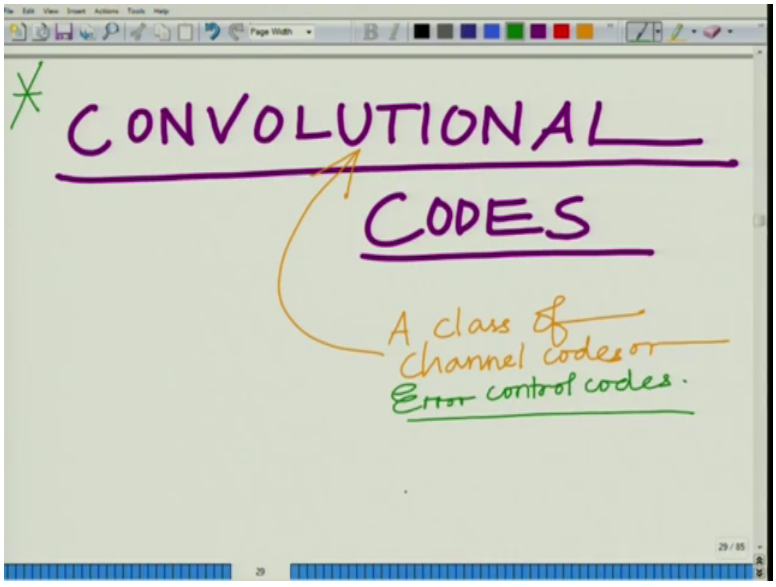
Principles of Communication Systems - Part II

Prof. Aditya K. Jagannatham
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture - 49
Introduction to Convolutional Code, Binary Field Arithmetic, Linear Codes

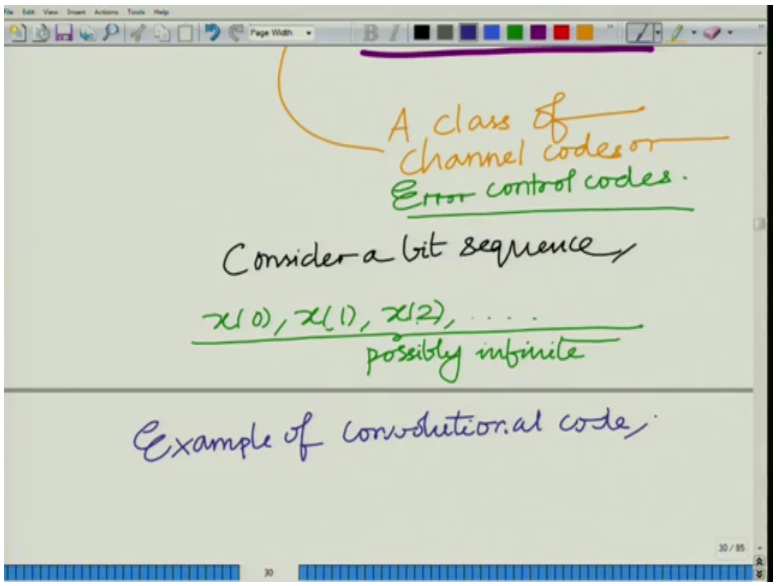
Hello welcome to another module in this massive open online course. So, we are looking at channel coding, and in this module we start looking at a different class of channel codes termed as convolutional codes ok.

(Refer Slide Time: 00:26)



