

Logical Adjacency

K–Maps are a standard method for simplification of Boolean expressions.

The difference between K–Maps and Quine–McCluskey (Q–M) is that K–M is well suited to manual solutions of small problems, and Q–M lends itself to automated solutions.

Logical adjacency is the basis for all Boolean simplification methods.

The K–Map approach is a manual procedure that transforms logical adjacency into physical adjacency on the paper. Simplification is done by inspection.

The key idea behind logical adjacency is expressed in the following simplifications of Boolean expressions.

$$\begin{aligned}
 X \bullet Y + X \bullet \bar{Y} &= X \bullet (Y + \bar{Y}) = X \bullet 1 = X \\
 (X + Y) \bullet (X + \bar{Y}) &= X \bullet X + X \bullet \bar{Y} + Y \bullet X + Y \bullet \bar{Y} \\
 &= X \bullet X + X \bullet \bar{Y} + X \bullet Y + 0 \\
 &= X \bullet X + X \bullet \bar{Y} + X \bullet Y \\
 &= X + X \bullet (\bar{Y} + Y) = X \bullet 1 = X + X = X
 \end{aligned}$$

Logical Adjacency (Part 2)

Two Boolean terms are said to be **logically adjacent** when they contain the same variables and differ in the form of exactly one variable.

Put another way, only one literal is different; a variable appears negated in one term and is not negated in the other term. All other variables appear in the same way.

Consider the following lists of terms.

List 1 X \bar{X}

List 2 $X \cdot Y$ $X \cdot \bar{Y}$ $\bar{X} \cdot \bar{Y}$ $\bar{X} \cdot Y$

List 3 $(X + Y)$ $(X + \bar{Y})$ $(\bar{X} + \bar{Y})$ $(\bar{X} + Y)$

In each of the lists, each term is logically adjacent not only to the term before it but also to the term following.

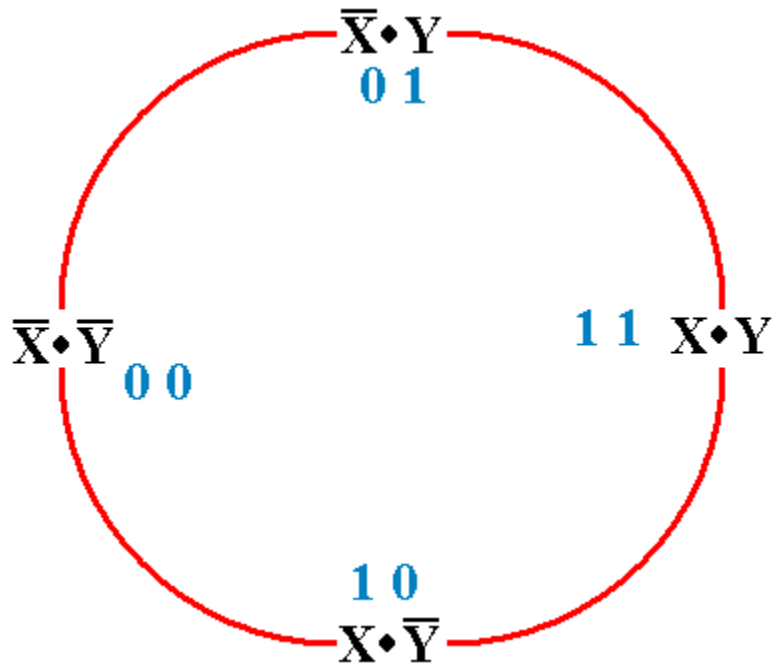
In particular $X \cdot Y$ is adjacent to both $X \cdot \bar{Y}$ and $\bar{X} \cdot Y$

$(X + Y)$ is adjacent to both $(X + \bar{Y})$ and $(\bar{X} + Y)$.

In other words, the adjacency list is “circular”; the first item follows the last one.

Logical Adjacency (Part 3)

Here is the “circle” for SOP terms.



