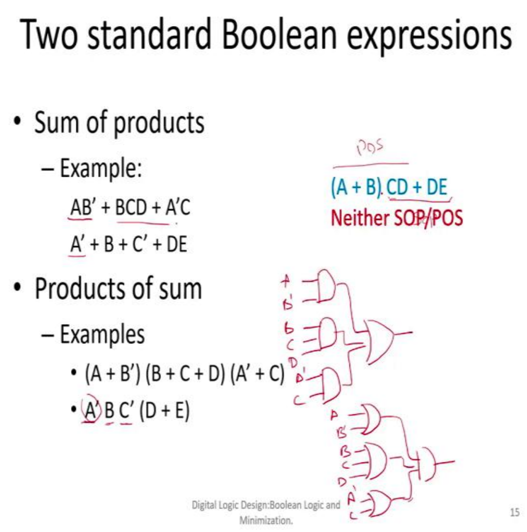
Digital System Design
Professor Neeraj Goel
Department of Computer Science Engineering
Indian Institute of Technology Ropar
SOP and POS Representation

Hello everybody. In our previous lecture we have seen how Boolean algebra works and we have also seen some properties of Boolean algebra and how various variables can be operated, we also learned that there are the 3, there are 3 basic operators AND, OR and NOT. In today's lecture we will see that he those Boolean expressions or Boolean functions can be optimized using the equations which you have learned in our previous lecture. Optimizing Boolean equations is very important or Optimizing Boolean expression is very important because whatever Boolean expressions are there, they represent the hardware cost.

So, let us say a Boolean expression is large, it has lot of terms and literals, so each term would correspond to a gate and each literal means that input to that gate. So, if we have lesser number of terms, lesser number of literals, then that would optimize the cost and that is the objective of our design. So, in this lecture we will see how we can utilize those Boolean equations or Boolean algebra theorems so that we can optimize or we can simplify Boolean expressions.

(Refer Slide Time: 1:52)



Two standard Boolean expression forms are Sum of product and Product of sum. So, in sum of products, like there would be each terms, so let us first understand the different between term and a literal. So, this is one term and A or A dash, B or B dash. So, basically one variable either in the complemented form or in the original form will form one literal, but when you combine them, so let us say there is a AND operator working over here, so this will be call a term.

So, when all the terms are actually product and then we are taking the sum or we are taking the OR, we are doing an OR operation over all these terms is called sum of product. So, in sum of product, you will see that A B dash, so B dash is a is itself a literal so we do not consider that NOT gate. So, A, AB dash is a product term, BCD is also a product term, A dash C is also a product term, and if we are sum, doing a sum or doing an OR operation of all these product terms are called sum of products.

So, here one more notion which we need to clear that this AND operation is also called product and whenever we are doing an OR operation that is also called sum. So, that is the algebraic notions which we have studied yesterday. So, in algebraic notions, whenever we are doing AB that means, we are multiplying or we are doing a product, but it is actually an AND gate. And similarly, when a whenever we are saying A plus B that essentially means we are doing an OR gate, but we are also doing a sum, so the sum is in the form of in the Boolean algebra. So, basically all the variables, all the literals they are assumed taking the value of either 0 or 1.

So, in the sum of product, this we can ask whether it is a sum of product or not, so here this is a very special case of sum of product where because this is only a single literal, there is nothing else getting multiplied to it, but it is still we can call it a product term. And A dash plus B plus C dash plus DE because all of these terms can be denoted as a product term, so it is also a sum of product expression.

The other form of standard expression is called product of sum expression. So, a product of sum expression we can take this couple of example. Again you will see that this A dash plus B dash, so this will form one term because this is sum of two literals. Similarly, this is sum of three literals, so this will form one term and if we are talking product of these sums, this is called product of sum.

Similar to this case, this also has a very similar example here that A dash B C dash. So, all of these 3 still form a one term which are multiplying in, multiplying to each other, but this will form one term, so the this is still classified as product of sums. It does not look like, but it is because each of these terms are individual terms which we can consider. Now, what is the significance of these two standard Boolean expression? So, if we have these Boolean expressions, we can implement these Boolean expressions in form of AND and OR gate and there is a standard like two level logic which we can implement.

