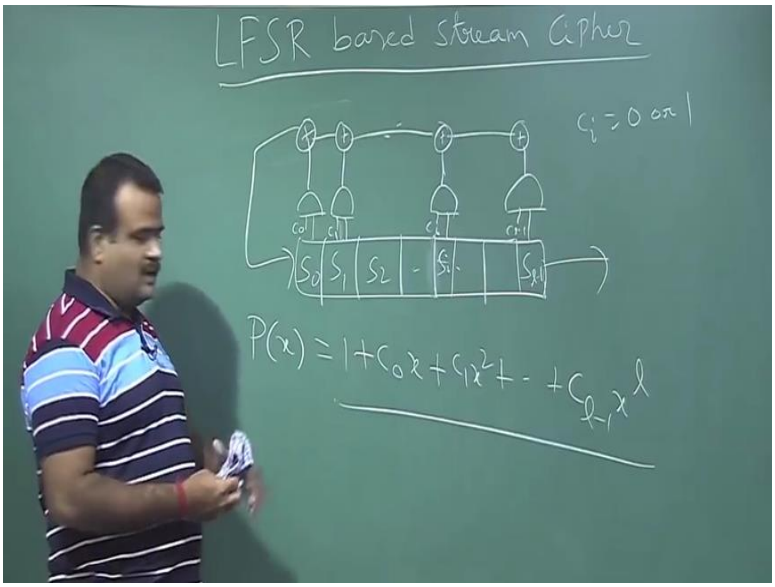
**Internetwork Security**
**Prof. Sourav Mukhopadhyay**
**Department of Mathematics**
**Indian Institute of Technology, Kharagpur**

**Lecture - 11**
**LFSR based Stream Cipher**

We will talk about LFSR based stream cipher. So, we have seen the LFSR as a key stream generator or pseudorandom bit generator.

(Refer Slide Time: 00:29)

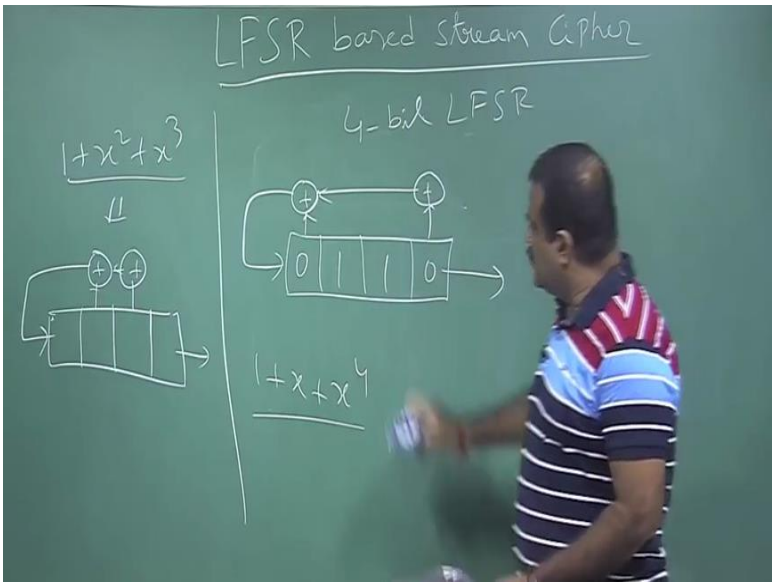



Let us start with a say L bit LFSR, how we define a L bit LFSR. So, it is basically L bit register we have L bit flip flop, so L bits, so S0, S1, S2, like this so dot, dot, dot, dot, S l minus 1, S l minus 1. So, this is we have l; l bit register, now each of these we have a l gate along with some values C 0 and this is taking this, so this is C 1 like this. So, this if this is a S i then we have a S i going and we have C i, this Ci's are either 0 or 1 like this. So, this is basically C l minus 1. So, these are all XOR dot, dot, dot, XOR with this and this is coming out.

This is a typical l bit LFSR. So, these are l bit flip flops, l bit register and these are these we called state; this is the first 0, state 1 state like this. So, this is a state. So, now, this Cis are either 0 or 1, if the C i is 0 then whatever the value over here, it will not contribute there so that corresponding bit is not contributing in the feedback, if the C i is

1 then it will contribute it in the feedback and this will corresponding to a polynomial, p x is equal to 1 plus C 0 x plus C 1 x square up to C l minus 1 x to the power l. So, this is a l degree polynomial. Depending on the value of C i's we have this polynomial. So, then let us take an example of 4 bit LFSR.

