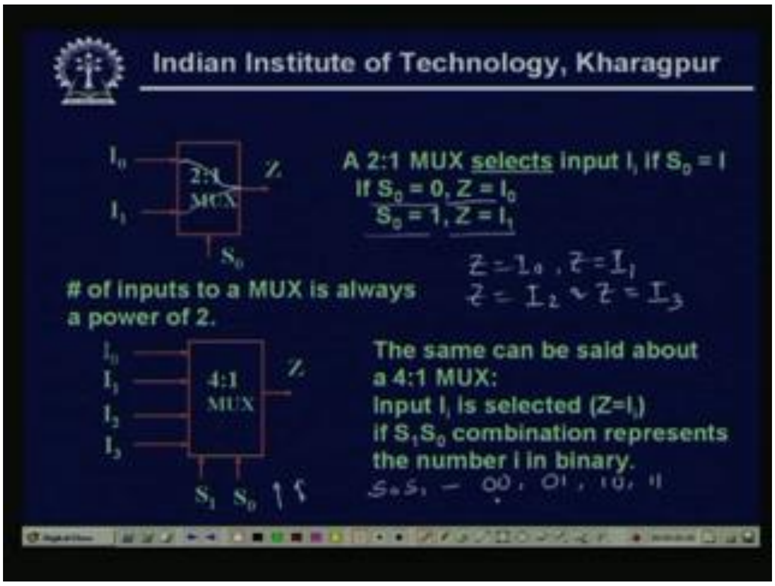
# Digital Systems Design
## Prof. D. Roychoudhury
## Department of Computer Science and Engineering
## Indian Institute of Technology, Kharagpur

## Lecture - 17
## Design of Combinational Circuits Using Multiplexer

In the last lecture we have seen, how the complex combinational circuits can be designed using fundamental circuits, this fundamental circuits mean the we have considered half adder, full adder, the decoder**.** Now, today we will read how the combinational circuits can be designed using another fundamental circuits called multiplexer.

(Refer Slide Time: 01:27)



Indian Institute of Technology, Kharagpur

A 2:1 MUX selects input $I_i$ if $S_0 = i$
If $S_0 = 0$, $Z = I_0$
$S_0 = 1$, $Z = I_1$

$Z = I_0$, $Z = I_1$
$Z = I_2$ ~ $Z = I_3$

# of inputs to a MUX is always a power of 2.

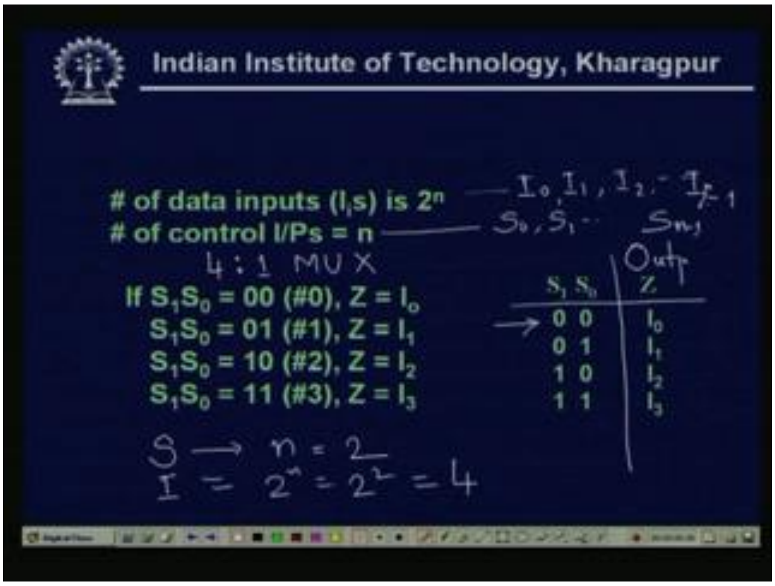
The same can be said about a 4:1 MUX:
Input $I_i$ is selected ($Z=I_i$)
if $S_1S_0$ combination represents the number i in binary.
$S_0S_1$ – 00, 01, 10, 11

Now, all ready we have read the functions of multiplexer, but again quickly we revisit that thing. So, how we have defined 2 to 1 multiplexer, it has 2 input line and 1 select line, now function is that if the inputs are passed to the output depending on the particular value of the select line input. That means, if the Z is I 0 that I 0 is passed to Z, when S 0 is 0 similarly, when I 1 is passed to Z; that means, Z equal I 1 if S 0 is 1, now if we consider a 4 to 1 multiplexer, 4 to 1 means that 4 input 1 output.

Now, here we have to select 4 lines; that means, either I Z equal to I 0, Z equal to I 1, Z equal to I 2 and Z equal to I 3. So, I need 4 combinations of select lines and if it is 4 combinations, then I need at least 2 select lines, so it will be 0 0 0 1. That means, my S 0

S 1 values are 0 0, 0 1, 1 0, 1 1, so for this 4 combinations that I 0, I 1, I 2 and I 3 will be passed to Z.

(Refer Slide Time: 03:32)



Indian Institute of Technology, Kharagpur

# of data inputs ($I_i$s) is $2^n$ — $I_0, I_1, I_2, \cdots I_{2^n-1}$

# of control I/Ps = n — $S_0, S_1 \cdots S_{n-1}$

4 : 1 MUX

If $S_1S_0$ = 00 (#0), Z = $I_0$

$S_1S_0$ = 01 (#1), Z = $I_1$

$S_1S_0$ = 10 (#2), Z = $I_2$

$S_1S_0$ = 11 (#3), Z = $I_3$

|  | Outp |
|---|---|
| $S_1$ $S_0$ | Z |
| → 0 0 | $I_0$ |
| 0 1 | $I_1$ |
| 1 0 | $I_2$ |
| 1 1 | $I_3$ |

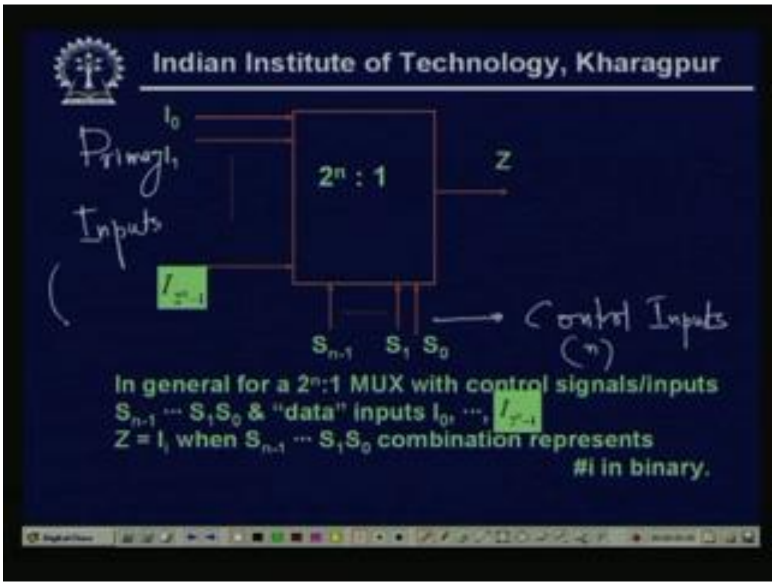
$S \longrightarrow n = 2$

$I = 2^n = 2^2 = 4$

So, in general what we can tell that number of data inputs is 2 to the power n and number of control input is n, control input means my our select lines. So, this is my select lines should be S 0, S 1 up to S n, then my number of data inputs is I 0, I 1, I 2 then I 2 to the power n minus 1. So, here actually S should be n such lines, so it will be S 0 to S n minus 1 and for that n values it will be input should be 2 to the power n number of inputs, means 0 2 to the power n minus 1.

So, just now what we have seen this is for my 4 variable inputs or 4 to 1 multiplexers that S should be two lines or n equal to 2 and I equal to 2 square 2 to the power n equal to 2 square equal to 4. So, just the earlier slide what we have told that if S 1 S 0 is 0 0 then my output this is my output Z that will be I 0, if it is 0 1 output is I 1, if it is 1 0 output is I 2, if it is 1 1 output is I 3. So, earlier we have read only 2 to 1 multiplexer 4 to 1 multiplexer can be designed, again using 4 to 1 how I can design 8 to 1 multiplexer al this things.

