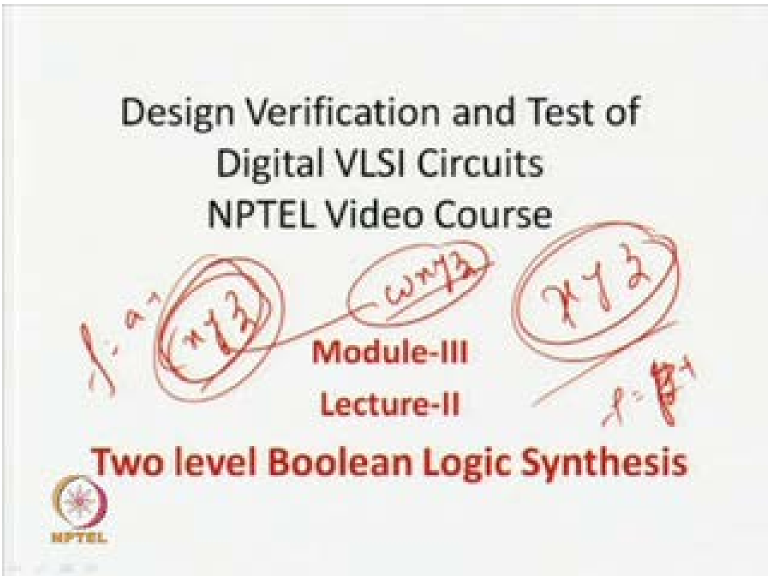
Design Verification and Test of Digital VLSI Designs
Prof. Dr. Santosh Biswas
Prof. Dt. Jatindra Kumar Deka
Indian Institute of Technology, Guwahati

Module - 3
Logic Optimization and Synthesis
Lecture - 2
Two level Boolean Logic Synthesis - 2

So good morning and welcome to lecture 2 of module 3. So as we are discussing in last class that this is the three lecture module (( )) three lecture the triple lecture of staff on two level Boolean synthesis. So we started in the last lecture what is a Boolean synthesis, so in Boolean synthesis we consider the output of the high level synthesis tool. In other words we take some Boolean functions and then the try to Boolean represent them using minimum number of gates, minimum number of logic gates.

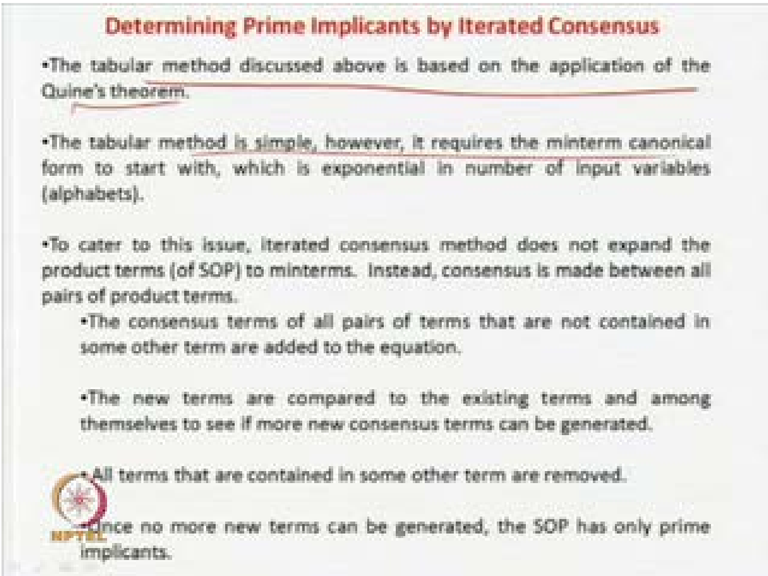
(Refer Slide Time: 01:34)



In other words we have to have a simplified representing of Boolean function. Here simplicity means the number of literals should be minimum and number of terms should be minimum. So that the representation of Boolean functions can be done by the minimum number of gates. Then what we have seen is that it is very good way of doing a quickly we can represents the numbers by prime implicants. So what is the prime implicants already we have seen that definition of the prime implicants and then we have seen that if you take only the number of a prime implicants of the function, so we are actually going to the minimality, because prime implicants is such that which is not covered in any of the other implicant of the function.

So that means prime implicants will have at least some minterms which is means that there is no other term or you can say in which all the all the minterms which is in the inner prime implicants can be a subset of that; that means if there is a term like say you can say x y z so if you think that x y z is prime implicants like kind of the stuff, so there will be no other term in the if for representing the function such that where the whole stuff where x y z totally is embedded in that. Like for example, mean this is not a prime implicants if we have a function like which is says that some z plus sorry some f is equal to say a plus w x y z.

(Refer Slide Time: 03:21)



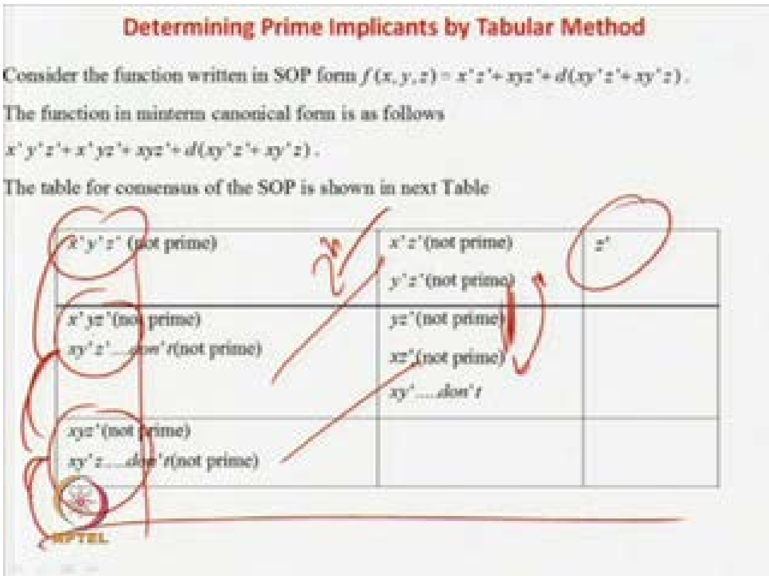
.

So if you think there is a function like this, sorry this we know this x y z so if you think like this, so in that case w z y z. This will not be prime implicants or this cannot be a prime implicants because it is totally embedded in this one. So what does it mean that if you include this automatically this stuff gets covered and if you assume that there is a prime implicants of the function if this case this is. Also it is a prime implicants in this case. So if you that is what the whole idea if you include this term so you will actually have 4 literal, 1 2 3 4 but, if you are implementing if you are introducing a prime implicant in this case x y z is and this is not because x y z is totally inside x y z. So in this case you have three implicants so you require a three inputs and get and in this case you require a four input and gates.

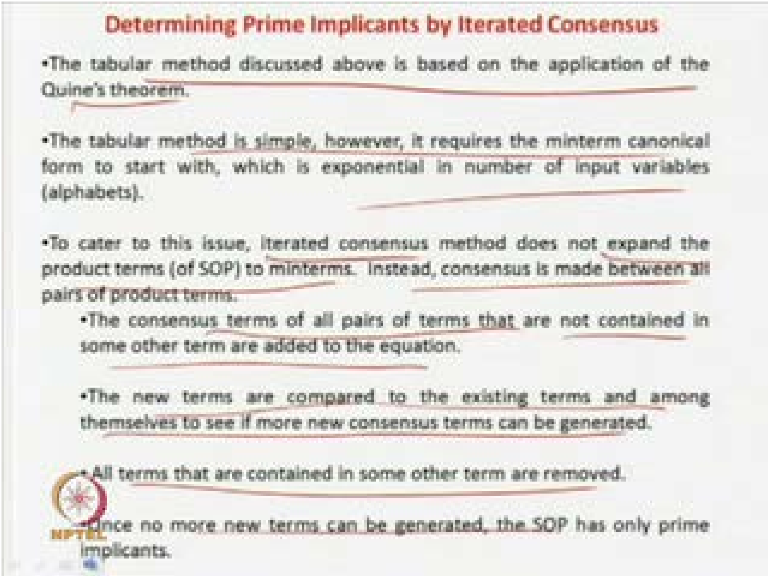
So the basic logic was that if you are we are trying to take such terms where with whose number of lirtrals are small less in number as well as number of terms also small in number less in number. So we are going towards that. So first goal was if you given a function we are trying to find out the prime implicants for this. And that we want to incorporate the prime impliments in the implementation of function, prime implicants are such so that they are not totally inside any other implicants. So that is why obviously the number of terms and number of literals will tend to be more. And then we will see how can we include some of the essential prime implicants or the prime implicants which will cover the whole function. That we will think slowly.

So last class we decide I mean discussed one method which is called the tabular method or the Quine’s method in this case, this was one way of generating the prime implicants of the function. But it is not at all very simple because we have to just divide out the whole number of what you can say we represent the function in the canonical form and then we separate with low what you call the low complimented terms and one complimented terms and then two complimented terms and so forth. And then we tried to find out the concerns between the terms of one row or the other. So which we have already seen. This is a very simple procedure.

(Refer Slide Time: 03:58)



(Refer Slide Time: 05:17)



.

But if you observe this was our basic idea if you remember so we had a table this, so this one is having only one primes and this one is having two primes and this is having three primes there are having the rows having no primes. So and concerns of them between these two and then between these two and terms of this set. And that finally, we generate the next set of terms. And then again we have constants terms between this two and finally, we generate this one. So the procedure we have already seen. The main problem is the good thing about this is that it is a very simple procedure. So you just have to see some concerns between the alternate I mean near about rows in adjacent rows.

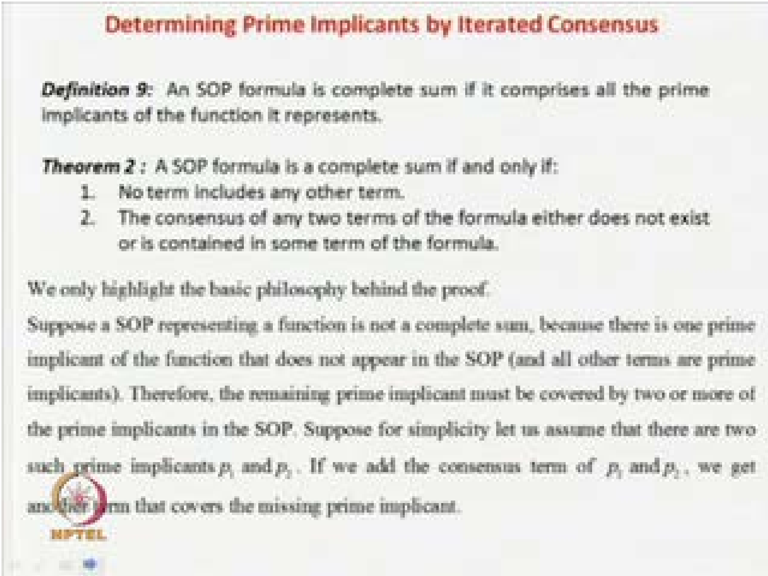
But, the thing is that the procedure is exponentially in nature. Because if we first you have to represent stuff in canonical form, so if is canonical form in the most case the number of terms is two to the power of n so if n is the number of inputs. So you see this is the three variable function so the number of terms can be in the most case can be it. So in this case you can grow up with exponentially in this length. So that was the main difficulty, so today in today's class we will see another scheme of doing the same thing, but, here will not take the canonical form, here we will not blow up the function and then bring it down.

So if you look at the Quine's ,method tabular method what we have done we have taken a function and we have made it in a canonical form. Canonical form means you are opening up the function and then you are trying to merge terms which are having concerns and finally, we will land up in a situation where no more terms can have consequences can be merged. So what we are going to have is the prime sets, set of prime implicants of the function. But, the main thing was it was in exponential so we had a problem. So how to get out from this issue we will see we will see I mean another will see another procedure which is called the iterated consensus method. So in this case what we will do, we will not expand the product terms of SOP form to the mean terms. That is why we will not go for canonical and will not blow it up.

So how will you do so what you will have the of given function as it is we not open it up. Then the concerns of terms of all pair of terms that is not contained in any other terms is added to the equation. So what we will do in this case we will keep the terms as it is and we will not make in the canonical form and then will try to have we are going to find out if they are having any consensus. If there is a consensus will take out that term and will find out that all the terms which are directly included in some other term will be eliminated from the expression. That is what in other words we will take all the pair wise terms and we will try to find out a new term which is having a consensus. Now we will add that term new term with the equation our expression and we will find out that if there is any term which is totally embedded in any other term and if it is there we will eliminate those and again we will redo it.

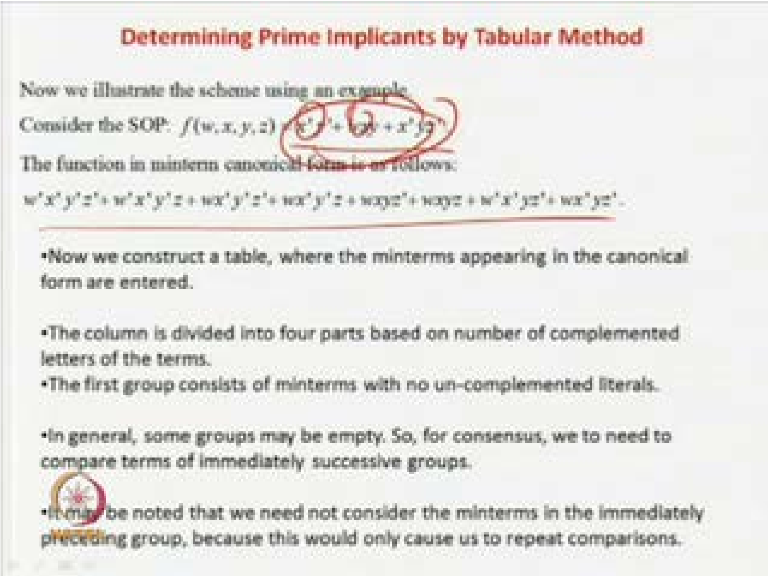
Then the new terms are compared with the existing terms and among themselves to see if more new terms are generated.

(Refer Slide Time: 06:50)



It is first we will take all the pair wise terms and then we find out the new consensus terms, keep it aside and then we check whether any term is totally embedded in between in any of the terms including new one generated and we eliminate all the terms which are already included in some other term and then we again go for some iterative consensus and new term that is added into picture. And then again we repeat the procedures. Again all the terms that are contained in some other term are removed that we do and if no more terms can be generated then we actually stop. Because all the prime implicants has been generated. So that is the basic law.

(Refer Slide Time: 07:21)

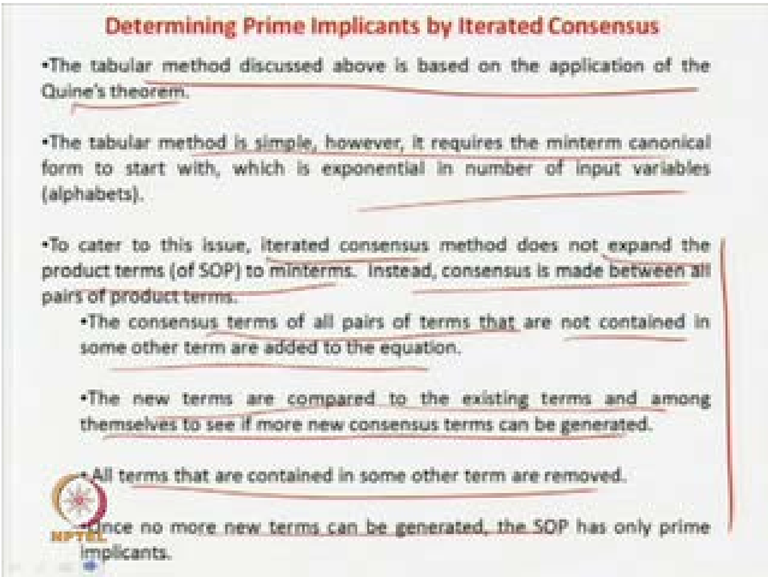


The basic law is very simple, you take pair wise stuff, pair wise terms. Do not open it up take a SOP formula and SOP Boolean expression or whatever. And then a expression given to you and then what you do you go for pair wise term concerns. Very important to note here that we are not going for a canonical expression for. We are just taking terms in this in the whatever order they are so it will be smaller if you remember and then we will see with their example also. In the last class example if you see the our expression was smaller so you can see over this if you just recall we had the small expression, this was the expression and then in this case we have blown it up into this one. This was the canonical but, it size (( )) but, it can go to the exponential bounds.

In iterations consensus what we do, we will have consensus between these two terms. And again between these two terms and again between these two terms. At this step we generate some new terms. Add and then whenever you generate some new terms, you have to check that all the new terms which are in incorporated involved, or the terms incorporated has to be eliminated. And then we have to repeat again and again we have to go for the new consensus, that are added and again if anything is included. And then you keep on doing final case. So no more new terms are generated. Then this procedure will give you what you call this is a set of prime implicants.

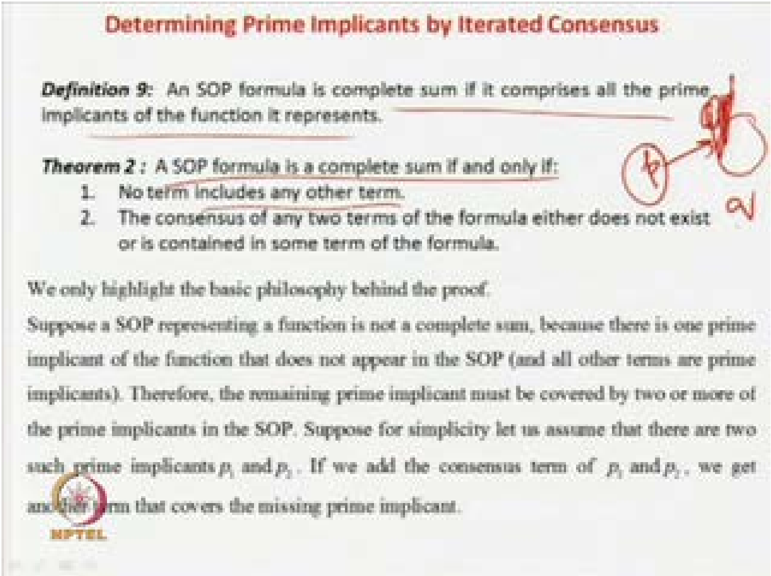
You have to again prove it the procedure is in the tabular method. Then it is obvious that construction is itself, so what you that open it up and you try to do there is any consensus and trying it exhaustedly. So it is obvious that you are going to get the prime implicants. But, now you see the other way here you are not going to open it up. So what we are doing we are finding, so in this case we are not opening the term. We are keeping the term as it is and iterating. So here we require some amount of, though why in the other ways what happens? We do already we have opened it up fully. All the individuals are getting into minterms and we are trying to find out by each of that we are taking pair wise terms, any kind of consensus and then we do not require to do pair wise. So we already separated in term of no compliment, single compliment, two little compliment and so forth.

(Refer Slide Time: 09:30)



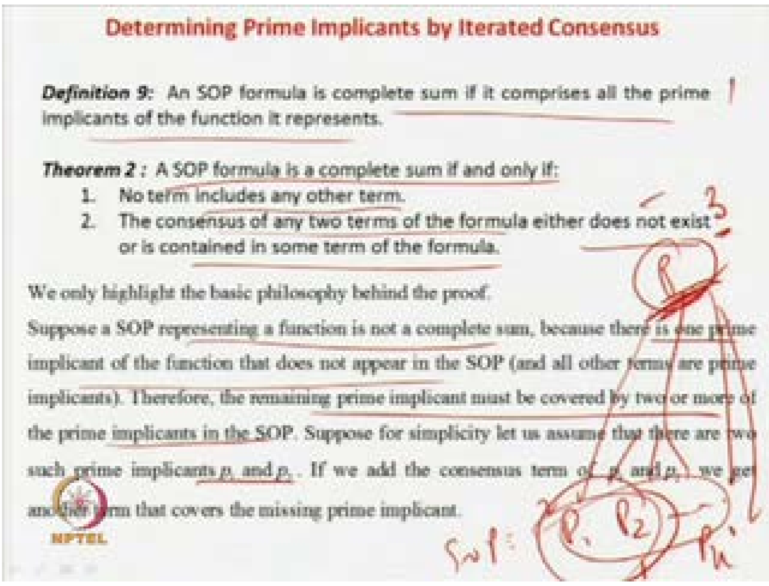
So every term one compliment, two compliments, all terms of compliment no compliment. That I mean single literal compliment. We compare and keep on doing it; that means in other words all the terms, minterms especially group it up into minterms. This terms are compared to themselves. Find out new terms and so forth. So obviously we get the prime implicants, but, here actually the terms are not broken up. So we need some amount to prove this process lower bound this proceed here. It will be done, So how is it? So here is the definition, if the SOP formula is complies, sum if it consensus all the prime implicants of the function it represents. So we say that all the prime implicants complete, it is a complete sum.

(Refer Slide Time: 09:35)



Now we say SOP is a complete sum if and only if there is a SOP formula has all the terms all the implicants if and only if no term includes other term. So by first term is by definition only because if any term totally incorporated or included in any other term, it must eliminate by definition of the prime implicants. That is not a prime implicant. For example if a term called P and all the prime implicants inside from P to another term, called prime say q, so we have not put this compliment. So p is P is a term, if all the minterms P are included in q then obviously P is not the prime implicants, q is the one. So that is the definition.

(Refer Slide Time: 10:27)



The consensus of the two terms of the formula either does not exist or is contained in some term of the formula. So what it is saying if you if you take any say complete sum has been generated. I mean already the whole iteration has been done, and then finally, we get an expression in terms of we say we say all the terms and finally, no new terms are generated. We have stopped, then what we say that the complete generated; that means whenever we stop there, therefore, no two terms can be found. Two consensus can be maintained or if it exist. Also it is containing some other formula, so that is why we have not generated it. So if you take two terms some consensus can be formed. Either it cannot be formed then it is fine, if it can be formed also, that is why we have not written in this expression.

