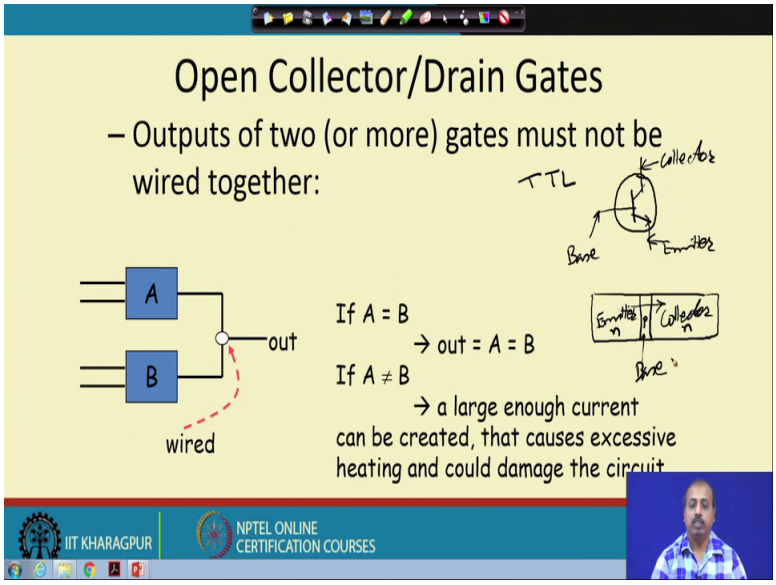
**Digital Circuits**
**Prof. Santanu Chattopadhyay**
**Department of Electronics and Electrical Communication Engineering**
**Indian Institute of Technology Kharagpur**

**Lecture - 27**
**Decoders, Multiplexers, PLA**
**(Contd.)**

Next we look into another problem that occurs when we are connecting outputs of 2 or more logic gates together or say logic blocks together.

(Refer Slide Time: 00:26)



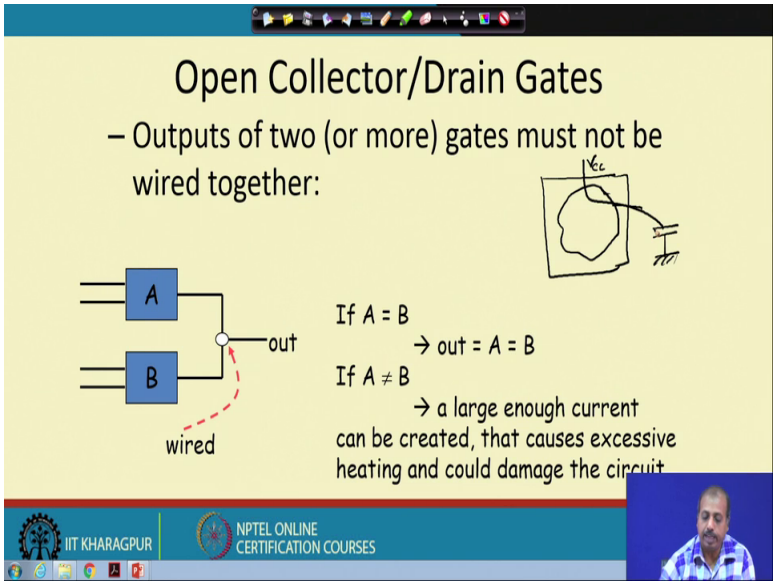

So, this comes under the broad heading called open collector or open drain gain. Now this, so, this if you remember this term drain we have seen when we talked about this CMOS transistors and this collector. So, collector term we have not seen, so this is basically, this is basically coming from the fact that previously if you look into the in the logic families, so there was a family called TTL Transistor Transistor Logic and in the transistor transistor logic, so the circuits were realized by means of transistors. So, a transistor has got 3 terminals, one is known as the base terminal so this is the symbol of a transistor this is the symbol of a transistor. So, this terminal is called the base terminal, this is called the collector terminal and this is called the emitter terminal ok.

So, architecture wise it is like this so we have got this, if you say that we have got a silicon vapor then some part of it is the emitter then after that we have got a thin base, so

this region is called base and then this is called the collector. Now this particular transistor that I have drawn here this emitter is of n type, base is of p type and this collector is of n types, so this is an n p n transistor it is called ok.

So, electrons they go from emitter cross this by base and reach the collector region so that way the current flow is done. Anyway so, that is a different issue so this logic gates can be realized using this type of transistors also because, that also acts as switch, now the current that can be drawn like say if I have got some circuitry here.

(Refer Slide Time: 02:36)




So, if I have got some circuitry here which is say TTL circuitry and then the so it is connected to the supply voltage which is in general known as VCC So, this is the output of the, this is the some logic is realized here and this is output of that logic and it is feeding some load here, load is generally shown as a capacitive load because the charging current is necessary for taking the particular signal point to logic high. So, the amount of current that can be drawn here is limited ok.

