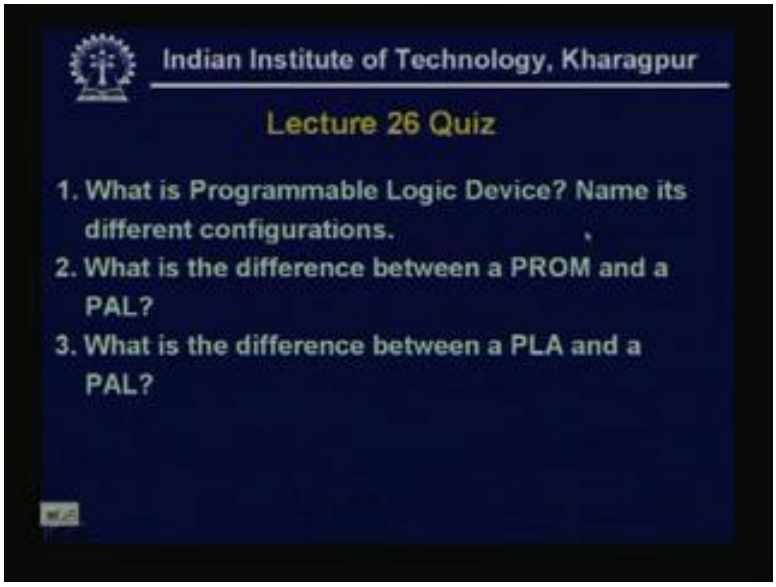
Digital Systems Design
Prof. D. Roychoudhury
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 27
Programmable Logic Devices (Contd.)

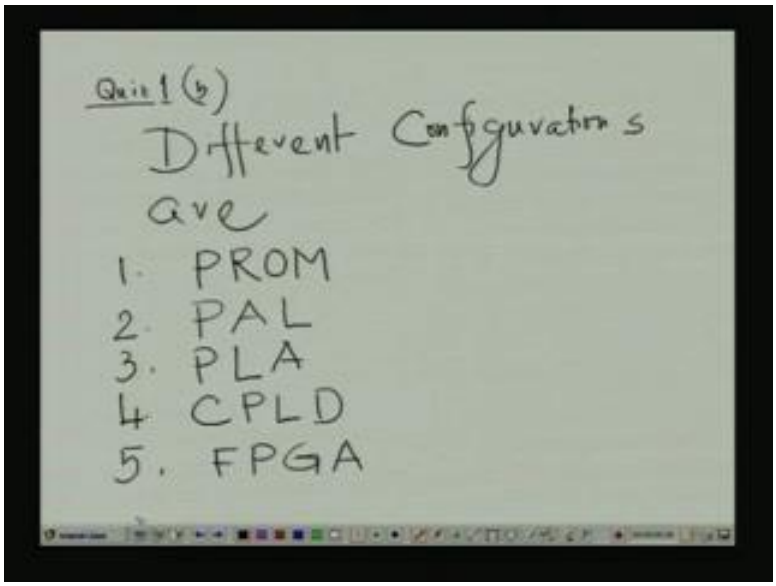
Last class, we have started the discussion of programmable logic devices. So, far we have read, the, what do you mean by programmable logic devices and the different configurations. Now, we will continue in this class, the same thing programmable logic device, but before that quickly we see the last.

(Refer Slide Time: 01:14)



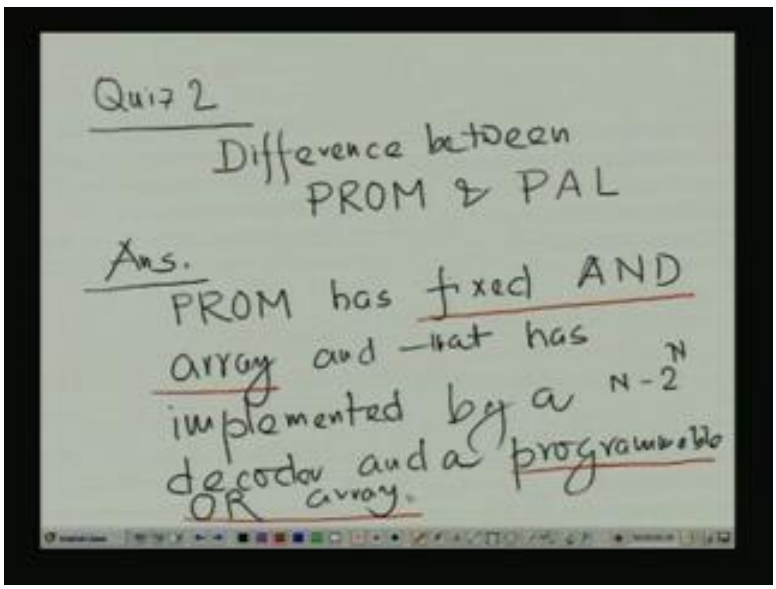
Like the quiz questions of the last lecture, so first question was, what is programmable logic device? Name its different configurations. So, programmable logic device is, we have seen that, it is mainly of some array type of structures, that can be divided into two parts one is called the left hand side designed array, right hand side is OR array and the either one or both, can be programmable to implement or to realize a logic device. This is called the programmable logic device and name it is different configurations. We have read the configurations, first one is programmable ROM, so I first read the question.

(Refer Slide Time: 02:19)



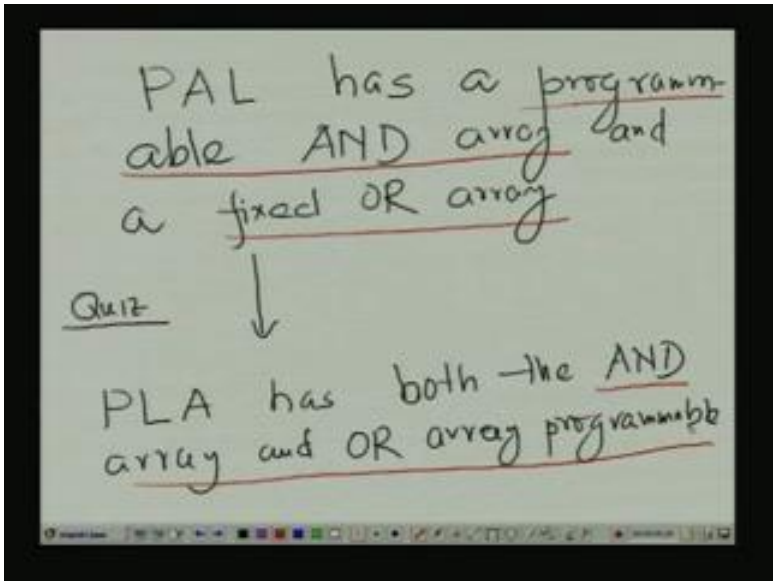
The question quiz 2 or this is question 1, b that different configurations are, one, the PROM, programmable read only memory, second is programmable array logic, PAL, third is programmable logic array PLA, fourth is complex programmable logic devices or CPLD and fifth is field programmable gate array or FPGA. So, these are the five configurations we have read, there, some more can be there in the literature. But they are, many of the most of the cases that part of, or some variety of one of these configurations. ((Refer Time: 03:49)) Now, quiz 2 is, what is the difference between a PROM and PAL? and 3, what is the difference between PLA and a PAL?

