

10/12/26



COMP 311: *Computer Organization!*

Lecture 14: K-Maps! Yay!!



The University
of North Carolina
at Chapel Hill

<https://tinyurl.com/comp311-fa26>

Logistics Update



- Lab 2 Due 10/22

lecture 11

Plan for Class Today



- 3:35-4:15: K-Map Definitions
- 4:15-4:20: Break
- 4:20-4:50: K-Map Practice

Guiding Question 😊

How can we simplify Boolean logic without doing a long algebraic derivation?



Karnaugh Maps



Minimizing Logic



- Previously, we have discussed how to systematically convert logic specified as a truth table into sum-of-products (SOP).
- SOP gives us a correct Boolean expression, but not necessarily the simplest one.
- Boolean algebra can simplify expressions, but finding the right transformations can be tricky.
- Today we will discuss a more visual, systematic method for logic minimization, called a *Karnaugh map*, that is based on the *clustering* of terms.
 - Why learn them? To build intuition for logic optimization and understand what modern synthesis tools automate.

Karnaugh Maps (K-Maps)



- Provide a graphical method to find simplified Boolean Equations
- K-maps make combinable terms (like the ones we just saw) easy to see by placing them next to each other on a grid!
- If you follow the procedure properly, you will end up with the most simplified equation!
- As an example consider the 2-variable Karnaugh map for the NAND function.

y

A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

NAND truth table

→

		B	
		0	1
A	0	1	1
	1	1	0

(Red circles highlight the 1s in the K-map)

NAND Karnaugh map

Because of this arrangement it is easier to see relationships between variables!



Karnaugh Maps (K-Maps)

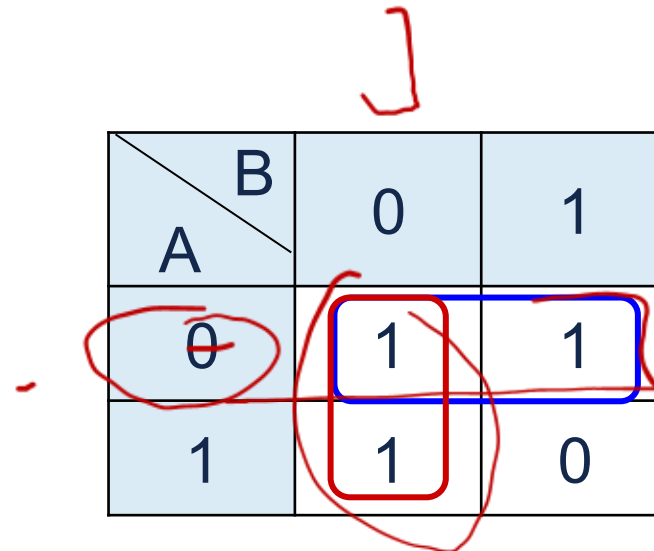
$$y = \overline{AB}$$



A Karnaugh map is an ***alternate truth-table layout*** that places output terms such that shared variable states are adjacent. As an example consider the 2-variable Karnaugh map for the NAND function.

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

NAND truth
table



The Karnaugh map for the NAND function is a 2x2 grid. The top row corresponds to A=0 and the bottom row to A=1. The left column corresponds to B=0 and the right column to B=1. The values are: (0,0)=1, (0,1)=1, (1,0)=1, (1,1)=0. A red circle groups the two 1s in the A=0 row. A blue circle groups the two 1s in the B=0 column. A red bracket above the map indicates the A=0 row.

A \ B	0	1
0	1	1
1	1	0

NAND Karnaugh
map

The first row of 1s is where
 $Y = \overline{A}$

and the left column is
where

$Y = \overline{B}$, which together gives

$$Y = \overline{A} + \overline{B}$$

the De Morgan version of a
NAND gate



Some K-Map Definitions



K-Map Definitions

- Literal: A single variable, may be complemented
 - $A, \bar{A}$
- Product Term: AND/conjunction of literals
 - $\bar{A}\bar{B}C$
 - Not a product term: $\bar{A}\bar{C}B$ $\bar{A} \bar{C} B$
- Minterm: Product term in which all variables appear once.
 - EX: if there are 3 input variables (A, B, C)
 - Minterms: $\bar{A}\bar{B}C, ABC$
 - Not minterms: $A, \bar{A}C$ $\times$
- Maxterm: Sum term in which all variables appear once
 - Mixterm: $A + \bar{B} + C$
 - Not maxterm: $\bar{C}, A + \bar{C}$

Minterms



A	B	C	Minterm	Minterm name
0	0	0	$\bar{A}\bar{B}\bar{C}$	m_0
0	0	1	$\bar{A}\bar{B}C$	m_1
0	1	0	$\bar{A}B\bar{C}$	m_2
0	1	1	$\bar{A}BC$	m_3
1	0	0	$A\bar{B}\bar{C}$	m_4
1	0	1	$A\bar{B}C$	m_5
1	1	0	$AB\bar{C}$	m_6
1	1	1	ABC	m_7