(Refer Slide Time: 02:52)



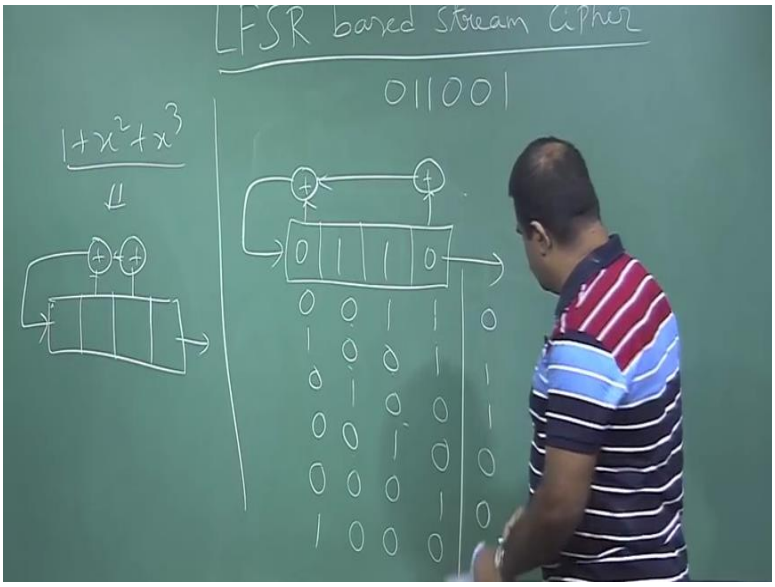

For 4 bit, we have 4; 4 bit register 1 2 3 4 and we have connections there. So, say we have a connection over here and we have a connection over here. So, this is 1 LFSR is going out, this is 1 LFSR. So, this LFSR is corresponding to the polynomial 1 plus x. So, this corresponding C 0 is 1, C 1 is 0, C 2 is 0, C 3 is again 1 because these are the bit contributing in the feedback. So, linear feedback shift register so, this is corresponding to this polynomial. So, if you have this polynomial this is basically disconnection.

Now, if we have to say this formula 1 plus x square plus x cube. So, these will corresponding to with LFSR, this is again 4 bit. So, this is basically x is not there, so x square x cube, so, these 2 and it is, so this is the LFSR corresponding to this polynomial. So, given a polynomial, we can have a LFSR and given a LFSR one can have this polynomial.

Suppose we take this polynomial, this polynomial is as a particular character, it a characteristic, this is the elusive polynomial not only that it is a primitive polynomial, we will talk about primitive polynomial, let me discuss mathematical background of this course. So, if you take this LFSR and if you start with any initial vector so, actually we

initialize this state by the secret keys shared between Alice and Bob. So, if you start with any nonzero initial vector, initial key, so suppose here, so this is the secret keys shared between Alice and Bob and we want to run this LFSR and here why we want to get the key stream.

(Refer Slide Time: 05:09)



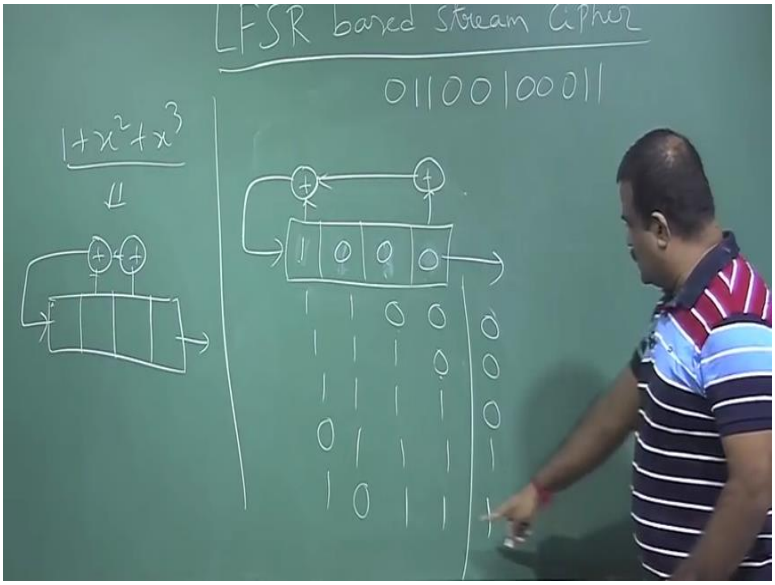

If you do so then what value will be getting? So in the first run, this 0 will come here, 1 will come here, this 1 will come here, this 0 will come here and this bit will be replaced by XOR with this and this which is basically 0.

