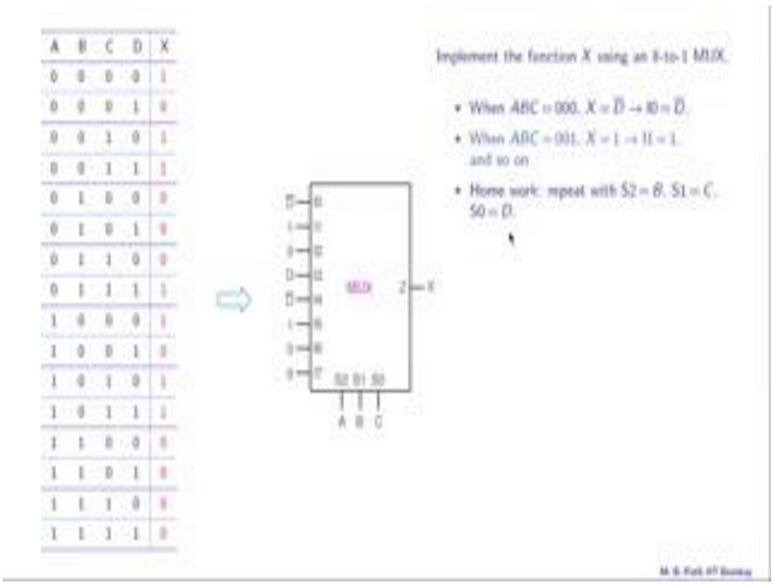
Basic Electronics
Prof. Mahesh Patil
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Lecture - 60
Combinatorial circuits (continued)

Welcome back to Basic Electronics. In this class we will continue with implementation of logical functions with multiplexers. We will then look at several other commonly used blocks including some commercializes. Let us begin.

(Refer Slide Time: 00:35)



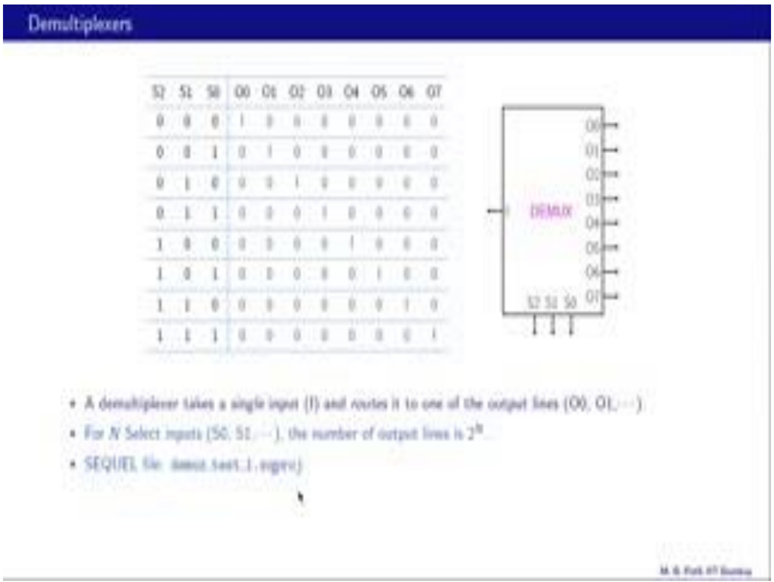
Let us consider this example. Implement the function X using an 8-to-1 multiplexer. Now in this case the function X is not given in the form of a logical expression, but it is given in the form of it is truth table here; so here is our 8-to-1 multiplexer that is the output X. These are the input lines I0 to I7 and these are the select lines. Now suppose we choose S2 equal to A S1 equal to B and S0 equal to C.

Now the question is; what do we connect to these input lines so as to implement this function X here. Let us begin with A B C equal to 0 0 0; that means, this first row and the second row. And now let us relate the output X to the remaining variable that is D. And we see that when D is 0 X is 1, when D is 1 X is 0. In other words, for these 2 rows X is equal to D bar. And that is what we have said over here. When A B C is 0 0 0 X is equal to D bar.

