جامعة الملك سعود
King Saud University

CSC 220: Computer Organization

Unit 3
Simplification of Logic Function

Prepared by:
Md Saiful Islam, PhD

Updated by:
Isra Al-Turaiki, PhD

Department of Computer Science
College of Computer and Information Sciences

Overview

- Minimal Sum of Product (MSP)
- K-Map Simplification Procedure
  - Two Variable K-Map
  - Three Variable K-Map
  - Four Variable K-Map
- Don't Care Condition
- Examples

Chapter-2

M. Morris Mano, Charles R. Kime and Tom Martin, **Logic and Computer Design Fundamentals**, Global (5th) Edition, Pearson Education Limited, 2016. ISBN: 9781292096124

Overview

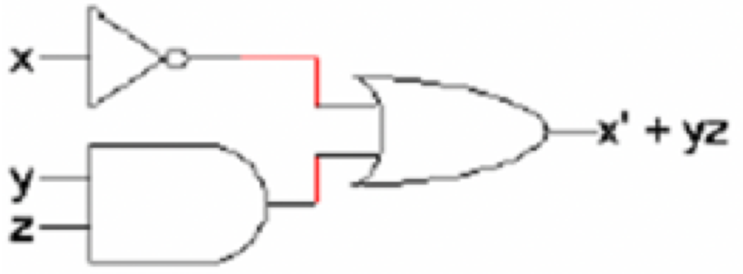
- Complexity of a logic circuit is directly related to the algebraic expression.
- To obtain the simplest implementation for a given function, use: Karnaugh map, or K-map.
- **What is a K-map?**
  - a graphical representation of a Boolean function
  - a reorganized version of the truth table
  - lead directly to minimum cost two-level circuits composed of AND and OR gates.

Minimum Sum-of-Products

Minimum sum-of-products is a sum of product terms which has:

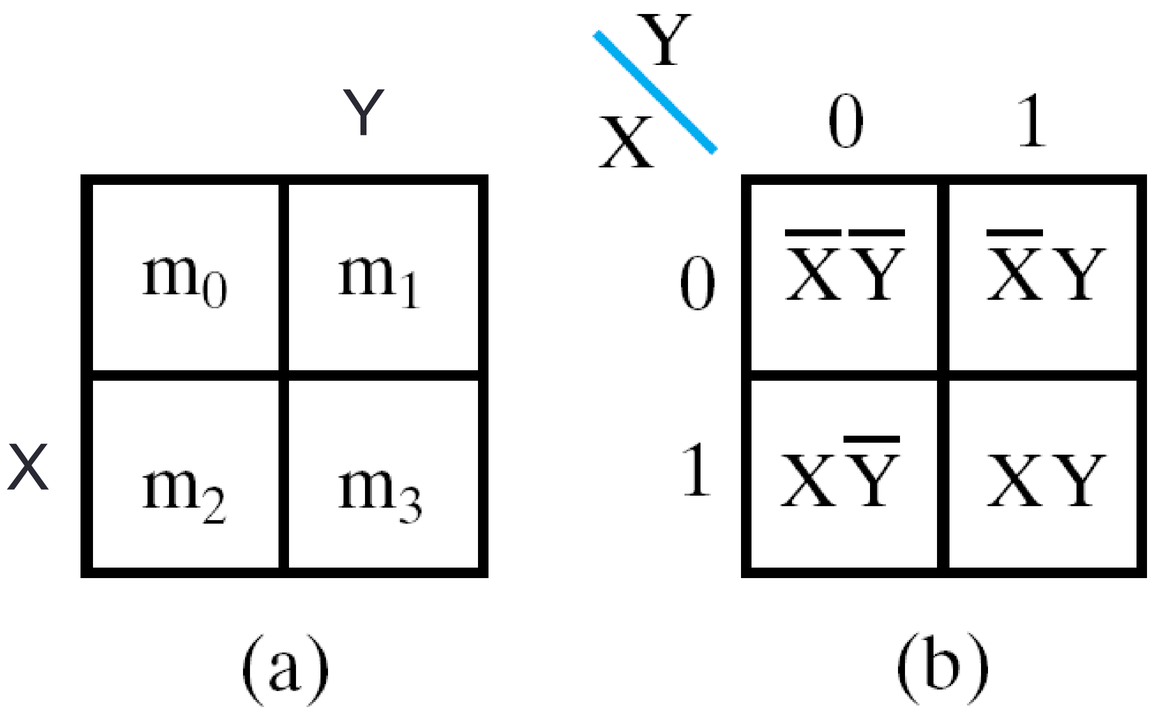
- a minimum number of terms
- a minimum number of literals
- **Why do we care?**
  - corresponds to a minimum two-level gate circuit which has (a) a minimum number of gates and (b) a minimum number of gate inputs

$$x'y'+yz+x'y = x'+yz$$



Two-Variable Maps

- The map is a diagram made up of squares, each square representing one row of a truth table, or correspondingly, one minterm of a single output function.



Every 2 *adjacent* cells differ in the value of 1 variable only.

Two-Variable Maps

- **Example:**

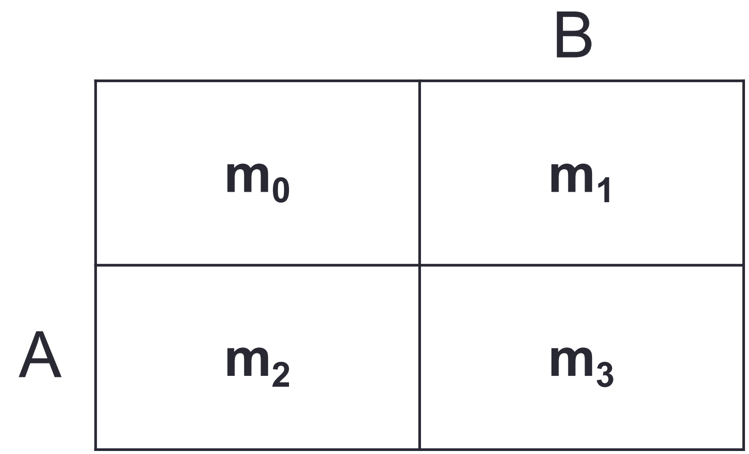
TABLE 2-12
Two-Variable Function F(A, B)

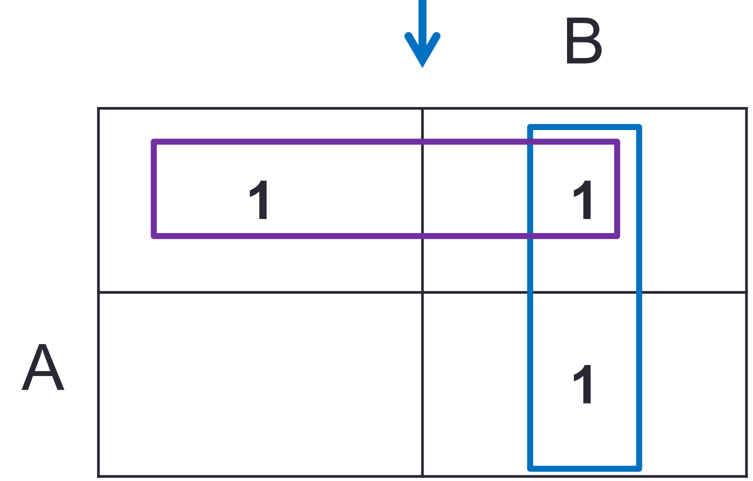
A	B	F
0	0	1
0	1	1
1	0	0
1	1	1

Algebraically:

F= A'B'+A'B+AB
=A'(B'+B)+AB
=A'+AB
=(A'+A)(A'+B)
=1. (A'+B)
=A'+B

K-Map

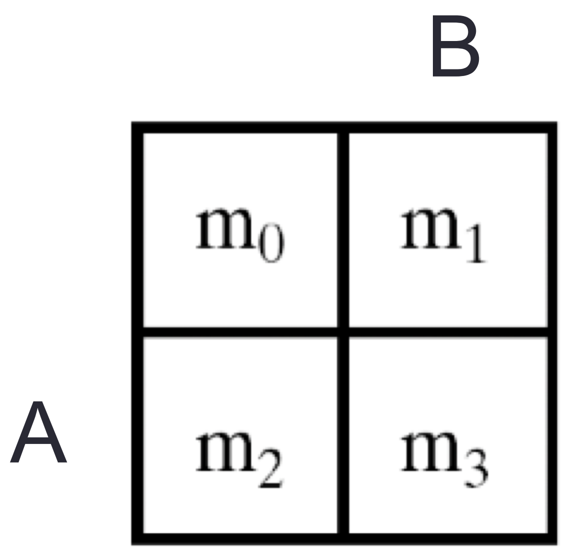


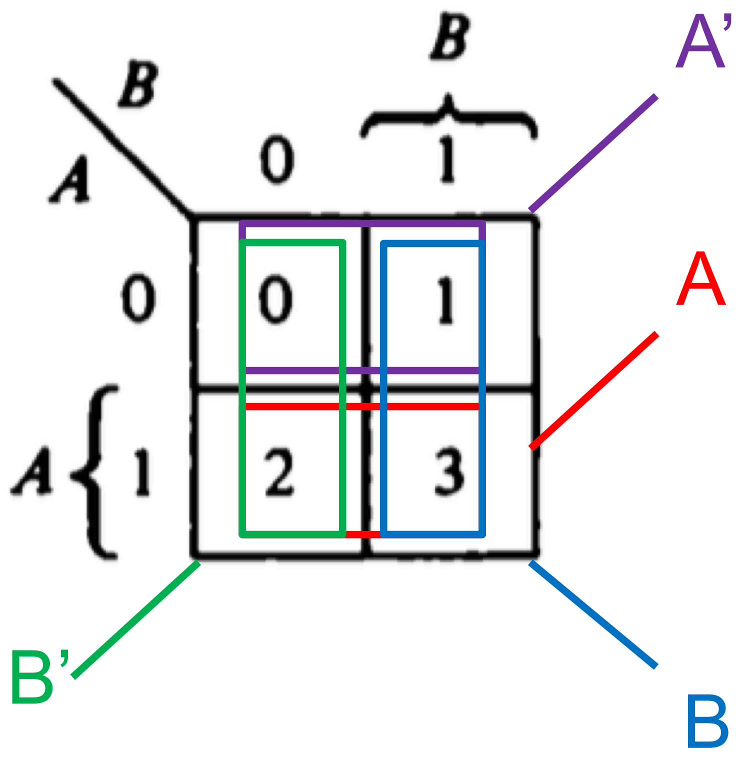


$$F = \overline{A} + B$$

Two-Variable Maps

- whenever we have two squares sharing edges that are minterms of a function, these squares can be combined to form a product term with one less variable.