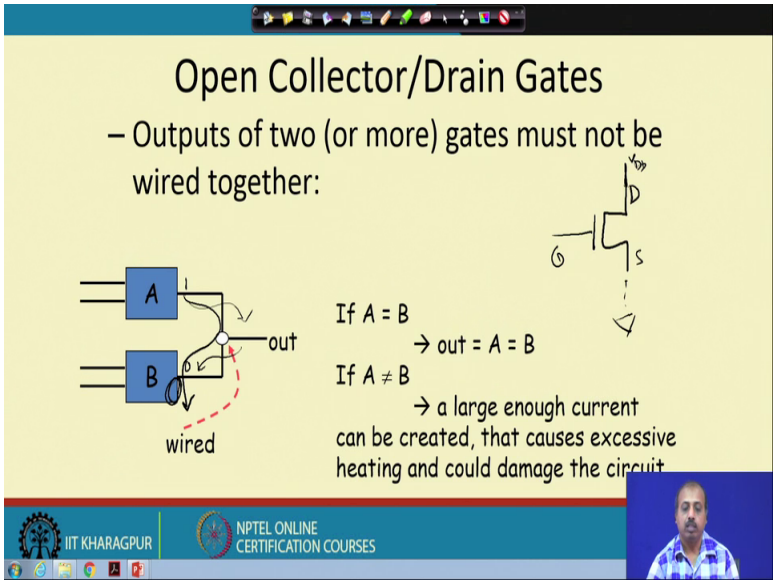
So, it is limited by the TTL family TTL logic family so it is not very high. Now so, if you have to connect large number of input large number of inputs from this outputs so it creates problem. So, another difficulty that we have with this TTL family of logic is that suppose here we have got 2 blocks A and B, we have got 2 blocks A and B and then. So, this outputs of A and B so they are shorted together now if A equal to B then output is so there is no conflict so A and B are same so output also is equal to that value.

However, if A is not equal to B so, what can happen is that say A equal to, A output is 1 and this output is 0, so that means, A is trying to pump in current like this and B is trying to sync current like this. So, as a result there is a there may be a large amount of current flow through this circuit ok.

So, a large enough current that can be can get created and this current may be so high that this transistors that are there in the lower part of this B the logic, so that may not be with may not be able to which stand that current. So, it may damage the circuit, so a large enough current can be created that causes excessive heating and could damage the circuit, so what is the way out ok.

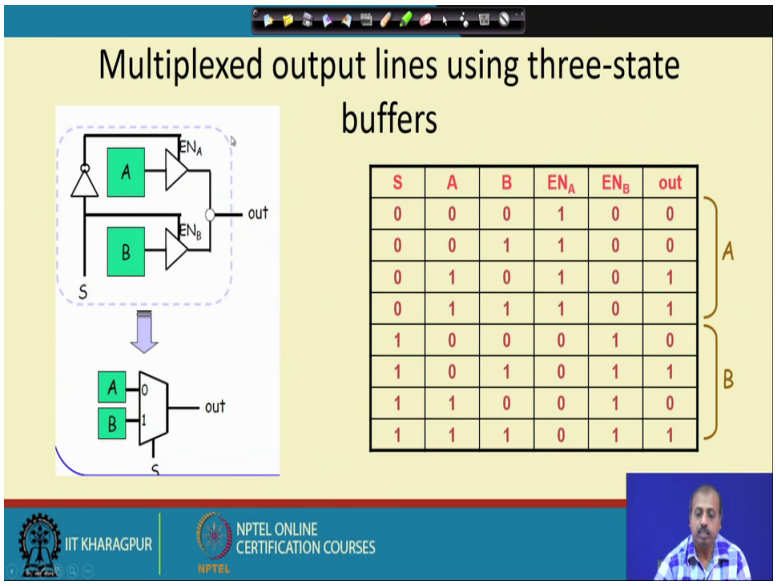
So, this open collector is a solution to this problem and so this is also there are there are on the CMOS family if you look into so they are called open drain connection. So, open drain so you know that in CMOS it is in CMOS there are 3 terminals like this one is called drain another is called source and another is called gate ok.

(Refer Slide Time: 05:11)




So, here also with the if this drain is connected normally it is connected to V DD side and this is ultimately connected to the ground and this G is getting the input. So, in the process, so it may so happen that I mean some of the logic family so this drain is not connected to the supply voltage, because of the same reason that if you have to drive a large load it may create some problem.

(Refer Slide Time: 05:41)