Now let us look at the multiplexer. When A B C is 0 0 0 that is decimal 0, this line I0 gets connected to X, and we want X to be D bar. And therefore, we connect D bar over here all right. Now let us take up the next 2 rows that is A B C equal to 0 0 1 that row and that row, and for these 2 rows we observe that for D equal to 0 X is 1, for D equal to 1 X is 1 again; that means, X is independent of D. And X is simply equal to 1 so when A B C is 0 0 1 X is equal to 1, and when we think of the multiplexer when A B C is 0 0 1 line I1 gets connected to X law and therefore, we connect 1 over here because we want X equal to 1 in this situation like that.

Next we can consider A B C equal to 0 1 0; that means, this row and this row, and then we find that X is 0 irrespective of D. And when A B C is 0 1 0 that is decimal 2 this line I2 gets connected to X and therefore, we will connect I2 to 0 and so on. So that is our complete solution and you should verify these other entries as well all right. Now there is some homework for you, repeat with S2 equal to B S1 equal to C and S0 equal to D.

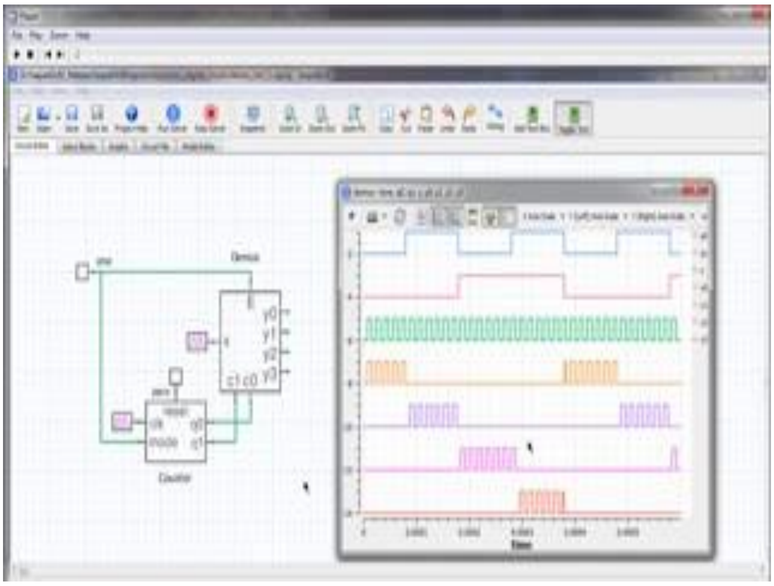
(Refer Slide Time: 04:13)



Let us now look at Demultiplexers and here is an example. We have one input pin, and in this example 8 output pins, O0 O1 O2 up to O7. And 3 select inputs S2 S1 S0. Let us see how it works. Demultiplexer takes a single input i this one, and routes it to one of the output lines, in this case O0 O1 up to O7. So in terms of functionality it is exactly the opposite of what a multiplexer does. And if there are N select inputs in this case N is equal to 3, then the number of output lines is 2 raised to N. So in this case we have 2 raised to 3 or 8 output lines.

Let us now look at the truth table. S2 is the MSB, and S0 is the LSB. So if S2 S1 S0 is 0 0 0 that is decimal 0, then the output line O0 is equal to i and all others are 0 so; that means, this input is routed to the output line O0. Let us take some other example say this one. So S2 S1 S0 is 0 1 1 here, 0 1 1 is decimal 3, so the output line O3 becomes equal to i and all other lines are 0. So this i gets routed to O3 and so on.

