

# Penn State Abington

## CMPEN 271

### Lecture Set #7

# K-Maps

R. Avanzato © 2013-2015

#### Topics:

- Karnaugh Maps (K-Maps) I

**Video part 1 of 3** ←

- 
- Karnaugh Maps II

**Video part 2 of 3**

- 
- Sample Exercises

**Video part 3 of 3**

# What is a Karnaugh Map (K-MAP)?

- The **Karnaugh Map** is a technique invented by Maurice Karnaugh in the 1950s to simply digital logic minimization.
- A K-map is a **2-d, graphical representation** of the 1's and 0's from a truth table
- Using graphical methods of **circling groups of 1's**, K-maps result in the minimization of a Boolean function
- The K-map method is generally **quicker and more reliable** than using algebra by hand
- If a K-map is used correctly, **no Boolean algebra is necessary** -- the function will be completely minimized
- In order to use a K-map you must first **start with a truth table** or a Boolean function to be minimized
- K-maps are useful for **small tasks**, and for demonstrating minimization techniques.
- **Computer programs** exist for the minimization of complex Boolean function (e.g. Espresso II). Software such as Multisim can also minimize Boolean functions.

# Review of Algebraic Approach

Example:

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	0

$$F = \Sigma m (2, 3)$$

$$= A'BC' + A'BC$$

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

$$= A'B \text{ (final answer)}$$

Logical Adjacency (discuss)! Can we leverage?

# K-map -1

Same Example:

Start with truth table:

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	0

$$A'B'C' = m_2$$

$$A'BC = m_3$$

Step 1: Construct and fill K-map

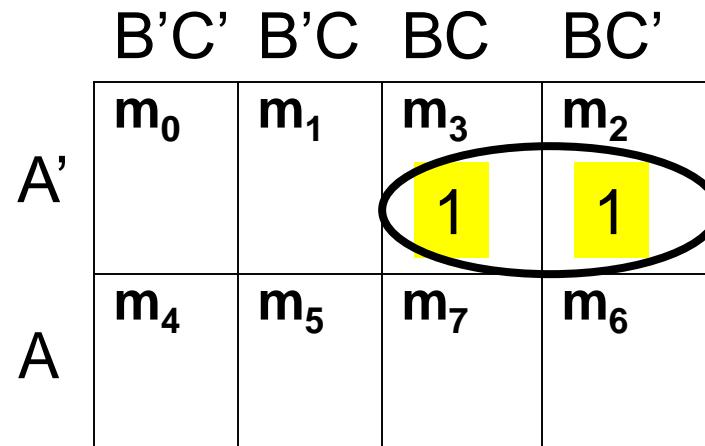
	B'C'	B'C	BC	BC'
A'	$m_0$	$m_1$	$m_3$	$m_2$
A	$m_4$	$m_5$	$m_7$	$m_6$

3-var. K-map

- 1) K-map - same contents as truth table
- 2) normally show 1's; assume empty squares are 0's
- 3) notice labels and ordering (based on log. adj.)

# K-map -2

Step 2: Group 1's according to rules and  
Step 3: Obtain product terms to be OR'd



3-var. K-map

Directly from K-map:  $F = A'B$  (now we are done!)

# K-map -3

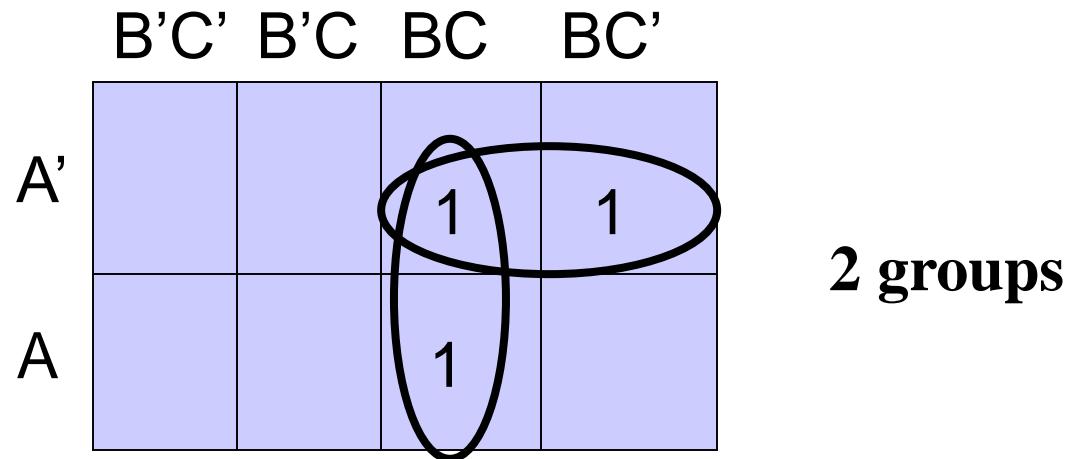
**Step #1:** Draw K-map, label, and fill with 1's and 0's (optionally leave 0's blank/empty) from truth table.