(Refer Slide Time: 06:05)

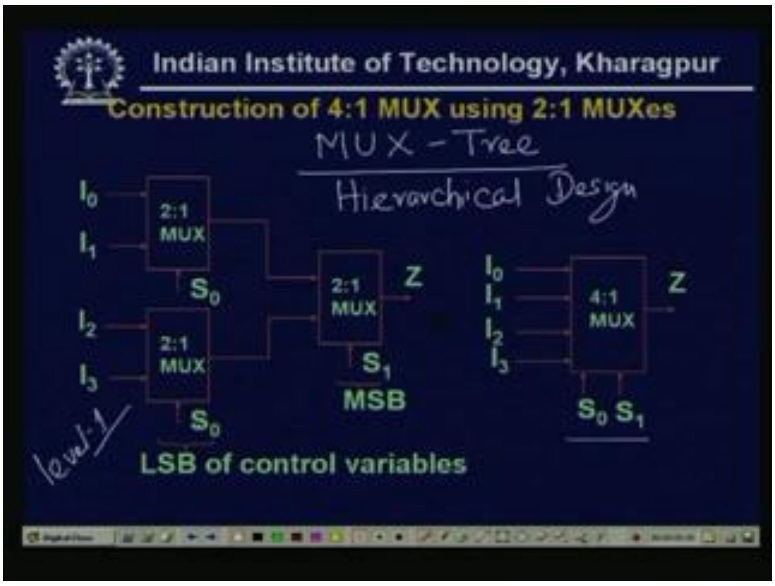


Indian Institute of Technology, Kharagpur

In general for a $2^n$:1 MUX with control signals/inputs $S_{n-1} \cdots S_1S_0$ & "data" inputs $I_0, \cdots, I_{2^n-1}$
$Z = I_i$ when $S_{n-1} \cdots S_1S_0$ combination represents #i in binary.

Now, today we will see first the general structure, so if now we draw the general diagram, then this will be my n control inputs these are my control inputs and these are my primary inputs. So, for S 0 to S n minus 1 means n control inputs primary input should be 2 to the power n and only 1 output.

(Refer Slide Time: 06:56)



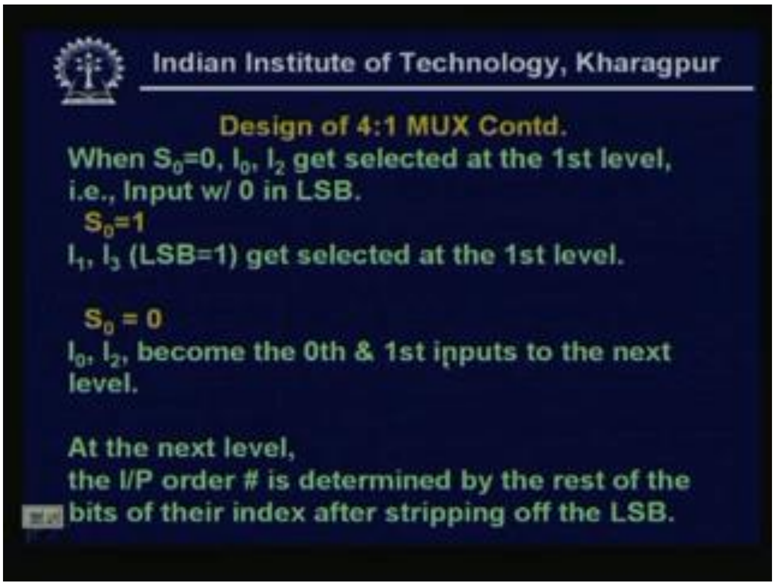
Indian Institute of Technology, Kharagpur

Construction of 4:1 MUX using 2:1 MUXes

LSB of control variables

Now, that 4 to 1 multiplexer using 2 to 1 multiplexer, we call this is as a MUX tree or multiplexer tree, this is in general called the MUX tree or sometimes it is called the hierarchical design of MUX's, this is the hierarchical design. So, we take 2 to 1 multiplexer and it is fed to another 2 to 1 multiplexer. So, for 4 to 1 multiplexers we need 4 inputs I 0, I 1, I 2, I 3 this and two control inputs. They are actually hierarchically fed in the first level, this is my level 1. Here, the control input is S 0 for both the 2 to 1 multiplexer, in the second level this is my level 2 here S 1 is fed. So, these are the two control variables or the control variables 2 bit control variable.

(Refer Slide Time: 08:36)



Indian Institute of Technology, Kharagpur

Design of 4:1 MUX Contd.

When $S_0$=0, $I_0$, $I_2$ get selected at the 1st level, i.e., Input w/ 0 in LSB.

$S_0$=1

$I_1$, $I_3$ (LSB=1) get selected at the 1st level.

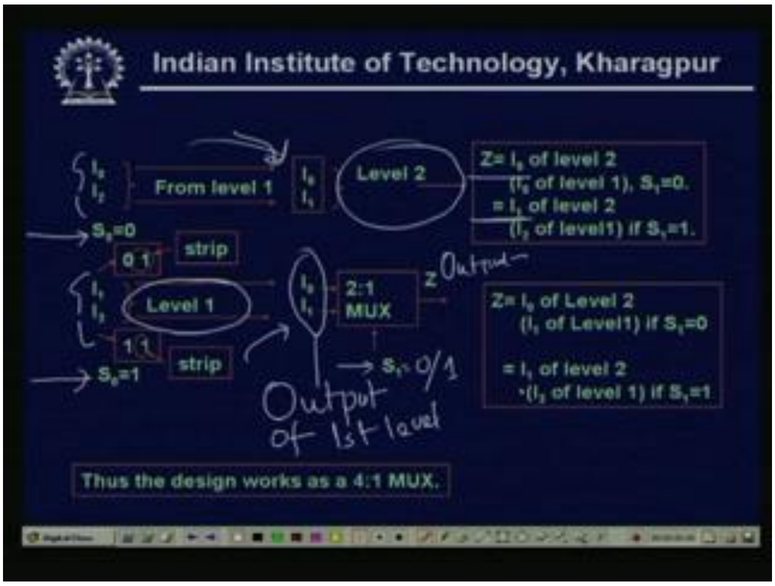
$S_0$ = 0

$I_0$, $I_2$, become the 0th & 1st inputs to the next level.

At the next level, the I/P order # is determined by the rest of the bits of their index after stripping off the LSB.

So, when S 0 is 0, I 0 I 2 get selected in the first level and when S 0 is 1, then I 1 I 3 get selected at the first level. So, if S 0 equal to 0 this become the 0'th I 0, I 2 0th and the first inputs to the next level. Now, at the next level mean the second level the input, output determine by the rest of the bits of their index after stripping off the LSB.

(Refer Slide Time: 09:14)


Indian Institute of Technology, Kharagpur

Z= $I_0$ of level 2
($I_0$ of level 1), $S_1$=0.
= $I_1$ of level 2
($I_2$ of level1) if $S_1$=1.

Z= $I_0$ of Level 2
($I_1$ of Level1) if $S_1$=0

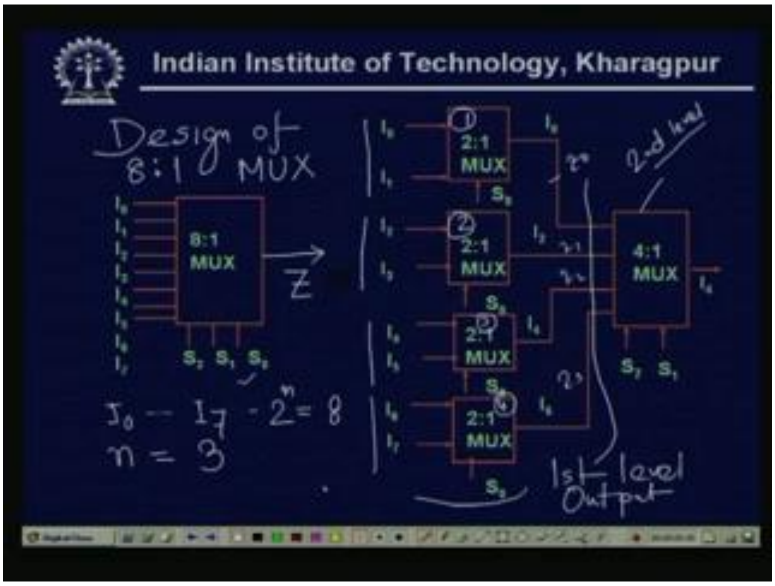
= $I_1$ of level 2
·($I_3$ of level 1) if $S_1$=1

Thus the design works as a 4:1 MUX.