(Refer Slide Time: 04:01)



So, again quiz 2, answer is, the difference between PROM and PAL, so PROM, we have seen that, answer, the programmable ROM has fixed AND array and that has implemented by, a into 2 to the power N decoder and a programmable OR array. So, here the main thing is, it is a fixed AND array and a programmable OR array, whereas the PAL, it is totally different reverse or we can tell that, it is reverse.

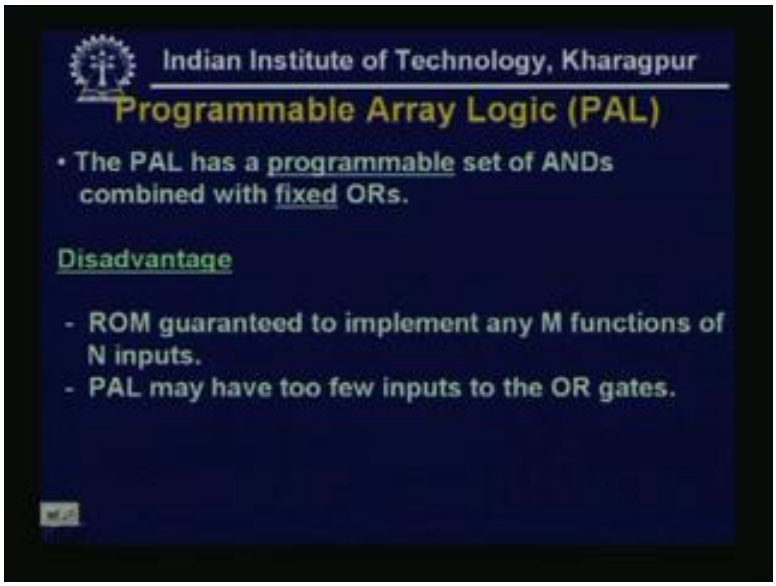
(Refer Slide Time: 06:21)



That, PAL has a programmable array logic, has a programmable AND array and a fixed OR array. So, this we can take it is total, totally reverse thing on the, if we consider the programmability. Now, question 3 was, the quiz 3, it was the difference between PAL and PLA, so already have seen, PAL has a programmable AND array and a fixed OR array. So, these will be as it is, it will be coming as PAL and PLA is, PLA has both the, has both the AND array and OR array programmable.

So, it is more flexible, both the AND array and the programmable. So, it is more flexible, that is all, these are the answers of the last lectures quiz question. So, now we continue the discussion on programmable logic devices.

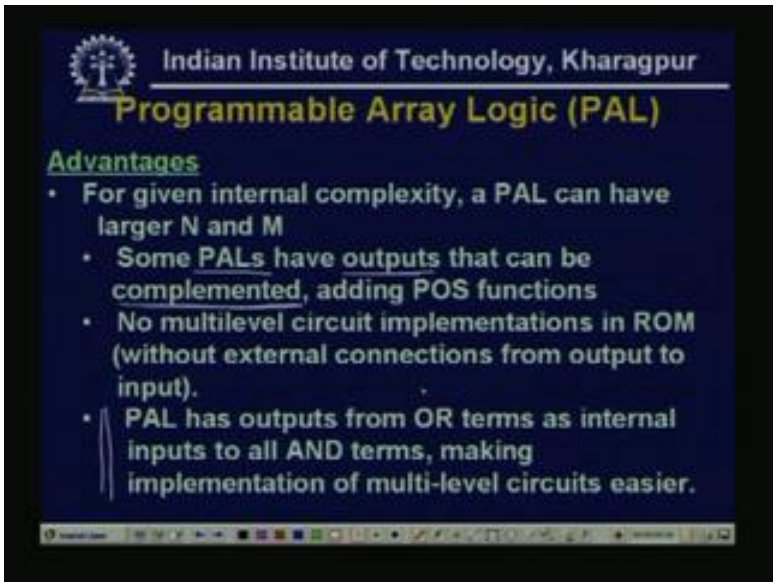
(Refer Slide Time: 08:27)



So, well, we have seen in details or we have read the details of PROM, now we see that, the actual architecture and the design of the PAL, the programmable array logic. Just now, we have seen the PAL has a programmable set of ANDs combined with OR, fixed ORs, that means AND array is programmable OR array is fixed. Now, what is the disadvantage, see, ROM guaranteed to implement any M functions of N inputs, because it has a fixed AND array and we have implemented that thing by a decoder.

So, all for N variable or N input variables functions, all 2 to the power N minterms are available, in the output of the end function, AND array and for this M output, means M different functions are there. So, for the M functions, any set of, that 2 to the power N min terms, that means, all minterms are available, all 2 to the power N is the, all possible minterms, for N input variable. So, one guaranteed that thing, PROM, but PAL may have two, three an inputs to the OR gates, because PAL has a programmable set of ANDs combined with fixed ORs. So, the advantages that it may not realize, the all possible functions, that are available or possible for M output, N input programmable logic devices, so it has some restriction.

