UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SKEE1223: Digital Electronics

6 – Simplication of Logic Gates using Karnaugh Map

Dr Michael Tan Loong Peng
PhD (Cambridge)
Senior Lecturer
Faculty of Electrical Engineering
Universiti Teknologi Malaysia

Karnaugh Maps (K-Map)

- K-Map structure
  - 2,3, and 4-variable Karnaugh Maps
- K-Map Grouping and Logic Simplification
- K-Map SOP and POS terms
- K-Map Don’t Care Conditions
- Logic Design using K-Maps

Introduction

- Karnaugh Map (K-Map) is a tool for simplifying digital logic with 2-6 variables
- K-Map, if properly used will produce the simplest SOP and POS expression possible, known as the *minimum expression*
- K-Map simplifies logic through SOP and POS boolean expressions, and truth table
- In this class, we'll look at logic simplification of 2, 3, and 4 variables

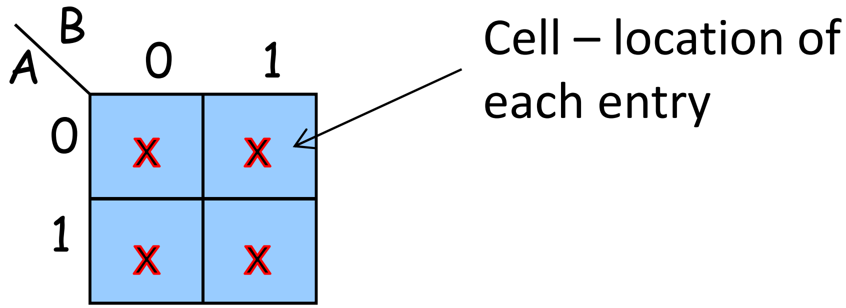
2-variable K-Maps

K-Map is a representation of a truth table, but can be used to obtain Boolean expressions

F(A,B) truth table

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

2-variable K-Map



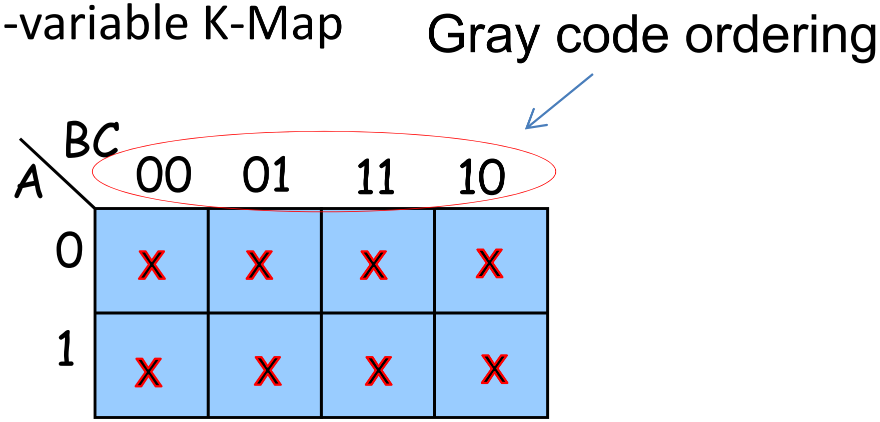
Dr Michael Tan, michael@fke.utm.my

3-variable K-Map

F(A,B,C) truth table

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

3-variable K-Map



4-variable K-Map

F(A,B,C,D) truth table

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

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

4-variable K-Map

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

K-Map Example

- Given the following standard form of SOP, complete the truth table and K-map

$$F = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + ABC$$

A B C	F	
0 0 0	0	
0 0 1	1	$\overline{A}\,\overline{B}C$
0 1 0	1	$\overline{A}B\overline{C}$
0 1 1	0	
1 0 0	0	
1 0 1	0	
1 1 0	1	$AB\overline{C}$
1 1 1	1	ABC

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

K-Map Example

- Given the following SOP expression, complete the K-Map

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

F = 1

when B = 0 and C = 0 (1st minterm)
when A = 1 and B = 0 (2nd minterm)
when A = 1, B = 1, and C = 0 (3rd minterm)
when A = 1, B = 0, C = 1, and D = 0 (4th minterm)
when A = 0, B = 0, C = 0, and D = 1 (5th minterm)
when A = 1, B = 0, C = 1, and D = 1 (6th minterm)

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

K-Map Grouping

- After SOP expression has been mapped, minimum expression is obtained by grouping the 1's and determining the minimum SOP expression from the map
- When grouping the 1's, the goal is to maximize the size of the groups, and minimize the number of groups

K-Map Grouping (cont.)

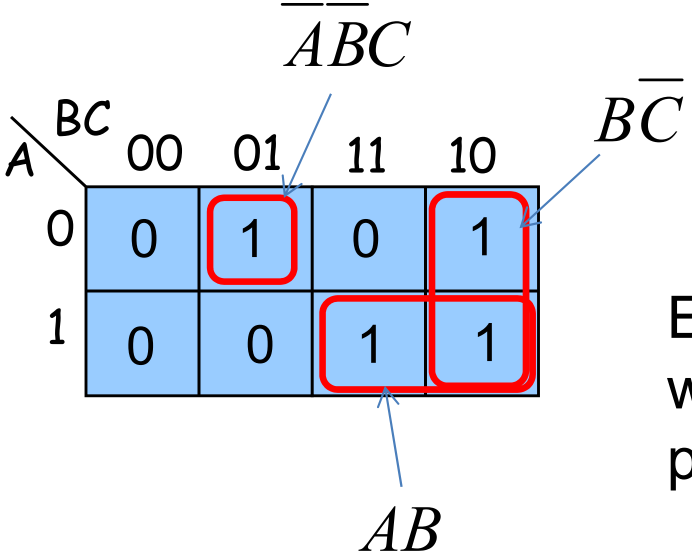
- Rules for grouping of 1's
  - A group must contain either 1, 2, 4, 8, or 16 cells. For x-variable K-map, 2^x cells is maximum
  - Each cell in a group must be adjacent to one or more cells in that same group, but all cells in the group don't have to be adjacent to each other
  - Always include the largest possible number of 1's in a group
  - Each 1 on the map must be included in at least one group. The 1's already in a group can be included in another group as long as the overlapping groups include common 1's