So, now if I want to design a general architecture using multiplexer, then actually we have to consider the multiple levels. So, in the just now what we have seen, we have level 1 and level 2, so this is my level 1 and this is my level 2, so, in the first level I 0 I 2 is fed and in the second level again in the first level I 1 I 3 these are fed 2 2 to 1 multiplexer. So, here S 0 equal to 0, so when S 0 equal to 0 this is when S 0 equal to 0, then I 0 I 2 is passed to the second level input, when S 0 equal to 1, then I 1 I 3 is passed to the second level.

Now, in the second level there are 2 inputs, see here this is my 2 inputs I 0 and I 1, so these 2 inputs are the output of the first level and this is my final output Z. So, S 1 is the control input and that can be 0 or 1, so if it is 0 Z equal to I 0, if it is 1 Z equal to I 1. So, if I summarize that thing that Z equal to for the Z equal to I 0 of level 2, when S 1 equal to 0 when S 1 equal to 1, Z equal to I 1. So, the design works as a 4 to 1 multiplexer.

(Refer Slide Time: 11:52)



Indian Institute of Technology, Kharagpur

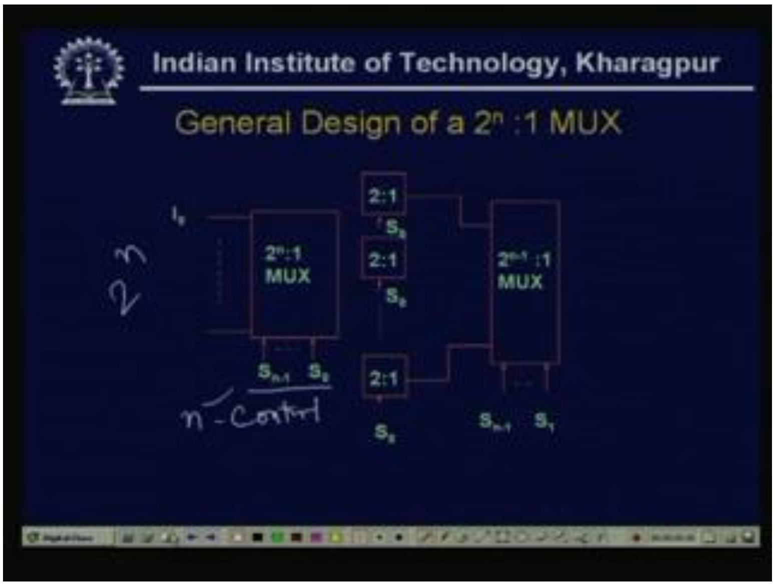
Now, if I apply the same thing for constructing a 8 to 1 multiplexer, so we can write that design of 8 to 1 MUX. Now, for 8 to 1 MUX there are 8 inputs, primary inputs are 8 that is I 0 to I 7. So, my 2 to the power n equal to 8, so n equal to 3, so I need 3 control inputs. So, in this way I define my or I choose how many control lines will be there or how many select lines will be there, so 8 inputs I 0 to I 7 and there are three select lines.

Now, if I want to design 8 to 1 multiplexer using small multiplexers say 2 to 1 or 4 to 1, then as 8 to 1 if I use 2 to 1 multiplexer. So, I need 4 such multiplexers because, I have 8 inputs. So, these I have given as for each 2 to 1 multiplexer I have given 2 inputs I 0, I 1 for the first one this is my number 1 multiplexer, this is my number 2 multiplexers where I 2, I 3 is there for 3 this is I 4, I 5 for 4 this is I 6, I 7. So, I 0 to I 7 primary inputs are applied 4 2 to 1 multiplexer.

Now, I got actually these are the first level outputs, so these are my first level output which will be treated as the second level inputs, as there are 4 outputs. So, I am at the second level I am taking a 4 to 1 multiplexer, now all ready we have seen the how 4 to 1 multiplexer can be designed using 3 2 to 1 multiplexer. So, this 4 to 1 multiplexer can be replaced by 3 2 to 1 multiplexers, then it will be a hierarchical thing, first 4 2 to 1 then 2 2 to 1 then 1 2 to 1 multiplexer, that is also another alternative design of the 8 to 1 multiplexer.

Now, this second level here these are the first level outputs actually these are Z 0, Z 1 I can tell this is Z 0, Z 1, Z 2, Z 3 that first level output they are treated as the second level input. So, these are also first level output, they are also the second level input, then they are fed to a 4 to 1 multiplexer. Now, here 4 to 1 we know that 4 primary inputs and 2 control lines, so and so this will be 1 output that is my Z output. So, in this way I can design.

(Refer Slide Time: 16:03)

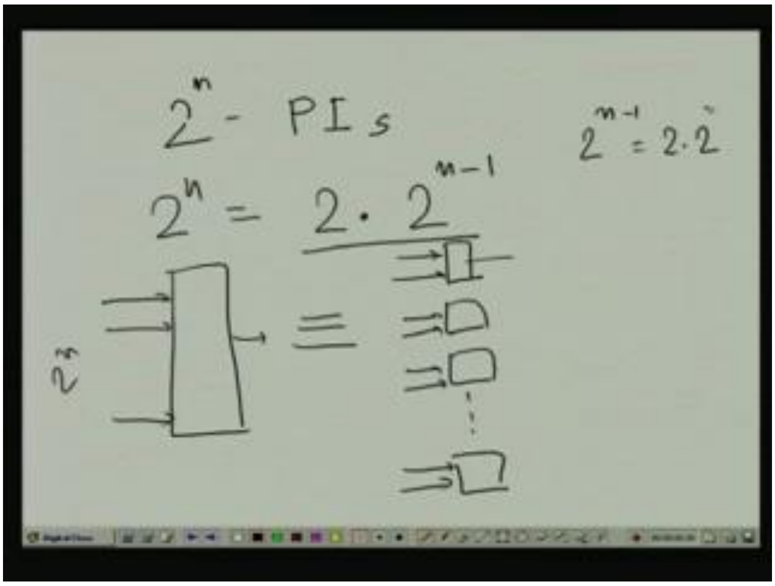


Indian Institute of Technology, Kharagpur

General Design of a $2^n$ :1 MUX

So, if we see that general design… Now if we generalize this design structure… So generalize I have n control lines, n control inputs then 2 to the power n primary inputs. It is fed to a 2 to the power n to 1 multiplexer. How I can design that thing in a hierarchical structure? So, 2 to the power n, so I need that n number of or 2 to the power n minus 1 number of 2 to 1 multiplexer.

(Refer Slide Time: 17:01)

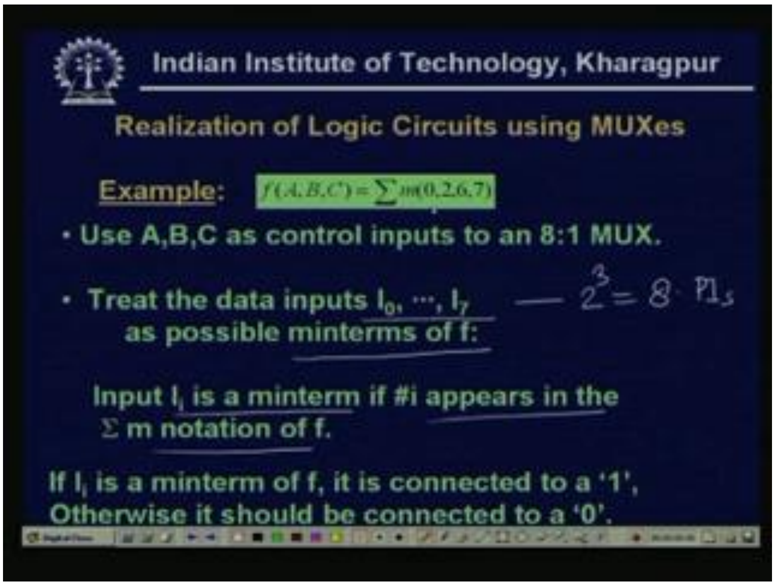


Why, see I have 2 to the power n primary inputs now 2 to the power n can be written as 2 into 2 to the power n minus 1. So, instead of applying 2 to the power n to the at a time the primary inputs these are 2 to the power n what I can do? These I can replace by say 2 to the power n minus 1 number of 2 to 1 multiplexer, where every multiplexer has 2 inputs, so this will be 2 into 2 to the power n minus 1.