(Refer Slide Time: 10:31)

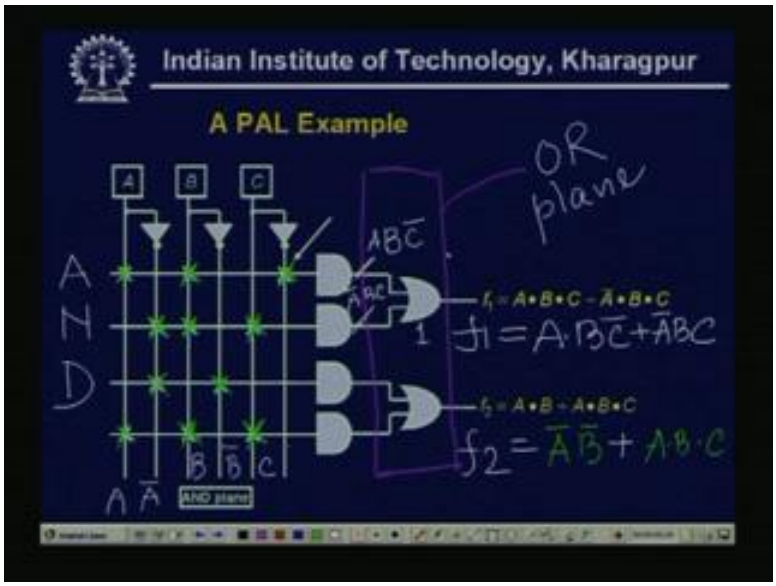


Now, what are the advantage, so given internal complexity, a PAL can have larger N and M, because it has AND fixed or OR fixed and AND programmable. So, it can handle, that larger input variables and as well as OR fixed, so as it is fixed. So, in that design itself, as it is a fixed OR plane. So, that M can be initially itself, that M can be the larger value can be selected, some PALs have outputs, that can be complemented adding POS functions, so this is a very important property.

See, that some PAL functions, that are output functions that are complemented. So here, we get again some more minterms using this. Now, no multilevel circuit implementations in ROM, without external connections from output to input. But as here, the PAL has fixed OR, so as if some more additional circuits if we add to the fixed OR plane, then we will be getting the more, we have the more choices. PAL has outputs from OR terms as internal inputs to all AND terms, making implementation of multilevel circuits easier.

So, here it is possible to get the multilevel circuits and if we take, it is a programmable and though, but we can take worst case or exhausted case, all possible mean terms and then, implementation of multilevel circuits are easier by using PAL. But normally in PROM, the, this type of connections cannot be done, because here, it is a fixed AND and only programmable OR.

(Refer Slide Time: 12:58)



Now, we take one PAL example, so it has a programmable AND plane, so this is the, this is the AND plane, this is my AND plane, see it has 3 inputs and we have taken, this A and A complement, both are available, so this line is A and this is my A complement. Similarly, this is B, this is B complement, this is C, this is C complement, now it has, fixed OR gate, OR plane, so we see, it has fixed OR plane, see, this is the, this is the fixed OR plane and what is there.

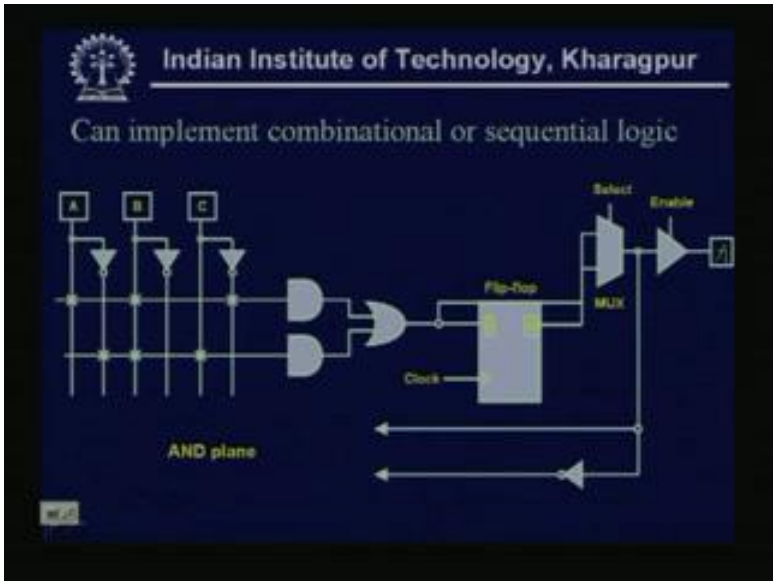
So, in the AND plane realizes the mean terms, as it is a programmable, some of the minterms are as needed or as required, that are in synthesized and then, these are the inputs of AND gate and see, these are then, then realized by, these are the OR. So, actually the fixed OR, we should give the fixed OR plane as, as these two, two input OR gate. So, see here, there are cross are here, so that means, this, this is A, this is a A and cross is here, so this is a B and this is C bar.

So, that means, the output of, output of these AND gate, output of these AND gate is A, B, C bar. Now, similarly the, the second line realizes, see this is, this cross point means A bar, this is B, because B line is here and this is C line, so this is A bar, B,C. So, the output of second AND gate is A bar, B, C, then these are the two inputs of the first OR, this is my first OR gate, so this is A, B, C bar. So, this is nothing but f 1, f 1 is A, B, C bar plus A bar, B,C.

Similarly, the f 2 realizes the OR of, that means, sum of two, the output of 2 AND gates, see the first one, the first input of the first AND gate is, this is A bar and this is B bar,. So, first one is A bar, B bar, now, second input of the OR gate is A, B and this is C, so this is A, B, C. Now see, here this is a fixed OR plane, so that is why, the two output functions, that are already, always should be coming from this a fixed, that means it has two input OR gate. So, if we need a three input or many of the ((Refer Time: 18:22)).

This is not a variable, if I want three input, it is not possible from this structure, if we want more than two input OR gate, that means, two input OR gates means, more than two input OR gates means. If the number of minterms in the function f is more than two, then it is not possible to realize by this PAL structure, so this is a restriction of PAL, it is a fixed. That means, here it is always, it is already defined, already pre designed, that my output function is the sum of two minterms only, but the minterms can be consist of any of the letters((Refer Time: 19:24)) restriction.

(Refer Slide Time: 19:37)

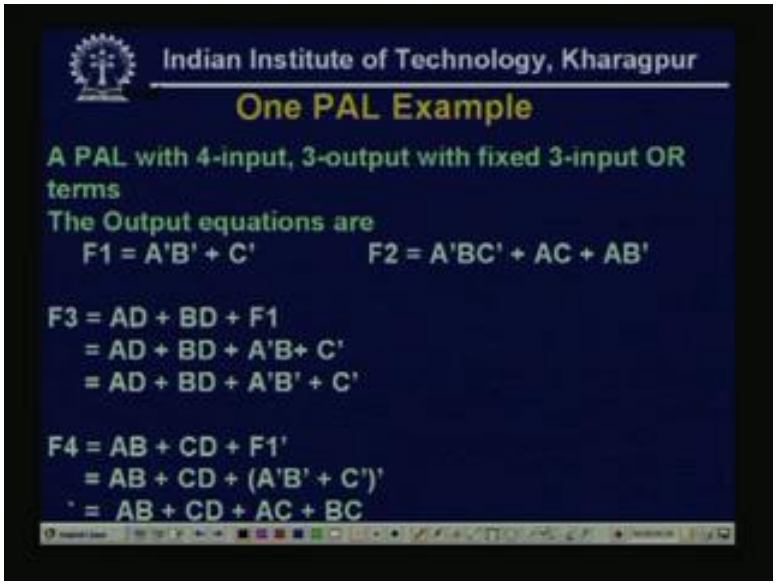


Now, some more advantages also there for this, fixed OR gate, OR plane, see earlier it was only combinational. Now, we can realize the sequential logic also by using a PAL structure, as this output plane, normally OR plane we call the output plane, because the output function is, can be computed or can, is taken from the OR plane. So, the, in the output of the OR, or the PAL plane PAL, we can add some d flip flop, so if we see this example.