**Step #2:** Group (circle) all 1's in “rectangular” sets of 1, 2, 4, or 8. Each group corresponds to a product term in answer. LARGER groups are better. FEWER groups are better. Groups can OVERLAP. All 1's must be circled. (We will discuss more formal methods later).

**Step #3:** Write down a product term for each group of 1's. Any variable that changes (from inverted to non-inverted) throughout the group will vanish in product term. “OR” all product terms together to create final simplified function. More variables will vanish in a product term with the larger groups.

# K-map -4

Given:  $F(A,B,C) = \sum m (2, 3, 7)$

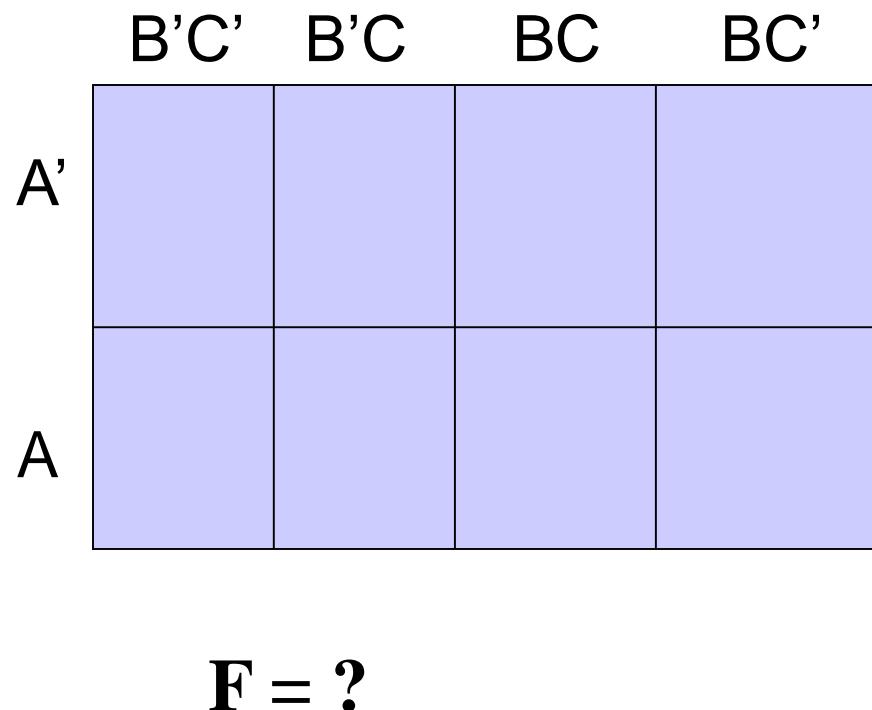


Answer:  $F = A'B + BC$

# K-map - 5

Given arbitrary truth-table, simplify  $F(A,B,C)$ :

