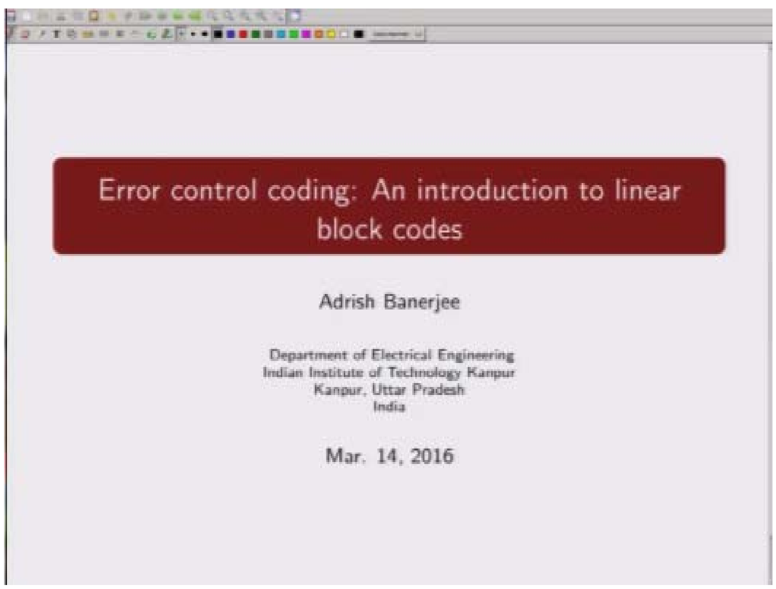
Indian Institute of Technology Kanpur
National Programme on Technology Enhanced Learning (NPTEL)
Course Title
Error Control Coding: An Introduction to Linear Block Codes

Lecture – 3A
Syndrome, Ereor Correction and Error Detection

by
Prof. Adrish Banerjee
Department of Electrical Engineering, IIT Kanpur

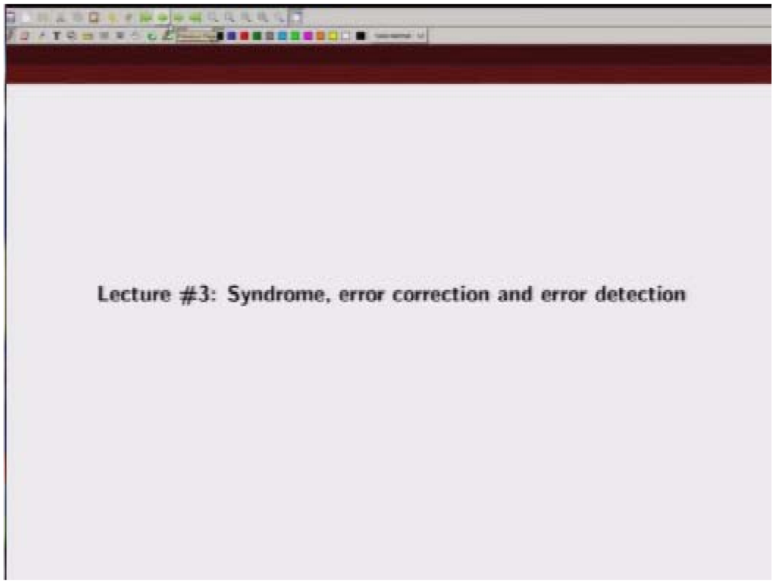
Welcome to the course on error control coding.

(Refer Slide Time: 00:17)