K-Map Minimum Product Term

- For 3-variable K-Map
  - 1 cell group yields a 3-variable product term
  - 2 cell group yields a 2-variable product term
  - 4 cell group yields a 1-variable product term
  - 8 cell group yields a value of 1 for the expression
- For 4-variable K-Map
  - 1 cell group yields a 4-variable product term
  - 2 cell group yields a 3-variable product term
  - 4 cell group yields a 2-variable product term
  - 8 cell group yields a 1-variable product term
  - 16- cell group yields a value of 1 for the expression

K-Map Simplification

- Group the 1's and find the minimum SOP expression in the K-Map below



Expression is minimized when taking large cell possible

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

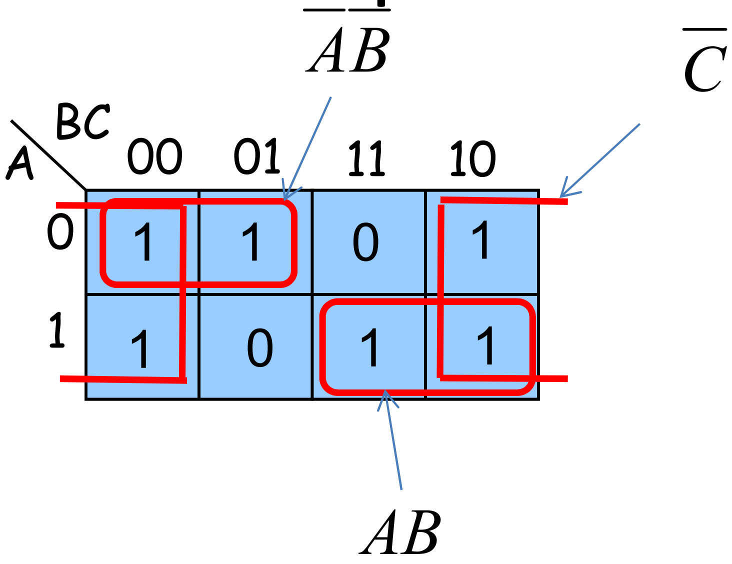
What is the SOP expression if each cell is taken as a group?

$$F = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + ABC$$

Dr Michael Tan, michael@fke.utm.my

K-Map Simplification

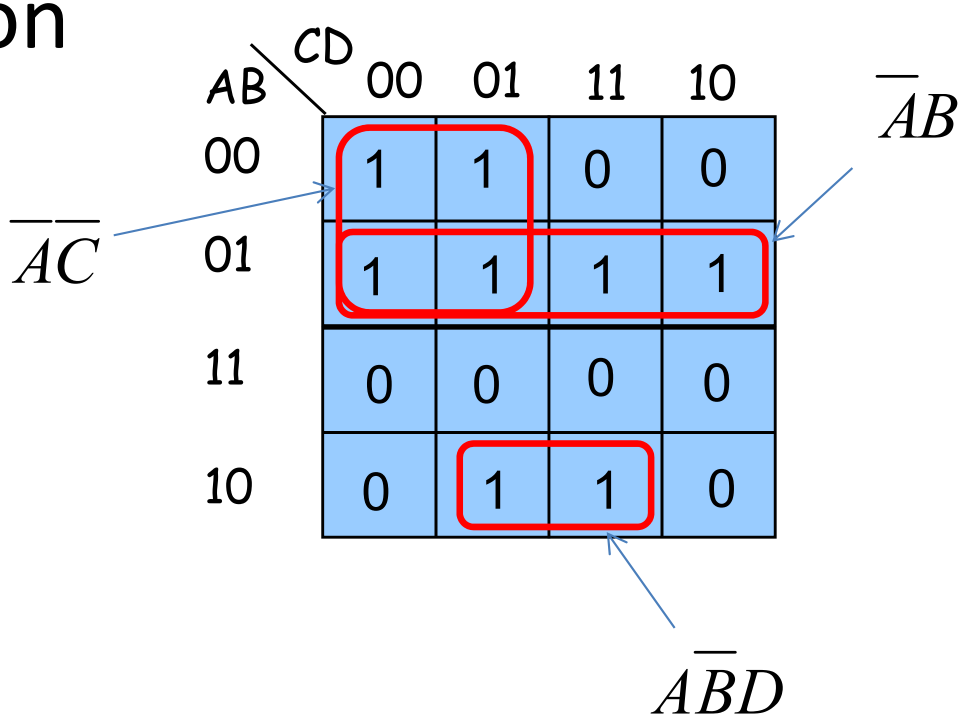
- Group the 1's and find the minimum SOP expression in the K-Map below