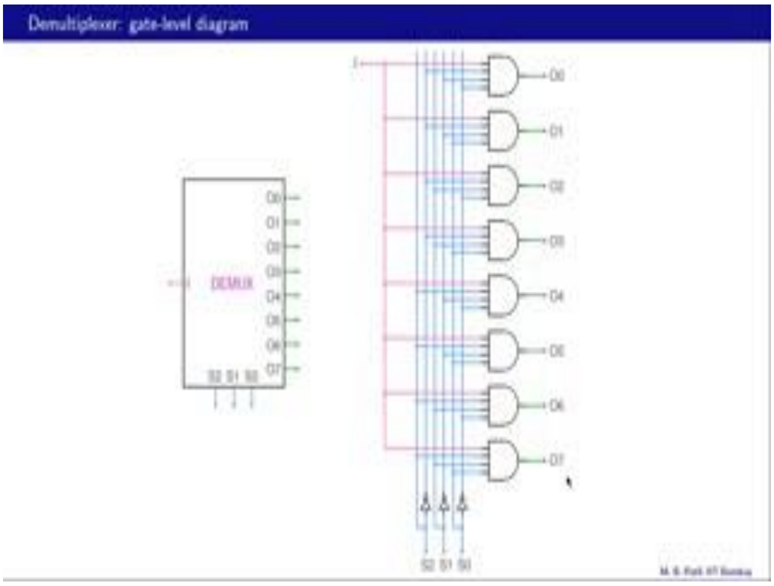
(Refer Slide Time: 06:20)



Let us now illustrate the functionality of a Demultiplexer with this circuit file here. So here is a Demultiplexer. That is the input and the input is a square wave signal, this one here. And it has got select pins, S1 S0 is called c1 c0 here, c for control. And the select lines are coming from this counter so q1 is connected to c1 and q0 is connected to c0, and of course, we are going to see later how this counter works, for now it is only supplying the select lines, and this is what our S0 looks like. And this is what our S1 looks like. And that is the input. Let us now see what happens.

When S1 is 0 and S2 is 0; that means we have binary 0 0 or decimal 0 the input gets routed to y0 and that is what we see over here; this is our line y0 and in this period the input has got routed to this line. When S1 is 0 S0 is 1; that means, binary 0 1 or decimal one X gets routed to y1, that line and therefore, the input appears on this line in this interval here, and so on. So this is how the Demultiplexer works.

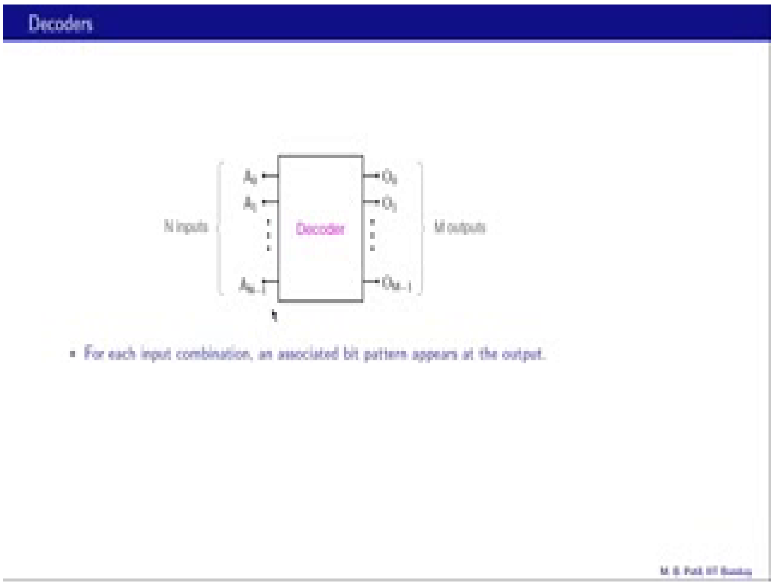
(Refer Slide Time: 08:02)



Let us now look at the gate-level diagram of a Demultiplexer. Here is 1 to 8 Demultiplexer or D-MUX in short. And we have 3 select inputs of course, because we have 8 output lines. And here is the corresponding gate-level diagram. And in this circuit we have 8 gates, the same number as the number of output lines. And each gate gets 4 inputs. One of them is this input line i that one, and then there are 3 others and those 3 others are derived from S2 S1 S0 the select inputs. So, these other 3 inputs can be S2 or S2 bar S1 or S1 bar and S0 or S0 bar. Now let us take some examples and see how it works.

