Karnaugh Maps

CMSC 313

Raphael Elspas

Karnaugh Map

- A **Karnaugh Map** also known as a **K-map**, is a graphical method used for simplifying Boolean algebra expressions.
- It provides a systematic way to simplify logical expressions by grouping together adjacent cells in a table representing all possible combinations of input variables.

2-variable Karnaugh Maps

- Layout is a 2x2 grid, where the columns correspond with the values of one variable, and the rows correspond with the values of another variable.
- The values in the grid equal the output of the expression made up of the variable inputs

A \ B	0	1
0	A=0, B=0	A=0, B=1
1	A=1, B=0	A=1, B=1

Using 2-variable Karnaugh Map

- Example: given truth table, draw Karnaugh map and find equation

A	B	F
0	0	0
0	1	1
1	0	0
1	1	1

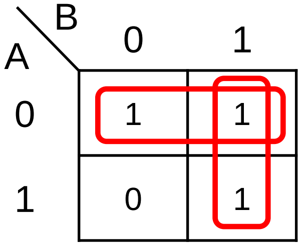
A \ B	0	1
0	0	1
1	0	1

$$F = B$$

Using 2-variable Karnaugh Map

- Example: given truth table, draw Karnaugh map and find equation

A	B	F
0	0	1
0	1	1
1	0	0
1	1	1



$$F = \bar{A} + B$$

Note: groups can overlap.
Choose the largest grouping you can in each direction. Must be a rectangle

3-variable Karnaugh Maps

- Layout is a 2x4 grid, where the columns correspond with the values of 2 variables, and the rows correspond with the values of the third variable.
- The values in the grid equal the output of the expression made up of the variable inputs

Note the order of the bits: nonstandard order

A \ BC	00	01	11	10
0	A=0, B=0, C=0	A=0, B=0, C=1	A=0, B=1, C=1	A=0, B=1, C=0
1	A=1, B=0, C=0	A=1, B=0, C=1	A=1, B=1, C=1	A=1, B=1, C=0

Using 2-variable Karnaugh Map

- Example: given truth table, draw Karnaugh map and find equation

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Note the order of the bits: gray code order

A \ BC	00	01	11	10
0	0	1	0	0
1	1	1	1	1

$$F = \bar{B}C + A$$

Note: groups can overlap. Choose the largest grouping you can in each direction, as long as it is a rectangle.

Gray Codes

- A **Gray code** is a sequencing of the binary numeral system in which two successive values differ in an only binary digits.
- Karnaugh maps use gray code ordering to allow **wraparound** grouping.

1-bit	2-bit	3-bit
0	00	000
1	01	001
	11	011
	10	010
		110
		111
		101
		100

Grouping Rules

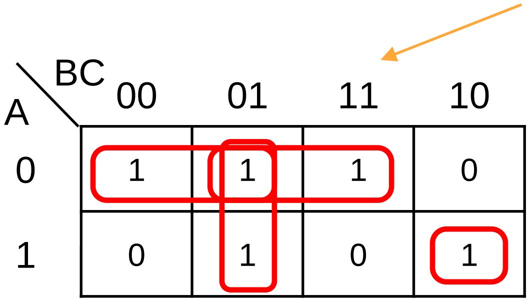
- Must cover all of the 1s with as few groups as possible
- Groups must be a rectangle, no L shapes or diagonals.
- Groups must have dimensions that are powers of 2. 1 is a power of 2!
- Valid groups examples: 1x4, 8x2
- Invalid groups examples: 2x6, 1x3
- Groups can wrap around the edge of the K-Map since a gray code is used.
- Use the largest groups possible

Using 3-variable Karnaugh Map

- Example: given truth table, draw Karnaugh map and find minimal equation

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Note: gray code order



$$F = \bar{A}\bar{B} + \bar{A}C + \bar{B}C + AB\bar{C}$$

Note: groups can overlap. Choose the largest grouping you can in each direction, as long as it is a rectangle.

