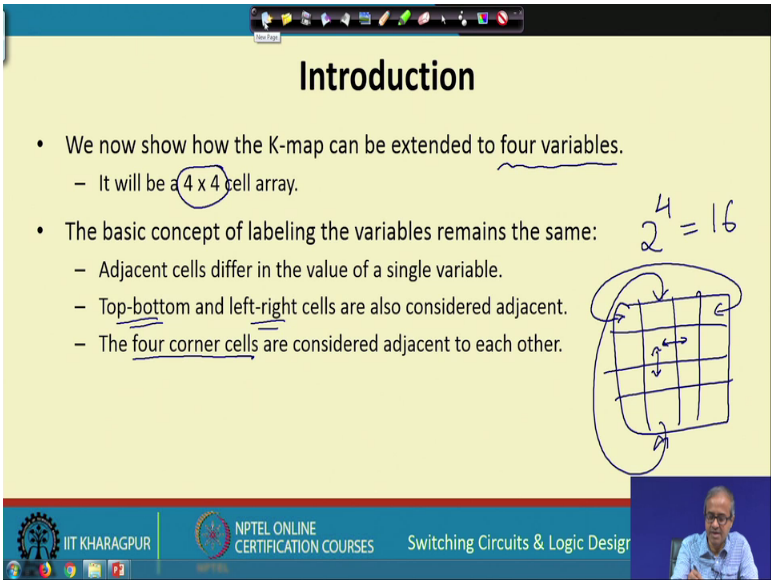
**Switching Circuits and Logic Design**
**Prof. Indranil Sengupta**
**Department of Computer Science and Engineering**
**Indian institute of Technology, Kharagpur**

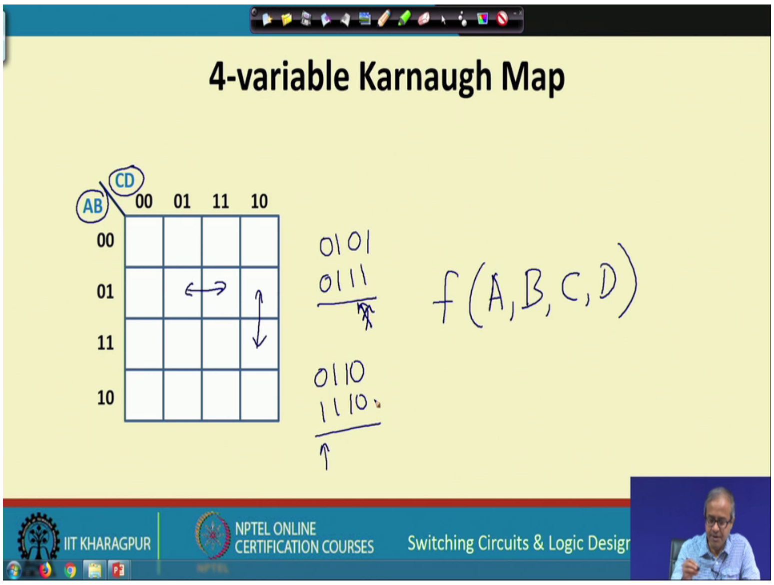
**Lecture – 17**
**Minimization Using Karnaugh Maps (Part- II)**

So, in the last lecture if we recall we are talked about the concept of Karnaugh map method of minimizing a function and through examples we showed how we can handle or minimize 3 variable functions.

(Refer Slide Time: 00:35)



(Refer Slide Time: 00:36)




So, in this lecture we shall extending our discussion to handle 4 variable functions so, we shall be extending our discussion to 4 variables. Now, recall we mentioned that it is not so easy to extend Karnaugh map concept to larger number of variables. Of course, in books we will see that the authors have said that you can do it up to 6, but up to 4 is easy, 5 becomes difficult, 6 becomes quite difficult beyond 6 is impossible. So, whatever examples I showed here will be only up to 4.

So, if you have 4 variables then how many minterms are possible 2 to power 4 equal to 16. So, first thing is that you need to have 16 cells and they will organized in a 4 by 4 cell array. So, there will be 4 rows and 4 columns, 4 rows and 4 columns. The basic concept of labeling remains the same, you will see that the way you label the rows and columns it will always be ensured the neighboring cells will always be deferring in a single variable.

This will and also there will a neighborhood relationship between the right and left and also top and bottom. Top-bottom and left-right adjacency will also be provided or ensured. Not only that, you will see that the four corner cells will be considered are adjacent to each other. Why? We shall see just a little later.

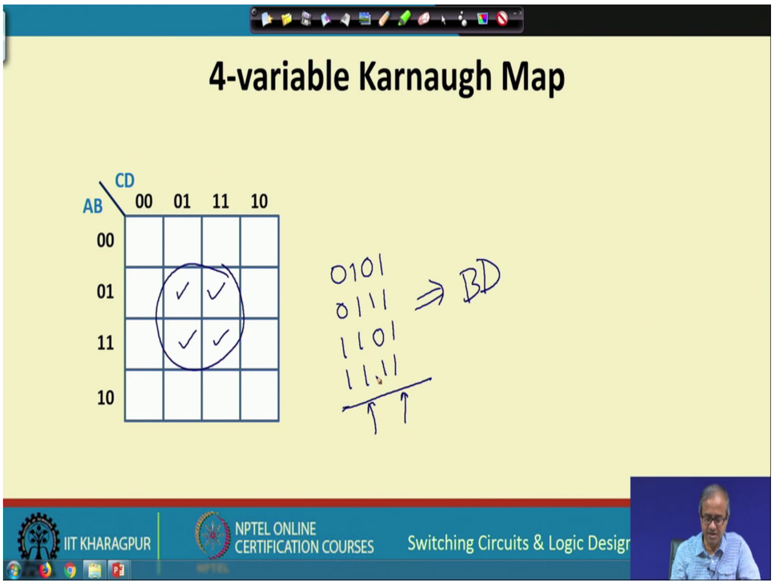
Let us consider a 4 variable Karnaugh map like this; this is just an extension of a 3 variable map where we consider a function with 4 variable let us say A, B, C, D where in this diagram I am showing AB along the rows and CD along the columns. And just like a

3 variable function the way we label the columns in the gray code order 0 0 0 1 1 1 1 0 here we are doing the same thing for both rows and columns 0 0 0 1 1 1 1 0. Why?

