

Chapter 3 –L2

Boolean Algebra and Logic Simplification

KARNAUGH MAP METHOD

□ is a graphical device used to simplify a logic equation or to convert a truth table to its corresponding logic circuit in a simple, orderly process.

□ abbreviated **K map**

□ can be used for problems involving **any number of input variables,**

□ practical usefulness is limited to **six variables.**

Karnaugh Map Format

1. In order for vertically and horizontally adjacent squares to differ in only one variable, the top-to-bottom labeling must be done in the order shown-.

$$\overline{A}\overline{B}, \overline{A}B, AB, A\overline{B}$$

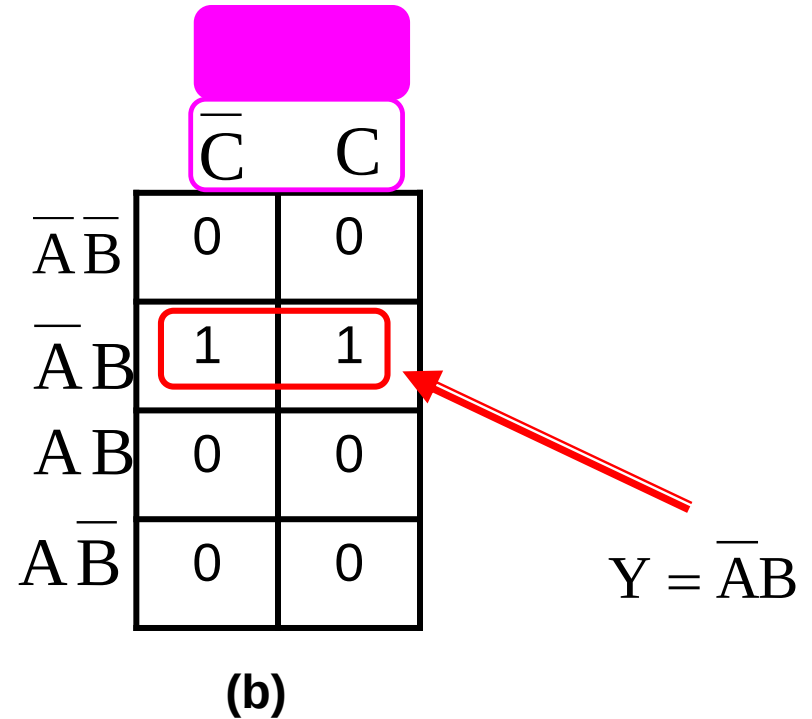
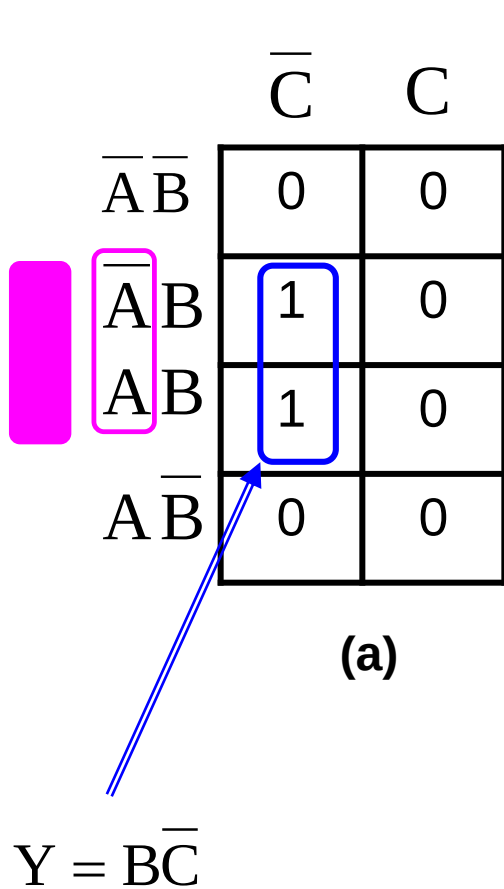
The same is true of the left-to-right labeling.

2. Once a K map has been filled with 0s and 1s, the sum-of-product expression for the output X can be obtained by ORing together those squares that contain a 1.

