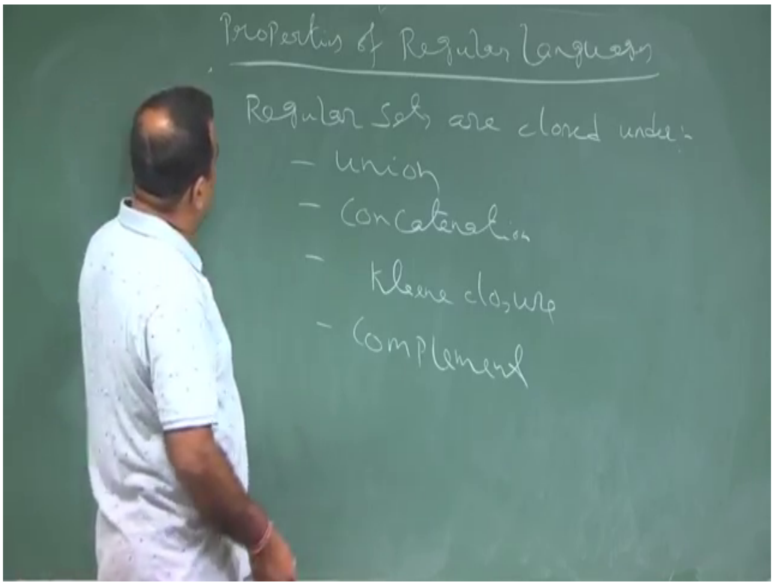
Introduction to Automata, Languages and Computation
Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

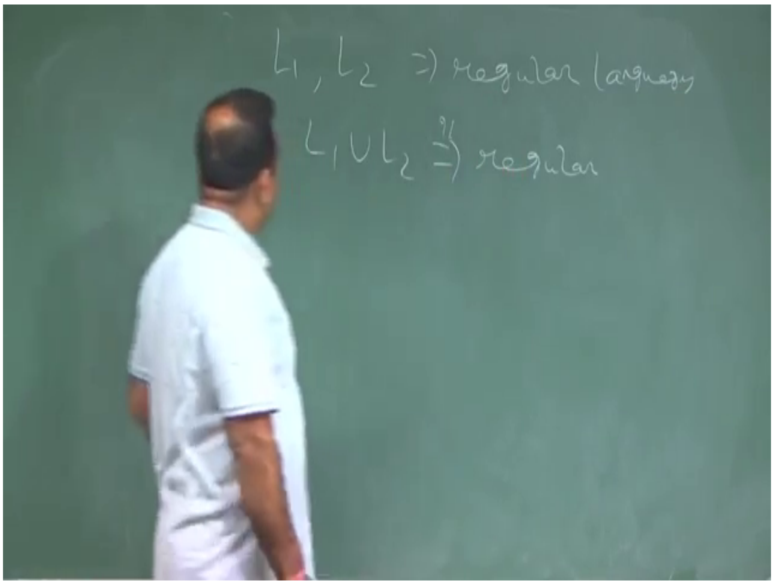
Lecture - 24
Closure Properties of Regular Set

(Refer Slide Time: 00:17)



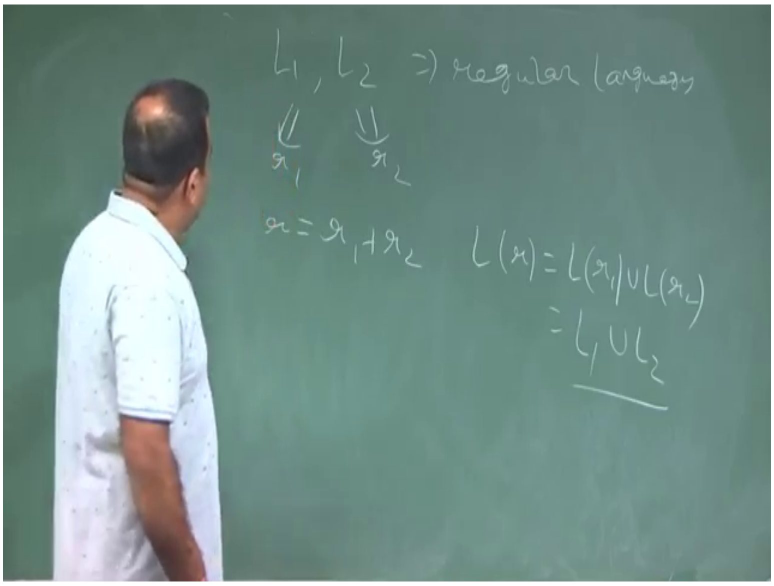
So, we are going to talk about the regular language and their properties that means, if we have given 2 regular languages that they are in union, intersection, concatenation, complement is regular or not. So, this type of properties. So, we want to see whether this regular language is close under this properties like; whether a given a regular language is closed under regular set are closed under following operation on: union, intersection, union intersection if a union concatenation, then closer (Refer Time: 01:25) I mean then the complement, intersection, reversal all these things we are going to study now.