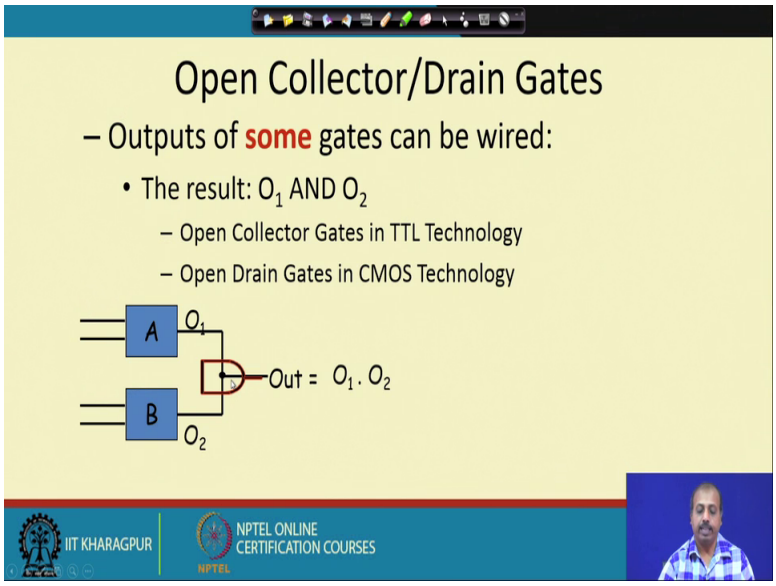



So, we will see how this can be solved like say here so 1 possibility is that we can use some sort of multiplexer, so multiplexed output lines using 3 state buffers. So, this solution we have seen some time back say like this A and B. So, these are the 2 block output and we put some tri state buffer here this is controlled by EN A and this is controlled by EN B and this S this actually for multiplexer it is select since in the multiplexer, so it will select one of these 2 outputs and that will go to the outline. So, if S equal to 0 then this input is equal to 1 and this enable B is equal to 0.

So, this B equal to if S equal to 0 so this is equal to 1. So, as a result this a output is passed through this buffer to the output. Similarly if S equal to 1 then enable B is equal to 1 and enable A equal to 0; as a result this output of B it passes through this buffer and it comes to the output. So, if you look into the truth table you see that for this combination when enable A equal to 1 and naturally in that case enable B has to be 0.

So, whatever be the value of A that comes to the output and similarly if my enable A is equal to 0 and enable B has to be 1 in that case then whatever be the value of B that comes to the output. So, this way we can realize this we can we can use this 3 state buffers or tri state buffers for DLF buffer this type of multiplex realization. But you see that this is, this does not always solve the problem because, many times we do not want this type of multiplexing we just want some logic by which that will be followed when this type of shorting of 2 lines are there.

(Refer Slide Time: 07:36)




So, they come under this under the broad heading called open collector or open drain gates. So, outputs or some gates maybe wired like this so if it is A and B, so they are connected like this.

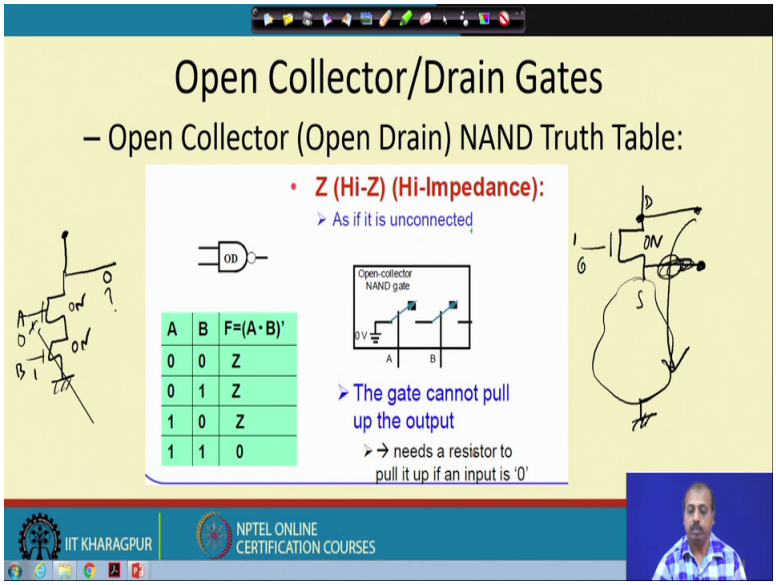
Now, this so this what is the logic that you get at this point. So, that is the functionality that you get is dependent on the logic family that you are using; for example, if you are using TTL technology then this connection so this will result in to and operation, so O 1 and O 2 so you will get the AND operation similarly for open drain gate since CMOS technology.

So, that will you will get a get an AND operation ok, so you will get a output as the and of these two. Now can I connect this O 1 and O 2 like this so logic level is fine so logic level logic value I get as and of this and something or some things like that. So, if I do not use open collector if I use normal TTL gate or normal CMOS gate, then also I will get the output as O 1 O 2, but the problem that will come is the current flows through the lower device, the current flows through the through the devices that can occur if one output is high and another output is low.

So, for that purpose we need some protection, so what we do we put a tri state gate here. So, this is the, this so this is known as the wired and so this AND gate is physically not there. So, it is just the 2 wires that are connected that are sorted at this point, but it

essentially gives the AND operation ok. So, this gate is the wired AND gate we say and we but physically there is no such AND gate.

(Refer Slide Time: 09:21)




So, open collector or open drain NAND truth table is like this, so if you this is open drain so if this A B is 1 1 then only output is 0, so since the problem with his open collector or open drain is that you see if this is your open drain gate this is the open drain gate so this is my drain, so this is gate this is source and whatever be the connection so and this lower part of the logic. Suppose this gate is equal to 1 and that we are taking the output from this point sorry not from this point we are taking the output from this point.