Now, in the next step what we can do that 2 to the power n minus 1 that can be written as 2 into 2 to the power n minus 2. So, in that way I can replace the circuits, ((Refer Time: 18:21)) see here the 2 to the power n primary inputs they are replaced by 2 to the power n minus 1 number of 2 to 1 multiplexer. So, here there will be 2 to the power n minus 1 here it was 2 to the power n number of primary inputs, where it is the 2 to the power n minus 1 number of primary inputs.

Now, this is a 2 to the power n minus 1 to 1 multiplexer. Even if I want another hierarchical structure here, than in that way the 2 into 2 to the power n minus 2 or 2 into 2 to the power n minus 3. In that way I can design that hierarchical tree or the multiplexer tree. So, what we can conclude from here, that using this 2 to 1 multiplexer or using this 2 to 1 multiplexer. Any large multiplexer circuit, any large multiplexer can be designed, so hierarchical design of 2 to 1 multiplexer is possible.

(Refer Slide Time: 19:56)



Indian Institute of Technology, Kharagpur

Realization of Logic Circuits using MUXes

Example: $f(A,B,C) = \sum m(0,2,6,7)$

- Use A,B,C as control inputs to an 8:1 MUX.
- Treat the data inputs $I_0, \cdots, I_7$ as possible minterms of f: — $2^3 = 8 \cdot PI_s$

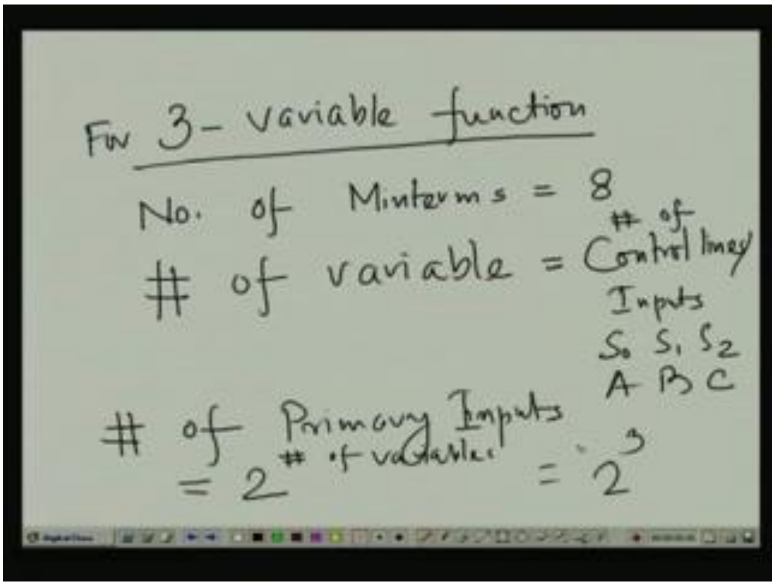
Input $I_i$ is a minterm if #i appears in the $\Sigma$ m notation of f.

If $I_i$ is a minterm of f, it is connected to a '1', Otherwise it should be connected to a '0'.

So, now we want to concentrate that the today's topic that how the any combinational circuits can be designed using this multiplexer. Because, just now what we have seen that any larger multiplexer can be designed using the smallest or the fundamental multiplexer 2 to 1 multiplexers. So, if now we any combination circuits we can realize using multiplexer then; obviously, that design structure has some regularity that using this multiplexer only I can the whole circuit can be designed.

First we take one simple example, say I have a 3 variable function, f where the min term exist in the function is 0, 2, 6, 7. So, min terms are 0, 2, 6 and 7. Now, how we can map this thing to a multiplexer. See if A, B, C we define as the three select lines or the control inputs, if it is a three select lines. Then the multiplexer will be 2 to the power 3, means that 8 to 1 multiplexer primary inputs will be 8.

(Refer Slide Time: 21:55)



For 3- variable function

No. of Minterms = 8

# of variable = # of Control lines Inputs

$S_0$ $S_1$ $S_2$

A B C

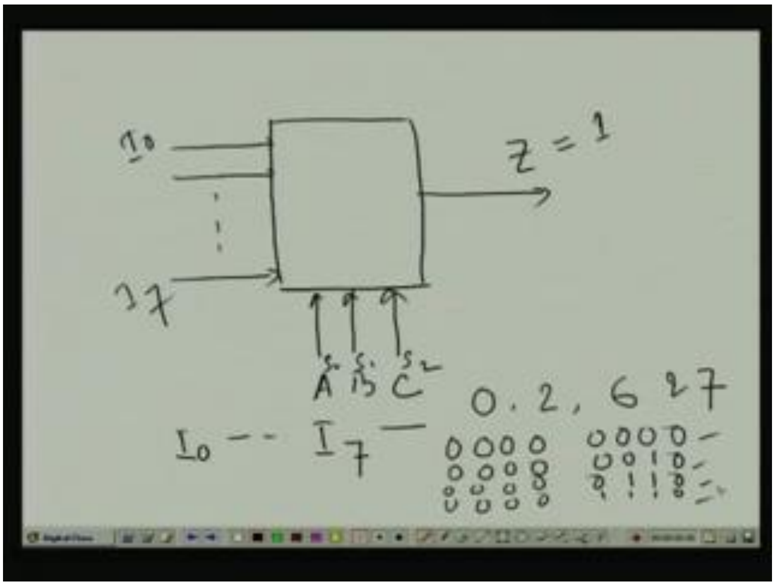
# of Primary Inputs

$= 2^{\# \text{ of variables}} = 2^3$

And if it is a we know that if it is a 3 variable function, the number of min terms. So, for 3 variable function number of min terms at 8. So, the number of variable we can take then or we can map the number of variable as the control lines or control inputs. So, in this case it is S 0, S 1, S 2 that is my A, B, C. And the number of primary inputs will be the 2 to the power number of variables, see in this case it will be 2 to the power 3 ((Refer Time: 23:41)).

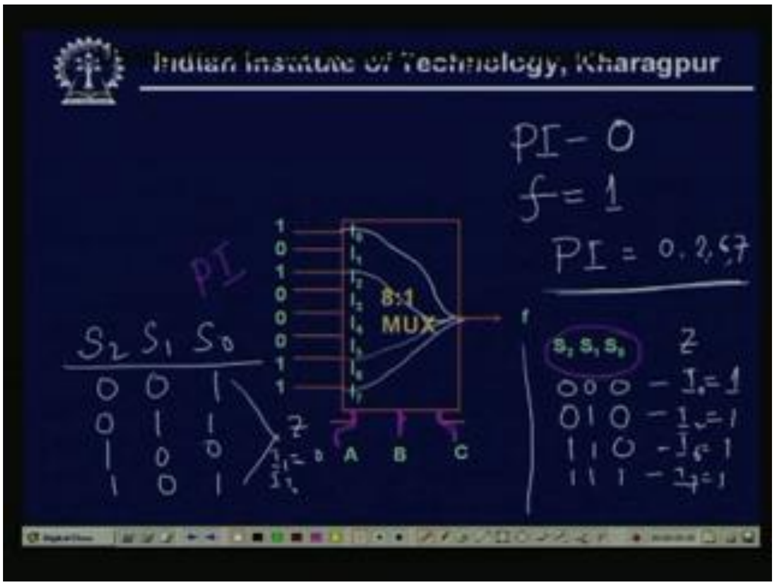
So, A, B, C as the control inputs for 8 to 1 multiplexer and the data inputs are I 0 to I 7. Because, already we have seen this is 2 to the power 3 equal to 8 primary Inputs, 8 PI's and these are the possible min terms of f. So, the decimal equivalent of the primary inputs that represents my min terms. So, input I i or the primary input is a min term and that number appears in the notation f.

(Refer Slide Time: 24:35)



So; that means, I want to design a multiplexer whose, if I want to design a multiplexer, that it has 3 input lines, say these are my S 0, S 1, S 2 and 8 primary inputs I 0 to I 7. And only my Z equal to 1 means the Z or output it will give the output for the primary inputs. That I 0 to I 7 these values will be the decimal equivalent will be that 0, 2, 6 and 7 means, these values will be all 0 0, this value will be 2, then 6, then 7. So, only for these values that output will be 1 ((Refer Time: 26:03)). So, if that particular I i is the min term of f it is connected to a 1, otherwise it should be connected to a 0.