You consider any two cells let us say these two cells the first cell corresponds to 0 1 0 1 0 1 0 1, the second cell to corresponds to 0 1 1 1 0 1 1 1. We check they defer in a single variable this last position sorry second last position 1 0 1 it defers in the second last position right.

Now, you take any pair of cells vertically take this and this the upper cell corresponds to 0 1 1 0, lower cells corresponds to 1 1 1 0. You see this are all adjacent only the first variables is different, the other 3 are the same right.

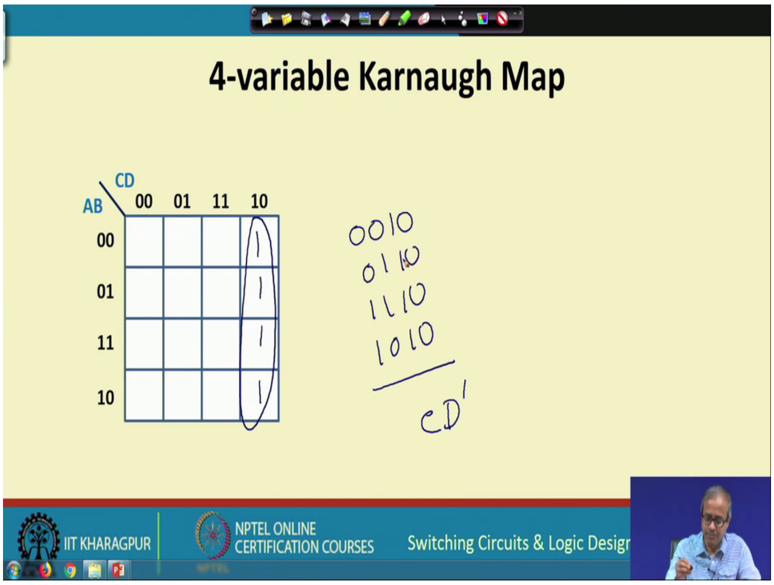
(Refer Slide Time: 03:54)




The same thing you can extend to a cube of size 4, let us consider these 4 this 4 corresponds to I am just writing on the binary 0 1 0 1 then 0 1 1 1 then 1 1 0 1 and 1 1 1 1.

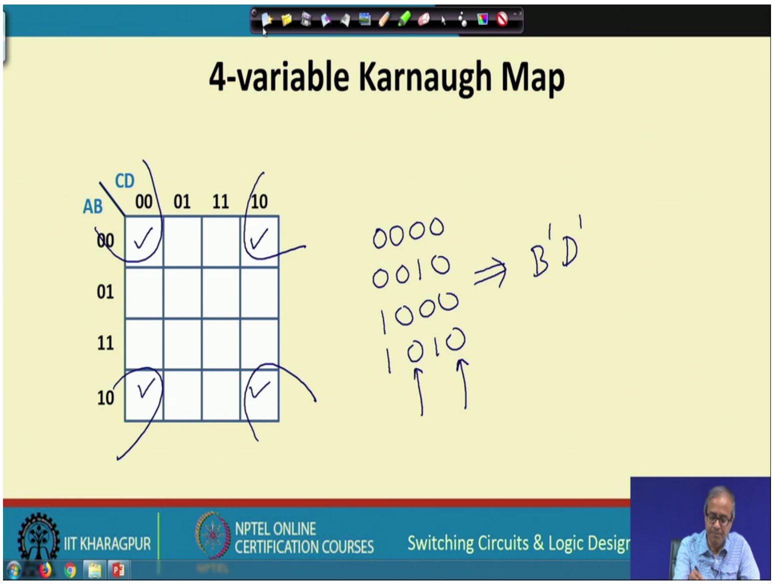
So, you see here the second and fourth variables are the same 1 this is also 1, but the first and the third variables which is A or C they are changing. So, if you form a cube out of this 4 this will be corresponding to B and D minimized form will be only BD, A and C will they will be cancelled out.

(Refer Slide Time: 04:47)




Similar, is the case if you take a cube vertically like this they will correspond to 0 0 1 0 0 1 1 0 1 1 1 0 1 0 1 0. So, again this 1 0 is common the first two are changing they will cancelled out. So, C and D will remain it will be CD bar 1 0 right.

(Refer Slide Time: 05:16)

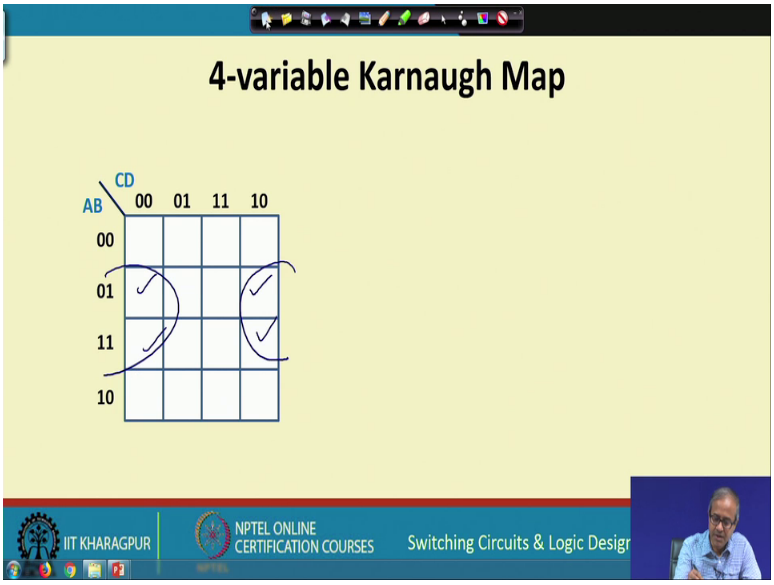



Now, I talked about the 4 corner cells rights suppose, there are 2 minterms in the 4 corner cells they are also considered neighbors. So, I can make a big cube of size 4 using the 4 corner cells which is showed like this, how let us write down and see.