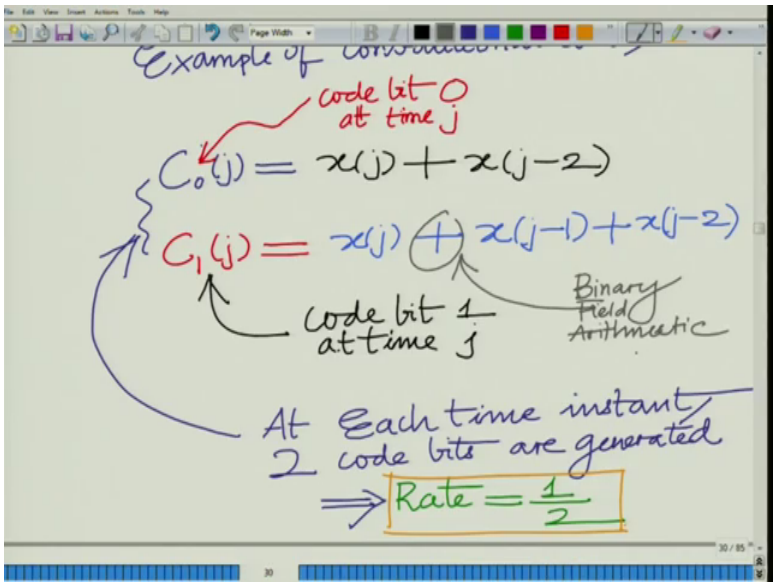
So, we want to look at convolutional codes which are basically a class of channel codes and we have seen the channel code is basically nothing, but a paradigm which introduces systematical redundancy to recover from errors introduced by the channel. This is a class of channel codes or which are also known as error control codes or error control codes ok.

(Refer Slide Time: 01:31)



Now, consider a an information bit sequence; consider a bit sequence and this can be possibly infinite. Now an example of a convolutional code can be defined as follows example of a convolutional is given as follows.

(Refer Slide Time: 02:31)

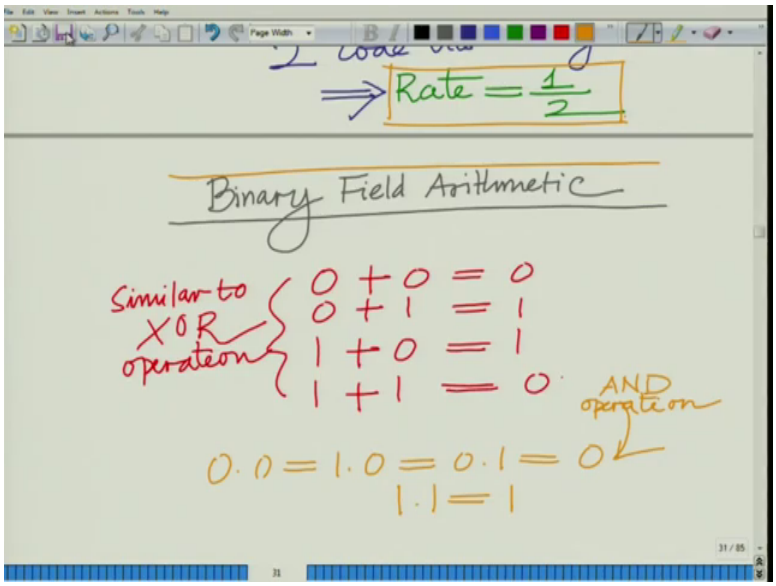


We have C 0 j this is equal to x j plus x j minus 2. So, this is code bit 0. So, x j s are; obviously, the information bits at time instant time j time instance j minus 2, this is the code bit 0 at time j, code bit 1 at time j equals x j, x j minus 1, plus x j minus 2 and this is your code bit 1 code bit 1 at time j ok.

Now, what is the meaning of this? So, remember there is for each at each time instance we are generating that is for each information bit we are generated. So, at each time instance first observe at each time instance that is a d time instance for a single information bit x j or at each time instant at each time instant or for each information bit for each time instant 2 code bits are generated. C 0 j implies now obviously, you know i am getting at the rate, remember the rate is number of information bits divided by number of code bits this is equal to half, because for every information bit you are generating 2 code bit. So, this simple code has rate equal to half this is a rate half code, for every information bit you are generating **2** code bits. So, redundancy is half remember we said as the rate as the redundancy increases the rate decreases. So, this has a redundancy of half ok.

Now, further this addition if you look at this addition this is the important this is an important point, this is addition on the binary field this is something new, this is known as binary field arithmetic.

