Ministry of Higher Education and Scientific Research

University of Anbar

College of Computer Science and Information Technology

Department of Computer Science

Logic Design – Lecture 4

Karnaugh Maps (K-Maps) and Logic Simplification

Instructor: Mohammed Shehab Hamad

1. Introduction to K-Maps

• Karnaugh Maps (K-Maps) are a method of simplifying Boolean algebra expressions. They provide a visual means of grouping terms to eliminate variables and reduce expressions to their simplest form.

• K-Maps were introduced by Maurice Karnaugh in 1953 to improve the simplification process of Boolean functions.

• Each K-Map corresponds to a truth table, and it provides a graphical method of representing logic functions.

• K-Maps are more efficient than algebraic simplification for functions with up to 6 variables.

2. K-Map Structures and Notation

• A 2-variable K-Map contains 4 cells, each representing one minterm.

• A 3-variable K-Map contains 8 cells. A 4-variable K-Map contains 16 cells.

• Each cell corresponds to a unique combination of variable states (e.g., A'B, AB', etc.).

• Gray code is used in K-Maps to ensure only one bit changes between adjacent cells.

3. Simplification Rules

• Groups must contain 1, 2, 4, 8, etc. (powers of 2) cells.

• Groups must be rectangular and can wrap around the edges.

• Larger groups result in simpler expressions.

• Overlapping groups are allowed and often necessary for optimal simplification.

• Each 1 in the K-Map must be included in at least one group.

4. Examples of Simplification

• Example 1: Simplify F(A, B) = Σ (1, 3)

• K-Map: [0, 1; 0, 1] → Group (1, 3) → F = B

• Example 2: F(A, B, C) = Σ (1, 2, 3, 5, 7)

• Group adjacent 1s: (1, 3), (5, 7), (3, 7) → F = B'C + AC

• Example 3: 4-variable function with Σ (0,1,2,5,6,7,8,9,10,14)

• Multiple possible groupings → Choose largest rectangles to reduce the function

• Example 4: Simplify F(A,B,C,D) = Σ (0,1,2,5,6,7,8,9)

• Step 1: Construct the 4-variable K-Map with 16 cells.

• Step 2: Fill the cells with 1s at specified minterms.

• Step 3: Group adjacent 1s into largest possible rectangles.

• Step 4: Derive simplified expression.

• Result: F = B'D' + AC

• K-Map Table (Text Representation):

• CD → ¦ 00 ¦ 01 ¦ 11 ¦ 10 ¦

• AB ↓ ----------

• 00 ¦ 1 ¦ 1 ¦ 0 ¦ 1 ¦

• 01 ¦ 1 ¦ 1 ¦ 0 ¦ 0 ¦

• 11 ¦ 1 ¦ 1 ¦ 0 ¦ 0 ¦

• 10 ¦ 0 ¦ 0 ¦ 0 ¦ 0 ¦

5. Don ’ t Care Conditions

• Don't care conditions are outputs where the function can be either 0 or 1 without affecting functionality.

• They are marked with 'X' in K-Maps.

• Used to create larger groups when simplifying expressions.

• Example: F(A,B,C) = Σ(1,3,7), d(A,B,C) = Σ(2,6) → include 2 and 6 in groups if beneficial

6. POS (Product of Sums) Simplification with K-Maps

• K-Maps can also be used to simplify expressions in Product of Sums (POS) form.

• Instead of grouping 1s, we group 0s and derive the maxterms.

• Example: F = ΠM(0,2,5) → K-Map shows zeros in these positions → derive sum terms from ungrouped areas

7. Common Mistakes

• Not using wrap-around groups.

• Making groups that are not powers of 2.

• Overlooking don ’ t care terms that could simplify the expression further.

• Incorrectly assigning variable combinations to K-Map cells.

8. Practice Exercises

• Simplify: $F(A,B,C,D) = \Sigma(0,2,5,6,8,10,13,15)$

• Simplify: $F(W,X,Y,Z) = \Sigma(1,3,7,11,15)$ with $d(W,X,Y,Z) = \Sigma(0,2,5)$

• Simplify: $F = \Pi M(0,1,3,7,8,9)$

• Design a logic circuit from the simplified expression of $F = \Sigma(1,2,3,5,7)$

• Draw a 4-variable K-Map and show all possible groupings for $F = \Sigma(0,1,2,5,6,7,8,9)$

• Design a combinational logic circuit for a vending machine that accepts A and B inputs (coins) and outputs 1 if the total equals 10 cents.

• Draw the K-Map for this system and simplify the logic.

• Draw the truth table and K-Map for a 3-input majority function (outputs 1 if at least 2 inputs are 1).

• Construct the 4-variable K-Map for a logic function where the output is 1 only when exactly two inputs are true.

• Challenge: Find the minimal expression using K-Map for $F(W,X,Y,Z) = \Sigma(0,1,2,6,8,10,12,14)$ with don ' t cares at $\Sigma(3,11)$.

9. Homework Assignment

• 1. Use K-Maps to simplify the following Boolean expressions:

• a. $F(A,B,C) = \Sigma(0,1,3,5,7)$

• b. $F(W,X,Y,Z) = \Sigma(1,2,5,7,9,11,13)$

• 2. Identify the optimal groupings in a 4-variable K-Map for $F = \Sigma(0,2,3,8,10,11,14,15)$

• 3. Convert the following truth table into a simplified Boolean expression using a K-Map.

• 4. Draw the logic diagram for the simplified result of $F(A,B,C,D) = \Sigma(1,3,7,11,15)$

10. Summary

• K-Maps are a visual tool for simplifying logic expressions.

• Grouping 1s (or 0s for POS) in power-of-2 rectangles minimizes variables in the result.

• Don ’ t care conditions can help further reduce logic expressions.

• K-Maps reduce errors and simplify circuit design for up to 4–6 variables effectively.

12. In-Class Discussion Questions

- 1. Why are overlapping groups allowed in K-Map simplification?
- 2. How does Gray code ordering help in the grouping process?
- 3. What would happen if we group non-adjacent 1s? Why is that invalid?
- 4. How can don't care conditions change the final simplified expression?
- 5. Why is grouping more than the minimum number of 1s often beneficial?

11. References

- M. Morris Mano, Digital Design, Pearson Education, 5th Edition.
- Ronald J. Tocci, Digital Systems: Principles and Applications, 11th Edition.
- Floyd, Digital Fundamentals, Pearson, 10th Edition.