(Refer Slide Time: 26:29)

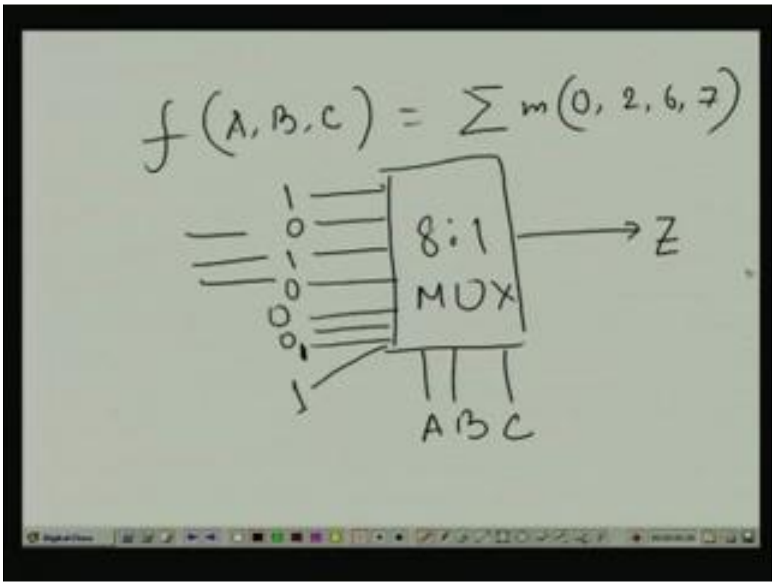

Indian Institute of Technology, Kharagpur

8:1 MUX

So, just now what we have used say this is my 3 select lines A, B, C these are my this S 0, S 1, S 2 and these are the my primary inputs. That means, when this primary inputs take the value 0 in decimal, then it will be that f will be 1 or what we can tell that say for PI equal to 1 of the PI values are 0, 2, 6, 7. So, which value of A, B, C these PI values are will be taken because, these are variables. So, for 0 0 say for we know that, if my select lines are 0 0 0, then I 0 will be passed to 1, so this is my I 0 which is kept as 1.

Then, 2 means it is 0 1 0 then my Z is and it is my Z this is I 2 which is kept as 1. If it is 6 means 0 or 1 1 0 this is 1 1 0 then I 6 is 1 it is 1 1 1, then I 7 equal to 1. That means, only these values when passed. For the rest of the values, that means, say for 0 0 1 for 0 1 1 3, 1 0 0 4, 1 0 1 5 for these 4 values my output Z is 0. That means I 1 equal to 0, I 3 equal to 0, I 4 equal to 0 and I 5 equal to 0. So, this MUX this 8 to 1 multiplexer is nothing but, the realization of the function sigma m 0, 2, 6, 7.

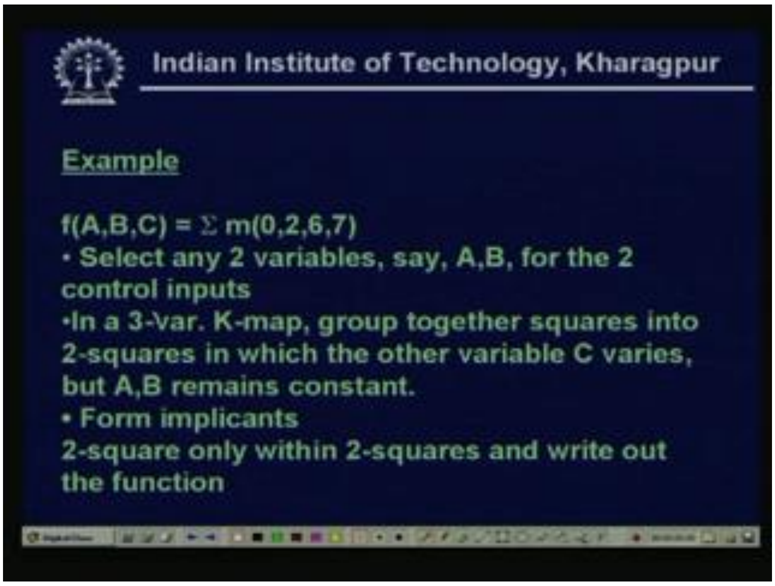
(Refer Slide Time: 30:22)



So, this combinational circuits what we can tell that the combinational circuit that a 3 variable combinational circuits f A, B, C is sum of m min terms are 0, 2, 6, 7. That is nothing but, a 8 to 1 multiplexers, where these values are A, B, C 3 select lines and the output will be only this 0, 2, 6, 7 lines. In these inputs are, 1. That means, this is 0, this is 1, 3 line is 0, 4 line is 0, 5 line is 0, 6 line is 1 and 7'th line is 1. So, this is the realization using multiplexer.

See, only the decimal equivalent of the min terms the particular min term and that is mapped to the decimal equivalent of the primary inputs. And that particular min terms which min term exist in the function I have to keep that particular input as 1. Because, only for that for this MUX that particular input will be passed to the output as 1.

(Refer Slide Time: 32:07)



Indian Institute of Technology, Kharagpur

Example

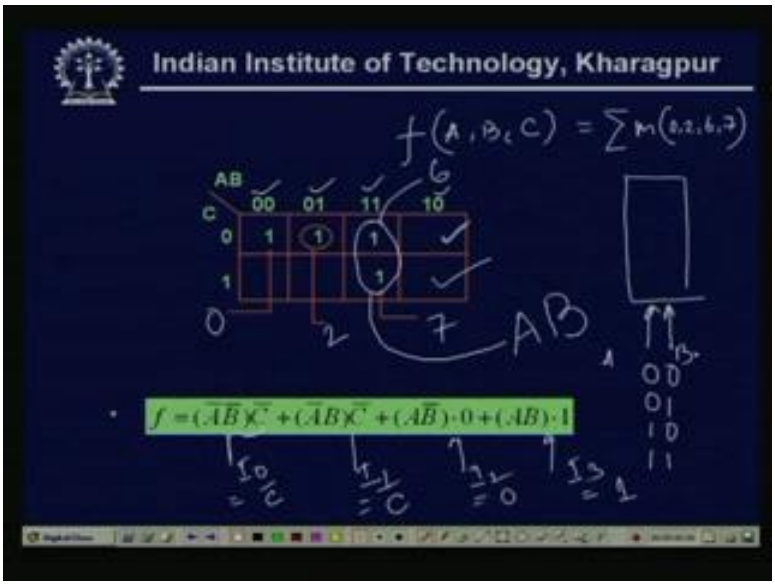
f(A,B,C) = $\Sigma$ m(0,2,6,7)

- Select any 2 variables, say, A,B, for the 2 control inputs
- In a 3-var. K-map, group together squares into 2-squares in which the other variable C varies, but A,B remains constant.
- Form implicants

2-square only within 2-squares and write out the function

So, for the same example, if we see the general design rule what we will be doing, that we take that a 3 variable function m 0, 2, 6, 7 then select any 2 variable A, B for the 2 control inputs. Now, earlier I have taken the 3 variables as the control inputs, now in the 3 variable Karnaugh map group together squares into 2 squares, in which the other variable C varies but A, B remains constant. So from the implicants 2 squares only within 2 squares and write out the function.