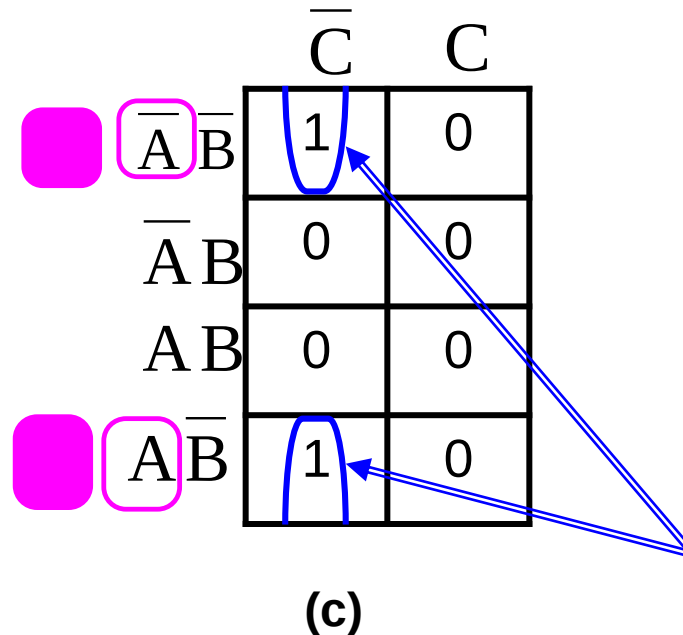
Looping

- ❑ The expression for output X can be simplified by properly combining those squares in the K map which contain 1s.
- ❑ The process for combining these 1s is called looping.

Looping Groups of Two (pairs)