Deriving Equations Using Minterms

- Take the sum of the minterms of the rows whose output is 1

A	B	C	F	Minterm	Minterm Name
0	0	0	1	$\bar{A}\bar{B}\bar{C}$	m_0
0	0	1	0	$\bar{A}\bar{B}C$	m_1
0	1	0	1	$\bar{A}B\bar{C}$	m_2
0	1	1	0	$\bar{A}BC$	m_3
1	0	0	0	$A\bar{B}\bar{C}$	m_4
1	0	1	1	$A\bar{B}C$	m_5
1	1	0	0	$AB\bar{C}$	m_6
1	1	1	1	ABC	m_7

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

 $F(A, B, C) = \sum(m_0, m_2, m_5, m_7)$

Recall: Sum of Products Form



When two or more products (AND) are summed (OR) together

Sum of Products

$$Y = AB + AC$$

Not Sum of Products

$$Y = (A + B)(C + A)$$

Recall: Sum of Products Form

When two or more products (AND) are summed (OR) together

Sum of Products

Not

Sum of Products

$Y =$

Canonical Sum of Products: A sum of products form in which each product contains all literals

Canonical Sum of Products Form

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

Simplified Sum of Products Form

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

Maxterms



A	B	C	Maxterm	Maxterm Name
0	0	0	$A + B + C$	M_0
0	0	1	$A + B + \bar{C}$	M_1
0	1	0	$A + \bar{B} + C$	M_2
0	1	1	$A + \bar{B} + \bar{C}$	M_3
1	0	0	$\bar{A} + B + C$	M_4
1	0	1	$\bar{A} + B + \bar{C}$	M_5
1	1	0	$\bar{A} + \bar{B} + C$	M_6
1	1	1	$\bar{A} + \bar{B} + \bar{C}$	M_7

Maxterms



- Take the product of the maxterms of the rows whose output is 0

A	B	C	F	Maxterm	Maxterm Name
0	0	0	1	$A + B + C$	M_0
0	0	1	0	$A + B + \bar{C}$	M_1
0	1	0	1	$A + \bar{B} + C$	M_2
0	1	1	0	$A + \bar{B} + \bar{C}$	M_3
1	0	0	0	$\bar{A} + B + C$	M_4
1	0	1	1	$\bar{A} + B + \bar{C}$	M_5
1	1	0	0	$\bar{A} + \bar{B} + C$	M_6
1	1	1	1	$\bar{A} + \bar{B} + \bar{C}$	M_7

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

$$F = \Pi(M_1, M_3, M_4, M_6)$$

Why does this work?



A	B	C	F	$\bar{F}$
0	0	0	1	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

Let's create the canonical SOP equation for $\bar{F}$

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

If we negate both sides, we get

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

Applying DeMorgan's, we get the same equation

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



K-maps from Truth Tables

Find the Equation



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

$$y = \overline{A}\overline{B}C + \overline{A}B\overline{C}$$

$$y = \overline{A}C$$

Find the Equation



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

$$\bar{A}\bar{B}C$$

$$\bar{A}BC$$

$$Y = \bar{A}\bar{B}C + \bar{A}BC$$

$$Y = \bar{A}C$$

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0				
	1				

Each box in the K-Map corresponds to a single row in the truth table

3-input Function



	A	B	C	Y
0	0	0	0	<u>0</u>
1	<u>0</u>	<u>0</u>	<u>1</u>	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC	00	01	11	10
A				
0	0			
1				

Each box in the K-Map corresponds to a single row in the truth table

Row 0
 $A = 0, B = 0, C = 0$
 $Y = 0$

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

→

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

↓

Each box in the K-Map corresponds to a single row in the truth table

Row 1
A = 0, B = 0, C = 1
Y = 1

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC		00	01	11	10
A					
0		0 ₀	1 ₁	1	0 ₂
1					

Each box in the K-Map corresponds to a single row in the truth table

Row 2
A = 0, B = 1, C = 0
Y = 0

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0	0 ₀	1 ₁	1 ₃	0 ₂
	1				

Each box in the K-Map corresponds to a single row in the truth table

Row 3

A = 0, B = 1, C = 1

Y = 1

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC		00	01	11	10
A					
0		0 ₀	1 ₁	1 ₃	0 ₂
1		0 ₄	0 ₅	0 ₇	0 ₆

Each box in the K-Map corresponds to a single row in the truth table

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

Gray Code

↓

BC					
A		00	01	11	10
	0	0 ₀	1 ₁	1 ₃	0 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

Gray Code



- Successive values differ by only one bit

→ Gray Code

00
01
11
10

Binary Code

00
01
10
11



In binary code, two bits flip when we go from 1 to 2

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC		00	01	11	10
A					
0		0 ₀	1 ₁	1 ₃	0 ₂
1		0 ₄	0 ₅	0 ₇	0 ₆

