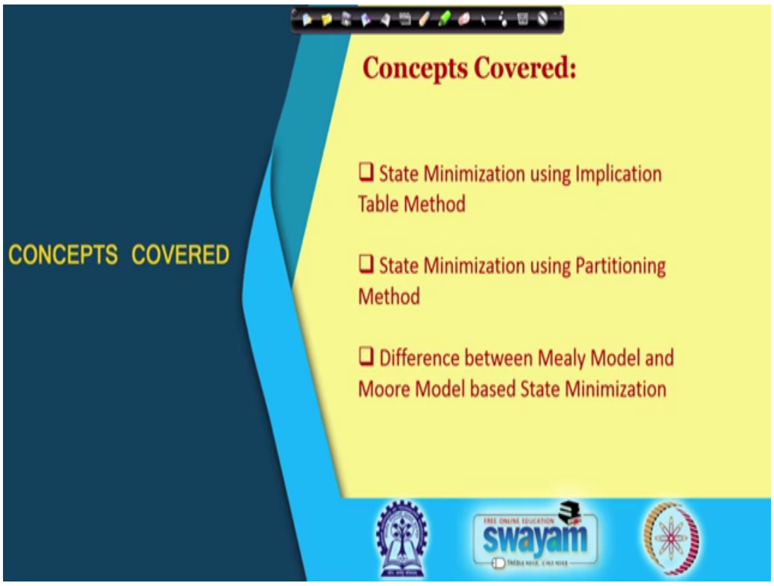
Digital Electronic Circuits
Prof. Goutam Saha
Department of E & EC Engineering
Indian Institute of Technology, Kharagpur

Lecture – 50
State Minimization by Implication Table and Partitioning Method

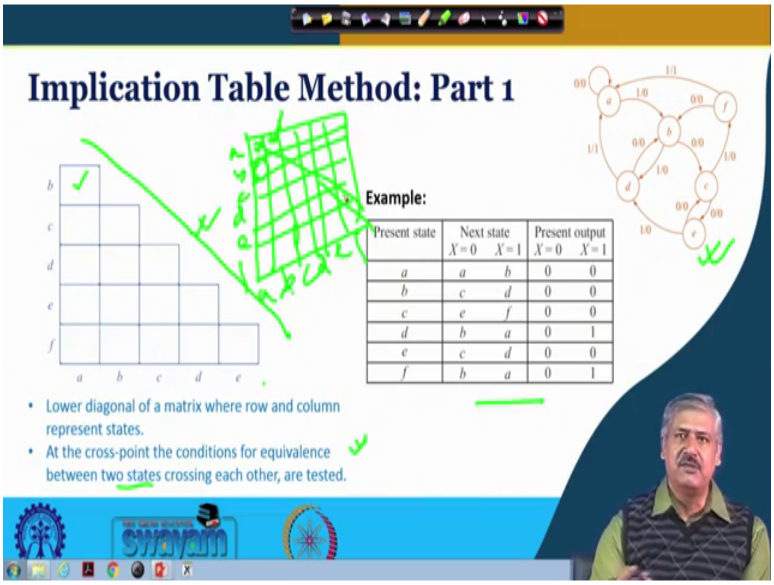
Hello everybody. In the last class, we ended with a discussion on State Minimization and we looked at one method called row elimination method. In today's class, we shall look at two other methods; one is called Implication Table based method, another is called Partitioning Method.

(Refer Slide Time: 00:30)



So, in the last class, we discussed Mealy model we shall continue with that problem statement only and later on we shall look at a Moore model based State Minimization problem and we shall consider the difference between these two ok.