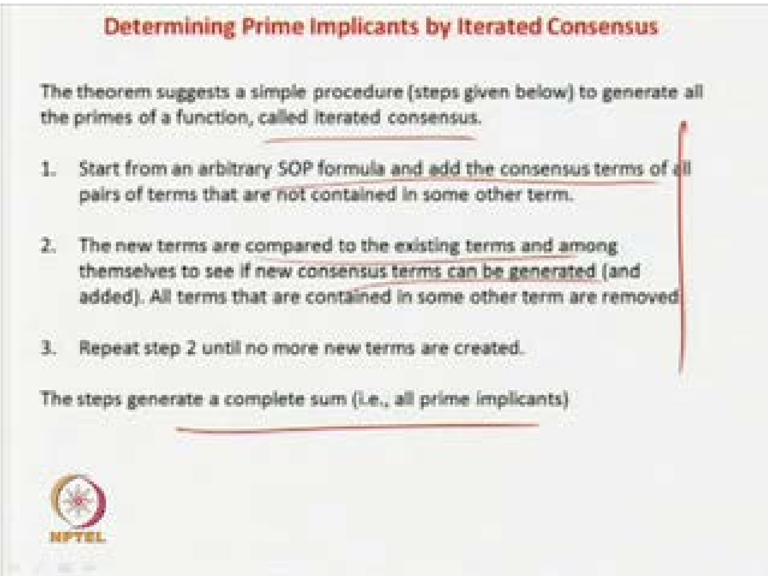
So this holds there is some complex example. Let us see the basic philosophy of the proof suppose SOP representing is not a complete function. Counter logging because one prime implicants of the function does not appear in the SOP. That is the idea because you are trying to find the iterative consensus of all the prime implicants. So let us see that the procedure is wrong, all but, one is prime implicants. By this procedure s one prime implicants is not there. Therefore, the two prime implicants must be covered by other like some SOP is there, you have to generate like P 1, P 2, dot P q. So let us assume that which is one prime implicants, which is missing. Now what happens, some 2, 3, 4 prime implicants have generated the prime implicants p, because SOP is implemented by prime implicants.

So we are taking the assumption. Assumption is that by doing P 1, P 2, P 3 prime implicants, one prime implicants is however missing. Because the procedure discussed in the last slide, this slide, this one, this procedure this is the case one prime implicants is missing. Now all this P 1, dot represent the SOP obliviously. This P must be P is the prime implicants, so it cannot be directly totally included in that P 1, P 2 or P k. So what it can happen, it can inside this one and this one is also possible for first thing. We will assume that two prime implicants can actually generate p. We say that P prime implicants so two of them are in P 1 and remaining one is in P 2. Because it cannot happen that all of the P are included in P 1, P 2 or P 3. It is not possible because then minterm will not have the prime implicants itself.

By the definition of the prime implicants, it cannot be the any one of the other prime implicants that like P 1, P 2 dot dot P k. That is what the definition. Now what happens 3 minterms 2 is in 1 and 3 is in 1. Now all you get this consensus of this two you will get p. P will not be included in any one. So obliviously P has to be in this one. So, that means what happens our basic assumptions was wrong. If you go about all the consensus all the prime terms of it, so this the proof called counter logging. So we call it by prove by contradiction. First you assume that the procedure was wrong. And then you prove that the it cannot be, because the procedure, so the answer will always be correct. So if you have done all the full algorithm, you have executed here then, we left one prime implicants p. So now prime implicants generated are P 1 to P k. So we assume that P has 1 to k or l number of terms.

So which are include in P 1 dot dot P k. For simplicity we assume that the some of the some of them are in P 1 and some of them are in P 2 and somehow we forgot to put p. So what happens?

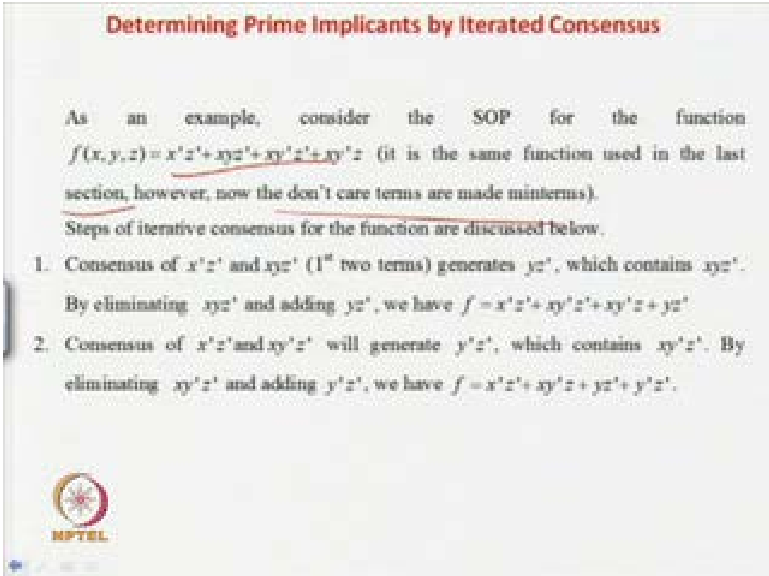
(Refer Slide Time: 14:22)



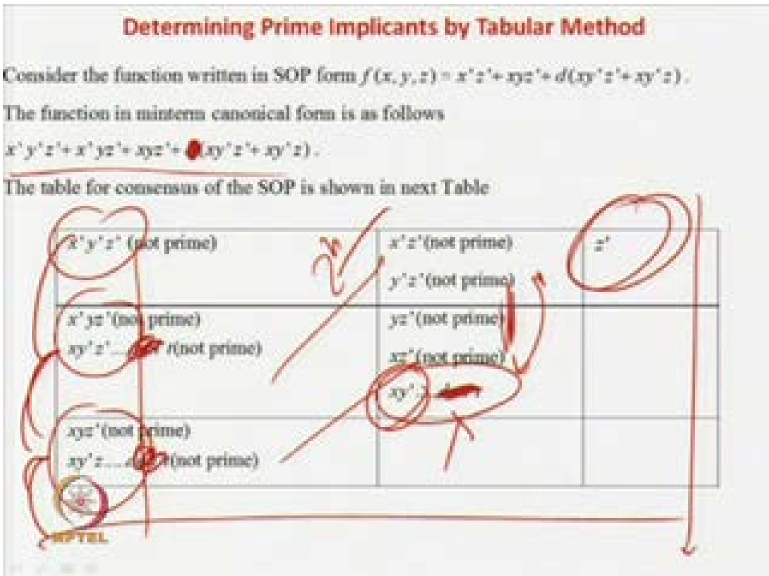
Now we can say that the iteration is not yet complete. Or you take it 1 and 2 obviously you will generate P 3. Then P 3 will have also be included. Because P 3 are not in any one of them, directly not in P 1, not in P 2. Partially they are in P 1 and partially they are in P 2. Not fully in any in any more of them. So just by iterating the algorithm this one more you will get P inside. So that carries if you take the example. So obviously you will get the prime implicants. So this is called the iterative consensus. So what we do we start from arbitrary procedure, arbitrary SOP formula.

And add the consensus terms in the, this is not containing the some other term. Already I told you again it are repeated. So new terms are existed and then you see this consensus can be. If new terms are generated you add that and again you go for iterative consensus. And any point of that time. If you find out that some term is already inside another term, you eliminate it. So these steps already discussed complex or the prime implicants. So we take some example, always this type of complicated are illustrated by example.

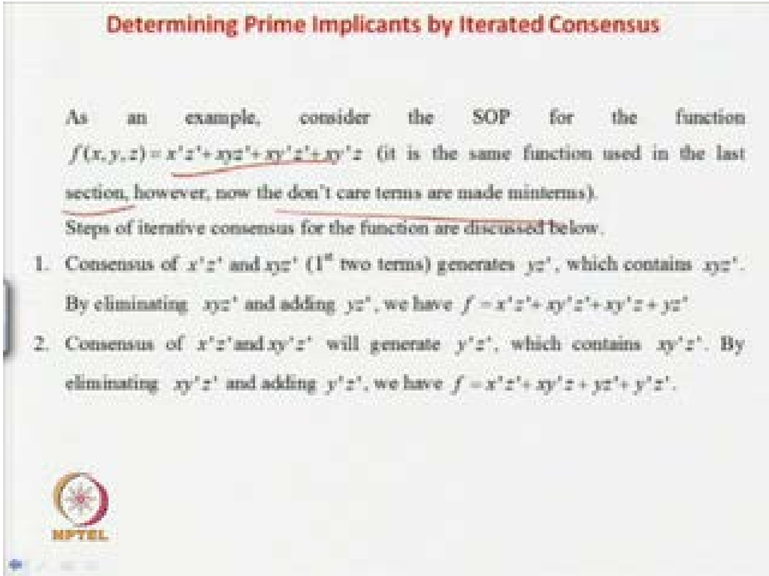
(Refer Slide Time: 15:01)



(Refer Slide Time: 15:21)



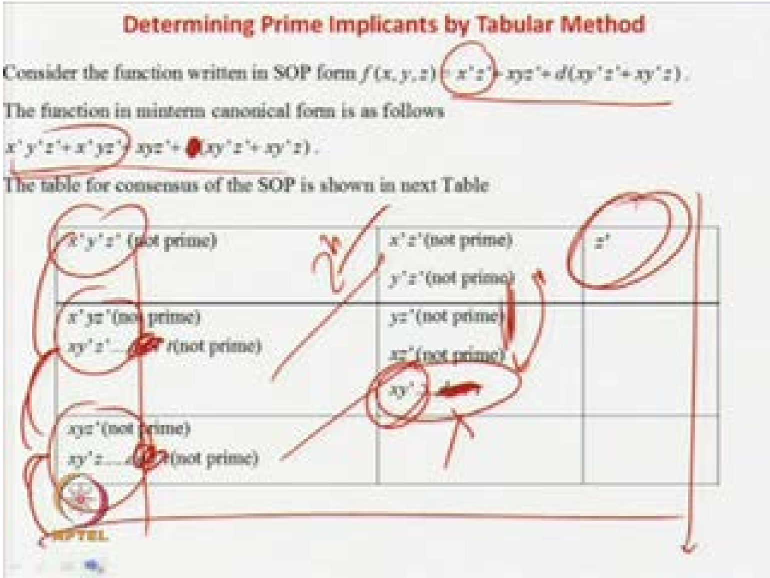
(Refer Slide Time: 15:01)



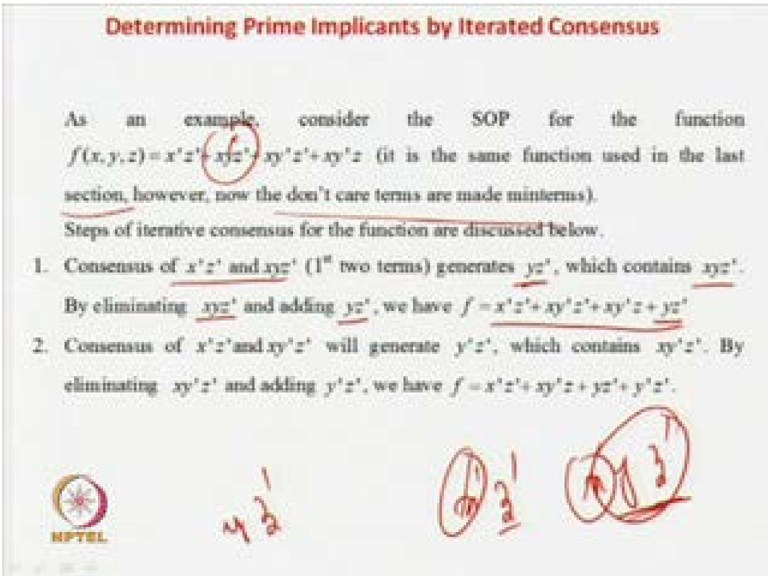
So we take this function. If you remember in the last class this was the function x prime, z prime x prime z prime This is same as the last session or last lecture, however do not care terms are minterms. So if you remember this was our example last class last. This one so this was the expression in this case, this was the do not care expression. Last two terms are do not care terms. Actually we have eliminated, those two rows are minterms. So in this case what final this what one prime implicate? If you remember and this was another prime implicants, if you remember but, we eliminate this one. Because it contains all the do not care conditions, so there is no question of do not care condition.

Our prime implicants are x prime and y prime in tabular method. Let us see same thing comes here, nothing called do not care. So if you do not eliminate this you will get x prime and z prime by the tabular method same expression we have used. But, we have eliminated whatever the do not care terms correct. So do not care are maintained..

(Refer Slide Time: 16:08)



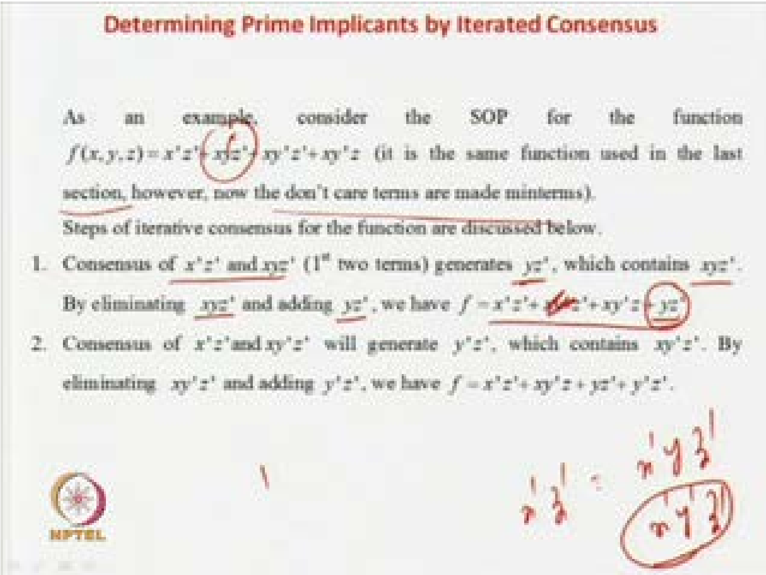
(Refer Slide Time: 16:21)



So now what you do in this case we do not expand it. If you remember this in this table form if you see we have actually expanded it. So this one they have expanded it So here this term they have expanded it this one they are expanded it, so this utility consensus they have not done. Because expanded actually makes an exponential complexity. So what you do now you see in sense of this and this x, x prime, z prime and x y z prime, now you see z prime and z prime are matching. So only this x and x prime is there. So we eliminate this x and x, so what happens you get y z prime. So first two terms y z primes, now this one is generated.

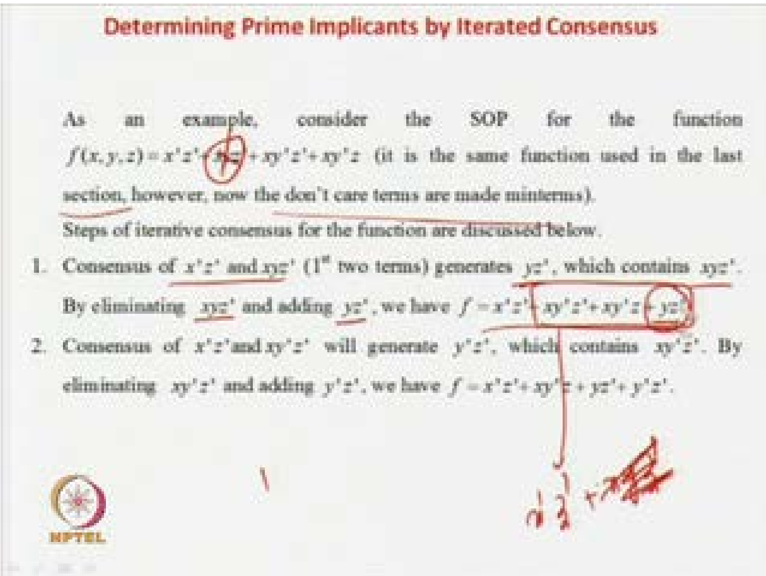
How this you see this is x prime, y prime, x y z prime this z prime z prime is common and x prime and x prime is there. So we are eliminating it and finally, we get y z prime. This is what you get from this one. Now you what you have to do, now you see this y z prime if you take it obviously take x y z prime. Now you can eliminate this one from the expression.

(Refer Slide Time: 17:56)



.

(Refer Slide Time: 18:47)



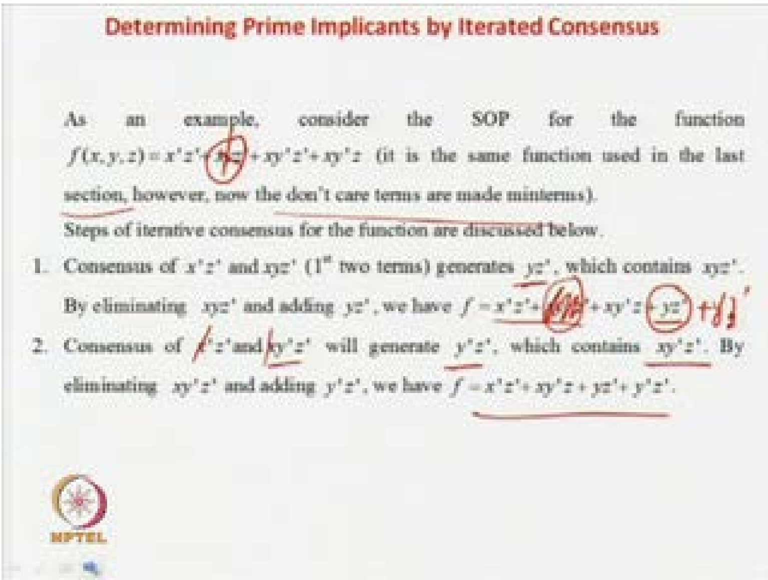
.

So now iterating this one by z prime you generate the term. And then x y z prime but, eliminating this you just eliminate this. And x y prime, so what we have x prime, z prime remains and this the new term you have added y z prime and y z prime. You incorporated and this is what you have x y z prime. This one and this one x y z prime, this one and already embedded in the new term x y prime. So this can be eliminated. So finally, you have this as the equation. So by this step we compare these two guys x z prime and x y prime. And then generated this term which contains this one and added. So first we add say y z prime, so if you can see y z prime already this guy is included y z prime is already included in this.

So you can check that the no other terms totally in this one. Obliviously this one this one is x y z prime. So this guy is also not included here. x prime, z prime can also be represented by x prime, y prime, z prime x prime y prime and z prime. So x prime, y prime, and z prime will not be included in. So this two terms will this three terms will so only one terms will consensus was there x prime, z prime. Now what you can do is that, this all is the new expression. So the new expression z x y z prime sorry sorry sorry x y z prime this is remaining x prime, y prime, x prime z prime is not included in this one. This is the new expression in case this is the x prime in case this was x prime, z prime plus x y z prime.

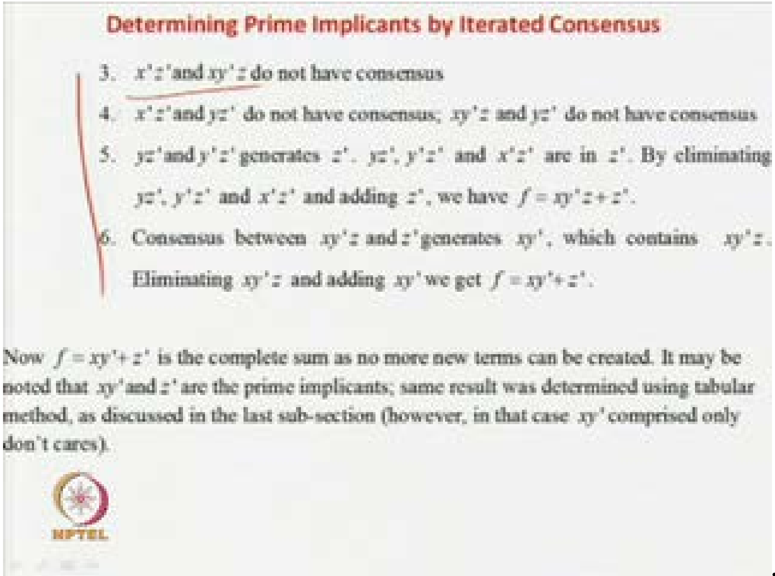
Last two terms so you can see that x prime is included in the last term are added. So you remove this one and finally, you get this one x prime, z prime, x z prime.

(Refer Slide Time: 19:17)

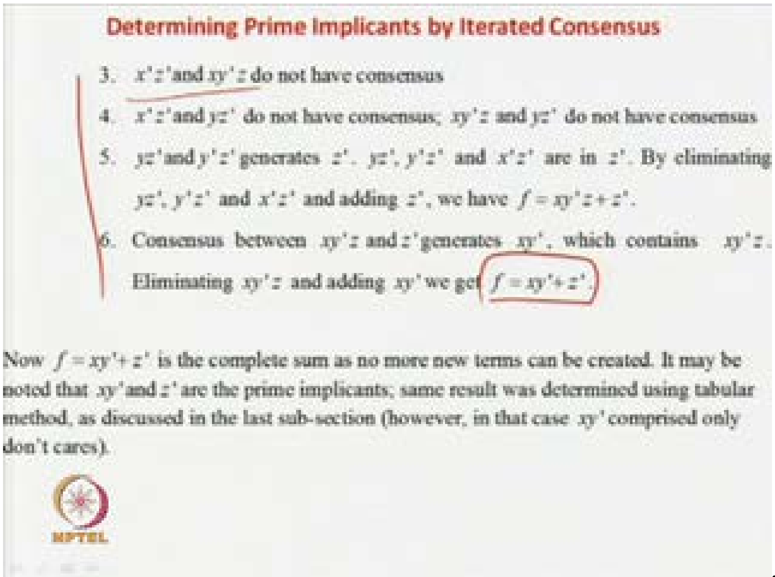


.

(Refer Slide Time: 19:53)



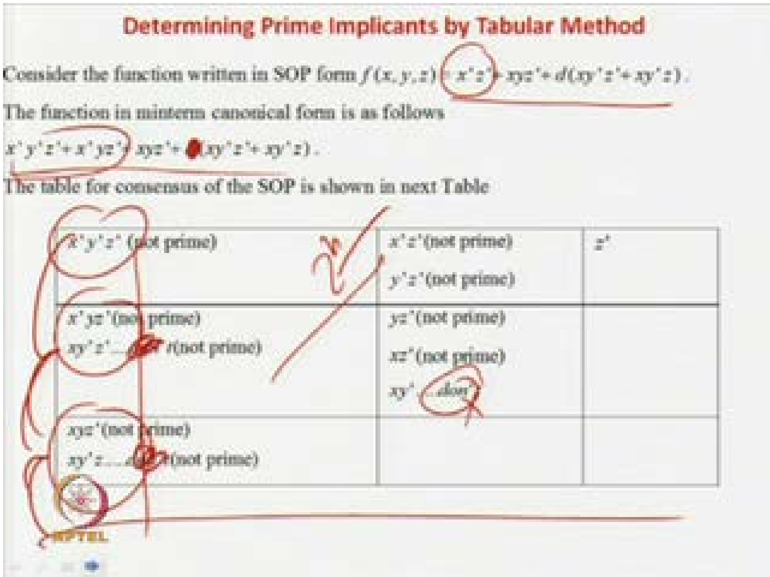
(Refer Slide Time: 20:19)



So now what you do you, again u take the consensus of x prime, z prime and xy prime. Every alternative you have to do, if you take this one you will do this one. So x prime, z prime, x y prime. Now you see again x and x prime is there and z and z prime is there. z prime and z prime are common, so this x and x prime will be eliminated. It will eliminate the term y prime and z prime. Now this y prime and z prime you can add over here. So in this case you can add y prime and z prime, now you can see that x yz prime is already included in the term. You can eliminate this one each contain this one by eliminating this one and adding this you will get the new function. So you can keep on doing this, you can add up little steps over here. So I am not going into details of all this, so you can find sum of the terms like we will not have any consensus, like because this is x prime and x and z prime and z.

