Logic Gate

College of
Engineering & Technology

Al-Mustaqbal
University

Level 1 , Semester 1
@ Department of prosthetic and orthotic Engineering

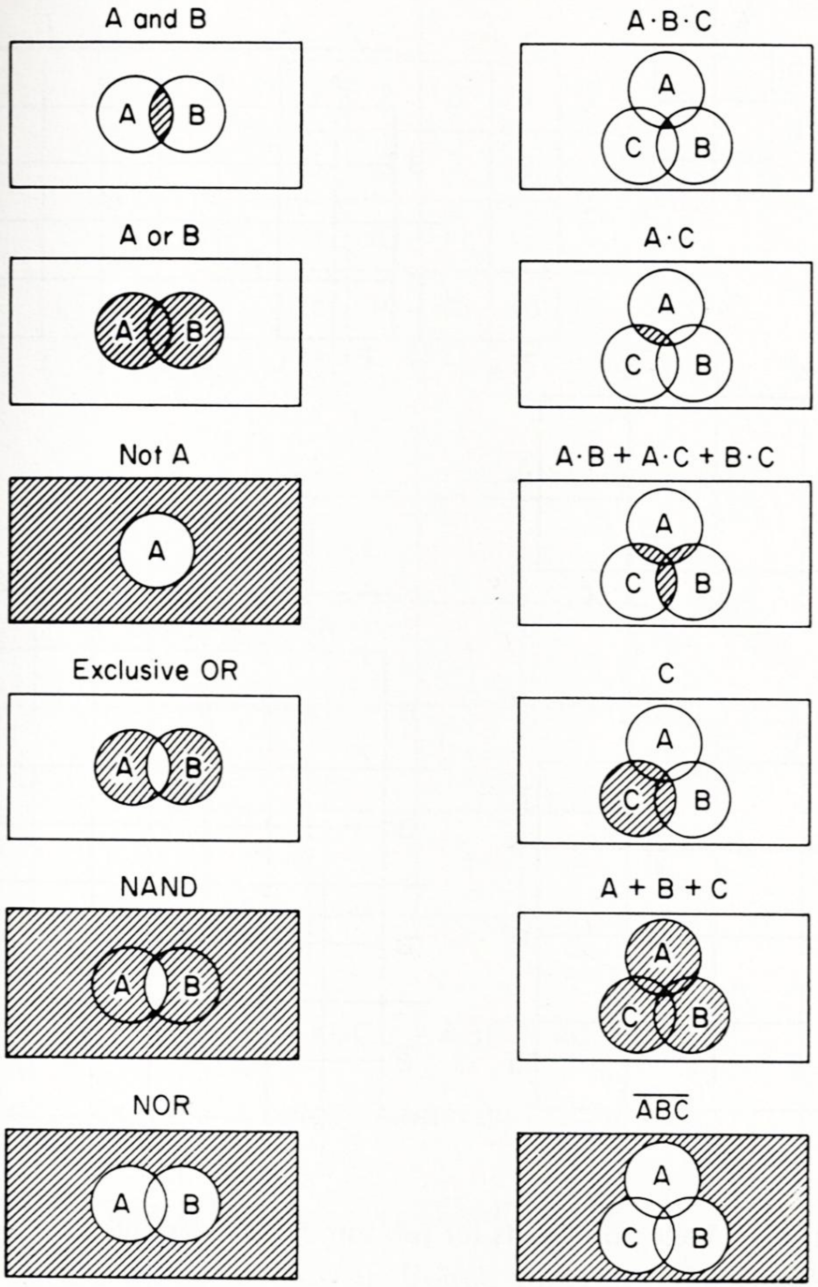
Prepared by
Dr. Samir Badrawi
2024-2025

Arithmetic Operations & Boolean Algebra

The majority of this course material is based on text and presentations of :

Rules of Boolean Algebra

1. $A + 0 = A$
2. $A + 1 = 1$
3. $A \cdot 0 = 0$
4. $A \cdot 1 = \mathrm{A}$
5. $A + A = A$
6. $A + \overline{A} = 1$
7. $A \cdot A = A$
8. $A \cdot \overline{A} = 0$
9. $\overline{\overline{A}} = A$
10. $A + AB = A$
11. $A + \overline{A}B = A + B$
12. $(A + B)(A + C) = A + BC$



Typical Venn diagrams.