$\bar{X} \cdot \bar{Y}$	$\bar{X} \cdot Y$	$X \cdot Y$	$X \cdot \bar{Y}$
00	01	11	10

$\bar{X} \cdot Y$	$X \cdot Y$	$X \cdot \bar{Y}$	$\bar{X} \cdot \bar{Y}$
01	11	10	00

$X \cdot Y$	$X \cdot \bar{Y}$	$\bar{X} \cdot \bar{Y}$	$\bar{X} \cdot Y$
11	10	00	01

$X \cdot \bar{Y}$	$\bar{X} \cdot \bar{Y}$	$\bar{X} \cdot Y$	$X \cdot Y$
10	00	01	11

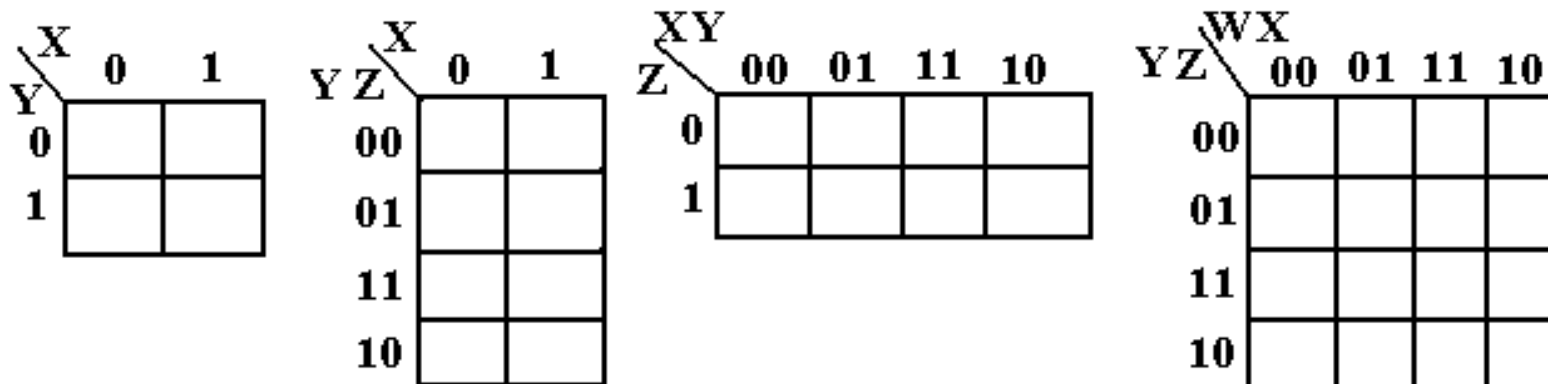
To produce a valid ordering, start at any part of the circle and read either clockwise or counter-clockwise. Four valid orderings are shown at the right.

K-Maps for Small Variable Count: 2, 3, and 4

Some textbooks show K-Maps for 5 and 6 variables.

These are so complex as to be useless for manual computation. In those cases the Quine-McCluskey method is used.

K-maps are shown as rectangles. Here are K-maps for 2, 3, and 4 variables.



Note the structure of the list in cases where either a row or column corresponds to two variables: 00, 01, 11, 10.

These are in the form of a Gray code, in which each term differs in exactly one bit from the term preceding it and following it.

The Gray code structure causes logical adjacency to be seen as physical adjacency.

The “Complete” K–Map and Its Common Forms

Every entry in a K–map is either a 0 or a 1. These are the two values that a Boolean function can assume.

While a “Complete” K–map has all entries filled, all common forms have only a subset filled. Either the 1’s are placed and the 0’s omitted, or vice–versa.

	$Z \backslash XY$	00	01	11	10
0		0	0	1	0
1		0	1	1	1

	$Z \backslash XY$	00	01	11	10
0				1	
1			1	1	1

	$Z \backslash XY$	00	01	11	10
0		0	0		0
1		0			

The three K–maps shown above contain identical information.

