**DLD PROJECT REPORT**



**Title: BINARY TO GRAY CODE CONVERTER**

**Course: Digital Logic Design**

**Member1: Muhammad Omer Siddiqui (24K-0022)**

**Member 2: Syed Anwar Kamal Rizvi (24K-0033)**

**Institution: Fast Karachi**

**Instructor: Ma’am Mehak Mazhar**

# 2. Abstract

This project aims to design and implement a 4-bit **Binary to Gray Code Converter** using basic digital logic components and ICs. The converter receives a 4-bit binary number from user input switches and converts it into its corresponding gray code, which is then displayed using LEDs. The design is based on XOR logic and implemented using a 7486 IC. Key components include push buttons, LEDs, resistors, and a 9V battery regulated to 5V using an LM7805. This project provides practical insight into digital encoding and combinational circuit design.

# 3. Introduction

Gray code is a binary numeral system where two successive values differ in only one bit. It is widely used in digital communication and mechanical encoders to prevent errors during bit transitions. This project was chosen to demonstrate a real-world application of digital logic gates using fundamental components. The converter helps visualize how gray code differs from binary and reinforces the application of XOR logic in practical circuits.

# 4. Objectives

* To understand the working of XOR gates in encoding.
* To design and implement a 4-bit Binary to Gray Code converter.
* To simulate and test the logic using software and hardware.
* To apply theoretical digital logic concepts to real-world circuits.

# 5. System Design / Methodology a. Truth Table

|  |  |
| --- | --- |
| **Binary Input (B3 B2 B1 B0)** | **Gray Output (G3 G2 G1 G0)** |
| 0000 | 0000 |
| 0001 | 0001 |
| 0010 | 0011 |
| **Binary Input (B3 B2 B1 B0)** | **Gray Output (G3 G2 G1 G0)** |
| 0011 | 0010 |
| 0100 | 0110 |
| 0101 | 0111 |
| 0110 | 0101 |
| 0111 | 0100 |
| 1000 | 1100 |
| 1001 | 1101 |
| 1010 | 1111 |
| 1011 | 1110 |
| 1100 | 1010 |
| 1101 | 1011 |
| 1110 | 1001 |
| 1111 | 1000 |

# b. Boolean Expressions

* G3 = B3
* G2 = B3  B2
* G1 = B2  B1
* G0 = B1  B0

# c. Logic Diagram

* Implemented using the **7486 XOR gate IC** (4 gates total)
* Push buttons for B0–B3 inputs
* Pull-down resistors for button stability
* Output LEDs for G0–G3 through 220Ω resistors
* Circuit powered via **9V battery regulated to 5V using LM7805**

# d. Flowchart

1. User enters binary input using push buttons
2. Logic circuitry computes gray code using XOR operations
3. Output displayed on LEDs

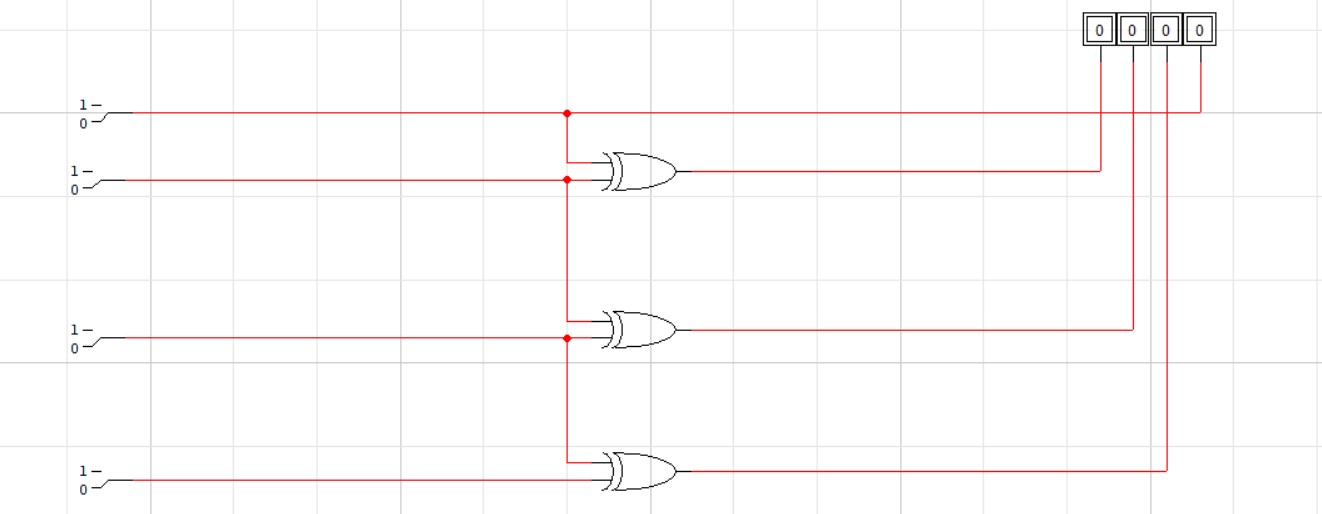
# e. Hardware Components

* 7486 IC
* 4 Push buttons (self-latching)
* 8 LEDs (4 input, 4 output)
* 9V battery with LM7805 regulator
* Diode 1N4148 (for voltage drop safety)
* 220Ω resistors for LEDs, 1KΩ pull-down resistors
* Breadboard and jumper wires

# 6. Simulation / Implementation a. Simulation

Simulated using Logic Works

* Logic was tested by applying binary inputs and confirming gray outputs.
* XOR gate behavior was validated according to the Boolean expressions.



# b. Hardware Testing

* Connected the circuit on a breadboard using a regulated 5V supply.
* Verified correct output on LEDs for each binary input combination.
* Ensured stable operation using pull-down resistors to avoid floating inputs. **c. Issues Faced**
* **Voltage mismatch** initially (solved using LM7805 voltage regulator)
* **Button bouncing** — reduced effect by choosing latching buttons
* **Wiring clutter** — minimized by planning layout before breadboarding

# 7. Results Analysis

The circuit successfully converted 4-bit binary inputs into the corresponding gray code values. The LED outputs matched simulation results across all input combinations.

# Binary Input Gray Output (Expected) Gray Output (Actual)

|  |  |  |
| --- | --- | --- |
| 0101 | 0111 | 0111 |
| 1111 | 1000 | 1000 |
| 0010 | 0011 | 0011 |

**8. Limitations and Future Work Limitations:**

* Manual push-button input is not scalable.
* LEDs limit the output to visual verification only.
* Noise can affect the mechanical buttons.

**Future Improvements:**

* Replace buttons with DIP switches for cleaner input.
* Add a 7-segment display for gray code output.
* Upgrade to a microcontroller for automatic conversion.