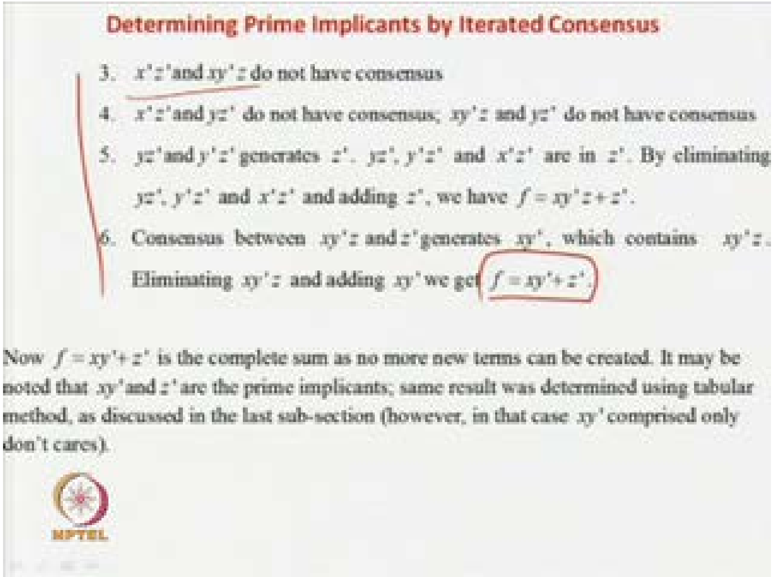
So there is difference in two terms, so in compliments x and x prime and z and z prime, so two places have different in terms of compliment. So you do not have any consensus if you keep on repeating this 3, 4, 5 and 6. So you get the final term this xy prime plus z prime. This is similar nothing to but, our if you see the table form see this is what our x prime, z prime in the previous stuff. You do not include this one this is already comprised of do not cares. .

(Refer Slide Time: 20:30)

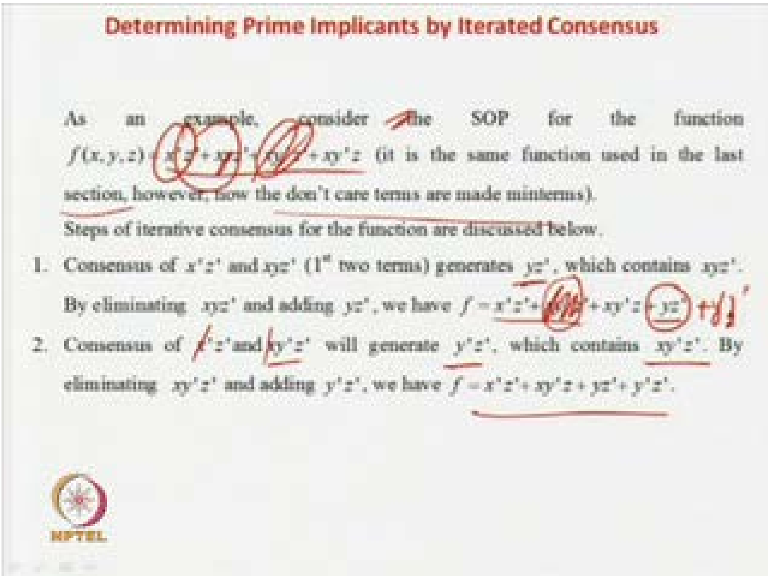


.

(Refer Slide Time: 20:19)

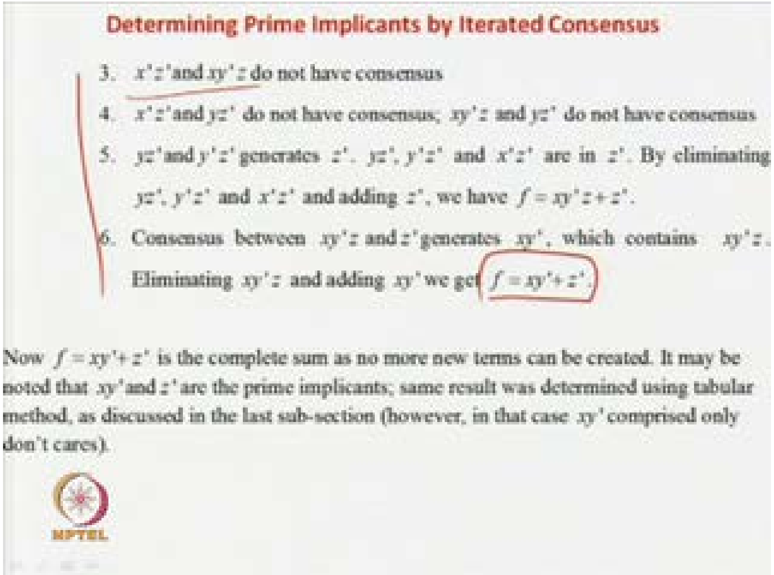


(Refer Slide Time: 21:05)

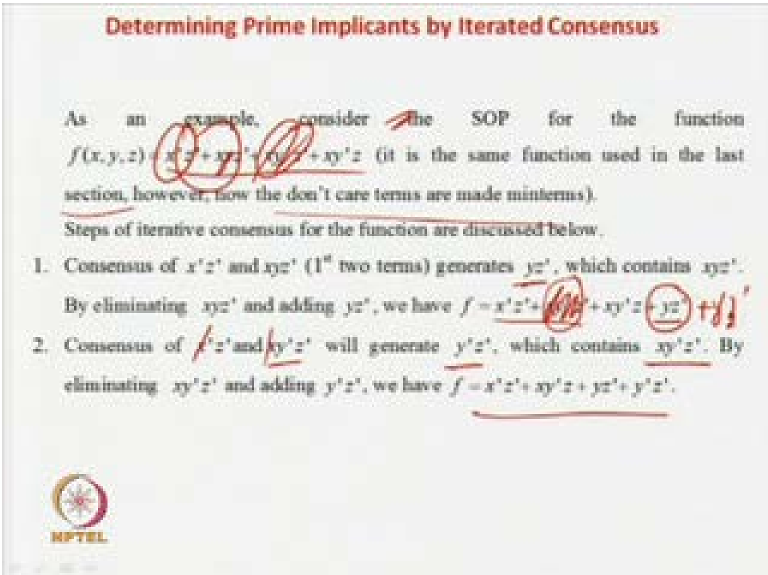


Here you have to include because now in this case actually nothing was a do not care. So actually you have considered a minterm this is the case. So finally, you have generated, now in nutshell what you understand. So what you get over here, here two things first thing is that complexes will remain low. How the complexity is remaining low because in this case we are not opening up. That is we are not opening the this one into a canonical form. We started with this is what we started with and we never opened the data, which is in canonical form. If you have started with this one and then tried this one and tried this two and generated this two terms and eliminate this. And we tried this one, this one and this one and find out new term and eliminated this and we keep on doing this and we finally, get this expression.

(Refer Slide Time: 20:19)

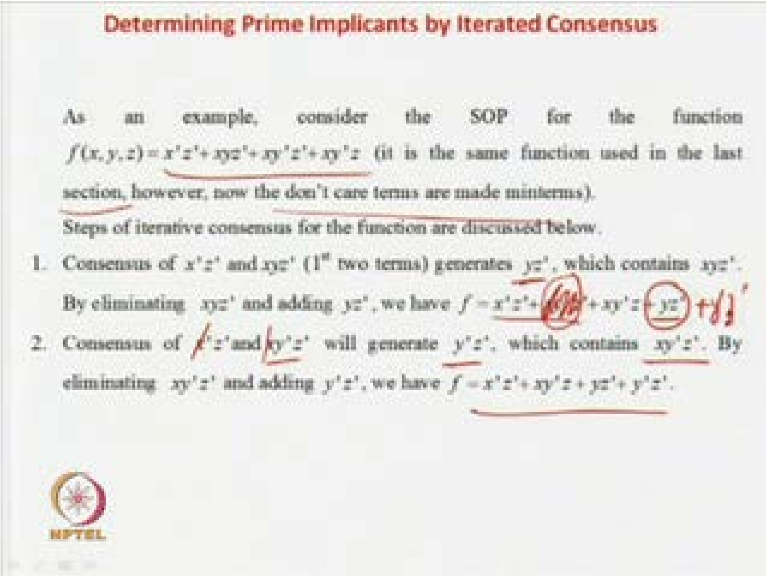


(Refer Slide Time: 21:05)

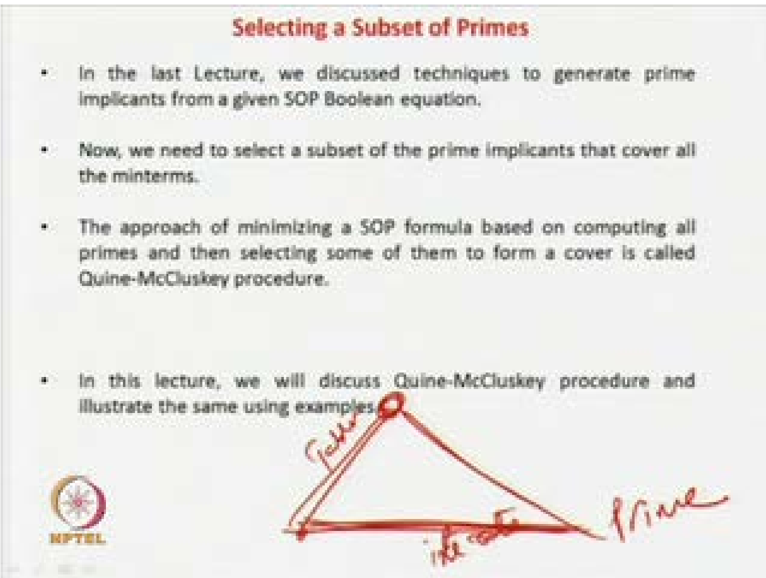


So in this case as we have not blown it up. So the complexity is much much more than the tabular method full exponential block. So in this if you see it is totally depended on how large you have written your SOP formula.

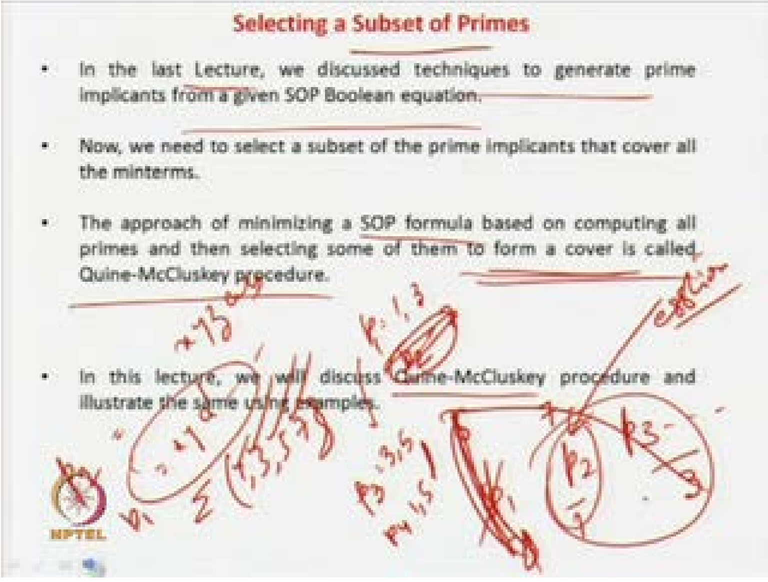
(Refer Slide Time: 21:37)



(Refer Slide Time: 21:47)



(Refer Slide Time: 22:48)



So generally SOP formulas are written in a small way. They are not very large and even if they are large it is somewhat and if you take any SOP formula and go for canonical representation, obviously you have taken a full what you say call this what you say this full exponential. Start of which the starting with the very big size of the problem and ending with the prime implicants. Then the smaller subset of the problem, then you are not blowing it up, you are directly going to find the prime implicants. So in one case you start, you go high and then you find the number of prime implicants. In this procedure we just try this, can say the tabular. So here is the actually prime implicants in this case the iterative consensus iterative start going into this way.

And actually this way we go up in this way blowup all the minterms come down and you are actually, you are doing this and this is how the things are done. So actually it is order by the lower complexity. By now what you have to do, now our job is that. So what you have to do, now we have found out all the prime implicants. Now by the definition what we have seen that by the definition of prime implicants, prime implicants means P 1, P 2, P 3 dot dot dot. So what is the basic philosophy P 1 cannot be included here similarly, P 2 cannot be included in P 1, P 3. So all this terms none of the terms can include in any of the terms. And it can be blown away that is not possible. But, what can happen is that it may happen have some three it may can have four minterms.

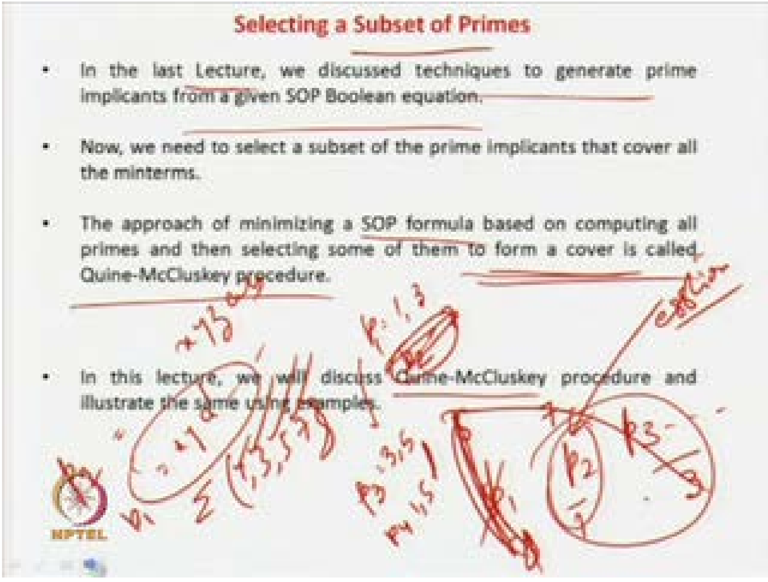
So 3 may included here and 1 may included here. It may or it may not also it may happen that prime implicants, say P 2 has a say 7 minterms out of that 6 may here something like that. And one is nowhere, then it becomes an essential prime implicants this is essential. Because there is no available, so that I obviously you have to take the prime implicants to case 1 benefiting the circuit. But, other things like if you say that P 1 has 4 prime implicants, 1 is in P 2 and 2 is in P 3. So you can drop P 1 and you can take P 2 and P 3. But, essential prime implicants are those which are having minterm, which is nowhere available.

So that the meaning but, some other prime implicants such that some of the prime implicants available in I mean in prime implicants a and available in prime implicants b. So what you can do you can blow the through and you can take the rest of the pq. So in the last lecture or last class we discussed the prime implicants have different, so we have seen the tabular method and today in the beginning of the class we have seen method. Now we have all the method, now our main idea will be the subset of the prime, such that you can cover up the whole all the minterms of the SOP. That is one thing and then what you have to see is that you will have to cover all the cover the function.

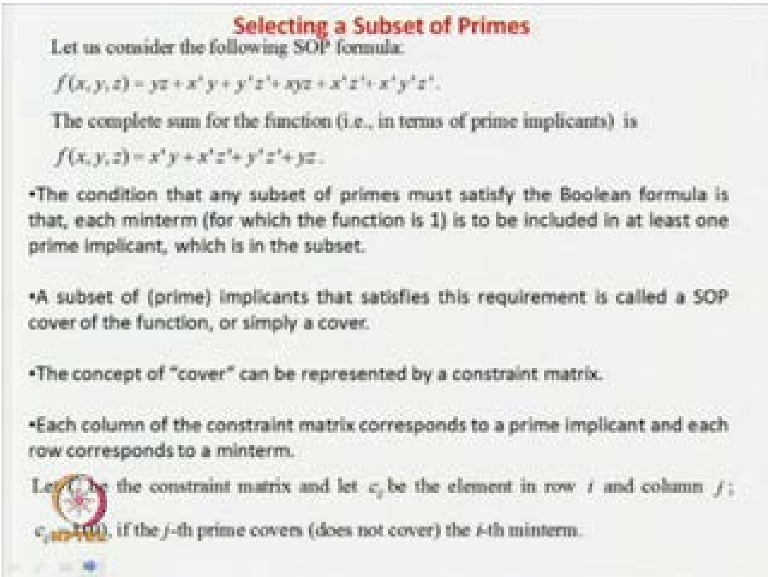
Why, what do you mean by covering the function? I think you know in b tech and under graduate course, we write it up 1, 3, 5, 7 these are the minterms. So that means 1, 3, 5 and 7 are the terms which are actually 1. If one is applied, if 3 is applied, 7 is applied you have to leave the answer as 1 are the minterms. So you have to include those prime implicants, which actually cover 1, 3, 5 and 7. Then what you have to do, now I told that some of the prime implicants will be essential those have those have to be there. Other cases it may happen that P 1 is actually having, say 1 and 3, P 2 may be having 7. No one is having this 1 and say P 3 is having, say 3 and 5 and P 4 is having. Say in this case in 1 and 5, this may be your pictures.

So P 1 is having 3 and P 2 is having 7. P 3 is having 3, 5 and P 4 is having 4, 5 So P 2 is prime implicants. Because it is covering 7, it is covering no other. So obviously you have to include P 2, now you can see if you take P 3, 3, 5, 1 it will cover the 3, 5 and one P 5 is covers. So all I can do one thing also is I can also say P 3 and P 1, 3 and 5. So I can take either P 1, P 3 or P 4. You can take this 2 or you can take this 2 anyone you can take. This 1 for now any 1 you can take and your job is done.

(Refer Slide Time: 22:48)

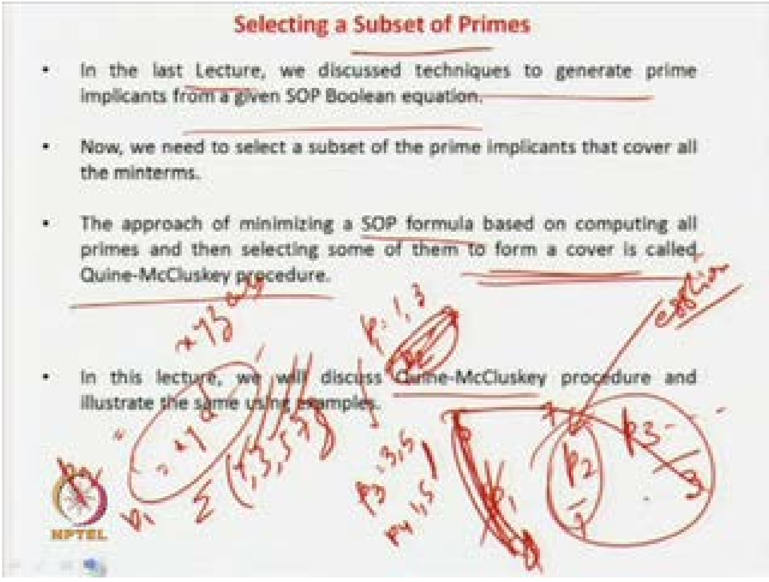


(Refer Slide Time: 26:43)



.

(Refer Slide Time: 22:48)

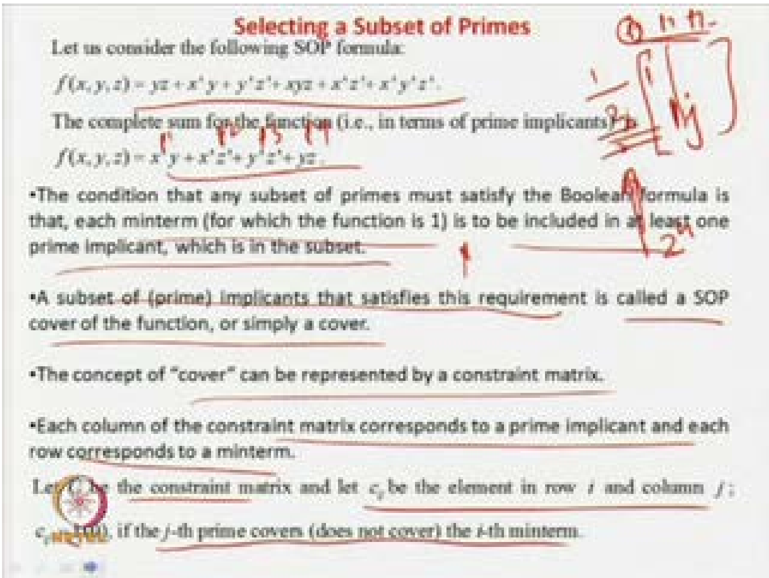


That means what you are taking the prime implicants, such a way that this is covered, this is covered, this is covered and this is covered. So that is why they are called the cover. So how you can do it, you have to select the subset of the prime and all the minterms are covered. So we will find out how you can do in this lecture, in Quine-mccluskey procedure is illustrated by the example this is actually Quine-Mccluskey in this case you select the subset of all the primes. So the all the minterms are covered. Now there is a choice P 1, P 3, 4 and element. So we will take those minterms which are having less number of literals.

So if one prime implicants say x, y, z may be in another prime implicants say P 1, say for example, x, y, a prime something or some something like this. So obviously having less number of minterms will be taken into consideration. So that is the basic idea and that is oblivious also. So you have to take the minimum also it is a subset problem, like you have to find the proper subset. So it cover all the set subset covering problem. And again if you say exponential covering loop you can cover the sub set problem, which is very difficult to solve. No polynomial can solve the problem so again we go for some heuristic that we will see later. Overall the problem is the subset covering problem. So what we have to do, we have to cover some prime implicants, which is a proper subset which is a subset.