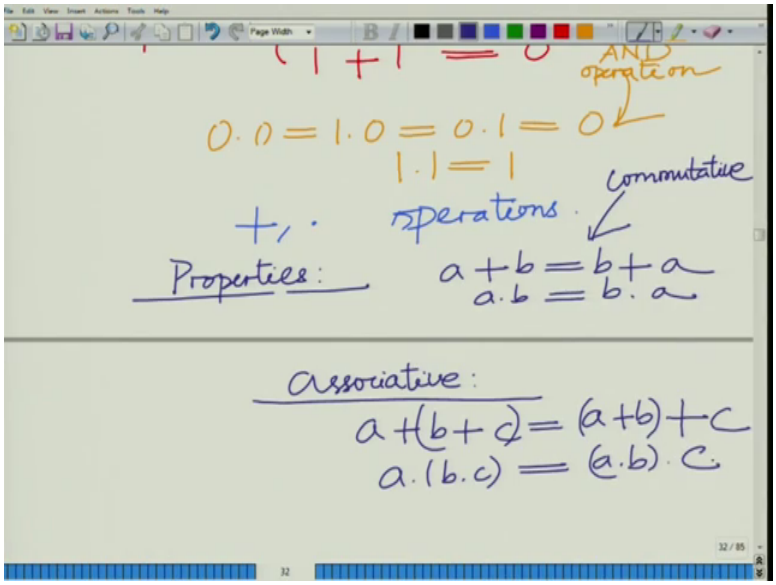
(Refer Slide Time: 05:26)



binary field arithmetic means that is if you look at binary field arithmetic is basically in addition is basically 0 plus 0 equals 0, 0 plus 1 equals to 1, 1 plus 0 and this is a bit wise operation 1 plus 0 equal to 1, 1 plus 1 equal to 0, this is an XOR. So, you can see this is similar to an XOR addition over the biology field is similar to an XOR operation and multiplication 0 into 0 equals 1 into 0 equal 0 into 1 equal 0 1 into 1 equals 1. So, this is

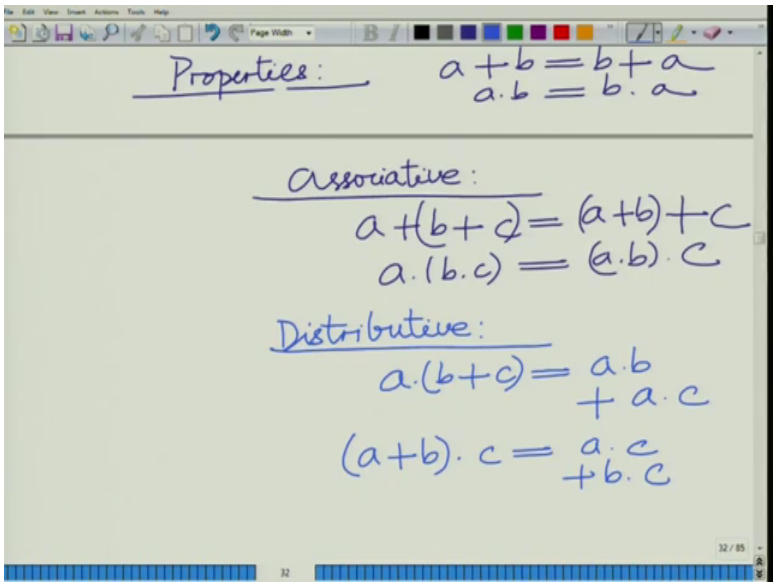
similar to an and operation. So, this is basically your binary field arithmetic; this is a bit wise operation binary field addition which is similar to the XOR operation and multiplication which is similar to the and operation. And you can see the satisfies several properties will not go into the details of that, but you can see the property several properties are satisfied by this plus and dot.

(Refer Slide Time: 06:53)