(Refer Slide Time: 00:49)

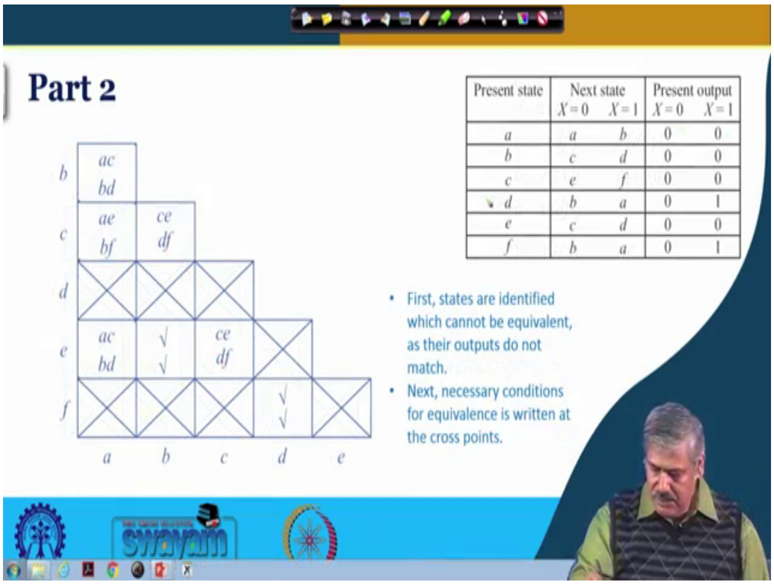


So, in the Implication table based method we have a particular table called Implication table made in this manner ok. So, this particular table you can see one side is a b c d e ok. So, we are continuing with the same problem right. So, the problem that we had seen and we used row elimination method, the same problem we are continuing with ok. So, six states are there a b c d e and the other side b c d e f right. So, one is missing here a is missing, another is f is missing here right and in some sense, it is a lower diagonal of a matrix where row and column represent states. So, what is happening here, if you can see? So, basically a b c d e and f ok; a b c d e f and similarly, this side one a b c d e f; a b c d e f.

So, this is the diagonal right ok. So, we are only considering this of diagonal these elements. Why? So, because this is; with in this implication table these are the cross points, these are the cross points. So, in this cross points, we shall consider, we shall test the equivalents between two states right. Now a and a are already equivalent, we do not need to consider that. b and b are already equivalent, we do not need to consider that. So, those are the diagonal elements right and upper diagonal is also not required because we are testing a and b over here.

If it is equivalent, then whether you test b and a, by this upper diagonal this element all the same; is not it? If they are not equivalent, it will not be equivalent by that also; is it fine. So, a b c d e over here and b c d e f over here because the diagonal elements, they are already equivalent with each other, the same states are there and upper diagonal and lower diagonal carry the same significance carry the same information. So, we do not need to the need them. So, that is why it looks something like this ok. Why it looks this way? So, this is the design fine.

(Refer Slide Time: 03:29)



What we do next? Now, at every cross point we shall be writing the condition for equivalence. So, first we identify the states; first we identify the states which cannot be equivalent as their outputs are not equivalent ok. We know the equivalence relationship right. So, the outputs are not equivalent. You can see that these are the outputs. Next state part we shall take later. First we are looking at the outputs. So, 0 0 0 0 0 0 0 1 0 0 0 1 so, d and f ok, they are having output 0 1 and rest of the cases are 0 0.

So, d and a where they are meeting? This is the cross point; since the outputs are different they cannot be equivalent. So, d and a you are crossing it that output they cannot be equivalent. Similarly, d and b cannot be equivalent because outputs are not matching. Similarly, d and c cannot be equivalent right; d and c cannot be equivalent; is it fine?

What about a and f? f is the another one where the output is 0 1. So, a and f cannot be equivalent; f and b cannot be equivalent; f and c cannot be equivalent right; f and e cannot be equivalent; also f and e because the outputs are different and d and e they also cannot be equivalent; d and e because the outputs are equivalent; outputs are different. Is it fine? So, this is the first thing that we would find and note ok. Next is we are looking at once this is done, for the remaining places we will look at what are the necessary condition for equivalence ok.

So, for example, this is the cross point for a and b. So, a and b will be equivalent outputs are matching. So, we do not have any problem with the output, now we are only bothered about the next state ok. So, a and b are equivalent if this is we are comparing. So, x is equal to 0, it is going to a; it is going to c, if a and c are equivalent right and b and d are equivalent right then a and c can be a and b can be considered as equivalent; is it fine right.