See, again we are taking the three input, so this is my AND and the three input variables. So, this is A, A bar, B, B bar, C, C bar and the output of the two AND gate, this is A, B, C bar and this one is A bar B,C, this is A bar B,C. So, this output, it is latched to this, by this d flip flop, which is touched here. So, when the clock is ON or clock is 1, so here we are applying one clock, so when it is one, clock is one, this is latch to this d flip flop and the circuitry here, one circuitry we have taken, that one multiplexer enable.

That means, when we need, it is as if, this is a stored and the F 1 circuitry is, we are getting, another thing is, if I add a inverter here, see this is one inverter. So the function, if it is enabled we are getting F 1 and the function can be taken as the, as the complemented of that thing, F1 as well as F 1 complement is also possible. So, adding a d flip flop or more than one flip flops, depending on the circuit complexity, what we can do, we can realize the sequential circuits also by using the PAL logic. So, this is one, good advantage over the PROM.

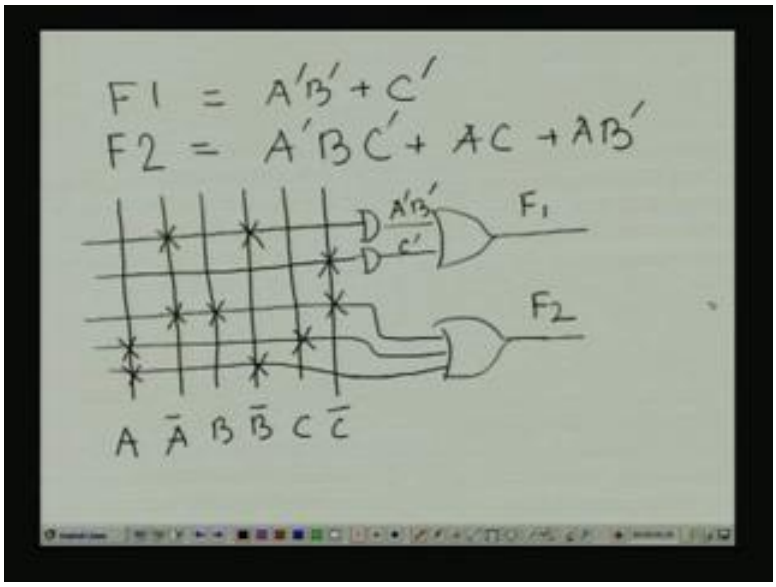
(Refer Slide Time: 23:00)



So, here, if we see one example, of another PAL example, that how we can realize this thing, say PAL with four input and three output, with fixed three input OR terms and output equations, there are four output equations, F 1 is A dash B dash plus C dash, F 2 is A dash B C dash plus AC plus AB dash, F 3 is AD plus BD, actually AD plus BD plus A dash B dash plus C dash. So, first we realize F 1, F 2 and then we will be seeing that F 3, F4.

Actually, this is was special type of functions, where the two output functions, consist of other two in output functions, see F 3 consist, F 3 consist, F 1 also, A dash B dash plus C dash is nothing, but F 1, this is, this is F 1. Similarly, F 4 consist F 1 dash was, see if it is A dash B dash plus C dash dash, so, this is F 1 dash. Now, we see that, how we can realize this thing, so first write, A dash B dash plus C dash and A dash BC dash AC plus AB dash.

(Refer Slide Time: 24:51)

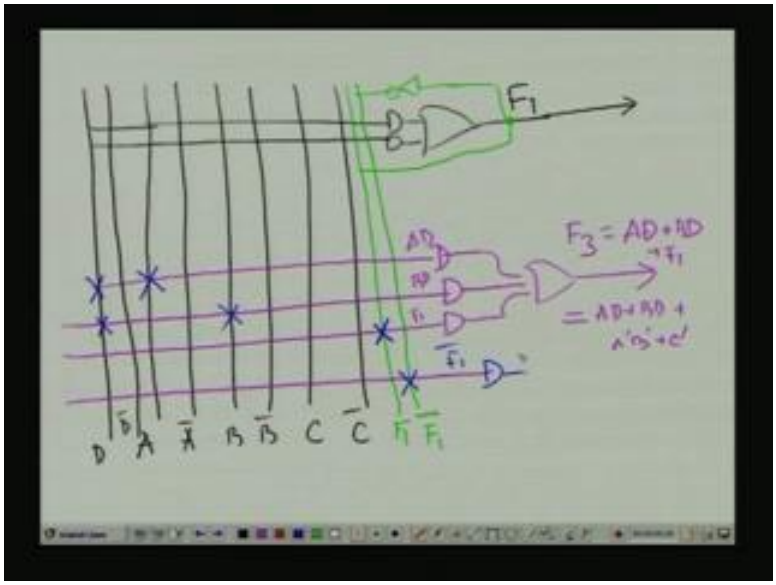


C dash AC plus AB dash and F 1 is A dash B dash plus C dash, so first we, how many, there are three inputs here A,B,C. So, if we draw the programmable OR plane, AND plane, so these are, there will be six, because six signal lines, because this is for A, this is A dash, this is B, B dash and C, C dash. So, this is A dash B dash plus C dash, so one cross will be here, one cross will be here, B dash, so this is, if I put one AND gate, that this is A dash B dash.

Similarly, see it is only C dash, so here only, this is C dash, now, we give a OR gate and it will realize, the F, F 1, it will realize the F 1 circuitry. Similarly, if it is F 2, we are seeing that, there are three, so that means, these OR gate must have three inputs, must have and this will be A dash BC dash, so this is A dash BC dash AC, so this is A, this is C, AC and then AB dash, so this is my F 2. Now, what we see that ((Refer Time: 27:44)) the F 3 is actually AD plus BD plus A dash B dash plus C dash, A dash B dash

plus C dash is nothing but F 1. So, this is, F 3 is AD plus BD plus F 1, so now what we can do, if we take some feedback type of thing, see again, from F 1 can be a input line.