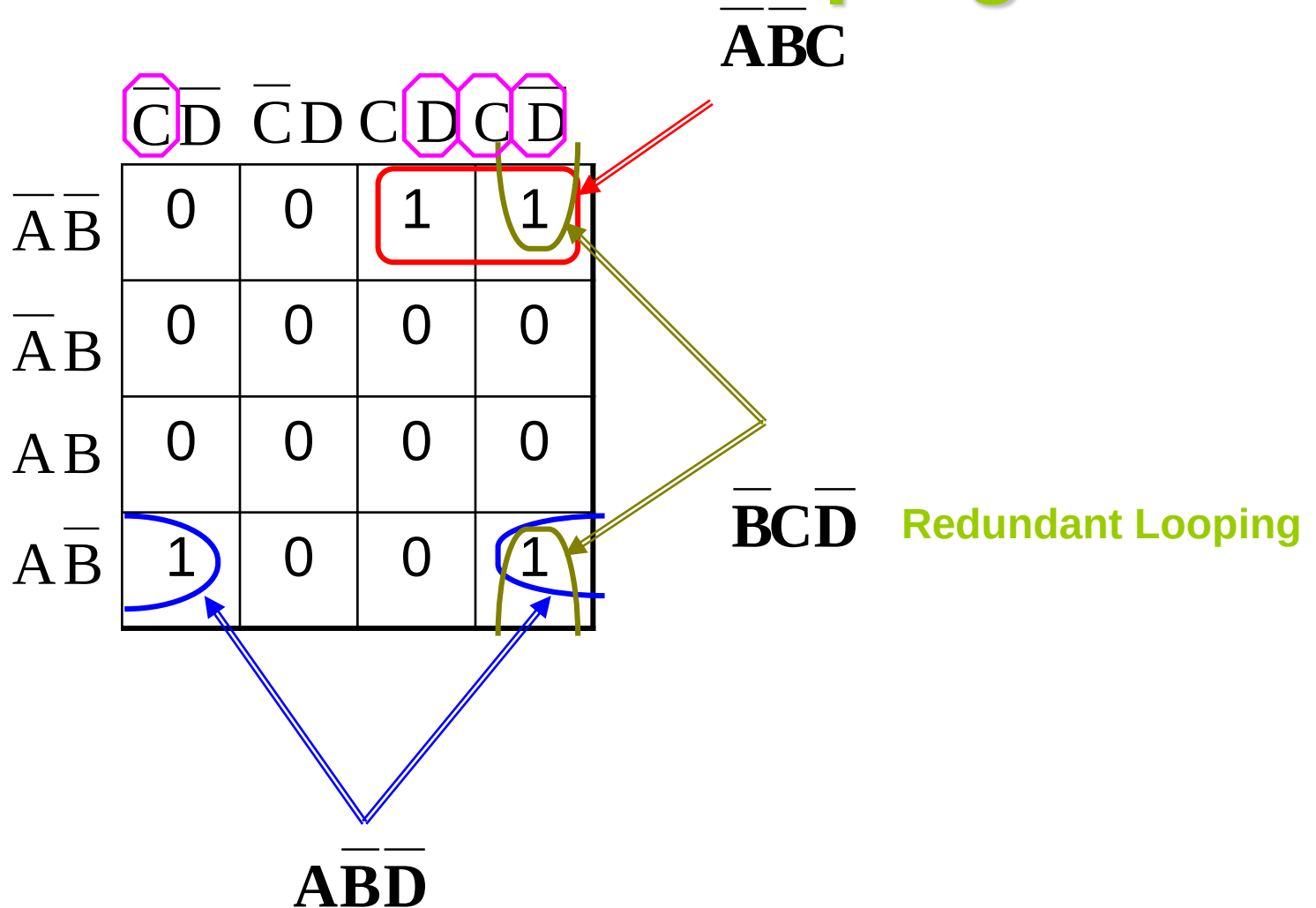


Looping Groups of Two (pairs) *(Continued)*



❖ Looping a pair of adjacent 1s in a K map eliminates the variable that appears in complemented and uncomplemented form.

Redundant Looping



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

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

Correct

Incorrect

Looping Groups of Four (Quads):

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	0	0	1
$\bar{A}B$	0	0	0	1
AB	0	0	0	1
$A\bar{B}$	0	0	0	1

(a)

$$X = C\bar{D}$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	0	0	0
$\bar{A}B$	0	0	0	0
AB	1	1	1	1
$A\bar{B}$	0	0	0	0

(b)

$$X = AB$$

Looping Groups of Four (Quads) (continued...)

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	0	0	0
$\bar{A}B$	0	1	1	0
AB	0	1	1	0
$A\bar{B}$	0	0	0	0

(c)

$$X = BD$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0	0	0	0
$\bar{A}B$	0	0	0	0
AB	1	0	0	1
$A\bar{B}$	1	0	0	1

(d)

$$X = A\bar{D}$$

Looping Groups of Four (Quads) (continued...)

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1	0	0	1
$\bar{A}B$	0	0	0	0
AB	0	0	0	0
$A\bar{B}$	1	0	0	1

(e)

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

❖ Looping a quad of 1s eliminates the two variables that appear in both complemented and uncomplemented form.

Looping Groups of Eight (octets):

	$\overline{C}\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\overline{B}$	0	0	0	0
$\overline{A}B$	1	1	1	1
AB	1	1	1	1
$A\overline{B}$	0	0	0	0

(a)

$$X = B$$

	$\overline{C}\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\overline{B}$	1	1	0	0
$\overline{A}B$	1	1	0	0
AB	1	1	0	0
$A\overline{B}$	1	1	0	0

(b)

$$X = \overline{C}$$

Looping Groups of Eight (Octets) (cont'd...)

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1	1	1	1
$\bar{A}B$	0	0	0	0
AB	0	0	0	0
$A\bar{B}$	1	1	1	1

(c)

$$Y = \bar{B}$$

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1	0	0	1
$\bar{A}B$	1	0	0	1
AB	1	0	0	1
$A\bar{B}$	1	0	0	1

(d)

$$Y = \bar{D}$$

❖ Looping an octet of 1s eliminates the three variables that appear in both complemented and uncomplemented form.