(Refer Slide Time: 01:53)



So, first question is whether it is closed under union than that means, if you have given 2 regular set, L 1 and L 2, L 1 and L 2 the question is L 1 union L 2 is regular or not I mean. So, regular means this is we can have a constructive proof for this, but we can use the help of the regular expression to prove this that is the easier part I mean immediate from the regular expression. Because, if this is a regular language if these 2 are regular languages or regular set; that means, we have a; that means, there is a regular expression for this and there is a regular expression for this. We have a DFA which is accepting this and this.

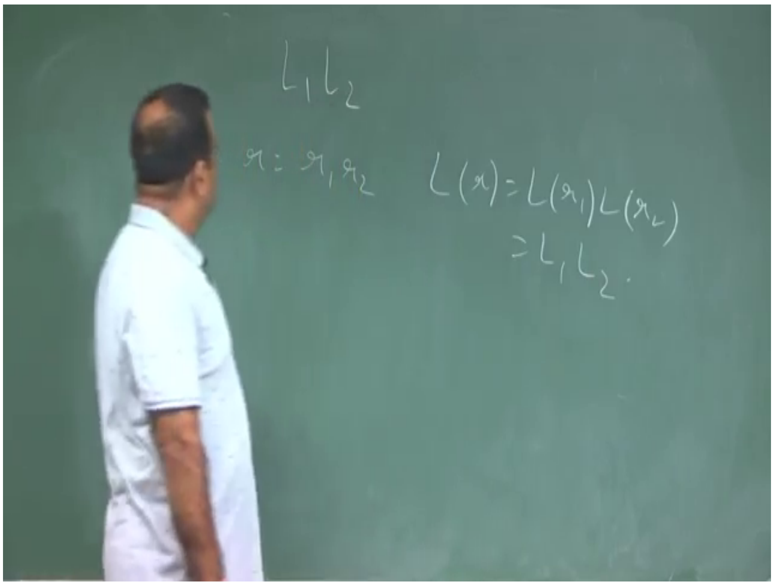
(Refer Slide Time: 02:35)



So, then we have a regular expression for these 2, then we can have regular expression this r, which is the language of this is nothing, but language of r 1 union language r 2.

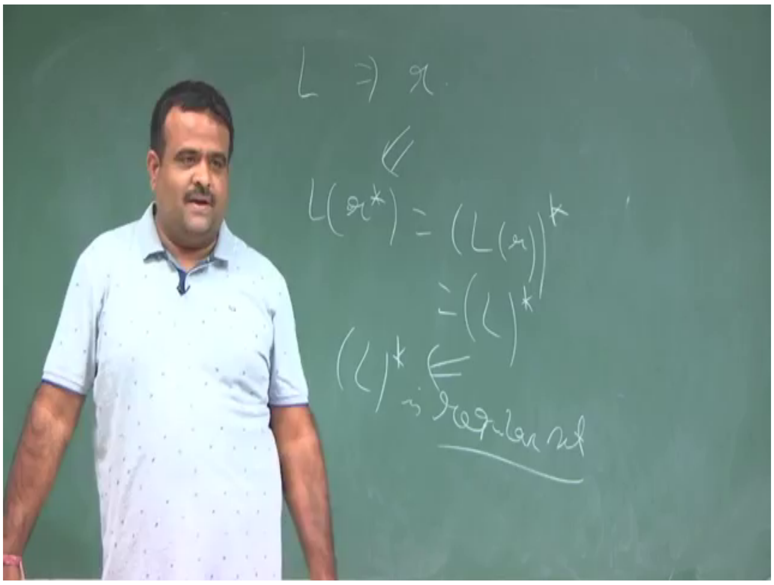
So, this is L 1 union L 2. So, this is regular. So, once you have a regular expression, then you their that will corresponding to the epsilon NFA that will cause that means, it is a regular language. So, that is straightaway coming from this thing regular language. Now even the concatenation if r 1 or r 2, if L 1 and L 2 is 2 regular language or regular set.