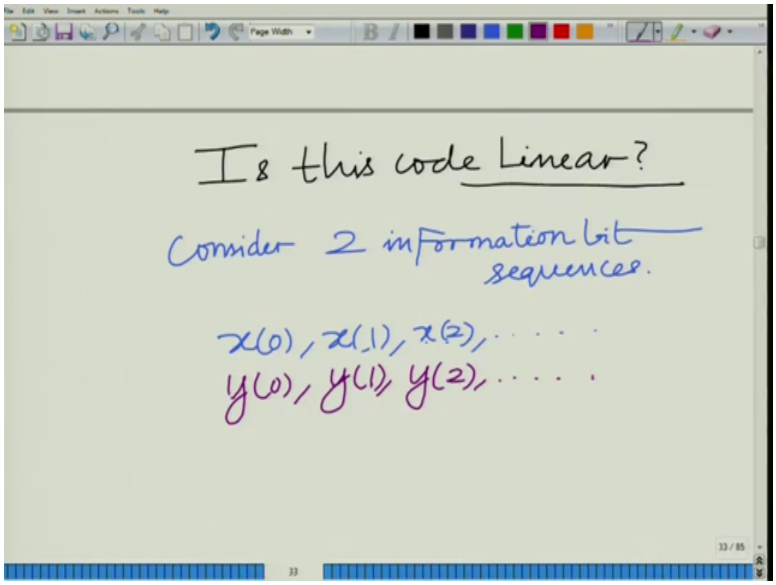
Operations if you look at this properties, we have commutativity these are termed as commutative that is it commutes that is a plus b equals b plus a, a dot b equals d dot a then its associative both the operations are associative, that is a plus b plus c equals b plus c or equals a plus b plus c, a dot b dot c it was a dot b dot c further distributive.

(Refer Slide Time: 08:04)



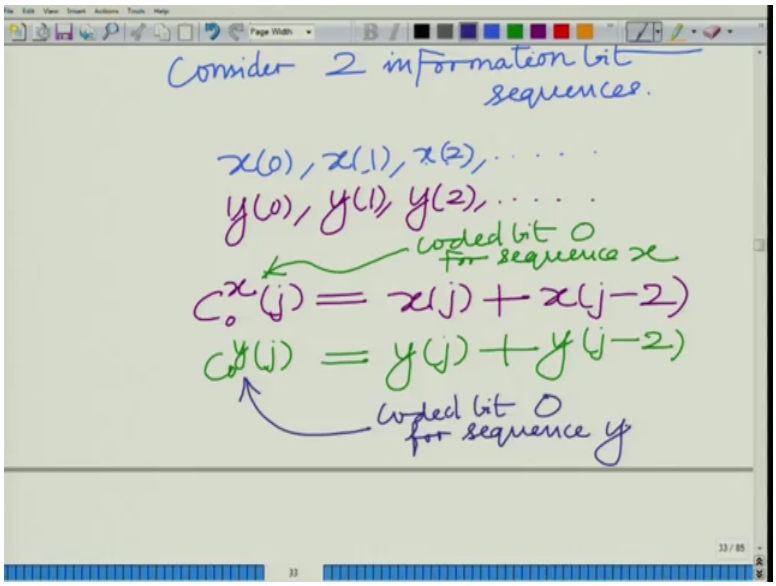
You can see that these operations are also distributive a dot b plus c equals a dot b plus a dot c at. So, on or a plus b dot c as a dot c plus b dot c and so on now. So, this is the binary field arithmetic which were going to use exhaustively, that is as we deal with convolutional codes we are going to use this binary field arithmetic exhaustively, all right. So, it is important to familiarize yourself with this binary field arithmetic that is addition is the x or operation x or operation and multiplication is the and do obey the series of properties which is commutativity that is order is not important, that is a plus b is b plus a, a dot b is b dot a associativity and also the distributive property ok.

(Refer Slide Time: 09:08)