Some problem if you are mapping those subset problem. So you have to go to heuristic or the time taken will be very very large. So let us see the subset of prime problem. So basic thing is clear to you, the basic philology is that you have to cover all the minterms from for that you have to take the subset of primes. So essential primes are obviously going in, so essential minterm have some which are available anywhere. Now after that you have to take the subset of the remaining primes further all the minterms are covered. So that you will get the choice as given in the last lecture last slide, so you will get say P 1 is 1 and 3 P 3 is 1 and 5 and P 4 is 4 and 5. So you can take either P 1, P 3 or P 1, P 4. So if you take any one of both 1, 3, 5 will be common, so how to go about it.

(Refer Slide Time: 28:06)



So let us take this one this is the sum and if you go for complete sum you can find that then you can find out the sum of complete prime implicants. Now some condition any subset must satisfy is each miniterm has to be included in the subset. Now you can say P 1, P 2 and P 3 and P 4. And now you have to take a sub set of this such that all the minterms of this has to be included in either P 1, P 2 and P 3 and P 4. That is very opting that is what the representation. So this actually the subset of implicants that satisfies the requirement is called SOP cover of the function or simply the cover. If it takes P 1, P 2 it means it covers all the minterms of this function.

Then you say that P 1, P 2 forms are covered. So if you cover P 1, P 2 and P 4 you actually require that these are the P 1, P 2, P 4 covered. Then you will give the options also P 4, P 2, P 3 can also be the covered. So among all the choices which all we keep, we will keep those choice of prime implicants whose subset sorry a multiple choices which we will get those choice where the prime implicants have minimum number of literals in that k concept of cover. Now we will see that now it was circuit Boolean expression all. Now we will see that the by solving this computational problem you do not about to think circuits and all it can be represented in a constraint matrix. So what is that each column of the constraint matrix corresponds to the prime implicants.

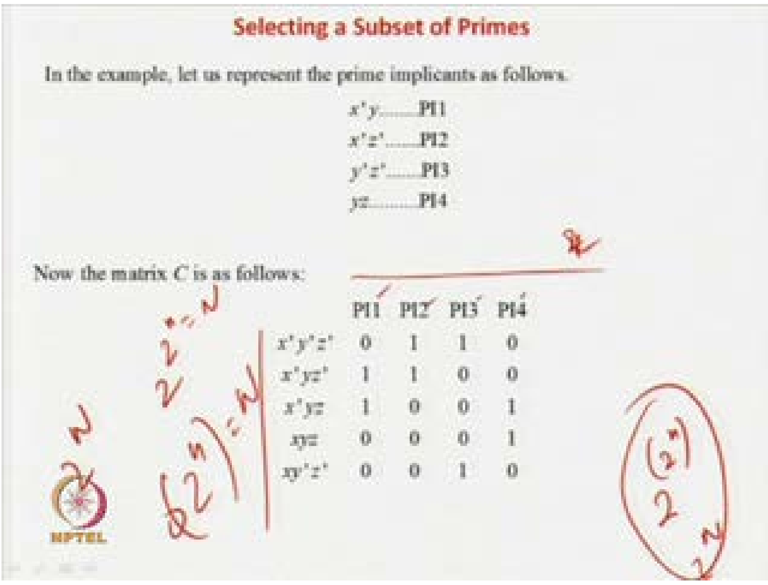
And each row corresponds to the minterm that is we will have a matrix like this. This one will have a prime implicants P 1, P 2, P 3 dot dot dot and there will be a minterm, 1 minterm 2 dot dot dot. So whatever the minterm had been there it has to be here. So here I have to say one point at this point we are going to, so at least not blow it up to the ice. This selective subset primes by the method I have told you exponential problem again we can see some other way of doing it which can be complexity. But, it is very difficult kind of problem to solve it in the next lecture next to next lecture. In this lecture we will see only the exponential algorithm and after that we will go for the heuristic. In this lecture we will see that the problem is difficult to solve known complexity, because it will be compared problem.

So what is our agenda, now our agenda in this three lectures, so at least we go for the tabular method generation. Prime implicants we have to blow the function out but, if you go for iterative functions at least for generating the prime, not blowing up your function. But, to generate the sub set of problems you are going to the cover you require to know which minterms are covered by which prime implicants. So at least you have to again blow it up to find the which function will cover the minterm. What we have to do, any way at least we are not going to do in the first place, so our, now we are solving it in two phases. First you are generating the prime implicants and sub set prime implicants cover it. Now you try to open the function while blowing it up, that is if you go for canonical method, so obviously all the minterms are brought out.

So if all the minterms are brought out, you are blowing the function. So you try to avoid it in first we are differences but, in the second step when you are generating the sub set of primes here you cannot do that. You have to open the minterms and you have to actually cover the minterms. So cover the functions actually, the problem is that you have to blow it up at this stage you have to reach the exponential part, exponential complexity first. The two step problem if the first problem you are avoiding the data, so we are representing it in the constraint matrix. So here all the prime terms are there prime implicants are there. An minterms are 1, 2, 3 and 4. So it can be the power of to the power of n if the expression is 1. All the minterm all the possible are there so, 0, 0, 0, 0, 0, 0 dot dot 1, 1, 1, 1 ,all are the minterms.

If you say this is the very long and exponential and then we put a 1 over here. And if this prime implicants actually covers this is the what the idea is the element of the row I and column J.ci is 1 or 0 if j compliments are Ith term, if jth term 1 over if the jth column cover the Ith column.

(Refer Slide Time: 32:11)

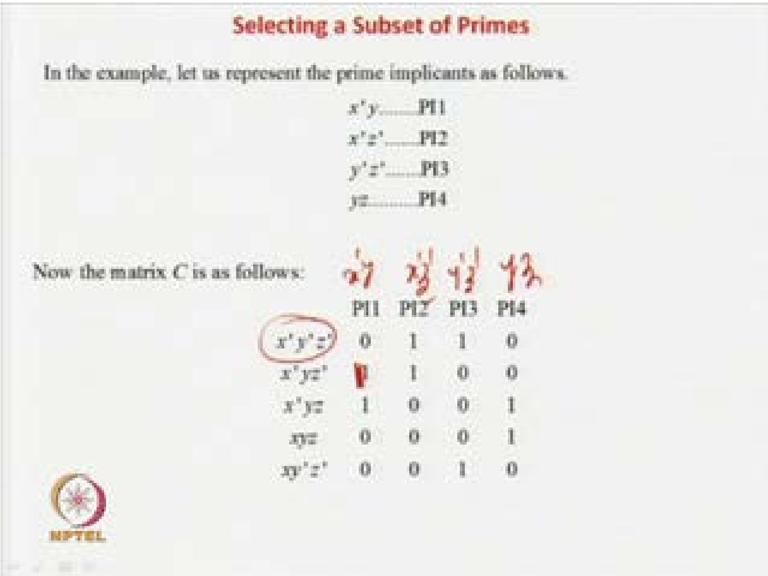


.

Let us see the example. In this case xy, xy prime are the prime implicants. So let us take the assumption in this case and these are the minterms. So this you have to blow it up, nobody can help you there blown up function if there is a three variable function. The order be 2 to the power of 3, it is like the number of rows in this. If there is the n number of function, it will be 2 to the power of exponential. overall the subset of covering of a problem is itself a exponential. So you have 2 to the power of n here. And some order of k or some that can also be exponential in most case. And again the 5 round you have to solve the problem, which of the prime implicants will cover it and exponentially 2 to the power n is again if you exponentially 2 to the power of capital N.

So that problem is actually 2 to the power of n. So in this it becomes the double form of exponential. It is a very very difficult problem to solve. So like all other stuff, like you can say scheduling problem etcetera there were no operation. So in this doubly exponential in the variable by exponential in the number of terms. So this is your problem space this are your prime implicants and this your min terms. So problem space 2 to the power of n variables, so all the problems actually capital N to solve the problem is 2 to the power of N. So otherwise it is 2 to the power of N, so this are very difficult problem in terms of time complexity. So in the next to next lecture we will be process we will see an expression, which is actually a heuristic which can give you the minimize the Boolean function representation in the heuristic manner. We will go about it in the next to next lecture and we will start up with how do you go about the covered problem.

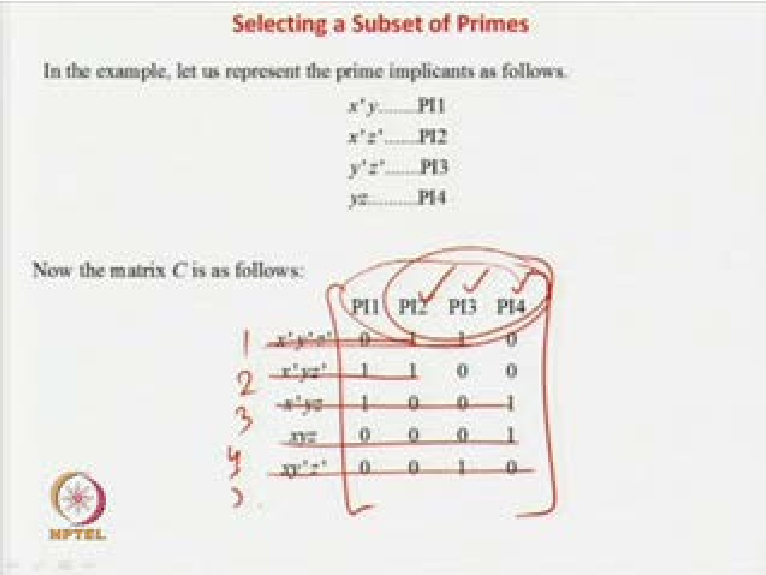
(Refer Slide Time: 34:02)



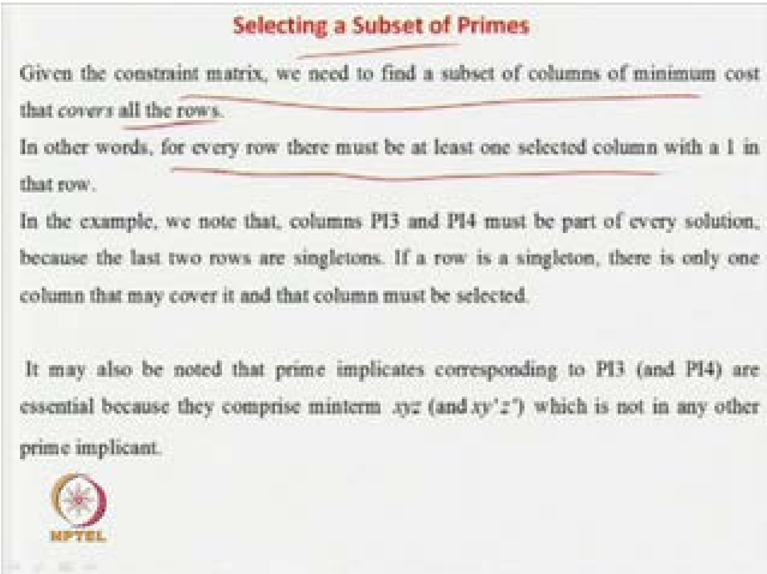
.

So in this case we have seen these are the three prime implicants, say x prime, y and this is x prime, y prime and this is y prime, z prime and this is xy sorry this is yz. Now you see why do we put a 1 over here, because this x prime, y prime, z prime, x prime, z prime. So obviously this guy is this minterm is included in prime implicants 2 similarly, y prime, z prime, y prime, z prime. So this term this term is actually included in the prime implicants 3. So this 1 is not in the y prime So you can easily find out that similarly, we can draw out this matrix where ever the if 1 is over means x prime, z prime. This is the simple idea whole matrix is drawn. So you can forget the whole circuit business, whole minimization business you can forget about this once you make the matrix. So this matrix is there your job is done so you can also write here 1, 2, 3, 4, 5.

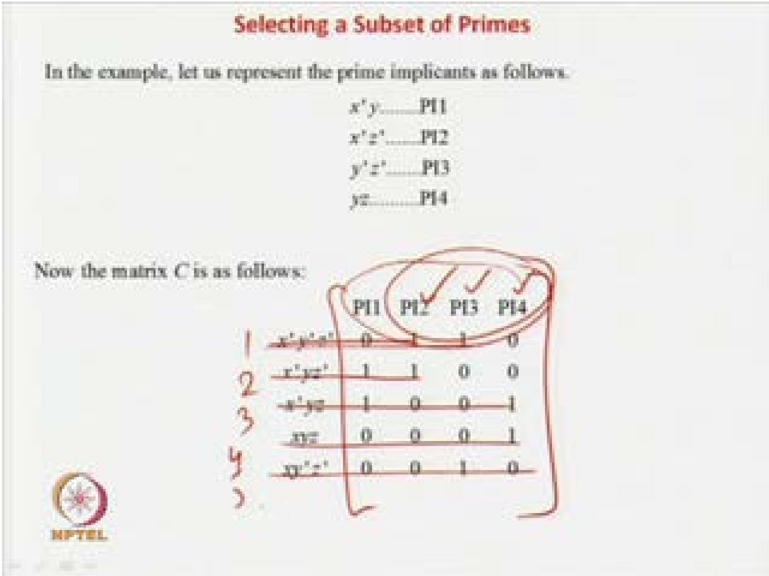
(Refer Slide Time: 34:51)



(Refer Slide Time: 35:45)



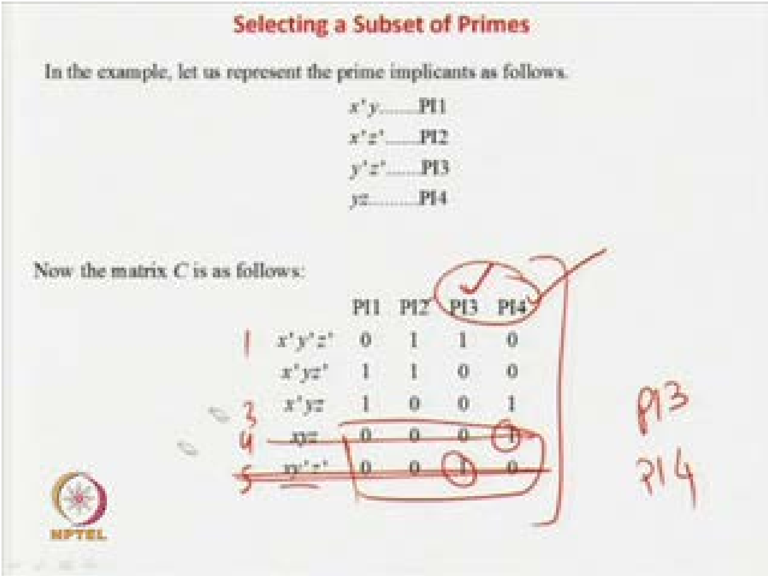
(Refer Slide Time: 34:51)



So now forget about vlsi. Forget about circuit to solve this problem, so what you have to do, which subset of P 1, P 2, P 3, P 4, P I 1, P I 2, P I 3, P I 4 and prime implicants 1 prime implicants 2 and subset of this such that the whole stuff is covered. The whole stuff is covered means, if I take P 1; that means it is covering term 1 and it is covering term 2 and then if I take P I 4, it is covering term 3 and term 4, 1 is reaming. So if I take 1 in this 1, so in this case 5 is also covered and some of the covered twice that is not the problem. So you are taking P I 3, P I 4. So if you take this all the terms are covered. So if you take the minimal subset of the problem all the prime implicants are covered.

So let us see how it can be done, so that is what I told you selecting the subset of prime constraint matrix. We need the subset of the problem that is minimum sub set column, which covers all the rows. So in other words there must be every column, which must be in this row. So that is for some of the columns you have selected. The requirement is that for each of the columns you have to have one column in common. So that is what the idea for example, the now we will go about this example.

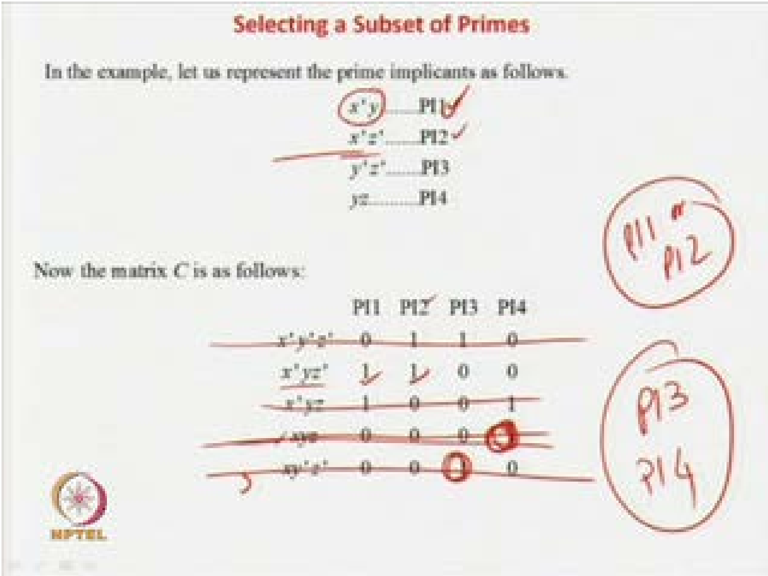
(Refer Slide Time: 36:17)



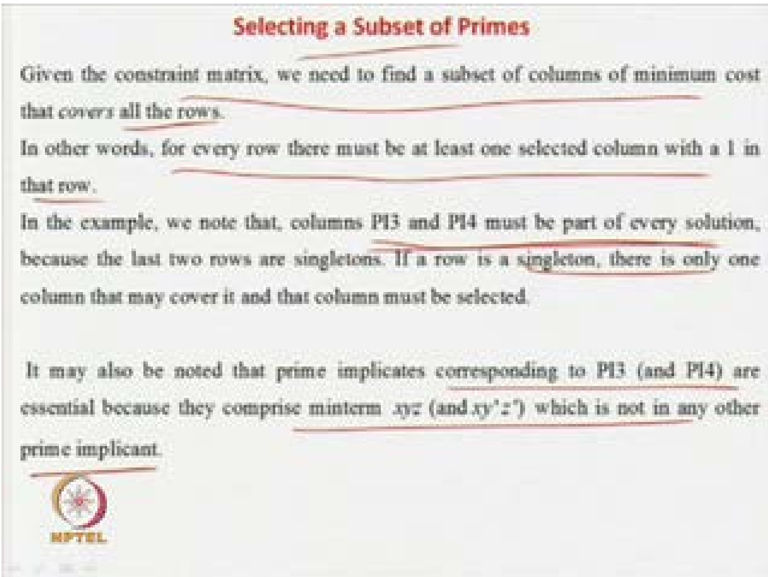
Now how do you go about it? Sorry In this case first you have first see that, so let us first see in this case first what they say note, columns this is the P of columns. So you see this are the two columns, that are very important this is a row this is a row to cover this row 1, 2, 3 to cover this one row. So 1, 2, 3 and this 4 to cover this row you have to be very careful that this only one column have to be this one singleton; that means what P I 4 has to be taken? If you do not take P I 4, then nobody will cover this row 4. Because it is only one in this case similarly, for this term 5 also only one in this row this is called a singleton row.

These two rows are the singleton; that means what to cover this row P I 3 is and also for this and P I 3 and P I 4 become two essential primes, or they must be included. Because P I 3 and P I 4 actually contains one term, in which case it is different, where it contains any other terms. Actually P I contains this term which is low. Similarly, P I 4 contains the term corresponding to term which is no other term. So again in between this two you can take any other. So you can take obviously this term is covered. So if you take this term this one is also covered this one is also covered, so reaming term is 2 and 3. So if you take P 1, P 1 obviously these two terms are covered. So you can take sorry if you are taking 4, this one is also covered. So question remains about 2 is left blank, so to take 2 you can either take P 1.

(Refer Slide Time: 38:02)



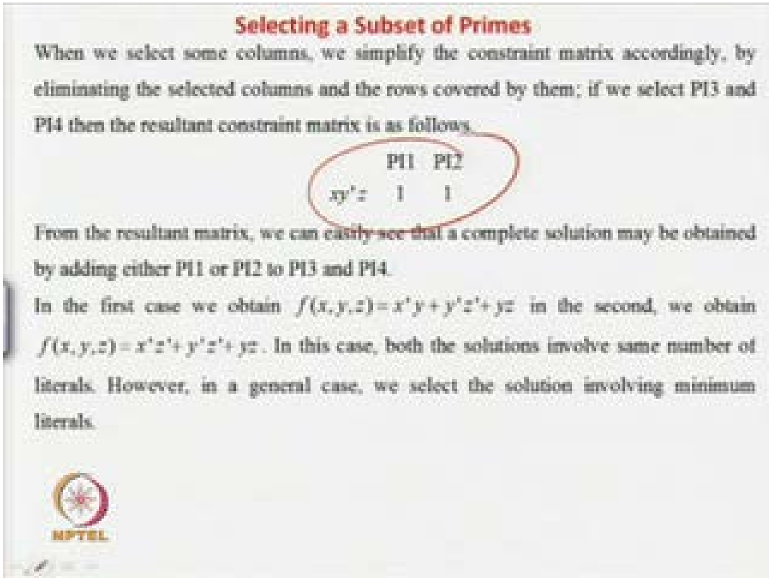
(Refer Slide Time: 39:12)



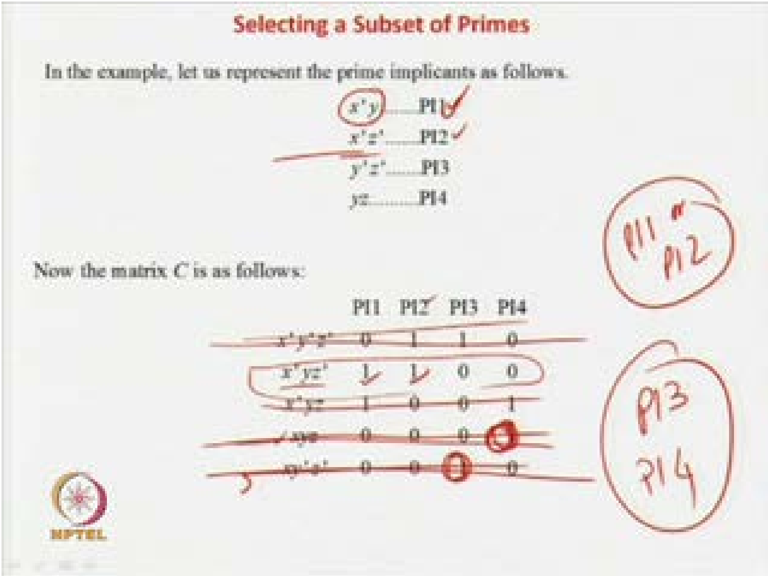
So again we will recap quickly what we have seen. We are discussing this is the singleton to find out the singleton elements to cover this 1 pI 3 has to be taken. If you take P I 2 this one is covered, this one is also covered and this contains this one. But, by virtue of this one you have to take P 3. So this row is also singleton. If you have to cover this element you have to take P I 4, if you take this one is covered. Now nothing else is obviously this one is also 1. So this one is also covered by default of this by virtue of this minterm covered. This element also, we have only one thing remaining. That is 2 which can be covered by this or by this.