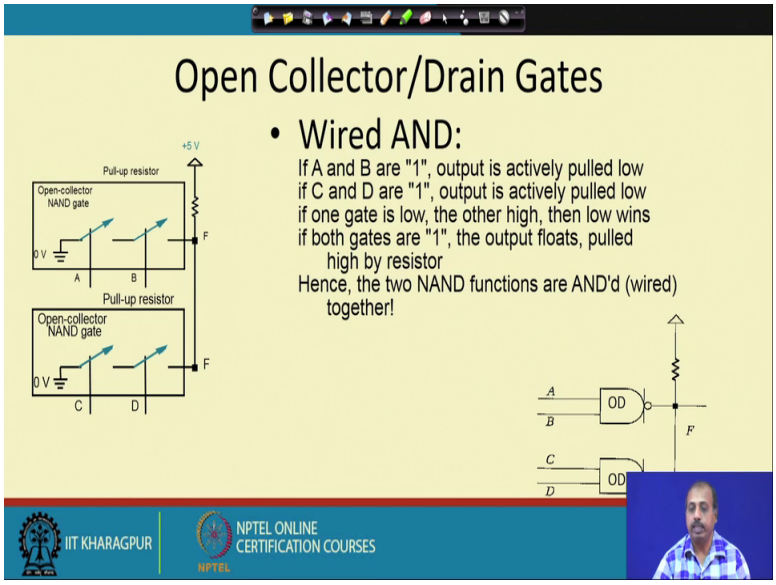
Now, if this transistor is say oh if this transistor is say on then this side is eventually connected to ground so this will be going down to 0. So, if this NAND gate both the inputs are 1, then this then there is a current flow path; like in the NAND operation this lower n network is shown like is like this, so this is A and B. So, we have got so this is grounded and suppose this is open drain.

So, I am connecting it, so this side is open it is not connected to V DD, now what will happen is that if I put a 1 1 here. So, these 2 transistors will be on, so this is also on and this is also on as a result this value voltage will be come this point voltage value will become equal to 0. So, for 1 1 I am getting is 0, but if I if any of them is equal to 0 sorry if any of them is equal to 0, then I do not know what is the value here because that that

side so this lower side is cutoff and upper side is also not connected to any voltage source so this is open fine.

So, this is as it as it so it is the it is high impedance or high Z as if it is unconnected. So, this if the switches so this switches are connected both of switches are equal to 1, then this switch will drop down as if I will get a connection to 0 so this way it operates. So, what to do if I do not have this out if I do not have any connection to the supply voltage? So, what it says is that you need a resistor to pull it up when one of the input is equal to 0, let us see how this is done.

(Refer Slide Time: 12:02)




So, this is the situation so we have got this point. So, we have got this if this A and B both of them are 1 then definitely this F point is getting connected to 0 and it is getting 0 here. So, but if it is if both of them are equal to 1 if both of them are equal to 0 in that case what I will do, I will be pulling this to this point to high via this register ok.

Similarly, so if ABCD all of them are equal to all of them are equal to 0 ok, if all of them are equal to 0. Then so all these switches are open, so in that case at this addition of this extra register here so that will ensure that at this point you get a 5 volt value. So, this so this is outside the chip, so this register and the supply so this is outside the chip. So, from inside the chip I am getting just a this switches as open. So, this point this line is open similarly this line is open and by means of this external register and this pull up this voltage value. So, I am getting a 5 volt value nearly 5 volt value at this point.

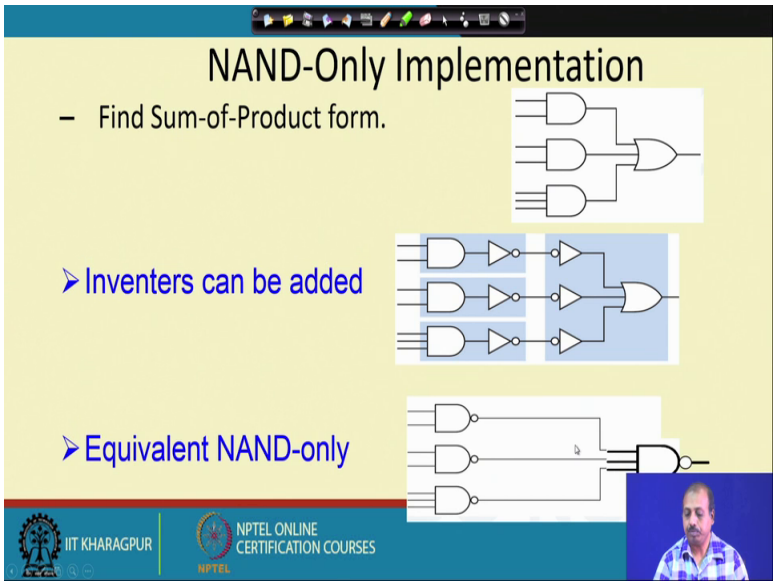
So, this is the advantage of this wired and connection so you see that pictorially you can see that the individual gates that I have, so they are actually open drain NAND gates. So, this is an open drain NAND gate and this is another open drain NAND gate. So, they are connected together and then when they are shorting so we will be connecting via this open, we will be connecting via this external register and pulling it up to high. So, that way it is this is going to F ok. So, this is the wired AND connection and the advantage that you get is that see depending upon the load current that you need, so you can control the value of this register.

So, as a result you can make this can you can make this resistance value low if you want a large current to drive the next state device next state devices, which is not possible with the normal gates because they are the current is fixed and it is fixed by the technology. So, if A and B are 1 output is pulled low so similarly if C and D are 1 then also output is actively pulled low, if one gate is low and the other gate is high then low wins.