(Refer Slide Time: 33:04)



Indian Institute of Technology, Kharagpur

$f = (\overline{A}\,\overline{B})\overline{C} + (\overline{A}B)\overline{C} + (A\overline{B})\cdot 0 + (AB)\cdot 1$

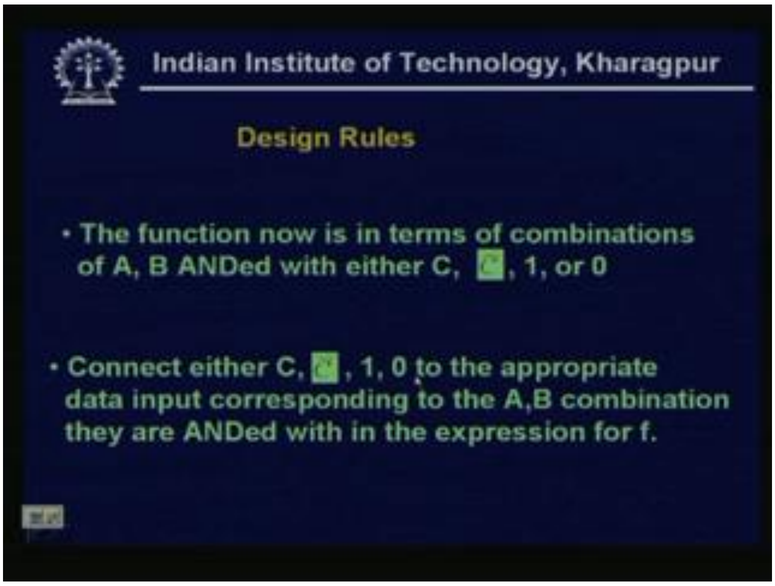
See, here I have taken a 3 variable function, where the 0, 2, 6, 7 that values are there. That means that my function is f of A, B, C and same example that 0, 2, 6 and 7 only that min terms will be there. So, this is my Karnaugh map realization, so this is my 0, this is my 0 1 0 means 2, this is my 1 1 0, means this is my 6 this is 1 1 1 7. So, what will be the function, function will be A bar, B bar, C bar, then A bar B C bar.

And this is my, see if I group together this is one couple and this will be my A, B, so this is my A B, this is my actually this is A bar, B bar, C bar this is A bar B, A bar B C bar and see which one is A B bar. See, A B bar is this one, this is totally 0; that means what I am doing, if I take the… I want to design this combinational circuit using a multiplexer. So, for this multiplexer I know that earlier what we have seen, that all the 3 variables I have made that 3 variables as the control inputs or the select lines, now I want that only 2 variables are the control inputs.

So, if A B are the control inputs, then if it is a multiplexer A B are the control inputs, then it will be either 0 0. So, these are my A B 0 0, 0 1, 1 0, 1 1, see, so these are my values means that which represents the columns this is 0 0, 0 1, 1 0, 1 1, now 0 0 means only C bar exist. So, actually what I can tell this is my I 0 term A bar B bar means 0 0 for 0 0, this is my I 0 term, see this is my I 1 term 0 1, 0 1 means second column A bar B. So, this is my I 1 term, A B bar means that actually 4th column A B bar is the 4th column. So, that is my I 2 and this is A B means 1 1 this is my I 3.

Now, see what is my I 0, I 1, I 2, I 3, when A B is 0 0 I 0 is nothing but, C bar I 1 is also C bar, I 2 is 0 because, my 4'th column is empty I 3 is 1. So, that 3 variable actually I have changed 2 as the control lines and 1 as the in primary input.

(Refer Slide Time: 38:05)



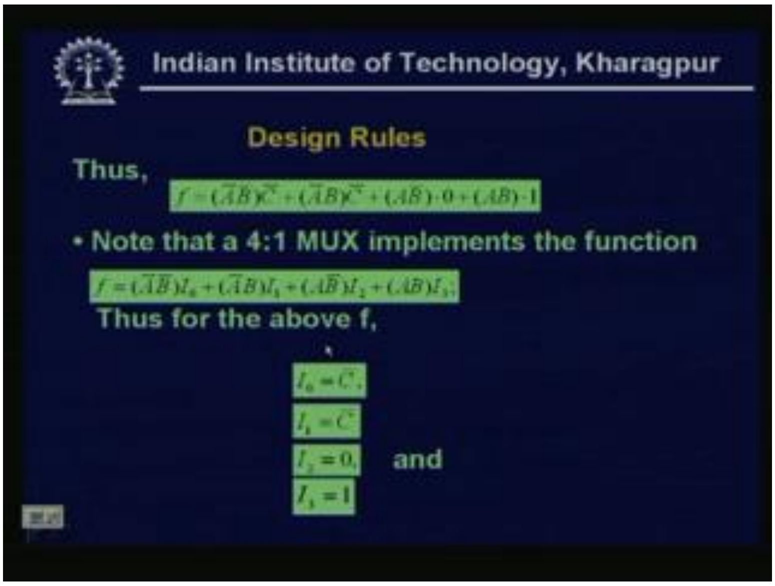
Indian Institute of Technology, Kharagpur

Design Rules

- The function now is in terms of combinations of A, B ANDed with either C, $\bar{C}$, 1, or 0
- Connect either C, $\bar{C}$, 1, 0 to the appropriate data input corresponding to the A,B combination they are ANDed with in the expression for f.

So, if we see the design, what will be my design rules, function will be in terms of combinations of AB ANDed with either C C bar 1's are 0. What we have seen that this is my C C bar, C bar 0 and 1 here there is no C term or a C bar. So, actually 0 1 and the other primary input left that will be treated as the primary input of the multiplexer and the 2 A B lines are treated as the ((Refer Time: 38:47)) select lines. So, we will connect either C C bar 1, 0 to the data input or the primary input corresponding to the A B combination they are ANDed with the expression for f.

(Refer Slide Time: 39:02)



Indian Institute of Technology, Kharagpur

Design Rules

Thus, $f = (\bar{A}\bar{B})\bar{C} + (\bar{A}B)\bar{C} + (A\bar{B})\cdot 0 + (AB)\cdot 1$

- Note that a 4:1 MUX implements the function $f = (\bar{A}\bar{B})I_0 + (\bar{A}B)I_1 + (A\bar{B})I_2 + (AB)I_3$;

Thus for the above f,

$I_0 = \bar{C}$,

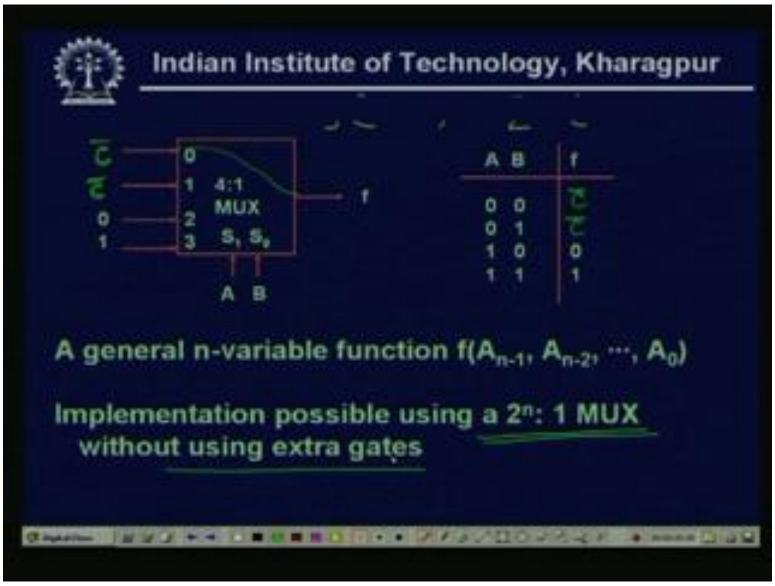
$I_1 = \bar{C}$

$I_2 = 0$, and

$I_3 = 1$

So, already we have seen this is my function. If we want to realize that using multiplexer that 4 to 1 multiplexer implementation because, there are 2 I have selected 2 control lines or select lines. And so for 2 control lines there will be 4 inputs, 4 primary inputs they C bar, C bar 0 1.