UMBC

Using 3-variable Karnaugh Map

- Example: given truth table, draw Karnaugh map and find minimal equation

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

Note: gray code order

A \ BC	00	01	11	10
0	1	1	0	1
1	0	1	0	0

$$F = \bar{A}\bar{C} + \bar{B}C$$

Note: groups can wrap around

Using 3-variable Karnaugh Map

- Example: given SOP, draw Karnaugh map and find equation

$$F = \sum(0,1,2,4,5,6)$$

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

A \ BC	00	01	11	10
0	1	1	0	1
1	1	1	0	1

Note: groups can wrap around

$$F = \bar{B} + \bar{C}$$

Using 3-variable Karnaugh Map

- Example: given truth table, draw Karnaugh map and find equation

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

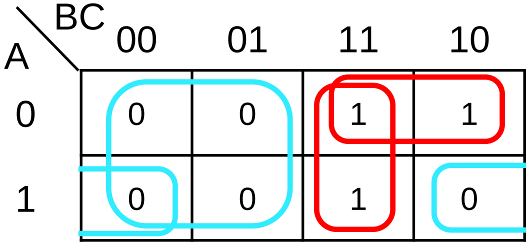
A \ BC	00	01	11	10
0	1	1	1	1
1	1	1	1	1

$$F = 1$$

K-map complement

- To find the complement of a Boolean expression, we can circle 0s

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



$F = BC + \bar{A}B$ (SOP of 1s)

$\bar{F} = \bar{B} + A\bar{C}$ (SOP of 0s)

4-variable Karnaugh Maps

- Layout is a 4x4 grid, where the columns correspond with the values of 2 variables, and the rows correspond with the values of the other 2 variables.
- The values in the grid equal the output of the expression made up of the variable inputs

AB \ CD	00	01	11	10
00	A=0,B=0, C=0,D=0	A=0,B=0, C=0,D=1	A=0,B=0, C=1,D=1	A=0,B=0, C=1,D=0
01	A=0,B=1, C=0,D=0	A=0,B=1, C=0,D=1	A=0,B=1, C=1,D=1	A=0,B=1, C=1,D=0
11	A=1,B=1, C=0,D=0	A=1,B=1, C=0,D=1	A=1,B=1, C=1,D=1	A=1,B=1, C=1,D=0
10	A=1,B=0, C=0,D=0	A=1,B=0, C=0,D=1	A=1,B=0, C=1,D=1	A=1,B=0, C=1,D=0

Note: gray code

4-variable Karnaugh Maps minterms

- The numbered minterms correspond with the spots on the 4-variable Karnaugh map as follows:

AB \ CD	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

Note: gray order for both rows and columns

- Maxterms occupy the spots in the same locations for each given input.

Using 4-variable Karnaugh Map

- Example: given SOP, draw Karnaugh map and find minimized equation

A	B	C	D	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

$$F = \sum(0,1,2,3,4,6,11,12,14)$$

AB \ CD	00	01	11	10
00	1	1	1	1
01	1	0	0	1
11	1	0	0	1
10	0	0	1	0

$$F = \bar{A}\bar{B} + B\bar{D} + \bar{B}CD$$

Note: groups can wrap around

Using 4-variable Karnaugh Map

- Example: given SOP, draw Karnaugh map and find minimized equation

A	B	C	D	F
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

$$F = \sum(0,2,8,10)$$

AB \ CD	00	01	11	10
00	1	0	0	1
01	0	0	0	0
11	0	0	0	0
10	1	0	0	1

$$F = \bar{B}\bar{D}$$

Note: groups can wrap around corners

Using 4-variable Karnaugh Map

- Example: given POS, draw Karnaugh map and find minimized equation

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

$$F = \prod(0,1,4,5,9)$$

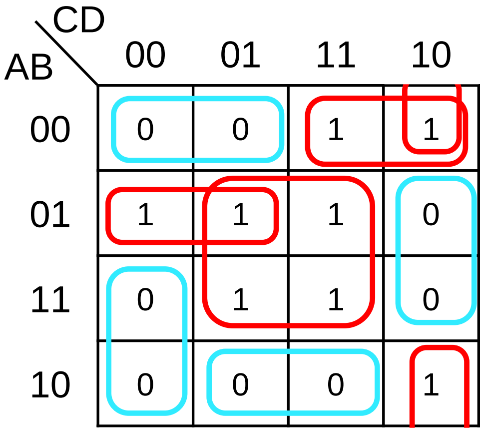
AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	1	1	1	1
10	1	0	1	1

$$F = C + AB + A\overline{D}$$

Karnaugh Map Complement

- Example: given SOP, draw Karnaugh map and find the minimized F and its minimized complement

$$F = \sum(2,3,4,5,7,10,13,15)$$



$$F = \bar{A}B\bar{C} + BD + \bar{A}\bar{B}C + \bar{B}C\bar{D}$$
$$\bar{F} = \bar{A}\bar{B}\bar{C} + A\bar{C}\bar{D} + A\bar{B}D + BC\bar{D}$$