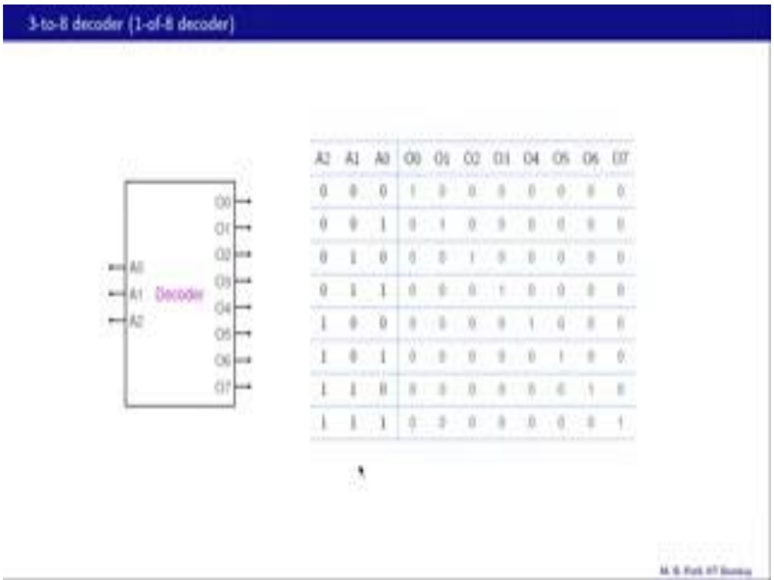
Let us take this example S2 S1 S0 equal to 0 1 1. 0 1 1 is decimal 3 and therefore, we expect the input line to get connected to O3 and all other outputs to be 0. So we expect O3 to be equal to i and O0 O1 O2 O4 O5 O6 O7 to be 0. Let us see if that is happening. Let us look at this gate O3. One of the inputs of course, is i and let us now look at these other 3 inputs marked in blue. This input is S2 bar, we can see that. S2 bar, the second input is S1 and the third input is S0. And if S2 S1 S0 is 0 1 1 then all of these are 1 and therefore, the output of this gate is simply equal to i. And you can verify that all these other outputs are 0. So that is how this Demultiplexer implementation works.

(Refer Slide Time: 10:31)



Let us look at some other digital blocks. Here is a decoder it has got N inputs. A0 A1 etcetera up to a N minus 1. And it has got m outputs O0 O1 up to Om minus 1. How does it work for each input combination only one output line is active and what is the meaning of active; that means, 0 or one depending on whether the outputs are active low or active high all right since there are 2 raised to N input combinations.

(Refer Slide Time: 11:19)

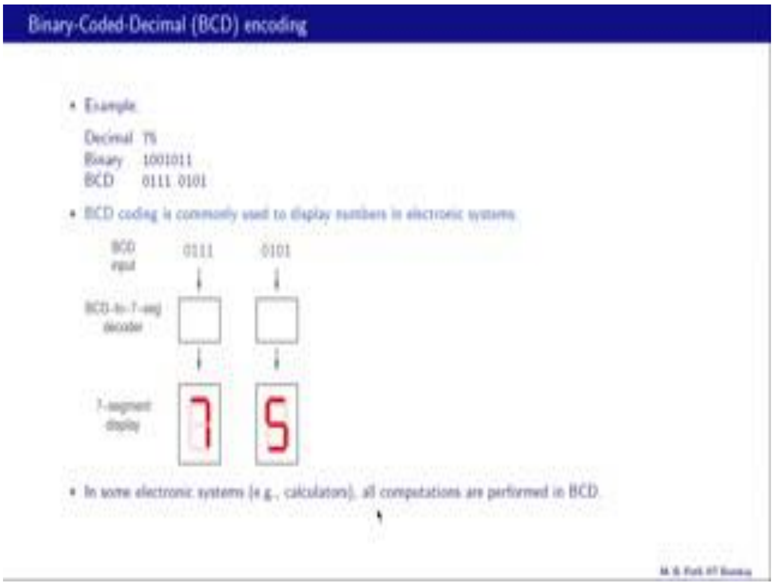


A2	A1	A0	O0	O1	O2	O3	O4	O5	O6	O7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Let us take an example if there are 4 inputs here then there are 2 raised to 4 or 16 input combinations possible and therefore, there could be 2 raised to N output lines in this case there could be 16 output lines and therefore, the maximum value for m is 2 raised to n; however, there are decoders with m less than 2 raised to N as well.