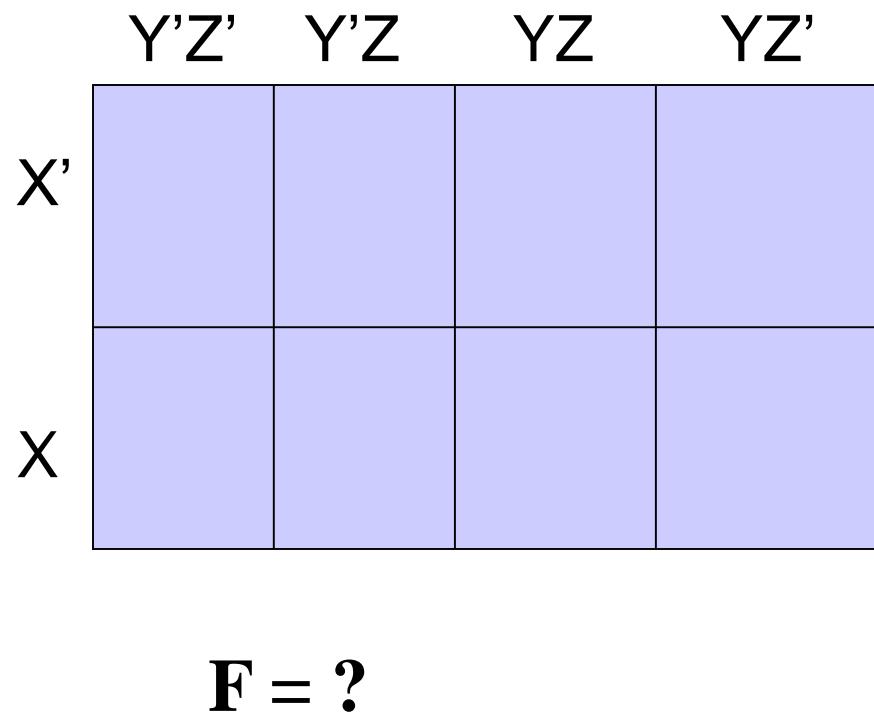
A	B	C		F
0	0	0		0
0	0	1		1
0	1	0		0
0	1	1		1
1	0	0		0
1	0	1		1
1	1	0		1
1	1	1		1



# K-map - 6

Given an arbitrary truth-table, simplify  $F(X,Y,Z)$ :

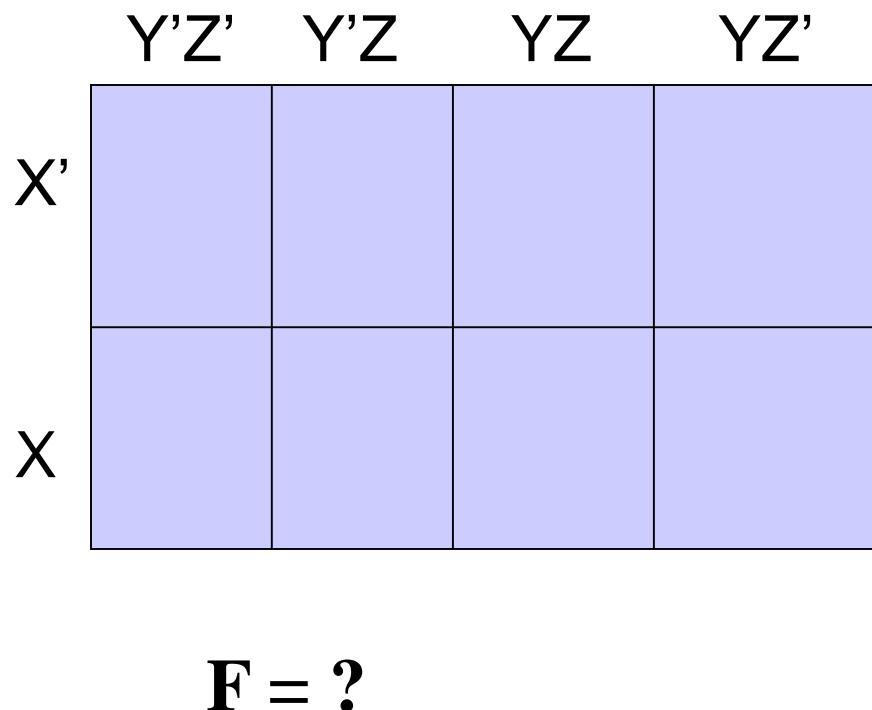
X	Y	Z		F
0	0	0		0
0	0	1		1
0	1	0		1
0	1	1		0
1	0	0		0
1	0	1		0
1	1	0		1
1	1	1		0



# K-map - 7

Given an arbitrary truth-table, simplify  $F(X,Y,Z)$ :