This cells corresponds 0 0 0 0, this cell corresponds to 0 0 1 0. The left one corresponds to 1 0 0 0 and this one is 1 0 1 0, you see last variable is always 0, second variable is also 0, the first and third are changing.

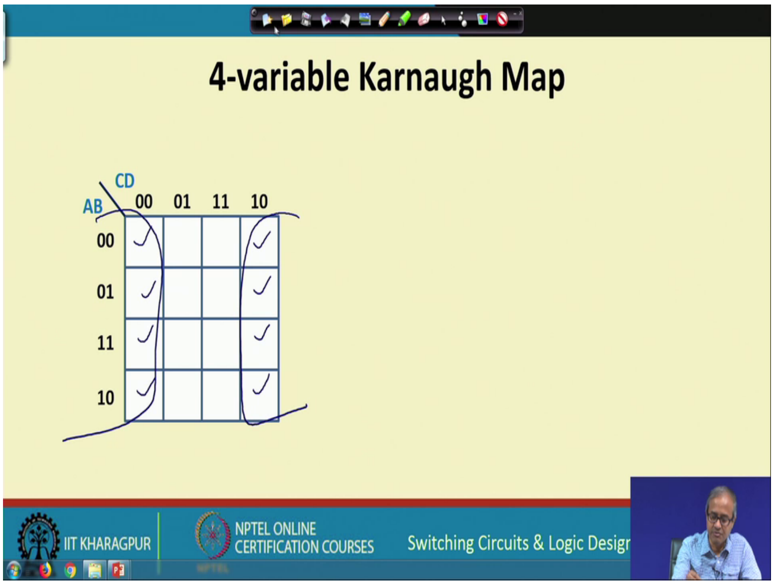
So, same way first and third will cancel out and this will be equivalent to B bar D bar right. So, the same concept you can make cubes as you wish.

(Refer Slide Time: 06:08)



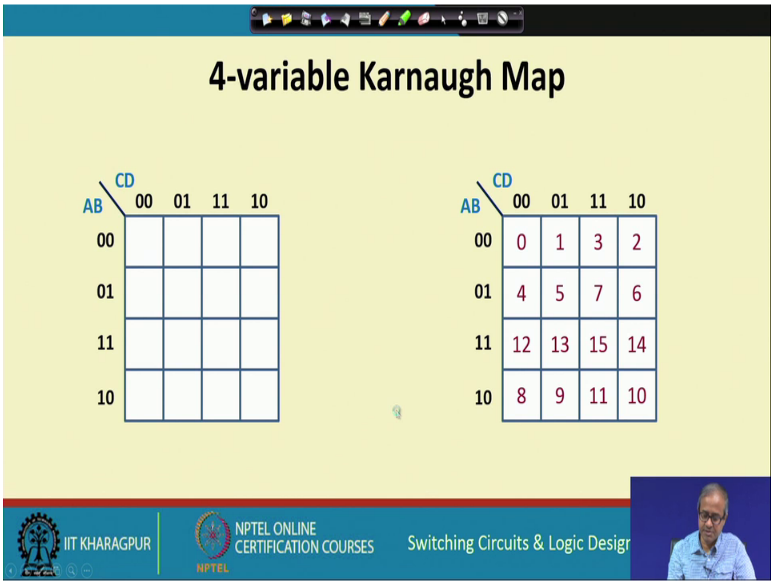

If you have this, this and this, this you can makes cube like this right.

(Refer Slide Time: 06:15)



If you have this 4 and also this 4, then you can make a cube of size 8 like this; same concept you can use right.

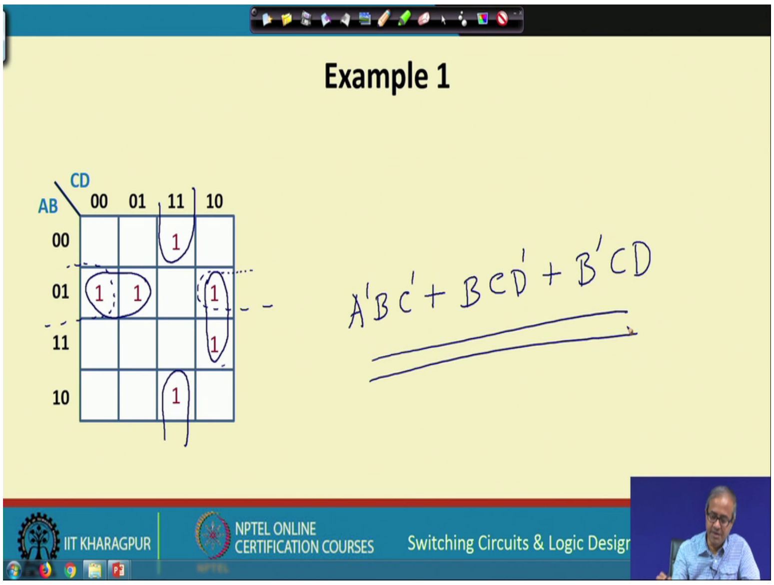
(Refer Slide Time: 06:28)




So, this is the same Karnaugh map where I am showing the cells labeled by the decimal equivalent of AB CD values. So, numbers are made like this 0 1 2 3 4 5 6 7 and you skip to the last one 8 9 10 11 then 12 13 14 15. This is because this 11 is coming first before 10 right similarly in the columns these are reverse because 11 is coming first then 10 all right.