(Refer Slide Time: 28:23)



So, if we draw again, then actually earlier, we haves three input variables, three input variables, from there we have realized F 1. Now, see if this F 1 is fed and it is taken as another, so this is my A, A dash, B, B dash, C, C dash and this is my, this is my F 1 and similarly, I can add a inverter and this is my F 1 dash. So now, now F 2, F 2 was realized, F 3 is AD plus BD, so F 3, I can realize, how F 3 is AD. So, A, AD plus BD, so I need another input D and D complement.

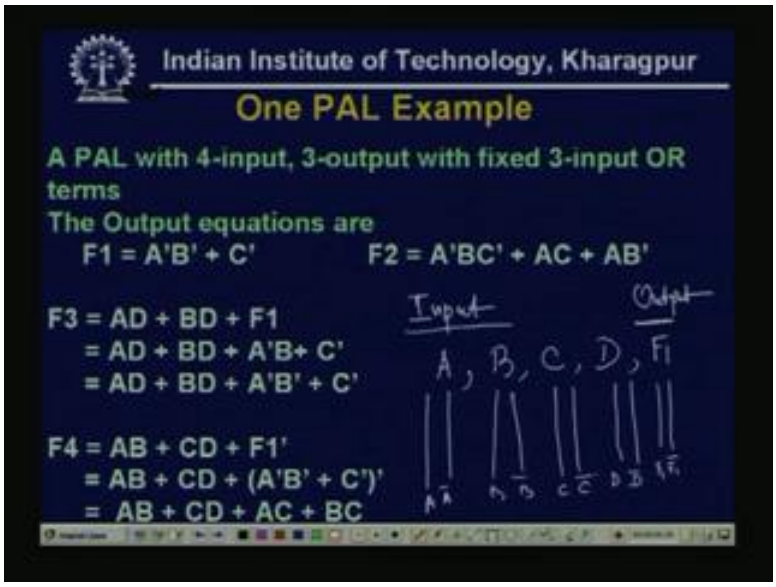
So, now this is my D, this is D complement, so this is my AD, another is BD, so this is D, this is B, so these two are, this is two AND gate, this is AD, this is BD. Now, AD plus BD plus F 1. So, I need another, this is, as now here F 1 is also a, F 1 becomes an input line. I have taken the output of the first OR gate, that is the function F 1 as the input of the, one of the input variable. That means, it is fed back to the programmable AND plane.

So, one output of the fixed OR plane is taken or is fed back to the AND plane as the input variable. So, this is my F 1, so now, this becomes my F 1, so now I, if I give a OR gate here, the F 3 will be AD plus BD plus F 1, which is actually AD plus BD plus A dash B dash plus C dash. Similarly, F 3 is nothing but the, the some function that is F 1

dash, in that case, I will be taking one another line, which is taken that F 1 dash has the input, so this is my F 1 dash, this is F 1 dash.

So, here one advantage we have seen or one special type of PALs structure, where actually the sum of the output, of the fixed OR plane k is again fed, they are fed to the input of the programmable AND plane. And in that way, it actually extends the or increases the, programmability, as well as the design variety of the PLAs.

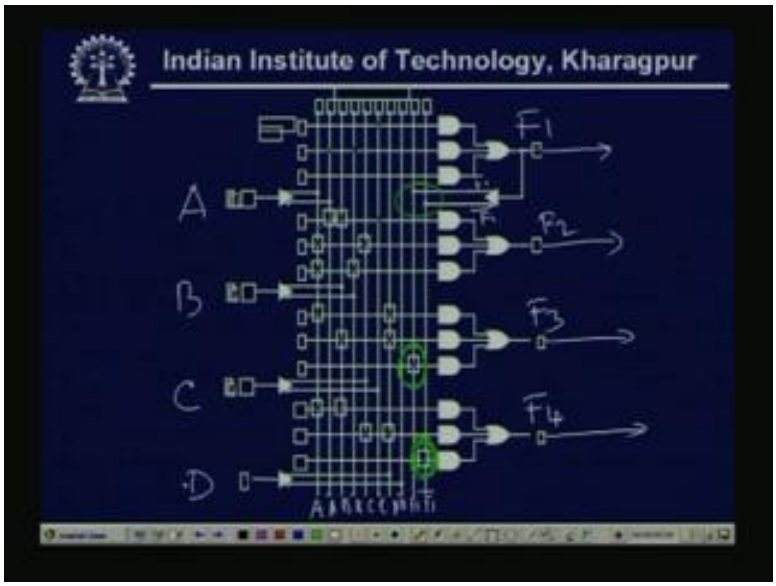
(Refer Slide Time: 33:53)



So, we can realize F 1, F 2, F 3, F 4 by using four input variable A, B, C, D as well as, the F 1 is again, as the one of the output function, F 1 as the F 1 and F 1 complement as the input variable. So, here the number of inputs are, the input variables are A, B, C, D and F 1, so that means, for A A bar, for B there are two lines B B bar, for C, C C bar for D, D D bar and for F 1, F 1 F 1 complement. And actually, they are coming from the, as the, these are, this is the output.

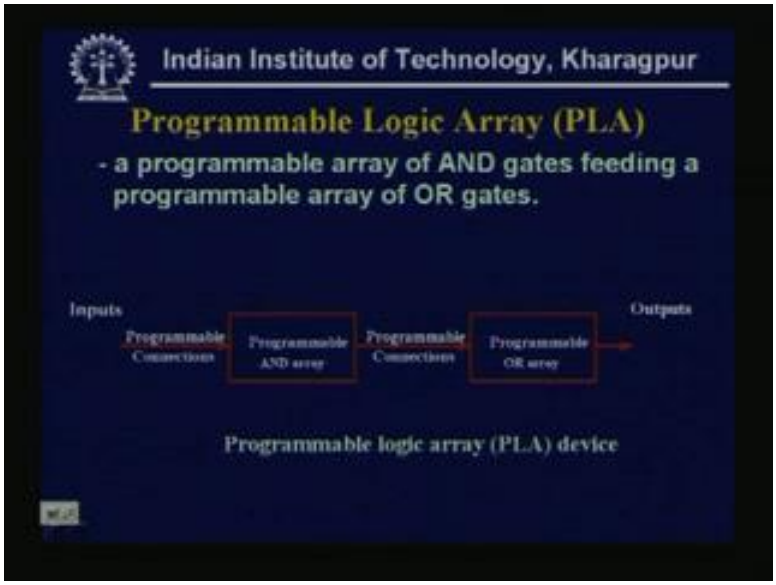
So, five, input with the, in the function, these are five input variables, but actually from outside, only four input variables are there. So, in this way, the design is compact also, from outside, there are four, are four input variables and one in, one variable or one input is actually coming from the design itself, which is the another output of that particular design.