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

K-Map Simplification

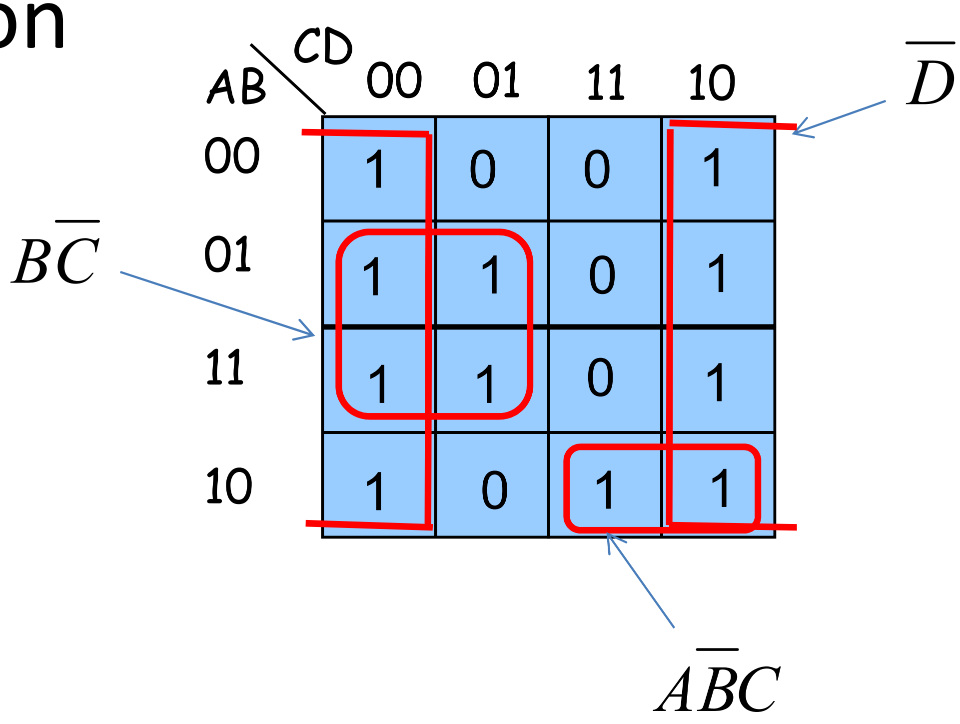
- Group the 1's and find the minimum SOP expression



$$F = \overline{A}B + \overline{A}\,\overline{C} + A\overline{B}D$$

K-Map Simplification

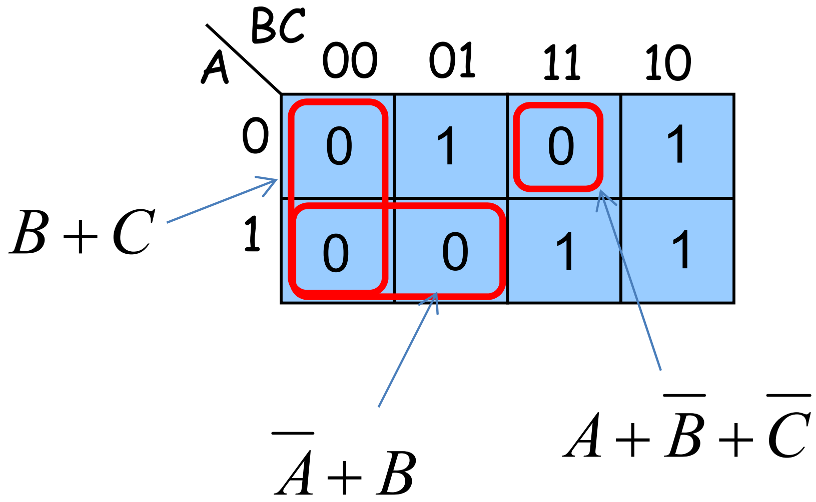
- Group the 1’s and find the minimum SOP expression



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

K-Map POS

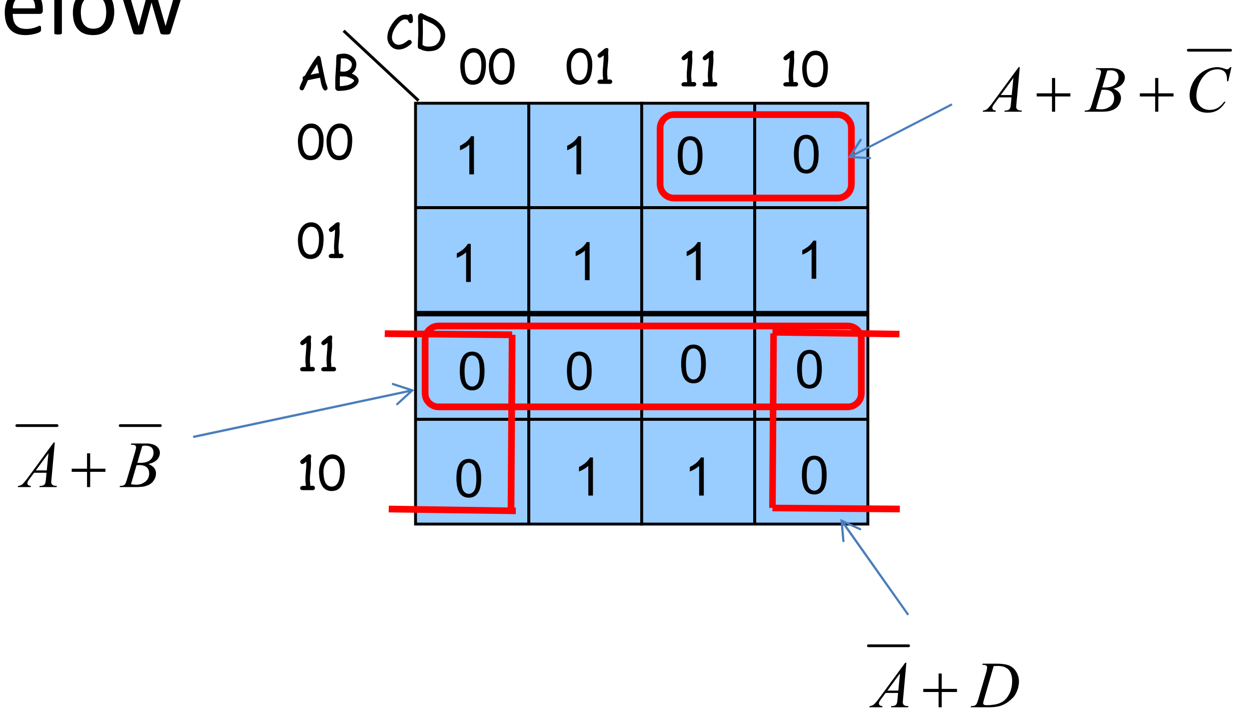
- K-Maps can also be used to obtain POS expressions by grouping the 0's



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

K-Map POS

- Find the minimum POS expression from the K-Map below



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

Don’t Care Conditions

- Don’t Care is the condition when the output can either be ‘1’ or ‘0,’ which is denoted by ‘x’ in the truth table or K-Map
- For both SOP and POS minimum expression, ‘x’ can be included or ignored

Don't Care Condition (cont.)