(Refer Slide Time: 03:33)



Then what about L 1 concatenating L 2, that is also coming from this r 1 r 2 and this is the language which is accepting L 1 L 2 because by definition of regular expression this is. So, this is L 1 concatenating L 2. Now, closer if L is regular.

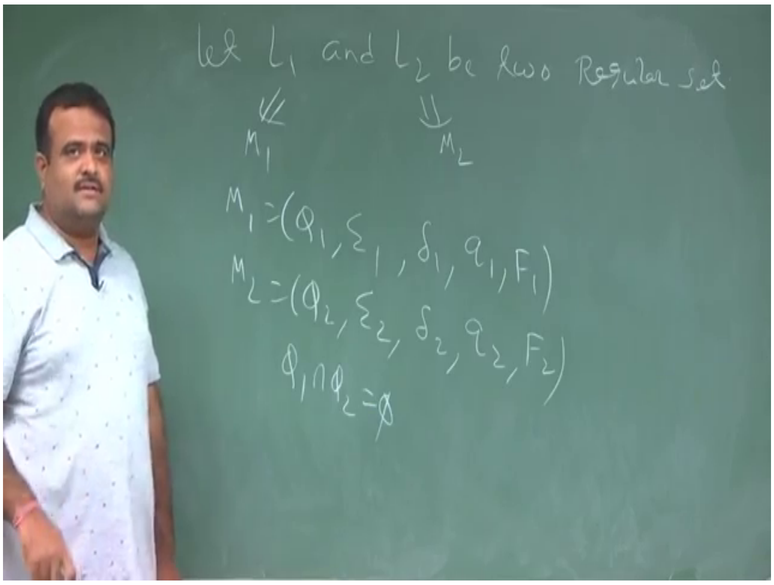
(Refer Slide Time: 04:07)



So, it will corresponding to a regular expression r then, this imply r star is also regular; that means, the language of r star is nothing but the closure. So, this is L star. So, this implies L star is regular set.

So, this is with the help of regular expression which is immediate coming from the construction of regular expression. But we really direct we want to have a direct construction of even the intersection may not be straight forward from here, now we want to have a direct construction of the DFA which will accept this. So, let us try to do that. So, let us take 2 lang.

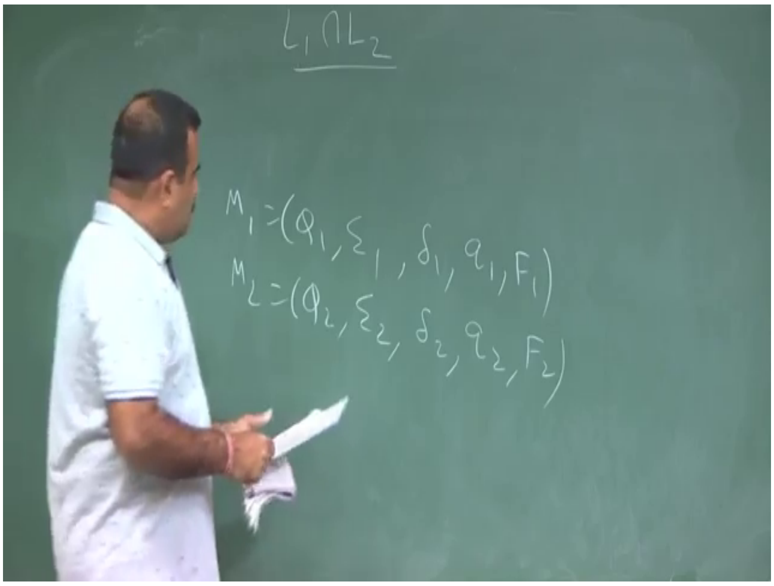
(Refer Slide Time: 05:05)



Let L 1 and L 2 be 2 regular set so; that means, L 1 will corresponding to a DFA M 1 sign to a DFA M 2. So, now, we are going to construct. So, you say M 1 is say Q 1, sigma 1 or maybe we can assume the same sigma, does not make any difference delta 1 and then q 1, F 1 and M 2 is Q 2 sigma 2 delta 2 F 2 we can take these 2 are same.