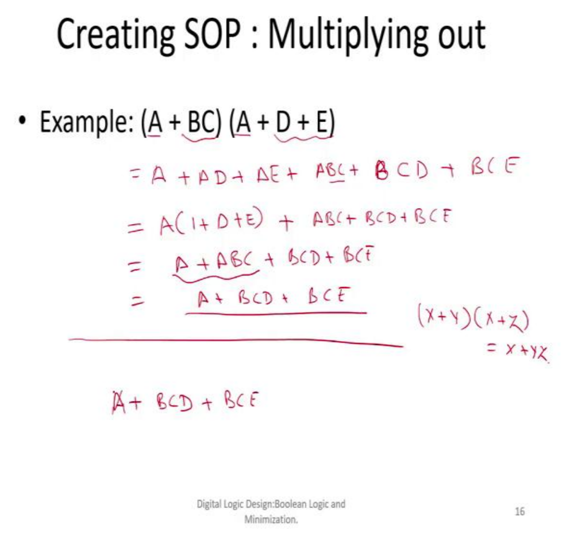
So, for example, if I want to implement this, so then I can have AND gate which we will say A and B dash, we are implementing this one, so another AND gate, another AND gate where B C and D, another AND gate where A dash is a input and C is a input and there could be an OR which could be given as the output. So, similarly all of these product of sum forms can also be represented using two level logics, so we call it two level logic because this is the first level, this is the second level.

The product of sum can also be, let us take this example so you see this is A, this is B dash, B C D A dash C and now the output is essentially product of them. So, if you see the complexity of implementation of sum of product or product of sum, they are essentially equal if the number of literals and number of terms are same. So, if you see here, the number of literals are 1, 2, 3, 4, 5, 6, 7, and number of terms are 3, so here also it is the case 1, 2, 3, 4, 5, 6, 7, so 7 is the number of literals and the total number of terms are 3. So, the total complexity of both of these are essentially same.

So, that is why we try to represent most of our equations either in the form of sum of a products or product of sum. So, sometime we get confusion whether it is sum of product or product of sum, so let us take this example. So, let us think over it whether it is sum of product or product of sum. So, if you see A plus B is we can (categori), if I so we have to see that this A plus B is getting multiplied with C and D so that means there is a this is a sum of product we can say or we can also call it is a one sum term, but it is getting multiplied with C and D which is a product term and again D and E is a product term, and it is getting sum here.

So, that means, this is neither sum of product nor product of sum. The reason is because it is a mixer of both. So, this part if I consider only this part, then I can say it is sum of product, and if I consider only this part, then I can consider it as a product of sum, but if it see in totality, so this part I can, I am writing this is SOP and this product of sum, but because product as well as sum, both of these terms are there, so it is neither sum of product nor product of sum. So, that is why if some expressions are not there in the standard form, product of sum are sum of product, we need to create them, we need to do some extra processing and maybe use our Boolean algebra which we have learned to create into standard forms.

(Refer Slide Time: 10:05)



So, let us see how do we create. So, I will take couple of examples, let us say I want to create sum of product. So, creating sum of product is easy because it is inline with our regular algebra. So, here because multiplication is very similar to our regular algebra. So, we can use those things and we can simply say multiplying out. So, let us say this is the example I want to create a sum of product. So, if I want to create a sum of product, so what will I do? I can do so basically I will first multiply A with all of these terms and then I will multiply BC with all these terms and then can simply it out.

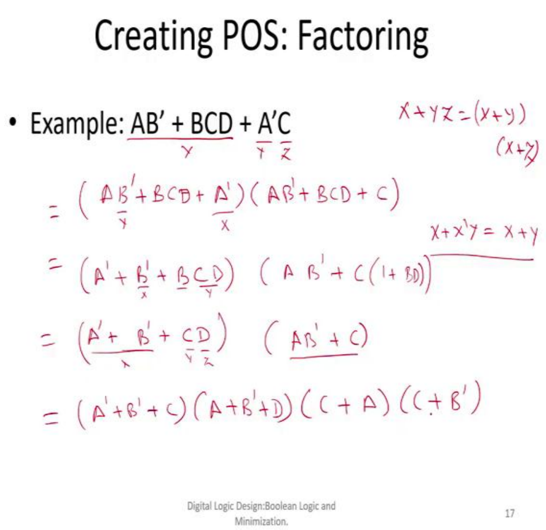
So, let us say if I want to do this, I will say A would be multiplied with A, it will still remain A, so and then after that AD plus AE plus now, A has been multiplied to all these three terms, now BC would be multiplied to all these three terms. I can write ABC plus BCD plus BCE. So, now because the purpose is not just multiplying out, but also to create a better form which is also simplified one. So, here I can see, I can take this A as common and I can say 1 plus D plus E plus here I think I cannot take anything better? So, ABC plus BCD plus BCE.