Venn Diagram

Venn Diagram
is a
pictorial representation
of
Logical expressions

Some typical examples of Venn diagrams for logic functions and basic equations

(the rectangle represents the Universe)

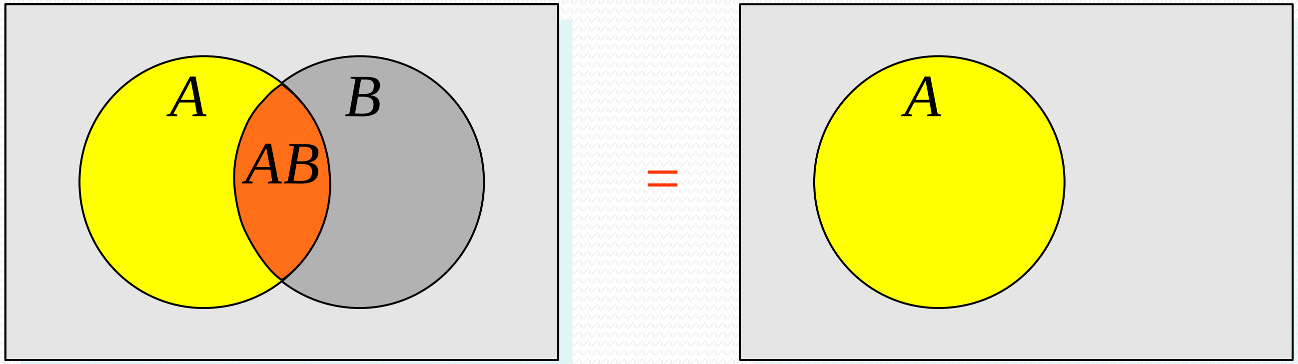
Rules of Boolean Algebra

Rules of Boolean algebra can be illustrated with *Venn* diagrams. The variable *A* is shown as an area.

Boolean rule number (10) : $A + AB = A$

can be illustrated easily with a diagram. Add an overlapping area to represent the variable *B*.

The overlap region between A and B represents *AB*.



The diagram visually shows that $A + AB = A$.

Other rules can be illustrated with the diagrams as well.

Rules of Boolean Algebra

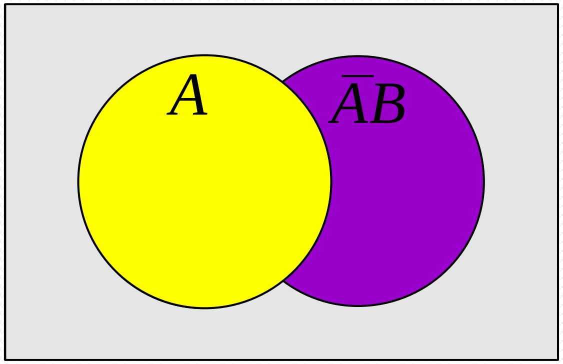
Example Illustrate the rule $A + \overline{A}B = A + B$ with a Venn diagram. (Rule 11)

Solution

This time, $\overline{A}$ is represented by the blue area and B again by the red circle.

The intersection represents $\overline{A}B$.

Notice that $A + \overline{A}B = A + B$



Rules of Boolean Algebra

Rule 12, which states that $(A + B)(A + C) = A + BC$, can be proven by applying earlier rules as follows:

$$
\begin{aligned}
(A + B)(A + C) &= AA + AC + AB + BC \\
&= A + AC + AB + BC \\
&= A(1 + C + B) + BC \\
&= A \cdot 1 + BC \\
&= A + BC
\end{aligned}
$$

This rule is a little more complicated, but it can also be shown with a Venn diagram, as given on the following slide…

Rule 12 :

$(A + B)(A + C) = A + BC$ **can be proven by Veen Diagram :-**

L. H .S. of the equ.

Three areas represent the variables *A*, *B*, and *C*.

The area representing *A* + *B* is shown in yellow.