Let us take a look at an example so here is a 3 to 8 decoder; that means 3 inputs and 8 outputs and it is also called one of 8 decoders, because only one of these 8 output lines will be active. Here is the truth table if A2 A1 A0 is 0 0 0 the inputs here, then O0 is active that is 1 and all others are 0. If A2 A1 A0 is 0 0 1 then O1 is active and all others are inactive that is 0 and so on. So this is an example of a 3 to 8 decoder.

(Refer Slide Time: 12:11)

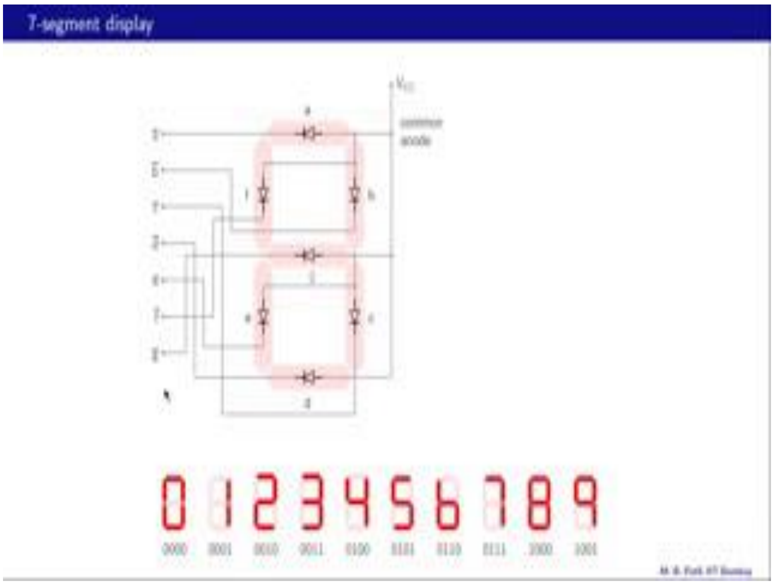


Let us now talk about binary coded decimal or BCD encoding. And we will illustrate that with an example. Let us take this decimal number 75 as an example. And we will first represent that has a binary number and that turns out to be 1 0 0 1 0 1 1. And how do we check whether that is indeed the decimal number 75. We can find the weight associated with each of these bits. This one is the LSB it gets the weight of 2 raised to 0, this one gets 2 raised to 1 and so on. So our number then is the decimal number is 1 plus 2 plus 8 plus 64 and that will turn out to be 75.

Let us now look at the BCD or binary coded decimal representation of this same number 75. Here is the BCD format and we notice that it is in 2 parts: part 1, part 2. Now this part corresponds to the digit 7 here and this part corresponds to the digit 5 here. And each of these is binary coded; that means, this binary number is decimal 7 and we can check that this is 1 plus 2 plus 4 that is decimal 7, and this one is 1 plus 4 that is decimal 5. So that is how this BCD system works.

BCD coding is commonly used to display numbers in electronic systems for example, a watch or a calculator or a display of temperature etcetera. And here is how it works. We will take the same example again decimal number 75, so that is represented by this BCD number part 1, 0 1 1 1 and part 2, 0 1 0 1. This binary input is given to this chip called the BCD to 7 segment decoder. And the output of that is given to this 7 segment display unit and that is how this 7 5 would be displayed. And we should remember that in some electronic systems such as calculators the application of BCD numbers is not just limited to display, even the computations are performed in BCD numbers because that turns out to be very convenient and economical to do.