The second K–map shows the entries with values equal to 1.

Those entries not shown have value 0.

The third K–map shows the entries with values equal to 0.

Those entries not shown have value 1.

Creating a K-Map

The standard K-map is used to simplify Boolean expressions in canonical form; each term has exactly one literal for every Boolean variable.

The method can be expanded to Boolean expressions not in canonical form.

SOP Expressions

Place a “1” in the K-Map for every term in the expression. The positions not filled with a “1” are assumed to have a “0”.

POS Expressions

Place a “0” in the K-Map for every term in the expression. The positions not filled with a “0” are assumed to have a “1”.

		XY			
	Z	00	01	11	10
	0	0	0		0
	1	0			

POS K-Map

		XY			
	Z	00	01	11	10
	0			1	
	1		1	1	1

SOP K-Map

Locating Terms for Sum of Product Expressions

In SOP, each term is to be represented by a “1” in the K-map.

Where should that term be put?

Apply the “SOP Copy Rule” to each of the terms.

1. Write the variables in a standard uniform order.
2. Replace each complemented variable with a 0
Replace each plain variable with a 1.

Consider $F(X, Y, Z)$

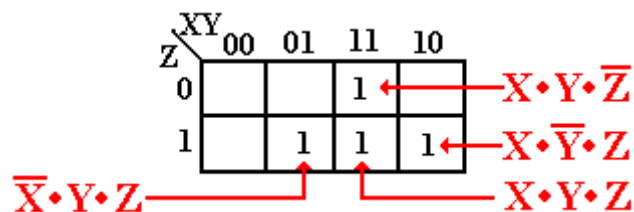
the term $\bar{X} \cdot \bar{Y} \cdot \bar{Z}$ corresponds to 0 0 0

the term $\bar{X} \cdot Y \cdot Z$ corresponds to 0 1 1

the term $X \cdot \bar{Y} \cdot Z$ corresponds to 1 0 1

the term $X \cdot Y \cdot \bar{Z}$ corresponds to 1 1 0

the term $X \cdot Y \cdot Z$ corresponds to 1 1 1



Locating the SOP Terms (Part 2)

Let us examine the SOP expression

$$F(X, Y, Z) = \bar{X} \cdot Y \cdot Z + X \cdot \bar{Y} \cdot Z + X \cdot Y \cdot \bar{Z} + X \cdot Y \cdot Z$$

We place four “1”, one for each of the terms.

For $\bar{X} \cdot Y \cdot Z$, the “1” is placed at $X = 0, Y = 1, Z = 1$.

For $X \cdot \bar{Y} \cdot Z$, the “1” is placed at $X = 1, Y = 0, Z = 1$.

For $X \cdot Y \cdot \bar{Z}$, the “1” is placed at $X = 1, Y = 1, Z = 0$.

For $X \cdot Y \cdot Z$, the “1” is placed at $X = 1, Y = 1, Z = 1$.

		XY				
		00	01	11	10	
Z	0			1		$X \cdot Y \cdot \bar{Z}$
	1		1	1	1	$X \cdot \bar{Y} \cdot Z$

$\bar{X} \cdot Y \cdot Z$ (points to cell 01, 1) $X \cdot Y \cdot Z$ (points to cell 11, 1)

Combining the SOP Terms

Terms are combined into rectangles containing 2, 4, 8, or 16 squares.

Valid combinations: 2 by 1, 1 by 2
 4 by 1, 2 by 2, 1 by 4
 8 by 1, 4 by 2, 2 by 4, 1 by 8, etc.