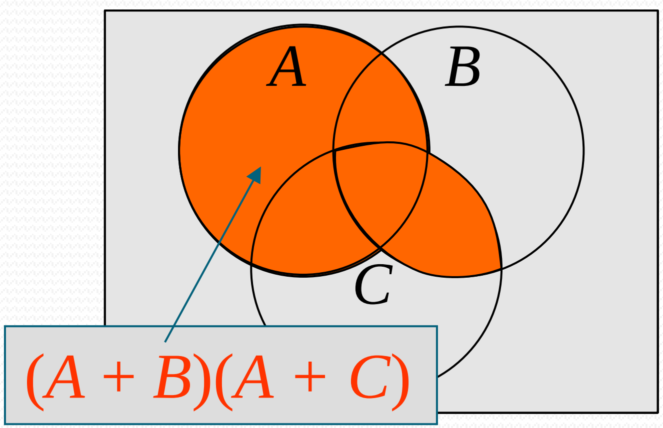
The area representing *A* + *C* is shown in red.

The overlap of red and yellow is shown in orange.

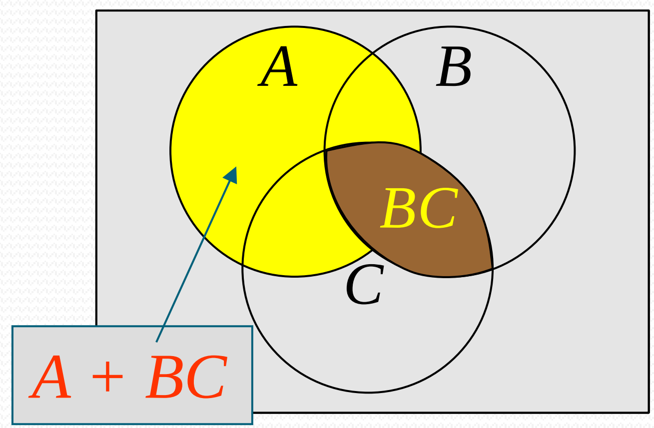
R. H .S. of the equ.

The overlapping area between *B* and C represents *BC*.

ORing with *A* gives the same area as before.



=



EXAMPLE 4–1

Determine the values of A, B, C, and D that make the sum term $A + \overline{B} + C + \overline{D}$ equal to 0.

Solution

For the sum term to be 0, each of the literals in the term must be 0. Therefore, $A = 0$, $B = 1$ so that $\overline{B} = 0$, $C = 0$, and $D = 1$ so that $\overline{D} = 0$.

$$A + \overline{B} + C + \overline{D} = 0 + \overline{1} + 0 + \overline{1} = 0 + 0 + 0 + 0 = 0$$

EXAMPLE 4–2

Determine the values of A, B, C, and D that make the product term $A\overline{B}C\overline{D}$ equal to 1.

Solution

For the product term to be 1, each of the literals in the term must be 1. Therefore, $A = 1$, $B = 0$ so that $\overline{B} = 1$, $C = 1$, and $D = 0$ so that $\overline{D} = 1$.

$$A\overline{B}C\overline{D} = 1 \cdot \overline{0} \cdot 1 \cdot \overline{0} = 1 \cdot 1 \cdot 1 \cdot 1 = 1$$

DeMorgan's Theorems

DeMorgan's 1st Theorem

The complement of a product of variables is equal to the sum of the complemented variables.

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

DeMorgan's 2nd Theorem

The complement of a sum of variables is equal to the product of the complemented variables.

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

To apply DeMorgan's theorem to the expression, first invert the sign, then break the overbar covering both terms.

DeMorgan's Theorem

Example

Apply DeMorgan's theorem to remove the overbar covering both terms from the expression $X = \overline{\overline{C} + D}$.

Solution

To apply DeMorgan's theorem to the expression, you can break the overbar covering both terms and change the sign between the terms. This results in $X = \overline{\overline{C}} \cdot \overline{D}$. Deleting the double bar gives $X = C \cdot \overline{D}$.

DeMorgan's Theorem

EXAMPLE 4–3

Apply DeMorgan's theorems to the expressions $\overline{XYZ}$ and $\overline{X + Y + Z}$.