(Refer Slide Time: 15:16)

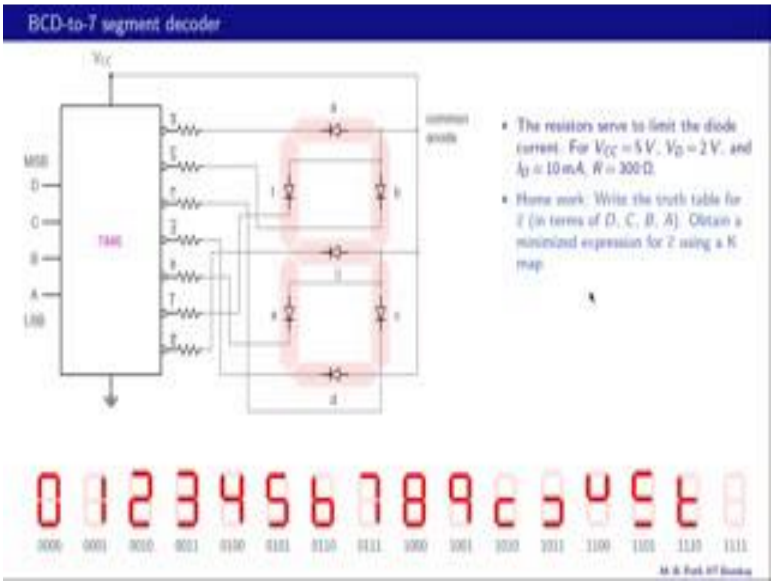


Let us look at this 7 segment display unit now. And the name comes from the fact that there are 7 segments. 1 2 3 4 5 6 and 7 and each segment can be either lit or it can be dark. And that depends on whether this diode conducts or does not conduct. So each segment has this diode which is a light emitting diode or LED and that can be either on or off. If there is a forward voltage across this diode, then it would conduct and it would emit light and we will see that segment in that case as lit. If not, then the diode does not conduct and the segment will appear as a dark segment.

These segments are given names for example, this one is called a this one is called b c d e f and g and each segment of course, is a diode as we mentioned earlier. And the p terminal of each diode that is the anode is connected to the power supply. So they are all tied together. All of them and they are connected to the power supply V CC which is 5 volts. And the other end the n end or the cathode is made available as a pin of this display unit. So this one is called A bar this one the N terminal of the diode is called D bar this one the n terminal of the c diode is called c bar and so on.

Notice that these pins are active low and that is why they are denoted by a bar b bar c bar etcetera and there is good reasons for that of course; suppose this voltage is high or V CC what would happen, the voltage across this diode would then be 0. Because this end is high and this end is also high and therefore, this LED would not light up would not emit light. If this end is low that is active, then this LED would conduct and then the segment would light up. So that is why these pins are active low.

(Refer Slide Time: 18:51)



Here are some patterns. If our input if the binary input is 0 0 0 0 then we would like all of these segments to light up here. If it is 0 0 0 1 then we would like these 2 segments to light up and so on. So what we need now is a decoder which will take this binary number as the input and it will produce these 7 outputs which we can use to drive the display unit and let us now look at such a decoder.

Let us now look at A B C D to 7 segment decoder such as the IC 7446. Now we should mention that these ICs are specific to the type of display unit they would be driving. This one for example, is the common anode type; that means all the anodes or p terminals of the diodes are connected together here. We also have the common cathode type display units in which all the cathodes or the end terminals of the diodes are connected together. And in that case we would be using a different IC in this place.

So let us try to understand the functionality of the 7446 decoder now. What is our overall objective we have a binary number such as 0 0 0 1 which is a part of a BCD number? And if the number is 0 0 0 1 for example, then we would like to see this pattern on the 7 segment unit over here. So what; that means, is that we must have segment B and segment C light up and all the other segments must be dark. And how can we make a specific segment light up. We need to turn on the corresponding LED for example, if we want this segment to appear right, then this LED must conduct. And when this LED s conduct the voltage drop across them is not 0.7 volts like we used to take for a silicon diode, but it is more than that. It could be 2 volts or 3 volts depending on the color. Let us say this is 2 volts.

So, we must provide a voltage drop of 2 volts here. And that is done by making this b bar low say 0 volts. So the current path now is like this, we are 5 volts here, of that 5 volts 2 volts will drop here, and then 3 volts will drop across this register, and that is 0 volts. Now, this register serves a very important purpose and that is to limit the current. And we can make a simple calculation say our V CC is 5 volts, and V D is 2 volts and we want to limit the current to 10 milliamps. We know that this voltage drop is 3 volts then, so the resistance must be 3 volts divided by 10 milliamps that is 300 ohms.