Now, let us take some examples.

(Refer Slide Time: 07:01)

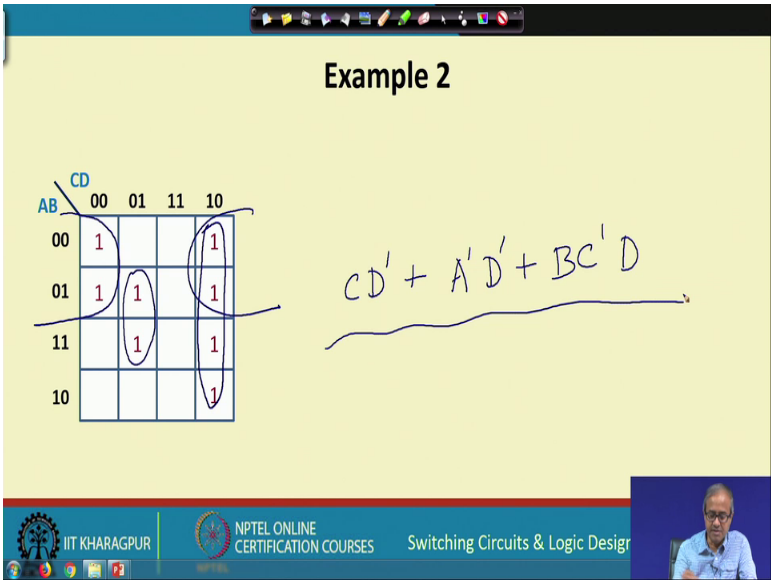



Let us take function like this where you can see that there are 6 true minterms. Now, if I follow a rule let us try to get the cubes constructed, this can be 1 cube I cannot make it any larger. This is 1 cube, these two 1's they are adjacent this is 1 cube. Now, one thing you see this is also a cube I am showing it dotted, this and this you can also make a cube out of this.

But this is not required because I have already covered all the true minterms with this 3 cubes so, I do not require these this cube any more ok. This will be my best solution, minimum number of cubes which is covering all of them. But if I consider this dotted cube also it will not make it any minimum, it will be even one more. But here what is the solution, this 2 cubes 0 1 which is A bar B and 00 01 D cancels out, C is 0 C bar then let us consider is one vertically

So, 01 and 11 A cancels out B is 1 it is B and it is 10. So, C D bar and the last one this top and bottom 00 and 10 A cancels out B is 0 B bar and 11 CD. This is the minimized form of this function right, let us take another example.

(Refer Slide Time: 09:05)



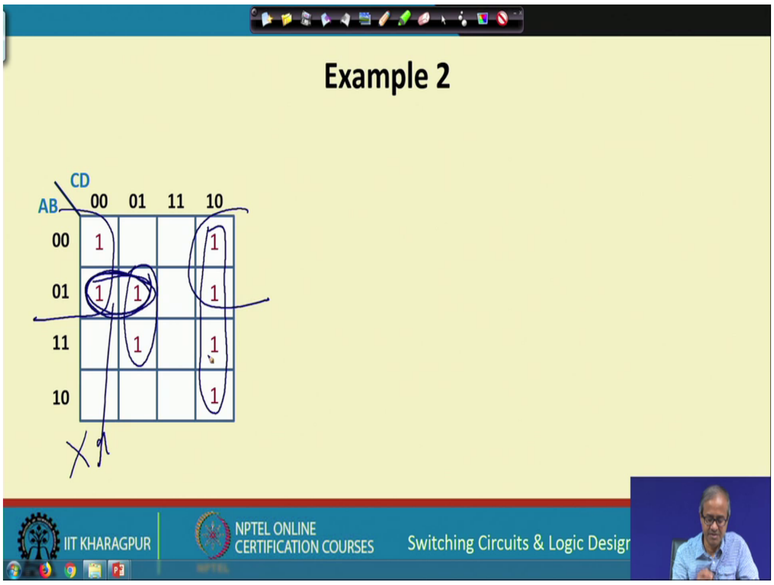

Let us take an example like this well some of the largest cubes immediately see is one is this cube of size 4 that is another size 4 you can see which will be these two plus these two.

You see you can take these two separately, but this will not the biggest one because you can also take these two, this will be the bigger one and these two 1's are still remaining. You can of course, take a cube like this, but again this will left out, but instead you take cube like this and this will cover everything I do not need any further cube.

So, what will be the expression this column this A and B are cancelling out, AB will not be there 1 0. So, just CD bar plus these two these two 4 00 01 which is A bar and 00 and 10 C cancels out D bar and lastly you have this two 01 11 means B and 01 C bar D, this is your minimized form.

So, once you do it like this you cannot minimize it any further just you think that a I mentioned tell me tell it once more.