Three-Variable Maps

X \ YZ	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

	X \ YZ	00	01	11 (Y)	10 (Y)
	0	$\overline{X}\,\overline{Y}\,\overline{Z}$	$\overline{X}\,\overline{Y}\,Z$	$\overline{X}\,Y\,Z$	$\overline{X}\,Y\,\overline{Z}$
X	1	$X\,\overline{Y}\,\overline{Z}$	$X\,\overline{Y}\,Z$	$X\,Y\,Z$	$X\,Y\,\overline{Z}$
			Z	Z	

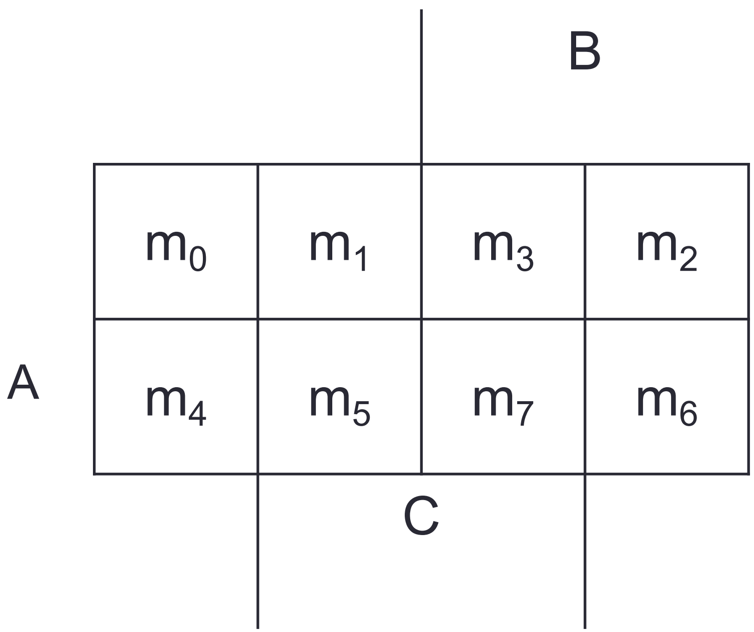
Every 2 *adjacent* cells differ in the value of 1 variable only.

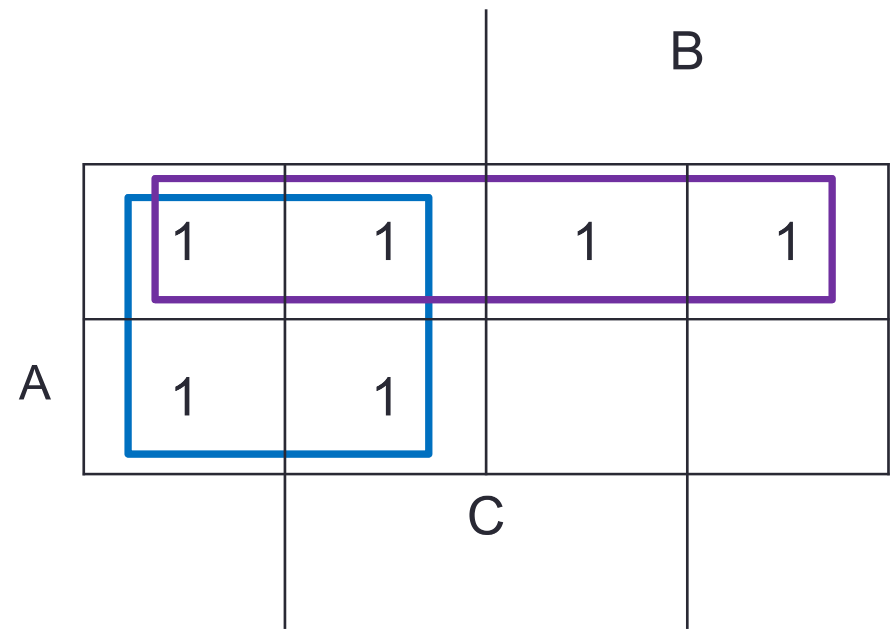
Three-Variable Maps

- **Example:**

Simplify the Boolean function $F(A, B, C) = \Sigma m(0, 1, 2, 3, 4, 5)$

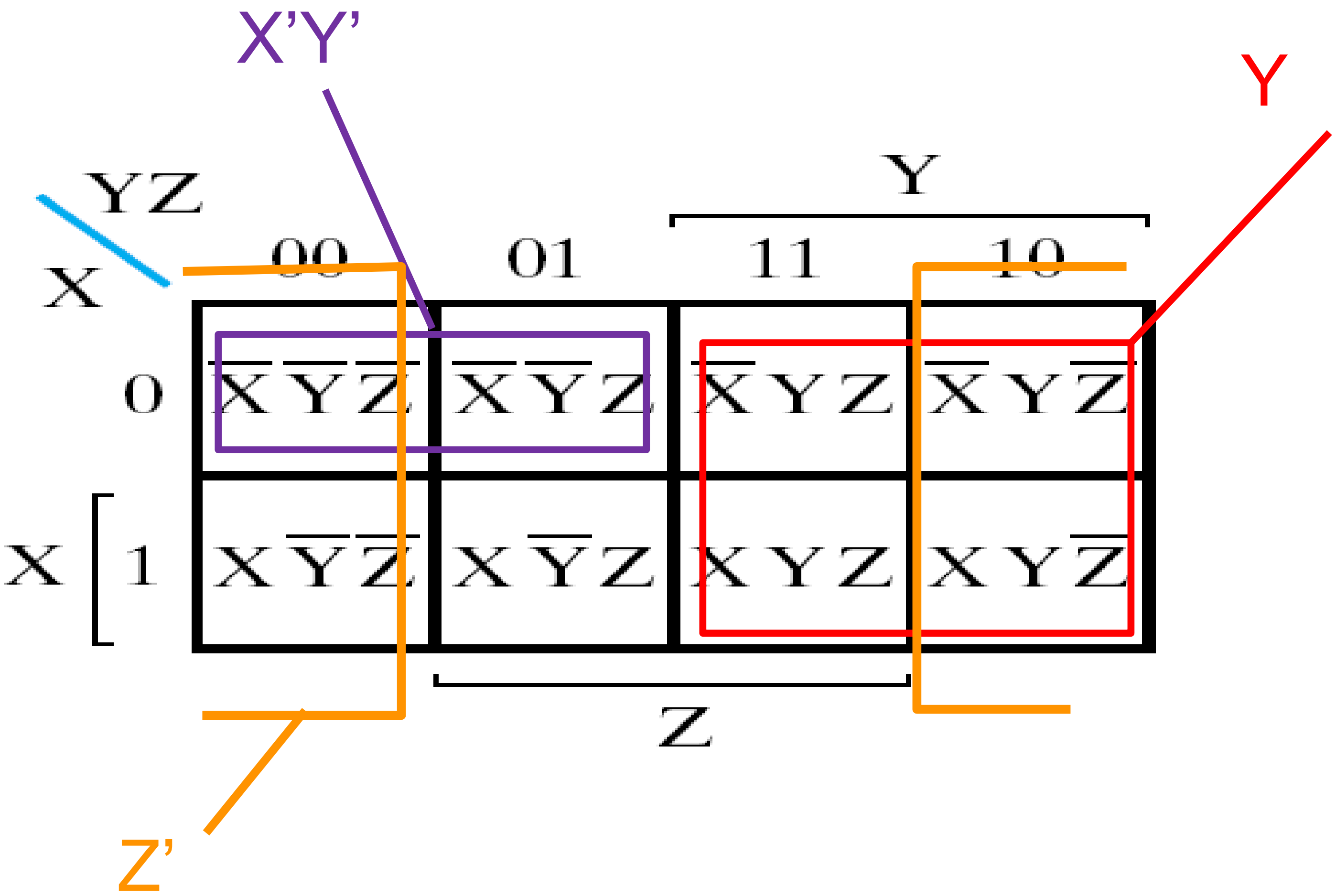
K-Map





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

Three-Variable Maps



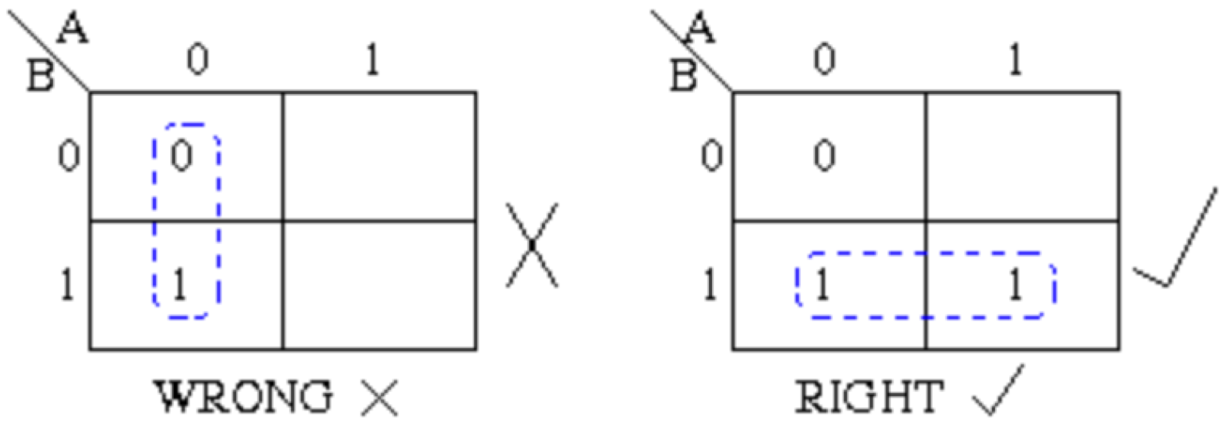
Combining Squares

Groupings to select product-terms must be:

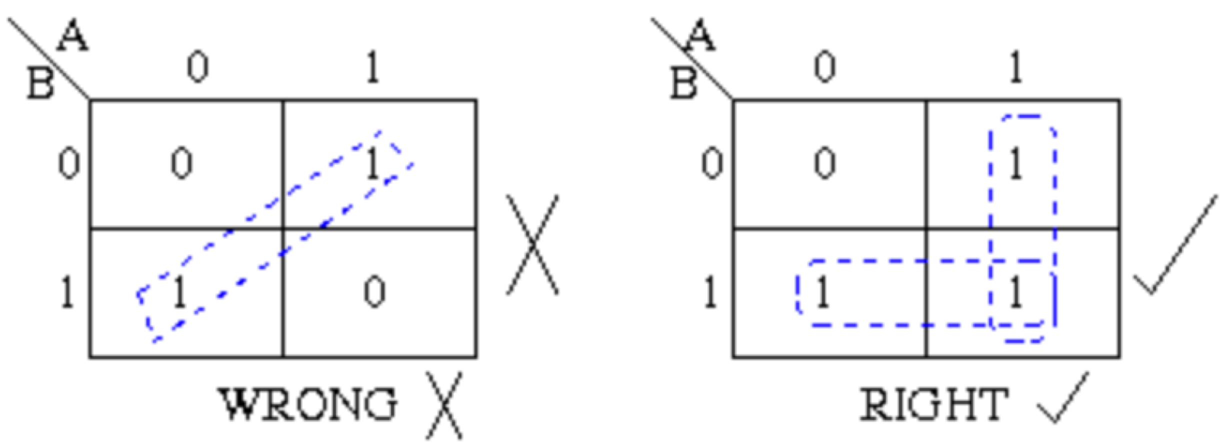
- Rectangular in shape
- In powers of twos (1, 2, 4, 8, etc.)
- Always select largest possible groupings of minterms (i.e. prime implicants)
- Eliminate redundant groupings

Combining Squares

- **Groups may not include any cell containing a zero**

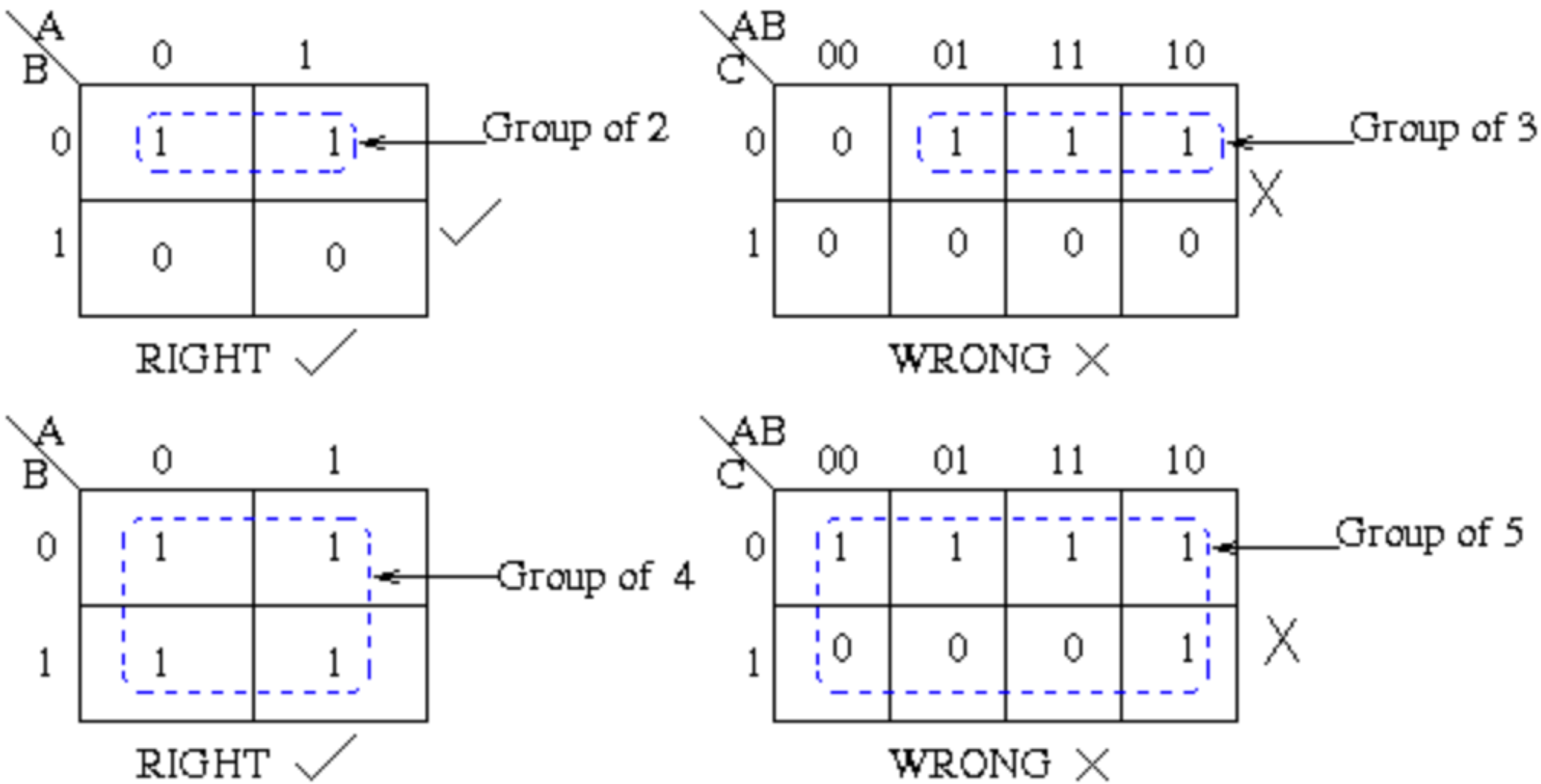


- **Groups may be horizontal or vertical, but not diagonal.**



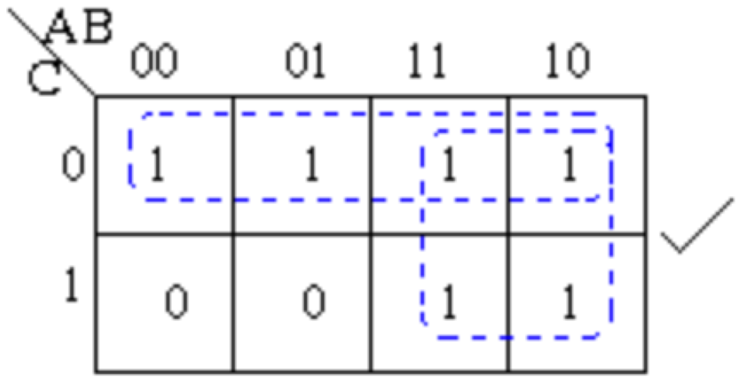
Combining Squares

- **Groups must contain 1, 2, 4, 8, or in general 2^n cells.**
  That is if n = 1, a group will contain two 1's since $2^1 = 2$.
  If n = 2, a group will contain four 1's since $2^2 = 4$.