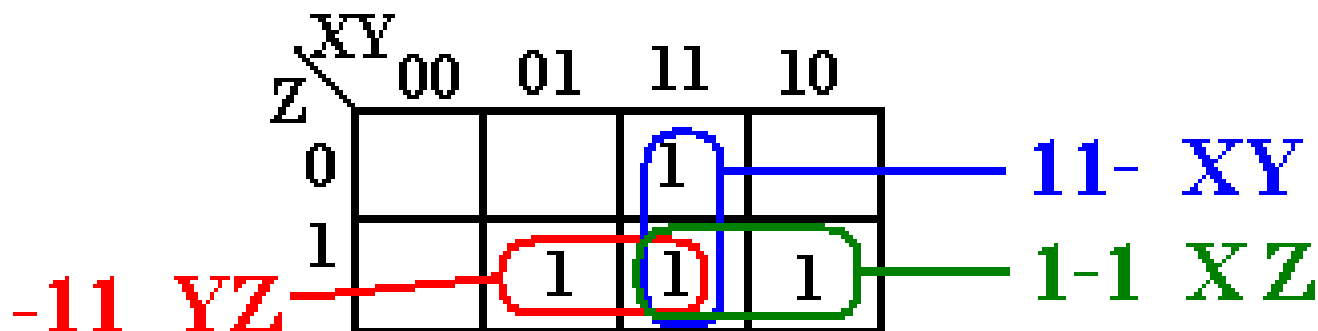
Terms can be combined if the squares are adjacent either vertically or horizontally.

Any term can be a part of more than one combination.

The terms $\bar{X} \cdot Y \cdot Z$ and $X \cdot Y \cdot Z$, represented as 011 and 111, combine to -11 , or $Y \cdot Z$

The terms $X \cdot \bar{Y} \cdot Z$ and $X \cdot Y \cdot Z$, represented as 101 and 111, combine to $1-1$, or $X \cdot Z$

The terms $X \cdot Y \cdot \bar{Z}$ and $X \cdot Y \cdot Z$, represented as 110 and 111, combine to $11-$, or $X \cdot Y$



$F(X, Y, Z) = X \cdot Y + X \cdot Z + Y \cdot Z$. Note 111 is used three times.

Adjacency and Logical Inclusion

Take another look at the above K-Map, with two of the simplifications removed.

		XY			
		00	01	11	10
Z	0			1	
	1		1	1	1

In terms of logical adjacency, the term $X \cdot Y$ covers the term $X \cdot Y \cdot Z$.

In terms of standard expressions, the term $X \cdot Y \cdot Z$ includes the term $X \cdot Y$.

In algebraic terms, the term $X \cdot Y$ stands for the sum term $X \cdot Y \cdot \bar{Z} + X \cdot Y \cdot Z$, so that $X \cdot Y + X \cdot Y \cdot Z$ is the same as $X \cdot Y \cdot \bar{Z} + X \cdot Y \cdot Z + X \cdot Y \cdot Z$, which is obviously the same as $X \cdot Y \cdot \bar{Z} + X \cdot Y \cdot Z$. The last is equivalent to $X \cdot Y$.

Simplification Using Algebra Alone

The straight algebraic simplification illustrates the K-Map approach.

It uses two basic Boolean identities: for any X , $X + X = X$.

for any X , $\bar{X} + X = 1$.

$$\text{Given } F(X, Y, Z) = \bar{X} \cdot Y \cdot Z + X \cdot \bar{Y} \cdot Z + X \cdot Y \cdot \bar{Z} + X \cdot Y \cdot Z$$

$$\begin{aligned} \text{This is } F(X, Y, Z) &= \bar{X} \cdot Y \cdot Z + X \cdot Y \cdot Z \\ &+ X \cdot \bar{Y} \cdot Z + X \cdot Y \cdot Z \\ &+ X \cdot Y \cdot \bar{Z} + X \cdot Y \cdot Z \end{aligned}$$

$$\begin{aligned} \text{This is } F(X, Y, Z) &= (\bar{X} + X) \cdot Y \cdot Z \\ &+ X \cdot (\bar{Y} + Y) \cdot Z \\ &+ X \cdot Y \cdot (\bar{Z} + Z) \end{aligned}$$

$$\begin{aligned} \text{This is } F(X, Y, Z) &= Y \cdot Z + X \cdot Z + X \cdot Y, \\ \text{or } F(X, Y, Z) &= X \cdot Y + X \cdot Z + Y \cdot Z. \end{aligned}$$

Locating Terms for Product of Sums Expressions

In POS, each term is to be represented by a “0” in the K–map.