Rule for loops of any size

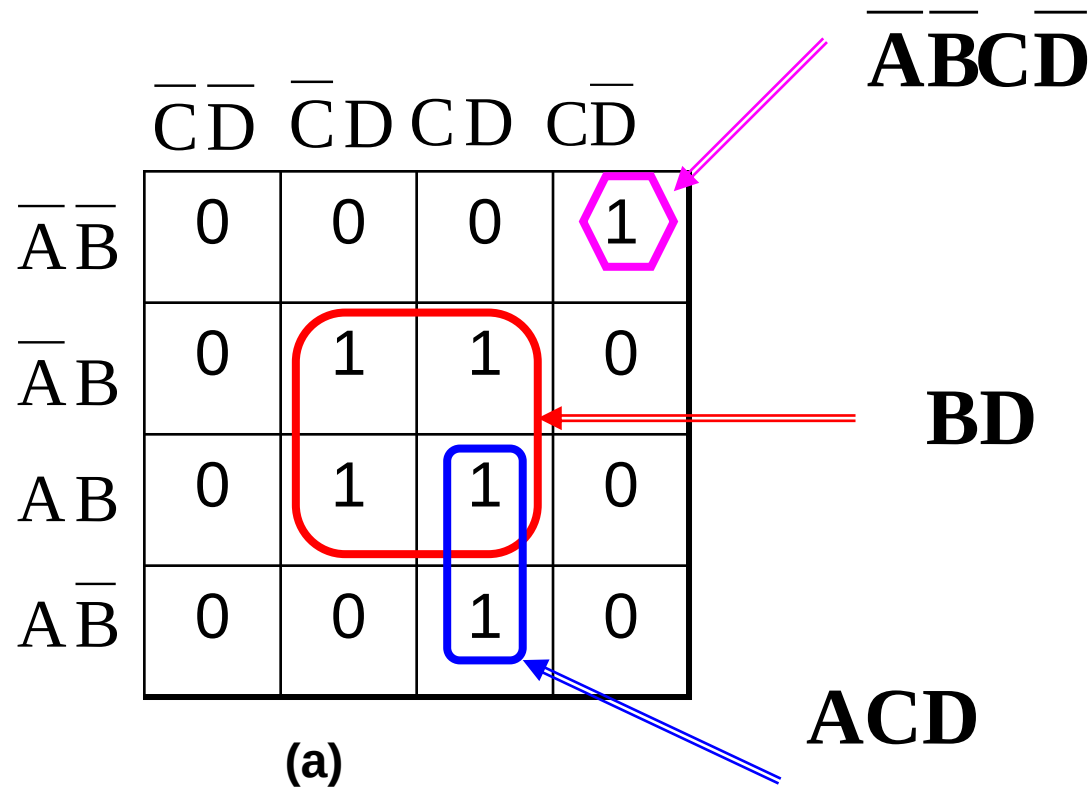
We can summarize the rule for loops of any size:

- ❑ When a variable appears in both complemented and uncomplemented form within a loop, that variable is eliminated from the expression.
- ❑ Variables that are the same for all squares of the loop must appear in the final expression.
- ❑ It should be clear that a larger loop of $1s$ eliminates more variables. To be exact, a loop of two eliminates one variable, a loop of four eliminates two, and a loop of eight eliminates three.

Complete Simplification Process

1. **Construct the K map and place 1s in those squares corresponding to the 1s in the truth table. Place 0s in the other squares.**
2. **Examine the map for adjacent 1s and loop those 1s, which are not adjacent to any other 1s. These are called isolated 1s.**
3. **Next, look for those 1s, which are adjacent to only one other 1. Loop any pair containing such a 1.**
4. **Loop any octet even it contains some 1s that have already been looped.**
5. **Loop any quad that contains one or more 1s, which have not already been looped, is making sure to use the minimum number of loops.**
6. **Loop any pairs necessary to include any 1s that have not yet been looped, making sure to use the minimum number of loops.**
7. **Form the OR sum of all the terms generated by each loop.**