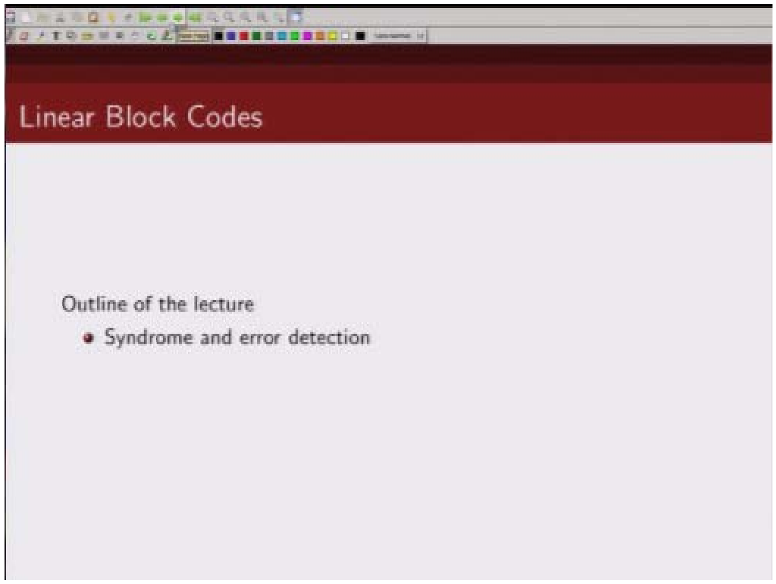
An introduction to linear block codes, in this lecture we are going to describe how we can use error correcting codes for error detection and error correction. So we will first describe what we mean by syndrome and then we will show how we can use the syndrome to do error correction and error detection.

(Refer Slide Time: 00:46)



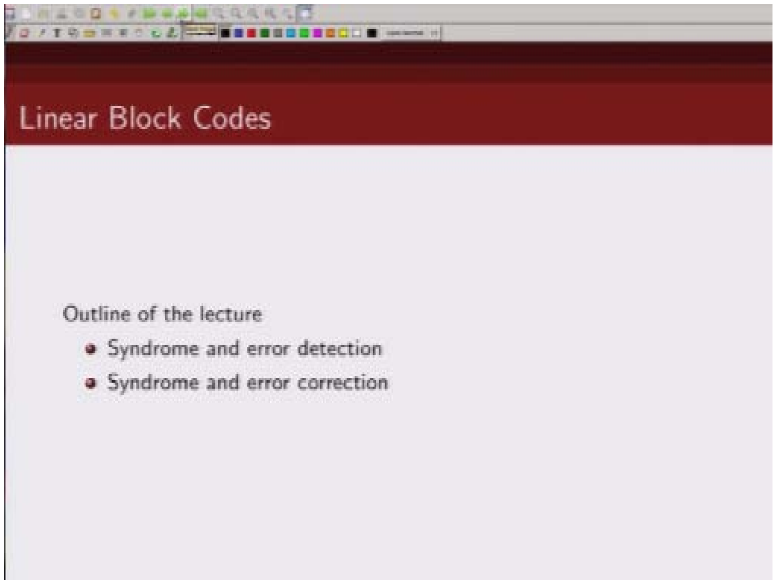
So this lecture is about syndrome and error correction and error detection.

(Refer Slide Time: 00:50)



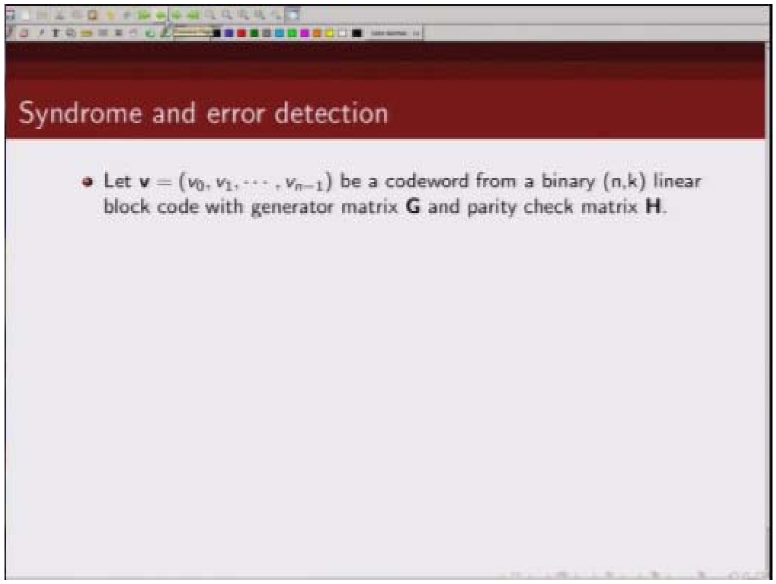
We will first talk about what is a syndrome and how we can use it for error detection and then we will talk about.