Each square in a K-map differs from an adjacent square by a change in a single literal

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

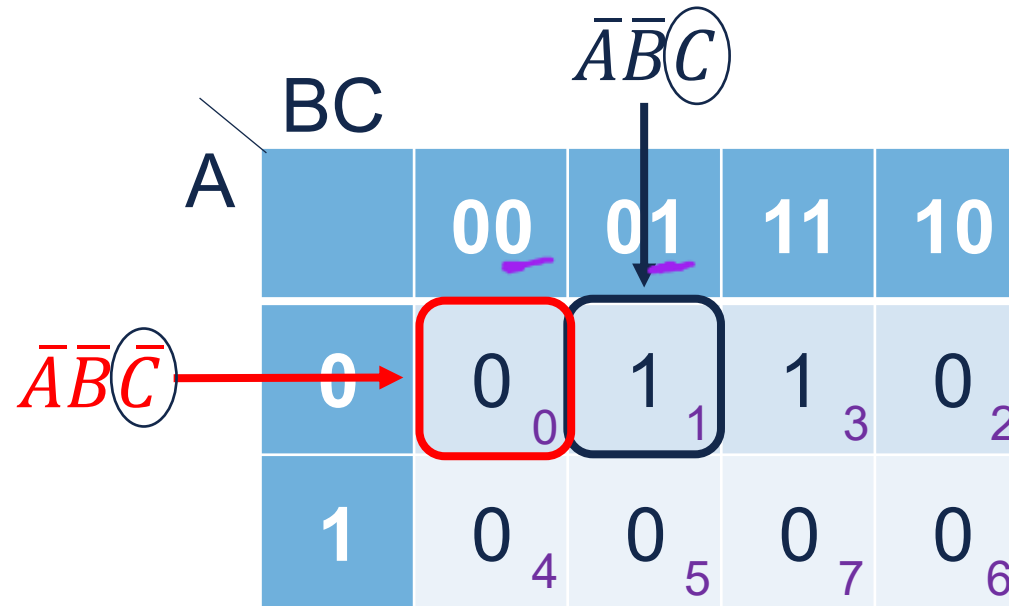
		BC			
A		00	01	11	10
	0	0 ₀	1 ₁	1 ₃	0 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