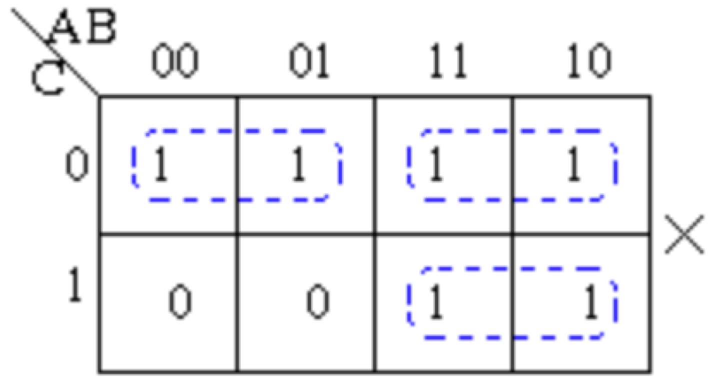


Combining Squares

- **Each group should be as large as possible.**



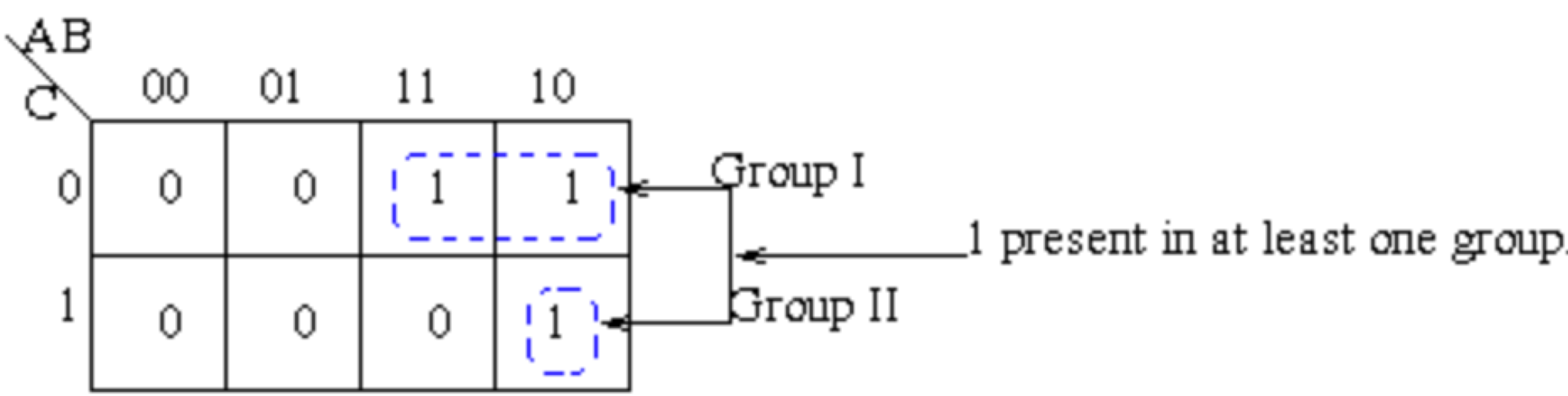
RIGHT ✓



WRONG ✕

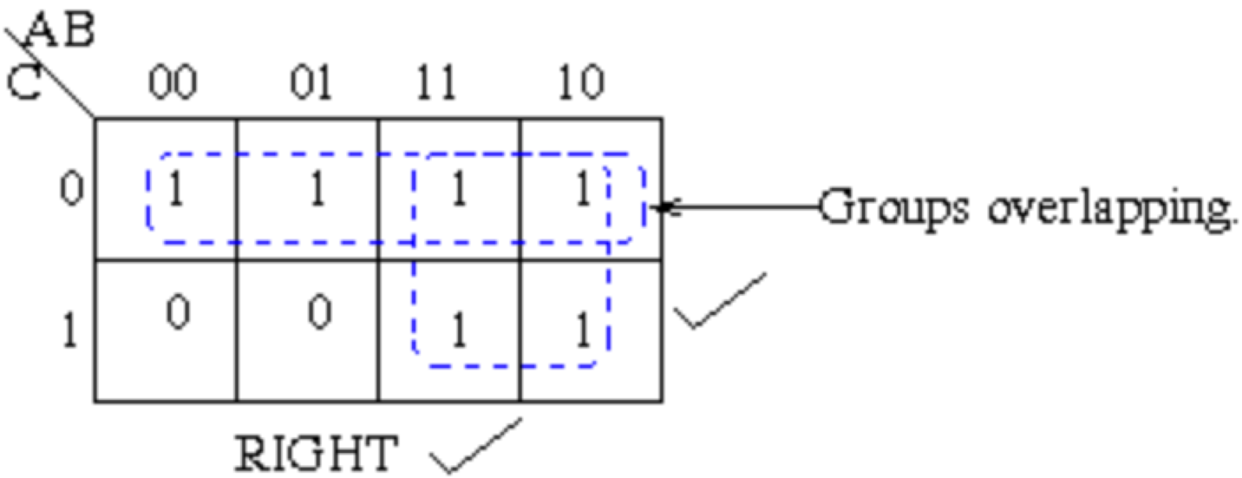
(Note that no Boolean laws broken, but not sufficiently minimal)

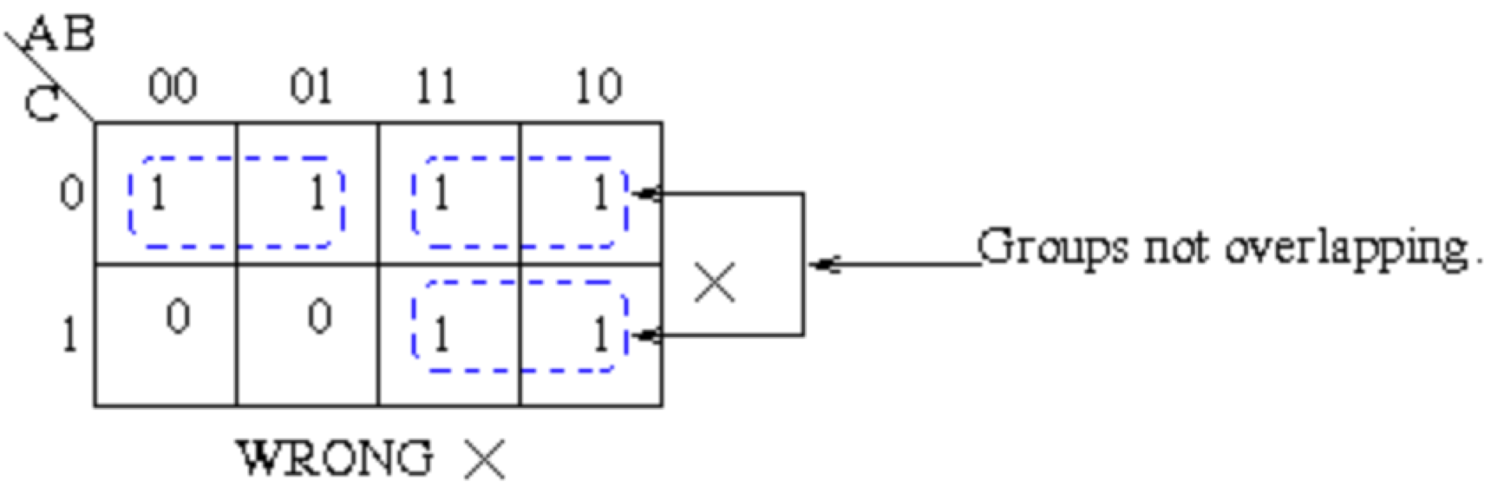
- **Each cell containing a *one* must be in at least one group.**