- Find minimum SOP expression for the following K-Map

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

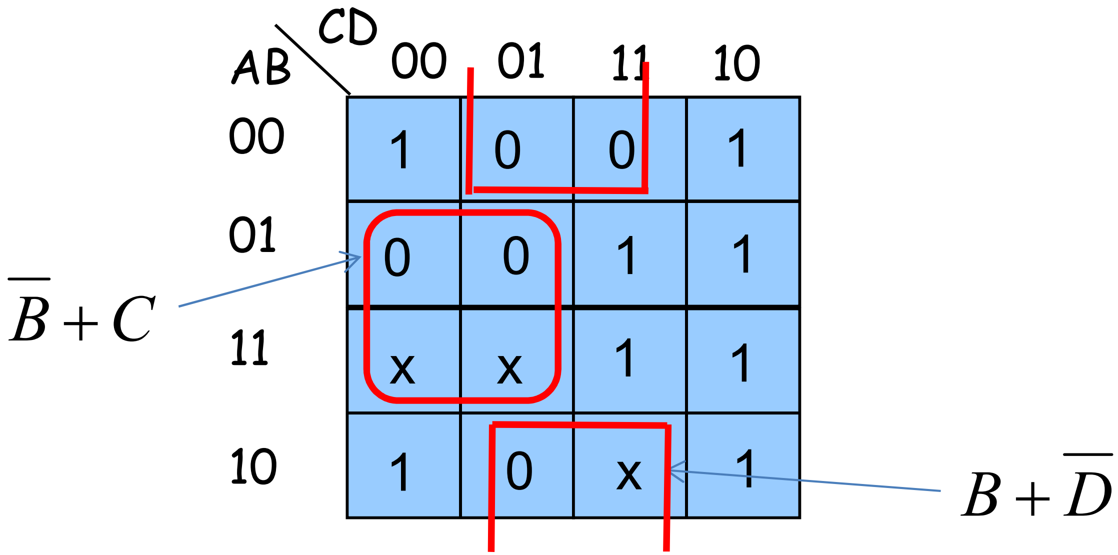
$$F = \overline{A}D + AB$$

If the 'x' is replaced by '0,' find the minimum SOP expression

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

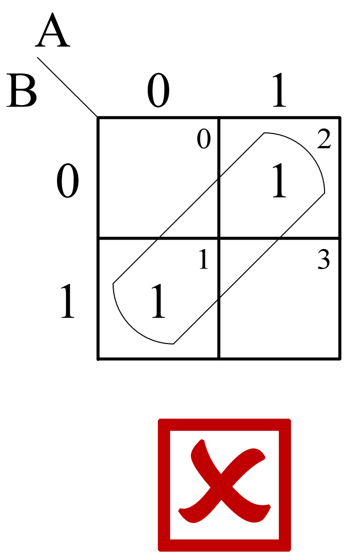
Don’t Care Condition (cont.)

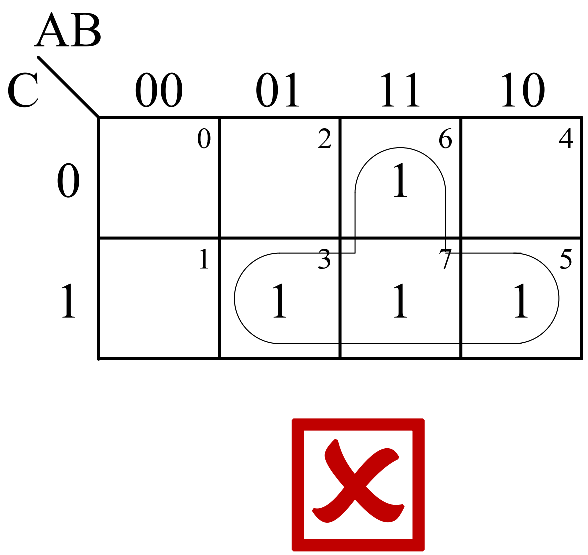
- Find minimum POS expression for the following K-Map

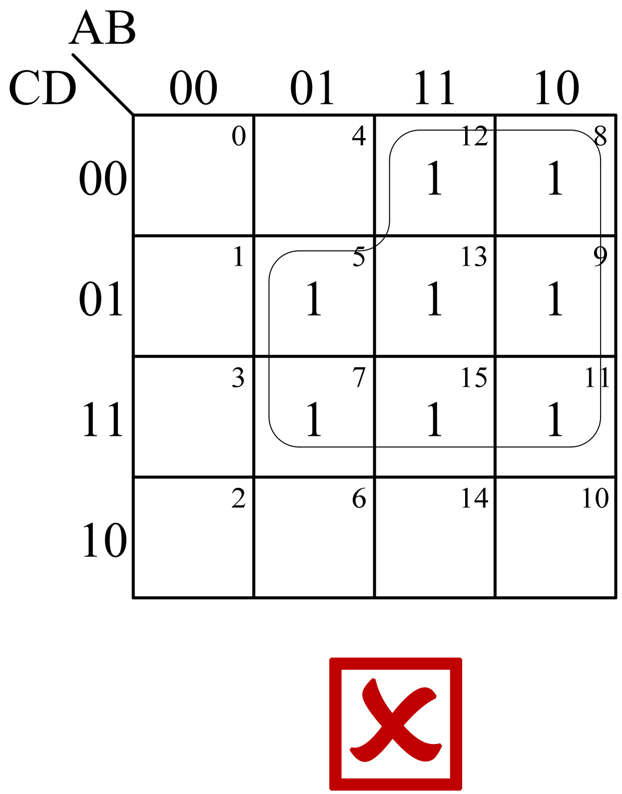


$$F = (\overline{B}+C)(B+\overline{D})$$

Wrong ways of looping!