Let's examine the squares
around box 1

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

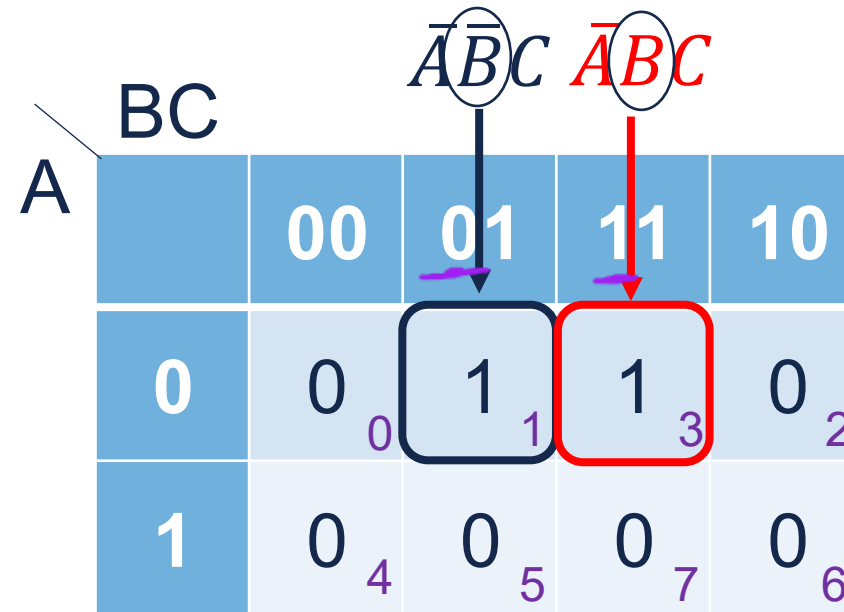


Box 0 differs by the C literal

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

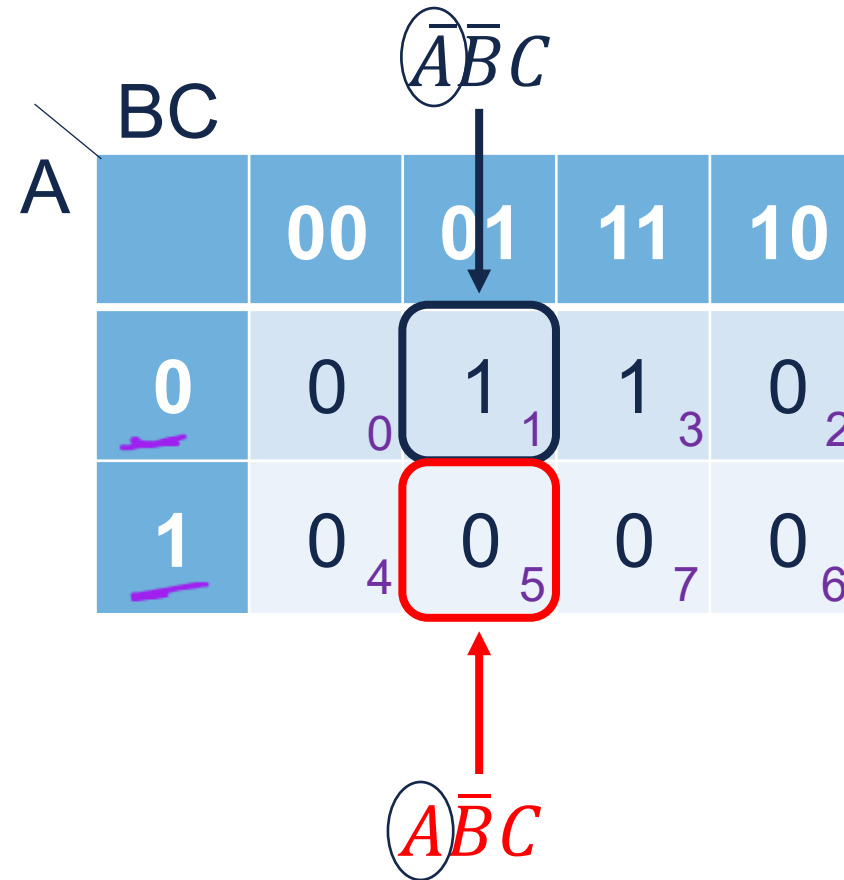


Box 3 differs by the B literal

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0



Box 5 differs by the A literal

3-input Function



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

		BC			
A		00	01	11	10
	0	0 ₀	1 ₁	1 ₃	0 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

$\bar{A}\bar{B}C$ points to cell (0,1)

Diagonal entries (1,1) and (1,3) are not adjacent

Diagonal entries are not considered adjacent

Simplification using the K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC		00	01	11	10
A					
0		0 ₀	1 ₁	1 ₃	0 ₂
1		0 ₄	0 ₅	0 ₇	0 ₆

To graphically find the simplified equation, first group the terms together. Then, find the literals that the group has in common.

Simplification using the K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

A \ BC					
		00	01	11	10
	0	0	1 ₁	1 ₃	0 ₂
1	0	0 ₄	0 ₅	0 ₇	0 ₆

Which literals do these entries share?

Simplification using the K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

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

Annotations: Purple arrows point to the 01 and 11 columns. A red box highlights the group of cells (A=0, B=0, C=1), (A=0, B=1, C=1), (A=1, B=0, C=1), and (A=1, B=1, C=1). A purple circle highlights the cell (A=0, B=0, C=0). A purple arrow points to the cell (A=0, B=0, C=0).

We can see that the entries inside of the group each contain $\bar{A}$ and C .

Simplification using the K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

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

The literals that the group share form the simplified equation.

$$Y = \bar{A}C$$

Simplification using the K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC		00	01	11	10
A					
0		0 ₀	1 ₁	1 ₂	0 ₃
1		0 ₄	0 ₅	0 ₆	0 ₇

If we had used binary code instead of gray code, this visual simplification would not have been possible!

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

		BC			
		00	01	11	10
A	0	0	1	3	2
	1	4	5	7	6

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC		00	01	11	10
A					
0		0	1	3	2
1		4	5	7	6

Step 1: Fill in the K-map

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

A	BC			
	00	01	11	10
0	1 ₀	1 ₁	1 ₃	1 ₂
1	0 ₄	0 ₅	0 ₇	0 ₆

Step 1: Fill in the K-map

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0				
	1				
		0	1	3	2
		4	5	7	6

Step 2: Group the terms together

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0				
	1				
		4	5	7	6

Step 2: Group the terms together

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

		BC			
A		00	01	11	10
	0	1 ₀	1 ₁	1 ₃	1 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

Step 3: Find the literals that the group has in common

A

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0	1 ₀	1 ₁	1 ₃	1 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

Step 3: Find the literals that the group has in common

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0	1 ₀	1 ₁	1 ₃	1 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

Step 4: Derive the equation.

Example 2



	A	B	C	Y
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

BC					
A		00	01	11	10
	0	1 ₀	1 ₁	1 ₃	1 ₂
	1	0 ₄	0 ₅	0 ₇	0 ₆

Step 4: Derive the equation.

$$Y = \bar{A}$$



K-Map Grouping Rules

K-Map Grouping Rules



All squares in each circle must contain 1's

A \ B		0	1
	0	0	0
	1	1	1

Right

A \ B		0	1
	0	0	0
	1	1	1

Wrong



K-Map Grouping Rules



Groups may be horizontal or vertical, but not diagonal

	B		
A		0	1
	0	0	1
	1	1	1

Right

	B		
A		0	1
	0	0	1
	1	1	0

Wrong



K-Map Grouping Rules

Groups must contain 2^n cells where $n = 0, 1, 2$, etc.