(Refer Slide Time: 35:53)



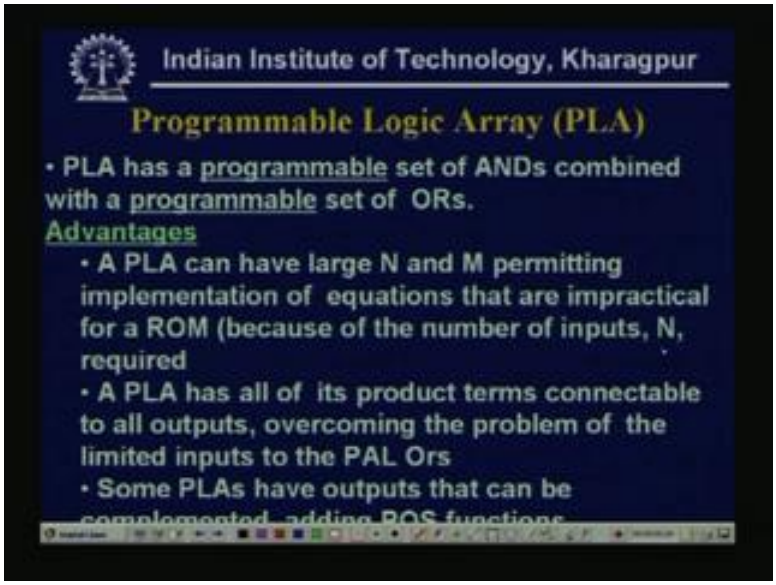
So, this is just now the structure I have drawn and see, this is actually, the A, A complement, B, B complement, C, C complement, D, D complement and F 1 and F 1 complement. So, this is, that F 1 and F 1 complement, this is my F 1, so these are the two feedback lines, this is F 1, this is F 1 complement and rest of the things are same, this is F 2, F 3, this is F 4, where in F 3, F 1 is one of the inputs and in F 4, F 1 bar is one of the inputs. These are the design of the previous functions.

(Refer Slide Time: 37:35)



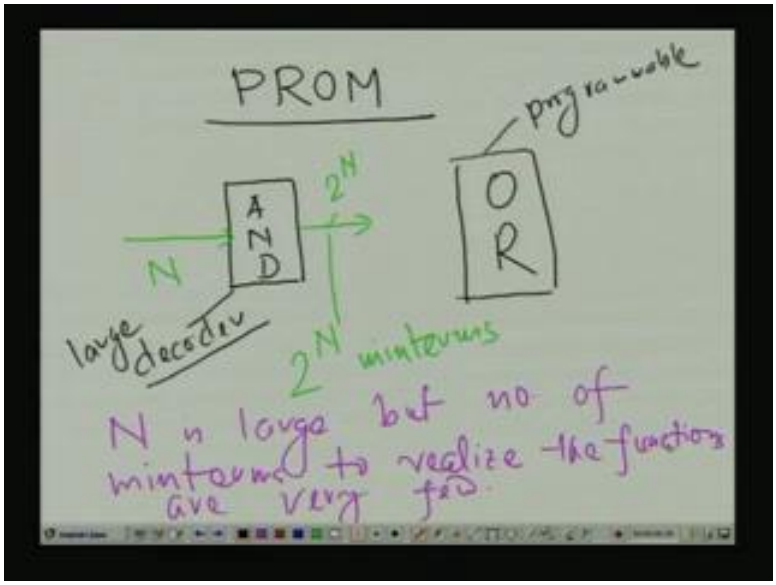
So, the PAL structure and the compactness or the flexibility of PAL to realize the sequential circuits, as well as the reuse the, of some of the functions already realized to get a new functions, that facility we have seen in PAL logic. Now, we see that programmable logic array or the PLA and as I already I mentioned, PLA has the, is the most, the programmable features of PLA is the maximum. So, because it has the two planes, of the both the planes are programmable. See here, the AND plane, this is my, this is my AND plane, this is also programmable and this is my OR plane, that is also programmable.

(Refer Slide Time: 38:42)



Now, what are the advantages, so PLA can have large N and M, permitting implementation of equations that are impractical for a ROM, because of the number of inputs, N required. So, here we have seen that, for ROM, that particularly as it is a, for ROM it is a fixed AND array. So, we have implemented by using, that is by a decoder and a larger decoder means it is, it will take a large area, it will be impractical, whereas, now, a now actually in real life, here the problem is, that in real life, often we do not need all of the 2 to the power N, mean terms.

(Refer Slide Time: 39:44)



So, if you remember that, the PROM, if you remember, that in PROM that, this is a fixed AND array, it has, say it has N inputs and all 2 to the power N. So this output means actually, these are here, all 2 to the power N minterms can be realized and then it is fed to the OR plane, this is programmable. Now, see here it is N, N, any N, N can be very large, see if N is large, but number of minterms to realize the functions, are not so, so larger, they are small, they are few, are very few.

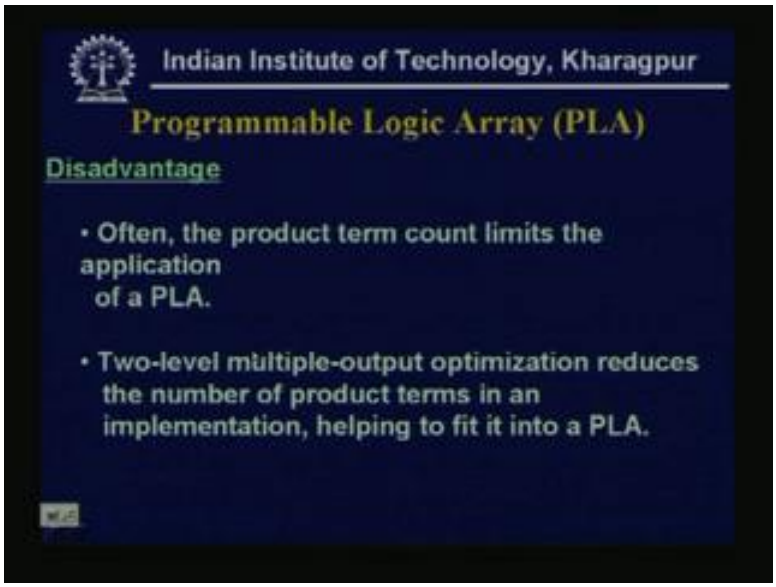
See in this case, as N is large, so we need a, this is actually a decoder, a N triple N decoder and as N is large. So, we need a large decoder, but here only, the minterms are very few, so this is a wastage, area would be very big, design this is a, the pre-define structure of the decoder is very large, but actually I have to realize, say only very few minterms. So, this is a very impractical in real life situation, ((Refer Time: 41:46)) whereas this is a, in this PLA, this can be a very compact thing.