Dr Michael Tan, michael@fke.utm.my

Examples : Looping the 1s (SOP form)

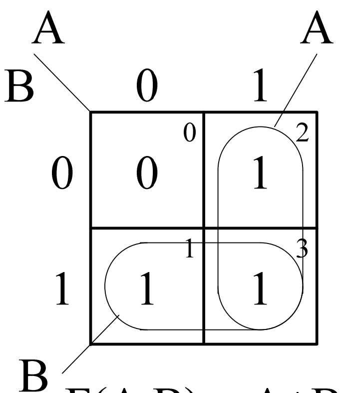
a. $F(A,B) = \Sigma m(1,2,3)$

b. $F(A,B,C) = \Sigma m(2,6)$
c. $F(A,B,C) = \Sigma m(1,3,5,7)$
d. $F(A,B,C) = \Sigma m(1,2,5,6)$

e. $F(A,B,C) = \Sigma m(0,2,3,7)$
f. $F(A,B,C) = \Sigma m(0,1,2,4,5)$

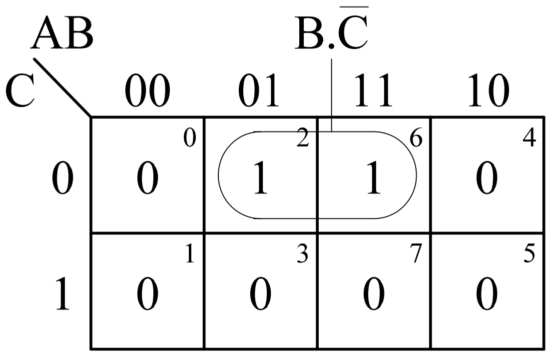
g. $F(A,B,C,D) = \Sigma m(3,4,5,7,11,12,13)$
h. $F(A,B,C,D) = \Sigma m(0,1,2,8,9,10)$

Examples : Looping the 1s (SOP form)



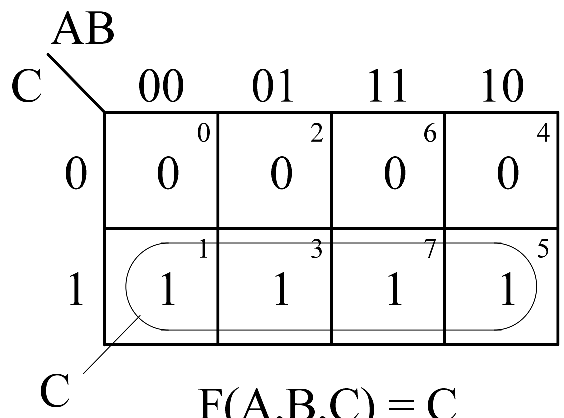
$F(A,B) = A+B$

(i)



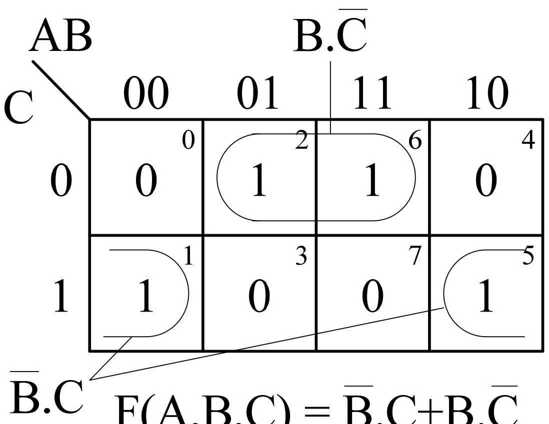
$F(A,B,C) = B.\overline{C}$

(ii)



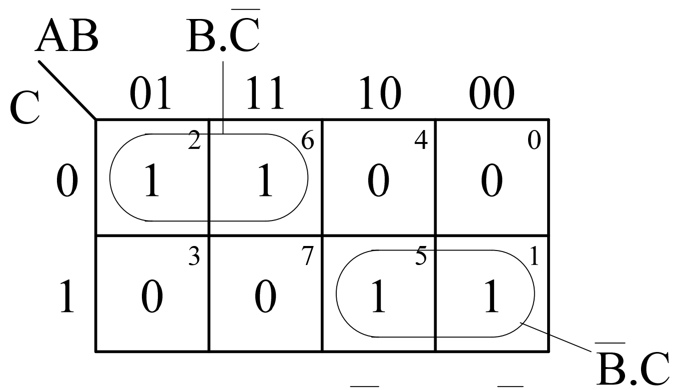
$F(A,B,C) = C$

(iii)



$F(A,B,C) = \overline{B}.C+B.\overline{C}$

(iv)

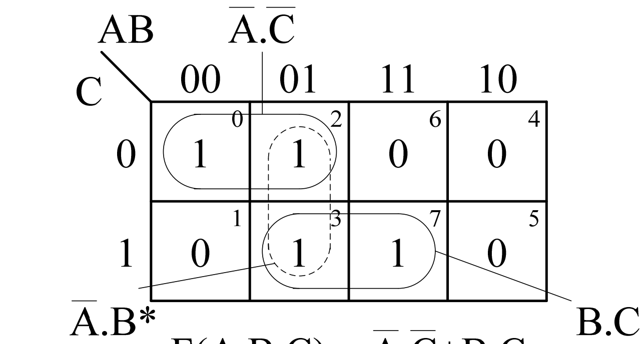


$F(A,B,C) = \overline{B}.C+B.\overline{C}$

(v)

