Base Conversions for Number System

Electronic and digital systems use various number systems such as Decimal, Binary, Hexadecimal and Octal, which are essential in computing.

Binary (base-2) is the foundation of digital systems. Hexadecimal (base-16) and Octal (base-8) are commonly used to simplify the representation of binary data. The Decimal system (base-10) is the standard system for everyday calculations. Other number systems like Duodecimal (base-12), are less commonly used but have specific applications in certain fields.

Types of Number System

There are four common types of number systems based on the radix or base of the number :

1. Decimal Number System The Decimal system is a base-10 number system. It uses ten digits: 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9. Each digit’s place value is a power of 10 (e.g., 100, 101, 102). It is the standard system for everyday counting and calculations.

2. Binary Number System The Binary system is a base-2 number system. It uses two digits: 0 and 1. Each digit’s place value is a power of 2 (e.g., 20, 21, 22). The Binary system is the foundation for data representation in computers and digital electronics.

3. Octal Number System The Octal system is a base-8 number system. It uses eight digits: 0, 1, 2, 3, 4, 5, 6 and 7. Each digit’s place value is a power of 8 (e.g., 80, 81, 82). It is often used to simplify the representation of binary numbers by grouping them into sets of three bits.

4. Hexadecimal Number System The Hexadecimal system is a base-16 number system. It uses sixteen digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F (where A = 10, B = 11, etc.). Each digit’s place value is a power of 16 (e.g., 160, 161, 162). Hexadecimal simplifies binary by representing every 4 bits as one digit (0-F).

Number System Conversion Methods

A number N in base or radix b can be written as: (N)b = dn-1 dn-2 -- -- -- -- d1 d0 . d-1 d-2 -- -- -- -- d-m In the above, dn-1 to d0 is the integer part, then follows a radix point and then d-1 to d-m is the fractional part. dn-1 = Most significant bit (MSB) d-m = Least significant bit (LSB) Base-in-Number-System Base in Number System.

1. Decimal to Binary Number System Conversion

For Integer Part: Divide the decimal number by 2. Record the remainder (0 or 1). Continue dividing the quotient by 2 until the quotient is 0. The binary equivalent is the remainders read from bottom to top.

For Fractional Part: Multiply the fractional part by 2. Record the integer part (0 or 1). Take the fractional part of the result and repeat the multiplication. Continue until the fractional part becomes 0 or reaches the desired precision. The binary equivalent is the integer parts recorded in sequence. Example: (10.25)10 Base_Conversion_Example Decimal to Binary Conversion For Integer Part (10): Divide 10 by 2 → Quotient = 5, Remainder = 0 Divide 5 by 2 → Quotient = 2, Remainder = 1 Divide 2 by 2 → Quotient = 1, Remainder = 0 Divide 1 by 2 → Quotient = 0, Remainder = 1 Reading the remainders from bottom to top gives 1010. For Fractional Part (0.25): Multiply 0.25 by 2 → Result = 0.5, Integer part = 0 Multiply 0.5 by 2 → Result = 1.0, Integer part = 1 The fractional part ends here as the result is now 0. Reading from top to bottom gives 01. Thus, the binary equivalent of (10.25)10 is (1010.01)2.

2. Binary to Decimal Number System Conversion

For Integer Part: Write down the binary number. Multiply each digit by 2 raised to the power of its position, starting from 0 (rightmost digit). Add up the results of these multiplications. The sum is the decimal equivalent of the binary integer.

For Fractional Part: Write down the binary fraction. Multiply each digit by 2 raised to the negative power of its position, starting from -1 (first digit after the decimal point). Add up the results of these multiplications. The sum is the decimal equivalent of the binary fraction.

Example: (1010.01)2 1x23 + 0x22 + 1x21+ 0x20 + 0x2 -1 + 1x2 -2 = 8+0+2+0+0+0.25 = 10.25 Thus, (1010.01)2 = (10.25)10

3. Decimal to Octal Number System Conversion

For Integer Part: Divide the decimal number by 8. Record the remainder (0 to 7). Continue dividing the quotient by 8 until the quotient is 0. The octal equivalent is the remainders read from bottom to top.

For Fractional Part: Multiply the fractional part by 8. Record the integer part (0 to 7). Take the fractional part of the result and repeat the multiplication. Continue until the fractional part becomes 0 or reaches the desired precision. The octal equivalent is the integer parts recorded in sequence.