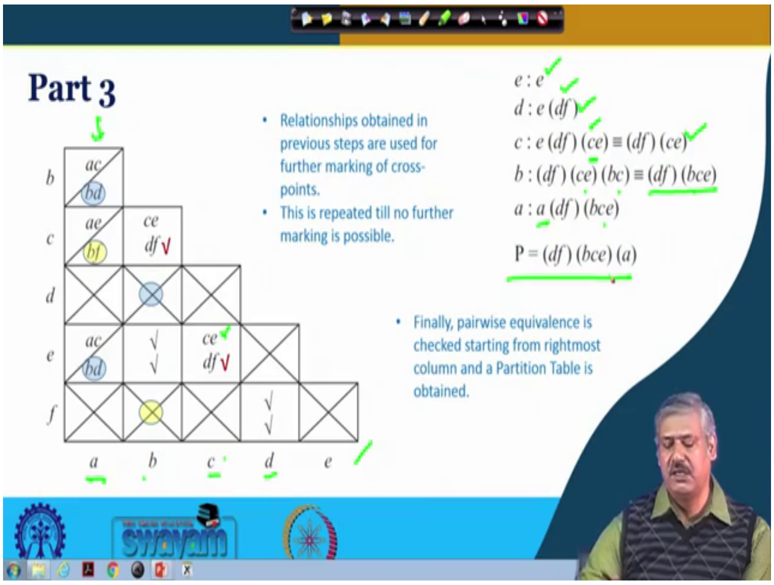
Similarly, a and c. When you compare a and c? So, that is you are comparing a and c. So, a and e and b and f, if they are equivalent right, a and c will be equivalent. a and e; so, you are looking at a and e. So, a c and b d, if they are equivalent; a c and b d if a is equivalent with c and b is equivalent with d, then a and e can be equivalent fine.

So, similarly b c, they will be equivalent. If c and e are equivalent; c and e are equivalent and d and f are equivalent fine; b and d cannot be equivalent, we have already crossed it out ok. Now what about b and d? b and d at this cross point we examine b and e. So, you see that it is c and c; c and c are equivalent. So, the first condition is put a tick mark because it is already equivalent right. What about the next one for x x is equal to 1? So, we see that b and d this is d and d right. So, they are also equivalent, already we made you know equivalent ok. So, we put both of them as tick right that condition for next state the equivalence is already established. Now, for c and e; c and e what we see? e and c that is c and e will be equivalent and f and d, f and d need to be equivalent ok.

So, the first thing from the last class the discussion we had the tautology and all we see that it is implied and d and f need to be equivalent and what about the other case d and f; d and f they will be equivalent, if b and b are equivalent and a and a are equivalent which are they I mean of course the case. So, we have put tick right that means, both are already equivalent from the state transition diagram or the table that we already have with us; is it fine.

So, this we have obtained in the first ah two steps right. First step with the output we are testing the equivalence, second we are writing down and some of the cases we have able to put tick because already the equivalence is there ok.

(Refer Slide Time: 08:26)



Next, we will go to you can say second you know it is iterative method next you know pass. Because earlier was pass 1, pass 2, then pass 3. So, in this what we are doing? So, relationship obtain in the previous steps are used for further marking of cross points. So, you will put more ticks or cross, based on what we have already seen in the previous steps ok. Whatever relationship we have already obtained. Now, we had seen that a and b to be equivalent a c and b d; need to be equivalent that you have seen right. But we have also seen that b and d, b and d they cannot be equivalent because the outputs are not equivalent ok.

So, this is the a blue circle that I have shown here. So, this is saying that b and d cannot be equivalent. So, if b and d cannot be equivalent so, this b and d, this is not possible; equivalence of then is not possible. So, a and b is not possible to be equivalent ok. Even if a and c can be equivalent because the other value for x is equal to 1; b and d need to be equivalent which is not the case because of this. So, we can put a cross mark over there right that a and b cannot be equivalent ok. So, b and d is also effecting this one alright. So, for which a and e cannot be equivalent. You have seen this over here. So, we will put this particular , this is also crossed ok.