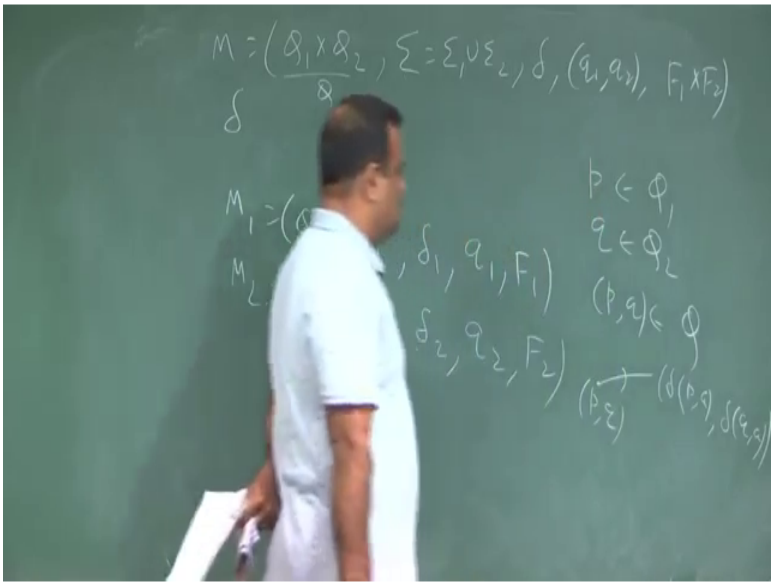
And Q 1 Q 2 we can assume there have no common, I mean Q 1 intersection Q 2 same things because we can always rename the states, that is always possible. So, now given this now we want to construct a DFA which will accept the language L 1 intersection L 2.

(Refer Slide Time: 06:27)



First of all this is our then if we can construct a DFA which is accepting using these 2 DFA, then we can say L 1 intersection L 2 is regular. Now how to construct?

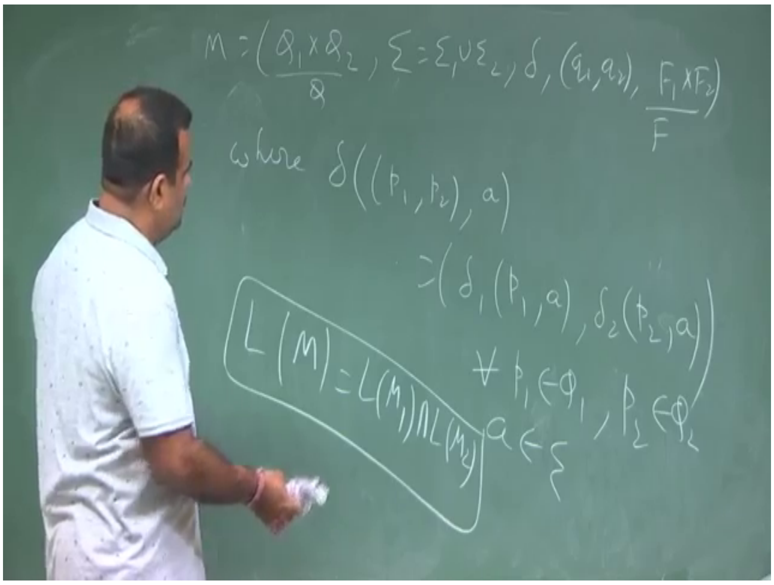
(Refer Slide Time: 06:49)



So, we choose a M which is Q 1 cross Q 2 and sigma which is sigma 1 union sigma 2, if sigma 1 and sigma 2 are same then it is just a sigma, then the delta we have to define when q 1 comma q 2 and it is F 1 cross F 2.

Now, how we define delta. So, it is taking the Cartesian product so, all states are in the pair wise. If p is a state in M 1 and q is a state in M 2 then, p comma q are state in M this is Q say. And now what is delta, delta means so, this is p comma q or p q we can say. So, we move so, this will be going to delta of p comma a and delta q comma a like this. Anyway this is going to define that yeah.

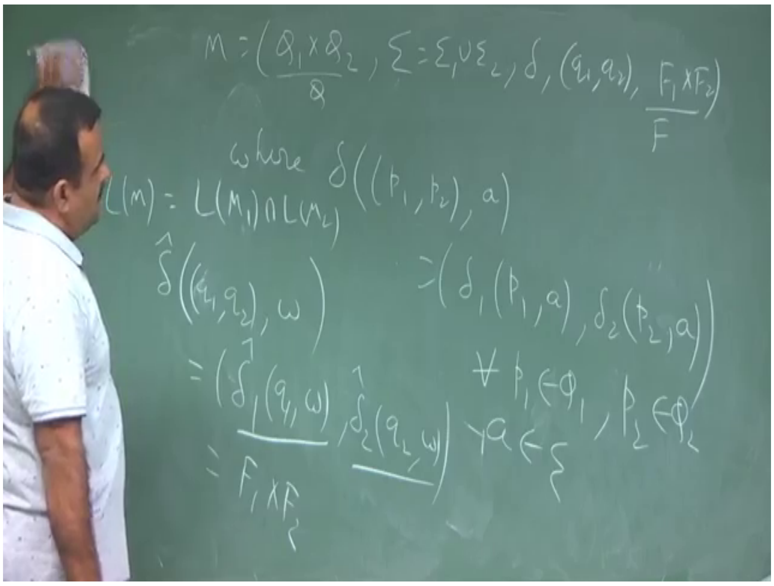
(Refer Slide Time: 08:19)