Example: (10.25)10 For Integer Part (10): Divide 10 by 8 Quotient = 1, Remainder = 2 Divide 1 by 8 Quotient = 0, Remainder = 1 Octal equivalent = 12 (write the remainder, read from bottom to top). So, the octal equivalent of the integer part 10 is 12. For Fractional Part (0.25): Multiply 0.25 by 8 Result = 2.0, Integer part = 2 The fractional part ends here as the result is now 0. So, the octal equivalent of the fractional part 0.25 is 0.2. The octal equivalent of (10.25)10 = (12.2)8

4. Octal to Decimal Number System Conversion

For Integer Part: Write down the octal number. Multiply each digit by 8 raised to the power of its position, starting from 0 (rightmost digit). Add up the results of these multiplications. The sum is the decimal equivalent of the octal integer.

For Fractional Part: Write down the octal fraction. Multiply each digit by 8 raised to the negative power of its position, starting from -1 (first digit after the decimal point). Add up the results of these multiplications. The sum is the decimal equivalent of the octal fraction. Example: (12.2)8 1 x 81 + 2 x 80 +2 x 8-1 = 8+2+0.25 = 10.25 Thus, (12.2)8 = (10.25)10

5. Decimal to Hexadecimal Conversion

For Integer Part: Divide the decimal number by 16. Record the remainder (0-9 or A-F). Continue dividing the quotient by 16 until the quotient is 0. The hexadecimal equivalent is the remainders read from bottom to top.

For Fractional Part: Multiply the fractional part by 16. Record the integer part (0-9 or A-F). Take the fractional part of the result and repeat the multiplication. Continue until the fractional part becomes 0 or reaches the desired precision. The hexadecimal equivalent is the integer parts recorded in sequence. Example: (10.25)10 Integer part: 10 ÷ 16 = 0, Remainder = A (10 in decimal is A in hexadecimal) Hexadecimal equivalent = A Fractional part: 0.25 × 16 = 4, Integer part = 4 Hexadecimal equivalent = 0.4 Thus, (10.25)10 = (A.4)16

6. Hexadecimal to Decimal Conversion

For Integer Part: Write down the hexadecimal number. Multiply each digit by 16 raised to the power of its position, starting from 0 (rightmost digit). Add up the results of these multiplications. The sum is the decimal equivalent of the hexadecimal integer.

For Fractional Part: Write down the hexadecimal fraction. Multiply each digit by 16 raised to the negative power of its position, starting from -1 (first digit after the decimal point). Add up the results of these multiplications. The sum is the decimal equivalent of the hexadecimal fraction. Example: (A.4)16 (A × 160) + (4 × 16-1) = (10 × 1) + (4 × 0.0625) Thus, (A.4)16 = (10.25)10

7. Hexadecimal to Binary Number System

Each hexadecimal digit (0-9 and A-F) is represented by a 4-bit binary number. For each digit in the hexadecimal number, find its corresponding 4-bit binary equivalent and write them down sequentially. Example: (3A)16 (3)16 = (0011)2 (A)16 = (1010)2 Thus, (3A)16 = (00111010)2

8. Binary to Hexadecimal Number System

Start from the rightmost bit and divide the binary number into groups of 4 bits each. If the number of bits isn't a multiple of 4, pad the leftmost group with leading zeros. Each 4-bit binary group corresponds to a single hexadecimal digit. Replace each 4-bit binary group with the corresponding hexadecimal digit. Example: (1111011011)2 0011 1101 1011 | | | 3 D B Thus, (001111011011)2 = (3DB)16

9. Binary to Octal Number System

Starting from the rightmost bit, divide the binary number into groups of 3 bits. If the number of bits is not a multiple of 3, add leading zeros to the leftmost group. Each 3-bit binary group corresponds to a single octal digit. The binary-to-octal conversion for each 3-bit group is as follows: Octal Binary Equivalent 0 000 1 001 2 010 3 011 4 100 5 101 6 110 7 111 Replace each 3-bit binary group with the corresponding octal digit. Example: (111101101)2 111 101 101 | | | 7 5 5 Thus, (111101101)2 = (755)8

10. Octal to Binary Number System

Each octal digit (0-7) corresponds to a 3-bit binary number. For each octal digit, replace it with its corresponding 3-bit binary equivalent. Example: (153)8 Break the octal number into digits: 1, 5, 3 Convert each digit to binary: 1 in octal = 001 in binary 5 in octal = 101 in binary 3 in octal = 011 in binary Thus, (153)8 = (001101011)2