Now, a and c to be equivalent, a e and b f needed to be equivalent. Now, b and f we can see cannot be equivalent, just with different colour to make us understand easily. So, since they are not equivalent, a and c cannot be equivalent even a e is equivalent also ok;

is it fine. So, this part we have understood ok. Now, we have seen that d and f are already equivalent because of this right and d f appears here, appears here. So, this is already ticked right. This is already ticked. So, what is required if c is equivalent and c is equal c are equivalent, then they will equivalent; is it fine? So, now what we do next?

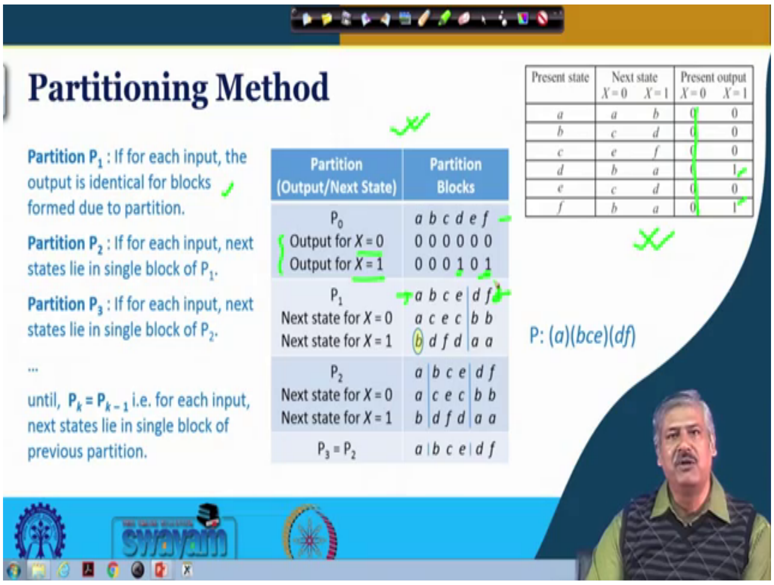
So, we could I mean if further you know proceed till there is a possibility of marking more cross and more ticks ok, but we can see that after this for this particular example here there is no further iteration required otherwise you would have continued; we would have continued from previous step to the next step with number of such passes by exploiting the relationship that are emerging in every pass. Is it fine? So, after that what we do? We now look at the final equivalent relationship by moving from rightmost column ok.

So, e from there we get this state e that is required from d, we will find that d and f are equivalent right. So, we can put them in ones particular partition within one block ok. Then here, we can see that c and e are equivalent, if c and e are equivalent right. So, that is from tautology that we know that we can establish an equivalence relationship because our objective is to minimize the number of states. So, we can write we can include this one. So, e d f c; now e and c we can put them in one block right. So, it is they are not two separate states. So, we can have an equivalent partition like this.

So, then we will come to b right what do we see here is that b and e are equivalent and b and c are equivalent. So, b c e are equivalent right. So, that is what has been written here right, is it fine? I missed one step here. So, d here, we can see that d and f are equivalent. So, that has been missed . So, we include that also. So, e then d f, then c is coming over here. So, here we are getting d and f are equivalent right so, d f, c e and b c; d f, c e and b c; so, d f, b c that what we get over here.

And finally, when you come to this, then we see that there is no equivalence that is they are along this column ok. So, a is there a is required. So, a d f and b c e. So, this is the final partition that we get. Is it fine? The same thing we obtained through row elimination method also, it has to be if it is reaching the final minimization and then that is what we see over here.