Where delta is delta of p 1, p 2 comma a is nothing, but delta 1 of p 1 a and delta 2 of p 2 a and for all p 1 p 2, all p 1 from Q 1 and p 2 from Q 2 and a is coming from sigma. I mean if we have assume the same sigma that is fine, no issue ok.

Now, for this we can easily verify that L of M is nothing but, L of M 1 intersection L of M 2, because we are taking the Cartesian product and our this is the final state is F, F is F 1 F 2. So, both the way it has to reach to the. So, a string which is so that means, yeah. So, this is the definition. Now, this we can prove it by the.

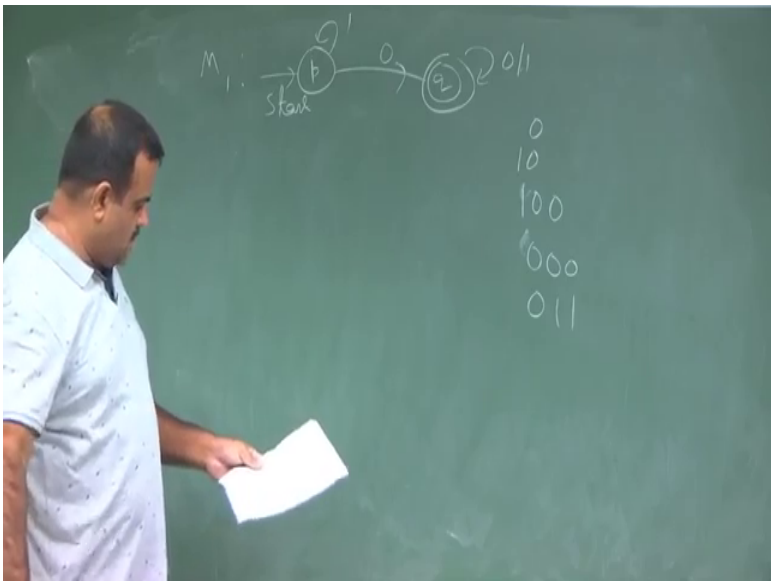
(Refer Slide Time: 09:51)



So, that means, delta hat of. So, this is q 1, q 2 comma w this is nothing but what? This is nothing but, delta hat 1 of q 1 and w and delta hat this we can easily prove 2 of q 2 w. Now this will be belongs to F 1 cross F 2 if an only this is belongs to F 1 and this is belongs to F 2.

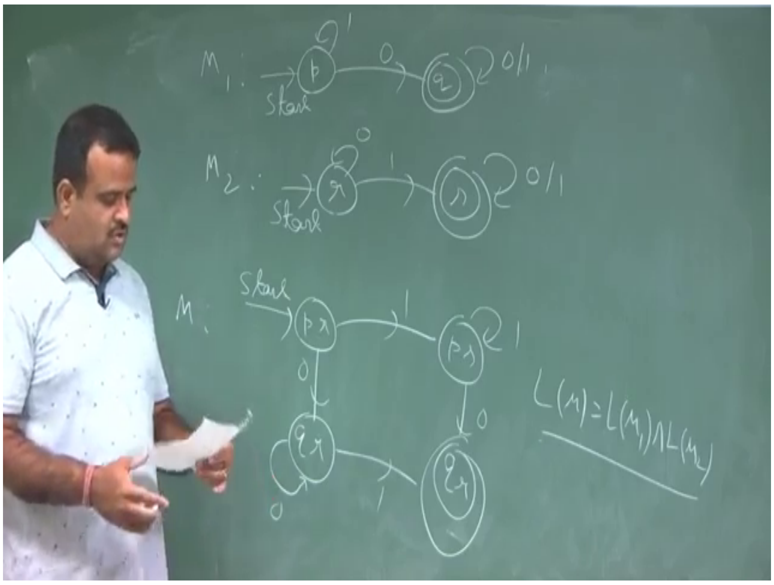
Now this is belongs to F 1, only if the w is coming from L of M 1 and this is belongs to F 2 because this is the string which is we are going to accept, this will be belongs to F 2; that means, if the string is accepted by this DFA, if and only if this is coming from F 2. So, that is the proof. So that means, and this is the L of M. So, L of M is basically L of M 1 intersection L of M 2. We can take an example then it will be more clear. Okay we can take 1 example.