(Refer Slide Time: 00:56)



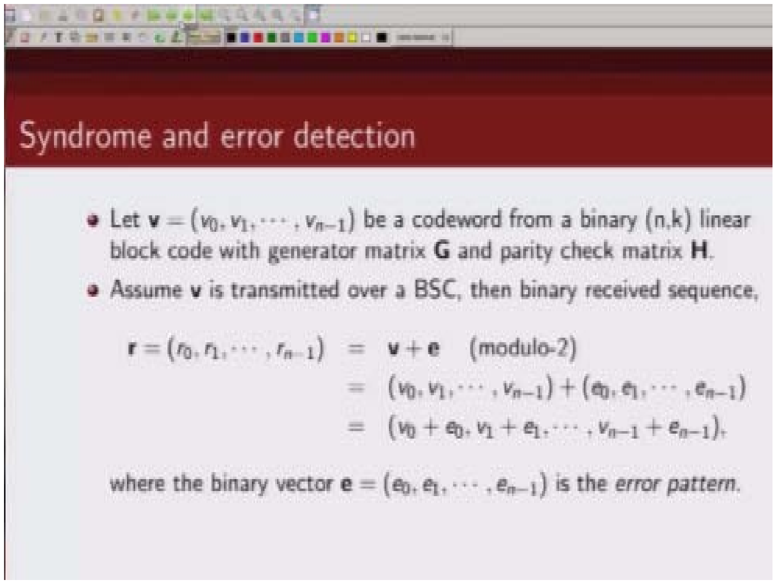
How the syndrome can be used for error correction.

(Refer Slide Time: 01:01)



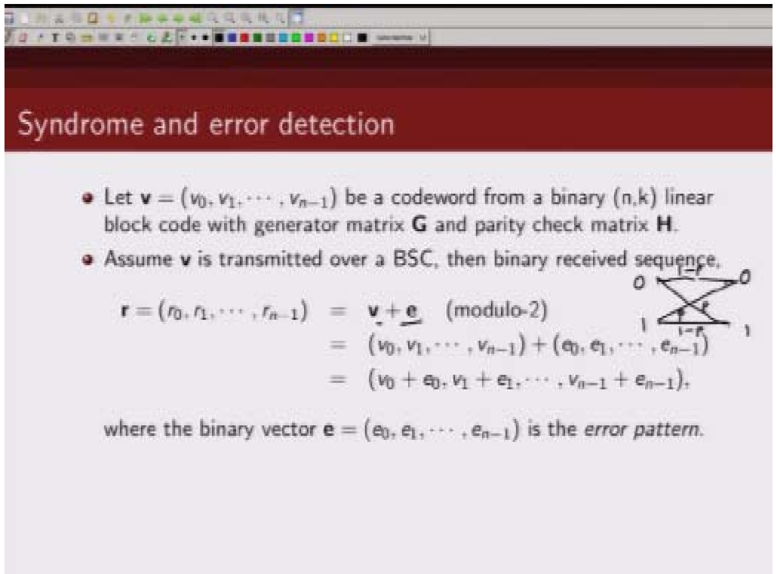
So we know so far if we have an (n, k) linear block codes so that means we have k information bits and n coded bits, and this (n, k) linear block codes is completely described by a generator matrix and a parity-check matrix.

(Refer Slide Time: 01:01)



So let us assume that we have a code word which is encoded using a linear (n, k) block encoder, so we have an output of a, encoder which is of coded bits of length n. Now we want to transmit this code word over a communication channel, for simplicity.

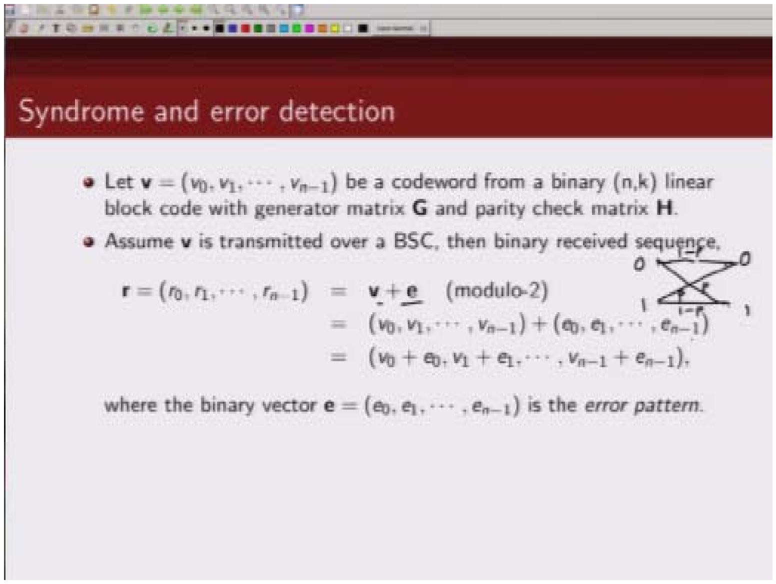
(Refer Slide Time: 02:05)