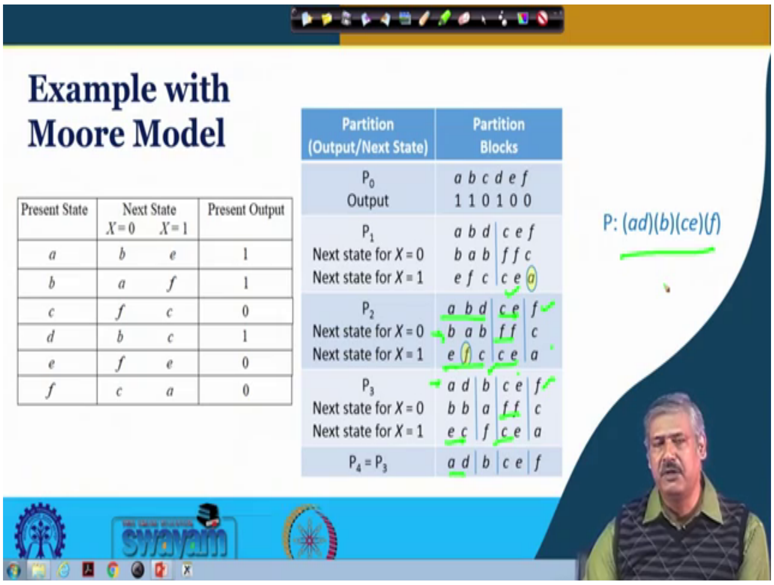
(Refer Slide Time: 13:55)



So, next we shall look at another method called Partitioning method. So, in the earlier method, what we had seen that initially they are all considered as separate and then, we tried we tried to group them ok, we tried to pull them based on equivalence and all. So, groups have formed like larger groups are formed. So, in partitioning method, it is just doing the opposite.

So, initially they are considered that they are all belonging to one group, one block and then, we keep examining it whether you know there is some such thing that differentiates them, that separates them and then, we put partition and then we go on doing this partitioning to the extent possible and after that when we there is no further partitioning is possible, we stop the process. Is it fine?

(Refer Slide Time: 14:54)



So, the same example we continue with right. So, this is the example that we had taken before ok; row elimination and implication table method, same example alright. So, here the; we have a table like this right, where we are doing the partitioning. So, here the partitions have mentioned as P naught, P 0, P 1, P 2, P 3, P 4 etcetera etcetera alright and this output or next state based on that ok. So, these blocks are defined. So, we shall look at that little later. So, initially we consider there is no partition P 0 all of them are put together; a b c d e f all the states six states are put together in one block. Is it fine?

So, first thing that we see is that for the output, for the output x is equal to 0 and x is equal to 1. So, what are the sorry, for input x is equal to 0 and x is equal to 1; what are the outputs ok? The way we had seen it before for implication table and all, so the same thing we are doing here ok; but in a different form, the you know presentation is different.

So, for x is equal to input x is equal to 0 and input x is equal to 1. So, the outputs we are writing it down. So, output is 0 for all the cases here right and output is 1 for d and f. So, d and f is it fine that you can see. So, based on that where the outputs are identical, they are put into 1 block and accordingly, based on the difference in the outputs a partition is made. So, what will be the blocks here? a b c e so, that will be one particular block and d and f because 0 1, 0 1 they are there, the outputs are identical. So, they will be 1 block. So, 1 partition will be made. Is it fine; this part is understood?

So, the first step is using the output for input x is equal to 0 and in input x is equal to 1; fine. So, we are at P 1. Now, after that after the first step, successive steps are based on what is the next state. So, next state for x is equal to 0 and x is equal to 1, we write it down ok. So, next states are from this particular table, we can see for a it is a and b; for b it is c and d. So, that way we complete. For c it is e and f; e c and d; d b and a; f b and a; is it fine right.

So, next we examine; next we examine if for each input next state lie in single block of P 1 right. So for each input means what so, x is equal to 0 and x is equal to 1. So, for x is equal to 0 is 1 input. So, the; so for this particular block a c e c, so all of them lie in same block of P 1 ok. So, then there is no issue. For d f, b and b they also lie in this block, first block; no issue right, no problem.