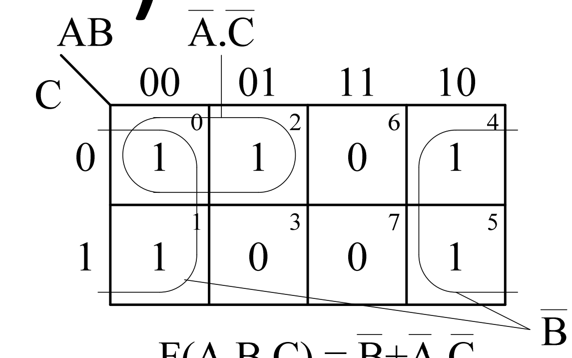
Dr Michael Tan, michael@fke.utm.my

Examples : Looping the 1s (SOP form)



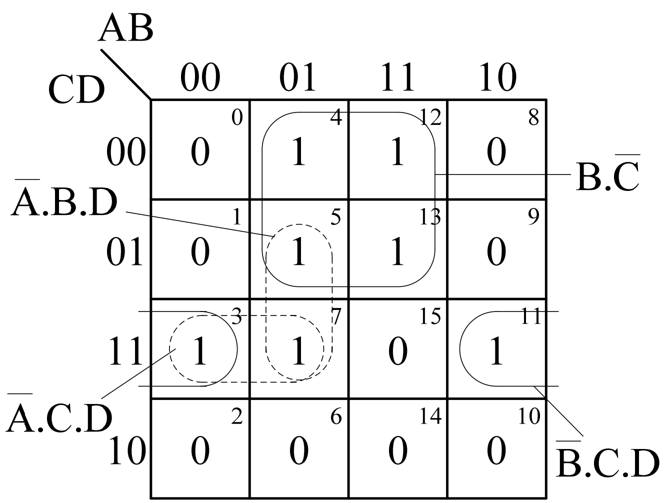
$F(A,B,C) = \overline{A}.\overline{C}+B.C$

(vi)



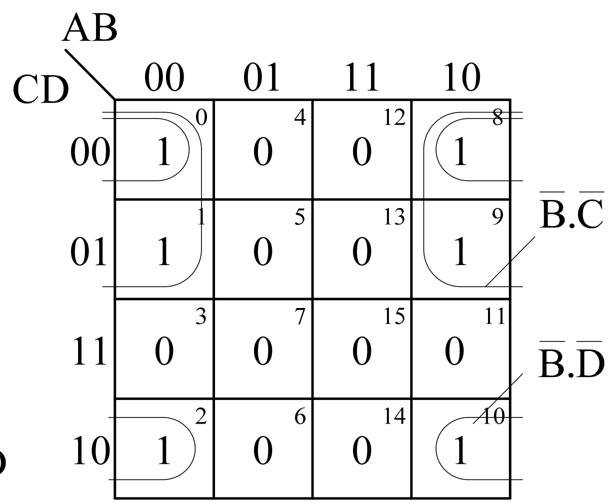
$F(A,B,C) = \overline{B}+\overline{A}.\overline{C}$

(vii)



$F(A,B,C,D) = B.\overline{C}+\overline{B}.C.D+\overline{A}.C.D$
$= B.\overline{C}+\overline{B}.C.D+\overline{A}.B.D$

(viii)



$F(A,B,C,D) = \overline{B}.\overline{C}+\overline{B}.\overline{D}$

(ix)

Examples : Looping the 0s (POS form)

a. F(A,B) $= \prod M(0)$

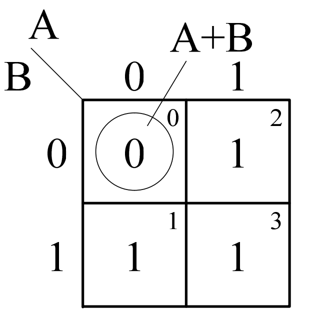
b. F(A,B,C) $= \prod M(1,4,5,6)$

c. F(A,B,C) $= \prod M(3,6,7)$

d. F(A,B,C,D) $= \prod M(0,1,2,6,8,9,10,14,15)$

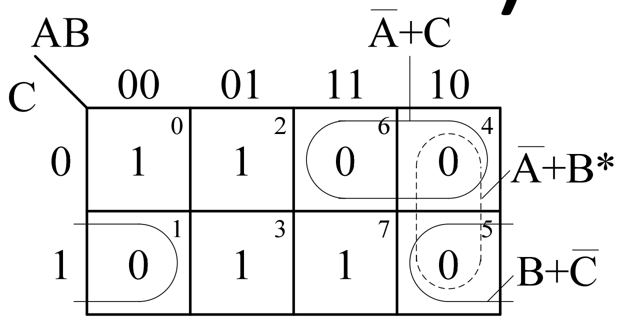
e. F(A,B,C,D) $= \prod M(3,4,5,6,7,11,12,13,14,15)$

Examples : Looping the 0s (POS form)



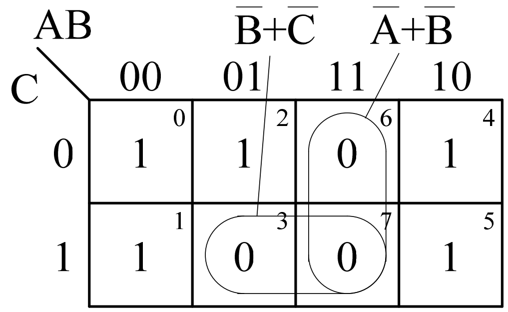
F(A,B) = A+B

(i)