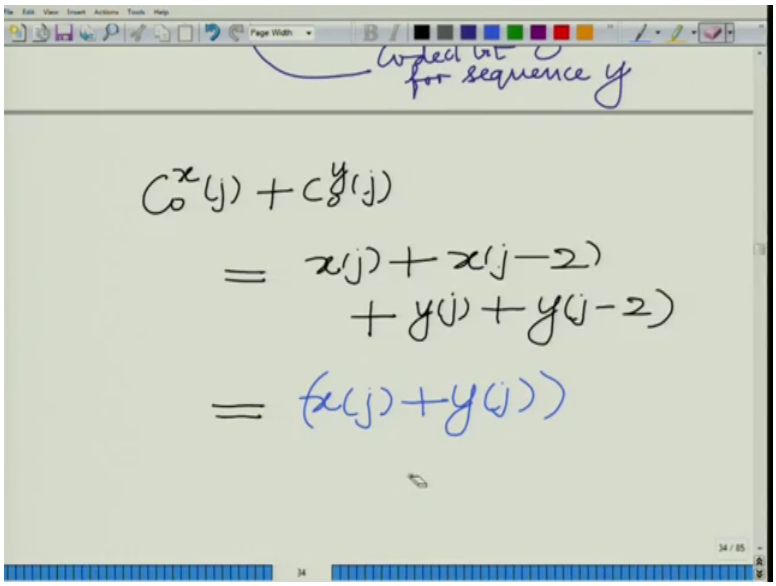
Now, 1 thing which is important is you want to see is this code you want ask the question is this code a linear code. For instants let us considered 2 information bit sequences x 0 x 1 up 2 x 0 x 1 x 2 so on. This is a first information bit sequence another information bit sequence is y 0, y 1, y 2 ok.

(Refer Slide Time: 10:08)



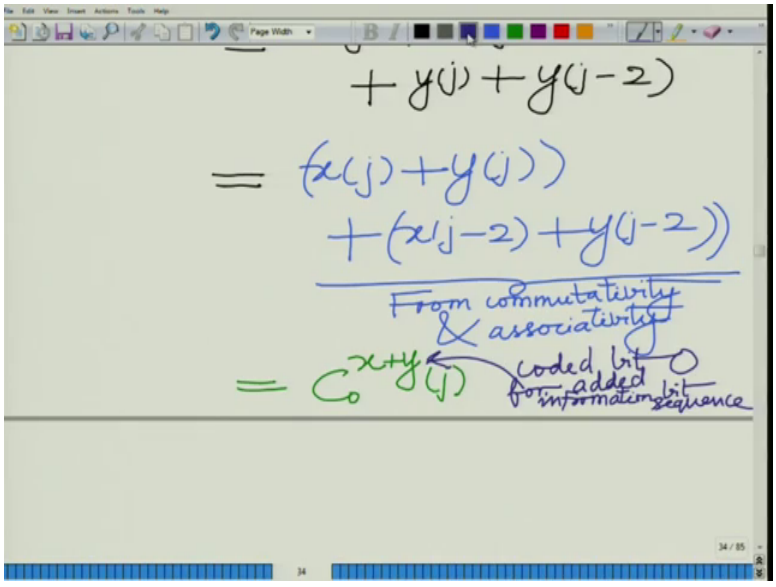
So, these are the 2 information bit sequences now the code bit 0 due x j, code bit 0 at time j due to information bit sequence x is x j plus x j 1 minus 2 this is a coded bit corresponding to for sequence x. Similarly the coded bit y coded bit 0 for sequence y will be y j plus y j minus 2 now if you look at. So, this is the coded bit for coded bit 0 for sequence y.

(Refer Slide Time: 11:05)



Now, you can see what happens when you add these 2 coded bits C 0 of x, C 0 of y of j that will be x j plus x j minus 2, plus well y j plus y j minus 2 which is equal to using commutativity and associativity you can prove that x j plus y j plus x j minus 2 plus y j minus 2.

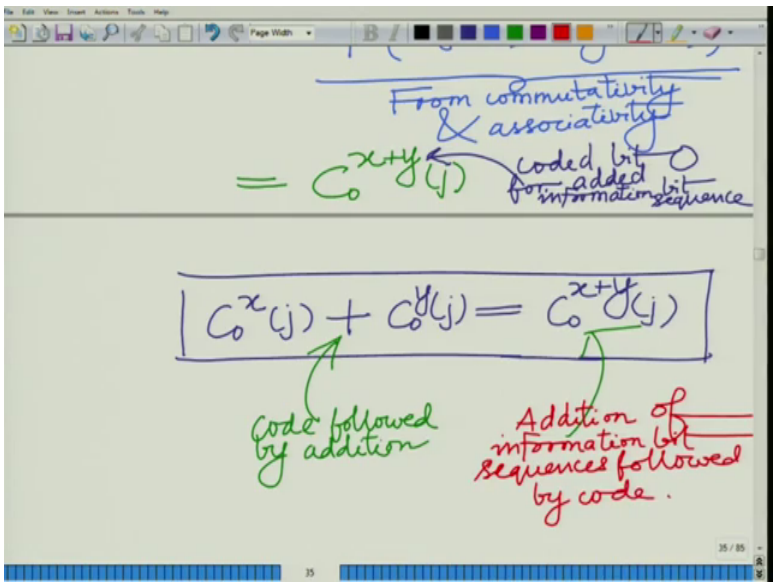
(Refer Slide Time: 11:50)