X	Y	Z		F
0	0	0		1
0	0	1		0
0	1	0		1
0	1	1		0
1	0	0		1
1	0	1		0
1	1	0		1
1	1	1		0



**(Note: edges of K-map connect!)**

# Penn State Abington

## CMPEN 271

### Lecture Set #7

# K-Maps

R. Avanzato ©

#### Topics:

- Karnaugh Maps (K-Maps) I

Video part 1 of 3

- 
- Karnaugh Maps II

Video part 2 of 3 ←

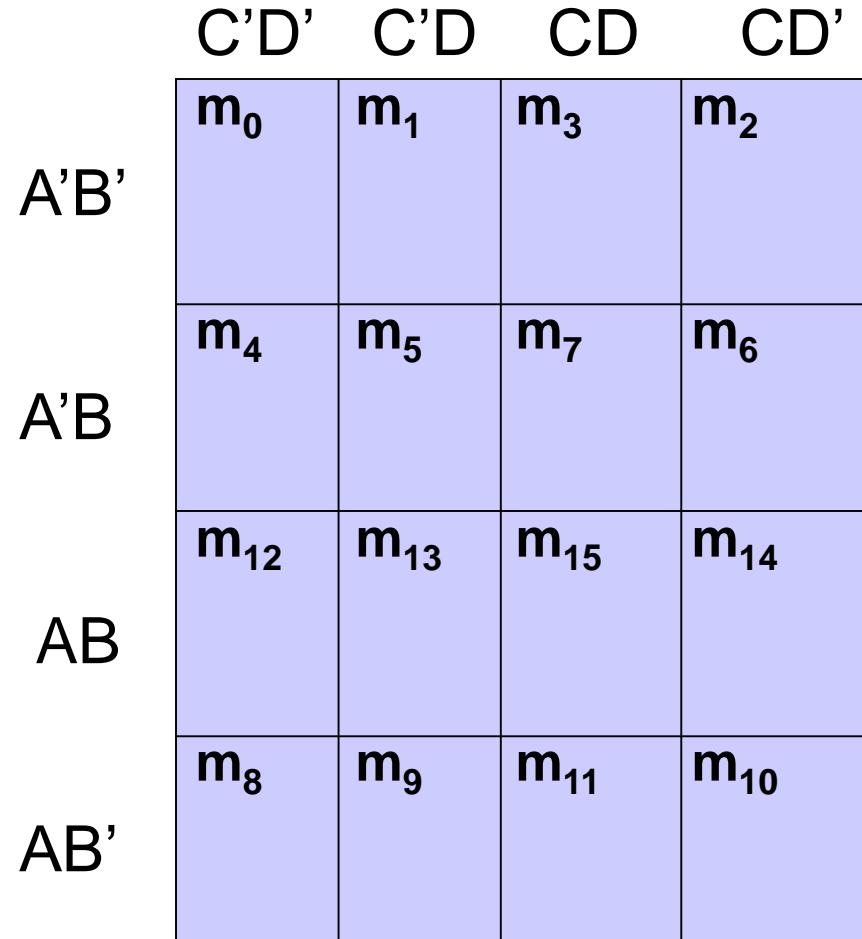
- 
- Sample Exercises

Video part 3 of 3

# 4-variable K-map (16 squares)

Notice the ordering; groups of 1, 2, 4, 8, 16 possible;  
edges are connected

A	B	C	D	F
0	0	0	0	$m_0$
0	0	0	1	$m_1$
0	0	1	0	$m_2$
0	0	1	1	$m_3$
0	1	0	0	$m_4$
0	1	0	1	$m_5$
0	1	1	0	$m_6$
0	1	1	1	$m_7$
1	0	0	0	$m_8$
1	0	0	1	$m_9$
1	0	1	0	$m_{10}$
1	0	1	1	$m_{11}$
1	1	0	0	$m_{12}$
1	1	0	1	$m_{13}$
1	1	1	0	$m_{14}$
1	1	1	1	$m_{15}$



# K-map - 8

4-variable K-map -- 16 squares

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

	C'D'	C'D	CD	CD'
A'B'	$m_0$	$m_1$	$m_3$	$m_2$
		1	1	
A'B	$m_4$	$m_5$	$m_7$	$m_6$
	1		1	1
AB	$m_{12}$	$m_{13}$	$m_{15}$	$m_{14}$
	1		1	1
AB'	$m_8$	$m_9$	$m_{11}$	$m_{10}$
				1
				F = ?

