Karnaugh Maps

Hui Chen [a]

[a]CUNY Brooklyn College, Brooklyn, NY, USA

October 3, 2023

Outline

Acknowledgement

The content of most slides come from the authors of the textbook:

Null, Linda, & Lobur, Julia (2018). The essentials of computer organization and architecture (5th ed.). Jones & Bartlett Learning.

Table of Contents

Lesson Objectives

Students are expected to be able to

1. Apply Boolean algebra and functions.
2. *Understand the relationship between Boolean logic and digital computer circuits.*
3. Learn how to design simple logic circuits.
4. Understand how digital circuits work together to form complex computer systems.

Table of Contents

Introducing Karnaugh Maps

A graphical way of visualizing and then simplifying Boolean expressions.

This graphical representation is known as a Karnaugh map or Kmap.

- A Kmap is a matrix consisting of rows and columns that represent the output values of a Boolean function.
- The output values placed in each cell are derived from the *minterms* of a Boolean function,
- where a minterm is a product term that contains all the function's variables exactly once, either complemented or not complemented.

Minterm: Example

Given inputs x and y, the $2^2 = 4$ minterms are xy, xy', $x'y$, and $x'y'$.

Given inputs x, y, and z, the $2^3 = 8$ minterms are xyz, xyz', $xy'z$, $xy'z'$, $x'yz$, $x'yz'$, $x'y'z$, and $x'y'z'$.

From MinTerm to Kmap

A Kmap has a cell for each minterm.

This means that it has a cell for each line for the truth table of a function.

From MinTerm to Kmap: Example 1

The truth table for the function $F(x,y) = xy$ is shown along with its corresponding Kmap.

x	y	xy
0	0	0
0	1	0
1	0	0
1	1	1

$x \backslash y$	0	1
0	0	0
1	0	1

From MinTerm to Kmap: Example 2

The truth table for the function $F(x, y) = x + y$ is shown along with its corresponding Kmap.

x	y	xy
0	0	0
0	1	0
1	0	0
1	1	1

$x \backslash y$	0	1
0	0	1
1	1	1

which shows that

$$F(x, y) = x + y = x'y + xy' + xy$$

From MinTerm to Kmap: Example 2: Not the Simplest!

The truth table for the function $F(x, y) = x + y$ is shown along with its corresponding Kmap.

$x \backslash y$	0	1
0	0	1
1	1	1

From the Kmap we obtain

$$F(x, y) = x'y + xy' + xy$$

The minterm function that we derived from our Kmap was not in the simplest terms.

From MinTerm to Kmap: Example 2: Simplify!

The truth table for the function $F(x,y) = x + y$ is shown along with its corresponding Kmap.

$x \backslash y$	0	1
0	0	1
1	1	1

From the Kmap we obtain

$$F(x,y) = x'y + xy' + xy$$

The minterm function that we derived from our Kmap was not in the simplest terms.

Find adjacent 1s in the Kmap that can be collected into groups that are powers of two in order to reduce our complicated expression to its simplest terms.

In this example, we have two such groups.

Kmap Simplification for Two Variables

The rules of Kmap simplification are:

- Groupings can contain only 1s; no 0s.
- Groups can be formed only at right angles; diagonal groups are not allowed.
- The number of 1s in a group must be a power of 2 – even if it contains a single 1.
- The groups must be made as large as possible.
- Groups can overlap and wrap around the sides of the Kmap.

Kmap for Three Variables

- A Kmap for three variables is constructed as shown in the diagram below.
- We have placed each minterm in the cell that will hold its value.
- Notice that the values for the yz combination at the top of the matrix form a pattern that is not a normal binary sequence.

x \ yz	00	01	11	10
0	$x'y'z'$	$x'y'z$	$x'yz$	$x'yz'$
1	$xy'z'$	$xy'z$	xyz	xyz'

- Thus, the first row of the Kmap contains all minterms where x has a value of zero.
- The first column contains all minterms where y and z both have a value of zero.

Simplification: Example 1

Consider boolean function $F(x, y, z) = x'y'z + x'yz + xy'z + xyz$

1. Construct Kmap

x \ yz	00	01	11	10
0	0	1	1	0
1	0	1	1	0

2. What is the largest group of 1s that is a power of 2?
3. The grouping tells us that changes in the variables x and y have no influence upon the value of the function – They are irrelevant.
4. This means that the function, $F(x, y, z) = z$

Simplification: Example 2

Consider (a more complex) boolean function
$F(x, y, z) = x'y'z' + x'y'z + x'yz + x'yz' + xy'z' + xyz'$

1. Construct Kmap

x \ yz	00	01	11	10
0	1	1	1	1
1	1	0	0	1

2. What are the largest groups of 1s that is a power of 2?
3. This group tells us that
   - y and z are not relevant to group 1 (what is it?) when $x = 0$
   - x and y are not relevant to group 2 (what is it?) when $z = 0$

   the values of x and y are not relevant to the term of the function that is encompassed by the group.
4. This means that the function, $F(x, y, z) = x' + z'$

Kmap Simplification for Four Variables

The model can be extended to accommodate the 16 minterms that are produced by a four-input function.