Solution

$$\overline{XYZ} = \overline{X} + \overline{Y} + \overline{Z}$$

$$\overline{X + Y + Z} = \overline{X}\,\overline{Y}\,\overline{Z}$$

Related Problem

Apply DeMorgan's theorem to the expression $\overline{\overline{X} + \overline{Y} + \overline{Z}}$.

EXAMPLE 4–4

Apply DeMorgan's theorems to the expressions $\overline{WXYZ}$ and $\overline{W + X + Y + Z}$.

Solution

$$\overline{WXYZ} = \overline{W} + \overline{X} + \overline{Y} + \overline{Z}$$

$$\overline{W + X + Y + Z} = \overline{W}\,\overline{X}\,\overline{Y}\,\overline{Z}$$

Related Problem

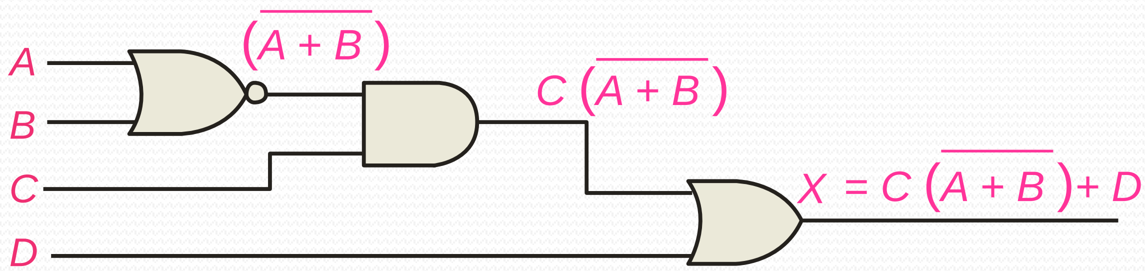
Apply DeMorgan's theorem to the expression $\overline{\overline{W}\,\overline{X}\,\overline{Y}\,\overline{Z}}$.

Boolean Analysis of Logic Circuits

Combinational logic circuits can be analyzed by writing the expression for each gate and combining the expressions according to the rules for Boolean algebra.

Example Apply Boolean algebra to derive the expression for *X*.

Solution Write the expression for each gate:



Applying DeMorgan's theorem and the distribution law:

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

SOP and POS forms

Boolean expressions can be written in the **sum-of-products** form (**SOP**) or in the **product-of-sums** form (**POS**).

These forms can simplify the implementation of combinational logic.

In both forms, an overbar cannot extend over more than one variable.

An expression is in **SOP** form when two or more product terms are summed as in the following examples:

$\overline{A}\,\overline{B}\,\overline{C} + A\,B$ $\quad$ $A\,B\,\overline{C} + \overline{C}\,\overline{D}$ $\quad$ $C\,D + \overline{E}$

An expression is in **POS** form when two or more sum terms are multiplied as in the following examples:

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

SOP Standard form

In **SOP standard form**, <u>every variable in the domain must appear in each term</u> in this standard form.

This form is useful for constructing truth tables or for implementing logic in certain digital devices called PLDs (Programmable Logic Devices)

Nonstandard form can be expanded to standard form by <u>multiplying</u> this term by a term consisting of the sum of the missing variable and its complement.

Example

Convert $X = \overline{A}\,\overline{B} + A B C$ to standard form.

Solution

The first term does not include the variable *C*. Therefore, multiply it by the $(C + \overline{C})$, which = 1:

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

POS Standard form

In **POS standard form**, every variable in the domain must appear in each sum term of the expression.

Nonstandard POS expression can be expanded to standard form by adding the product of the missing variable and its complement and applying rule 12, which states that $(A + B)(A + C) = A + BC$.

Note. can we Rewrite Rule (12) as: $X + YZ = (X + Y)(X + Z)$ (Yes/No) ??

Example Convert $X = (\overline{A} + \overline{B})(A + B + C)$ to standard form.

Solution The first sum term does not include the variable C.
Therefore, add $C\overline{C}$ and expand the result by rule 12.

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

(Prove it ?)