(Refer Slide Time: 11:21)



Suppose we have M 1 so, this is p q is the final state, this is 0, 0 or 1 and this is 1, this is the starting state. So, what is the language accepting by this. This is the language which is which is having 1 0. So, if we have say 1 0 this is also accepting, 1 only 0 is accepting 1 0 0 is also accepting, 1 or 0 0 0 is accepting as 0 1 1 is accepting. So, like this.

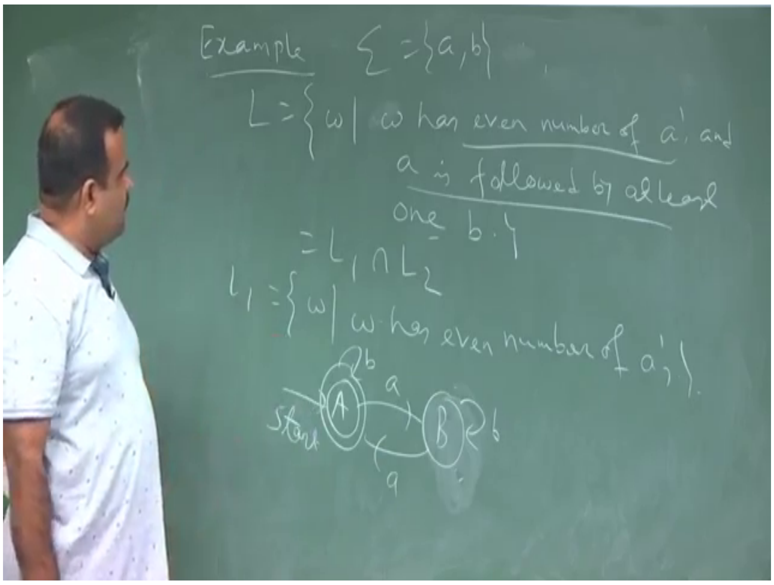
(Refer Slide Time: 12:29)



Now, M 2 is say r then it is say s. So, this is 0, this is 1, this is 0 or 1. Now we want to construct M. So, for M we need to have how many state? call the Cartesian product. So, p r, p s, q r, q s so, all the 4 so, this is our M. So, we can have p r, p s, q r, q s. Now which is the final state? Starting state, starting state is this one, this is Q 1 Q 2 kind of.

Now, this is the final state. Now what are the delta? So, with 1 we go here, with 0 we come here, then with 0 we come here, with 1 will go here and if that is the delta, delta rule because there delta of 1, delta p 1 is 1 and delta of r 1 is s, so that is going there we can easily. So, this is the language. So, if this is L of M is basically L of M 1 intersection L of M 2, because that is the only final state. So, to reach to the final state you have to so, it is either going to both the way we have to reach to the final state in M 1 and in M 2. So, you can take another example; we can take another example.

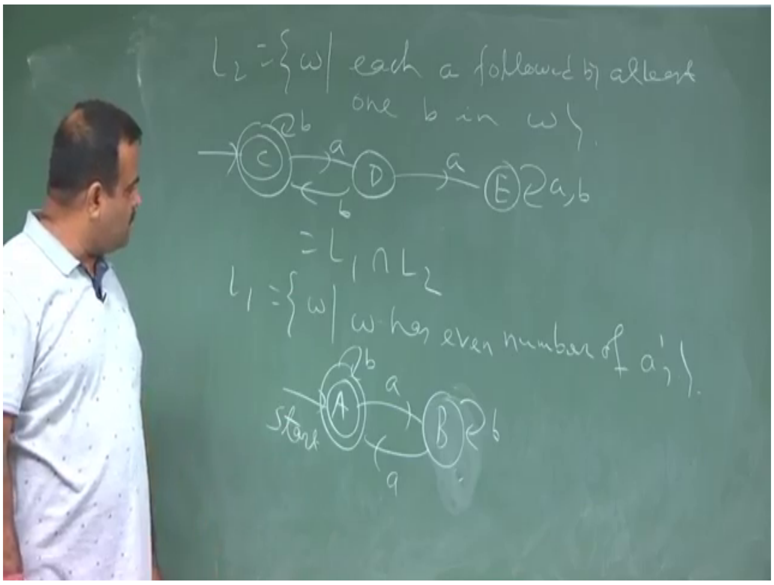
(Refer Slide Time: 14:57)



Suppose we have a, we want to construct a DFA language, which is w, w has even number of number of a's and a is followed by at least one b. So, our sigma is a, b. So, this we can split it into L 1 intersection L 2. So, what is L 1, L 1 is up to this part, even number of one's and L 2, L 2 is a is followed by at least one b and then we can find the intersection line. So, the L 1 is w where w has even number of a's.