That, even N is large we need only 2 N lines, because both the literals and it is complement. So, if we, I have N variable, then 2 N numbers of actually, signals for the N number of variables and it is complements, that many lines are needed, but it can realize very efficiently, that if the minterms are very less. So, if N is large, but the minterms are very less, then actually the PLA is a very efficient structure rather than PROM. Now, a PLA has all of it is product terms, connectable to all outputs.

Overcoming the problem of the limited inputs to the PAL, PAL ORs, see, that same min term, a PLA has all of it is product terms connectable all outputs. So, same min term can

be used by more than one functions, this is the advantage, some PLAs have outputs, that can be complemented. In this case, now in some special type of design that outputs can be complemented also.

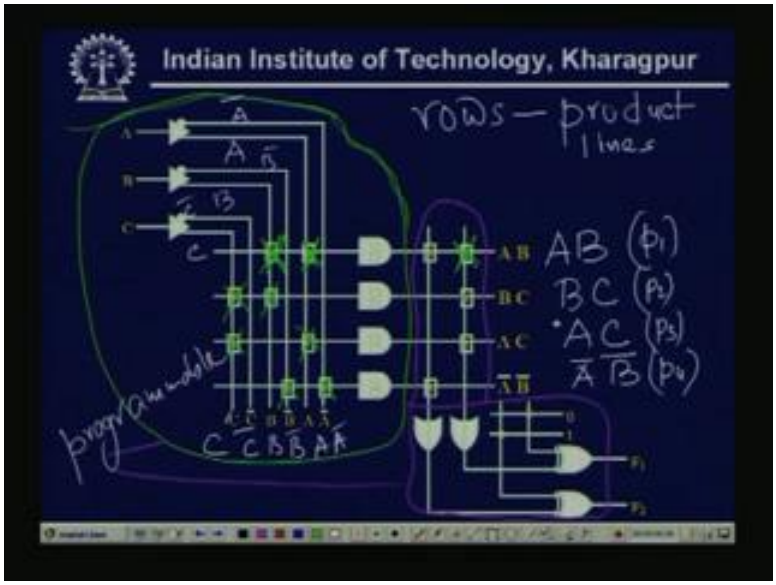
(Refer Slide Time: 43:32)



Now, what are the disadvantage, so often the product term count, limits the application of a PLA, if the product term, product terms are big or the, or, or it is very large number of product terms are needed, then it limits the application. Now, two-level multiple output optimization, reduces the number of product terms in an implementation, helping to fit into a PLA, but it has some number of improved or refinement is there, using the PLA a design and this problem can be overcome.

Another thing we can do, we call the PLA minimization, so in the first iteration, it may happen that, a number of product terms are large and or, of that type of situation what just now I mention for PROM. That, if N is number of input variables are very large, but the few product terms are actually used, then in that case, the size of PLA, that can be reduced also, so this is the PLA minimization it got, got, now we see one example.

(Refer Slide Time: 45:28)

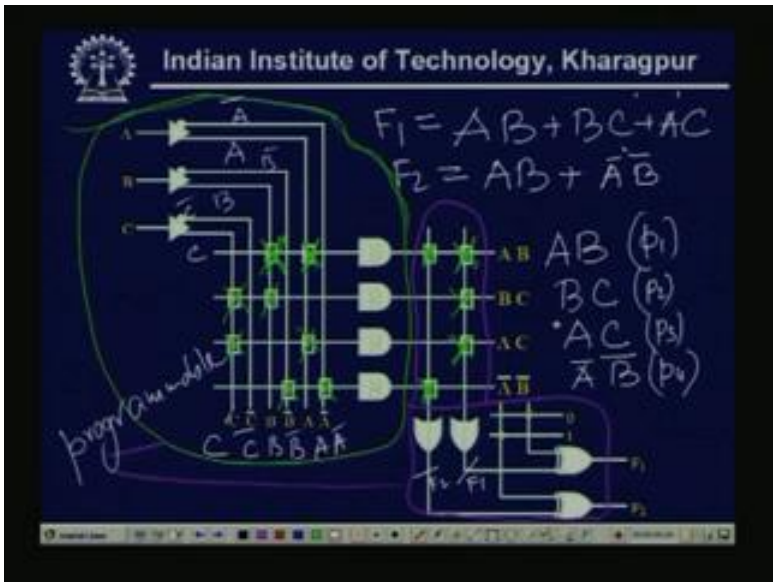


See, the PLA has both the AND plane and OR plane programmable, so this is my AND plane and this is the OR plane and both are programmable, so both are programmable. So, it has three inputs A, B, C and just like what we have seen that, both A and A bar are available that means, this is my A and this is A complement, B B complement, C C complement and that are given, we can write here also, this is C, C bar, B, B bar, A, A bar.

Now, as the cross points are here, so the first, normally we call these are the product lines, so the rows are called, the rows are product lines, because they realizes the mean terms, which is nothing but the product of the literals or the variables. So, these are, rows are products lines, so the first product line or the first row realizes B and A means AB. So, the first one is the AB, similarly the second one is CB, so this is my second product term is BC, third similarly, third product term is C B bar CA, this is AC.

And fourth product term, this is AC and similarly the fourth product term is B bar A bar, means A bar B bar, so AB, BC, AC, A bar B bar, these are my four product terms. Now, in this particular example, we have see that, this is my OR plane, this is my, this is my four inputs, the inputs are p 1, AB is p 1, p 2, p 3 and p 4. Now, what are the, for F 1, see this is my, p1 means AB, BC, AC, p 1, p 2, p 3, all three are the inputs of the OR.