Binary representation of SOP and POS forms

SOP standard form $A\overline{B}C\overline{D} = 1 \cdot \overline{0} \cdot 1 \cdot \overline{0} = 1 \cdot 1 \cdot 1 \cdot 1 = 1$

POS standard form $A + \overline{B} + C + \overline{D} = 0 + \overline{1} + 0 + \overline{1} = 0 + 0 + 0 + 0 = 0$

Converting standard SOP to POS

SOP standard form

$$\overline{A}\,\overline{B}\,\overline{C} + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}C + ABC$$

$$000 + 010 + 011 + 101 + 111$$

The equivalent POS standard form contains the other three remaining terms $001,\ 100\ \ and\ \ 110$

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

Converting SOP to truth table

1. First list all possible combinations of binary values of the variables in the expression.
2. Convert the SOP to standard form if it is not already.
3. Place a 1 in the output column for each binary value that makes the standard SOP expression a 1 and place a 0 for all the remaining binary values

<u>Example</u>: Develop a Truth Table for the standard SOP Expression :-

$$\overline{A}\,\overline{B}C + A\overline{B}\,\overline{C} + ABC$$

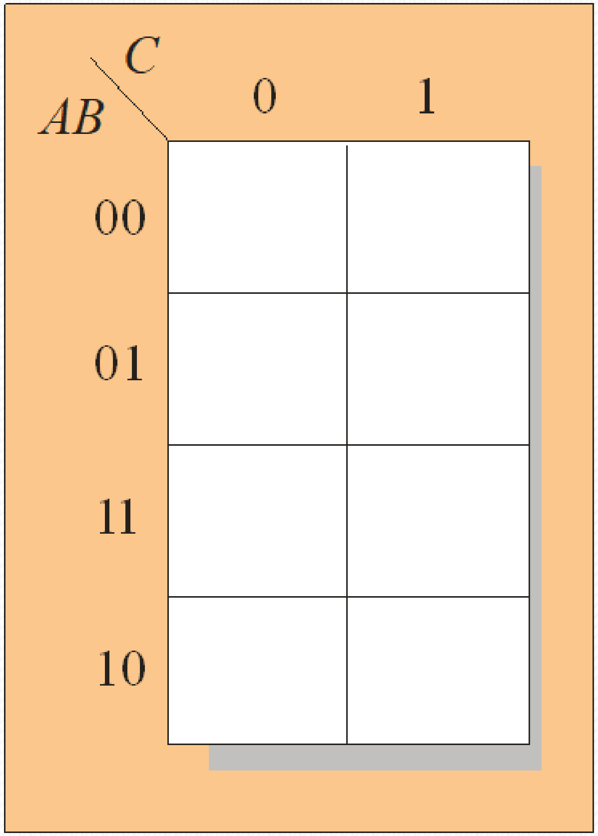
<u>Solution</u> :Three variables, then 8 possible combinations.
For each product term in the Expression, place (1) in o/p, and place (0) in for the other terms in o/p

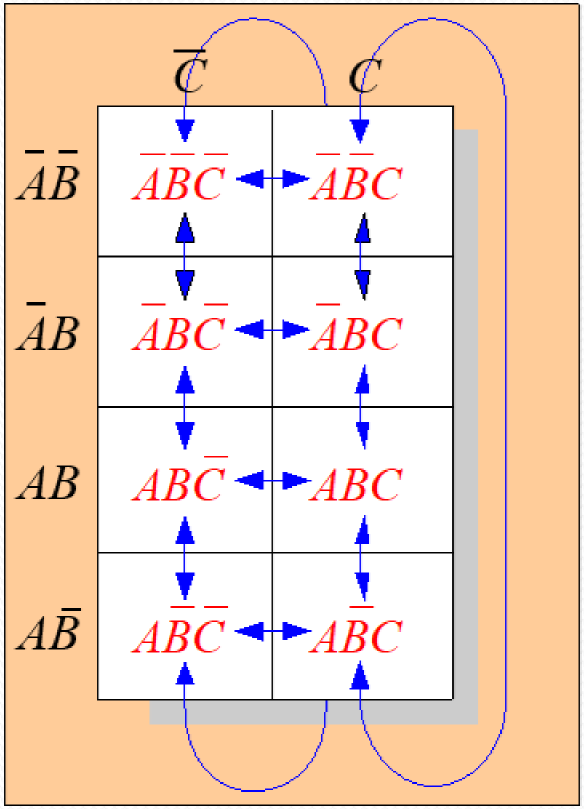
I/P A	B	C	O/P X	PRODUCT TERM
0	0	0	0	
0	0	1	1	$\overline{A}\,\overline{B}\,C$
0	1	0	0	
0	1	1	0	
1	0	0	1	$A\,\overline{B}\,\overline{C}$
1	0	1	0	
1	1	0	0	
1	1	1	1	ABC