$2^0 = 1$

	B		
A		0	1
	0	0	0
	1	1	0

Right

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

Wrong

$2^1 = 2$



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

Right

$2^2 = 4$

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

Right

K-Map Grouping Rules



Each group must be as large as possible

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

Right

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

Wrong



K-Map Grouping Rules



Each cell containing a 1 must be in at least one group

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

Right

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

Wrong

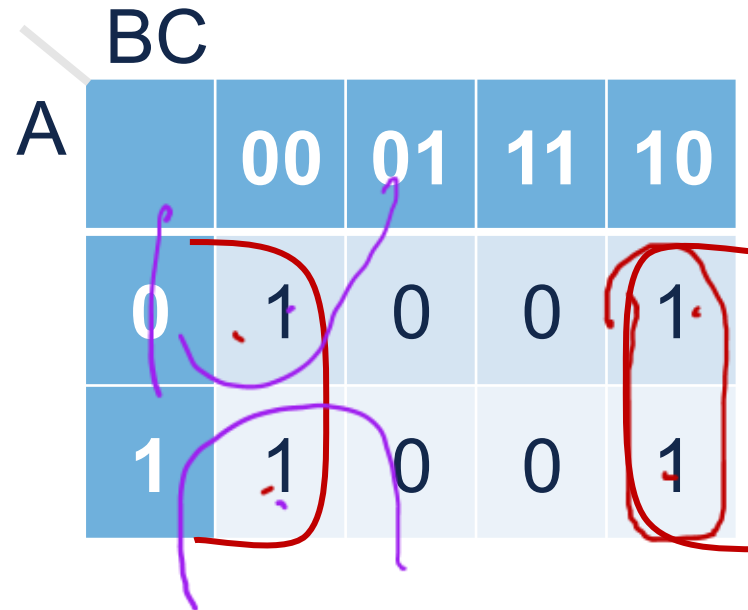


K-Map Grouping Rules



Groups may wrap around the table

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

The K-map is a 2x4 grid. The columns are labeled 00, 01, 11, 10 and the rows are labeled 0, 1. The cells contain 1s at (0,00), (0,10), (1,00), and (1,10). There are two red groupings: a vertical group of two cells (0,00) and (1,00), and a vertical group of two cells (0,10) and (1,10). There are two purple groupings: a horizontal group of two cells (0,00) and (0,10), and a horizontal group of two cells (1,00) and (1,10). These groupings illustrate the wrap-around rule where the first and last columns are adjacent.

K-Map Grouping Rules



Every circle should contain at least one unique 1

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

Right

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

Wrong



$$Y = AC + B\overline{C}$$

Ex: Reading a K-Map



A \ BC	BC			
	00	01	11	10
0	0 ₀	0 ₁	0 ₃	1 ₂
1	0 ₄	1 ₅	1 ₇	1 ₆

$$Y = AC + B\bar{C}$$

Ex: Reading a K-Map



A \ BC				
	00	01	11	10
0	0 ₀	0 ₁	0 ₃	1 ₂
1	0 ₄	1 ₅	1 ₇	1 ₆

Wrong – the blue group does not contain a unique 1. This grouping would produce a logically equivalent equation, but it would not be the most simplified.



A \ BC				
	00	01	11	10
0	0 ₀	0 ₁	0 ₃	1 ₂
1	0 ₄	1 ₅	1 ₇	1 ₆

Wrong – the blue group is not a power of 2. We cannot derive a term for this group.