For x is equal to 1, we see that for b c e, d f d; d f d is part of the next block, but for a it is b, b is part of the first block. So, they are in different. So, a is the odd one out in terms of the next state whether they belong to the same block or not ok. So, d f d ok, they are all they are part of the second block, but this is not. So, b c e, this three states for x is equal to 0 and x is equal to 1 outputs are of one single block. This one are is of the first block; this one is the second block, but for a this is first block, this is also first block that is a b c e block and for d f what we see that this is b b and this is a e. So, basically they are in the same block ok. So, there is no conflict between them.

So, in such a situation what we do? We put a partition because a is different from b c e in terms of this particular b not being in the same block that is second block as it is what b c e, this states where it is d f d ok. So, what we do? We put a partition a b c e and d f right and corresponding a b c d e f with c d b a b a right. So, these are the corresponding outputs ok. You have corresponding next states.

Now, we further examine whether now that a new partition is been formed whether situation changes. So a, it is a and b so, a is only single member. So, there is no nothing like partition that is possible. So, b c e what we see that is x is equal to 0 c e c. So, it is over here in one block and x is equal to 1; d f d, it is there in the second block so, no conflict ok; so, all are alike.

For d f it is b b that is this block a a; this is this block ok, the first block. So, no conflict again here; so, no further splitting or part partitioning of d f is possible ok. So, at we see that from P 2, we get P 3 these are the partition and further partition is not possible ok. So, whenever this P k is equal to P k minus 1; P 3 is equal to P 2, P 2 we will stop it right and then, what are the equivalent blocks, equivalent states? This from here, b c e from here and d f from here. So, three states same thing what you have seen; is it fine.

So, you have seen this mealy model example and now we shall end with a Moore model example right. So, let us see how it works? So, for which we have a different this table right. So, present state next state and relationship now in Moore model, they will be no two different output for x is equal to 0 and x is equal to 1; is not it? So, there will be only 1 output based on the present state; is it clear? Moore model output is generated only from the state right; Mealy model from current state as well as current input. So, because of the difference in the input x is equal to 0 or x is equal to 1, output can be 0 or 1.

So, accordingly you had two such columns in the output stage. Here only one column will be there; is it fine? So, that is the thing that we take note downright and these are the corresponding states b e a f; these one just example f c b c f e c a for this corresponding six states and outputs are 1 1 0 1 0 0 just an example right. So, since this is the difference, the first step will be of course will be differentiation using output ok.

Now, the difference is with Mealy model is that earlier there was two such rows x is equal to 0 row and x is equal to 1 row for two sets of output, which is not the case here; only one set of one row will be there ok. From that, we shall make a distinction whether one group or the other group, they have different set of outputs or not that is the thing. Otherwise, this steps are same ok; only thing here for the output what we see that only 1 row is there in the first block while getting P 1 from P naught, P 0 right unlike Mealy model, rest of the steps are similar.

Because rest of the states are depending on next state only and next state is depending on x is equal to 0 x is equal to 1, it goes to 1 state or the other right. So, two such rows will be there in every case. So, in this example to start with a b c d e f all of them are in 1 block. Now we see that 1 1 0 1 0 0. So, this is the thing so, a b and d right, they will be in one block and c e f they will be in another block; is it fine that is the first state we have done. Next state; next step is now you are looking at x is equal to 0 and x is equal to 1 next states. So, a b e right from this table we are importing it; b e a f, d b c, c is f c, e is f e and f is c a ok.

Now, what we shall do? For every such you know input each of this input, we shall see the next states whether they are in the same block or the different block or not. So, for the first case what we see a b d this particular block, the next state for x is equal to 0 is b a b right and b a b right. So, all of them are belonging to the first block and next is for x is equal to 1, it is e f c all of them belong to same, this block. So, no partitioning is required in this particular case right.

Now, what about this one? c e f, this particular block f f c; f f c all of them belong to this block and whereas, c e a, c e belong to this block, but a belongs to the other block. So, this is the odd one out and this next state becomes different. So, we have to make a partition. So, this is the partition what you see here; f c a that is been separated out.