Now this is the current state of this LFSR. We will copy this value over here and then again we will so each clock, we are omitting one bit last bit, so feedback and we are taking the feedback to fill up this bit. So, now, next this will be output and this is come here this will come here, this will come here, again this will be these and these XOR. So, this is 1. So, if we continue like this, in the next clocking this will come here this will come here, this will come here, this will come here and x 1 will be these are because this is the current state of the LFSR. So, these will this XOR bitwise XOR this is 0.

In the next 0, we will come here, this will come here, this will come here, this 0 will come here, we XOR this and this. So, this is the key stream bit actually so, this on this 0 again so then this is the current state. So, these will come here and these will come here. So, linear feedback shift register so we are just shifting the by 1 bit and we are taking the feedback based on this connection based on the polynomial.

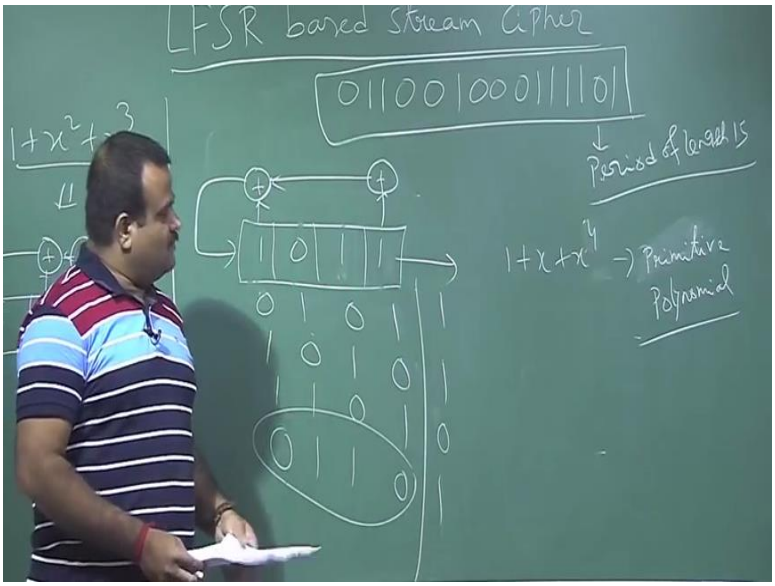
This will be coming here, this will be coming here, these 2 used basically 0 then we have 1 is going there and we have here, this will come here, this will come here, this will come here and these 2 is 1. So, this is the, so up to this we will got, we got like this 11001 up to this, now we copy this here because we do not have space yet so, we copied this value over here and we will continue. 1000, so, this is the current state.

(Refer Slide Time: 07:27)




Now we can run it in the next clocking, this 0 will come here and this 0 will come here, this 0 will come here, this 1 will come here and these and this XOR, it will give us 1. So, this is makes key stream will be 0 and then we have again, we take this out 0011 and this will be 1 again and we continue like this. So, this is again 0. So, now, this is 0 will come here, 111 and this will be again this is the current state, this will be again 1. So, now, we have 1111 and now these are all 1s, so this is the key stream. So, this will be 0 again and then this 111 and this 0 will come here and this 2 will give us a 1 again and so, this will basically another 011, now this is the current state. So, you just copy here. 1011, again we will run this to get the next key stream.

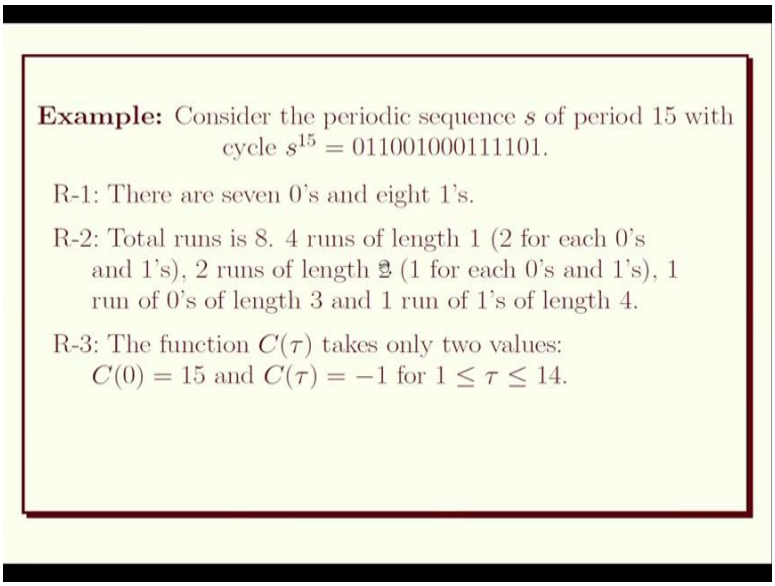
(Refer Slide Time: 08:52)