Where should that term be put?

Apply the “POS Copy Rule” to each of the terms.

1. Write the variables in a standard uniform order.
2. Replace each complemented variable with a 1
Replace each plain variable with a 0.

Consider $F(X, Y, Z)$

the term $X+Y+Z$	corresponds to	0 0 0
the term $X + Y + \bar{Z}$	corresponds to	0 0 1
the term $X + \bar{Y} + Z$	corresponds to	0 1 0
the term $\bar{X} + Y + Z$	corresponds to	1 0 0
the term $\bar{X} + \bar{Y} + \bar{Z}$	corresponds to	1 1 1

	$Z \backslash XY$	00	01	11	10	
$X+Y+Z$	0	0		0		$X+\bar{Y}+Z$
$X+Y+\bar{Z}$	1	0				$\bar{X}+Y+Z$

Locating the POS Terms (Part 2)

Let us examine the POS expression

$$F(X, Y, Z) = (X + Y + Z) \cdot (X + Y + \bar{Z}) \cdot (X + \bar{Y} + Z) \cdot (\bar{X} + Y + Z)$$

We place four “1”, one for each of the terms.

For $(X + Y + Z)$, the “0” is placed at $X = 0, Y = 0, Z = 0$.

For $(X + Y + \bar{Z})$, the “0” is placed at $X = 0, Y = 0, Z = 1$.

For $(X + \bar{Y} + Z)$, the “0” is placed at $X = 0, Y = 1, Z = 0$.

For $(\bar{X} + Y + Z)$, the “0” is placed at $X = 1, Y = 0, Z = 0$.

		XY			
		00	01	11	10
Z	0	0	0		0
	1	0			

$X + Y + Z$ — points to cell (0,0,0)
 $X + Y + \bar{Z}$ — points to cell (0,0,1)
 $X + \bar{Y} + Z$ — points to cell (0,1,0)
 $\bar{X} + Y + Z$ — points to cell (1,0,0)

Example K-Map on 3 Variables

Consider the K-map shown above. It can be shown to represent the Boolean expression $F(X, Y, Z) = (X + Y + Z) \cdot (X + \bar{Y} + Z) \cdot (\bar{X} + Y + Z) \cdot (X + Y + \bar{Z})$.

Z \ XY	00	01	11	10
0	0	0		0
1	0			

This is a K-map representation of a POS (Product of Sums) expression, also called an expression in Conjunctive Normal Form.

The K-map procedure works by grouping adjacent squares. Here are the adjacencies:

000 and 010 group to form 0-0 representing the term $(X + Z)$

100 and 000 group to form -00 representing the term $(Y + Z)$

000 and 001 group to form 00- representing the term $(X + Y)$

The expression is $F(X, Y, Z) = (X + Y) \cdot (X + Z) \cdot (Y + Z)$

Handling Non-Canonical Terms

Suppose we have $F(X, Y, Z) = (X + Y)(X + \bar{Y} + Z)(\bar{X} + Y + Z)$

How do we handle this?

For $(X + \bar{Y} + Z)$, denoted as **010**, the “0” is placed at $X = 0, Y = 1, Z = 0$.

For $(\bar{X} + Y + Z)$, denoted as **100**, the “0” is placed at $X = 1, Y = 0, Z = 0$.

But what about $(X + Y)$, which is denoted as **00-?**

The answer is to view the term as equivalent to the two terms

$(X + Y + Z)$, denoted as **000**

and $(X + Y + \bar{Z})$, denoted as **001**.

For the term $(X + Y)$ we would fill boxes 000 and 001 each with a “0”.