So, that is the wired and feature that we are getting here and then if both gates are 1, if the if both gates are 1 then the output will be will be floating and in that case it is pulled to high ok. So, if in fact, this should be if both gates are low this will be so if both gates are low.

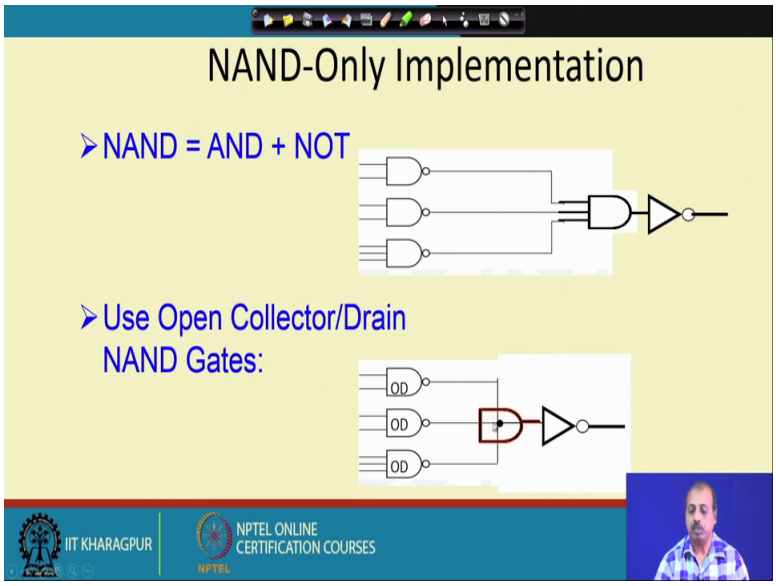
So, this output out output of this so if at this point, so if both A B are 0 in fact so whatever is there so what we mean is that if we get the both 2 both of these 2 gates as open, then we can then this has to be pulled to high via this register ok. So, if the 2 NAND gates they function as they functions as ANDed as their functions are ANDed together so this is wired together to get the output.

(Refer Slide Time: 15:51)



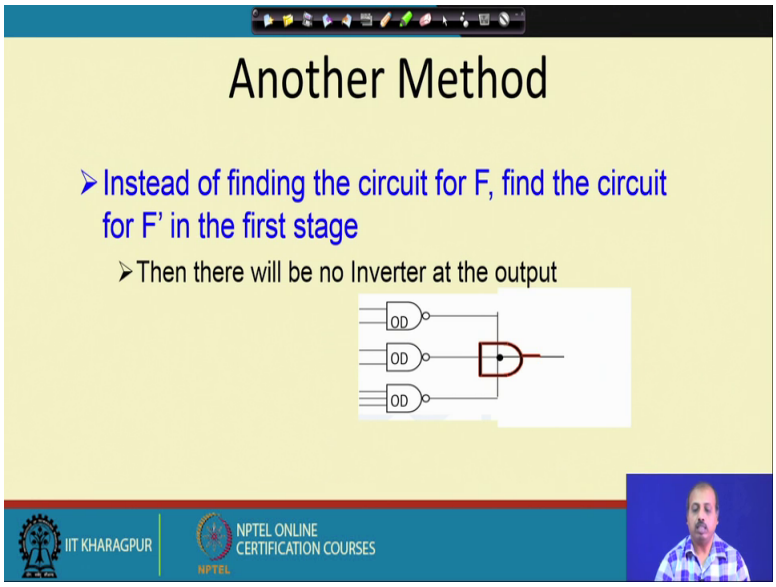

So, this is the wired AND connection so we will take an example and see how this NAND implementation can be done. So, we start with this sum of product formed suppose this is the sum of product realization that I have, then we add invertors between so after for every line. So, we say at the output of the at the in the beginning of the line and at the end of the line we have got inverter. So, this inverters are added and then so this is the, this is equivalent to NAND because this AND invert inverter, so that is equal to NAND. So, this is my NAND operation and similarly this inverter, see if I take it here, so this becomes A bar or B bar or C bar. So, that is A B C whole bar, so here also you get the NAND operation.

(Refer Slide Time: 16:44)




So, that way we can getting the NAND operation and NAND is nothing but AND plus NOT. So, then this and part we can replace by wired AND so this AND we can forget and then we can just join this together by means or to make some wired AND connection. So, then this open collector or open drain gates NAND gates can be used here to get the wired as a final circuit realization.