# K-map - 9

	$C'D'$	$C'D$	$CD$	$CD'$
$A'B'$		1	1	1
$A'B$	1	1	1	1
$AB$	1	1	1	1
$AB'$				

$F = ?$

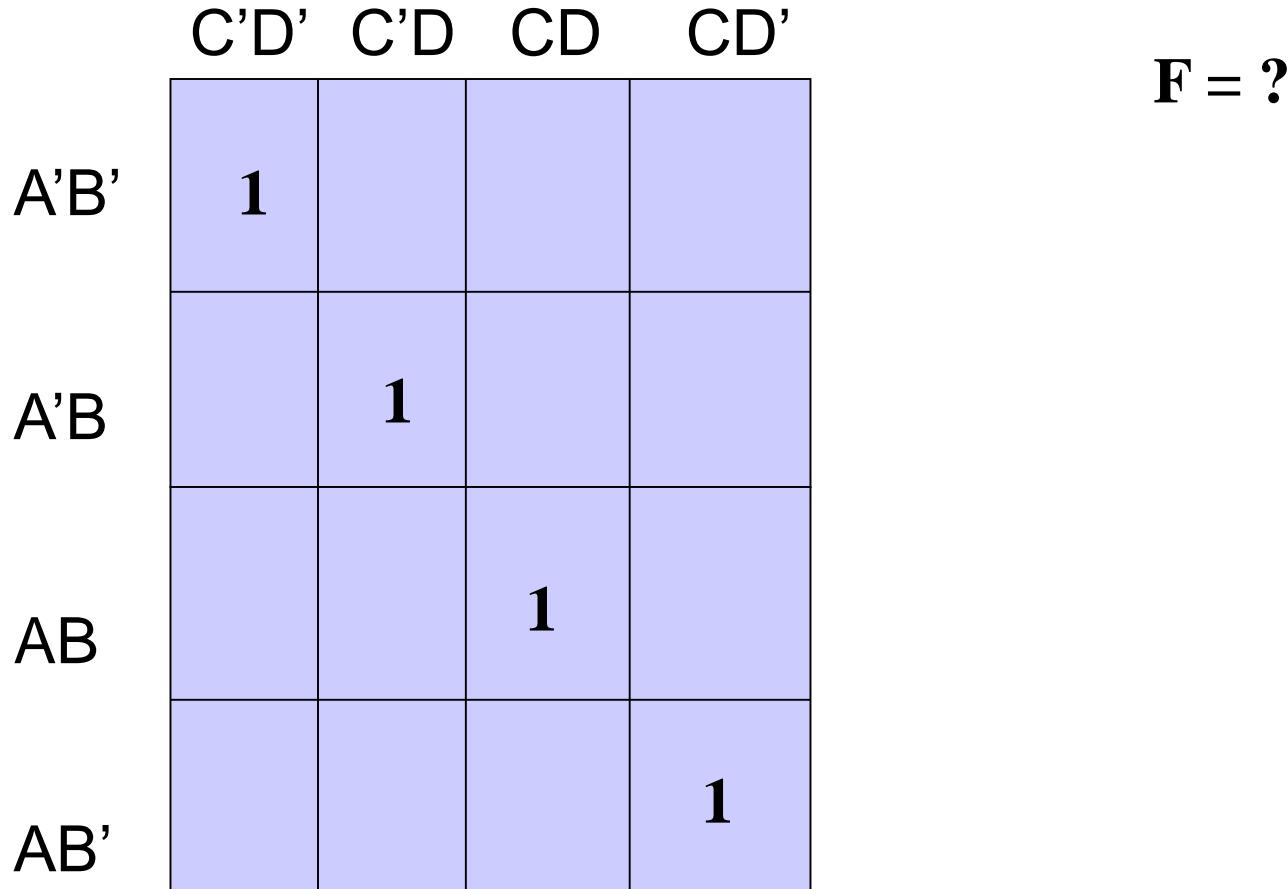
# K-map - 10

	$C'D'$	$C'D$	$CD$	$CD'$	
$A'B'$	1		1	1	$F = ?$
$A'B$	1	1			
$AB$	1	1			
$AB'$	1			1	

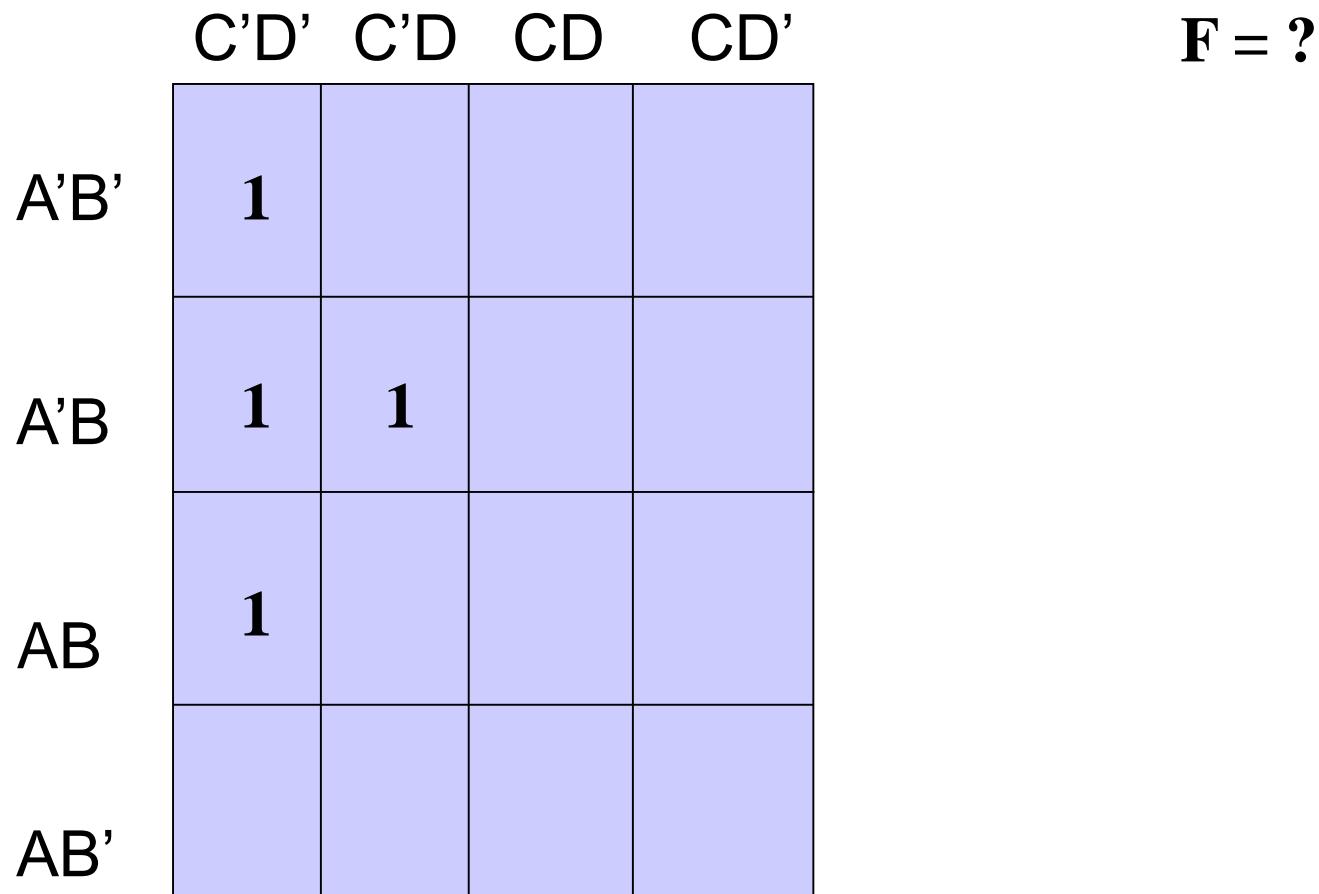
# K-map - 11

	$C'D'$	$C'D$	$CD$	$CD'$	$F = ?$
$A'B'$		1	1	1	
$A'B$	1	1			
$AB$	1	1			
$AB'$		1		1	

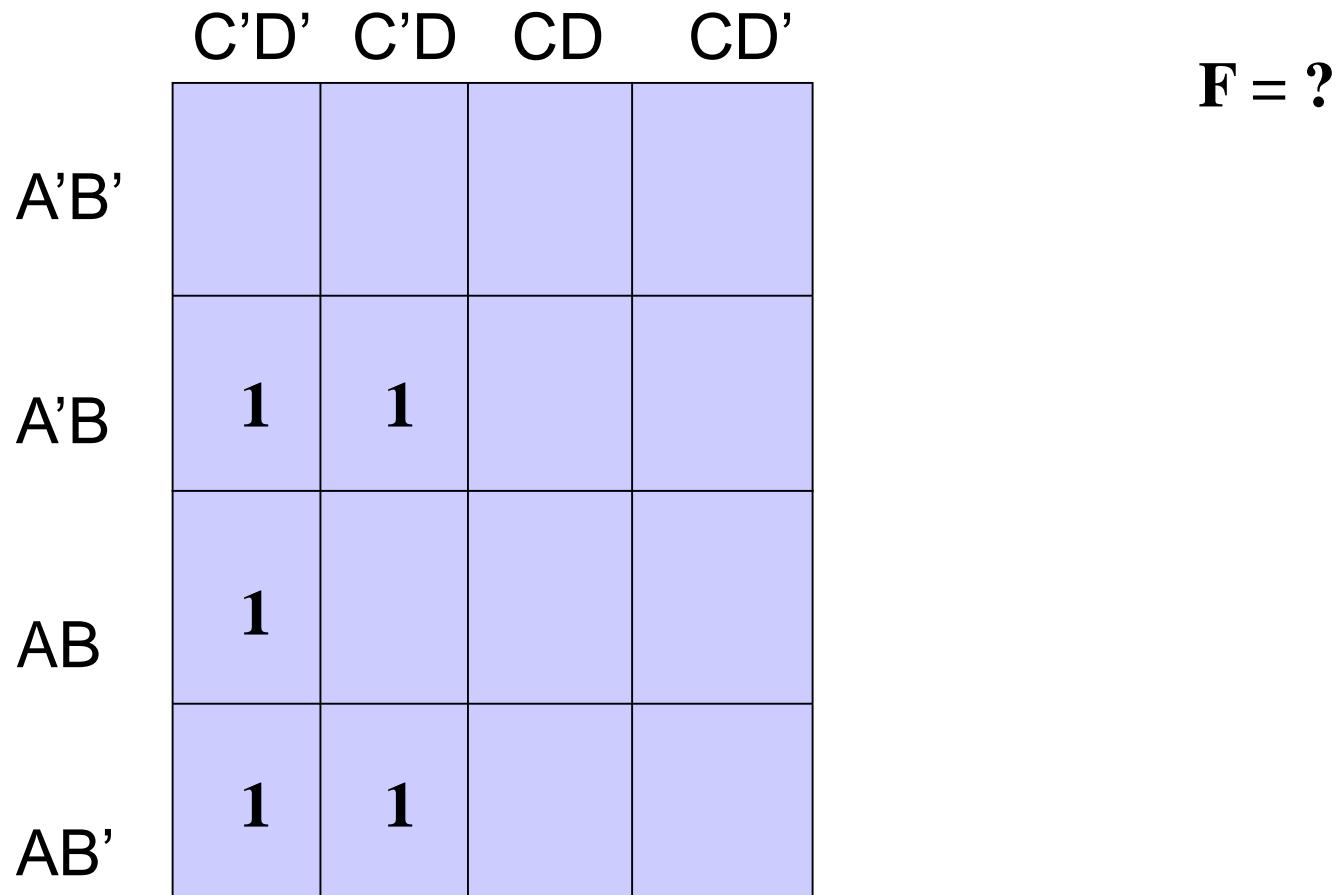
# K-map - 12



# K-map - 13



# K-map - 14



# K-map - 15

	$Y'Z'$	$Y'Z$	$YZ$	$YZ'$	$F = ?$
$W'X'$	1	1	1	1	
$W'X$		1			
$WX$		1			
$WX'$	1	1	1	1	

# K-map - 16

	$Y'Z'$	$Y'Z$	$YZ$	$YZ'$	