Another Simplification for SOP

Consider the Boolean Expression $F(X, Y, Z) = \Sigma(3, 5, 6, 7)$.

The use of a K-Map to simplify this is simple.

First replace the decimal numbers by binary.

We have $\Sigma(011, 101, 110, 111)$. Each of the numbers marks a cell in the K-Map.

		XY			
		00	01	11	10
Z	0			1	
	1		1	1	1

We have seen and solved this before, so we move on.

Still Another Try for SOP

Here is another function to consider. $F(X, Y, Z) = \Sigma(1, 2, 4, 7)$.

This can be written as $\Sigma(001, 010, 100, 111)$.

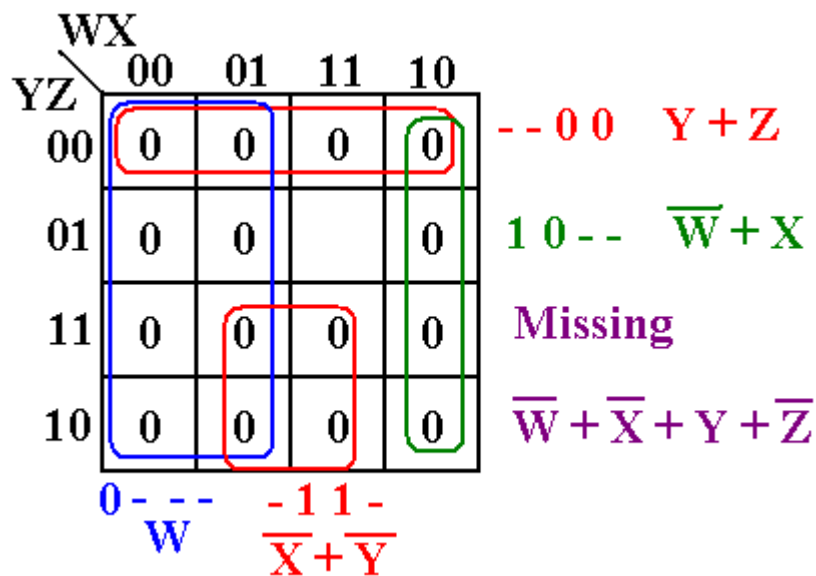
Here is the K-Map.

		XY			
		00	01	11	10
Z	0		1		1
	1	1		1	

There are no logical adjacencies. The function cannot be simplified.

A K-Map for CNF Satisfiability

Consider the Boolean expression $B_3 = (\bar{W} + X) \cdot W \cdot (\bar{X} + \bar{Y}) \cdot (Y + Z)$.
Is this satisfiable? Here is the K-Map generated for this expression.



Note that there is one square in the K-map that is not covered by these expressions. This corresponds to the term $(\bar{W} + \bar{X} + Y + \bar{Z})$. The assignments that make this missing term equal 0 are $W = 1$, $X = 1$, $Y = 0$, and $Z = 1$.

For these values, $B_3 = (\bar{1} + 1) \cdot 1 \cdot (\bar{1} + \bar{0}) \cdot (0 + 1) = 1$.
 $= (0 + 1) \cdot 1 \cdot (0 + 1) \cdot (0 + 1) = 1$. B_3 is satisfiable.

An Obvious Comment on the Above

Consider again the expression $B_3 = (\bar{W} + X) \bullet W \bullet (\bar{X} + \bar{Y}) \bullet (Y + Z)$.

It should be obvious that if $W = 0$, then $B_3 = 0$.