So you can take either P 1 or P 2. So this is the essential prime you have taken. So P 1, 1 or P 1, 2 any one of them. you can take. So which one have more number of literals or which one has more number of stuff. So in this case if you think In this case it is P I 1 and P I 2. I will select only this one, it has one compliment it has two compliment. So everything has buffer. So my choice will be this one. It may have sometime have more number of literals have more number of literals you can avoid them. So this is the simple way of doing it, so this P I is part of the every solution. Some rows are singleton it must be noted that because of prime implicate this one and this one, this one is not in any of these. So this is actually a essential prime implicants.

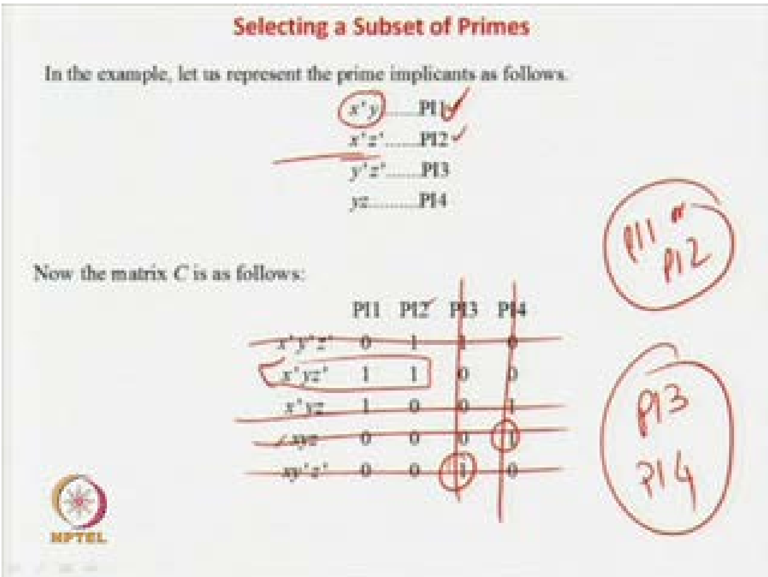
(Refer Slide Time: 39:23)



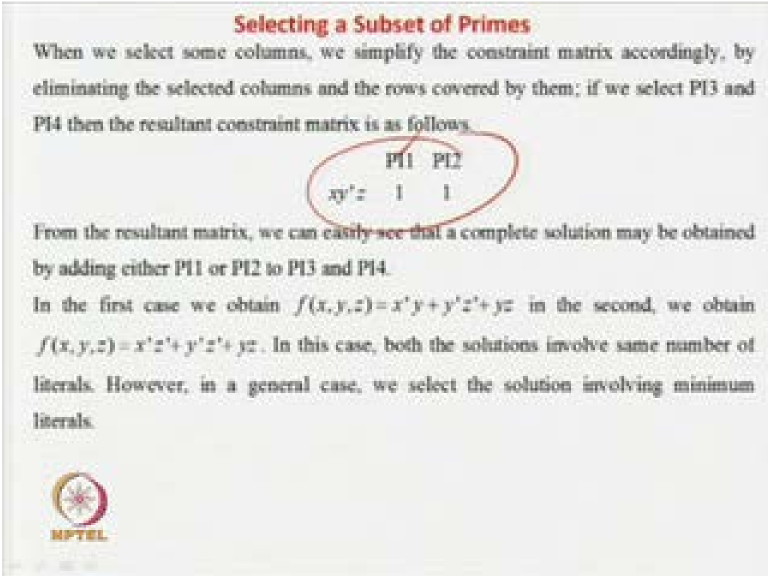
(Refer Slide Time: 39:32)



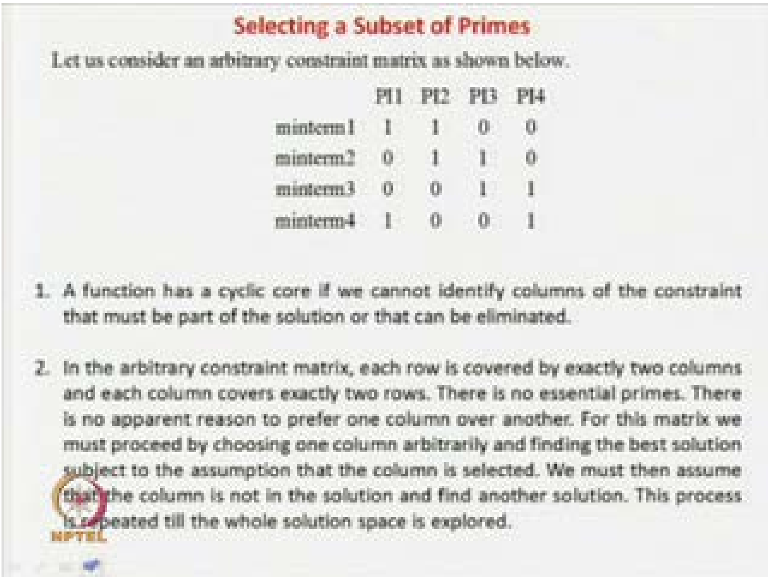
(Refer Slide Time: 39:43)



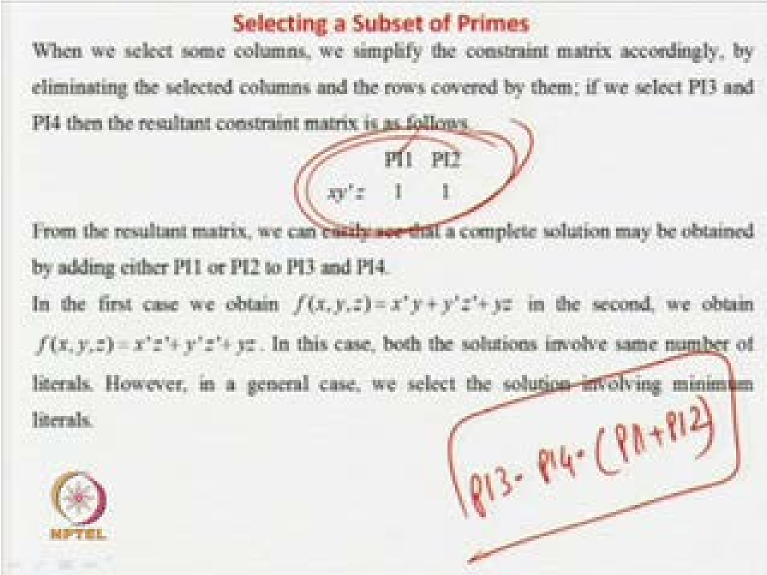
(Refer Slide Time: 40:05)



(Refer Slide Time: 40:38)



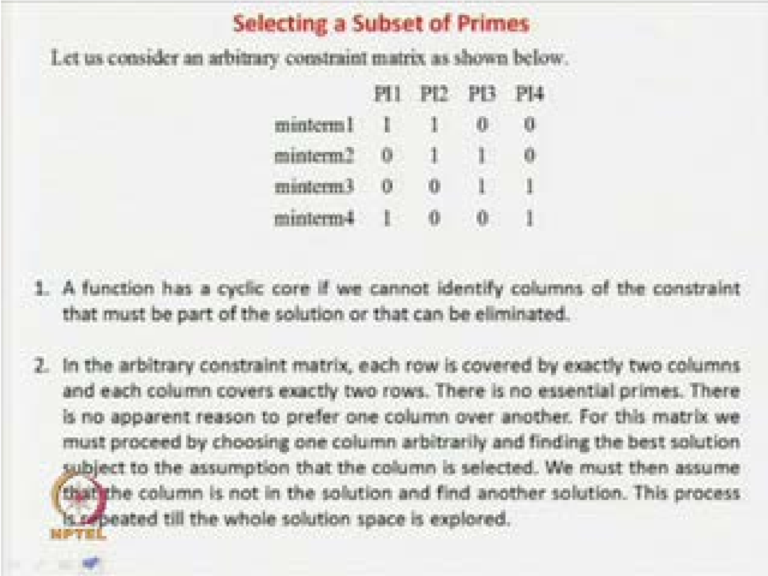
(Refer Slide Time: 40:42)



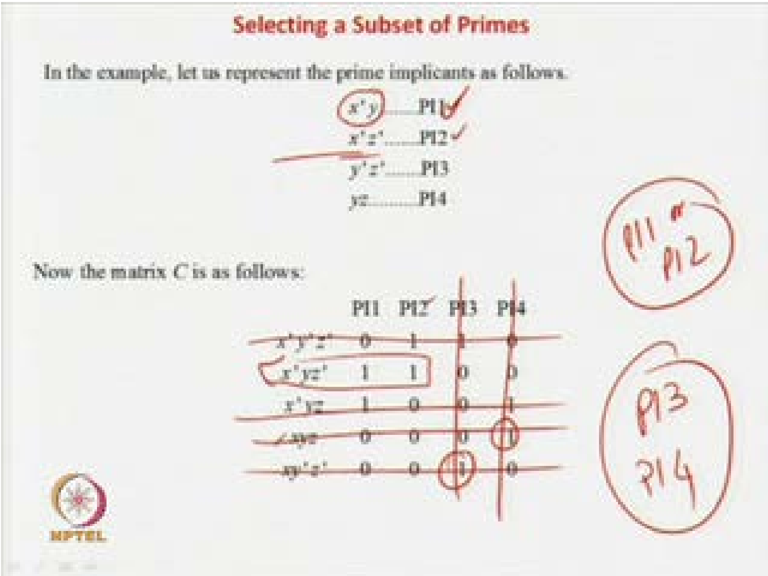
So what we have done, if you have reduced matrix, how it become the reduced matrix. By virtue of this one this is gone and by virtue of this one finally, we saw that finally, this is the only thing that remain. So I mean very quickly very quickly do that by virtue of singleton this one taken this one is taken and this one is gone by virtue of this one. 4 is taken this one is covered and this 1 is covered. Now you see only this element remains. So this 1 they have taken. It over here. So any one you can compare either P 1, 1 or P 1 2. So in this case it depends on number of literals. In our case we took this 1, so was less so this is the way we do it.

The proper algorithm we will be taking in the next class, which will be taking you to a very I mean whichever I discussed first you find out the singleton elements. You keep on decreasing it and finally, you land upon try to do this one. And this one and this one. So in next class a very proper algorithm, so let us try to see how we can preliminary steps or the preprocessing steps for an algorithm. So in this case this one is the idea we can write in terms of a formula. Also we can write P I 3 dot P I 4 dot P I 1 plus P I 2. So that means what you have to take P 1 and P 2 among P 3 and P 4. This also you can represent in terms of equation now you can see.

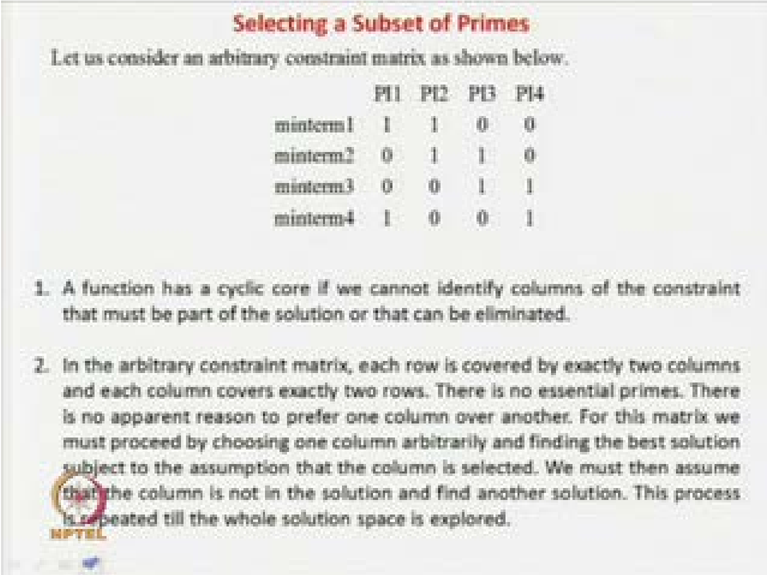
(Refer Slide Time: 40:38)



(Refer Slide Time: 39:43)



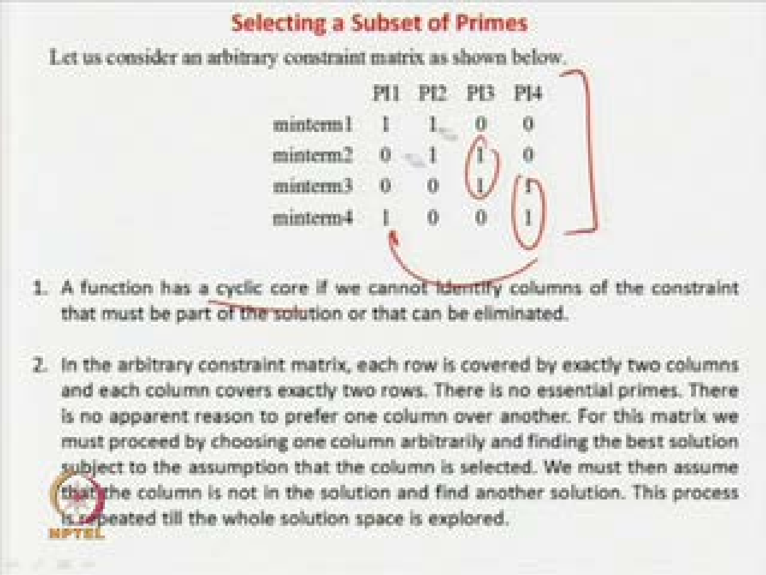
(Refer Slide Time: 40:38)



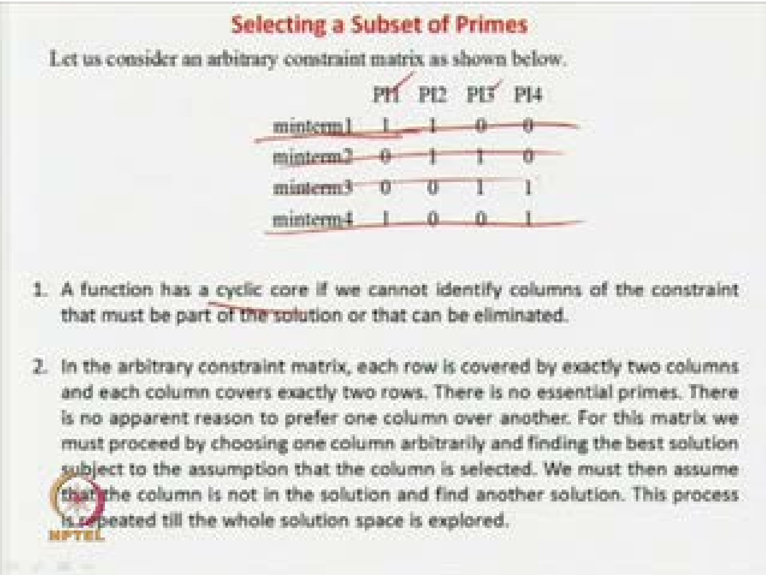
Now we will be doing in some preliminary steps or preprocessing steps which will be doing exact algorithm, before which I have done. So you have started this one and you have removed singleton elements. Then we have got 2 this matrix and we have got a choice and we can take any one of them. And then so forth. Actually nothing but, there is a way of solving some subset of prime covering problem. Now we will be going about the formal algorithm. I should not say exhausted algorithm formula steps algorithm. Before we have to go for some preprocessing steps of matrix, so that we can make smaller in size if it is possible. So we try to make possible as small as possible. So like example for singleton elements or singleton rows, singleton rows means some of the prime implicants has to be selected by then, if you select them your matrix will become smaller than some way we will make our matrix smaller.

And remaining elements of the matrix we will apply the algorithm to find out the, I mean choices if in this cases final matrix is this small, if you have to take this choice similarly, if you are thinking to remove the singleton stuff was nothing but, singleton element of preprocessing and all that stuff. Now we will see all the preprocessing cases. So the matrix size will become smaller and smaller and finally, when you have to run some choices among some choices. Because you have to take some path, some paths because then we can have a smaller matrix. In this case now once you have this matrix the vlsi problem and computational problem are such that we will take an arbitrary matrix and start. So you see this is an arbitrary matrix.

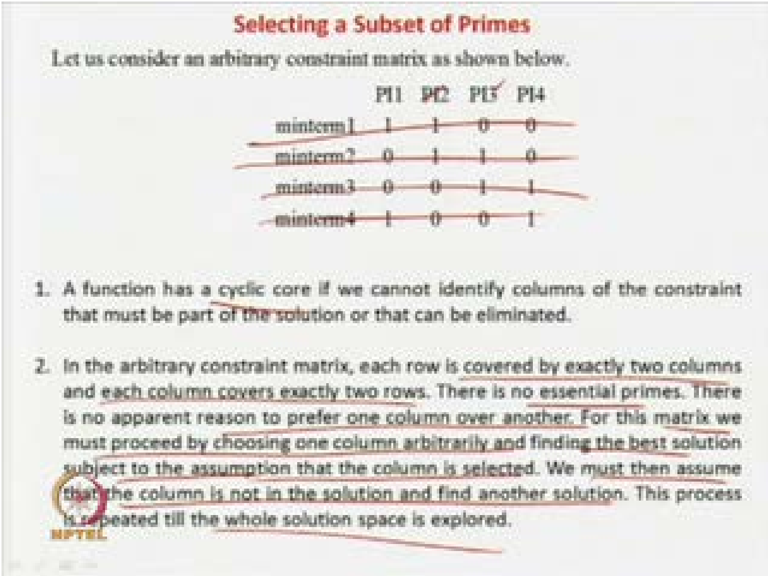
(Refer Slide Time: 42:22)



(Refer Slide Time: 43:10)



(Refer Slide Time: 43:23)



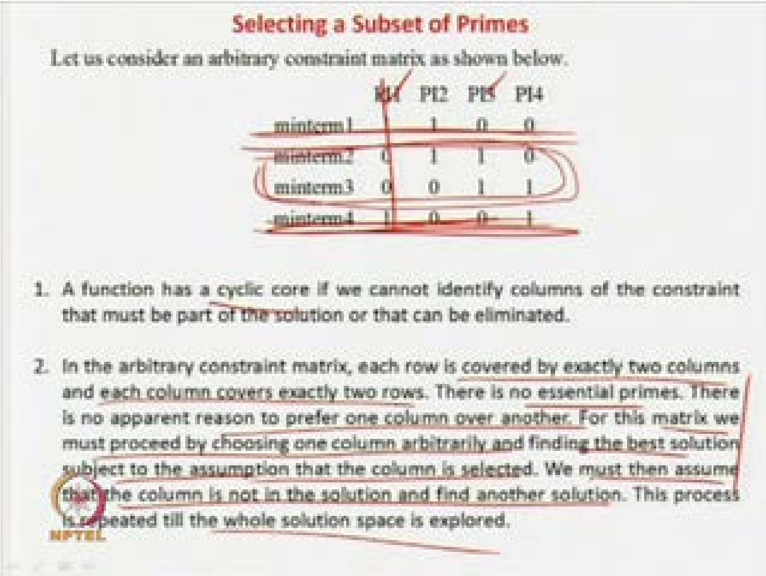
.

So this is the special kind of way, which is called cycle. So in this case you see there is no singleton row; that means that no rows which is having single. Only 1 to this row has to this row has; that means no prime implicants is a essential prime implicants. Because there is no row which is a singleton and there is no row which is having a available nowhere else. So in this case what we have to do, so there is a lot of choices over here. So you cannot directly reduce the matrix so 1, 1, 1, 1 and 1, 1 .So this is this is the cycle and this 2 and this 2 and this 2 is an anther cycle matrix.

Because in this case there is lot of cyclic matrix. So if I say ,will take P 1, if you take P 1 so this term is covered. And this term is covered if you take this one remaining are these 2. Now you can take P 3. This one will be covered and this one will be covered. Similarly, P 3 and P 4 also you can consider. But, if you take P 2, this 1 will be gone and this 1 will be gone. And if you take P 4, this 1 will be gone and this 1 will be gone. So any 2 you can take. So if you see the matrix is not in reduce lot of choices, so arbitrary matrix is each rows are covered by two columns and each row is exactly covered by two rows.

There is no essential primes, there is not one column to represent this. For this matrix you have to proceed one column arbitrarily and find the best solution that the column is selected. And we must then assume that the column is not in the solution and find another solution. This process is covered till the whole solution space is explored, this one is a cyclic matrix where we have to take this way. So in this I mean in this paragraph we will talk about the general procedure, which will be highlighting in next class. So this is the special matrix where the reduction is possible.

(Refer Slide Time: 44:10)

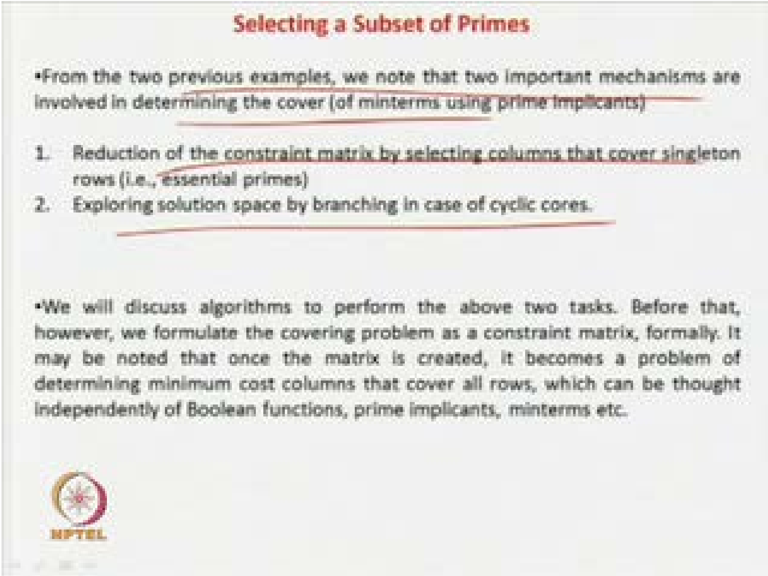


.