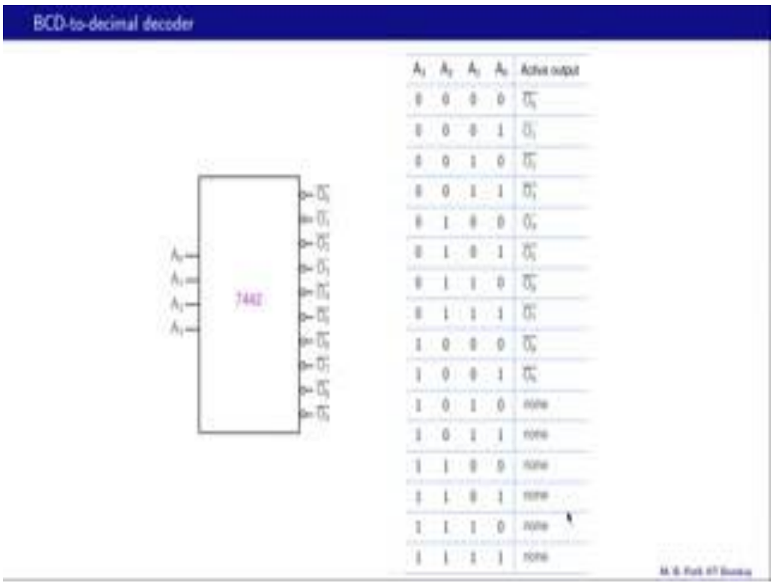
How can we make a specific segment appear dark? We can do that by making sure that the corresponding LED does not conduct. For example, if this segment has to appear dark then this LED should be turned off. And we can ensure that by providing a high voltage at this b bar pin say 5 volts. So then we have 5 volts here, we have 5 volts here. And then the total voltage drops across b the diode b and this resistor is 0. And therefore, there is no possibility of any current the LED is off and therefore, the segment will appear dark.

Now, we understand what this 7446 decoder must do for example, if we want this pattern to appear on the 7 segment display unit. Then we must have this segment light up and this segment light up. And in that case we must make b bar low and also c bar low because the segments to light up are b and c and all other outputs here like a bar and d bar and so on must be made high. So that gives us the truth table for the 7446 decoder.

Now, of these various patterns as a user we only care about numbers up to here; that means, binary numbers from 0 0 0 0 to 1 0 0 1. Because in our BCD number we are not going to encounter any of these binary numbers; however, manufacturers of 7 segment units' use these patterns for some kind of authentication and therefore, these patterns are not arbitrary. They are fixed so for 1 0 1 0 we do expect this pattern although as a user we do not use that. And therefore, we have the entire truth table; now for all combinations of A B C D with D as the MSB and A as the LSB.

So we can prepare a truth table with all possible combinations of the inputs A B C D and for each combination we can figure out whether a given output such as A bar should be high or low. And once we do that then we can figure out what logic can implement that function. And that brings us to this homework; write the truth table for c bar that is this output here in terms of D C B A and then obtain a minimized expression for c bar using a K-map.

(Refer Slide Time: 25:03)



Let us look at the 7442 BCD to decimal decoder. It is input is a 4-bit binary number with a 3 as the MSB and A0 as the LSB. So the input numbers will range from 0 0 0 0 to 1 0 0 1 that is decimal 0 to decimal 9. There are 10 output pins and notice that the output pins are active low, from O0 bar to O9 bar and corresponding to the decimal number that appears here, one of these output pins will be made active, and all others will be made inactive.

Here is the truth table and we are only listing the active output in this column, when A3 A2 A1 A0 is 0 0 0 0 then O0 bar is the active output. When it is 0 0 0 1 then we are O1 bar as the active output, and so on. All the way up to 1 0 0 1 and for this combination O9 bar is the active output. And these combinations are not expected to occur because we are talking about BCD numbers here. So our input binary number is going to correspond only to decimal numbers from 0 to 9. So in these cases none of the output pins is made active.

In summary, we have looked at several useful digital blocks namely multiplexer Demultiplexer and decoder. In the next class we will see how an encoder works, and then start with sequential circuits. See you next time.