The only way to get $B_3 = 1$ demands the assignment $W = 1$.

$$\begin{aligned}
 \text{Assuming } W = 1, \text{ we have } B_3 &= (\bar{1} + X) \bullet 1 \bullet (\bar{X} + \bar{Y}) \bullet (Y + Z) \\
 &= (0 + X) \bullet 1 \bullet (\bar{X} + \bar{Y}) \bullet (Y + Z) \\
 &= X \bullet 1 \bullet (\bar{X} + \bar{Y}) \bullet (Y + Z) \\
 &= X \bullet (\bar{X} + \bar{Y}) \bullet (Y + Z) \\
 &= (X \bullet \bar{X} + X \bullet \bar{Y}) \bullet (Y + Z) \\
 &= (0 + X \bullet \bar{Y}) \bullet (Y + Z) \\
 &= X \bullet \bar{Y} \bullet (Y + Z) \\
 &= X \bullet \bar{Y} \bullet Y + X \bullet \bar{Y} \bullet Z \\
 &= X \bullet 0 + X \bullet \bar{Y} \bullet Z \\
 &= X \bullet \bar{Y} \bullet Z
 \end{aligned}$$

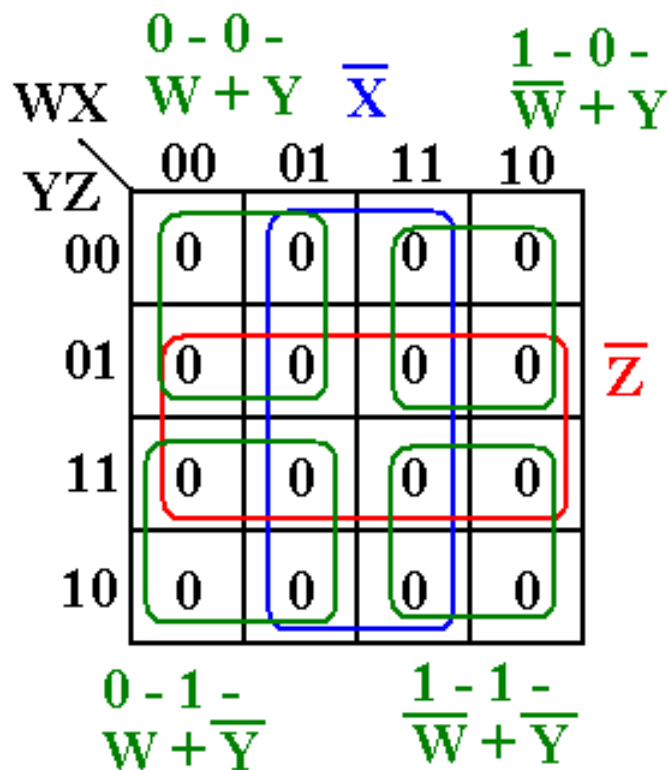
We now have $W = 1$, $X = 1$, $Y = 0$, and $Z = 1$.

Another K-Map for SAT

Consider the Boolean expression

$$B_4 = (\overline{W} + \overline{Y}) \cdot (\overline{W} + Y) \cdot (W + \overline{Y}) \cdot (W + Y) \cdot \overline{X} \cdot \overline{Z}$$

Here is the K-Map generated for this expression.



As a K-Map, this simplifies to $B_4(W, X, Y, Z) = 0$; it is not satisfiable.

More Comments on the Second Example

Note again the expression $B_4 = (\overline{W} + \overline{Y}) \cdot (\overline{W} + Y) \cdot (W + \overline{Y}) \cdot (W + Y) \cdot \overline{X} \cdot \overline{Z}$.

This can be true only if $X = 0$ and $Z = 0$.

Assuming that $X = 0$ and $Z = 0$, we have $B_4 = (\overline{W} + \overline{Y}) \cdot (\overline{W} + Y) \cdot (W + \overline{Y}) \cdot (W + Y)$.

This is a Canonical POS expression on the two variables W and Y .

It has $4 = 2^2$ terms; it cannot be satisfiable.

(In general terms, it has 2^N terms for N variables.)

Here is my version of the result on Circuit Satisfiability

Lemma: Let B be a CNF Boolean expression over N Boolean variables.
 B is satisfiable if and only if it cannot be minimized
by any valid simplification method to $B = 0$ identically.