0 | 2 | 1 |

Sum of weight of all bits $= 128 + 0 + 32 + 16 + 0 + 0 + 2 + 1$
$= 179$

Thus, the decimal equivalent of $(10110011)_2$ is $(179)_{10}$.

## 2.3.11 Conversion of Octal to Binary

Since it is easier to read large numbers in octal form than in the binary form, the primary application of octal numbers is representing binary numbers. Besides, each octal digit can be represented by a three-bit binary number; it is very easy to convert from octal to binary. The following steps are involved:

1. Convert the decimal number to its 3-bit binary equivalent.
2. Combine the 3-bit sections by removing the spaces to get the binary number.

**Example 1:** Determine the binary equivalent of $(231)_8$.

| Octal Number | 2 | 3 | 1 |
|---|---|---|---|
| Binary Coded Value | 010 | 011 | 001 |

Combining the 3-bits of the binary coded values, we have 010011001.

Thus, the binary equivalent of $(231)_8$ is $(010011001)_2$.

**Example 2:** Determine the binary equivalent of $(453267)_8$.

| Octal Number | 4 | 5 | 3 | 2 | 6 | 7 |
|---|---|---|---|---|---|---|
| Binary Coded Value | 100 | 101 | 011 | 010 | 110 | 111 |

Combining the 3-bits of the binary coded values, we have 100101011010110111.

Thus, the binary equivalent of $(453267)_8$ is $(100101011010110111)_2$.

## 2.3.9 Conversion of Binary to Octal

The conversion of an integer binary number to octal is accomplished by the following steps:

1. Break the binary number into 3-bit sections starting from the LSB to the MSB.
2. Convert the 3-bit binary number to its octal equivalent.

For whole numbers, it may be necessary to add a zero as the MSB in order to complete a grouping of three bits.

*Note: By adding a zero, the MSB will not change the value of the binary number.*

**Example 1:** Determine the octal equivalent of $(010111)_2$.

| Binary Number | 010 (MSB) | 111 (LSB) |
|---|---|---|
| Octal Number | 2 | 7 |

The octal equivalent of $(010111)_2$ is $(27)_8$.

**Example 2:** Determine the octal equivalent of $(1010111110110010)_2$.

| Binary Number | 001 (MSB) | 010 | 111 | 110 | 110 | 010 (LSB) |
|---|---|---|---|---|---|---|
| Octal Number | 1 | 2 | 7 | 6 | 6 | 2 |

The octal equivalent of $(1010111110110010)_2$ is $(127662)_8$.

**Note:** *In the above example, we have added two 0s in the MSB so as to complete the required grouping of 3-bits.*