(Refer Slide Time: 10:41)



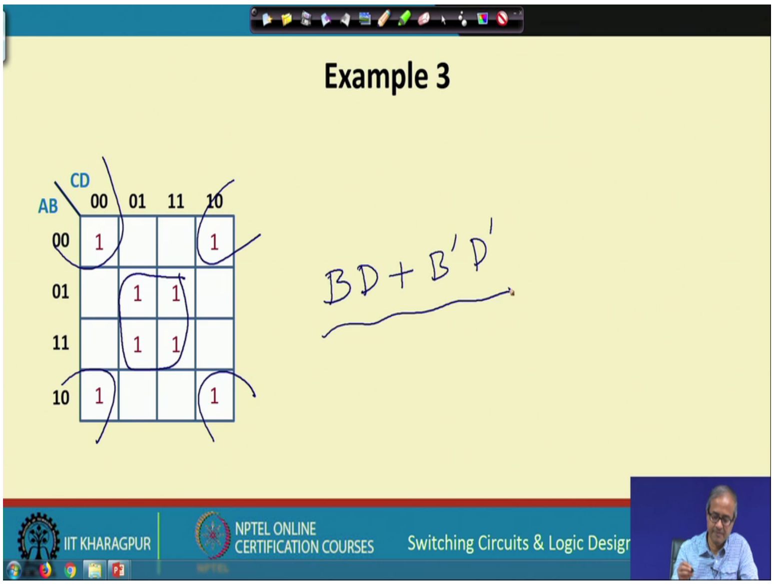

Suppose we had constructed a cube like this, we had constructed this cube also like this biggest possible, but suppose I also have constructed a cube like this then this cube will also this 1 is also remaining left out.

So, I I will also have to include it so you see I need one extra cube here and this cube this cube these two this is redundant because this cells that this cube is covering, these cells are already covered by some other cubes. So, a particular cell a true minterm need to be covered by at least one of the cubes.

So, if I have 1 cube whose cells are already covered by some other cubes I do not need to include that cube at all you can leave it out; that is why I can remove this cube from my map and only 1 2 and 3 this cubes will do ok. This is what we are done.

Take another example.

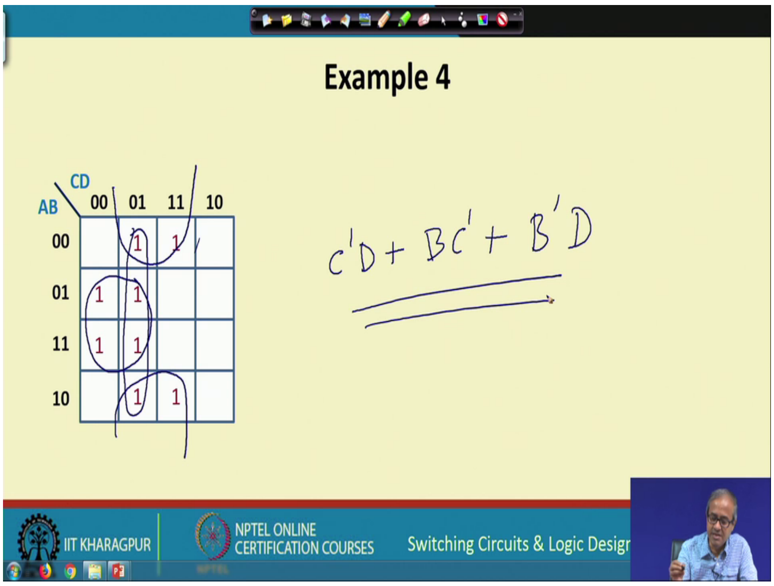
(Refer Slide Time: 11:58)




This is the very nice example regular example which shows four 1's in the corners. This is very easy, the middle four will be 1 cube and the four corners will be (Refer Time: 12:14), this will be the another cube.

So, what will be the expression? The four middle one will be 0 1 1 1 it will B 0 1 1 1 again D or the four corner 1's 0 0 1 0 B bar 0 0 1 0 D bar. This will be function BD or B bar D bar right.

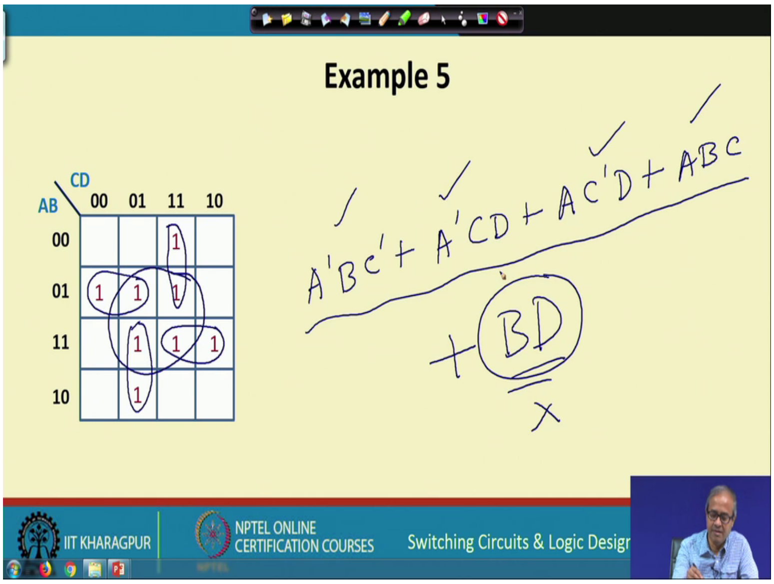
(Refer Slide Time: 12:48)




Let us look at some more examples. Take an example like this.

So, here again let us try to make the cube as large as possible like one I can see this four I can make, this four I can make, these four I can make. Now, you see these two 1's are have been left out.