This 1 will come here, this 1 will come here, this 0, this 1 will come here, we XOR this to this will give us a 0 and then 1 will come here, this 0 will come here, 1 will come here, 0 will come here, these 2 will give us a 1 and then again 0 will be the output then this 101, these 2 will give us 1 and then we have again this 1 will go here, this 0 will come here, 1 will come here, this 1 will come here and this is 0. So, this is the initial state it is coming back. So, this is a full sequence. So, this is 1101. So, this is a period, this is a full period, this is the period of length 15.

After that it will just generate the same key stream because we are coming back to the original; the initial internal state because we started with this state, this was the key again if we continue will get the same key stream. So, this is 1 period of this key stream generator pseudorandom bit generator.

(Refer Slide Time: 10:26)

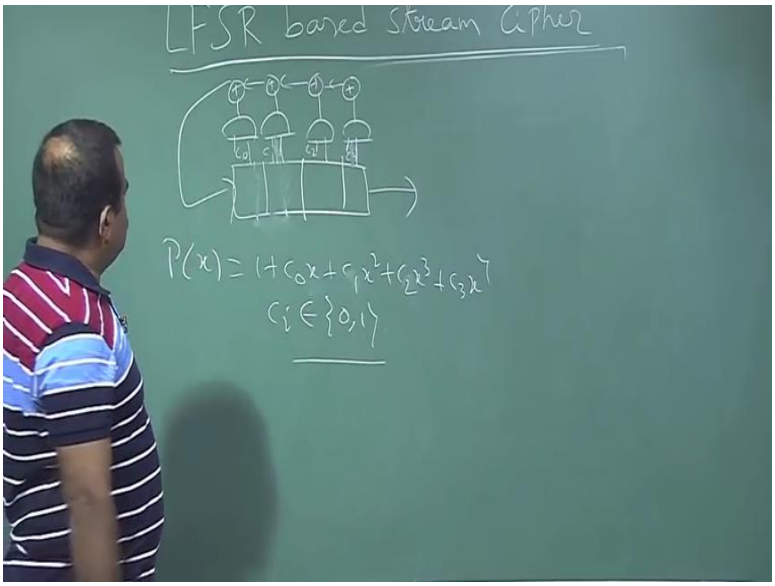



And this sequence we have seen in the, if you come to this slide, so this sequence, we have seen is satisfying, this is the same sequence here and we have seen that this sequence is satisfying all the properties of the Golomb's postulates. So, this sequence is a good random sequence in the sense of it is satisfying all the test of the Golomb postulates and even we can try for other few tests and it may pass.

So, the conclusion is so LFSR is giving a good sequence. So, LFSR is that good pseudo random bit generator provided, we choose the polynomial to be a good polynomial. So, here polynomial is basically this, 4, so this is we discussed when you talk about mathematics of this course this is what is called primitive polynomial primitive polynomial. So, we talked about these in details what is primitive polynomial. So, if we choose a primitive polynomial then those corresponding ellipses are will give us a good sequence of pseudorandom bit generator, so it is good.

Now, we will talk about so this is the stream cipher, this is the LFSR bit stream cipher, basically it is generate the key stream. Now if we just consider this, as a stream cipher key stream generation and if we have this connection like this. So, you want to see how good this key stream is. So, we know this is randomness wise it is good, but is there any other linear; it is this key stream is linear or non-linear that we want to talk about.

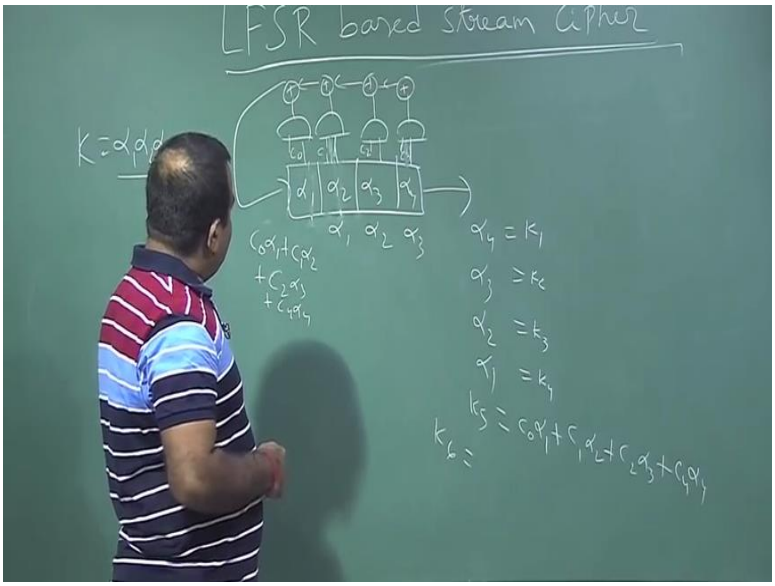
(Refer Slide Time: 12:27)




