**LAB #13**

**Karnaugh Map or K-MAP Minimization**

**OBJECTIVE:**

* 1. The aim of this lab is to minimize the digital circuit using K-Map
  2. To implement the reduced circuit on hardware / software.

**EQUIPMENT:**

20. IC: 7408LS, 7432LS, 7404LS, 7400LS and 7402LS.

1. Bread board.
2. Connection Wires.
3. Digital Logic Probe.
4. DC supply (0 and +5V).

**THEORY:**

The complexity of the digital logic gates that implement a Boolean function is directly related to the complexity of the algebraic expression from which the function is implemented. Although the truth table representation of a function is unique, when it is expressed algebraically it can appear in many different, but equivalent, forms. Boolean expressions may be simplified by Boolean rules. However, this procedure of minimization is awkward because it lacks specific rules to predict each succeeding step in the manipulative process. The map method provides a simple, straightforward procedure for minimizing Boolean functions. This method may be regarded as a pictorial form of a truth table. The map method is also known as the Karnaugh map or K-map.

**Logic Implementation: Example # 01:**

Implement the following function

𝐹1 = 𝐴𝐵𝐶 + 𝐴𝐵𝐶̅ + 𝐴𝐵̅𝐶

On the breadboard and fill in the truth table F1 by applying all possible inputs.

TABLE 4.1: Observation Table – Example 01

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input: A** | **Input: B** | **Input: C** | **F1** | **F2** |
| 0 | 0 | 0 |  |  |
| 0 | 0 | 1 |  |  |
| 0 | 1 | 0 |  |  |
| 0 | 1 | 1 |  |  |
| 1 | 0 | 0 |  |  |
| 1 | 0 | 1 |  |  |
| 1 | 1 | 0 |  |  |
| 1 | 1 | 1 |  |  |

###### Simplify the above Boolean equation (F1) using K-Map – Find Simplified Expression F2?

* 1. Simplify the above Boolean equation (F1) using the multisim instrument “LOGIC CONVERTER”.

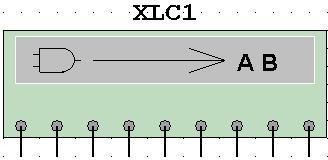


Figure 4.1: MultiSim Logic Simplification Tool – Logic Converter

###### Implement the simplified expression on the bread board and fill the in above truth table F2?

**Example # 02:**

Consider the Boolean Function as,

*F (x, y, z) = ∑ (1, 2, 4, 5, 6)*

On simplification using K-Map, the simplified equation is,

𝐹 = 𝐴𝐵̅ + 𝐵̅𝐶 + 𝐵𝐶̅

Verify both equations by Truth Table and by designing Circuit on Hardware.

**Example # 03:**

Consider the Boolean Function as,

𝑌 = 𝐹𝐹𝐹𝐸 (Hex)

On simplification using K –Map, the simplified equation is,

𝑌 = 𝐴 + 𝐵 + 𝐶 + 𝐷

Verify both equations by Truth Table and by designing Circuit on Hardware.

**Example # 04:**

Consider the Boolean Function as,

𝐹 (𝑥, 𝑦, 𝑧) = ∑ (0,1,4,5)

On simplification using K –Map, the simplified equation is,

𝐹 = 𝐵̅

Verify both equations by Truth Table and by designing Circuit on Hardware

**CONCLUSION:**

* + 1. It is easy to simplify the circuit using K-Map.
    2. After Simplification Hardware implantation is quite easy.
    3. Reduces the complexity.

**QUESTIONS:**

1. Simplify the following function using K-Map in Sum of Product. (SOP)

Product of sum. (POS)

𝐹 (𝐴, 𝐵, 𝐶, 𝐷) = ∑(0, 1, 2, 5, 8, 9 ,10)

Verify Boolean function ‘F’ before and after K-Map simplification using truth table and design circuit on hardware / software.

1. Simplify the Boolean Function using K-Map

𝐹 (𝐴, 𝐵, 𝐶, 𝐷) = ∑(1, 3, 7, 11, 15)

Which has the don’t care conditions

𝐷 (𝐴, 𝐵, 𝐶, 𝐷) = ∑(0, 2, 5)

Verify Boolean function ‘F’ before and after K-Map simplification using truth table and

design circuit on hardware / software.

1. Simplify the Boolean Function using K-Map

𝐹 = 𝐸8 (Hex)

Verify Boolean function ‘F’ before and after K-Map simplification using truth table and design circuit on hardware / software.