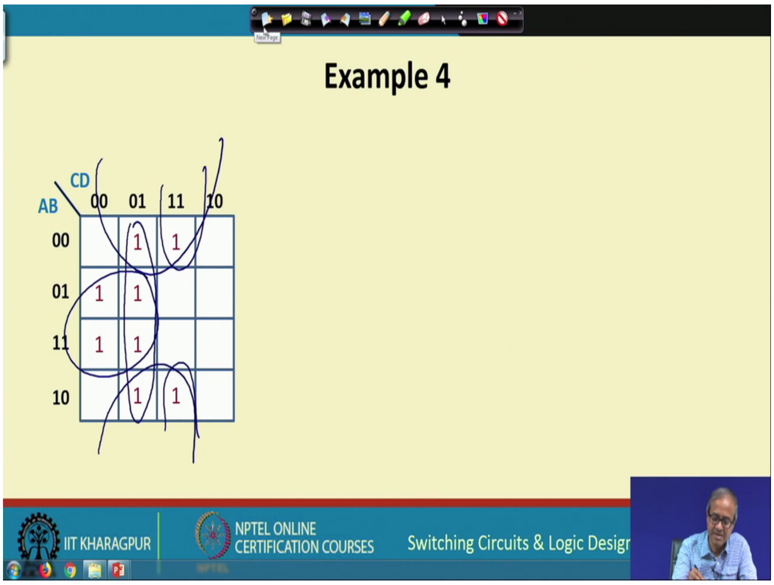
(Refer Slide Time: 13:02)




So, I can make these two together or I can make the cube of four these two plus these two right, this is the biggest.

So, this will be my best cover. So, if we do it this way this long one A B are cancelled it will C bar D plus this q 1 D left; it will be B and C bar. And these two 00 and 10 it will be B bar and 0 1 1 1 C cancels out D this will be the minimized form right.

(Refer Slide Time: 14:05)




So, when we choose this cells it should be judicious, you should not select something like this I select this, I select this then I select this, this is not minimum because you are then we get bigger I can these two and these two make it cube of size 4 right.

So, I will try here I shall try to make it as large as possible the cubes ok. This is another cyclic kind of a structure like you see I can find these 4 cubes, but the other four 1's are isolatives. So, I can have 1 cube with these two, 1 cube with these two, 1 cube with these two, 1 cube with these two. So, what does this mean see let me again go back.

So, our first premise or assumption was that we try to make the cube as large as possible ok. The second thing I mentioned that if I find that a cube is covering some cells which are already covered by some other cubes then drop or delete this cube, I do not need this ok. You see in this example the first I am saying is that this is the largest cube, but the other four well I cannot leave them out because each of them having at least 1 or 1 which is not covered by any one; these are not covered by any other cube.

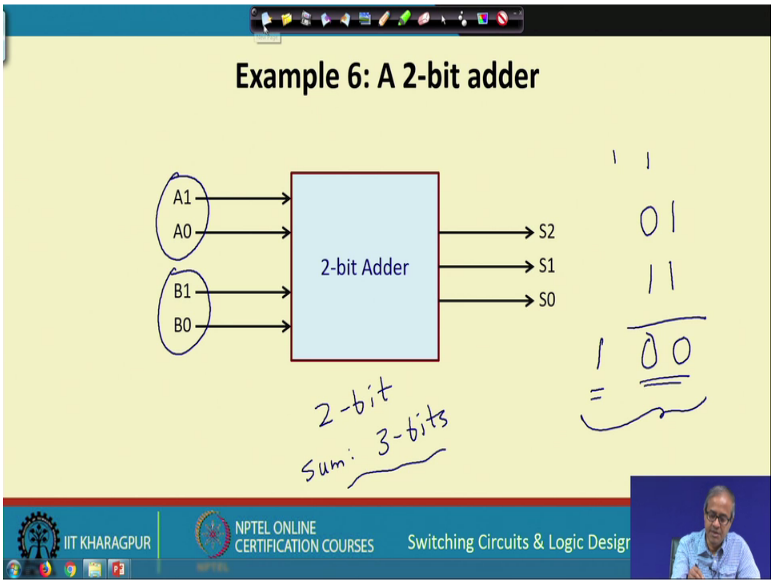
So, when we include this four you see that the bigger cube is no longer required because the four 1's in the bigger cube they are already covered by this 4 smaller cubes. So, even though this is the bigger one I do not need this. So, my cover here is look like this because this middle one will be redundant is not required.

So, what will the expression the left one will be A bar B and 00 01 C bar. This top one 00 01 will be A bar CD plus these two will be 11 10 will be A and 01 is C bar D plus this two 11 is AB and 11 10 is C ok.

This will be my minimized form, but if you also include this middle one then you also include another term which is redundant not required; it will be actually be B and D BD, BD is not required at all. So, you unnecessary you are using five terms.

These four are sufficient, this will covering BD automatically right, this something you should remember.

(Refer Slide Time: 17:19)

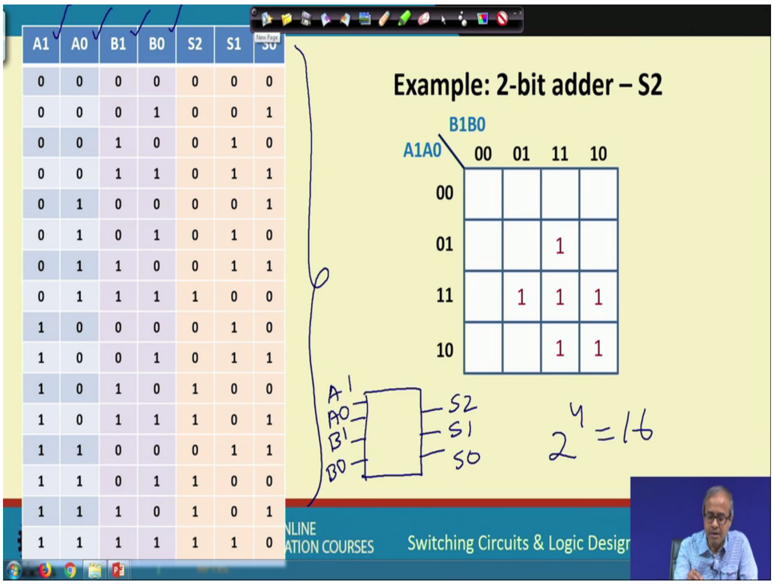



Ok here let us take a slightly bigger and a more realistic example suppose we are trying to design a 2-bit adder. So, how is this 2-bit adder it takes 2 numbers this numbers are 2 bits each and adds them up. Let us say one of the number is 0 1 and the other number is 1 1.

