

## TutorTube: Karnaugh Maps

Spring 2020

### Introduction

Hi, my name is Jeff, and I'll be going over how to use the Karnaugh Maps.

### Using Karnaugh Maps

Karnaugh Maps are a visual tool that help us derive Boolean expressions of 4 or less variables from a truth table. Normally, we would use Boolean theorems to simplify expressions. K-Maps however, provide a visual alternative to reducing canonical Boolean expressions. Even though it won't seem like we're using any Boolean theorems, under the hood of the K-Map we're using the Boolean theorem:  $AB + AB' = A$ .

Let's say we have a truth table for an unknown function. This table shows what my output will be for every combination of A, B, and C. This m column I added represents the decimal equivalent of my binary input. It's usually not given in truth tables, but if you're struggling with binary, I recommend adding it to your tables because it'll be useful here in a second, and it's good practice to get familiar with binary numbers.

A	B	C	m	F(A,B,C)
0	0	0	0	1
0	0	1	1	0
0	1	0	2	1
0	1	1	3	1
1	0	0	4	0
1	0	1	5	0
1	1	0	6	1
1	1	1	7	1

In order to use the K-Map, we need the minterms that are associated with it. Remember that a minterm is any set of inputs that give us a 1 as our output. We can see from the truth table that when we receive a binary 0,2,3,6, or 7 we get a one as the output, so our minterms for this function are:

$$\mathbf{m(2,3,6,7) = m_0 + m_2 + m_3 + m_6 + m_7}$$

Now with our minterms found, let's set up our K-Map. I have a 2x4 table here in white, and in grey is where I'll be putting my labels. I'm going to label A and B along the columns and C along the rows but, this is arbitrary and you can flip them if you want to. I'm now going to label my values of A, B, and C in gray code order. Gray Code is special because each number is only one bit different than the next. The K-Map has to be set up this way so we can utilize our Boolean theorem before.

Let's make sure to know what these numbers mean. Along the side, each row represents a different value of C. The first row represents C = 0 and the second row represents C = 1.

Along the columns, it's the same way, just with 2 variables. Remember here in the corner we labeled this top section as AB. That means that this first row represents A= 0 and B = 0. The next column represents A = 0 and B = 1. After that, they're both equal to 1, and in the last column, A = 1 and B = 0.

Next I'm going to label in the corner of each block its minterm equivalent. This isn't a required step, but it will help you keep track of everything. You can think

of AB and C as coordinates that tell you which minterm you are looking at. Now because my function has minterms 0, 2, 3, 6, and 7, I'm going to put a 1 in the cells that correspond to those minterms, and put a 0 everywhere else. The K-map is done so take a moment to pause and make sure you're comfortable so far with our setup.

	AB= 00	AB= 01	AB= 11	AB= 10
C=0	1	1	1	0
C = 1	0	1	1	0

Our K-Map is now complete, so now we can use it to simplify our expression from before. To do this I'm going to create groups of minterms. In general, the rules for grouping are: You can only group in powers of 2 (2, 4, 8, etc.), the groups must be made of cells adjacent to each other: I can't group diagonally. More specifically, this means I can only group cells that are 1 bit apart from each other. It's also okay if groups overlap with each other, as long as every minterm is included in at least 1 group.

You should try to always make the largest groups possible, because this will result in a more simplified expression. With this square in the middle, I could create two groups out of it, and my function at the end would be valid, but if I make a group of 4 instead, I'll get a smaller expression at the end, so I'll do that here. But this 1 on the left is still left out, so I'll group it in here. Now I have every minterm included in a group, so now we can move on to getting my Boolean expression from the groups.

Looking at my group of 4, I'm going to look for which variables do not change within the group. Those variables will be the ones that make up my term. I can see from this group that A changes from 0 to 1 and C also changes from 0 to 1. B is 1 for the whole group, so B is the only variable not changing. That means my term that represents this group is just going to be B. For the other group of 2, I'll do the same thing. I see that A and C remain a 0 between these two minterms, but B changes from a 0 to a 1. So the B is cancelled out, but this time since A and C are both 0 instead of 1, they become negated, so this term becomes A'C'. Now with our two terms, we just OR them together, to put them in Sum of Products form. Now we get our final equation:  $B + A'C'$

## K-Map Example

Let's see a bigger example where some more interesting things can happen. Let's say I'm given:  $f(A, B, C, D) = m(0, 2, 5, 7, 8, 10, 13, 15)$

This time I'll need a 4x4 K-Map, with two variables along each side. Again, I'll label what each minterm each cell corresponds too, and then place a 1 where my minterms are.

	AB= 00	AB= 01	AB= 11	AB= 10
CD= 00	1			1
CD= 01		1	1	
CD= 11		1	1	
CD= 10	1			1

The next step is to make my groups of minterms. You should try to always make the largest groups possible, because this will result in a more simplified expression. With this square in the middle, I could create two groups out of it, and my function at the end would be valid, but if I make a group of 4 instead, I'll get a smaller expression at the end, so I'll do that here.

Now, for the four ones in the corners, because the map is set up in gray code, the corner cells are all only bit different than each other. So that means my groupings are allowed to loop around the edges of the map like this and can make this group of four. Now every minterm has been included so I can go ahead and derive my expression.

Starting with the group in the middle, looking across the middle two columns I see that A changes from 0 to 1, while B stays at 1, so I'll put down B in my first term. Looking at the middle rows I see C changes from 0 to 1 while D stays the same, so I'll multiply the B I got with D.

Now for my second group, I'll look in the top corners and see that A changes from 0 to 1, but B stays at 0. Because it's a zero, I'm going to start my next term with B'. Next, looking at the upper left and lower left corners, I see that C changes from 0 to 1, but D remains at 0, so I'll multiply a D' onto my term.

Now I'll combine my terms in Sum of Product Form and get  $F(A, B, C, D) = BD + B'D'$

## Outro

I hope this video helps get you started on how to use the Karnaugh Map to derive Boolean expressions. If you have more questions, feel free to visit [learningcenter.unt.edu](http://learningcenter.unt.edu) to see what other resources we can offer you! See you next time!