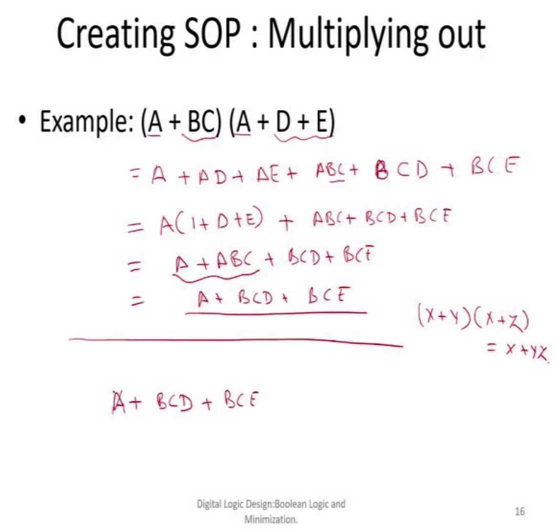
Now, I can, sorry plus I can also would have taken this BC as a part of this, so we can take that in a next step. A plus ABC plus BCD plus BCE. Now, there also we find A as a common, so I can similar to this, we can eliminate this, so this will become A plus BCD plus BCE. So, this would be the SOP form of this expression. So, you see that this expression is neither sum of product nor product of sum, but after multiplying out, we can create a good representation of this sum of product.

Now, let us see now we did this in a completely multiplication way, but is there a better way? Possibly yes. So, now if you remember yesterday we have seen couple of theorems, so one of the theorem which we can use here, so we can see that this is a common part so, this we can consider as a separate part. So, the algorithm, the theorem was saying that X plus Y X plus Z equal to X plus YZ. So, if I use this theorem, the theorem I can, if I use this theorem, then the alternative method would have been A I can say is X, BC is Y and D plus E is Z.

So, straight away we could have got this value A plus BCD plus BCE. So, this shows the utilization of those theorems which we have learned yesterday. So, they can make our task easier because when we use one of them, they will reduce the number of steps which we have, we are going consume. So, this is how we can create sum of product.

(Refer Slide Time: 14:40)





Now, let us see how to create product of sum. So, if I want to create product of sum, then the method would be that I would need to do factoring. So, factoring would essentially again use the same the same theorem which it says X plus YZ equal to X plus Y and X plus Z. So, this distribution of addition is what we are going to use for creating a factoring. Now, let us say if I want to do factoring, what will I keep as X, what I will keep at Y.

So, these things are little bit tricky but we can take any of them so, let us say we say we take this whole as X, this as Y and this as Z. So, I can rewrite from this theorem, I can write this as A B dash plus BCD plus what is the Y? It is A dash. Correct? And then there is a product form with again X, X is my A B dash plus BCD and then Z which is my C. So, I have started creating this, again I can do factoring in a recursive manner wherever there is any product term, I can go and simplify that.

So, if there is anything else which I can use for simplification, I can use those things as well. So, if I want to do it further, then the next theorem, next thing I can also do here, so if I see this part A B dash, BCD plus A dash, so this look like something where we can use some the some other, and so we call it elimination theorem. So, this elimination theorem we can use, elimination theorem says that X plus X dash Y equal to X plus Y. So, why do I recall and how do I say that I have to use elimination theorem that is a very tricky question but what do you observe?

We see that A is present in a non-inverted form as well as in this inverted form. So, that gives us a hint that there is some optimization which is possible. So, that is why we recall what all theorems we had, so we have used this elimination theorem. This elimination theorem here we have to keep this as X and this as Y. So, I can rewrite this as A dash plus B dash plus this will remain same because it is a plus term. So, plus term does not have any impact on this addition, this will become like this.

So, now we can do this, this product term we can do more simplification later, but let us first focus on this particular product term, sorry, sum term. So, here A dash B dash plus BCD. So, again I can use this elimination theorem and I can call it A dash plus, now B dash plus BCD. Do, again I have to use I have to call it B dash plus CD. Correct? So, why it happened because now I considered this as X and this this part as Y. So, this became X and X plus Y, so this become this.

Now, further I can use the factoring. In the factoring I can use this part of the theorem which is distribution, I can call it A dash plus B dash, so this part I can now keep at X and this is Y and this is Z. So, I can call it A dash plus B dash plus C and A dash plus B dash plus D. Now, let us focus on this thing. So, here, I can use A dash B dash, sorry this is not to be there, so this is A B dash plus I will say C and 1 plus BD. So, this part can be eliminated. I can rewrite this as A B dash plus C.

Now, I can again use my factoring to this distribution law to factor, I can say that this is C plus A and C plus B dash. So, this is how we can do factoring, whatever is the complex operation, any sum of product can be represented using product of some and vice versa is also possible and if something is mixed of sum of product as well as product of sum, then also we can use our factoring and multiplying out to create these standard expressions. So, this is one part of our lecture today.