Let us consider a binary symmetric channel again let us recall what is a binary symmetric channel so in a binary symmetric channel we have two inputs and two outputs, so binary inputs 0's and 1, binary output, 0's and 1 and with probability let us say probability 1-P, we received the bits correctly and there is a crossover probability of P that the bits get flipped. So this is a binary symmetric channel.

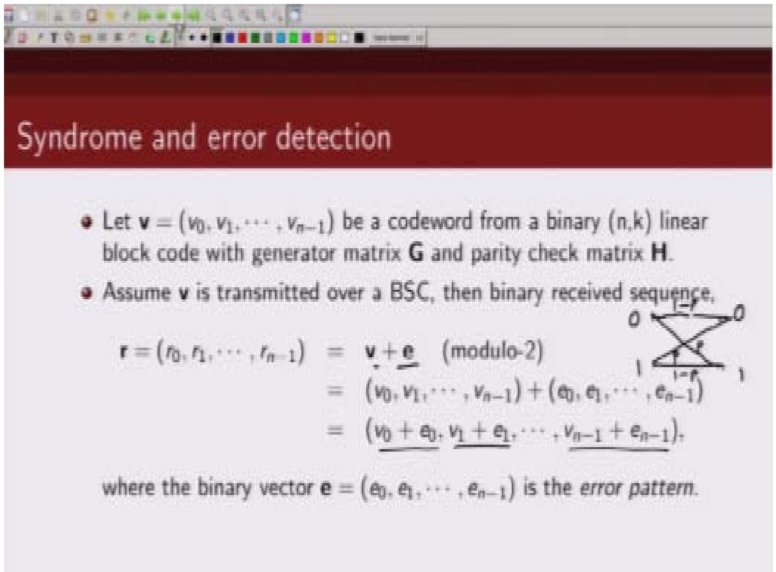
Let us denote by r, our receive sequence receive code word which we sent over this binary symmetric channel. So the types of, as I said the output of the binary symmetric channel is also 0 and 1, so we can describe the output of a binary symmetric channel r as our transmitted code word plus some error vector. So our transmitted code word is an n bit tuple, similarly our error vector is also am n bit.

(Refer Slide Time: 03:06)



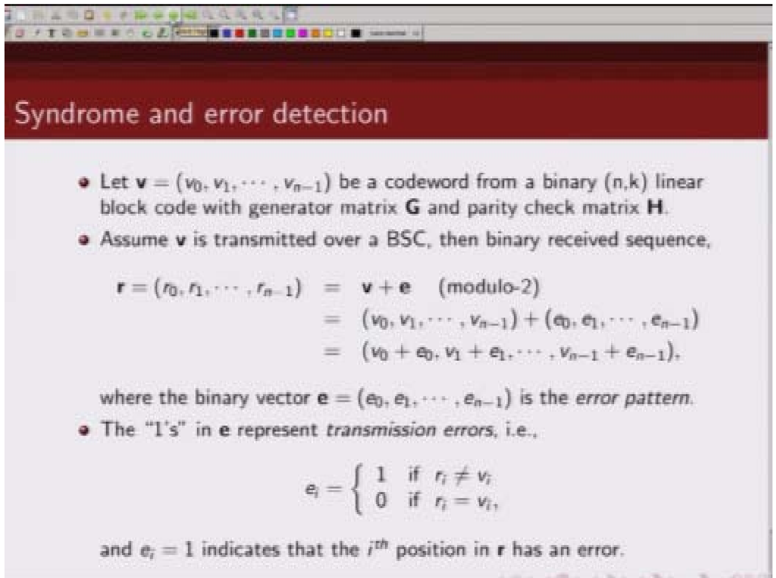
So we describe our error vector by e_0, e_1, e_2, e_{n-1}, and whenever an e_i term is 1 that means that particular bit was not received correctly. So we can write an output of a