So, what is the; so, this is a regular lang what is the DFA corresponding to this. We can start with A and this is the final state. Now, if we have a even number of a's. So, we can have or you can make this as a final state, sorry this is not a final state this is B. Now, we can go to a, then with a we can come back here and if you have a b will have here and with b we have here. This is this language. Now for this language L 2 is a set of followed by.

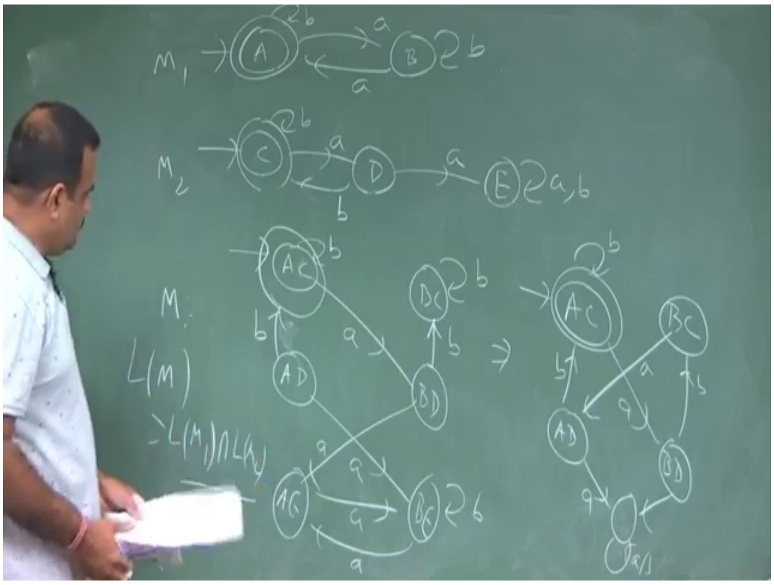
(Refer Slide Time: 17:15)



So, each a, so L 2 each a followed by a at least one b in w followed by at least one b in w ok. So, this is the language, this is also a regular language you can construct a DFA for this C D E. Now with a we go here, with b we remain here followed by b. So, with yeah with a we go here a b and this is our b. So, this is the language which is followed by at least one b ok.

Because if it is a a then we have to go to the dead state. So, now this L is so, now, we need to find a DFA which is accepting this intersection of this. So, you have to construct that ok. So, let us just.

(Refer Slide Time: 18:51)



So, this is our M 1, I will write M 1 in the top. So, this is b so, let me write this is our M 2 and M 1 is here, A is the starting state as well as final state and we have B over here, a even number of a and we have b over here. Okay now we want to combine these two by the construction which as we have seen the Cartesian product.

Now we have to find M by taking all the states as a Cartesian product. So, this is A, AC is one state, AD one state, AE one state. BC one state, BD one state, BE one state so will do that. So, we have all the states AC, AD and AE you can (Refer Time: 19:57) we need some space AC AD and AE and similarly we can have BC, BD, BE.

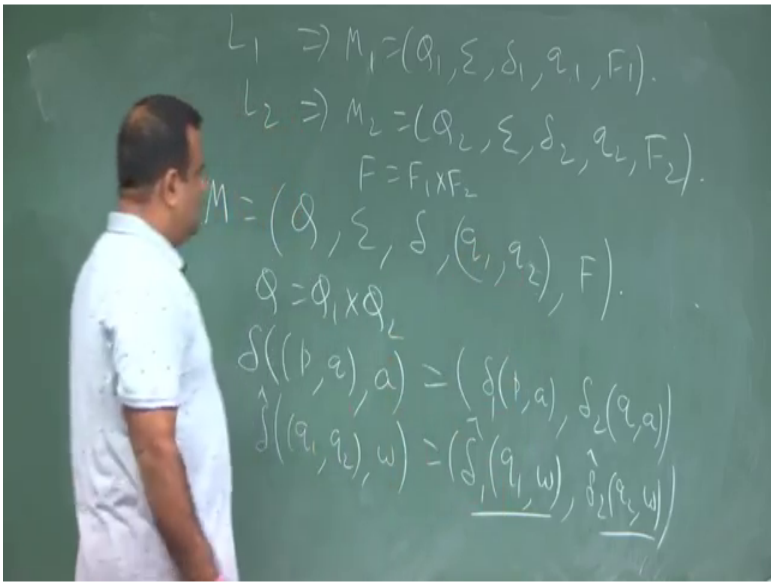
So, this is the starting state for this. Now with A AC. So, with A we are going to B in M 1 and we are C with A we are going to D so B D. So, with a we are going here. Now with b from AC we are going to a and so, with b we are going yeah. Now who are the final state? Final state is this one, because this is the final state for this, this is the final state for, this is the final state. Now similarly with AD with A we are going to B and D we are going to E.