Suppose you have a 4 bit LFSR, we have a 4 bit LFSR and suppose we have this is C 0 and this is say C 1 and this is C 2, C 3 and we have this XOR and this is coming out, this is going out and this is 4 bit LFSR, but C i are either 0 or 1. So, the corresponding polynomial is basically C 0 plus C 1 plus; 1 plus C 0 x plus C 2 x square plus C 3 x cube sorry C 0 plus C 1 x plus C 2 x square C 3 x to power 4 C 1, C 2 are basically 0 or 1. So, it depending whether that corresponding bits is participating in the feedback or not. If it is participating in the feedback that corresponding C i is S i is participating in the feedback then the corresponding C i is 1 otherwise that C i value is 0.

Now suppose so this is the polynomial.

(Refer Slide Time: 14:06)



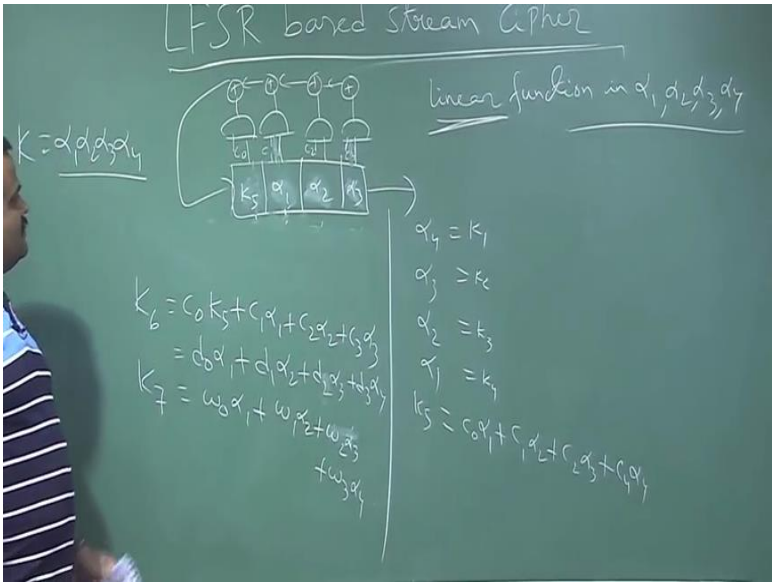

Now, suppose this is the key, key is alpha 1 alpha 2 alpha 3 alpha 4, 4 bit key which is shared between Alice and Bob. So, how we are getting key stream from this LFSR? So, we are first loading this alpha 1, alpha 2, alpha 4 as a initial value of this state, this is we can consider as the state and each time the state is changing. So, this could be the initial value of the state. So, now, if we just so, key is our secret key is basically alpha 1, alpha 2, alpha 3, alpha 4 which is shared between Alice and Bob.

Now, what will be the key stream? Key stream is first key stream bit is alpha 4 will be coming out then so this will be basically alpha 1 will come here, alpha 2 will be coming here, alpha 3 will be come here and this will be basically summation of; so this is basically C 0 summation of C 0. So, C 0 alpha summation of so this is C 0 alpha 1 plus C 1 alpha 2 plus C 2 alpha 3 plus C 4 alpha 4 like this, but the next key stream will be alpha 2, alpha 3, alpha 2, alpha 1. So, all the key bit will be first key stream basically and then next one, the alpha 5 maybe basically this is k 1, k 2 these are the key stream k 3, k 4. So, k up to k 4, we are getting basically the key, the secret key then after that what about k 5, k 5 is basically this one which is having this linear form C 0 alpha 1 plus C1 alpha 2 plus C 2 alpha 3 plus C 4 alpha 4. So, this is basically our k 5.