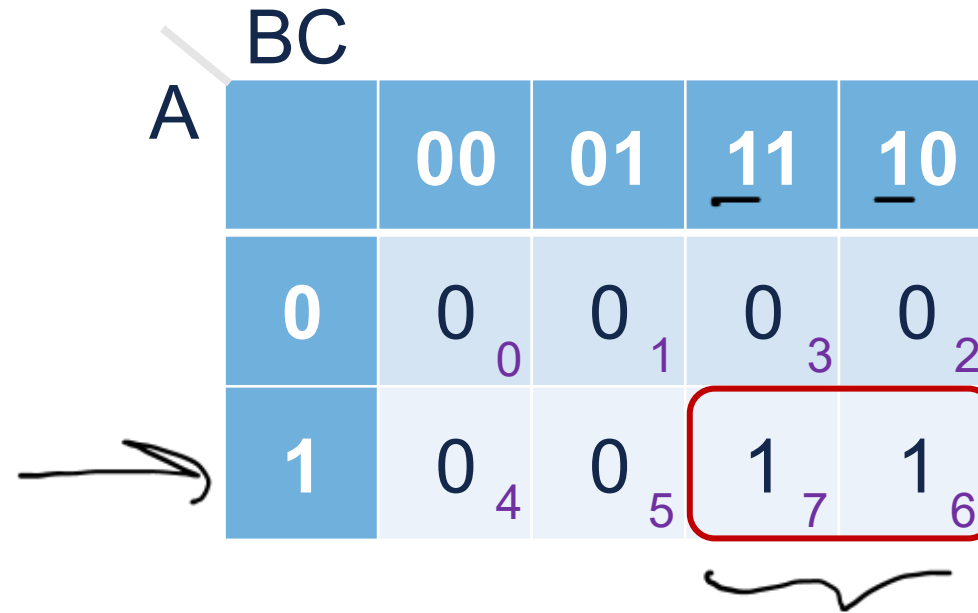


K-Map Groups

Ex1: Reading A K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1



Ex1: Reading A K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

A \ BC				
	00	01	11	10
0	0 ₀	0 ₁	0 ₃	0 ₂
1	0 ₄	0 ₅	1 ₇	1 ₆

$$Y = AB$$

Ex2: Reading A K-Map



	A	B	C	Y
0	0	0	0	0
→ 1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

	BC			
A	00	01	11	10
0	0 ₀	1 ₁	0 ₃	0 ₂
1	0 ₄	0 ₅	0 ₇	0 ₆

$$y = \overline{A} \overline{B} C$$

Ex2: Reading A K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

A \ BC	BC			
	00	01	11	10
0	0 ₀	1 ₁	0 ₃	0 ₂
1	0 ₄	0 ₅	0 ₇	0 ₆

$$Y = \bar{A}\bar{B}C$$

Ex3: Reading a K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

→

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

Handwritten annotations: A circle around '01' in the top row, a circle around '11' in the top row, and a red box around the bottom row (1,00 to 1,11). Below the red box, a handwritten '1' is followed by a bracket grouping the three cells.

$$Y = A\bar{B} + AC$$

Ex3: Reading a K-Map



	A	B	C	Y
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

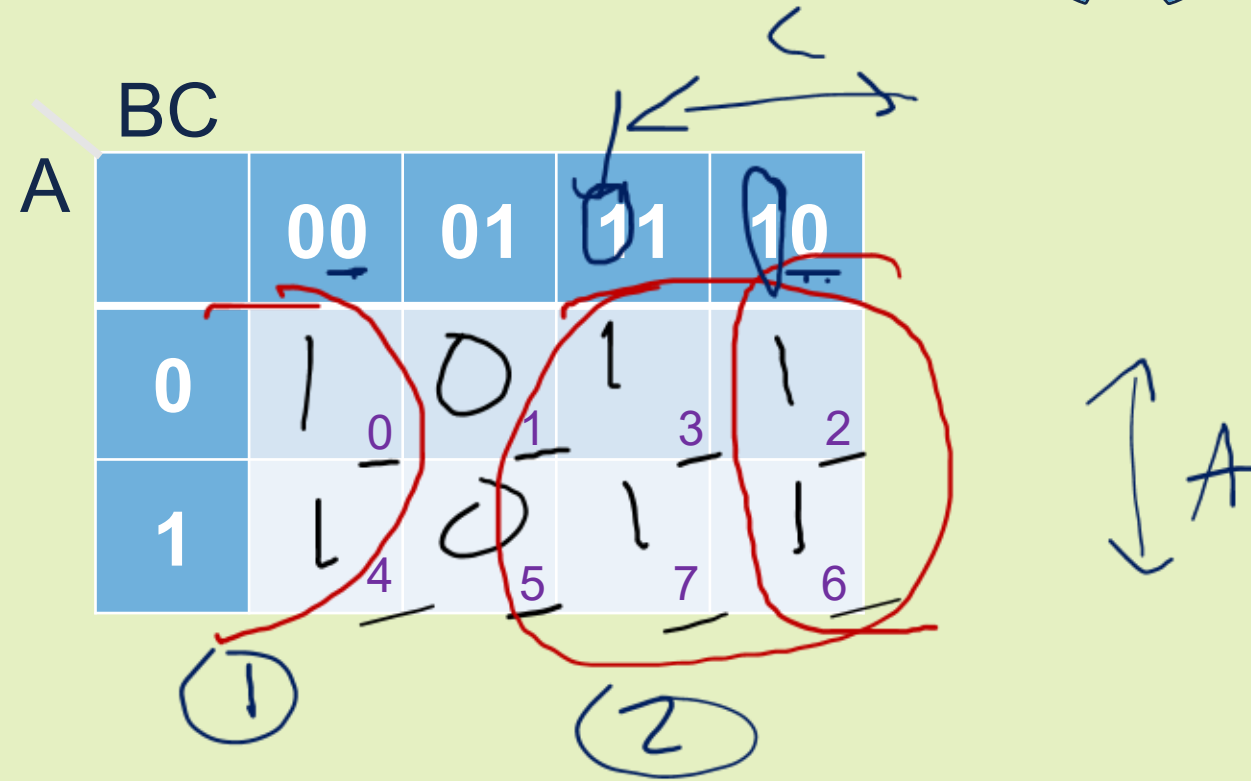
A \ BC				
	00	01	11	10
0	0 ₀	0 ₁	0 ₃	0 ₂
1	1 ₄	1 ₅	1 ₇	0 ₆