Combining Squares

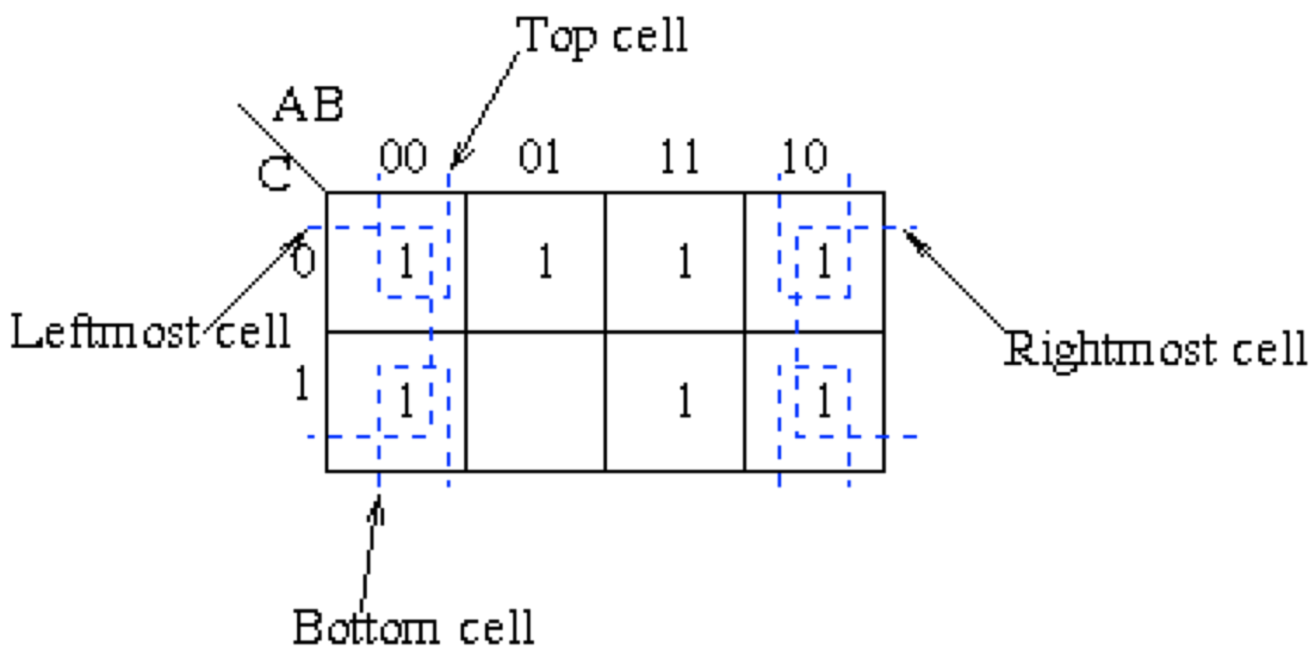
- **Groups may overlap.**





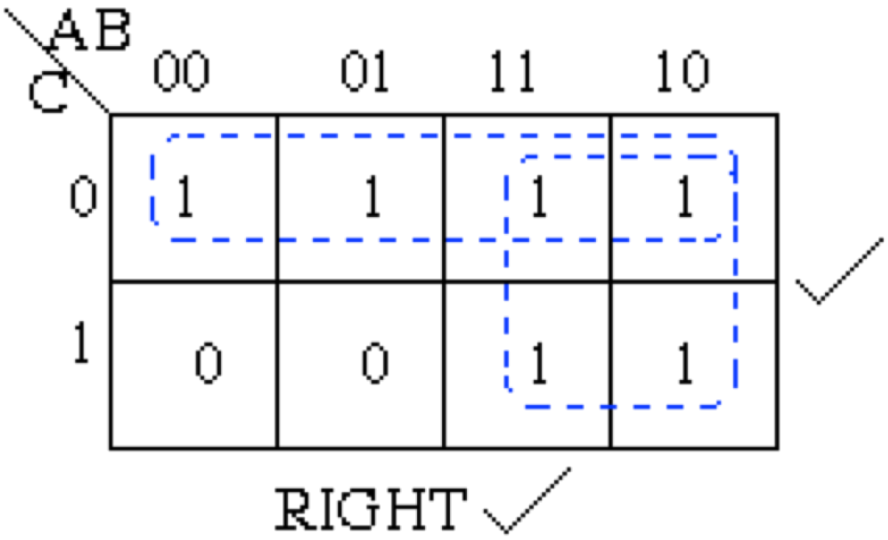
Combining Squares

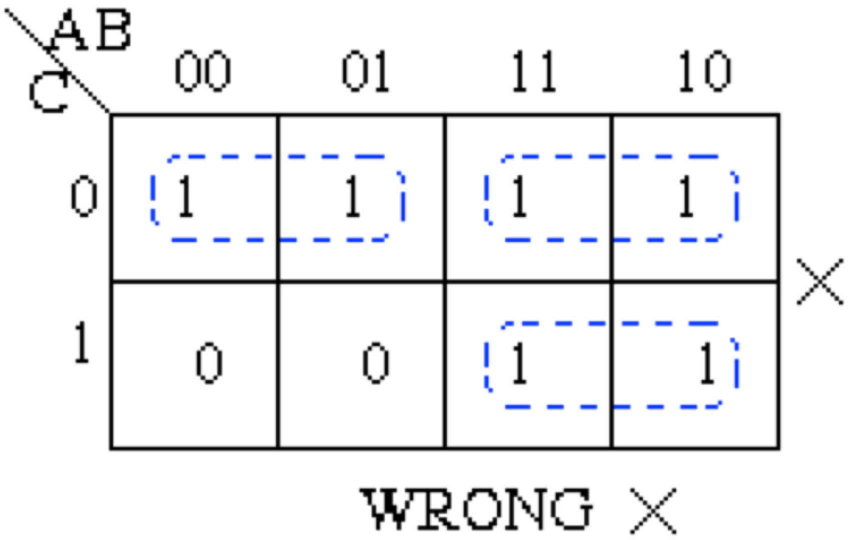
- Groups may wrap around the table. The leftmost cell in a row may be grouped with the rightmost cell and the top cell in a column may be grouped with the bottom cell.



Combining Squares

- There should be as few groups as possible, as long as this does not contradict any of the previous rules.





Simplification

Four steps in K-map simplifications

1. Start with a sum of minterms or truth table. x’y’z’ + x’y’z + xyz + xyz’

2. Plot the minterms on a Karnaugh map.

			y	y
	1	1	0	0
x	0	0	1	1
		z	z	

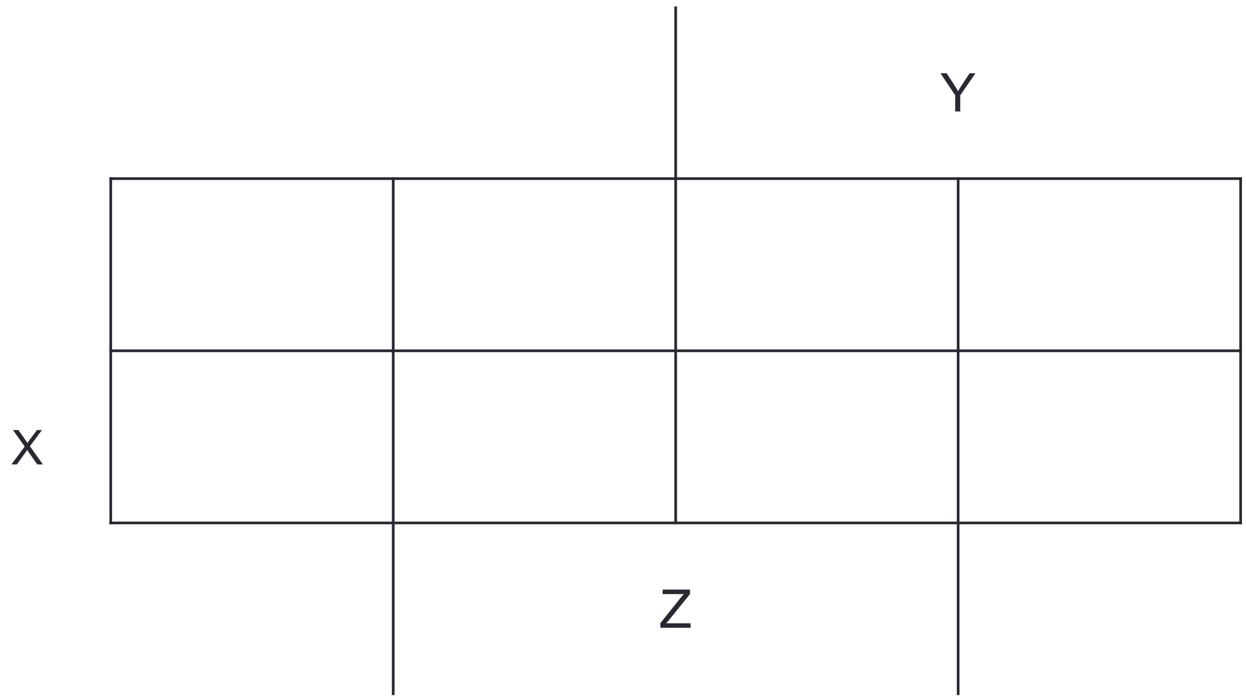
3. Find rectangular groups of minterms whose sizes are powers of two. Be sure to include all the minterms in at least one group!

			y	y
	[1	1]	0	0
x	0	0	[1	1]
		z	z	

4. Reduce each group to one product term. x’y’ + xy

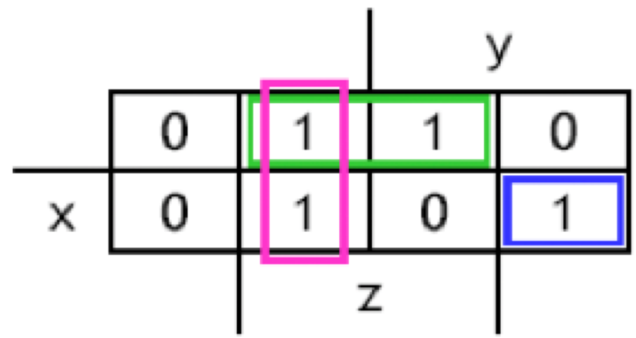
Example

- Here is the K-map for $f(x,y,z) = m_1 + m_3 + m_5 + m_6$, with all groups shown.
  - The magenta and green groups overlap, which makes each of them as large as possible.
  - Minterm m_6 is in a group all by its lonesome.



Example

- Here is the K-map for $f(x,y,z) = m_1 + m_3 + m_5 + m_6$, with all groups shown.
  - The magenta and green groups overlap, which makes each of them as large as possible.
  - Minterm m_6 is in a group all by its lonesome.



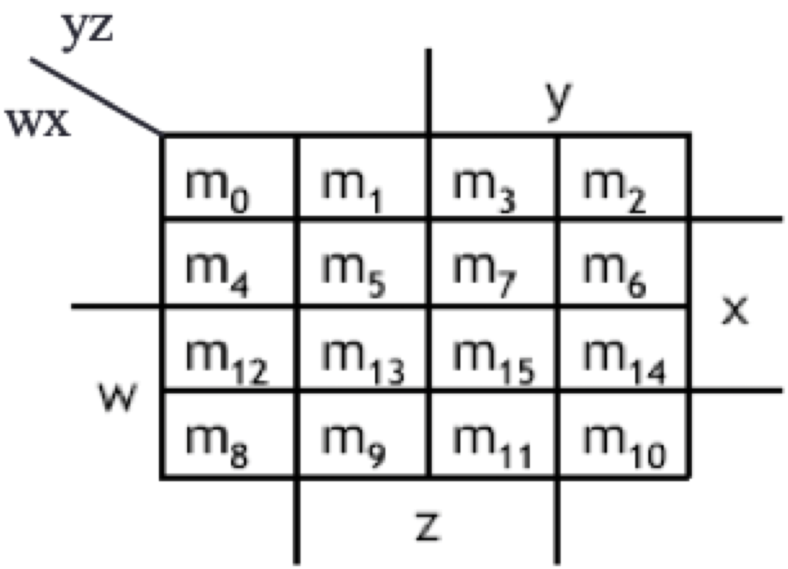
Minimal **S**um of **P**roduct

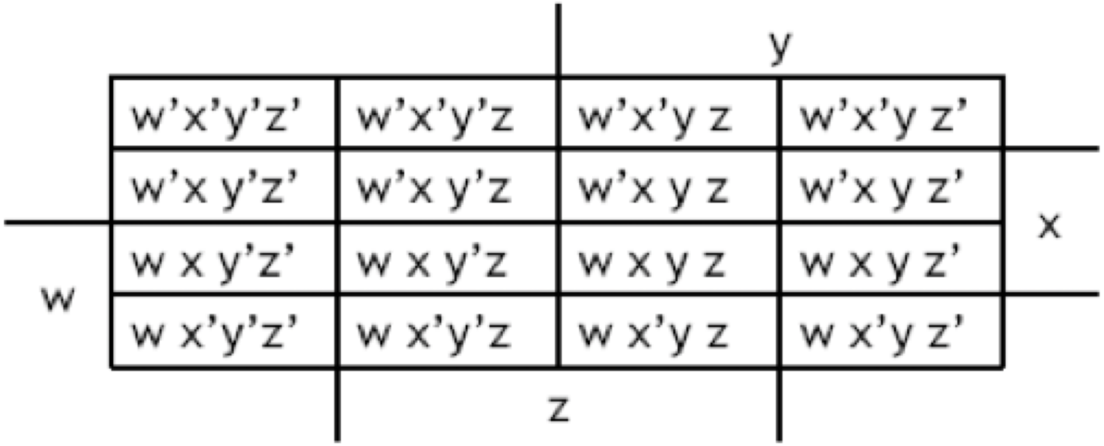
MSP

- The final MSP here is $x'z + y'z + xyz'$.