$W'X'$	1	1	1	1	$F = ?$
$W'X$	1	1	1	1	
$WX$	1	1	1	1	
$WX'$	1	1	1	1	

# K-map - 17

	$Y'Z'$	$Y'Z$	$YZ$	$YZ'$	$F = ?$
$W'X'$	1	1	1	1	
$W'X$	1		1	1	
$WX$	1	1	1	1	
$WX'$	1	1	1	1	

# K-map Summary

- K-map is a **graphical method** for minimizing Boolean functions. It is an alternative to algebraic methods.
- If a K-map is performed correctly, then the resulting function is minimum -- **no further reduction** (by algebra or any other method) can reduce it further.
- When grouping 1's in a K-map, **maximize** the size of the group, **minimize** the number of groups. Groups can overlap.
- In a K-map, you are permitted to group 1's in **group sizes** of 1, 2, 4, 8, 16. Groups can be square or rectangular
- **Edges** of a K-map are connected (also the **4 -corners** are in a group)
- If you group the 0's in a K-map (instead of the 1's), then you will obtain the **complement** (inverse) of F. Sometimes this is useful if you want F' on SOP or if F has very few zero entries in the K-map.
- We will examine more **formal methods** for K-maps in a future lecture.
- There are **computer programs** which can automate the minimization of Boolean functions, especially when there are a large number of variables (example: Quine-McKluskey, espresso s/w)