Karnaugh maps

- Array of cells, each cell represents one possible product term
- It is a tool for **simplifying combinational logic** with 3 or 4 variables.
- *For 3 variables, 8 cells are required (2^3).*
- *Each cell is adjacent to cells that are immediately next to it on any of its four sides.*
- ***A cell is not adjacent to the cells that diagonally touch any of its corners.***
- ***"wrap-around" adjacency means the top row is adjacent to the bottom row and left column to right column.***

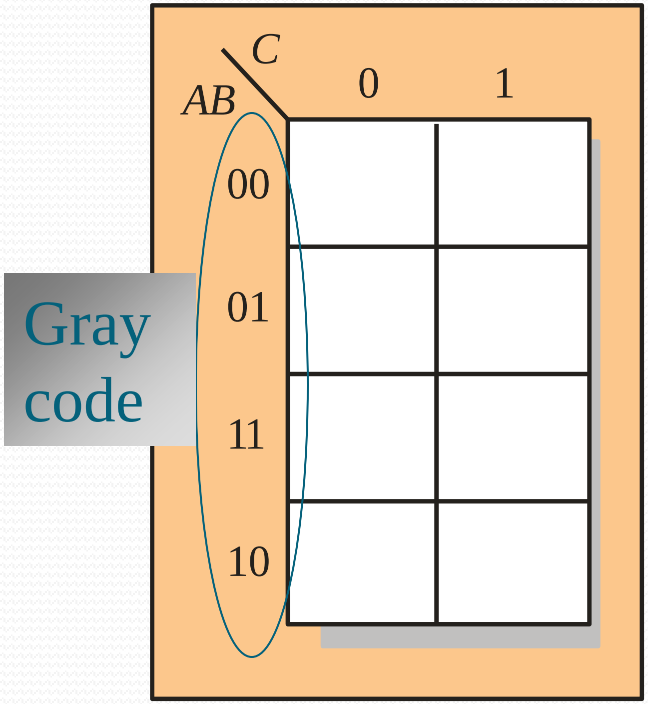
Karnaugh maps

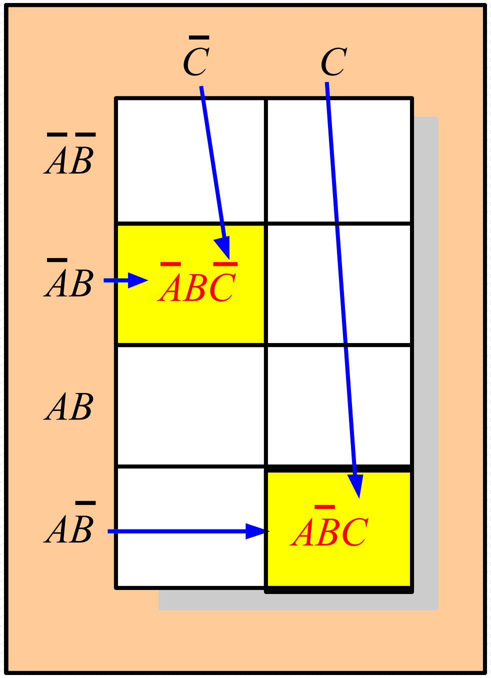




Karnaugh maps

The numbers are entered in gray code.





Grouping the 1s

The goal in **simplifying combinational logic** is to maximize the size of the groups and to minimize the number of the groups

•A group must contain either 1, 2, 4, 8, or 16 cells.

•Each cell in a group must be adjacent to one or more cells in that same group.