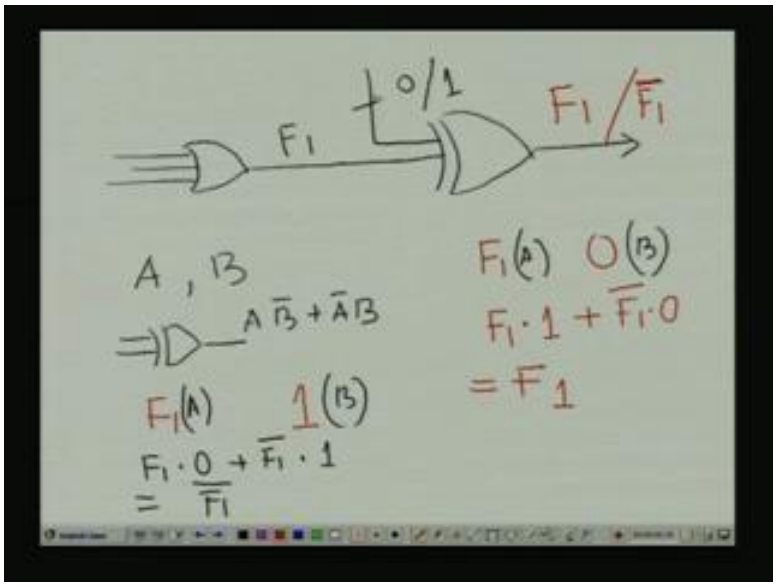
(Refer Slide Time: 51:06)



So, my F 1 is, F 1 is, see AB, AB plus BC plus AC, now we are assuming AB plus BC plus AC, we are assuming the other, this is, this is actually here F 1, I am considering the OR gate output of the first OR gate and if I assume my F 2 is here, then my F 2 is, F 2 is the sum of two mean product terms p 1 and p 4. So, p 1 plus p 4 and p 1 plus p 4 is AB plus A bar B bar, so these are the two functions F 1 and F 2. Now, some more modifications I can do here.

See, what we have done in the OR plane, this is some extension or some refinement of the PLA structure, we can add some EXOR and other input, can be A 0 or 1. So, if we, we see in details that structure.

(Refer Slide Time: 52:54)

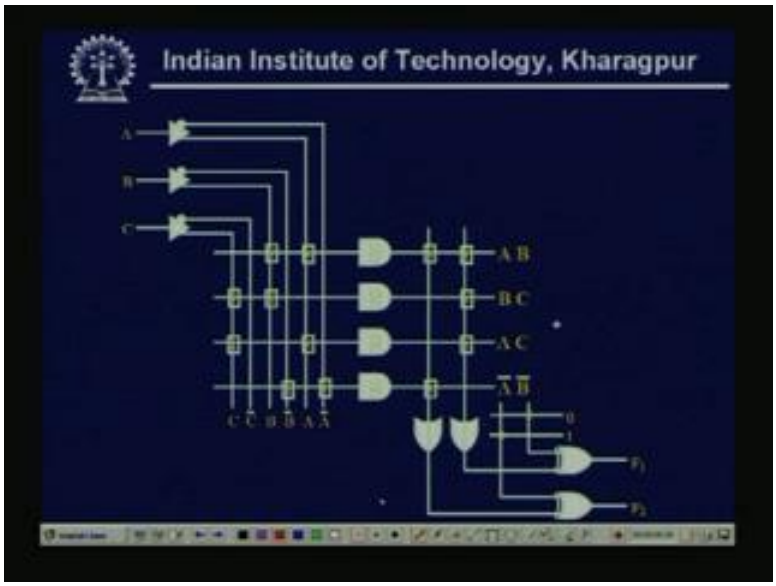


See, what we have done, now the, as if, the one we are considering, the output of this OR gate is a F 1 and that, we fed to the, one of the input of my EXOR, other input can be 0 or 1. Now, if we remember, if we remember the two input EXOR, see if A and B are the inputs of a two input EXOR, then the function will be AB bar plus A bar B, so here, if it is F 1 and 0 is in this line, F1,0. So, that means as if, this is my, see F 1, so this, as if this is my A and this is my A and this is my B.

Then, it should be AB bar plus A bar B, AB bar means F 1, B bar means 1, 0 bar means 1 plus A bar means F 1 bar and B means 0, so this becomes F 1. So, that means, if it is a 0, then as it is F 1 will be, so my EXOR input, if the other EXOR input is the 0, then as it is, I will be getting F 1 here. Now, if it is 1, then we see, if it is F 1 and if it is 1, that means, this is my A, this is my B and then AB bar, then F 1, B bar means 0 plus A bar means F one bar and B bar, B means 1.

So, this becomes F 1 bar, so if it is, if it is 1, EXOR is 1, I will be getting a F 1 bar, so in this way, I can get a complement of the output function by, actually this is another programmability, we have added here, just feeding is 0 or 1. So, this can be again a control line, if it is 0, as it is I will be getting the function of what actually I want or the output of the OR gate and if I want the complement, then my control line is 1, then we, I will be getting a complement of the output.

(Refer Slide Time: 56:40)



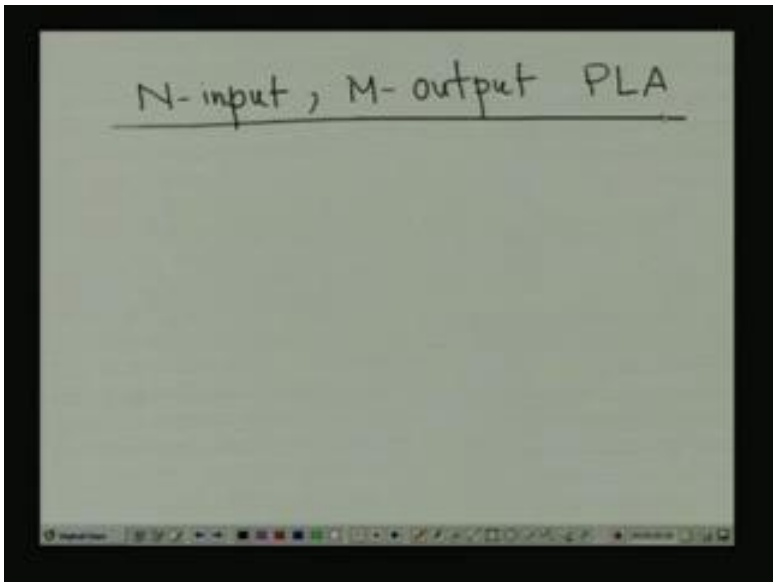
So, here, that OR plane gives me, that this functions, that F 1 and F 1 or F 1 complement and here F 2 or F 2 complement. So, this is my PLA design, a PLA design for a particular, next day we will be discussing that one, how the general structure of a PLA, that means, for N input M output equations, how the PLA can be designed. So, will continue, our this discussion in the next lecture

Thank you.

Programmable Logic Devices (Contd.)

In the last class, we have read the design of a programmable logic array, we have seen that, how one PLA can be designed for a, set of output functions. Now, today we will continue the discussion on the programmable logic devices, first we will see, that how the, what will be the general structure of a PLA design.

(Refer Slide Time: 58:24)



So, if we consider the, one N input, M output PLA, then what will be the programmable AND array and the OR array structure. So, first we will consider the programmable AND array, as it is a N input, so there will be.