A

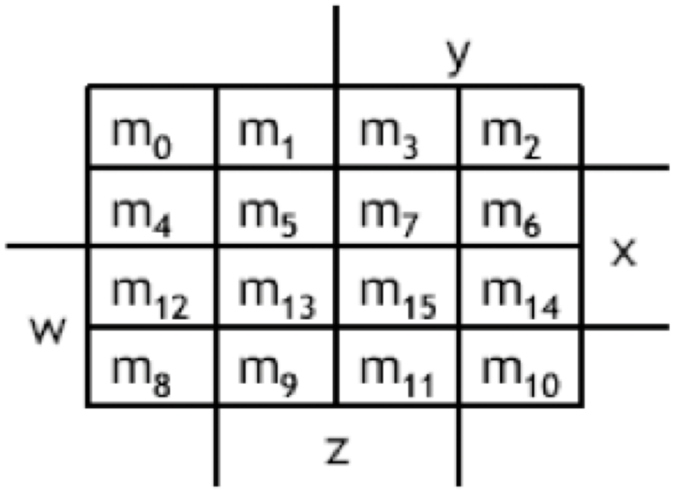
Four-Variable Maps

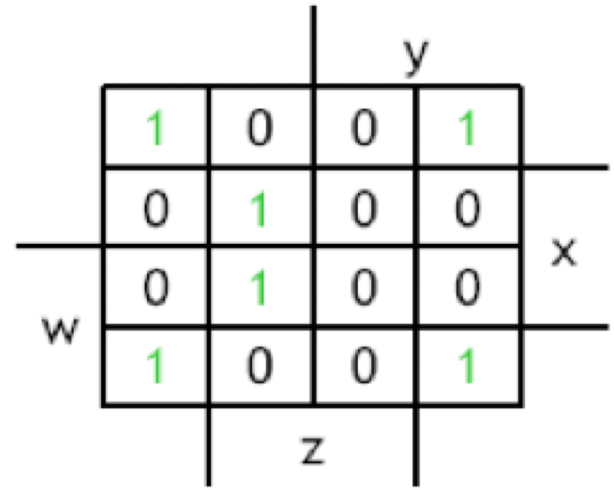




Four-Variable Maps

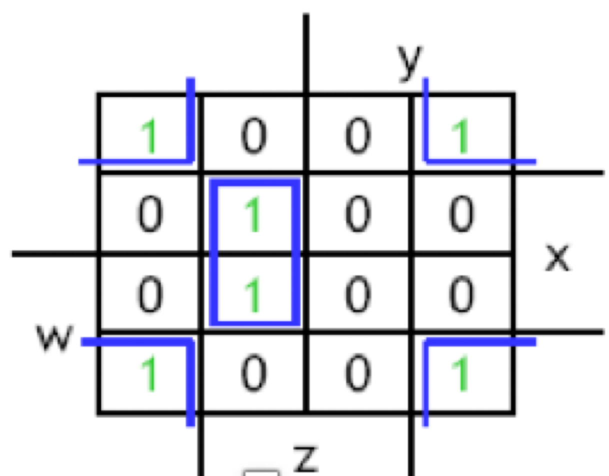
- Let's say we want to simplify $m_0 + m_2 + m_5 + m_8 + m_{10} + m_{13}$



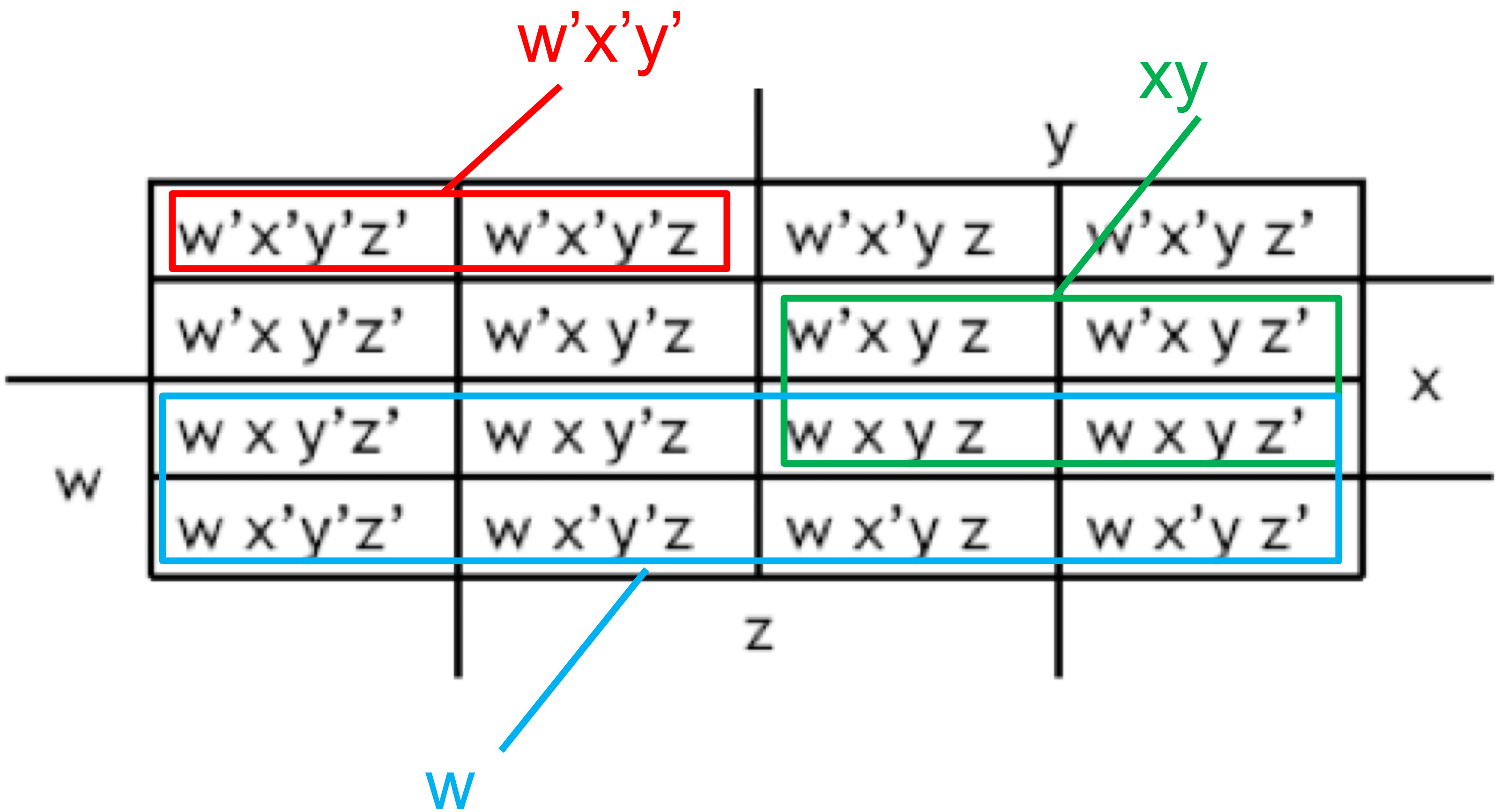


Example 6

- The following groups result in the minimal sum of products $x'z' + xy'z$.



Four-Variable Maps



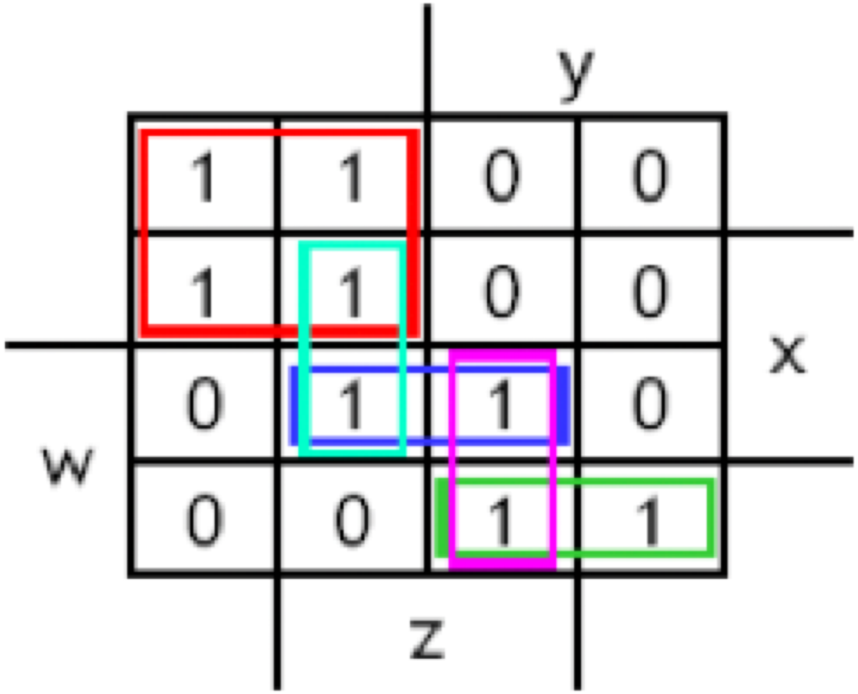
Map Manipulation

- *Implicant:* A product term that has the value 1 for all minterms of the product term.
  - all rectangles on a map made up of squares containing 1s correspond to implicants.