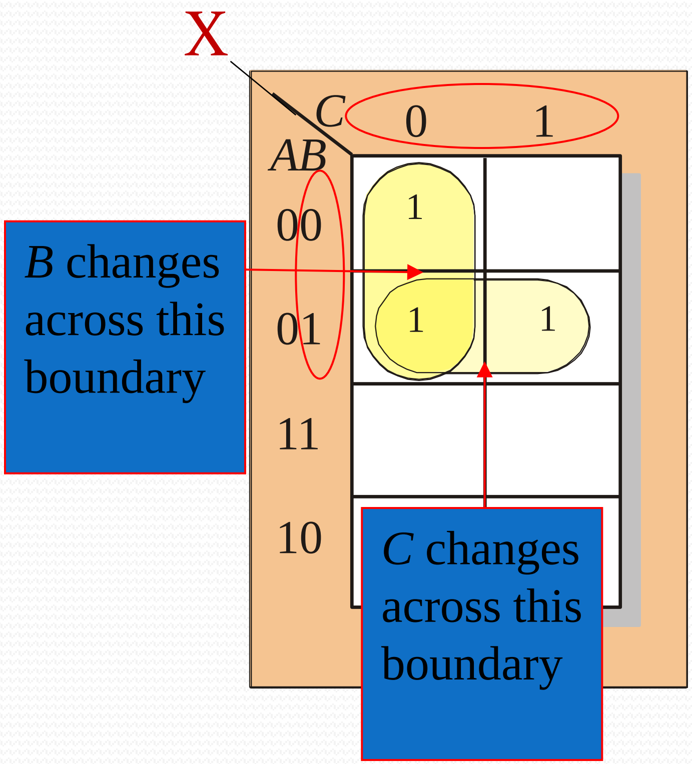
•Include the largest possible # of 1s in a group in accordance with rule 1

•Each 1 on the map must be included in at least one group.

Karnaugh maps

K-maps can simplify combinational logic by grouping cells and eliminating variables that change.

Example Group the 1's on the map and read the minimum logic.



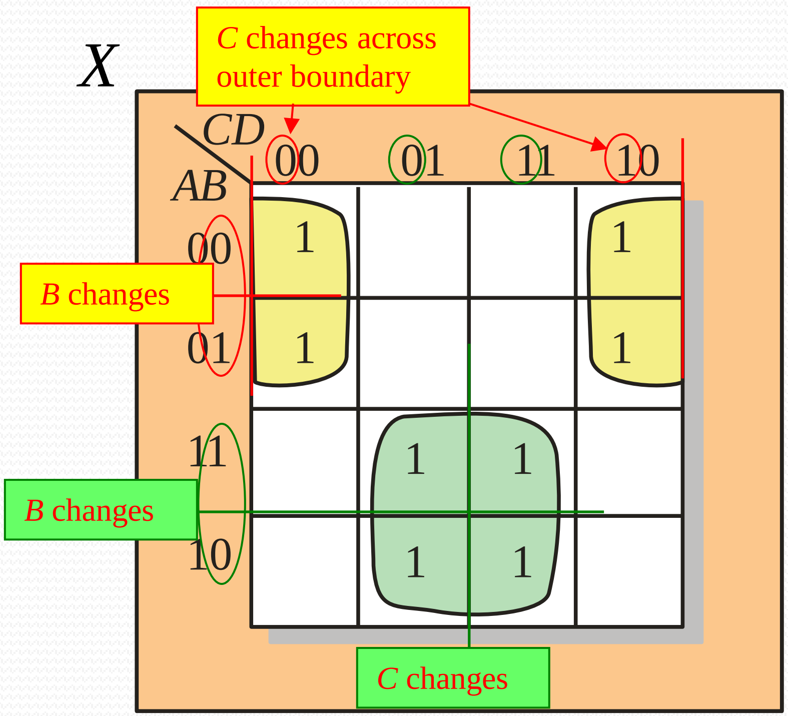
Solution

1. Group the 1's into two overlapping groups as indicated.
2. Read each group by eliminating any variable that changes across a boundary.
3. The vertical group is read $\overline{A}\overline{C}$.
4. The horizontal group is read $\overline{A}B$.

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

Karnaugh maps

Example Group the 1's on the map and read the minimum logic.



Solution

1. Group the 1's into two separate groups as indicated.
2. Read each group by eliminating any variable that changes across a boundary.
3. The upper (**yellow**) group is read as $\bar{A}\bar{D}$.
4. The lower (**green**) group is read as AD.

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