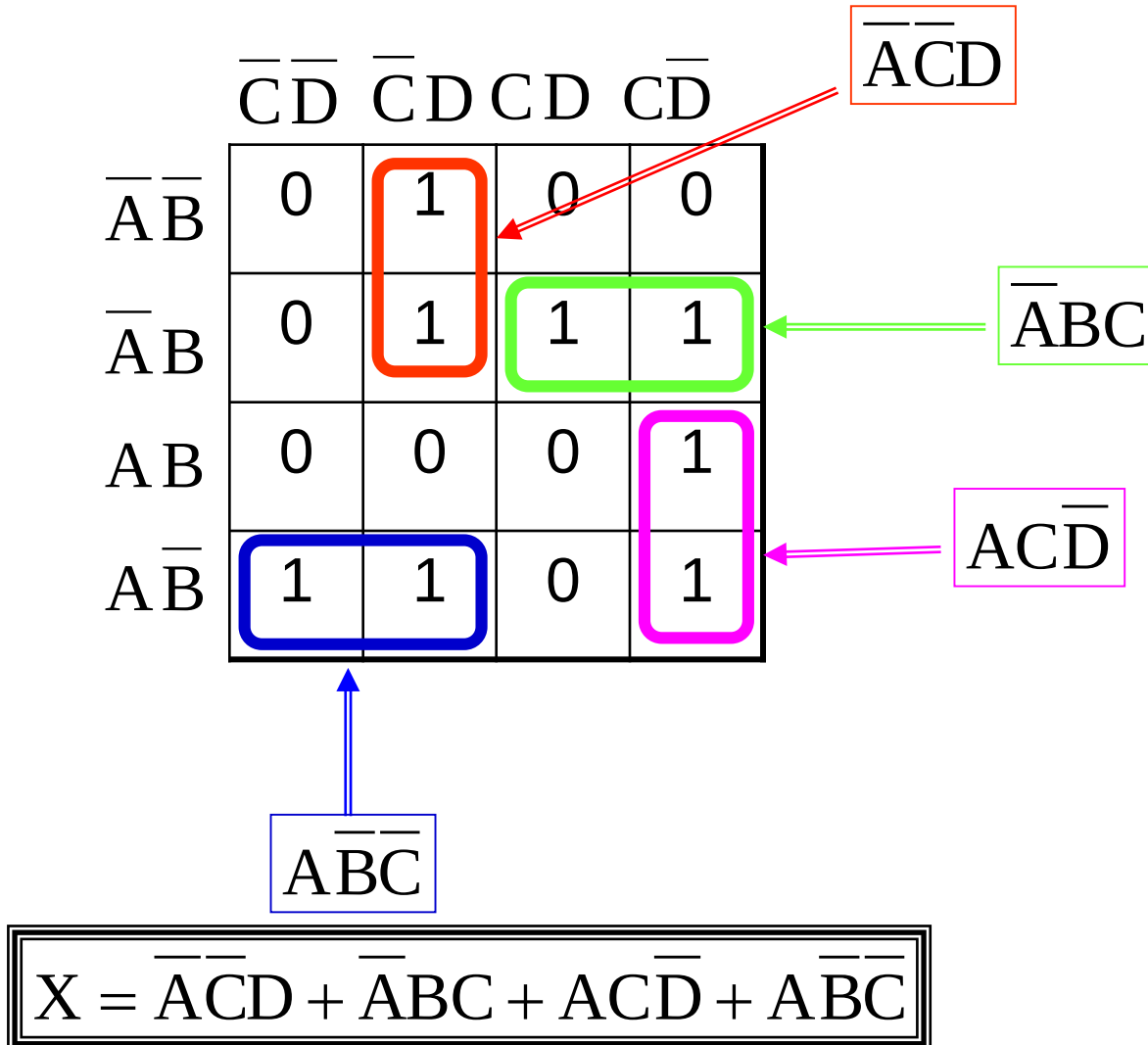
Example#1



$$X = \bar{A}\bar{B}\bar{C}\bar{D} + ACD + BD$$

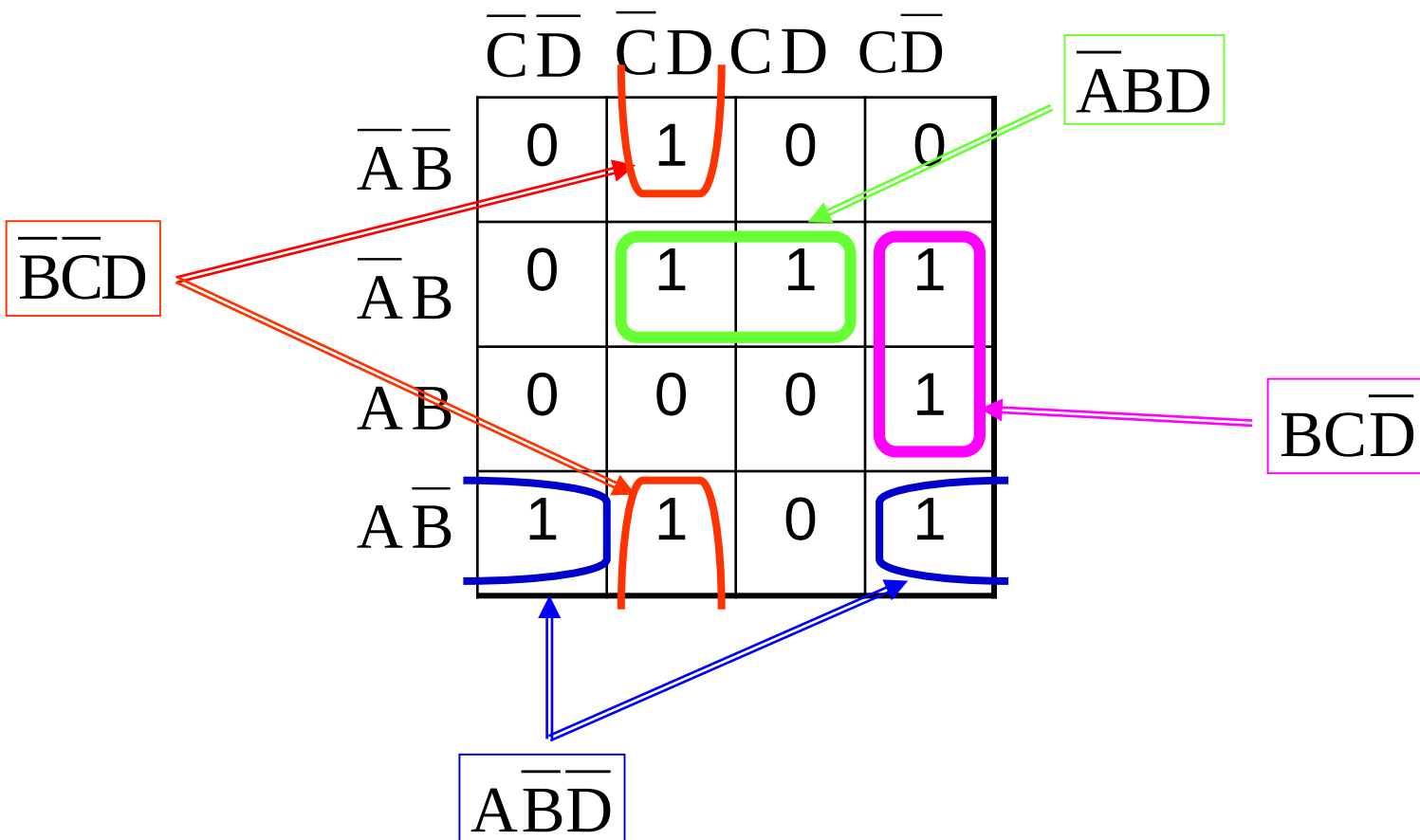
Example#2

Option #1



Example#2

Option # 2



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

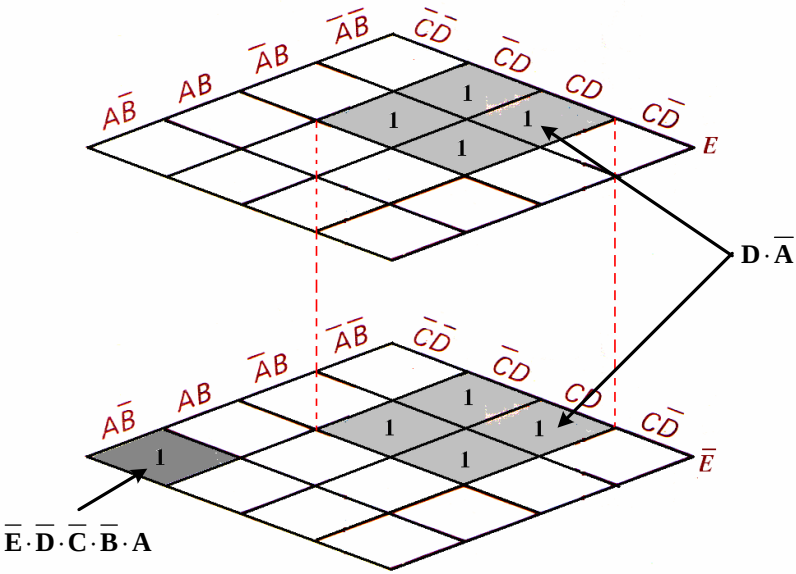
A Five Variable Karnaugh Map

- ❑ The Karnaugh map becomes three dimensional when solving logic problems with more than four variables. A three-dimensional Karnaugh map will be used in this section.
- ❑ Consider the truth table shown in Figure below (a). The truth table must have 32 (2^5) rows to show all the combinations for five variables.