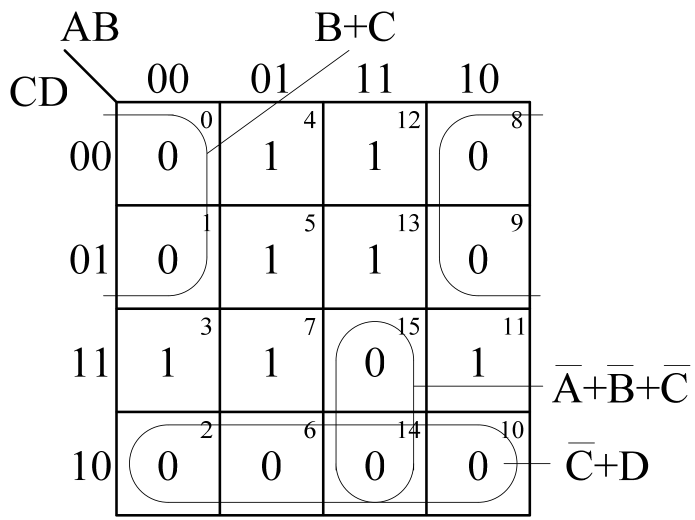
F(A,B,C) = ($\overline{A}$+C).(B+$\overline{C}$)

(ii)



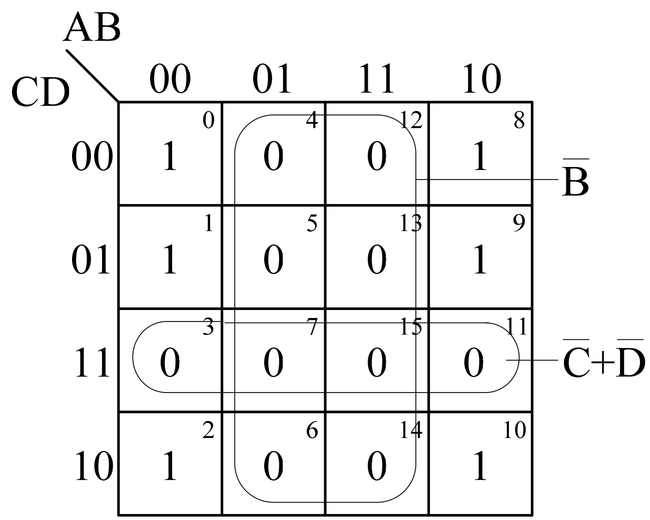
F(A,B,C) = ($\overline{A}$+$\overline{B}$).($\overline{B}$+$\overline{C}$)

(iii)



F(A,B,C,D) = (B+C).($\overline{C}$+D).($\overline{A}$+$\overline{B}$+$\overline{C}$)

(iv)



F(A,B,C,D) = $\overline{B}$.($\overline{C}$+$\overline{D}$)

(v)

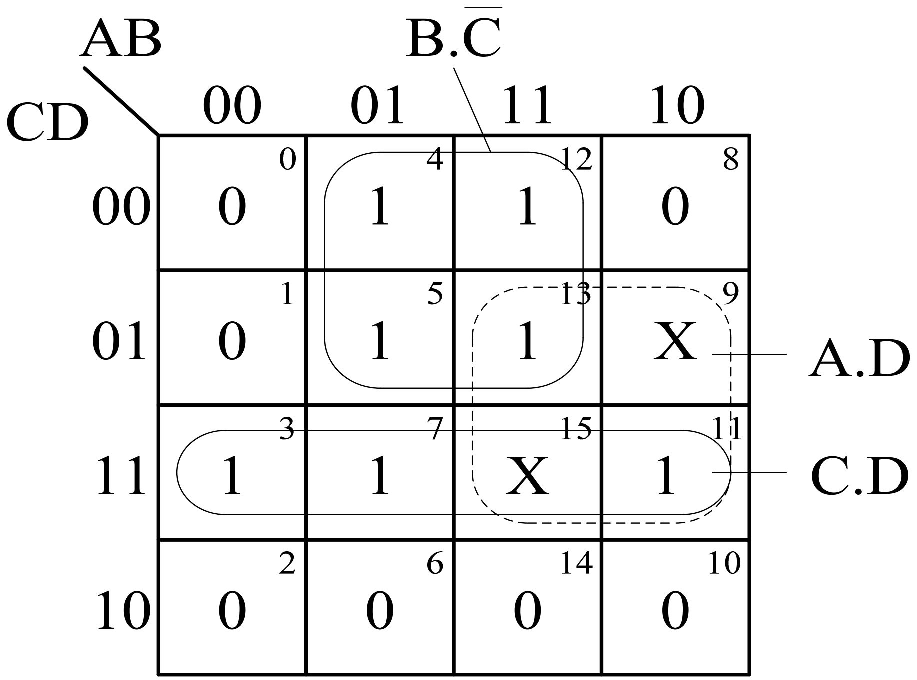
Dr Michael Tan, michael@fke.utm.my

Examples : with Don’t Cares

- Output is not specified because certain input combinations does not exist or are not allowed. Labelled as X.
- X can be assumed to be either 0 or 1 and can be used to simplify Boolean function if it can form a larger group with the 0s or the 1s.
- Unnecessary to loop the unused don't cares.