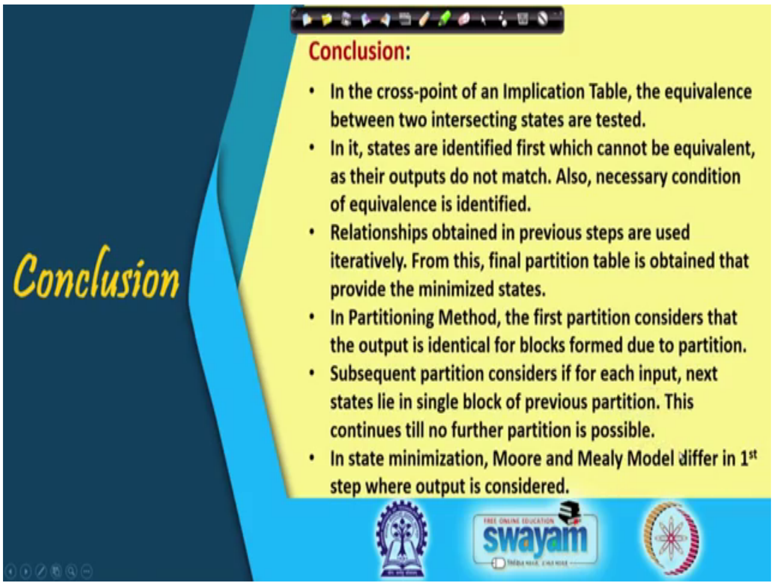
So, next what we will we shall do? Again, we shall examine because of this partition whether you know further partitioning is required or not, some status of the previous block has changed or not. So, that we shall do; so, this is a b d; so, this was b a b. Again, we examine because now there are two partitions; I mean so, three blocks are there earlier 1 partition, 2 blocks are there right. So, this is b a b. So, b a b all of them belong to this block.

Next is e f c; e and c belong to this block, f belongs to another block right. So, this is the odd one out. Now, for c e, f f belongs to this block; c e belongs to this block. So, there is no conflict right. So, only thing we what we examine that this b is to be separated from a and d and a partition is required. So, a and d are there and b is put over here right and further we examine we see that for x is equal to 0, it is b b belonging to one block; e c belonging to one block only; no issue or b it is only one member; so, no such partitioning possible. c and e f f this blocks c e same block.

So, we see that no further partitioning is possible. So, your P 4 is same as a d b c e f d 1 that we had seen for P 3. So, it stops here and this will be the equivalence. So, four such states will be there from 6, we have come down to 4 right. So, we need 2 flip-flops to realize it. Earlier if we had not minimized it, 3 flip-flops would have been required and associated complexity would have been more; is it fine? So, this is the difference between Moore and Mealy model and for the other cases also. Other methods also the same thing only the first output stage first stage when output is considered that could be different ok.

(Refer Slide Time: 27:40)

(Refer Slide Time: 27:42)



So, with this we come to the conclusion for implication table method in the cross point, the equivalence between two intersecting states are tested and in it states are identified first which cannot be equivalent as their output is not matched and after that the relationships in previous steps are used iteratively and final partition is obtained to provide minimized states. In partitioning method, initially we begin with all of them put together the first partition considers the output is identical for blocks formed due to partition and subsequent partition considers the next state ok, whether they lie in single block of previous partition or not ok. This continues till no further partition is possible.

And difference between Moore model and Mealy model is in the first step where output is considered; Moore model output is only from the current state. So, the reference to x is equal to the input is not there and remember ah just one more point before we conclude. So, there we considered two such only one input right; had there been say two inputs; so, x 1 and x 2. So, then there would be 4 rows 0 0 0 1 1 0 and 1 1 and accordingly, the next states value and those are the things that need to be considered ok. So, this is another thing that we take note of if the numbers of inputs are Moore ok. So, with this we come to the completion of week 10.

Thank you.