So, if I add them up 1 1 is 0 with the carry of 1 1 0 1 is 0 with the carry of 1. So, the final carry out will be 1. So, my sum will be actually 3-bits 2-bits and the possible carry out. So, the sum will be 3-bits

So, whenever we add two 2 bit numbers your sum will become 2 bits one more right because there is a chance of 1 carry bit coming out right. Let us look into these examples and see how you can use the Karnaugh map for minimization.

(Refer Slide Time: 18:39)



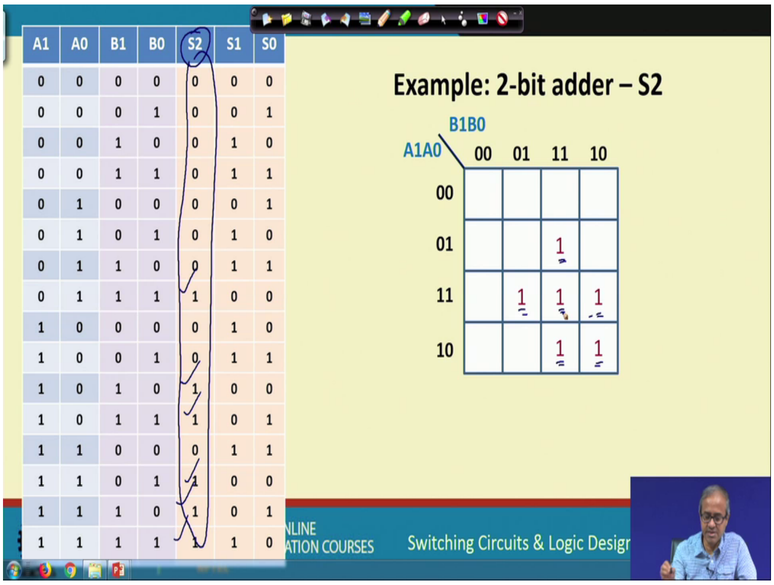

Well here, first let us look at the truth table this is the truth table of the adder.

So, just recall in the adder I have said that there will be 4 inputs; the first number A 1 A 0 then B 1 and B 0. And there will be 3 outputs S 2 S 1 and S0. Let us see in the truth table I am showing this 4 A 1 A 0 B 1 B 0 are the inputs and there are 3 outputs S 2 S 1 S 0.

So, for 4 input there will be 2 to the power 4 or 16 minterms, which I am showing in 4 rows 0 0 0 0 0 0 0 1 up to 1 1 1 1 there are 16. So, if we just add you think in terms of equivalent decimal it will be easier 0 0 plus 0 0 means 0 plus 0 is 0. So, sum is 0 0 0 0 plus 1 sum is 1 is 0 0 1 0 plus 2 1 0 is 2 sum is 2 2 is 0 1 0 0 plus 3 sum is 3.

So, 0 1 1 is 3 0 1 is 1 1 plus 0 is 1 0 0 1 1 plus 1 is 2 0 1 0 is 2 1 plus 2 is 3 0 1 1 1 plus 3 is 4 1 0 0 2 plus 0 is 2 0 1 0 2 plus 1 is 3 2 plus 4 is 4 1 0 0 2 plus 3 is 5 1 0 1 3 plus 0 is 3 3 plus 1 is 4 3 plus 2 is 5 and 3 plus 3 is 6 1 1 0.

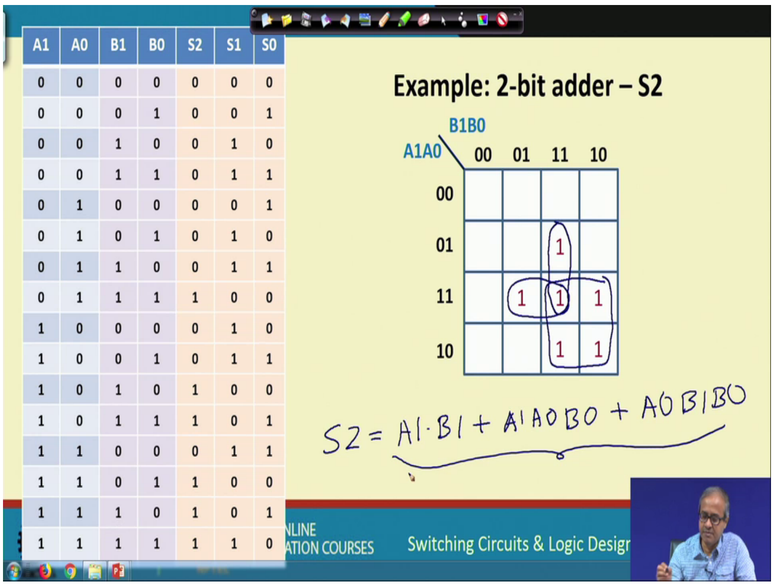
(Refer Slide Time: 21:01)




Now, in this Karnaugh map we are showing the Karnaugh map corresponding to the output is 2; that means, this column of the total. So, how many 1's are there 1 2 3 4 5 and 6?

So, you see there are 6 1's correspond to you see first one correspond to 0 1 1 1. So, 0 1 1 1 this one, second one is 1 0 1 0 1 0 1 0 this one, third one is 1 0 1 1 1 0 1 1 this one, fourth one is 1 1 0 1 1 0 1 1 1 1 0 1 1 1 0 and finally, 1 1 1 1 right. So, these are there.