Map Manipulation

Prime Implicant: *is a product term obtained by combining the maximum possible number of adjacent squares in the map into a rectangle with the number of squares a power of 2.*

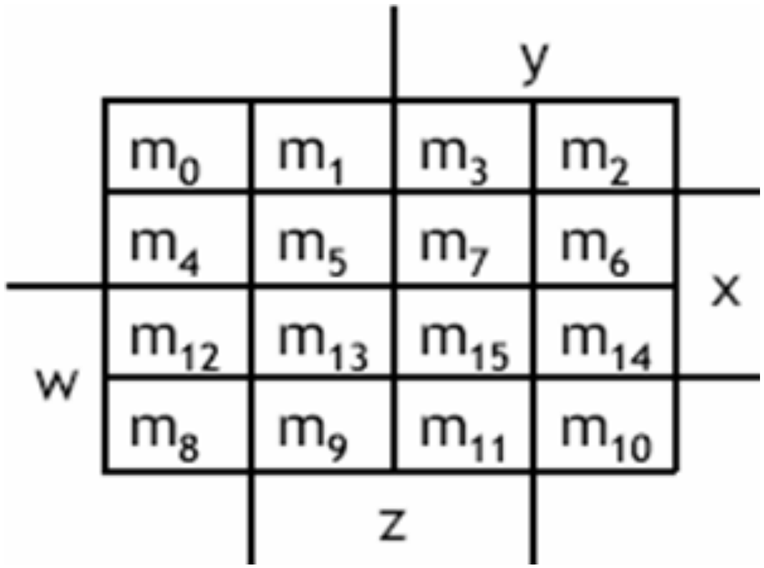
- Example: All prime implicants are marked

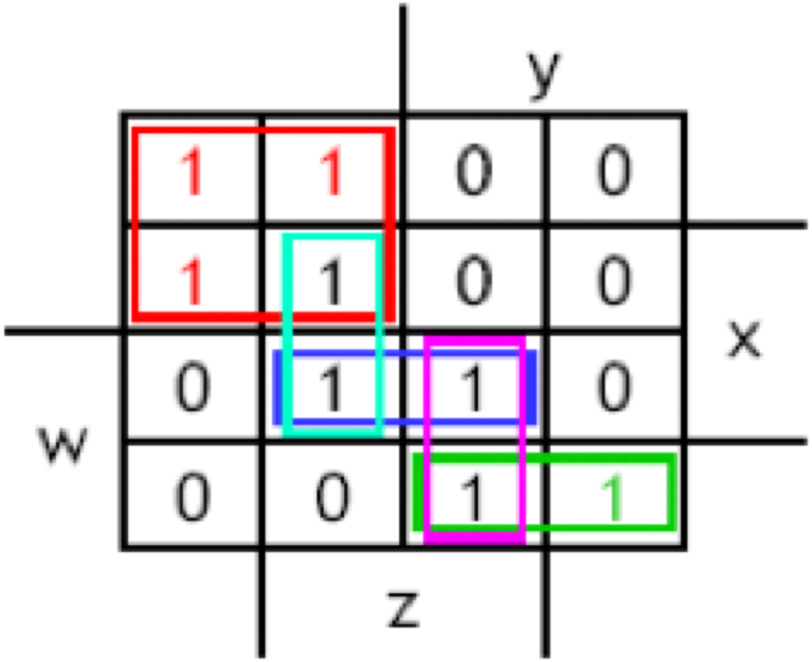


Map Manipulation

Essential Prime Implicant: *If a minterm of a function is included in only one prime implicant.*

- **Example:** red and green groups are essential





Don't Cares in K-Maps

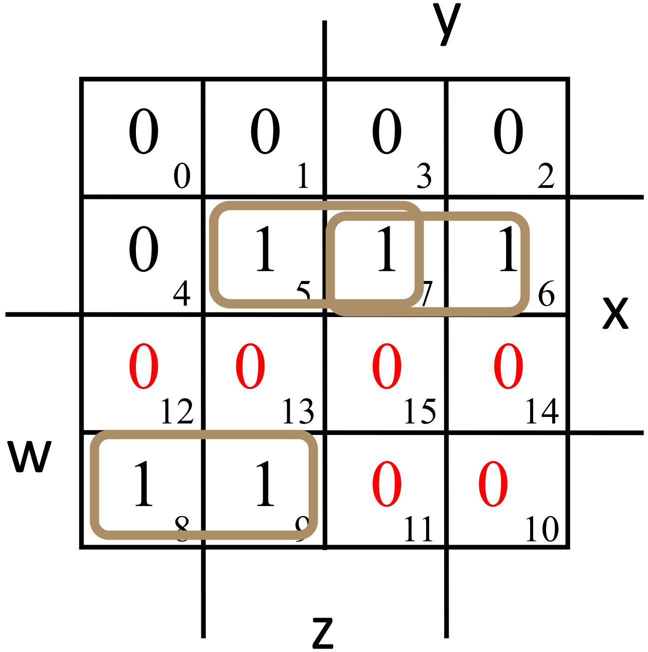
- There are times when we don't care what a function outputs—some input combinations might never occur, or some outputs may have no effect.
- We can express these situations with don't care conditions, denoted with X in truth table rows.
- An expression for this function has two parts.
  - One part includes the function's minterms.
  - Another describes the don't care conditions.

$$f(x,y,z) = m_3, \quad d(x,y,z) = m_2 + m_4 + m_5$$

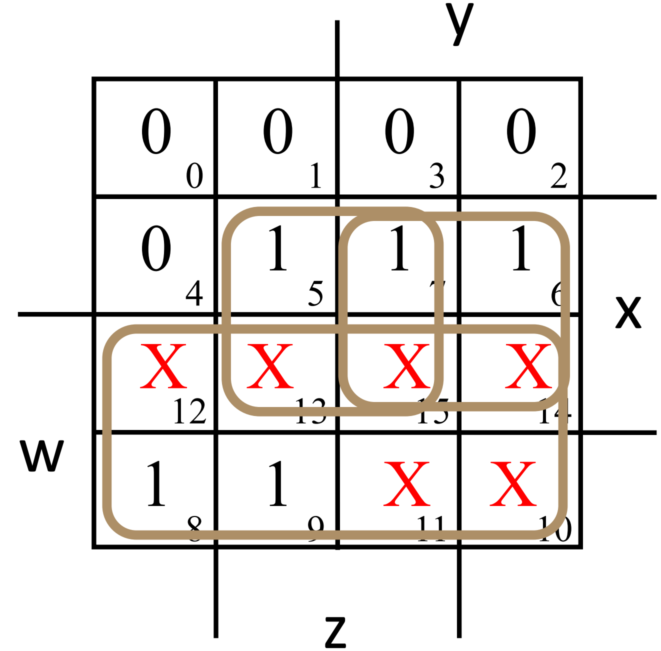
- Circuits *always* output 0 or 1; there is no value called "X". Instead, the Xs just indicate cases where both 0 or 1 would be acceptable outputs.