Which is this is arising from commutativity and associativity; and now if you see this is nothing, but corresponds to the coded bit generated by the addition of the information bit sequences x and y. So, this is nothing, but the coded bit 0 correct if you add the

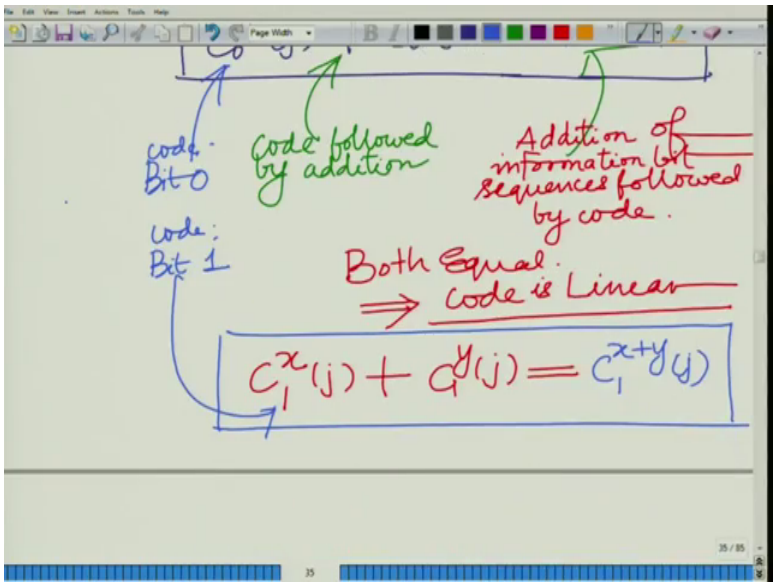
information bit sequences x and y, what you will have is x the j th bit will be x j plus y j the j minus 2 bit will be x j minus to plus y j. So, this is this corresponds to a coded bit generated by the addition. So, by adding the code sequences right. So, there are 2 different things one is you are adding the code sequences corresponding to the 2 different information bit sequences, the other is the coded sequence generated by the addition of the bit sequences. So, you can generate the code sequences and add them or you can add the bit sequences and generate the code both this things are similar. This is this implies that this code is linear code that is the important point.

(Refer Slide Time: 13:45)



So, here this is the coded bit for coded bit for additive added information bit sequence. So, what we have is this important property C 0 x, C 0 y is equal to C 0 x plus y that is what we have is basically code followed by addition, and this is basically addition of the information bites followed by code this is.

(Refer Slide Time: 14:47)



And both are equal which implies that code is linear. So, that is an important property which can 1 can keep in mind that this code we have looked at a convoloutional code.

Now, similarly this property can also be shown the for coded bit 1 all right we have only shown it for 0 you can very easily shown. In fact, it is almost trivial that C 1 x plus C 1 y of j equals C 1 of x plus y of j. So, this coded bit 1. So, this is for bit 0 we remember that there are 2 code bits 1 is code bit 0 or coded bit 0 or code bit not coded bit 0 code bit 0. So, that is what we have shown**.**

So**,** we have introduced a convolutional code which we have said is basic convolutional code an example of a convolutional code of courses that is not the only convolutional code, but this is one particular example of a convolutional code again convolutional codes are basically belong to the they are basically class of error control codes or a class of channel codes, which can be used to recover from the errors recover that is recover from the errors or basically reconstruct a correct the errors right arising over the channel correct the errors arising over the channel. And we will explore the properties of this convolutional code the generation the encoding and the decoding aspects of these convolutional codes in subsequent modules.

Thank you very much.