So take P 1, if you take P 1 this is covered and this is covered. Now this now this remaining part is this one part is, so if you take P 1, this is covered and this is. So remaining is this 1 to cover 2 and 3. The choices are you take P I 3 or you take P I or P 1, P 2 and P 4 for the new solution space. So the arbitrary take one row sorry one column and again repeat it. So here is the small ones, so obviously you will take P 3. It will cover both the terms otherwise, so next we will be taking a real examples. Large examples will be more clear more what I am saying is that in cyclic matrix no reduction is possible.

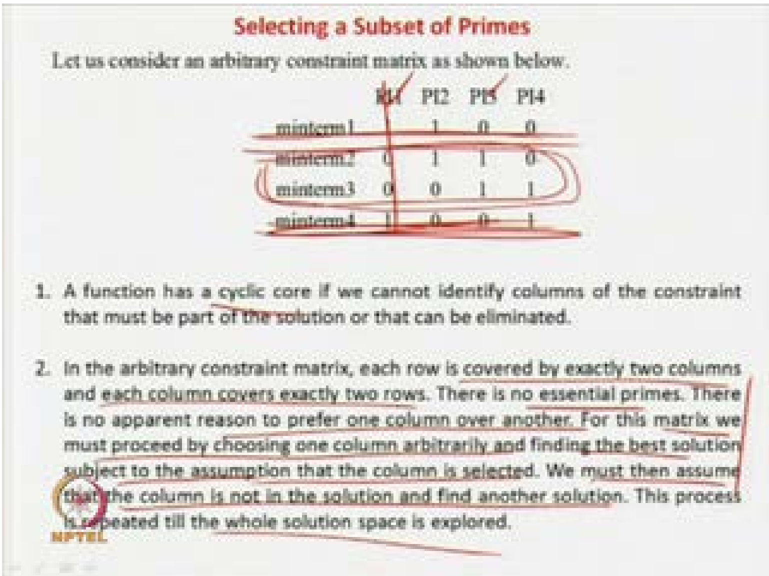
So you arbitrarily take one column, so in this case column and again redo what I am doing is that this is the sub set of the problem. So this is the best way of doing it, so keep on doing it.

(Refer Slide Time: 45:08)

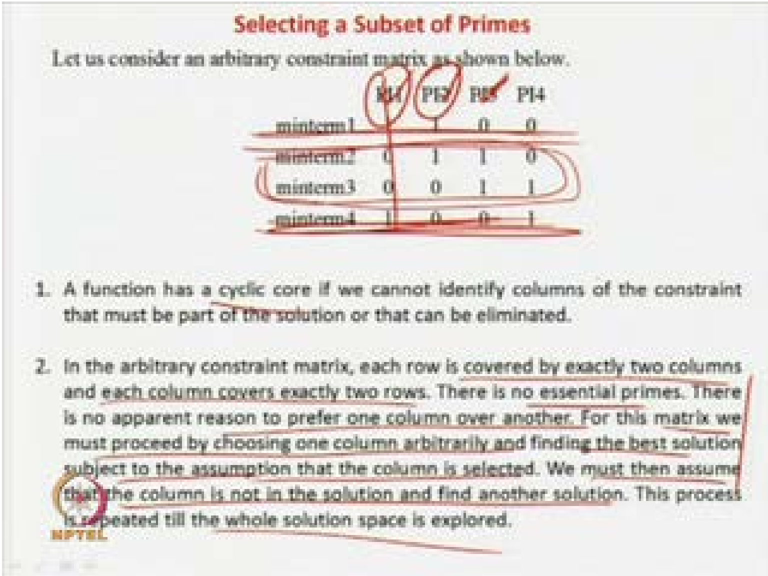


.

(Refer Slide Time: 44:10)



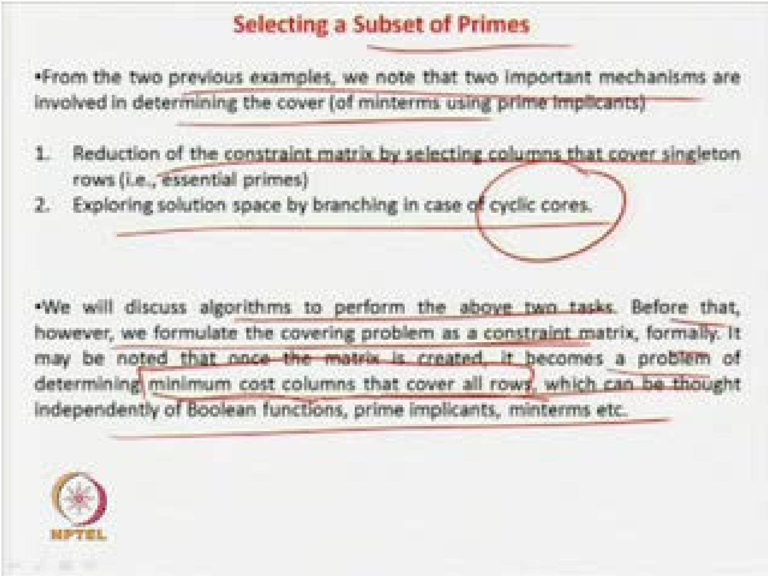
(Refer Slide Time: 45:43)



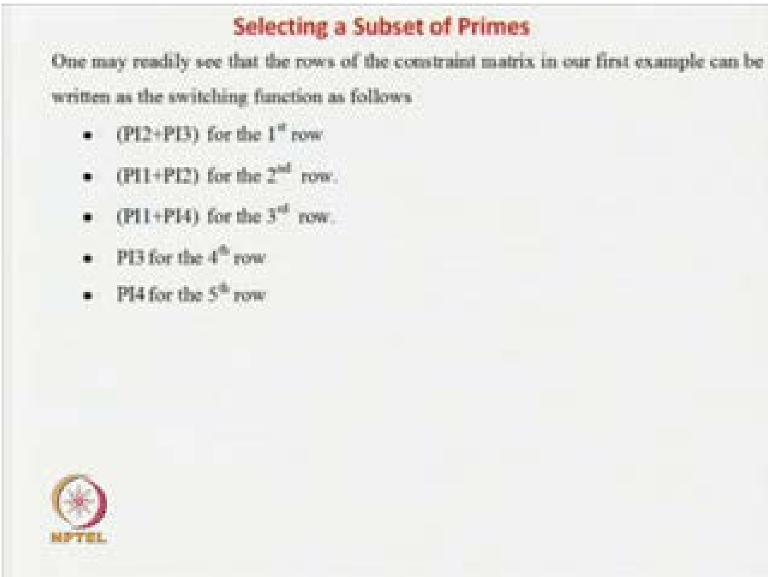
So what basically, so in previous examples we know that two important mechanisms. What was that reducing the constant matrix which are singleton and exploring the solution space. In case of cyclic that is resolving it first you reduce the matrix as much as possible. And we will land up where the cyclic matrix, where the reduction is not possible there is no singleton over here. So what we do, we take one column arbitrarily and try to find which are the next best solution. And find out next is better than the this one. Now if you have taken another column, this one solution space can be different. So you can drop this and start with this one and again see what is the cost.

Again it is a branch and bound kind of the problem. So you take one solution and see which are the solution available then you do not take that solution. And see another part of the solution and try to find out the minimum cost of solution in this way.

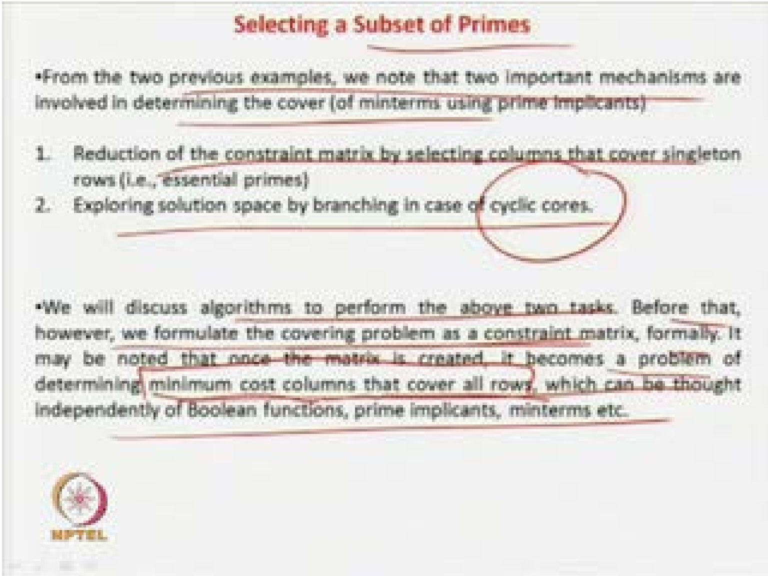
(Refer Slide Time: 46:06)



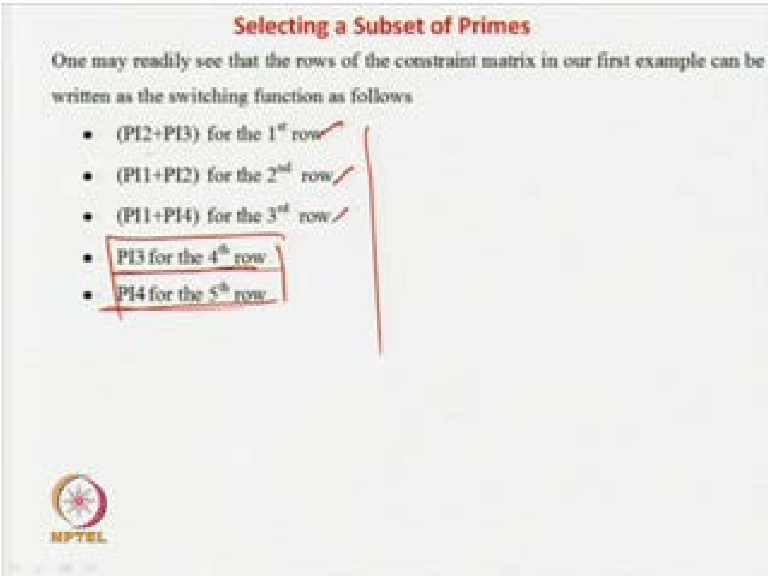
(Refer Slide Time: 46:47)



(Refer Slide Time: 46:06)



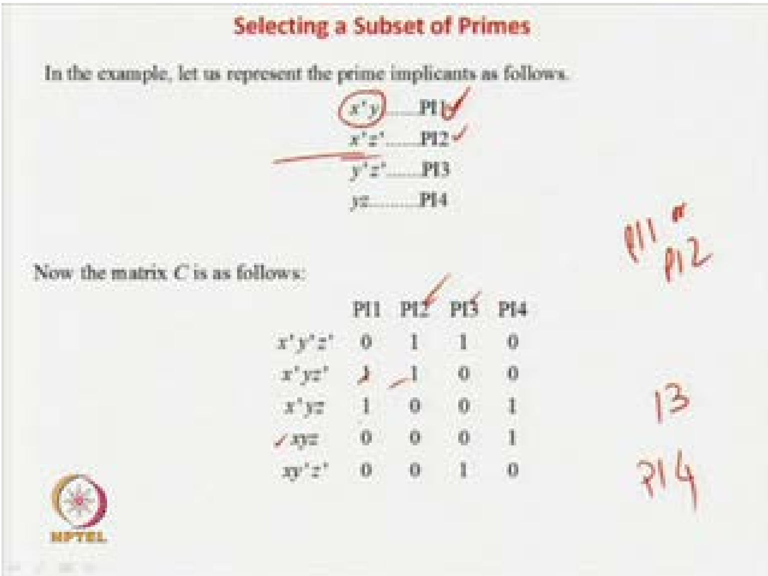
(Refer Slide Time: 47:09)



This is the special case when it is cycle and there is no reduction possible. We will discuss the algorithms to do in the next class. Before that we covering the constant matrix for before that before that we will see the covering matrix of the constraint matrix. Almost how the matrix will look like but, the minimum problems. All the rows can be covered thought of independently of Boolean function minterms and this one. So what you say is that now whole problem of sub set of sine selection problem is independent of prime terms, sub functions etcetera can be thought of all minimum cost columns to cover all this rows. So this is actually what we are going to do that is we have some rows of the matrix having 1; that means that is covering some row and then you have to find out the minimum cost.

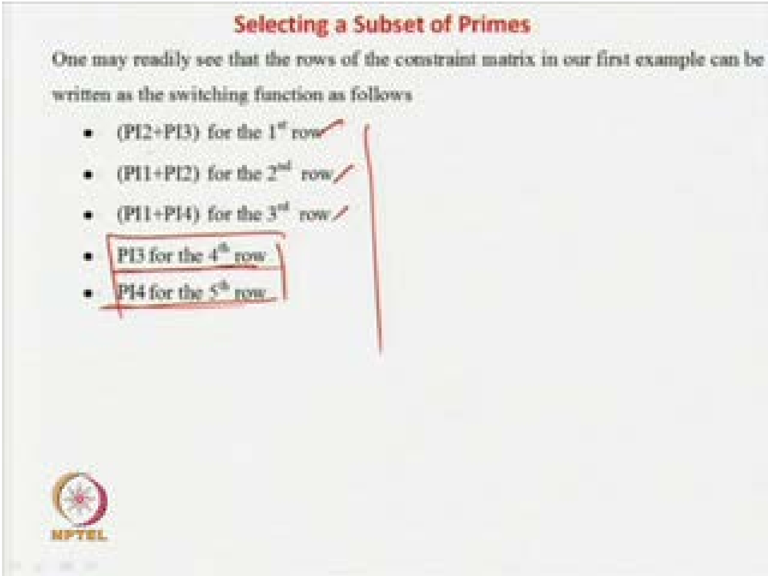
Minimum number of columns such that all the rows are covered. So that is actually called this one is the minimum cost that will cover all the problem. So this is the and you can forget the functions of circuit minterms etcetera. Now we will represent in the formal way as we said in this case we have seen that we have to cover 4th row and 5th row. So that is P I and this one is there. If you just look at it not this one so this example if you check, so P I 3 and P I 4 has to be covered. We write this way, so either P I 3 has to be taken, P I 4 you have to take for 4th row and 5th row. So they are singleton rows. So you have to take that now for the first row, second row and the first row first row can be covered by P 1 or P 3. First you look at the matrix you will find out just look at it.

(Refer Slide Time: 47:47)



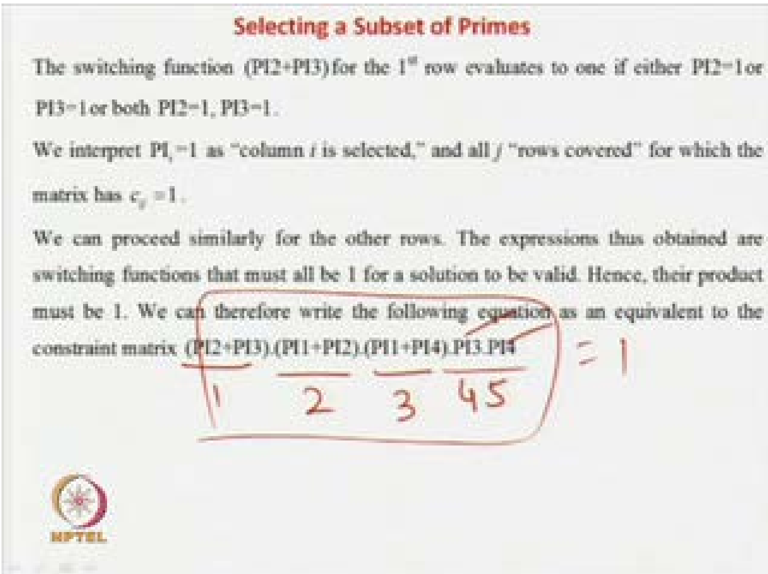
.

(Refer Slide Time: 47:09)



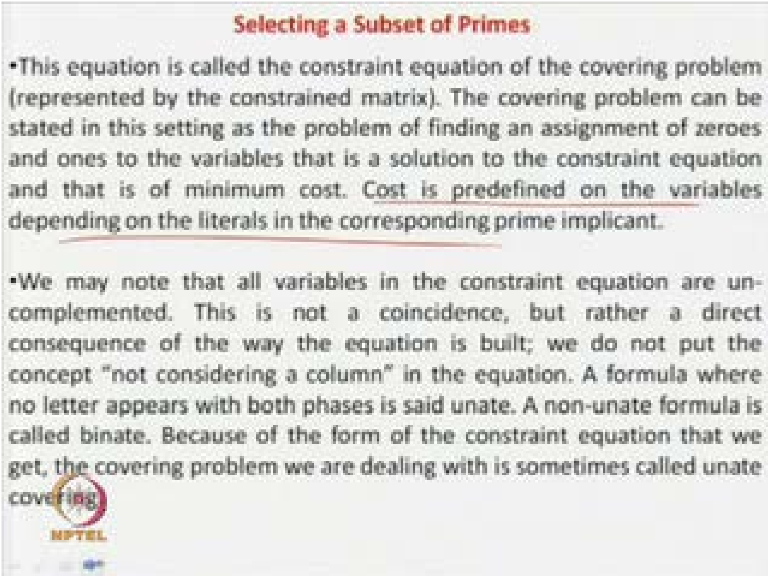
First row can be covered by P 1 and P 3 and second row can be covered by P 2 and P 3, third row can be covered by P 1 and P 4. If you just look at it this is how your equation is, so this are the singleton. So P 1 and P 3 are the P 1 and P 3 for the 2nd row and P 1 and P 3 for the 3rd row. So this is the equation for the term matrix..

(Refer Slide Time: 48:04)

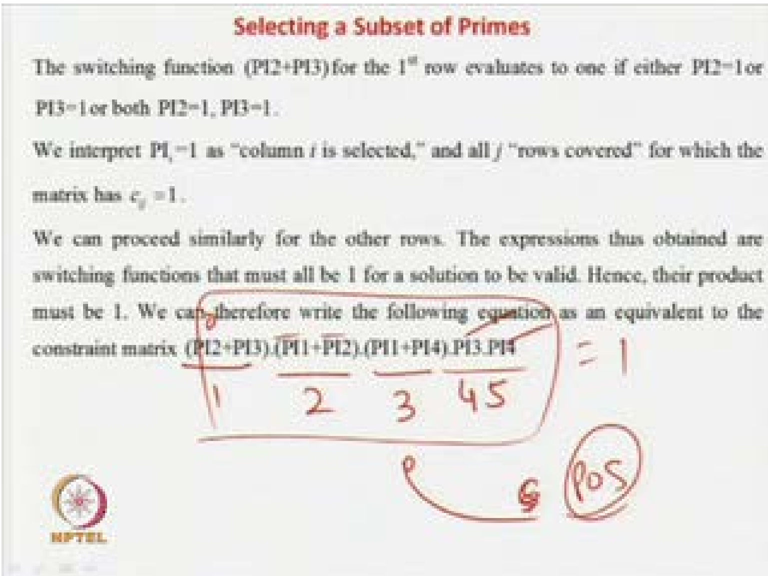


.

(Refer Slide Time: 49:04)



(Refer Slide Time: 49:58)



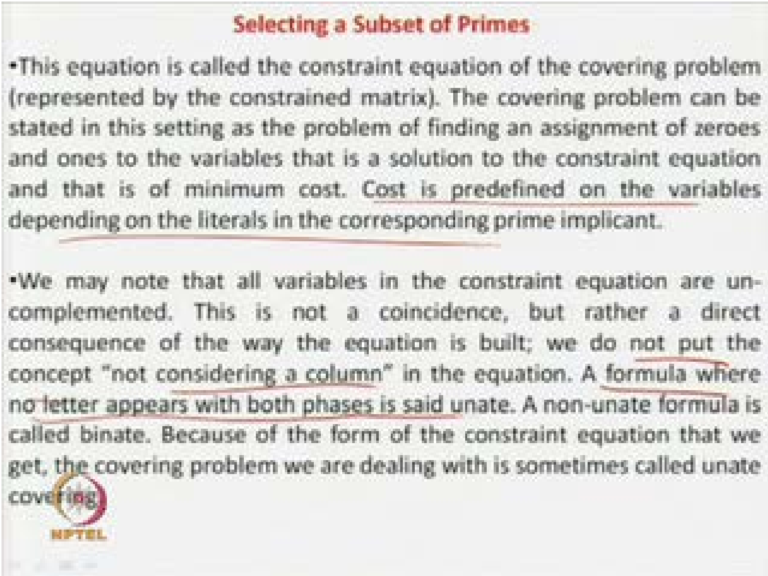
So now you can write this equation this way, P I 1 and P I 4 are mandatory. So you can either take this 2 or this 2 or this 2 anyone will cover the first second and third. So this one will cover the first term, this one will cover the second term and this one will cover the third term. So this one is for first term, this one is second term, for third term, fourth and fifth term. So singleton and mandatory, so you can take P 1 or P 4 or you can take P 1 or P 2 or so here one of them has to be true so here one of them has to be true. So here by default P I 3 and P I 4, so this is how you can write the whole expression, concerned matrix of the equation and then and find out the which one has to make it one minimal cost.

So more I can make all of them all the columns are covered, that is not equivalent so as you make minimum terms here 1, so that the minimum cost will be 1. Still I should get the answer 1, so the formal way of representing so this is actually called the constant equation covering. So you can just read about that So I have told you, you are given in the subset one thing is when you say about the cost. We have to predefine the cost of each of the columns. So how to define as I told you, we are forgetting the circuits, Boolean function and constraints. You can give a cost just say 1, 2, 3, 4 number of literals and prime implicants available.

