| Experiment No. 1 |
| --- |
| Implement a program to convert hexadecimal, decimal number to binary number. |
| Date of Performance: |
| Date of Correction: |

**Aim:** To implement a program that converts numbers from hexadecimal and decimal number systems into their equivalent binary representation.

**Objective:** To understand and implement the conversion of decimal and hexadecimal numbers into binary format, which is fundamental to data representation in digital systems.

**Theory:**

In computer systems, data is always processed and stored in binary—a base-2 numeral system that uses only two digits: 0 and 1. However, humans commonly use decimal (base-10), and system-level programs or hardware diagnostics frequently rely on hexadecimal (base-16) due to its compactness and readability.

**Number Systems Overview:**

| **Number System** | **Base** | **Digits Used** | **Common Usage** |
| --- | --- | --- | --- |
| **Binary** | 2 | 0, 1 | Used internally by all computers |
| **Decimal** | 10 | 0–9 | Used by humans for general calculations |
| **Hexadecimal** | 16 | 0–9, A–F | Used in memory addressing, debugging, etc. |

**Decimal to Binary Conversion:**

**Decimal to binary conversion involves repeated division by 2:**

* Divide the decimal number by 2.
* Store the remainder.
* Repeat the process on the quotient until the quotient is 0.
* The binary number is the remainders read in reverse.

**Example:  
Convert 13 to binary:**

13 ÷ 2 = 6 remainder 1

6 ÷ 2 = 3 remainder 0

3 ÷ 2 = 1 remainder 1

1 ÷ 2 = 0 remainder 1

**Binary = 1101**

**Hexadecimal to Binary Conversion:**

Hexadecimal to binary conversion is direct and efficient, as each hex digit maps exactly to a 4-bit binary number.

| **Hex Digit** | **Binary Equivalent** |
| --- | --- |
| 0 | 0000 |
| 1 | 0001 |
| ... | ... |
| A (10) | 1010 |
| F (15) | 1111 |

**Example:**Convert Hex 2F to Binary: **2 = 0010, F = 1111 → Binary = 00101111**

**Why Binary?**

* Binary aligns with the ON/OFF (high/low voltage) nature of digital electronics.
* It simplifies the design of hardware logic circuits using gates.
* All information (text, numbers, audio, video) in digital systems is represented in binary format.

**Real-World Relevance:**

* Microprocessors handle data and instructions in binary format.
* Hexadecimal simplifies representation of large binary numbers in system diagnostics and debugging.
* Understanding conversions is essential for memory management, instruction decoding, and low-level programming.

**Solution:**

**Conclusion:** We learned how to convert numbers from decimal and hexadecimal systems into binary. This helped us understand how computers represent and process different number systems internally.