(Refer Slide Time: 39:33)



Indian Institute of Technology, Kharagpur

| A | B | f |
|---|---|---|
| 0 | 0 | $\bar{C}$ |
| 0 | 1 | $\bar{C}$ |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

A general n-variable function $f(A_{n-1}, A_{n-2}, \cdots, A_0)$

Implementation possible using a $2^n$: 1 MUX without using extra gates

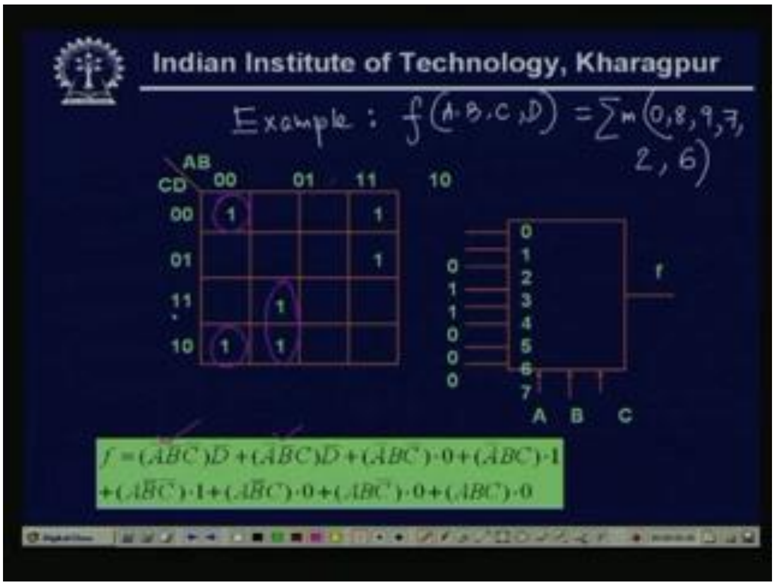
So, this is my design of that f equal to 0, 2, 6, 7 or we can tell that just now what design we have seen that A B are the 2 select lines and the primary inputs are C bar, C bar. That means, when A B equal to 0 0 then also C bar will be passed to f when f equal to 0 1 then also C bar will be passed to f, so this is C bar, C bar 0 1. So, general n variable function implementation possible using a 2 to 2 the power n 2 1 multiplexer. Because, this is a 4 to 1 multiplexer without using extra gates, so always this implementation is possible.

So, far what we have discussed that if it is a n variable function that what we can do, that all n variables we can take as the select lines, that you select a particular input primary input to be fed to the output. Now, the mean terms which, present in the combinational circuits; only for that min term or the binary of that min terms which fed to the select lines. Then that particular input will be kept or will be applied as 1. And all other terms which are not the min terms of that combinational circuits, they are kept as 0.

So, that multiplexer will realize that particular combinational circuits. Now, in this current example what we have seen, even we can reduce that control signals. What we have done that in this case that for n variable we have taken a n minus 1 select lines. And the primary input is replaced by 0 1 and the compliment and un complimented input which is left.

So, in this way what we can do, that we can optimize, design that select lines. We can divide the number of variables a some as the select lines and the combinations of some of the input variables are fed to the primary inputs.

(Refer Slide Time: 43:45)



Indian Institute of Technology, Kharagpur

Example : $f(A \cdot B \cdot C, D) = \sum m(0, 8, 9, 7, 2, 6)$

| CD \ AB | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | | | 1 |
| 01 | | | | 1 |
| 11 | | 1 | | |
| 10 | 1 | 1 | | |

$f = (\bar{A}\bar{B}\bar{C})\bar{D} + (\bar{A}\bar{B}C)\bar{D} + (\bar{A}B\bar{C})\cdot 0 + (\bar{A}BC)\cdot 1$
$+ (A\bar{B}\bar{C})\cdot 1 + (A\bar{B}C)\cdot 0 + (AB\bar{C})\cdot 0 + (ABC)\cdot 0$

Now, we see how we can do a larger design? So, 4 variable design using multiplexer, see this is one example, where the function is one combinational circuits we are taking on the function is A bar, B bar, C bar, D bar A bar B bar C D bar, A bar B C bar. Or actually this terms is not there A bar B C or what we can tell that number. We can define this function as, this is one example, where this is the 4 variable function A, B, C, D and the min term exist are see this is 0 0; that means, 0 is there.

See this is 1 0, this actually is shifted. This is 1 0, 0 0 means my 8, then for the second row this is 1 0, 0 1 this is 9. Third row, this is 0 1 1 1 7 for the fourth row it is 0 0 1 0 means 2 and this is 0 1 1 0 means 6, so these 6 min terms are there. So, this is one combinational circuit I want to realize using multiplexer. So, first what we do again if I say first I see what is my min terms available, see this is my A bar, B bar, C bar, D bar. So, this is my A bar, B bar, C bar, D bar this one.

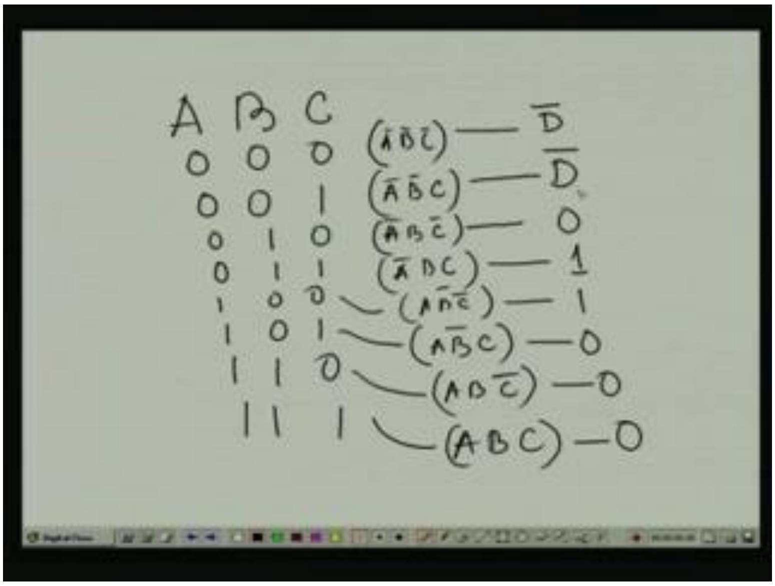
Then, A bar, B bar C D bar which one is that A bar B bar C D bar, so this one is A bar B bar C D bar. Then, A bar B C bar 0 means see my third column is empty, means that there is no C D term or there is no A B term also. We can tell, now A bar B C 1 this is my that second 1 is A bar B C, A bar B and for this couple this is A bar B C and this is D is 1 C exist D is not there so, A bar BC. Now, similarly we can write, we can represent this function like this. So, now what we can do that if I take first thing what we can do that, 4 A B the A, B, C, D the 4 variable can be taken as the select lines.

Then, it will be a 2 to the power 4 to 1 or 16 2 to 1 multiplexer, now I want to optimize this thing, so we have taken A B C are the 3 select lines. Now, if A B C are the 3 select lines, then and D can be my one of the input variables. That means, D, D bar and 0 1 this can be my input variables. And what we can do that A B C there are 8 combinations. So, the functions f I am writing in terms of this 8 possible combinations of A B C.

So, this is A bar, B bar, C bar, D bar which is my this one A bar B bar C D bar A bar B bar C D bar, means this is my this one. Now, next row this is A bar B, now A bar B means this A bar B C D. So, A bar B C D means this particular one and A bar B C D bar means this particular 1 and D plus D bar is 1 means there is not D term exist there. And this is also A bar B C 1. So, if I take together this is my A bar B C.

Now, A B bar C bar A A B bar means, if I take this one this couple, then this will be my A B bar C bar which is this term A B bar C 0 this not there. Then A B C bar 0 it is not there and A B C this is A B bar A bar B bar A B C 0 means A B this is my that third column is not there, third column full A B is zero. So, this term is not there, so now if I see that which combinations of A B C's are there.

(Refer Slide Time: 51:23)



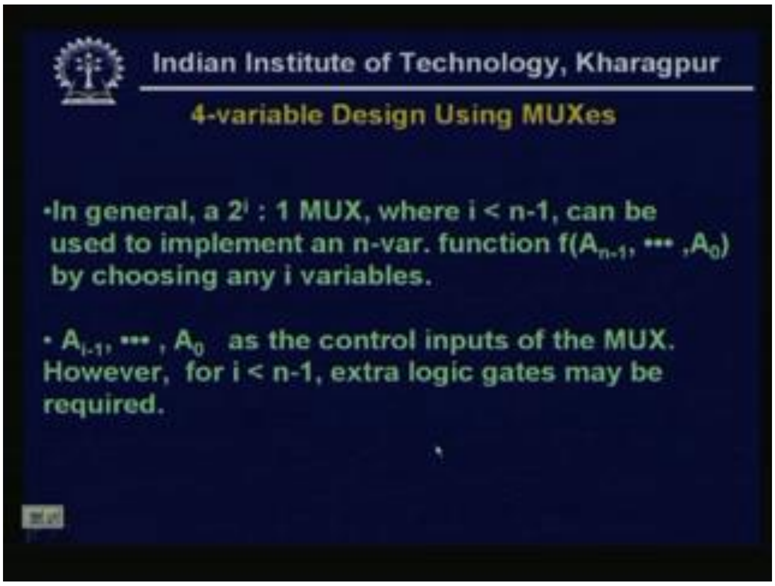
| A | B | C | | |
|---|---|---|---|---|
| 0 | 0 | 0 | $(\bar{A}\bar{B}\bar{C})$ | $\bar{D}$ |
| 0 | 0 | 1 | $(\bar{A}\bar{B}C)$ | $\bar{D}$ |
| 0 | 1 | 0 | $(\bar{A}B\bar{C})$ | 0 |
| 0 | 1 | 1 | $(\bar{A}BC)$ | 1 |
| 1 | 0 | 0 | $(A\bar{B}\bar{C})$ | 1 |
| 1 | 0 | 1 | $(A\bar{B}C)$ | 0 |
| 1 | 1 | 0 | $(AB\bar{C})$ | 0 |
| 1 | 1 | 1 | $(ABC)$ | 0 |