Don’t Cares

- A **don’t care** term is an input for which the function output doesn’t matter.
- We treat a don’t care as either a 1 or a 0 depending on which is more advantageous.
- Represented as **X**. Some use “DC”
- **d(**2,3,7**)** means the 2nd 3rd and 7th rows in the truth table are “don’t cares”.

3-variable Don’t care Karnaugh Map

- Example: given truth table, draw Karnaugh map and find equation

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	x
1	0	1	0
1	1	0	x
1	1	1	1

A \ BC	00	01	11	10
0	1	1	0	0
1	x	0	1	x

$$F = \bar{A}\bar{B} + AB$$

4-variable don't care Karnaugh Map

- Example: Draw Karnaugh map and find minimized equation from:

$$F = \sum(0,2,6,8) + d(10,11,12,13,14,15)$$

AB \ CD	00	01	11	10
00	1	0	0	1
01	0	0	0	1
11	x	x	x	x
10	1	0	x	x

$$F = \bar{B}\bar{D} + C\bar{D}$$

4-variable don't care Karnaugh Map

- Find minimized F <u>and its complement</u>

$$F = \sum(0,1,4,10,11,14) + d(10,11,12,13,14,15)$$

AB \ CD	00	01	11	10
00	1	1	X	X
01	1	X	0	X
11	X	0	X	1
10	X	X	1	1

$$F = \bar{A}\bar{C} + AC$$
$$\bar{F} = BD$$

K-map POS

- Up till now, we have been circling 1s to build **SOP** Boolean expressions
- When we circle the 0s, we get the complement, and we also get the **POS**
- The complement of the **POS** can be found with the 1s, like the regular **SOP**.
- The inputs are numbered in the same way the Maxterms are organized

x	y	z	Maxterm
0	0	0	$x + y + z = M_0$
0	0	1	$x + y + \bar{z} = M_1$
0	1	0	$x + \bar{y} + z = M_2$
0	1	1	$x + \bar{y} + \bar{z} = M_3$
1	0	0	$\bar{x} + y + z = M_4$
1	0	1	$\bar{x} + y + \bar{z} = M_5$
1	1	0	$\bar{x} + \bar{y} + z = M_6$
1	1	1	$\bar{x} + \bar{y} + \bar{z} = M_7$

A \ BC	00	01	11	10
0	M_0	M_1	M_3	M_2
1	M_4	M_5	M_7	M_6

K-map POS example

- Find the minimized **POS** for the given truth table

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

A \ BC	00	01	11	10
0	1	0	1	1
1	0	0	1	0

Note the format: this is a Product of Sums

$$F = (B + \bar{C})(\bar{A} + C)$$

K-map POS example

- Given SOP, draw Karnaugh map and find the minimized POS

$F = \sum(3,4,5,6,7,13)$

AB \ CD	00	01	11	10
00	0	0	1	0
01	1	1	1	1
11	0	1	0	0
10	0	1	0	0

$$F = (A + B + C)(B + D)(\bar{A} + D)(\bar{A} + C)$$

This seems long for a minimum POS?
Consider what the expansion for
$\prod(0,1,2,8,10,11,12,14,15)$ would be.

UMBC

Summary

- K-maps allow us to find minimal Boolean expressions by diagraming groups over a grid
- Groups must be powers of 2 in length/width and can wrap around edges of K-map
- Find complement by grouping zeros
- Don’t cares can be part of 1-groups or 0-groups
- Reduced format can be in SOP or POS.

References

- https://steemit.com/logic/@drifter1/logic-design-from-function-to-circuit-using-multi-input-gates
- https://courses.cs.washington.edu/courses/cse370/09wi/LectureSlides/06-Karnaugh.pdf
- https://courses.cs.washington.edu/courses/cse370/99sp/lectures/02-Comb/sld043.htm