x	y	z	f(x,y,z)
0	0	0	0
0	0	1	0
0	1	0	X
0	1	1	1
1	0	0	X
1	0	1	X
1	1	0	0
1	1	1	0

Don't Cares in K-Maps



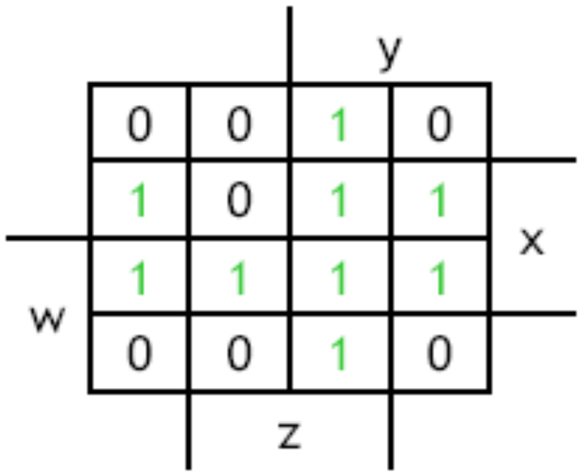
F= wx'y'+w'xz+w'xy

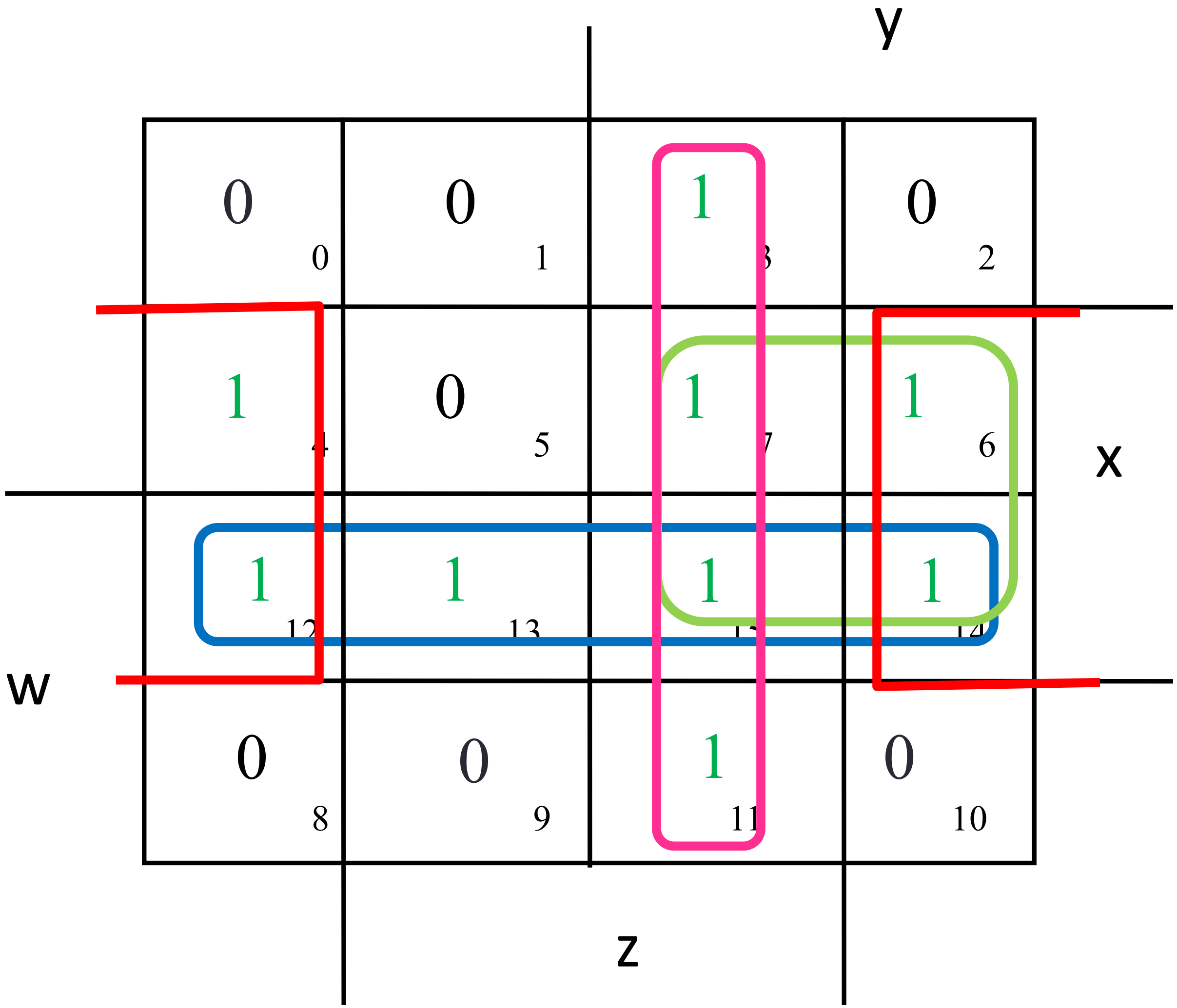


F= w+xz+xy

Practice K-map 2

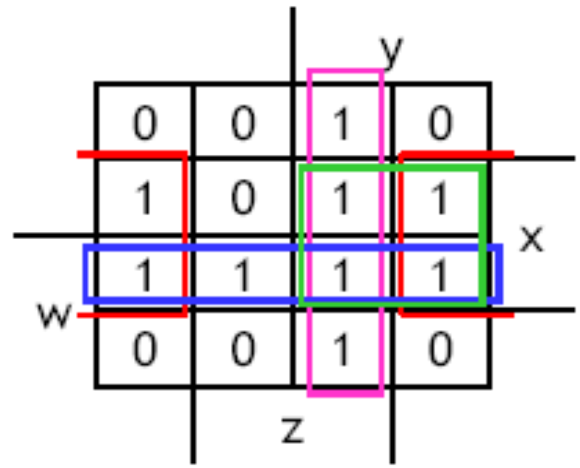
- Simplify the following K-map.




y
0 0
0 1
1
0 2
1
0 5
1
1 6
x
1
1 13
1
1
w
0 8
0 9
1 11
0 10
z

Solutions for practice K-map 2

- Simplify the following K-map.

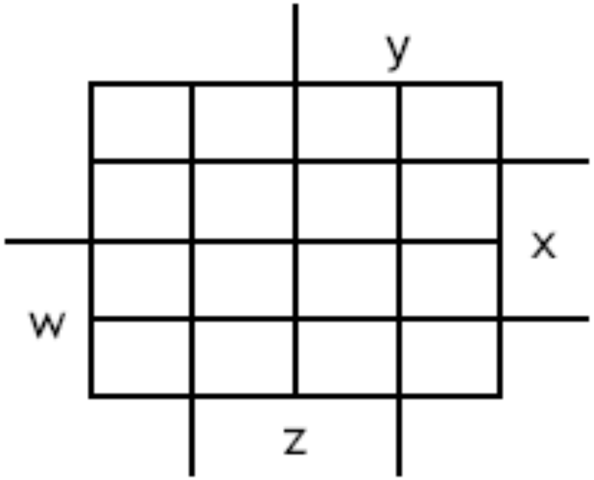


- All prime implicants are circled.
- The essential prime implicants are xz', wx and yz.
- The MSP is xz' + wx + yz. (Including the group xy would be redundant.)

Practice K-map 3

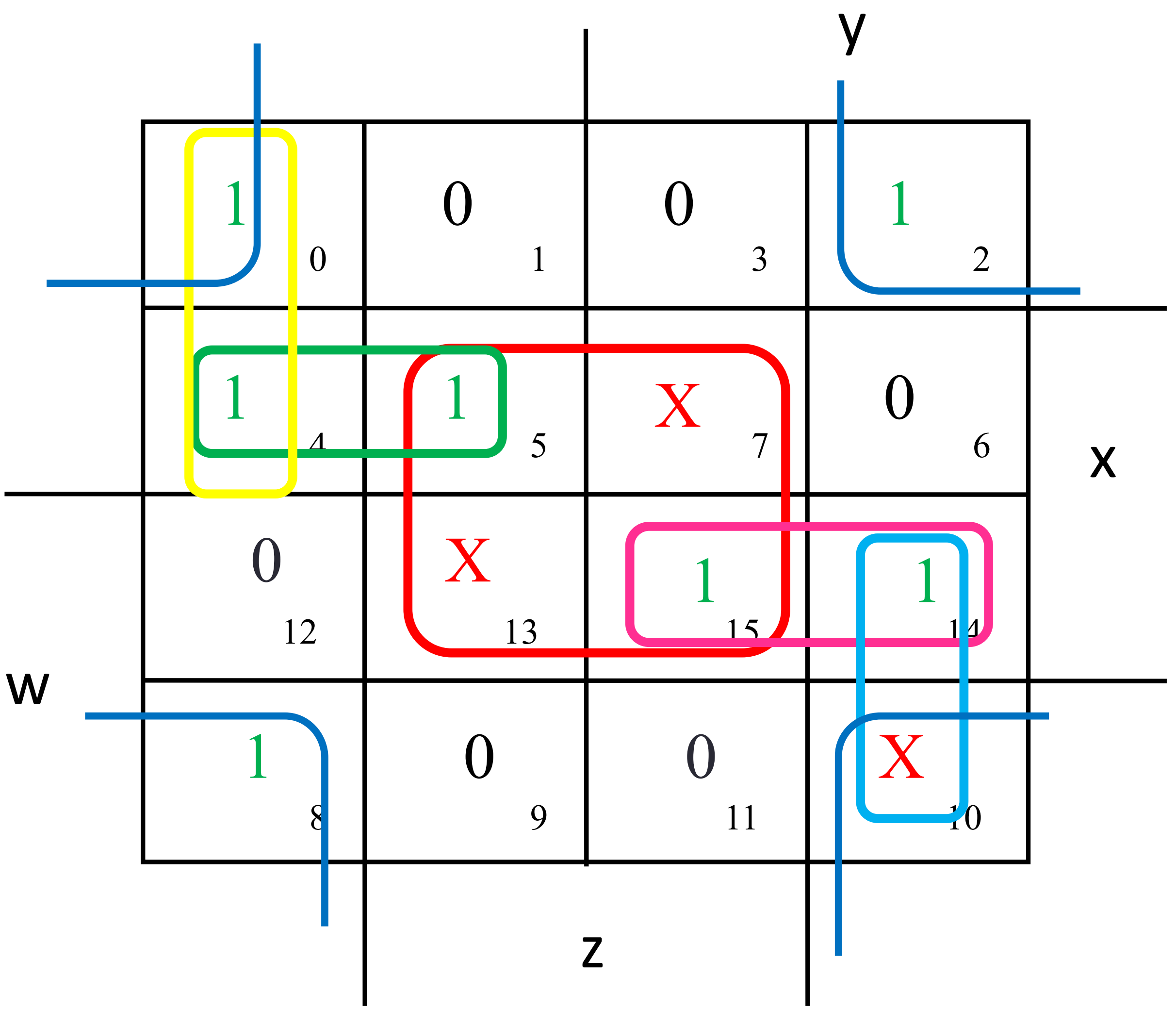
- Find a minimal sum of products for the following.

$$f(w,x,y,z) = \Sigma m(0,2,4,5,8,14,15),\ d(w,x,y,z) = \Sigma m(7,10,13)$$



- Find a minimal sum of products for the following.

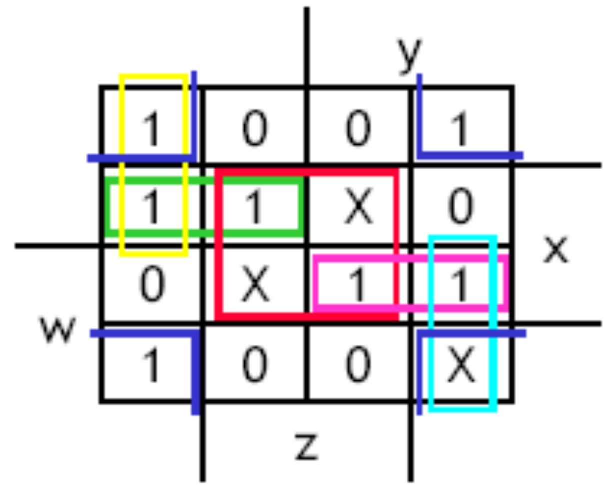
$f(w,x,y,z) = \Sigma m(0,2,4,5,8,14,15),\ d(w,x,y,z) = \Sigma m(7,10,13)$



Solutions for practice K-map 3

- Find a minimal sum of products for the following.

$f(w,x,y,z) = \Sigma m(0,2,4,5,8,14,15)$, $d(w,x,y,z) = \Sigma m(7,10,13)$



- All prime implicants are circled. We can treat Xs as 1s if we want, so the red group includes two Xs, and the light blue group includes one X.
- The *only* essential prime implicant is x'z'. The red group is not essential because the two minterms in it also appear in other groups.
- The MSP is x'z' + wxy + w'xy'. It turns out the red group is redundant; we can cover all of the minterms in the map without it.