See, now I write the 3 select lines A B C these are 0 0 0, 0 0 1, 0 1 0, 0 1 1, 1 0 0, 1 0 1, 1 1 0, 1 1 1. What are the terms, see this is my A bar B bar C bar, this is my A bar B bar C, this is my A bar B C bar, A bar B C, A B bar C bar, this my A B bar C, this is my A B C bar and this is my A B C. ((Refer Time: 52:19)) Now, I see which are there for A bar B bar C bar D bar is there and for A bar B bar C D bar. So, for this 2 it is C C bar this is D bar, so for this my output should be D bar and for this my output should be D bar.

See, for A bar B C bar output should be 0 and A bar B C output is 1, so A bar B C bar output is 0, A bar B C output is 1. Then, A B bar C bar and A B bar C bar 1 0, A B bar C bar 1 A B bar C 0, then A B C bar and A B C 0 0, then A B C bar and A B C this is 0 0. So, now these 8 will be my primary inputs D bar D bar 0 1 1 0 0 0 and for this combinations 0 1 1 0 0 ((Refer Time: 54:16)).

That means, if I draw that circuit that what will be the thing, that these will be D bar these will be my D bar and D bar D bar 0 1 1 0 0 0. So, these will be my 8 inputs and the 8 3 A B C are the combination, this is my 8 to 1 MUX and which realizes this particular function. So, in this way I can design any combinational circuit to using multiplexer, either A, B, C, D all can be select lines. And just in this examples what we have seen, that we have reduced the select lines by 1. We have taken among the 4 variables, we have taken 3 as say select inputs and the input are treated as the 0 1 combinations of 0 1 and D and D bar. For this particular example, there is no D exist only D bar only D bars are there. So, in this way we can design the n variable combinational circuits using multiplexers.

(Refer Slide Time: 56:06)



Indian Institute of Technology, Kharagpur

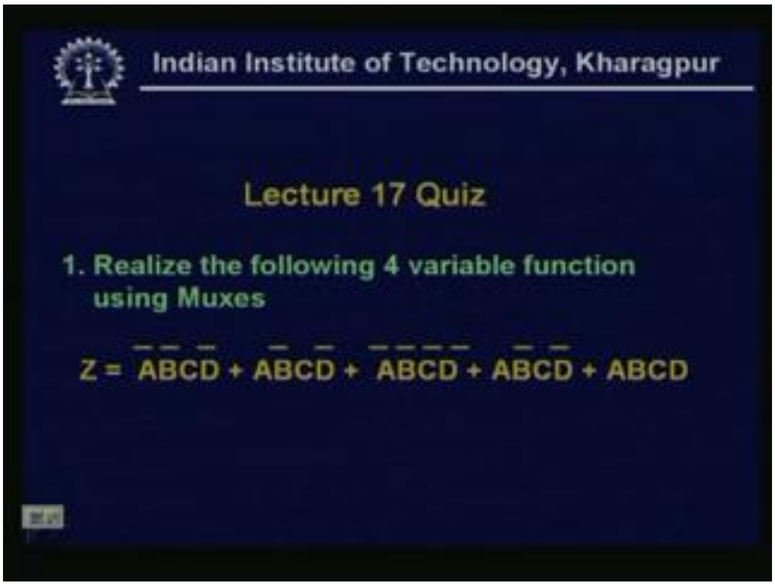
4-variable Design Using MUXes

- In general, a $2^i$ : 1 MUX, where $i < n-1$, can be used to implement an n-var. function $f(A_{n-1}, \cdots, A_0)$ by choosing any i variables.

- $A_{i-1}, \cdots, A_0$ as the control inputs of the MUX. However, for $i < n-1$, extra logic gates may be required.

So, this is my general 4 variable this is my design rule that for 4 variable design using multiplexers.

(Refer Slide Time: 56:29)



Indian Institute of Technology, Kharagpur

Lecture 17 Quiz

1. Realize the following 4 variable function using Muxes

$$Z = \bar{A}\bar{B}C\bar{D} + A\bar{B}C\bar{D} + \bar{A}\bar{B}\bar{C}\bar{D} + A\bar{B}C\bar{D} + ABCD$$

So, we take that the quiz question for this lecture, that realize the following 4 variable function using multiplexers and the function is Z is A bar B bar C D bar plus A B bar C D bar plus A bar B bar C bar D bar plus A B bar C D bar plus A B C D. So, today what we have discussed the design procedure, the design rule using that we can design this thing, so we will end the today’s lecture here.

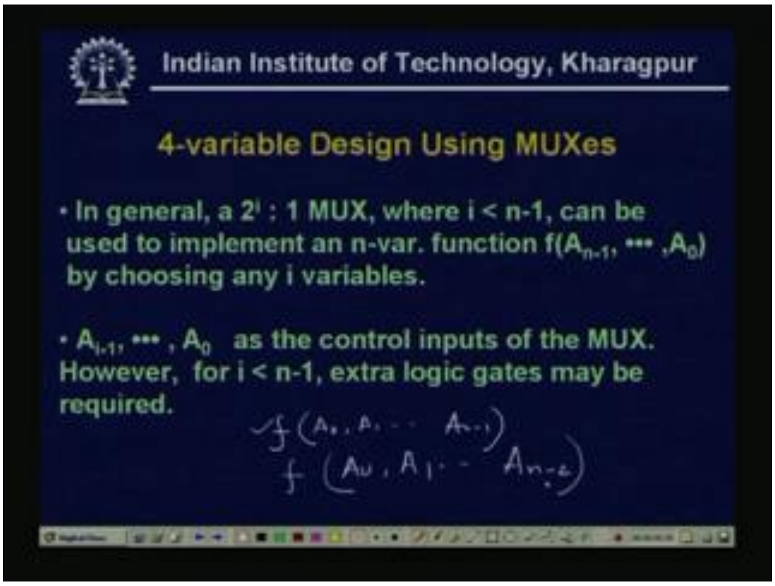
Thank you.

# Digital Systems Design
**Prof. D. Roychoudhury**
**Department of Computer Science and Engineering**
**Indian Institute of Technology, Kharagpur**

## Lecture - 18
## Combinational Logic Problem Design

We are discussing how we can design the combinational problems or complex combinational problems using fundamental circuits like, decoder, multiplexer, adders, etcetera. Last day, we have discussed how we have we can design combinational circuits using multiplexer, some the generalization of that design. Today, we will see some large problem design, but before that we continue some of the 4 variable designs that we are discussing in the last class.

(Refer Slide Time: 58:08)



Indian Institute of Technology, Kharagpur

4-variable Design Using MUXes

- In general, a $2^i$ : 1 MUX, where i < n-1, can be used to implement an n-var. function $f(A_{n-1}, \cdots, A_0)$ by choosing any i variables.
- $A_{i-1}, \cdots, A_0$ as the control inputs of the MUX. However, for i < n-1, extra logic gates may be required.

$f(A_0, A_1 \cdots A_{n-1})$

$f(A_0, A_1 \cdots A_{n-2})$

So, 4 variable design using MUXes, last day what we have seen that in general a 2 to the power I 2 1 MUX, where I less than n minus 1 can be used to implement an n variable function. So, that n variable is 0 to n minus 1 by choosing any I variables, means that if I have a function f n say n number of variables A 0, A 1 up to A n minus 1. Then, first what we have discussed that a multiplexer, where that n number of select lines can be used to realize this function, then always it is possible to design this function.