# What you should know...

- Be able to explain to **purpose** of the K-map method.
- Understand the **pros and cons** of K-map versus algebra for Boolean function minimization
- Be able to **minimize** any 3 or 4-variable function with 2 methods: 1) algebra, and 2) K-map. Start with truth table or Boolean expression in SOP or Sum of minterms form.
- Be able to **construct and label** a 3 and 4-variable K-map. Follow correct ordering and labeling of rows and columns. Use variable defined in problem statement.
- Be able to determine minimized  $F'$  (**inverse of F**) from a K-map (circle 0's in K-map instead of 1's)

# Further Reading

- Mano Kime, Logic and Computer Design Fundamentals, Prentice Hall, Chapter 2.
- Tocci R., Digital Systems, Prentice Hall, Chapter 3&4.
- [www.howstuffworks.com](http://www.howstuffworks.com)  
“How Boolean Logic Works”

# Penn State Abington

## CMPEN 271

### Lecture Set #7

# K-Maps

R. Avanzato ©

#### Topics:

- Karnaugh Maps (K-Maps) I

Video part 1 of 3

- 
- Karnaugh Maps II

Video part 2 of 3

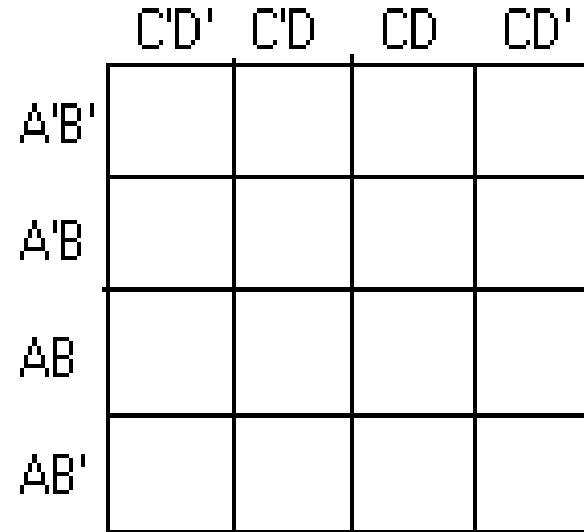
- 
- Sample Exercises

Video part 3 of 3 ←

# Sample Exercises

#1.- What is the minimal function described by this K-map?

- a)  $F = AC + B'C'$
- b)  $F = C'D' + CD$
- c)  $F = A'B' + C'D'$
- d)  $F = A'C' + BC'$
- e)  $F = A'B'C'D' + ABC'D$



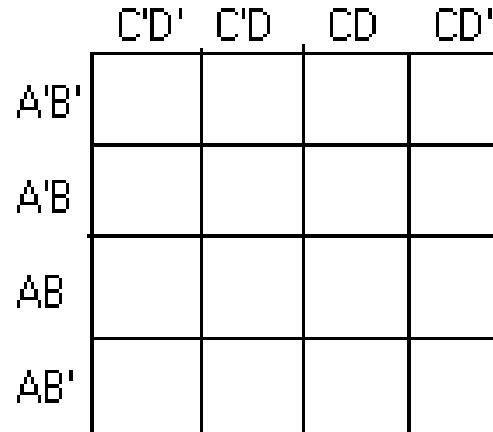
#2.- Based on the K-Map above, what is F as a sum of minterms?

- a)  $F = \sum m (0, 1, 2, 3)$
- b)  $F = \sum m (0, 1, 2, 3, 4, 5)$
- c)  $F = \sum m (0, 1, 4, 5, 8, 9)$
- d)  $F = \sum m (0, 1, 4, 5, 12, 13)$

# Sample Exercises

#3.- What is the minimal function described by this K-map?

- a)  $F = B' + C'D'$
- b)  $F = A'B' + B'CD'$
- c)  $F = AC + B'C'$
- d)  $F = B'D' + C'D'$
- e)  $F = A'B'C'D' + C'D'$



#4 Based on the K-Map above, what is  $F(A,B,C,D)$  as a sum of minterms?

- a)  $F = \sum m (0, 2, 4, 8, 10, 12)$
- b)  $F = \sum m (0, 3, 4, 9, 12, 15)$
- c)  $F = \sum m (0, 2, 4, 8, 12, 15)$
- d)  $F = \sum m (0, 1, 4, 5, 8, 9)$