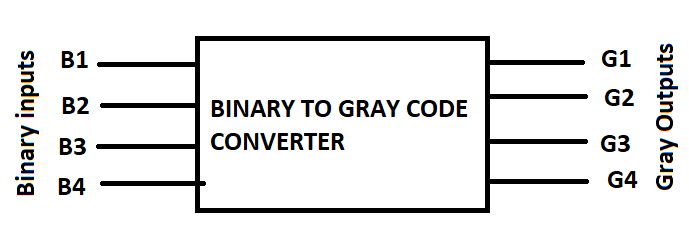
**AIM:** A code represents each number in the sequence of integers {0...2^N-1} as a binary string of length N in an order such that adjacent integers have code representations that differ in only one bit position. Design a convertor that has above property.

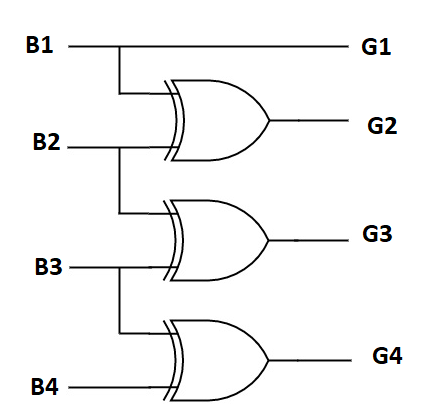
**APPARATUS REQUIRED:** Logic Trainer kit, connecting wires, bread board, IC 7486.

**1- Bit Magnitude Comparator**



**Fig:-5.1 4-bit Binary to Gray code converter**

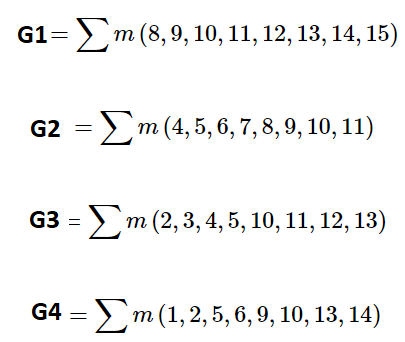
1. **Logical Diagram:**



**Figure: - 5.2**

1. **Truth Table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | | | | **Output** | | | |
| **B1** | **B2** | **B3** | **B4** | **G1** | **G2** | **G3** | **G4** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

****

1. **Logical Expression:**

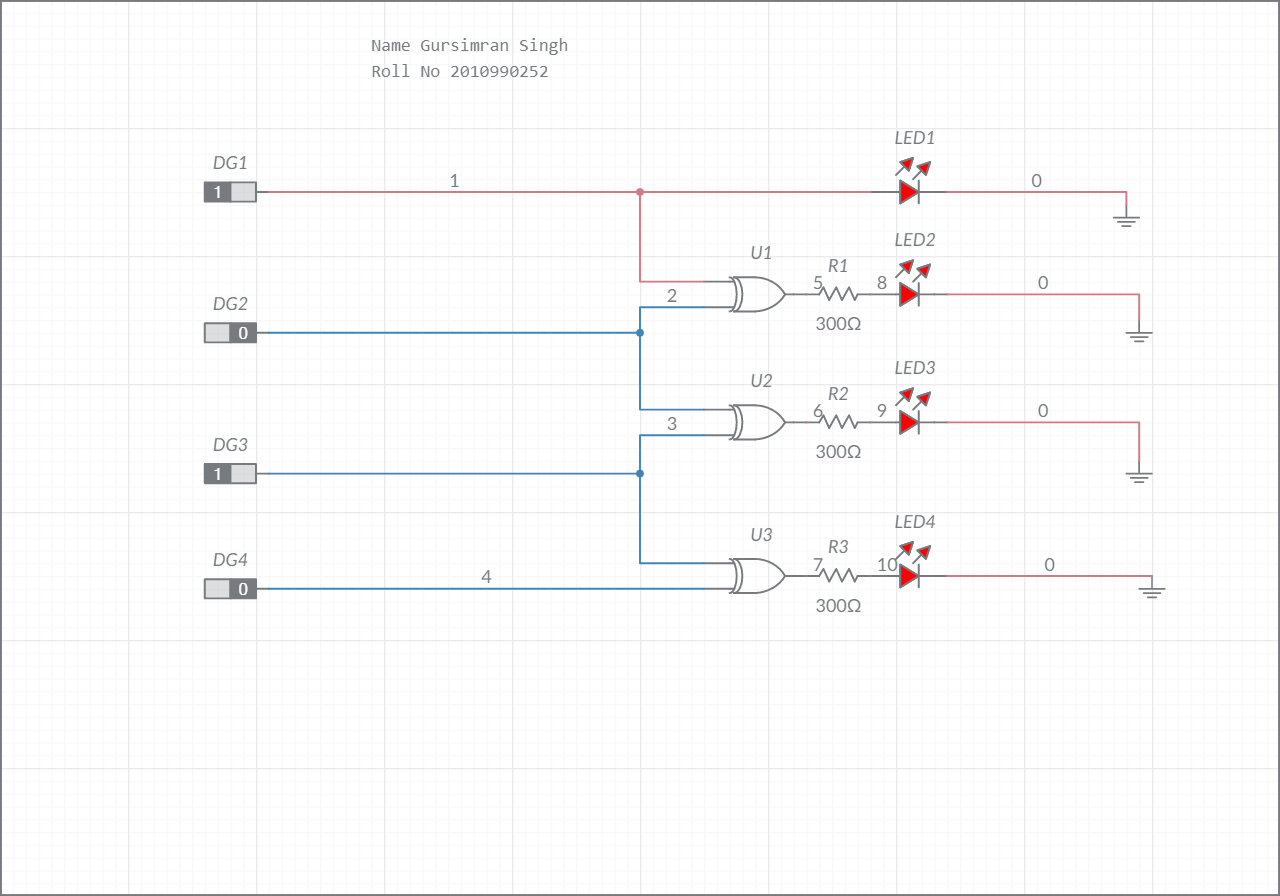
**G1 = B1**

**G2 = B1’B2 + B1B2’ = B1 (+) B2**

**G3 = B2’B3 + B2B3’ = B2 (+) B3**

**G4 = B3’B4+B3B4’ = B3 (+) B4**

**1 4-bit Binary to Gray code converter Logical Diagram (Made On Multisim)**



**CONCLUSION:**

Hence, we have successfully designed the circuit of binary to Gray Code converter circuit from the truth table and logical expression on Multisim.