$$Y = A\bar{B} + AC$$

Use a K-Map to find the simplified SOP Equation



	A	B	C	Y
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1



$$y = \overline{C} + B$$

Use a K-Map to find the Simplified SOP Equation



	A	B	C	Y
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

A \ BC				
	00	01	11	10
0	1 ₀	0 ₁	1 ₃	1 ₂
1	1 ₄	0 ₅	1 ₇	1 ₆

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

4-variable K-Map

		CD			
AB		00	01	11	10
	00				
	01				
	11				
	10				

Use Gray code order so that adjacent cells only differ by one literal.

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

4-variable K-Map

		CD			
AB		00	01	11	10
	00				
	01				
	11				
	10				

This cell is 0000, so it corresponds to row 0



0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

A	B	C	D	Y
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	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	

4-variable K-Map

		CD			
AB		00	01	11	10
	00				
	01				
	11				
	10				

This cell is 0001, so it corresponds to row 1



0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

A	B	C	D	Y
0	0	0	0	
0	0	0	<u>1</u>	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	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	

4-variable K-Map

		CD			
AB		00	01	11	10
00					
01					
11					
10					

This cell is 0110, so it corresponds to row 6

	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7					
8	0	1	1	1	
9	1	0	0	0	
10	1	0	0	1	
11	1	0	1	0	
12	1	0	1	1	
13	1	1	0	0	
14	1	1	0	1	
15	1	1	1	0	
	1	1	1	1	

4-variable K-Map

		CD			
AB		00	01	11	10
	00				
	01				
	11				
	10				

This cell is 1111, so it corresponds to row 15



0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

4-variable K-Map

		CD			
AB		00	01	11	10
	00	0	1	3	2
	01	4	5	7	6
	11	12	13	15	14
	10	8	9	11	10

Use Gray code order so that adjacent cells only differ by one literal.

	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

Ex: 4-variable K-Map

		CD			
AB		00	01	11	10
	00				
	01				
	11				
	10				

	A	B	C	D	Y
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

Ex: 4-variable K-Map

		CD			
AB		00	01	11	10
	00	0 ₀	0 ₁	1 ₃	1 ₂
	01	1 ₄	1 ₅	0 ₇	0 ₆
	11	1 ₁₂	1 ₁₃	0 ₁₅	0 ₁₄
	10	0 ₈	0 ₉	0 ₁₁	1 ₁₀

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

	A	B	C	D	Y
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

Ex: 4-variable K-Map

		CD			
AB		00	01	11	10
	00	0 ₀	0 ₁	1 ₃	1 ₂
	01	1 ₄	1 ₅	0 ₇	0 ₆
	11	1 ₁₂	1 ₁₃	0 ₁₅	0 ₁₄
	10	0 ₈	0 ₉	0 ₁₁	1 ₁₀

$$Y = \underline{BC} + \bar{A}\bar{B}C + \bar{B}C\bar{D}$$

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

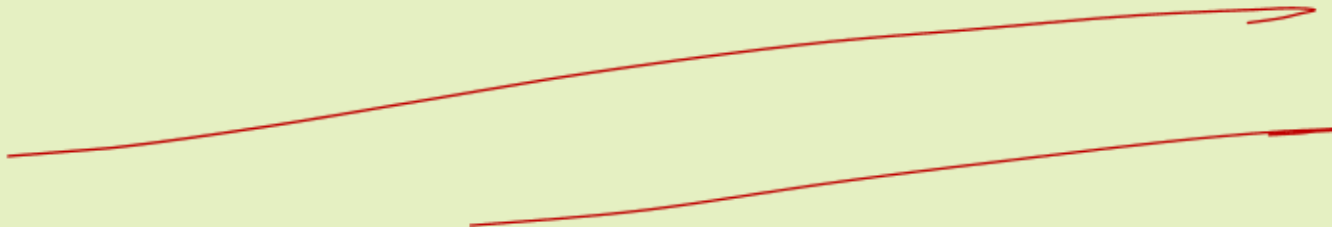
	A	B	C	D	Y
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

Worksheet Q2



do both

2.1 - 2.2



Q2.1: Reading a K-Map

→ ABCD - 1
ABC - 2
AB - 4
A - 8
1 - 16



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

AB - 4

A - 8

1 - 16

$$\overline{C}\overline{D} + BD + AB$$

WS Q2.1: Reading a K-Map



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

$$Y = \bar{C}\bar{D} + BD + AB$$

WS Q2.2: Reading a K-Map



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

WS Q2.2: Reading a K-Map



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

$$Y = \bar{A} + \bar{B}\bar{D}$$

WS 2.2: Reading a K-Map



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

Q2.2: Reading a K-Map



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

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

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

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

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + ACD + \bar{A}C\bar{D} + A\bar{B}C\bar{D}$$