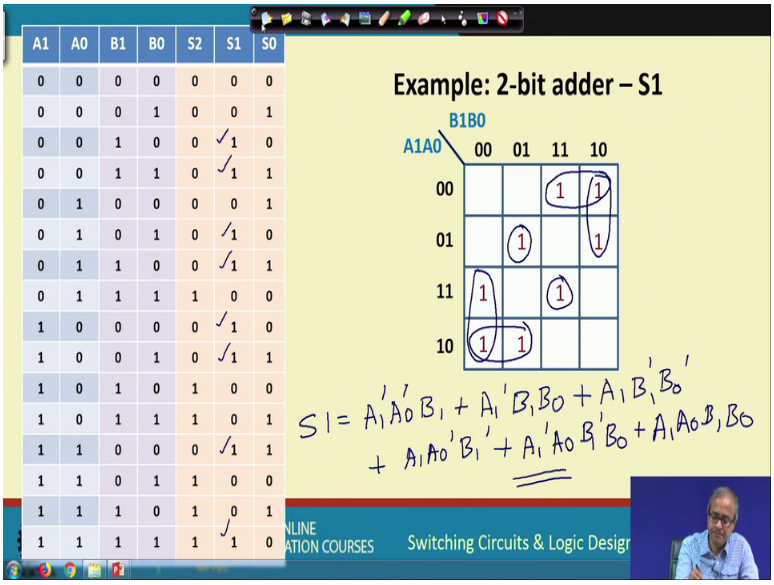
(Refer Slide Time: 21:59)



Now, if you try to form the cubes in this case you see that you will be getting large cube here, you will be getting smaller cube here and another smaller cube here that is all; you cannot minimize any further.

So, what will be the expression for S2? Your S2 will be this bigger one A 0 cancels out only A 1 A 1 and 11 on 10 on B 1 A 1 and B 1 or let us take this one first 11 which is A 1 A 0 and 0 1 1 1; that means, it will B 0; B 0 or this one 0 1 1 1 is A 0 A 0 11 B1 B0. This is the function minimized form, this is the expression for S 2 right ok.

(Refer Slide Time: 23:10)




Let us now, see what will be S 1? Similar is the case for S 1, if we look at S 1 there are 1 2 3 4 5 6 7 8 1’s and if you check the 8 1’s are distributed like this.

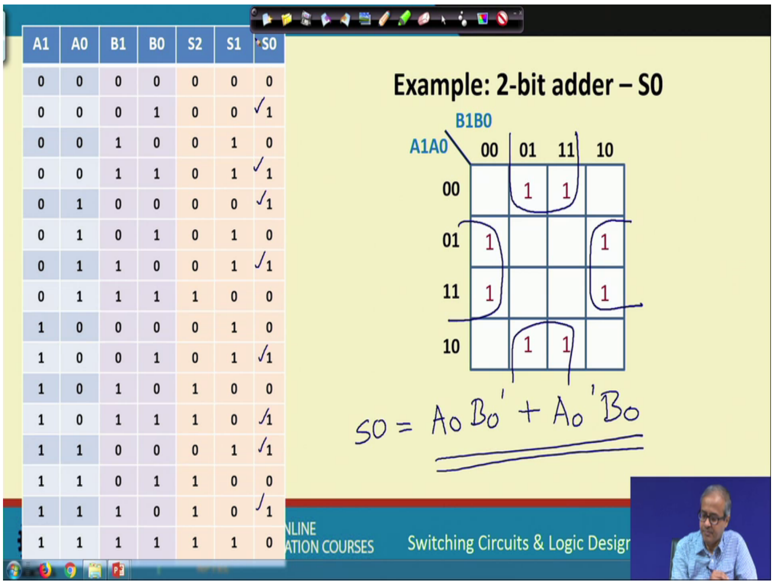
So, you see here the cubes will be like this you cannot minimize it too much. These two, then these two and here these two, these two and these two are isolated 1’s you cannot group them together; they will be isolated. So, there will be 1 2 3 4 5 and 6, 6 cubes.

So, the expression for S 1 will be slightly bigger; look at this one it will be A 1 A 0 A 1 A 0 and 11 10 B 1 sorry not A 1 A 0 A 1 dash A0 dash A 1 dash A 0 dash B 1 ok, plus let us look at this one 00 and 0 1 A 1 dash. This will be A 1 dash 10 B 1 B 0 plus let us look at this one 11 10 is A 1 A 1 00 B1 dash B0 dash.

And this one 10 A 1 A 0 dash and 00 01 is B 1 dash B1 bash and the two isolated 1’s this one will be 01 01; which means A 1 dash A 0 B 1 dash B 0 and the last one will be 1 1 1

1 which is A1 A 0 B 1 B 0. So, I see this expression is little more complex, but it is the minimum form you cannot minimize it any further all right.

(Refer Slide Time: 25:42)




Now, finally, the expression for S 0 the last one here you see for S 0 there are 1 2 3 4 5 6 7 8 1's which are disturbed like this. Now, these are pretty nicely distributed because you can very nicely form the cubes like this 1 cube will be this, this four 1's another cube will be this these four 1's.

So, S0 you can write this two will be A 0 and 00 and 10 and B 0 bar plus this two 00 and 10 A 0 bar 01 11 B 0. This is the minimum form right. So, in this way you can minimize the expressions this the minimum form.

So, we have seen that in case of this 2-bit adder so, how we can create or represent the 3 output functions S2 S1 S0 in the Karnaugh map and then use the cubes to minimize the sum of products expression. So, we have seen I mean how to minimize this functions earlier for 3 variables and now for 4 variables.

So, Karnaugh map is very easy to use, very simple graphically if you wants you can find the cubes from there directly you can write down the minimized form that is the big advantage.

So, with this we come to the end of this lecture. In the next lecture we shall be looking at a few other issues and the concepts regarding the Karnaugh map methods of minimization; before we move on to some other technique, which is more systematic which can be used for larger functions for minimizing switching expressions.

Thank you.