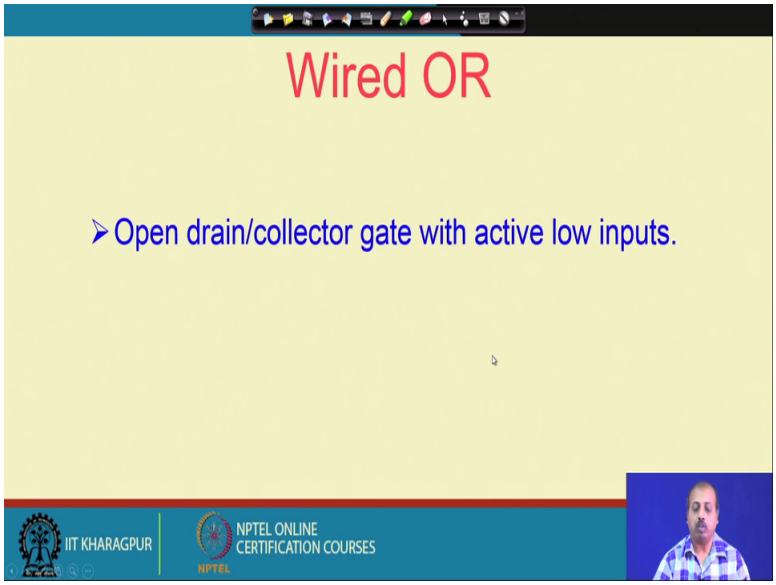
(Refer Slide Time: 17:11)




Another method may be like this instead of finding the circuit of F the circuit for F dash found in the first stage and then there will be no inverter at the output. So, we can do it

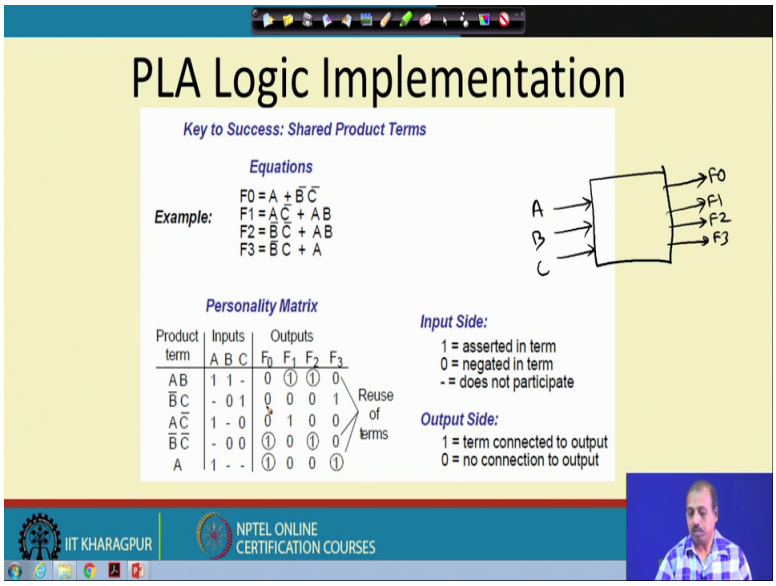
for F dash instead of F and then we proceed like this and in that case I do not need the inverter at the output.

(Refer Slide Time: 17:30)




Similarly, we can have wired OR connection when open drain or collector gate with active low input, so the some logic families so they will tell you so if you connect wires together. So, you will get or type of logic and that or type of logic so there so that will give us the wired OR connection so that depends on the technology that we are using. So, if it support so if it says that if you connect 2 gate output it will be acting as OR gate then it will be an wired OR connection rest of the discussion remains same, so you have to pull it low or pull it high ok. So, this that way we have to put some additional resistance and taking it to high or low.

(Refer Slide Time: 18:15)




So, next we will be looking into another logic device which is known as programmable logic array or PLA. So, PLA is like this that is many a times what happens is that the we need to realize multi output functional block. So, what I mean that if I have if we have got some function, we have got a logic block where this A B and C so these are the inputs. So, this A B and C they are they come as input and this there are 4 outputs F 0 F1 F 2 and F 3 there are 4 outputs F 0 F 1 F 2 and F 3. Now this so this function if we want to realize suppose this is the, so these are the functions, so these functions we have got after drawing the truth table after making the Karnaugh map and so after doing the minimization so, we have got it like this.

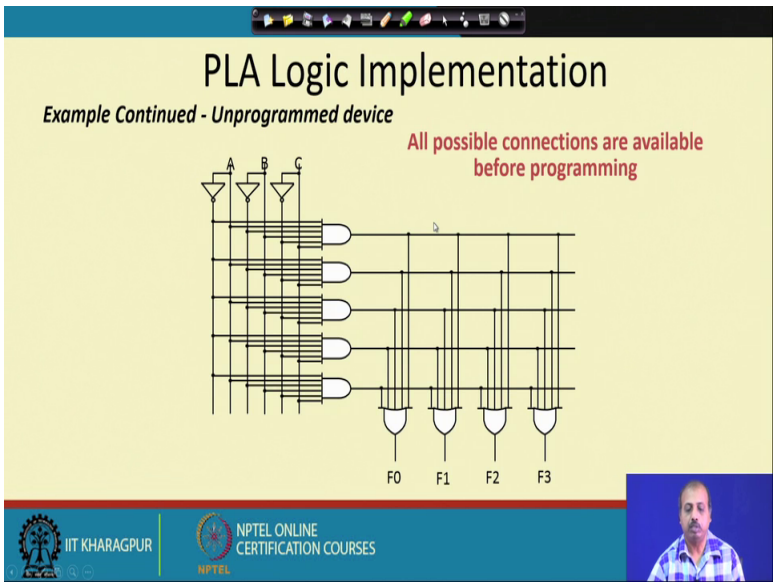
Now if you realize them separately so you will need 1 gate for doing this operation then 1 gate for doing this OR operation like that; however, if you look into the terms you see that there are many terms which are common between them; for example, say this F0 say this term B bar C bar. So, B bar C bar is used here as well as here similarly your term A B is used in function F 1 as well as in function F 2 this term A B is used similarly.