Now, how to get k 6? K 6 will be so, this is basically for k 6 we have basically what it will be again. So, this will be so now, for k 6, what is the status? So, the status is we have

a, so k 5 will come here. So, next one will be, so this is k 5 will come here and still this is then if k 5 is here then this is say alpha 1 and this is alpha 2.

(Refer Slide Time: 17:00)

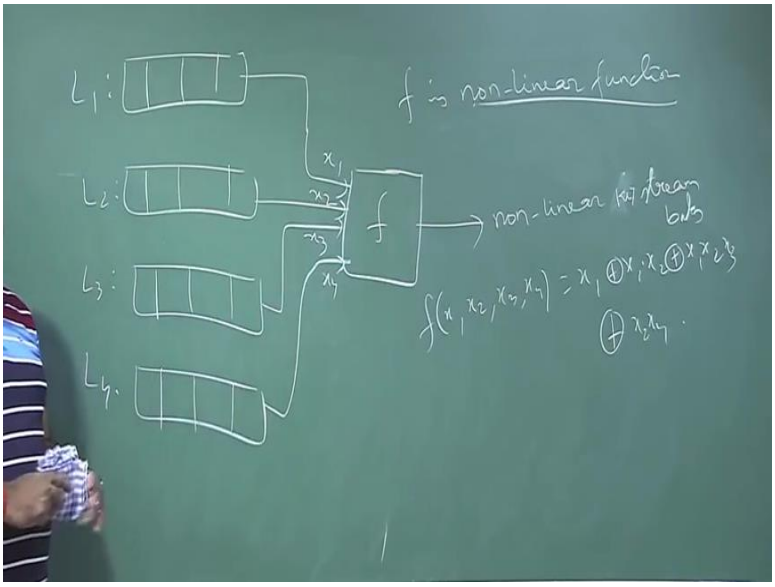



Then also then k 5 is here only, then what is k 6 then? K 6 is basically sorry, k 5 will be here and this is alpha 1 alpha 2 alpha 3 only k 4 is gone. So, what is k 6 then? K 6 will be C 0 k 5 plus C 1 alpha 1 plus C 2 alpha 2 plus C 3; C 0 alpha k 5 C 1 alpha 1 C 2 alpha 2 plus C 3 alpha 3, this is the k 5. Now if you convert this into this, if you use this formula. So, it will be again something like some d 0 alpha 1 plus d 1 alpha 2 plus d 3 alpha 3 sorry; d 2 alpha 3 plus d 3 alpha 4, 4 will be there, 4 will be here, 4 is coming from this 1. This is again a linear function in alphas. So, these are just a constant these are these d i are combination of this C i's basically. So, this is a linear function.

So, again k 7 also will be written as some w 0 alpha 1 plus w 1 alpha 2 plus w 2 alpha 3 plus w 3 alpha 4 like this. So, any key stream is basically any key in that key stream is basically linear combination of linear function in the secret key streams; secret key. So, this is a linear function. So, this maybe dangerous in a sense of we will talk about algebraic attack later on. So, we can have some equation linear equation then one can try to solve that equation and can guess what is the alphas because if we know some of the key stream. So, this is algebraic attack, I will talk details about that. So, this linearity is having some problem. So, this is a linear function. So, LFSR is that is why it is called linear feedback register.

It is a linear function in alpha 1, alpha 2, alpha 3, alpha 4. So, this is this has some problem with the say with the attacks like algebraic attack. So, we will talk about those later on. So, now, question is how we can make it non-linear. So, we want to use the LFSR, we want to use LFSR, but we want to make it a non-linear key stream generator. So, this is a linear key stream generator. So, if you just use the LFSR, so LFSR is a linear key stream generator. So, we want to make it non-linear key stream generator. So, idea is to have a non-linear function.

(Refer Slide Time: 20:22)