So, from AD we are going to with a we are going here and with b from A we are going to A and from D we are going to E, D we are going to E sorry. So, AD so, A we are going to with b we are going to B and from D we are going to C. So, this is our b. So, similarly we can compute all this. So, this will be b, this is b, we can easily check this and this is from b BD it is going to be a and this is coming here a, this is coming here a we can verify this easily, this is b yeah.

So, we have all the moves, this we can simplify as this AC this is the final state, BC, then AD, BD, then we have another state over here. So this is with a we are going here, with b we are hoping here, is the starting state and with a we are coming here, with b yeah with a we are going here, with a we are going here this is the dead state we can say, like this and with b anyway. So, this is the combination of this.

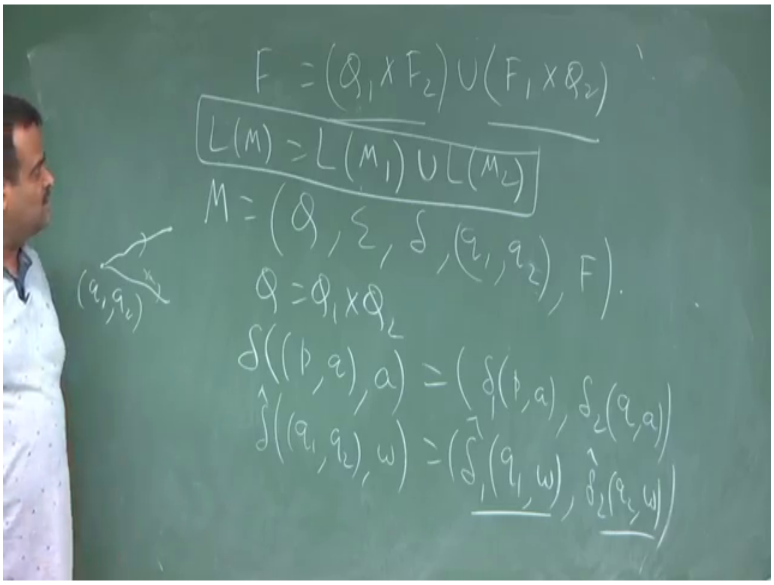
So, this is the way we can have. So, this is L of M; L of M is nothing but L of M 1 intersection L of M 2; now sorry intersection. Now for the union also we can construct a, union we know it is straightforward coming from the regular expression, but we can have a construction for this thing, for the DFA construction for union also we can have.

(Refer Slide Time: 23:45)



So, suppose you have L 1 which is regular you have M 1, sigma keeping the same q 1, F 1 and we have L 2, Q 2, delta 2, F 2. Now, M Q, sigma, delta, q 1 comma q 2 and F. Now Q is nothing but Q 1 Cartesian product of Q 2 and delta is nothing but p q a which is nothing but delta of delta 1 of p a, delta 2 of q a. So, from here we can extend this delta hat, delta hat of q 1 q 2 w is nothing but delta 1 hat of q 1 w comma delta 2 hat of q 2 w. Now, if we make this F to be a F 1 cross F 2 then, this both away we should have a reach to the final state, then it will give us the intersection, but if we choose F to be like this then will get that union.

(Refer Slide Time: 25:37)



Now, if we choose the F to be Q 1 cross F 2 union F 1 cross Q 2 sorry this is our F. Then at least if one way if we can go to the final state one of this then, we are done. Then in that case L of M is nothing but L of M 1 intersection L of M 2. So that means, we start from p q we start from q 1 q 2 this is the starting state for the, then we are going with x in the in M 1 if we.

So, earlier case in both the x in M 1 and M 2 we need to reach to the final state both the case, but here at least one of these branch is going to the final state of any one of this automata then we are done. So, that is the construction for union. So, we will see the more properties on the regular set.

Thank you.