(Refer Slide Time: 03:26)



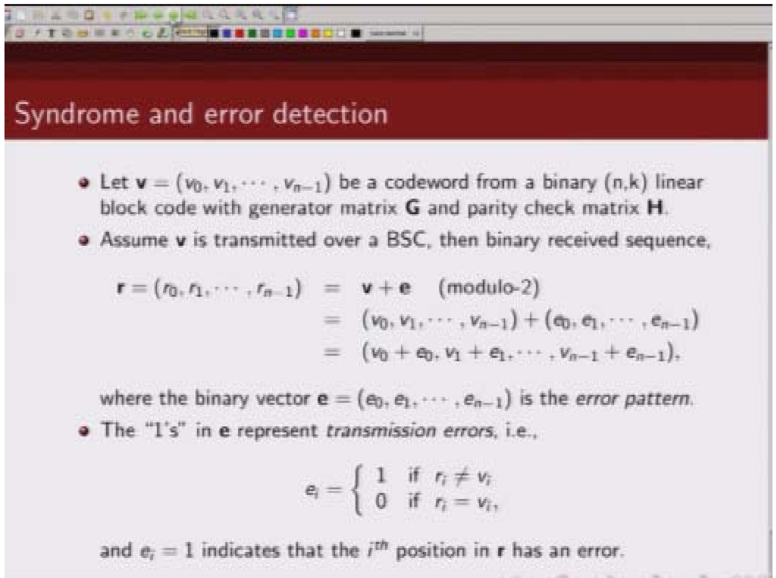
binary symmetric channel, so the first bit that we would receive is basically $v_0 + e_0$, v_1 plus, second bit would be $v_1 + e_1$, similarly $v_2 + e_2$ and last finally we will get v_{n-1}, e_{n-1} where this e_0, e_1, e_2, e_{n-1} is my error pattern.

(Refer Slide Time: 03:52)



Now when does an error occur, if my receive sequence is not same as my transmitted code word then there is an error.

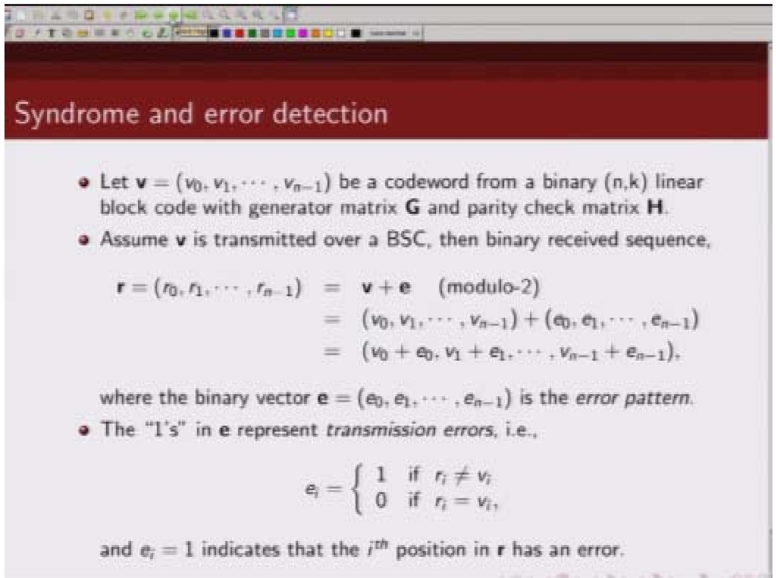
(Refer Slide Time: 04:01)



So the error is 1 only if my receive bit is not same as my transmitted bit. If the receive bit is same as my transmitted bit there is no error. So that, I will keep that bit e_i 0, okay? Now if a particular bit e_i is 1 what does it denote? It denotes that i^{th} bit is in error.