So, many a cases we have got reuse of terms ok. So, on the input side so 1 equal to a asserted term and 0 is negated term so and dash means it does not participate. So, that way it is this is just drawing the, it is just noting down the terms like AB B bar C etcetera. So, if AB the term A B term can be represented as a equal A and B these are 1 and C is not there in the term, so C is put as a dash.

Similarly, B bar C so A is not there A is put as a dash and B is put as 0 and C is put as a 1 ok. So, this way I am draw writing the input side part corresponding to the product terms that we have in this expansion and then in the output side if the corresponding product term is being used, then I will be putting a 1 otherwise I will putting a 0. For example, the term A B is used by F 1 and F 2 it is not used by F 0 and F 3.

So, this F 0 and F 3 I have got a 0 for the first row and I have got ones for the F 1 and F 2. So, that way so this is known as the personality matrix of the PLA, so or the programmable logic array so, we will see what is a programmable logic array, but from this function itself we can we can make this matrix, the this personality matrix. That is you identify the product terms you write down the product terms in terms of inputs and in terms of the variables, and then whether it is present or not and if it is in the output side you write down the functions and a put 0 as a 0 and 1 corresponding to the situation if the product term is not present in the function or present in the function.

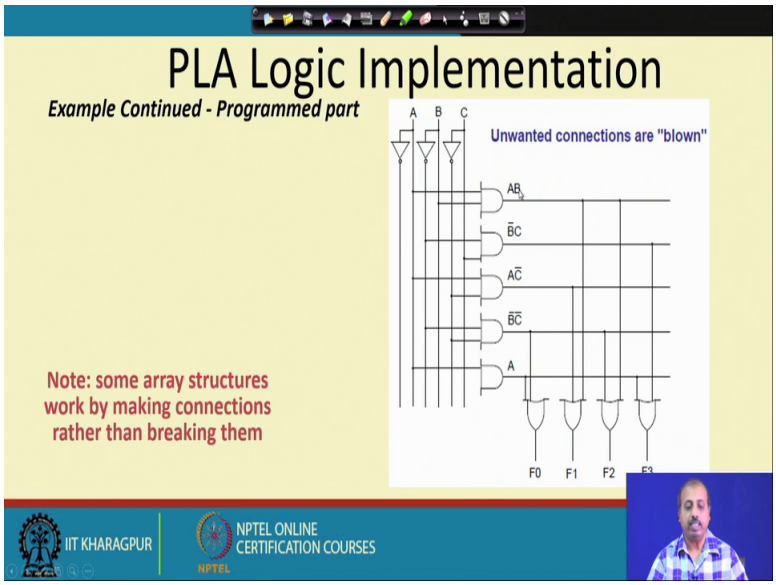
(Refer Slide Time: 21:42)




So, next we will see how this can be realized like see. So, we can see that as if I have got this ABC as the 3 inputs, so we have drawn this inverted lines A bar B bar and C bar and these are all possible connection. So, as if I have got this AND gate has got connection from all the lines. So, second AND gate has been got connection from all the lines, so like that similarly this OR gate I have got connection from all the 5 rows that are running so it connected here.

So, this is an un programmed device, suppose this device is given to you where you have got this type of connections available and I tell you that you choose the connections that you need to retain for getting the functionality. So, you can very easily choose the line connection that should be there to realize the function.

(Refer Slide Time: 22:32)



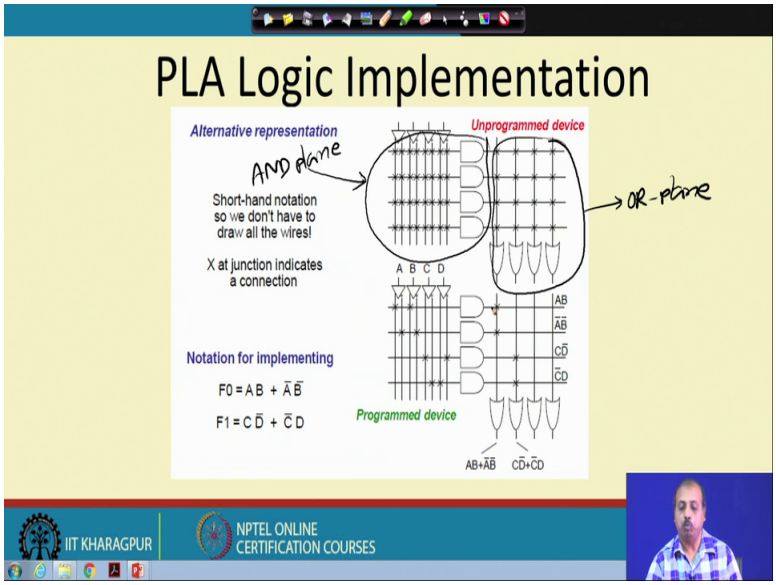

So, all possible connections are available before programming and after that we do some programming and then we take out some of the connections, so unwanted connections are blown. So, there are some switches and those switches can be blown out by applying appropriate voltages to those points and accordingly I can get a customization of the of this structure.

