

Simplification of Logic Expression Using K-Map

Objectives:

- ➤ To simplify logic expressions using Karnaugh Maps (K-Maps).
- ➤ To understand the process of grouping and reducing terms in a logic expression.
- ➤ To compare the simplified expressions with the original expressions.

Apparatus:

- i. Logic gates (AND, OR, NOT),
- ii. Breadboard,
- iii. Connecting wires,
- iv. LEDs and resistors

Components:

- i. Integrated circuits (ICs),
- ii. Push buttons,
- iii. Power supply

Theory:

Karnaugh Maps (K-Maps) are graphical tools used for simplifying Boolean expressions. They provide a visual representation of the truth table of a logic function. K-Maps are especially useful for simplifying expressions with a small number of variables. The process involves grouping adjacent cells containing 1s in the K-Map to eliminate redundant terms in the Boolean expression.

Procedure:

1. Design a truth table for the given logic expression.
2. Determine the number of variables and create a Karnaugh Map accordingly.
3. Fill in the K-Map cells with the corresponding values from the truth table.
4. Identify groups of adjacent 1s (2^n cells, where n is the number of variables) in the K-Map.
5. Write down the simplified expressions for each group, which represents a term in the simplified Boolean expression.
6. Combine the simplified terms to obtain the final simplified Boolean expression.
7. Implement the original and simplified expressions using logic gates and verify their outputs using LEDs.

Conclusion:

The process of simplifying logic expressions using Karnaugh Maps is successfully demonstrated. By analyzing the original logic expression and obtained a simplified expression by grouping and reducing terms based on the K-Map. Both the original and simplified expressions using logic gates and verified their functionality by observing the LED outputs are implemented. This experiment highlighted the importance of K-Maps in simplifying complex logic expressions, leading to more efficient and manageable circuit designs.