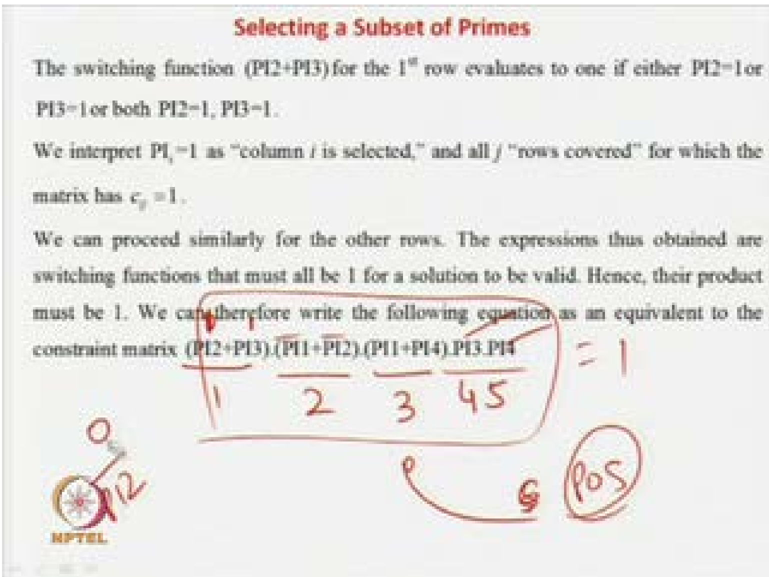
Then you will select the subset if there is a choice. Then you can take one column 1 versus column 2 and we will take the minimum of cost will be written in terms of literals. That is the minterms corresponding to the term. So one important thing is also there, so none of the terms are complimented. So this are Boolean kind of expression itself. So x P 1 is some kind of a pos forms right. Product of sums some sums and the product. So this product of sum has to be so this pos formula. Also one interesting part is that the, we do not have any upside here. So this is how you represent the constant matrix none of the term you can see compliment over here. So pos, we are representing the form in the pos formula but, there is no compliment over here.

So idea over here is not by chance it is a it is a way of organizing something like that. If it 0 we are not taking the column and if we writing that means we are taking 1.

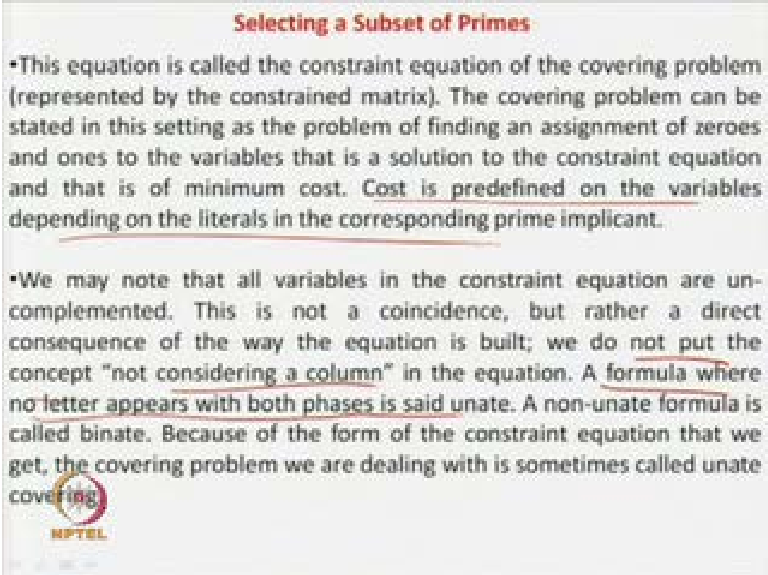
(Refer Slide Time: 50:47)



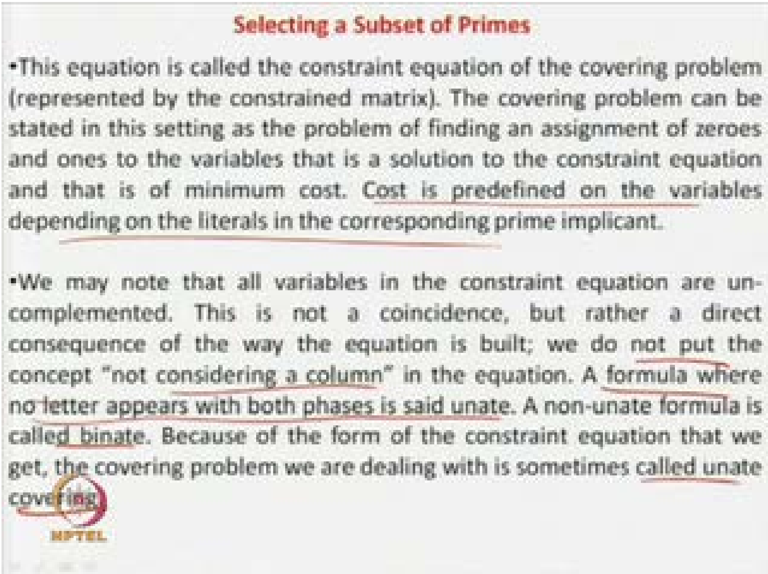
(Refer Slide Time: 51:00).



(Refer Slide Time: 50:47)

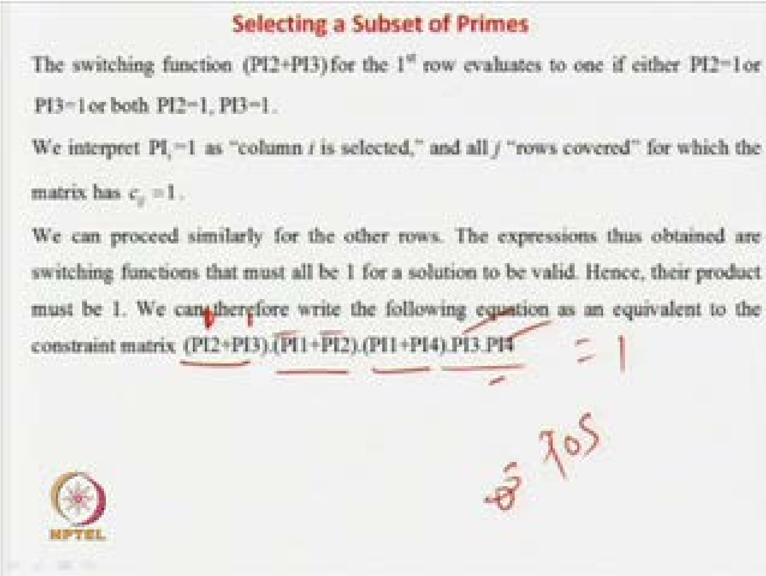


(Refer Slide Time: 51:19)

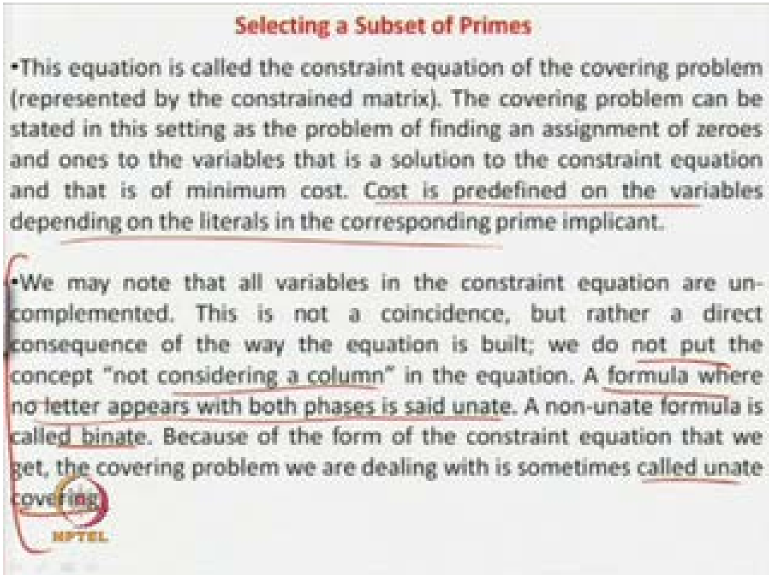


.

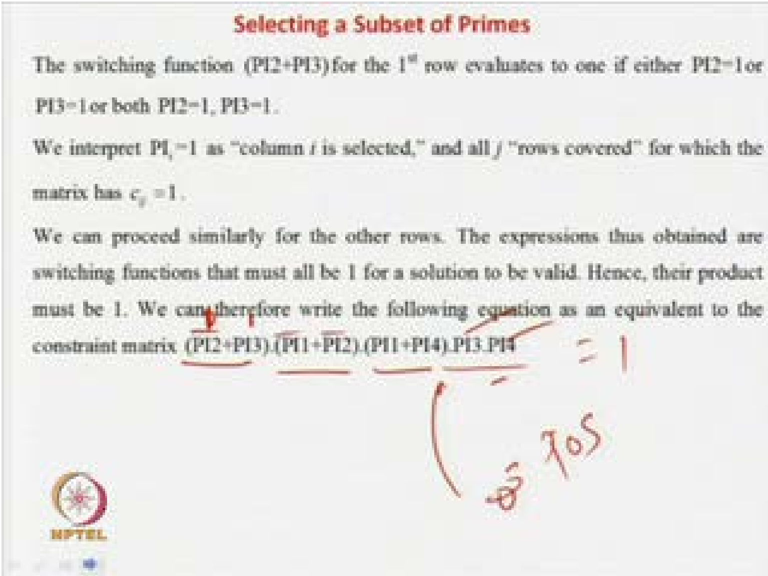
(Refer Slide Time: 51:29)



(Refer Slide Time: 51:44)



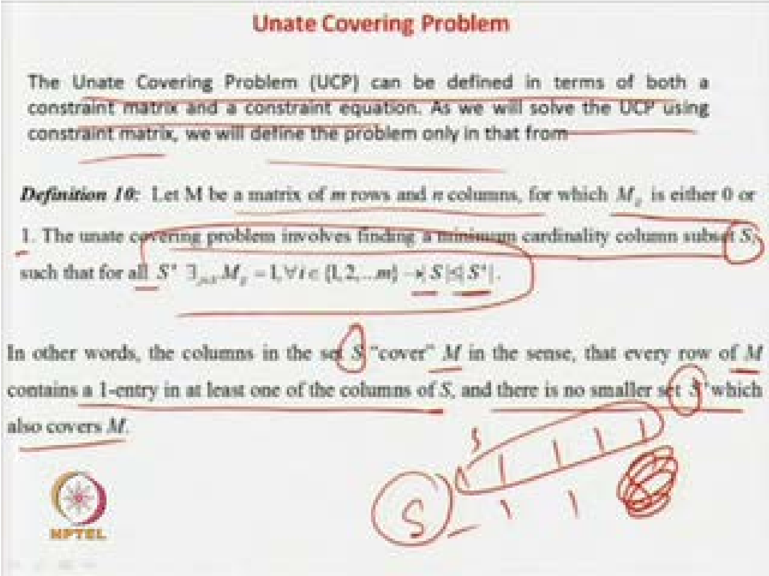
(Refer Slide Time: 51:58)



We are taking 1, we may note that all the variables are not compliment. This is not a coincident reduction, direct compliment reverse the equation. We do not put the concept not considering a column. Not considering means some value put at 0 a formula where no letter appears with both phases is said unite. So that means what, no concept of taking means that P I 2 inversion is not taking. You just put the inversion kind of the thing, this is not there. So obvious you have to take a 1. So actually this is our equation where there is no compliment, so that is actually called a unate formulate no letter appears in both the faces there is no compliment. All compliments, so that is also the one formula that is a binate because of the form we have constraint dealing with the this is called the unate covering. So in our case we do not have any complimented terms.

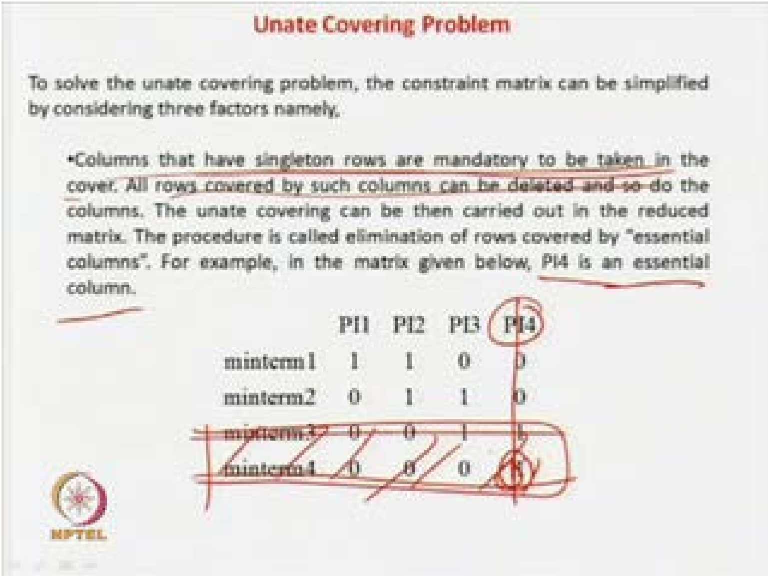
So in pos formula product of sum everything is in non product of form called a unate kind of the problem. So, sometime we call it a subset, covering some people called subset primes. And some people call the unate covering primes. This does not have any, simply in the lecture we will not be discussing. But, why some times we hear the name unate covering unate covering prime, why they call it this way, that is the reason. Because the pos formula representing which that is representing your constant matrix does not have any complimented terms. Because complementation does not have any meaning over here. So I mean that you do not require any row we do not say that P bar kind of the thing is you do not take a row, you put it as 0. That is it do not require to represent and required kind of stuff that is it right. So that is why it is also called as unate covering problem.

(Refer Slide Time: 52:16)



Sometimes you will be calling unate covering and sometimes you will be calling subset in the same way. So the unate covering problem can be defined in terms of both constraint matrix and a equation like this is the equation. The matrix like we have already seen then we will solve the unit covering problem of the matrix and we will be defined in the form. So in this lecture we will only be covering the tcp the net covering problem constraint matrix. And we will define the matrix as well. So there is the matrix definition you see.

(Refer Slide Time: 54:46)



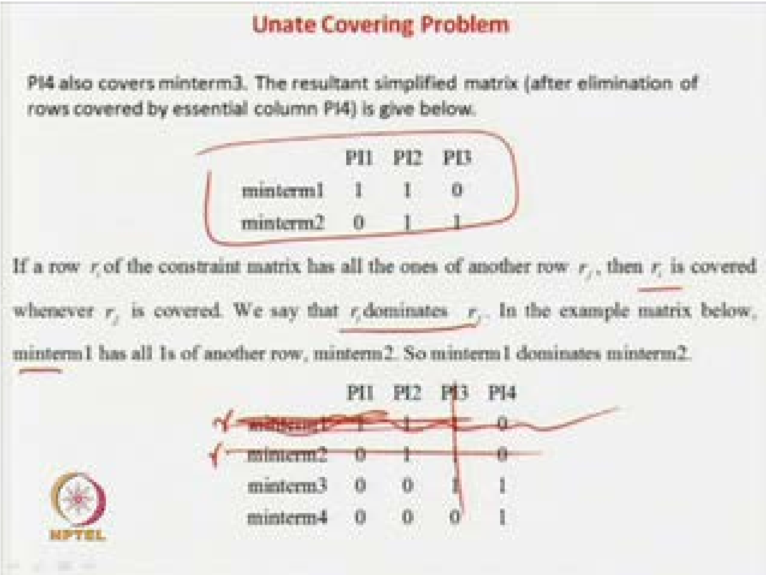
.

Let M be a matrix of m rows and n columns. mij is 0 or 1 depending on rows covers the problem unit covering problem finds the minimum cardinality columns of subset. So this whole set of described, so you have to find a minimum subset of a set of that column such that for all s there exist. a m belonging to s prime mj is equal to 1 for all I 1 dot dot n implies, that s is the subset of s prime. So what does that mean in other words the columns of set n cover n. So columns of s s you have to take subset actually in other words. The columns in the set s cover the m cover the matrix. In the same slide every row has a n, so that every columns have a and there is no smaller subset s, which also covers m. So what that means, you have to find a prime of subsets of column such that it covers m.

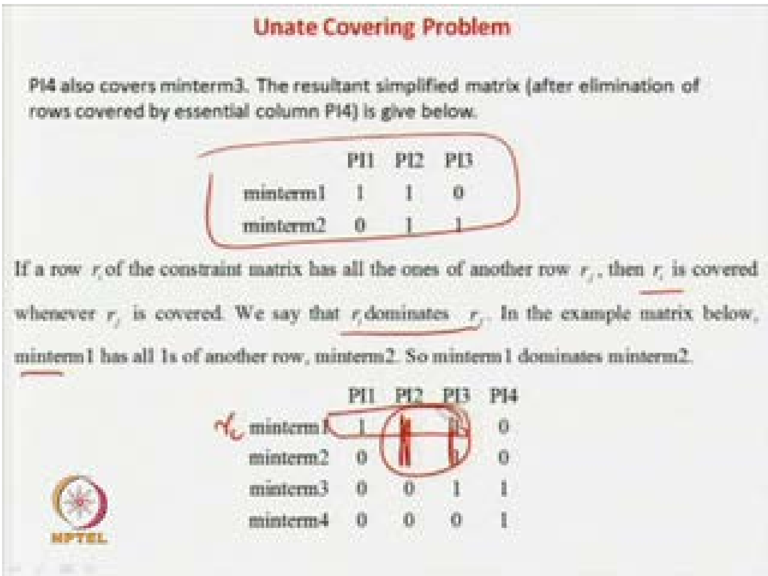
That says that row of m has a entry in at least one of the column of s and you say some columns like this and this covers m. So this 1 for this row and this 1 for this row at least there will be one ones for all the rows in the subset of the column s taken. And there is no another subset n which is smaller than the; that means you have already covered the minimum in other words what the complex mathematical equation says you have taken the subset of rows s and what does the sub set s do? And then all the columns in the s cover all the row s 1, all the rows; that means there will be 1s in all the rows and all the columns. Along with that you cannot find another set s prime which will also cover the matrix m. So size of s prime is less than s; that means s i the smallest subset possible.

So that is the formal definition of this one, now as I told you before I quickly I start how we will process the matrix before that we find out the procedure of minimal subset primes or the unate covering problems. So what we do, columns that here singleton rows are mandatory to be cover all the rows are covered by such columns can be deleted and to do so the columns.

(Refer Slide Time: 55:55)

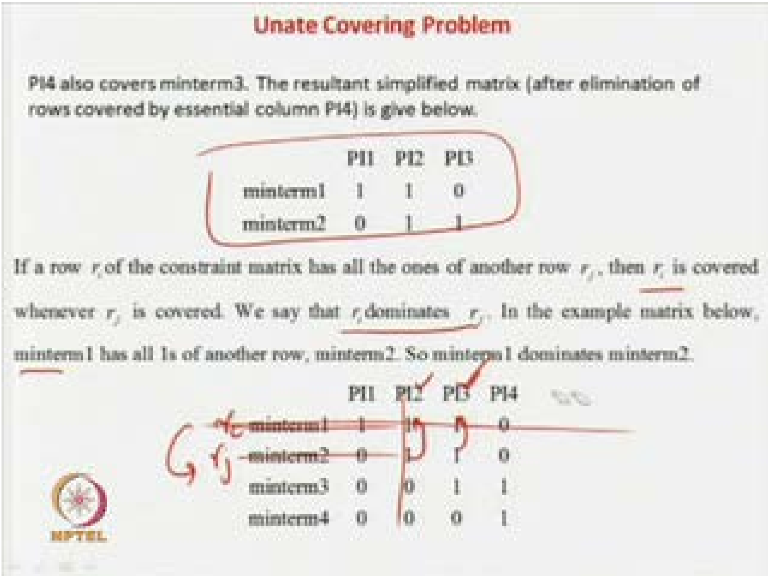


(Refer Slide Time: 56:27).

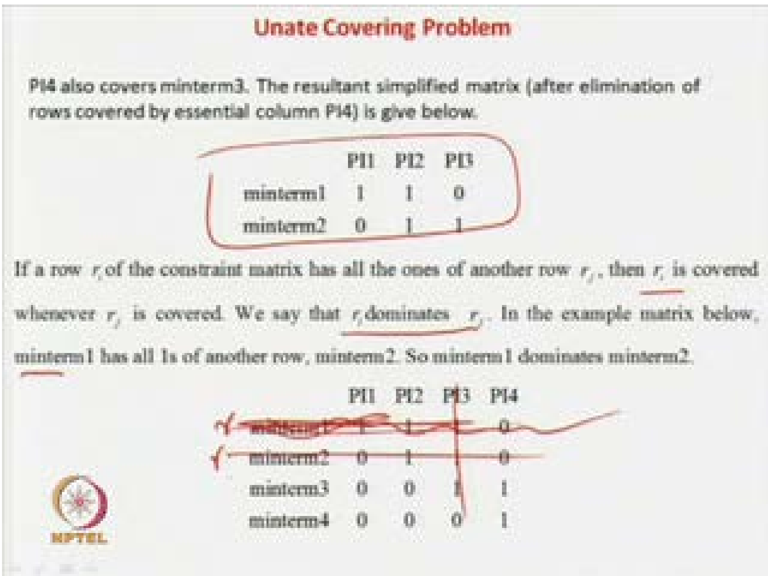


So this is the singleton row and this is, I mean example in the example for the given matrix P I is the essential column. So the arbitrary matrix this is why the all the rows which are having this. So this is the singleton matrix P I 4, obviously has to be there. This one is there you eliminate this and chance is that 1 P I 4. Nobody guarantees that on in here P I 4 minterm 1 1 is there. So minterm 3 is also covered. So the matrix becomes smaller, so this first preprocessing is to be done. First we will do process the singleton elements singleton rows and take the take the respective column and break the matrix. In this case P I 4 is an essential column essential way and write it accordingly.