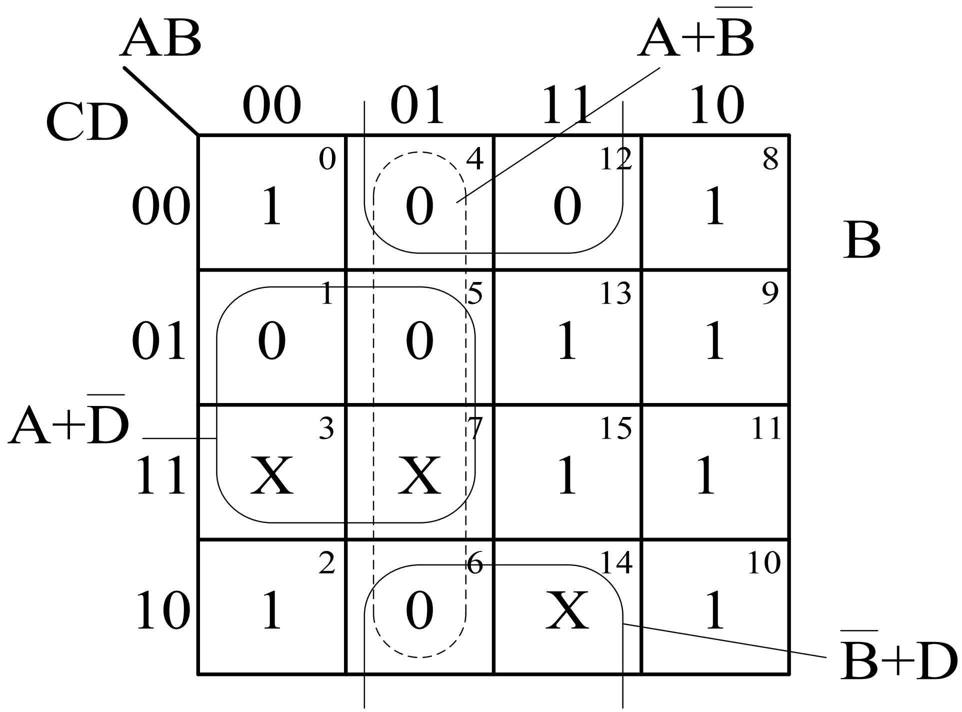
AB \ CD	00	01	11	10
00	X	0	0	X
01	1	0	1	0
11	0	0	1	X
10	1	1	1	1

Examples : with Don’t Cares



$F(A,B,C,D) = B.\overline{C}+C.D$

(i)



$F(A,B,C,D) = (A+\overline{D}).(\overline{B}+D)$

(ii)

X in cell 15 forms a quad with cells 3, 7 and 11. Although X in cell 9 can form a quad with cell 11, 13 and 15, this is unnecessary as all the 1s have already been grouped.

Cells 4, 5, 6 and 7 form a redundant quad.

Exercise :

- A function F(A,B,C,D) = Σm(7,9,11,12,13,15). The minimum SOP expression of the function is B.C' + B.D +A.D . Determine if there are any obvious don't care entries. If yes, find them.

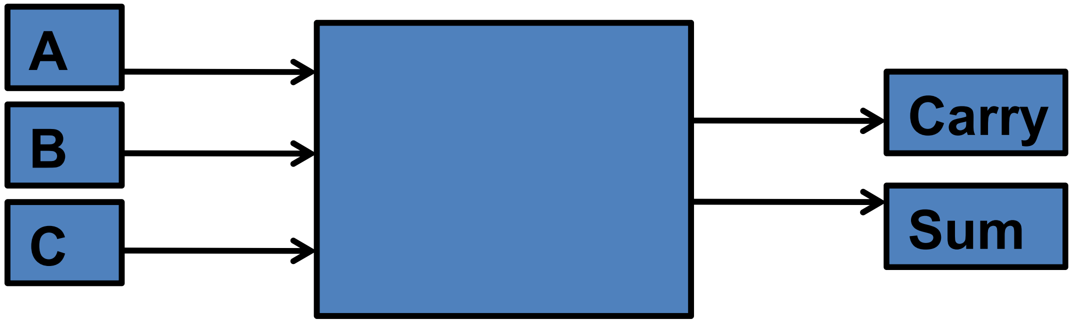
Summary

- Karnaugh map allows us to represent functions with new notation
- Representation allows for logic reduction.
  - Implement same function with less logic
- Each CELL represents
  - one minterm (SOP) or
  - one maxterm (POS)
- Each circle leads to
  - one product term (SOP) or
  - one sum term (POS)
- Not all functions can be reduced

Combinational Logic Design

Example 1:

- The block diagram of an adder circuit is shown below. Complete the truth table and implement the circuit in NAND-NAND configuration using minimum number of gates.



Combinational Logic Design

Exercise 1:

- A car alarm system has four sensors or switches which detect the following:
  - E : Engine, HIGH when turned on
  - L : Front light, HIGH when turned on
  - D : Door, HIGH when closed
  - B : Seat belt, HIGH when fastened
- The output A is a buzzer which will emit a sound (HIGH) when any of the following conditions occur:
  1. Engine is off, front light is on, and door is opened.
  2. Engine is on and seat belt not fastened.
  3. Engine is on, the seat belt is fastened but the door is open.

Design the system and implement it in NAND-NAND configuration using minimum number of gates.

Combinational Logic Design

Exercise 2:

- A combinational circuit has three binary inputs H,S, I and three binary outputs X, Y, Z. Each input represents an item that a person may order at a fast food place. The items are:
  - H: Hot dog, $3
  - S : Soda, $1
  - I : Ice cream, $2
- Each input can only be 1 or 0, which means that a customer can order each item only once (or none at all for H=0, S=0, I=0). The outputs X,Y,Z represents a 3-bit encoding $(XYZ)_2$ YZ of the total cost of the order. For example for H=1, S=1, I=0 (which adds to $4), the output should be X=1. Y=0, Z=0.
- Obtain the truth table for this circuit
- Determine the minimum expression for X,Y,Z in SOP form.
- Implement the minimum expression in NAND-NAND configuration.

Combinational Logic Design

Exercise 3:

- Design a combinational logic circuit that shows the results of total devices at alert level for patient monitoring devices: blood pressure (A), heart beat (B), oxygen level (C) and temperature (D). The outputs of the monitoring device, F1F0, will show the counting of the devices on alert level.
  - If only one device is active high (or at high risk level), then the output is 01.
  - If two devices are at alert level, then output is 10.
  - If three or more devices are at alert level, then output is 11.
  - If no device is at alert level, then the output is 00.
- Set up the truth table and find the minimum expression for F1 and F0, and draw the corresponding circuit.

K-Map

- To be good at analyzing K-Maps, you need practice
- Please look at some examples and exercises of K-Maps in your book
- Next class we’ll look at some real design examples that utilize K-Map