- ❑ The unsimplified Boolean expression for the truth table can be written as

INPUTS					OUTPUT	INPUTS					OUTPUT
E	D	C	B	A	Y	E	D	C	B	A	Y
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	1	1	0	0	0	1	1
0	0	0	1	0	0	1	0	0	1	0	0
0	0	0	1	1	0	1	0	0	1	1	0
0	0	1	0	0	0	1	0	1	0	0	1
0	0	1	0	1	1	1	0	1	0	1	0
0	0	1	1	0	0	1	0	1	1	0	0
0	0	1	1	1	0	1	0	1	1	1	0
0	1	0	0	0	1	1	1	0	0	0	0
0	1	0	0	1	0	1	1	0	0	1	0
0	1	0	1	0	0	1	1	0	1	0	0
0	1	0	1	1	0	1	1	0	1	1	0
0	1	1	0	0	1	1	1	1	0	0	1
0	1	1	0	1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	1	1	1	0	0
0	1	1	1	1	0	1	1	1	1	1	0

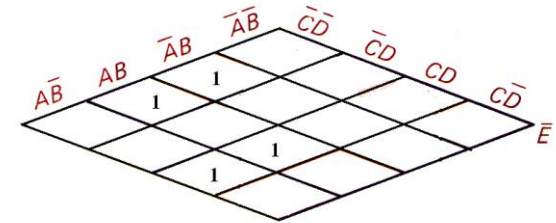
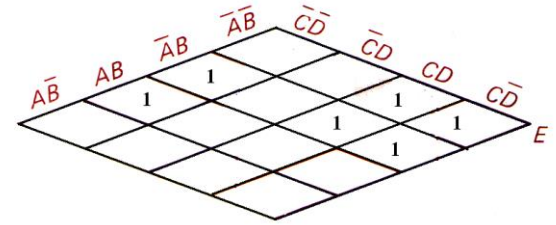
$$Y = \overline{E} \cdot \overline{D} \cdot \overline{C} \cdot \overline{B} \cdot A + \overline{E} \cdot D \cdot \overline{C} \cdot \overline{B} \cdot \overline{A} + \overline{E} \cdot D \cdot \overline{C} \cdot B \cdot \overline{A} + \overline{E} \cdot D \cdot C \cdot \overline{B} \cdot \overline{A} + \overline{E} \cdot D \cdot C \cdot B \cdot \overline{A} + \\ E \cdot D \cdot \overline{C} \cdot \overline{B} \cdot \overline{A} + E \cdot D \cdot \overline{C} \cdot B \cdot \overline{A} + E \cdot D \cdot C \cdot \overline{B} \cdot \overline{A} + E \cdot D \cdot C \cdot B \cdot \overline{A}$$