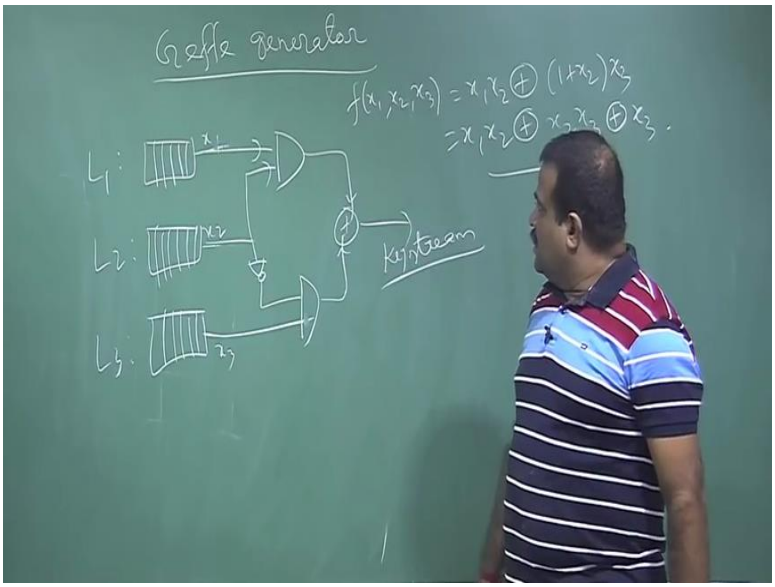

Suppose we have a say a LFSR, so, this is 1 4 bit LFSR L 1, we have another 4 bit LFSR L 2 like this, suppose we have is 1 example, how we can use LFSR to make it non-linear function, non-linear key stream generated. So, we suppose, we have 4 bit 4 LFSR. So, each is coming from a polynomial. So, once you have a polynomial, basically you have a 4 polynomial and corresponding to each polynomial you have a LFSR.

Now, what we do? We just take the output of this LFSR and we have a function which is a non-linear function. So, say this output say x 1 and this out to say x 2, this output say x 3 and this output say x 4. It could be a instead of 4, it could be 5 6 n in general, this is say n LFSR of L bit. So, each is giving output key stream. So, this is this is linear, but this is also each of output of this LFSR is giving linear, but here f, we use as a non-linear function. So, f is a non-linear function, f is basically non-linear function. So, this is basically non-linear stream bit.

So, f could be say if it is 4 bit x 1, x 2, x 3, x 4, it could be x 1 XOR with x 1 into x 2 XOR with x 1, x 2, x 3 any non-linear function like this x 2 x 4, any non-linear function, we can abuse to make it a long linear key stream bit generator. So, this could be extended up to n LFSR. So, and each could be a L bit. So, we have basically L degree n polynomial and each is giving us a 1 bit and this f is the taking this n L bit and giving output bit and which is a non-linear function.

We can take an example of such LFSR which is basically Geffe generator. So, it is basically based on 3 LFSR.

(Refer Slide Time: 23:09)




This is one example of non-linear key stream generator based on the LFSR, we can use the LFSR, but we have to apply a non-linear function f. So, here the non-linear function is basically here you have having 3 LFSR L 1, L 2, L 3 of any bits. So, this is giving us x 1, this is x 2. So, this f is a non-linear function, this is x 3. So, this is non-linear LFSR. So, this is this going to L bit. So, what we do how we get initialize this? We can initialize this by the secret key which is shared between Alice and Bob. So, we just put the secret key as the initial value of this state and then we will run this LFSR and once we run the LFSR we will get output this is x 1 and this is the x 2 output from this x 2 and this is x 3.

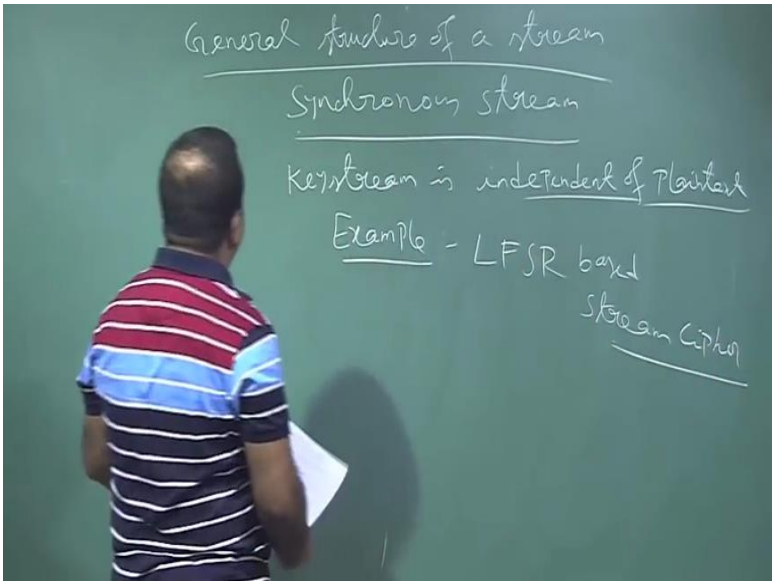
This function for Geffe generator is basically of this form, x 1, x 2, this is the end then x 1 plus x 2 into x 3. So, this is basically x 1, x 2 then x 2, x 3, x 4, x 3. So, this is a non-linear function. So, this will generate the non-linear key stream bit which is not linear.