AB \ CD	CD			
	00	01	11	10
00				
01				
11				
10				

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + ACD + \bar{A}C\bar{D} + A\bar{B}C\bar{D}$$

AB \ CD	CD			
	<u>00</u>	01	11	10
00	1			
01	1			
11	1			
10	1			

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + \textcolor{red}{ABC}\bar{D} + ACD + \bar{A}\bar{C}\bar{D} + A\bar{B}C\bar{D}$$

AB \ CD		CD			
		00	01	11	<u>10</u>
00	1				
01	1				
<u>11</u>	1			<u>1</u>	
10	1				

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + \textcolor{red}{ACD} + \bar{A}C\bar{D} + A\bar{B}C\bar{D}$$

AB \ CD		CD			
		00	01	11	10
00	1				
01	1				
11	1		<u>1</u>	1	
10	1		<u>1</u>		

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + ACD + \bar{A}C\bar{D} + A\bar{B}C\bar{D}$$

		CD				
AB		00	01	11	10	
	00	1			1	
	01	1			1	
	11	1		1	1	
	10	1		1		

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + ACD + \bar{A}C\bar{D} + \textcolor{red}{A}\bar{B}C\bar{D}$$

		CD			
AB		00	01	11	10
	00	1			1
	01	1			1
	11	1		1	1
	10	1		1	1

→

↑

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + ACD + \bar{A}C\bar{D} + A\bar{B}C\bar{D}$$

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

$D \rightarrow AC$

Simplifying an Equation Using a K-Map



$$Y = \bar{C}\bar{D} + ABC\bar{D} + ACD + \bar{A}C\bar{D} + A\bar{B}C\bar{D}$$

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

$$Y = AC + \bar{D}$$

Recall: Boolean Algebra simplification



$$ABC + AB\bar{C} + \overline{AC}B + ACB + \overline{\overline{AB}\bar{C}\bar{D}} + \overline{\overline{AB} + \bar{C}}$$

$$AB(C + \bar{C}) + B(\overline{AC} + AC) + AB + C + D + (\overline{\overline{AB}})C$$

$$AB(1) + B(1) + AB + C + D + (A + \bar{B})C$$

$$AB + B + AB + C + D + AC + \bar{B}C$$

$$AB + B + C + D + AC + \bar{B}C$$

$$B(A + 1) + C(1 + A) + \bar{B}C + D$$

$$B(1) + C(1) + \bar{B}C + D$$

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

$$B + C(1 + \bar{B}) + D$$

$$B + C(1) + D$$

$$B + C + D$$



Let's use a K-Map Instead

1. Put the equation in SOP form.
2. Fill out the K-Map.
3. Find the simplified equation.

$$ABC + AB\bar{C} + \overline{AC}B + ACB + \overline{\overline{AB}\overline{C}\overline{D}} + \overline{AB} + \bar{C}$$

Using a K-Map to Simplify a Boolean Equation



1. Put the equation in SOP form
2. Fill out the K-Map
3. Find the simplified equation

$$ABC + AB\bar{C} + \overline{AC}B + ACB + \overline{\overline{AB}\overline{C}\overline{D}} + \overline{\overline{AB} + \bar{C}}$$

AB \ CD	CD			
	00	01	11	10
00				
01				
11				
10				

Using a K-Map to Simplify a Boolean Equation



1. Put the equation in SOP form

De Morgan's!

$$ABC + AB\bar{C} + \overline{AC}B + ACB + \overline{\overline{AB}\overline{C}\overline{D}} + \overline{AB + C}$$

$$ABC + AB\bar{C} + \bar{A}B + \bar{A}C + ACB + AB + C + D + AC + \bar{B}C$$

Using a K-Map to Simplify a Boolean Equation



1. Put the equation in SOP form
2. Fill out the K-Map

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

A 4x4 Karnaugh map for variables A, B, C, and D. The map is filled with 1s in all cells except for (AB=00, CD=00) and (AB=10, CD=00). Hand-drawn black lines group the 1s into four prime implicants: a horizontal group of four 1s in the middle row (AB=01), a vertical group of four 1s in the middle column (CD=01), a horizontal group of four 1s in the bottom row (AB=10), and a vertical group of four 1s in the rightmost column (CD=10).

$$ABC + ABC\bar{C} + \bar{A}B + \bar{A}C + ACB + AB + C + D + AC + \bar{B}C$$

Using a K-Map to Simplify a Boolean Equation



1. Put the equation in SOP form
2. Fill out the K-Map
3. Find the simplified equation

$$B + C + D$$

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