INPUTS					OUTPUT	INPUTS					OUTPUT
E	D	C	B	A	Y	E	D	C	B	A	Y
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	1	1	0	0	0	1	1
0	0	0	1	0	0	1	0	0	1	0	0
0	0	0	1	1	0	1	0	0	1	1	0
0	0	1	0	0	0	1	0	1	0	0	1
0	0	1	0	1	1	1	0	1	0	1	0
0	0	1	1	0	0	1	0	1	1	0	0
0	0	1	1	1	0	1	0	1	1	1	0
0	1	0	0	0	1	1	1	0	0	0	0
0	1	0	0	1	0	1	1	0	0	1	0
0	1	0	1	0	0	1	1	0	1	0	0
0	1	0	1	1	0	1	1	0	1	1	0
0	1	1	0	0	1	1	1	1	0	0	1
0	1	1	0	1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	1	1	1	0	0
0	1	1	1	1	0	1	1	1	1	1	0

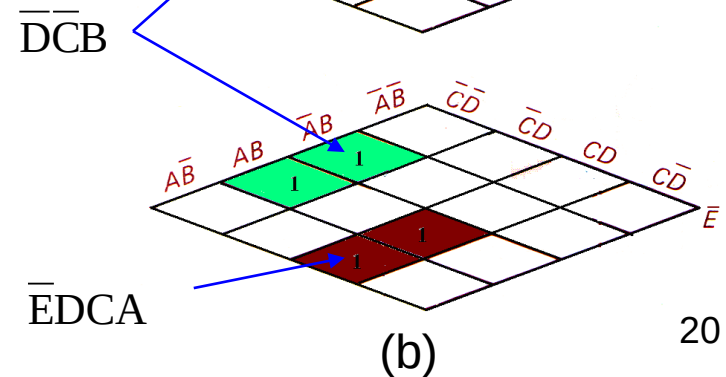
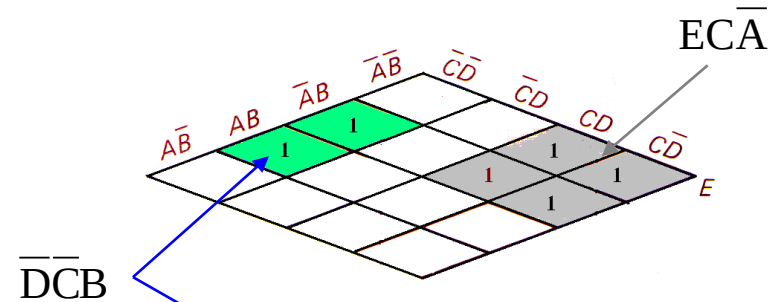


Example

- ❑ Simplify the following 5-input variables k-map shown in Figure (a).
- ❑ The simplified expression taken from the map of Figure (a) is developed and shown in Figure (b).
- ❑ Combining these AND terms using OR gate yields



(a)



$$Y = \overline{D} \cdot \overline{C} \cdot B + E \cdot C \cdot \overline{A} + \overline{E} \cdot D \cdot C \cdot A$$

“Don’t Care” Conditions

- ❑ Some logic circuits can be designed so that there are certain input conditions for which there are no specified output levels, usually because these input conditions will never occur.
- ❑ In other words, there will be certain combinations of input levels where we “don’t care” whether the output is HIGH or LOW.
- ❑ This is illustrated in the truth table of figure (a).

A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	X
1	0	0	X
1	0	1	1
1	1	0	1
1	1	1	1

	$\bar{C}$	C
$\bar{A}\bar{B}$	0	0
$\bar{A}B$	0	x
AB	1	1
$A\bar{B}$	x	1

(b)

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

Z=A

(c)

- ❖ Whenever “don’t care” conditions occur, we have to decide which ones to change to 0 and which to 1 to produce the best K-map looping (i.e., the simplest expression).
- ❖ This decision is not always an easy one.

Example on “Don’t Care”

	$\overline{C}\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\overline{B}$	0	0	0	0
$\overline{A}B$	1	0	0	X
AB	1	1	1	1
$A\overline{B}$	0	X	X	0

Redundant loop

$$Y = \overline{B}\overline{D} + AD$$

~~$$Y = \overline{B}\overline{D} + AD + AB$$~~