(Refer Slide Time: 56:44)



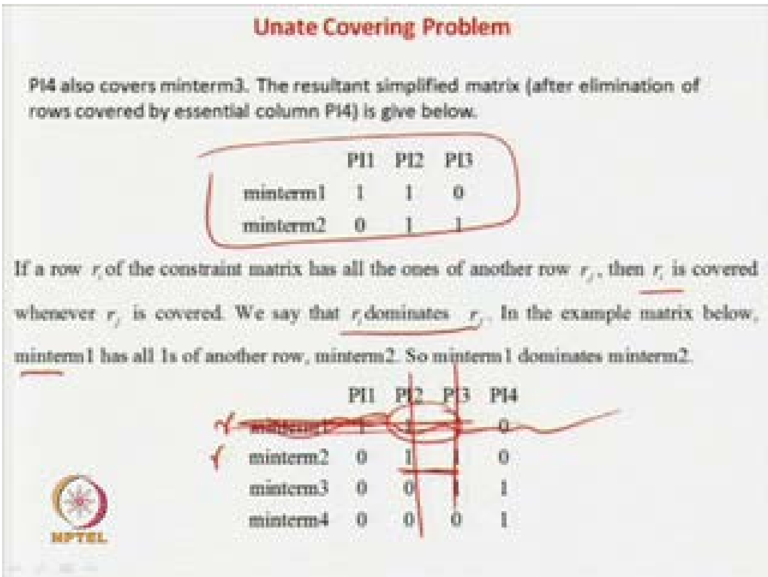
(Refer Slide Time: 57:54)



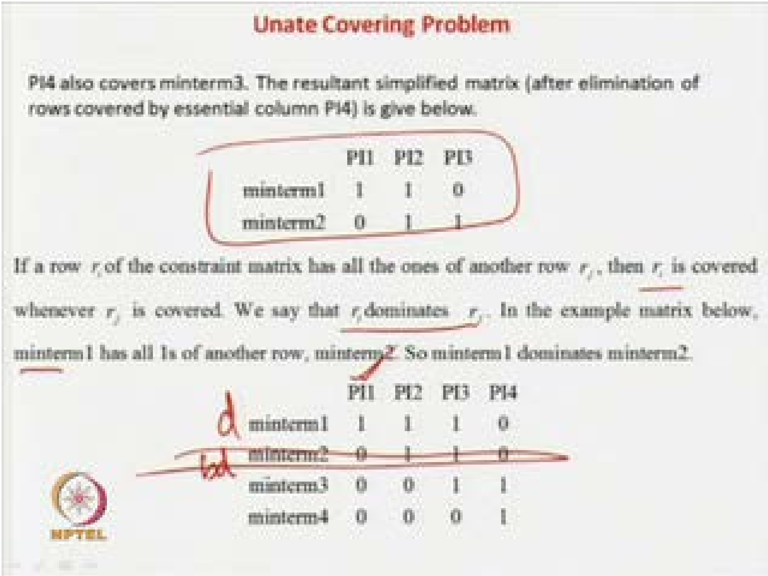
Now this is the simplified manner this is how we are doing next. Next we are reducing this if a row array of matrix of 1,s if all ht rj is covered rj sorry ri is covered, whenever the rj is covered we said ri dominates rj. So let us see matrix and come to definition. See the example below minterm 1 has all 1s of another 1 minterm 2. You see this 1 and 1 in minterm 2 this is common. So this is the larger part, so that means what is it saying? It is saying that for row ri of the matrix this is ri of matrix has all 1s, these are the 1 1s and these are the 1 1s. So it has row 1, in this case and ri all the 1s in the another row rj.

This is because this are some extra 1, this are allowed but, for row 2 you see it has all the 1s in all the position of 1s. And then in other words what I am saying for the minterm 2, rj in this position it has 1 in all the positions. So ri has a 1, so ri has for more that is not the, but, rj has 1 in all the positions, where ri has a 1. Then we say that ri dominates rj minterms 1 dominates r 2. That is what very interesting, so what we will do, so ri dominate ri dominates rj.

(Refer Slide Time: 58:20).



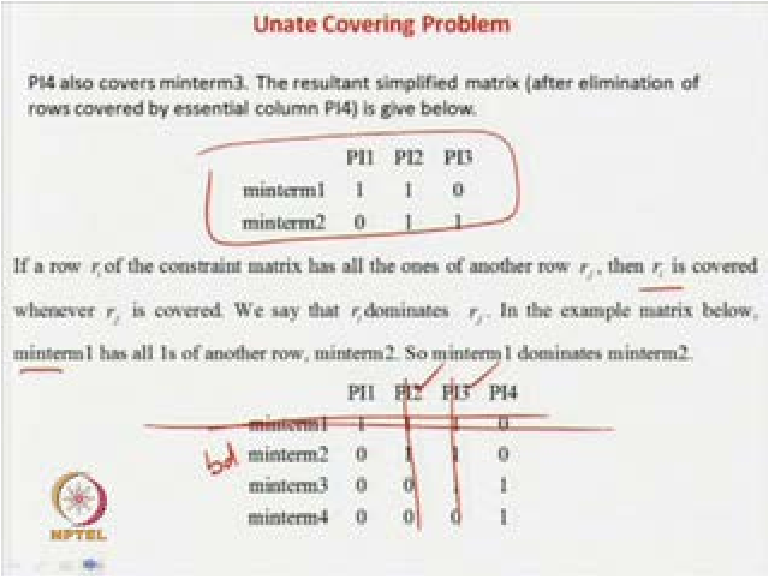
(Refer Slide Time: 58:44)



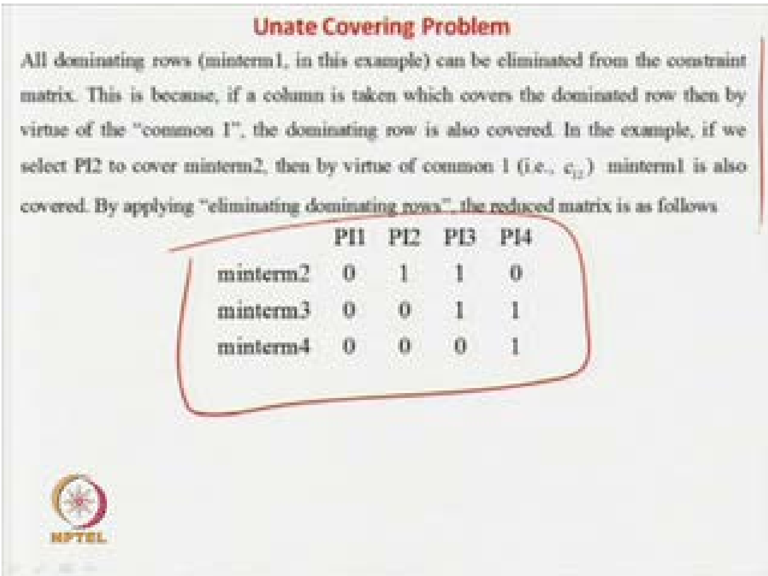
So in this case what we can say is that simply we will remove the dominant row. Now see interestingly what will happen. So if you are eliminating the dominating row dominating means that it has more number of 1s, then the row been dominated. So, some how you mean minterm 2, that is oblivious then they are arrested by P I 3 or P I 4. So whenever you take P I 2 minterm 2 will be covered and minterm 3 will be covered. I you take minterm 3 if you take minterm 3 to cover the row 2, this 1 will be covered by virtue this one will also be covered. So what happens is that if you remove the dominated row then there is no r.

Just remove the dominated row, so if you then dominating row if you remove row 2 dominating, that is row 2 will be there. So that means it has 1s in all the parts. All the ones in rows dominating. So in this you see these are the 1 2 plus 1 will be there. Obviously this 2 must be 1. So you will cover minterm 2. So either you will take minterm 2 or minterm 4. So automatically minterm 1 will be dominating. So that is called the elimination of the matrix or minimization of the matrix row dominance. This is very important interesting row domination will be, if I think that this is dominating row and this one is dominated being dominated. And dominating row. So if I do reverse row dominating, then there is a problem, minterm 1 will be a this 1 or this 1 and this 1 this 1. So if P 1 or P 3 automatically, I mean minterm 1 will be covered and minterm 2 will be covered by chance.

(Refer Slide Time: 59:16)



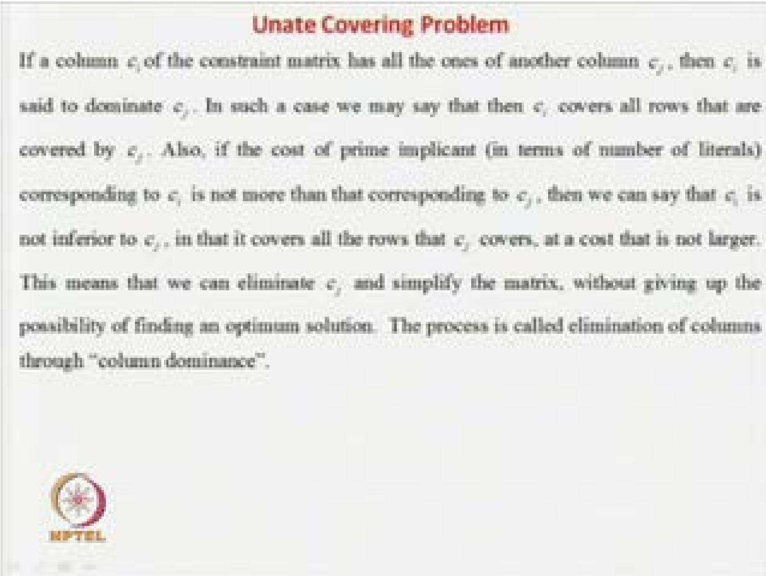
(Refer Slide Time: 59:55)



If I not select this 1 and I select this 1 minterm 1 will be covered and minterm 2 will not be covered. So that is why if I keep the row that is dominating, I may be in a problem if I keep the row that is been dominated I will remove this. I will keep minterm 2 obviously this can be covered by P I 2 or P I 3. So what will happen if I cover p I 1 1 automatically. Row 1 will be covered. If I cover P I 3 automatically 3 will be covered. So if I keep the row that is dominated automatically dominating row will be covered. So if I keep minterm 2, automatically row 2 will be covered. That is the interesting thing. So first you find out row which dominates and we remove that dominating row.

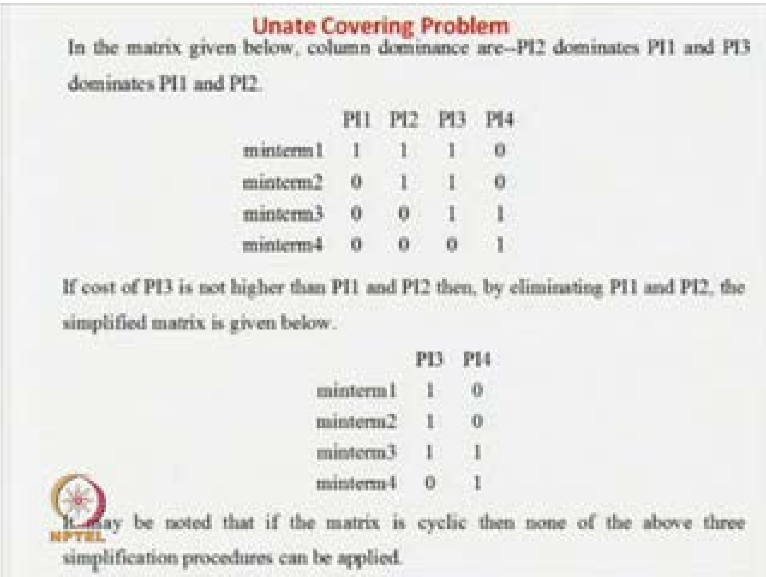
And the row being dominated will remove. So this case is the example and you can make the matrix smaller. So in this case you see sorry so in this case you see. We remove row 1 and finally, the matrix is this one the reason I told you is by remove the dominating and keep the dominated.

(Refer Slide Time: 60:04)



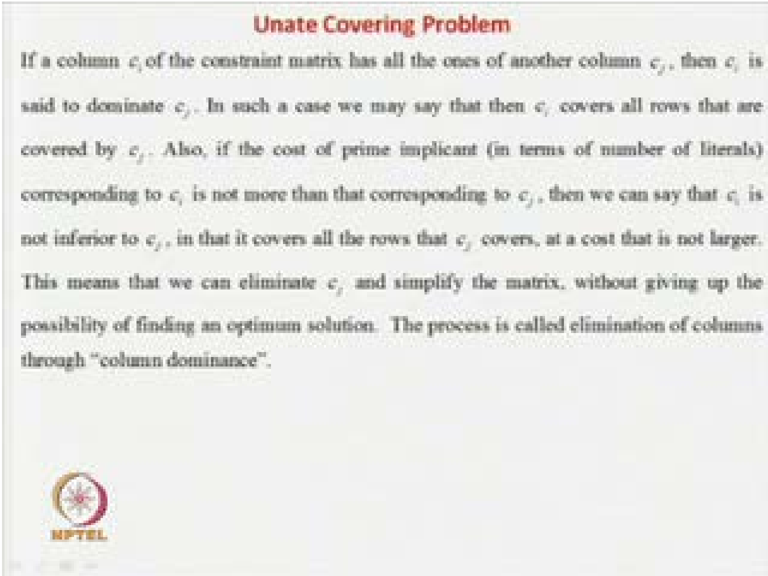
.

(Refer Slide Time: 60:09)

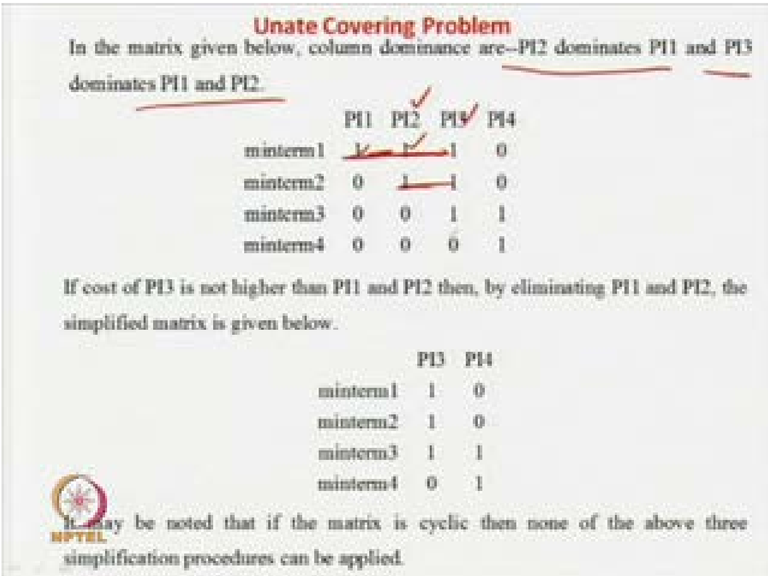


.

(Refer Slide Time: 60:04)



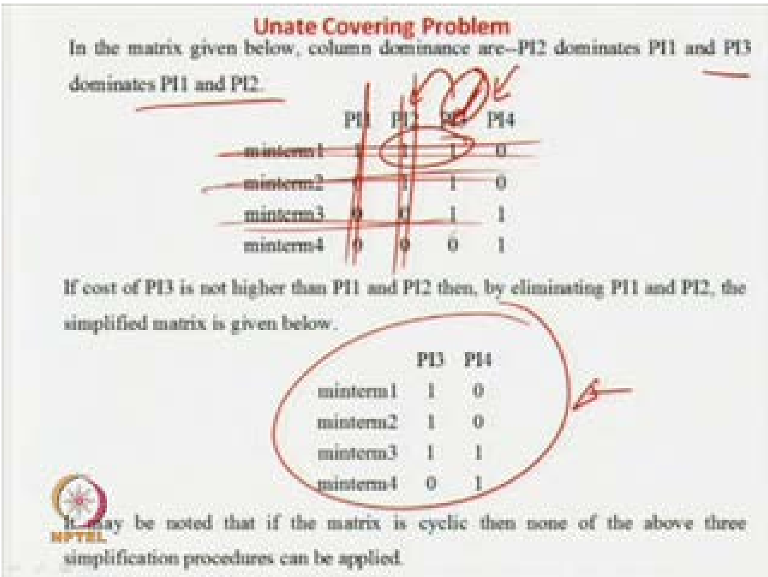
(Refer Slide Time: 60:22)



So now, yes now we will see about the column row. And so this is the definition. We will come back with the definition after the example again. In this case if you see you can say that you require a voice, if your column wise if you see row number 2 is having the 1 in all the place, 1 and 1 P I 3 has 1 in all the positions, row P I 2 has 1 and P I 1 has 1. So P I 2 dominate P I 1. Oblivious and P I 3 will dominate P I 1 and P I 2. So in this case P I 2 will dominate P I 1, so it has a 1 in this place and P I 3 is dominating both P I 1 and P I 2. Because it has 1s into this positions and 1 in this position. So P I 3 is the dominating row for both P I 1 and P I 2.So what you have to do in this case, how will you do it in this manner?

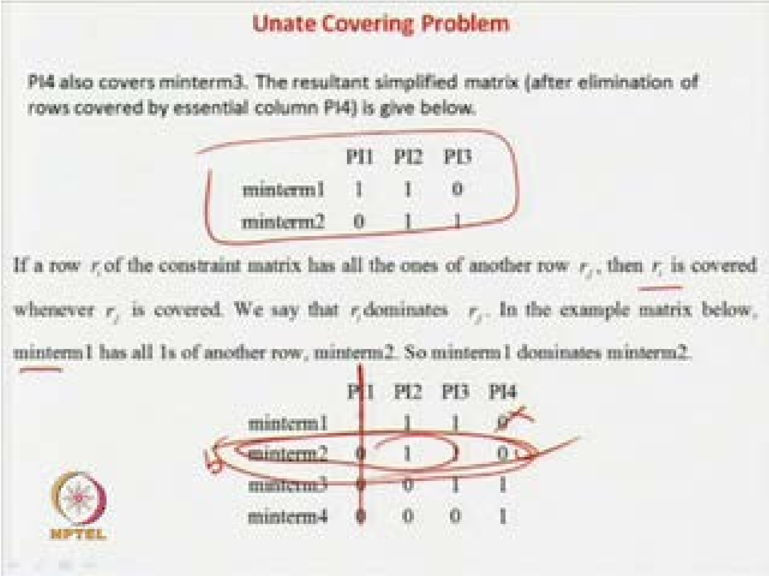
Here it is actually done in reverse way in row dominance. Row dominance what we have done? We have removed the dominating row, in this we will do the reverse we will actually keep the row which is dominating. In this case we will keep the dominating row.

(Refer Slide Time: 61:05)

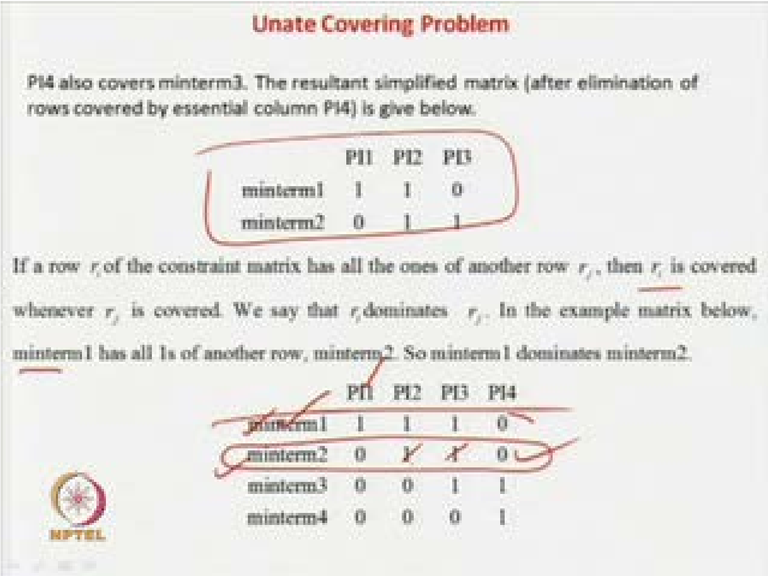


Now if I take the P I 3, obliviously this 1 will be covered this one will be covered and this 1 will be covered. So obviously we can eliminate the row by which it is dominated. So you have to understand the slight column difference. In the case of row dominance we eliminate the row which is dominated, in case of column dominance we eliminate the row which are been dominated. Just the reverse in column we just keep the row if you keep the column which are been dominating column of data, which is more number of 1's. It is better more the number of 1s in this column more number of minterms will be covered. So that is why I will keep this and we eliminate. This will be your reduced matrix. Now why, what is the philosophy? The philosophy is if the row has more number of one the terms.

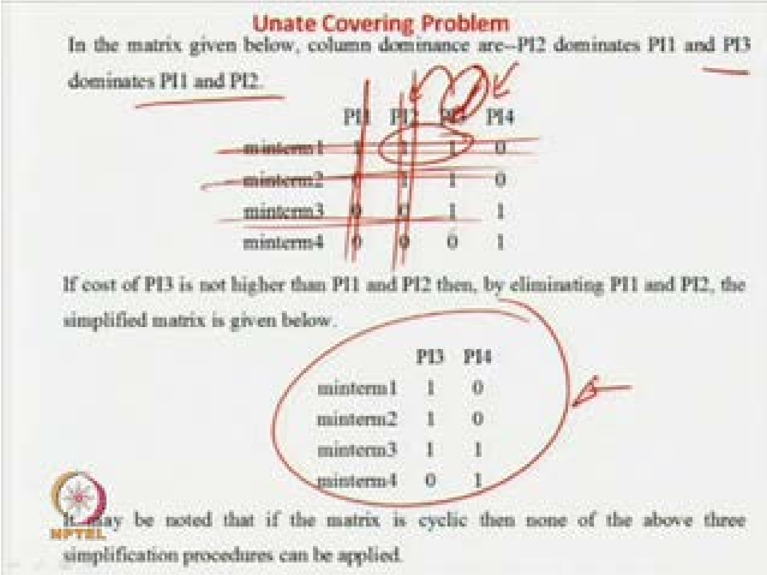
(Refer Slide Time: 62:37)



(Refer Slide Time: 62:54)



(Refer Slide Time: 61:05)



So now this row is dominating this one means, this column is dominating this column. Means this column has more number of this 1 and all in the similar position. So if I take P I 3, it will be covering this, this and this. So there is no more number of requirement of P I 2. Because these are common point in this one; that means P I 3 has more number of what you can say is that P I 3 has more number of P 3 and P I 2. If you take subset obviously there is no requirement of this. So automatically it can cover up the cases. So the matrix will be reduced, so in case of row dominance designations are same but, in this case keep the 1 with lesser number of 1s and eliminate the row which has more number of 1s.

Because if you keep the row with more number of 1, there is a probability that you keep the column may be covered and this row may not be covered. But, if you keep the row which is having the less number of 1s, if you keep the row then you have to cover the P 3 and P 4. So automatically it will be covered. If you keep row 1 may be in a problem. So row 1 will be covered by 1, which is not covering 2. In other words in row dominance and column dominance definition, I say in case of row dominance you keep the row which will be dominated that is less number of 1s in case of column dominance you do the other.

You keep the column which have more number of 1s, that is dominating. Because in column we require more number of 1 means more capability of covering the rows in a single. So prime implicants that is the idea, so with this we come to end of the lecture. In the next lecture what we will do, these are the 3 preliminary steps we have seen. First step is remove by finding out the singleton rows and go for row dominance and column dominance and then as small as possible after that it may become a cyclic matrix or some other matrix can be possible by row dominance and column dominance or singleton. You have you have to go for iterative or branch and bound technique to find out which is good solution or minimal solution of subset of the problem or the unate problem. That we will see in the next lecture.

Thank you.