So, we have got for the first term I need AB, so rest of the connections are blown out for this and term similarly B bar C I blow a blow out the next of the rest of the connections. So, AC bar that is also a rest are blown out. So, this way finally this A so this is only this AND gate is not necessary. In fact, so but this is this AND gate is physically present because that was part of the un programmed device. So, this A is coming directly here and that is there is only 1 inputs, so it is 1 input AND gate and then the OR gate we blow out the remaining connection and retain the minimum one.

So, some array structures work by making connections rather than breaking then, so this is also another type of PLA or programmable logic array that we have where we make

the connections rather than breaking the connections. So, both are available both the facile both the families are available.

(Refer Slide Time: 23:53)




So, this is another representation of this AND and OR so this PLA so in this PLA you see that the first part of the circuit so where I have got say this region. So, this region is representing the product of variables and in this part I am realizing the or of the product terms. The first in the first in the left side whatever we have that is the generating the product terms which are actually AND of this variables or it is complements and on the right side we have got the portion where it is doing the OR of the product terms. So, this part is known as AND plane of the PLA this is called and plane of the PLA and this part this right side, so this is known as or plane of the PLA.

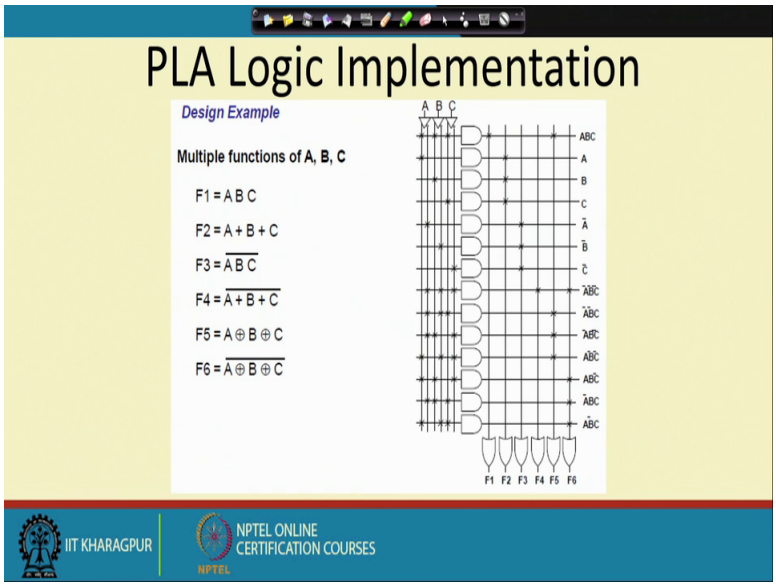
So, the AND plane it realizes the product terms and in the OR plane we have we realize the functions the final output function in terms of the product terms. Now, this connections so this in the un programmed device. So, that is represented by the presence of all this crosses so it indicates that there is a connection and there are fuses ok, so each connection has got a fuse and you can blow it if you do not want the connection.

For example, if you are trying to implement this function F 0 and F 1 F 0 is A B plus A bar B bar and F 1 is CD bar plus C bar d ok. So, then I have to do it like this so there is there no sharing of term here. So, this AB so these 2 crosses are retained so this 2 the rest

of the fuses they are blown out, so you get at the end this line on this line so, you get the term AB.

Similarly, on this line you have retain the connections A bar and B bar. So, they are ANDed here and you get the line A bar B bar similarly you get here this CD bar and here you get C bar D. So, those connections are there and now on the OR plane we make we retain these 2 connections and blow out these 2 as a result I get this AB or A bar B bar and on the second OR gate I realize these 2 connections are retained others are blown out, so I get CD bar plus C bar D. So, this way this F 0 and F 1 are used are realized rest of the lines and rest of the, so product if there are more product lines available in the device. So, they are not used, similarly if there are more number of output lines so they are also not used.

(Refer Slide Time: 26:38)




So, it is a more complex example so we have got a 3 variable function ABC and we have got 6 functions F 1 to F 6 multi output function with 6 outputs. Now here also the AND plane consists of this ABC lines and their inverted lines and we are so you identify first the product terms.

Now, you see this ABC and this A plus B plus C, so they are all separate separate terms ok. So, this A bar B bar C bar, so this A bar B bar C bar are coming here. So, they are OR here to get F 3 and for getting say F for getting say this a bar A plus B plus C whole bar F 4; so how are you realizing F 4 so this is by means of this product line. So, this is A bar B

bar C bar ok, so this is by mean so A bar B bar C bar are obtained and they that is coming to this OR gate to produce F 4.

So, this way we can realize any function so this F 5 F 6 so they are pretty complex ok. So, if you want to realize in terms of discrete gate, so that will require a large number of gates, but you see that once you have generated all the product lines and so you can just OR them together. For example, this a AX or B X or C the function F 5, so F 5 you see so it is consisting of all this terms A bar B bar C, A bar B C bar A B bar C bar ok.

So, this all the 3 odd lines odd terms are OR together to get a F5, so that is the a AX or B X or C function. So, this way we can very easily realize functions using PLA where the functions are, the terms are shared between the PLA between the functions. So, for multi output function realization this PLA gives us a very important strategy for getting them.