The format for a 16-minterm Kmap is:

$wx \backslash yz$	00	01	11	10
00	$w'x'y'z'$	$w'x'y'z$	$w'x'yz$	$w'x'yz'$
01	$w'xy'z'$	$w'xy'z$	$w'xyz$	$w'xyz'$
11	$wxy'z'$	$wxy'z$	xyz	$wxyz'$
10	$wx'y'z'$	$wx'y'z$	$x'yz$	$wx'yz'$

Simplification: Example 1

Consider boolean function $F(w,x,y,z) =$
$w'x'y'z' + w'x'y'z + w'x'yz' + w'xyz' + wx'y'z' + wx'y'z + wx'yz'$

1. Construct Kmap

wx \ yz	00	01	11	10
00	1	1	0	1
01	0	0	0	1
11	0	0	0	0
10	1	1	0	1

2. What are the largest groups of 1s that is a power of 2? (There are 3 groups! Hint: view the table as a cube)

Simplification: Example 1

Consider boolean function $F(w, x, y, z) =$
$w'x'y'z' + w'x'y'z + w'x'yz' + w'xyz' + wx'y'z' + wx'y'z + wx'yz'$

1. Construct Kmap

$wx \backslash yz$	00	01	11	10
00	1	1	0	1
01	0	0	0	1
11	0	0	0	0
10	1	1	0	1

2. Three groups:
   - A group entirely within the Kmap at the right (simplify to $w'yz'$).
   - A group that wraps the top and bottom (simplify to $x'y'$)
   - A group that spans the corners (simplify to $x'z'$).

Thus, $F(w, x, y, z) = w'yz' + x'y' + x'z'$

Alternative Groupings

It is possible to have a choice as to how to pick groups within a Kmap, while keeping the groups as large as possible.

What are possible groupings in the following kmap?

wx \ yz	00	01	11	10
00	1	0	1	0
01	1	0	1	1
11	1	0	0	0
10	1	0	0	0

which are logically equivalent.

Don't-Care Conditions

Real circuits don't always need to have an output defined for every possible input.

For example, some calculator displays 0 - 9 using a of 7-segment LEDs. These LEDs can display $2^7 - 1$ patterns, but only ten of them are useful.

- Example: https://pijaeducation.com/arduino/seven-segment/display-0-to-9-on-seven-segment-display/

If a circuit is designed so that a particular set of inputs can never happen, we call this set of inputs a don't care condition.

They are very helpful to us in Kmap circuit simplification.

Marking Don't-Care Conditions

In a Kmap, a don't-care condition is identified by an X in the cell of the minterm(s) for the don't-care inputs, as shown here.

$wx \backslash yz$	00	01	11	10
00	X	1	1	X
01	0	X	1	0
11	X	0	1	0
10	0	0	1	0

In performing the simplification, we are free to include or ignore the X's when creating our groups.

Simplification with Don't-Care Conditions: Example 1

$wx \backslash yz$	00	01	11	10
00	X	1	1	X
01	0	X	1	0
11	X	0	1	0
10	0	0	1	0

Groupings:

- 1st row
- 3rd column

which gives us $F(w, x, y, z) = w'x' + yz$

Simplification with Don't-Care Conditions: Example 2

$wx \backslash yz$	00	01	11	10
00	X	1	1	X
01	0	X	1	0
11	X	0	1	0
10	0	0	1	0

Alternative groupings:

- middle two columns in top two rows
- 3rd column

which gives us $F(w, x, y, z) = w'z + yz$

Truth Tables Differ!

The truth tables of the two resulting boolean functions $F(w, x, y, z) = w'x' + yz$ and $F(w, x, y, z) = w'z + yz$!

However, the values for which they differ, are the inputs for which we have don't care conditions.

More Variables?

- Kmaps provide an easy graphical method of simplifying Boolean expressions.
- A Kmap is a matrix consisting of the outputs of the minterms of a Boolean function.
- we only discussed 2-, 3-, and 4-input Kmaps. *This method can be extended to any number of inputs through the use of multiple tables.*

Recapping the rules of Kmap simplification

- Groupings can contain only 1s; no 0s.
- Groups can be formed only at right angles; diagonal groups are not allowed.
- The number of 1s in a group must be a power of 2 – even if it contains a single 1.
- The groups must be made as large as possible.
- Groups can overlap and wrap around the sides of the Kmap.
- Deals with don't-care conditions when you should.

Table of Contents

Summary and Q&A

You are expected to be able to

1. *using Karnaugh maps to simplify boolean functions up to 4 variables*;

Any questions on:

- Karnaugh Maps
- Minterm
- Two Variable Simplifications
- Three-Variable Simplifications
- Four-Variable Simplifications
- Kmaps with Don't-Care Conditions