So, we can just draw this in a logical way. So, this is x 1. So, x 1 is ending with x 2. So, this is x2, this is x 1, this is x 2, x 1 is ending with x 2 and then this x 2 is also having a negation. So, and this we are ending with x 3, this we are ending with x 3 and these 2 will we are just doing the x 1, beta is x 1 and it is giving us the key stream.

This key stream is a non-linear key stream because we are using the non-linear function f like this. So, this is a one example where we can use the LFSR, a LFSR is basically a linear key stream generator, but we can use LFSR as many as we want and we just initialize this internal state as the secret key shared between Alice and Bob and then each which of this LFSR, we just initialized by that key and then we run this an each time it is giving output that x 1, x 2, x3 and then we take a non-linear function f to make it non-linear key stream generator. So, this is the example or we can use LFSR as a non-linear LFSR to generate a non-linear key stream although LFSR is a linear.

Now, we talk about the general form of the stream cipher.

(Refer Slide Time: 26:36)

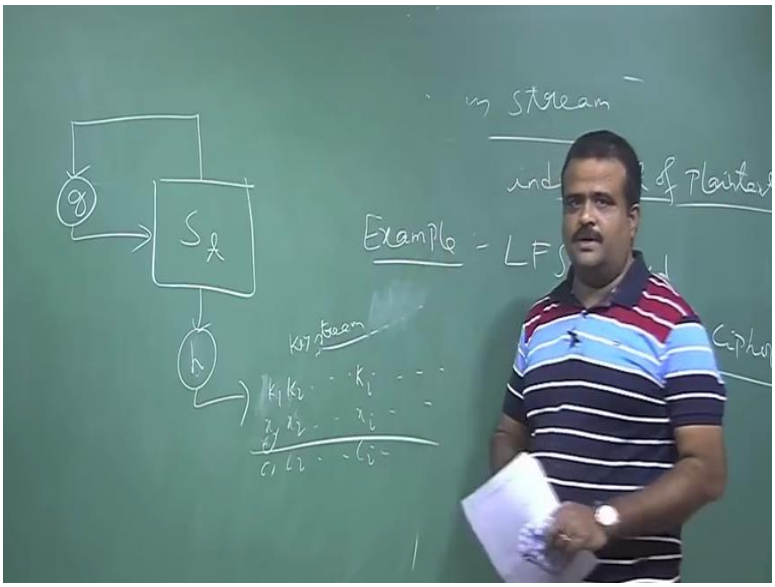



General structure of a stream cipher in particular a synchronized stream cipher it is called synchronized stream cipher, synchronous stream cipher. Synchronous stream cipher means the key stream will only depend on the secret keys shared between Alice and bob. So, key stream will not depend on the plaintext or cipher text. So, we are not getting a, we are not taking any feedback of the plaintext or the cipher text. So, it is independent of the plaintext and cipher text. So, that stream cipher is called synchronized stream cipher.

Another type of stream cipher is asynchronous stream cipher where will give the input another input as a plaintext or the cipher text bit.

For synchronized the key stream is independent of plaintext and the cipher text. So, we are not having any input of the plaintext or cipher text in the key stream. So, the stream cipher we have seen LFSR based stream cipher all are synchronized stream cipher because it is just taking the original key and it is giving us the key stream. So, one example is example of a synchronized stream cipher is LFSR of a stream cipher we have seen. So, now, we can write a general structure of a synchronized stream cipher.

(Refer Slide Time: 28:34)




So, it has basically having a state which we denote by S t for example, if it is LFSR then if it is 4 bit, so this state is basically 4 bit register and which is updating by a state update function the state is every time updating. Like for LFSR we have linear feedback shift registration linear feedback shifting, that is our g. So, this way we are updating this state and this state is initialized by the secret key shared between Alice and Bob.

And then we are having another function h which is taking the content of the state and which is giving us the key stream. So, this is k 1, k 2 like this k i. So, this is the stream, this is the pseudo random bit generator key stream; key stream. So, this is and then we XOR with this with the plaintext bit to get the cipher text bit C 1, C 2, C i so far. So, this is the at t-th time this is the state. So, which is initialized by the secret key and every time we are updating the state by the g function and every time we are taking the content of

the state and we are applying this function h, we are getting the key stream that i-th key stream so like this. So, LFSR is one example of the synchronized stream cipher. So, this is the general form advanced construction of a synchronized stream cipher.

Thank you.