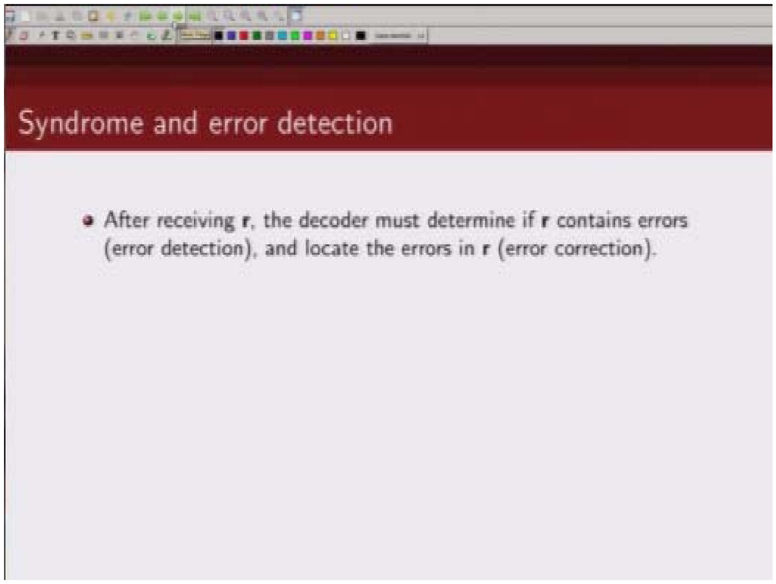
Now when we are sending this code word over a communication channel what are we interested in, we are interested to find out whether any error has occurred, if any error has occurred we are interested to find out the location where the error has occurred so that we can correct those errors.

(Refer Slide Time: 04:56)



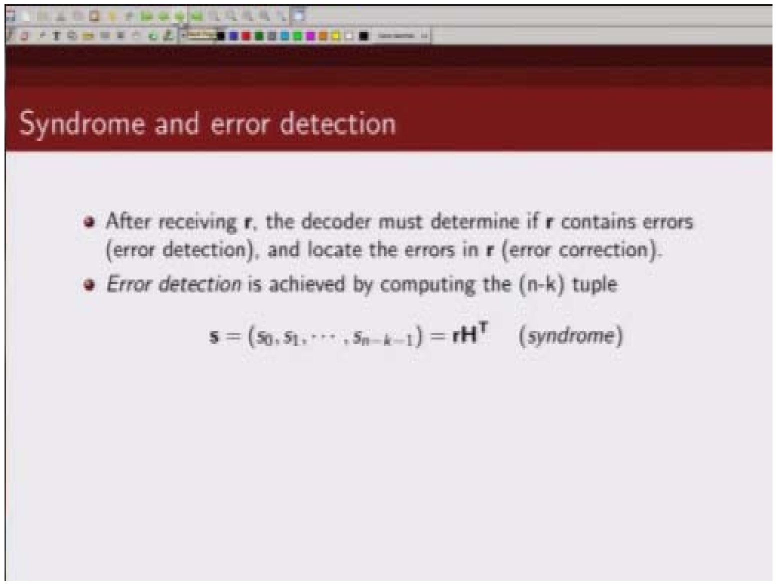
Okay?

(Refer Slide Time: 04:58)



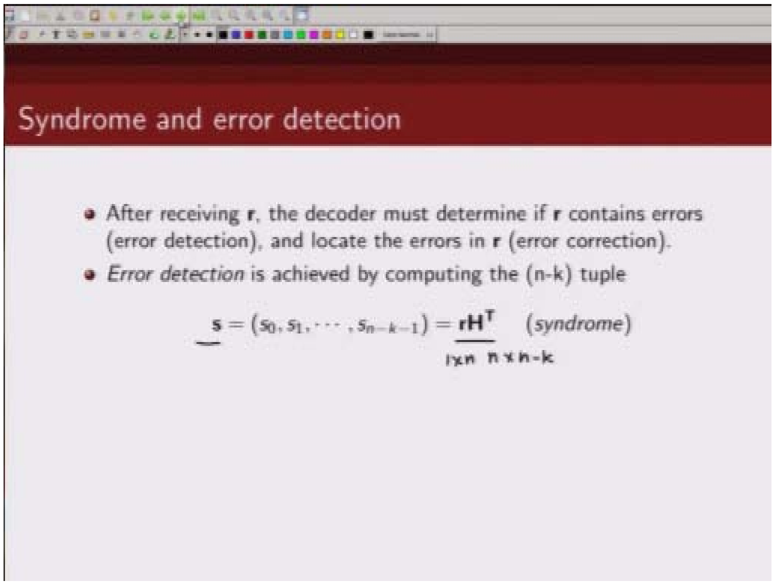
So first we are going to show how we can detect error.

(Refer Slide Time: 05:04)



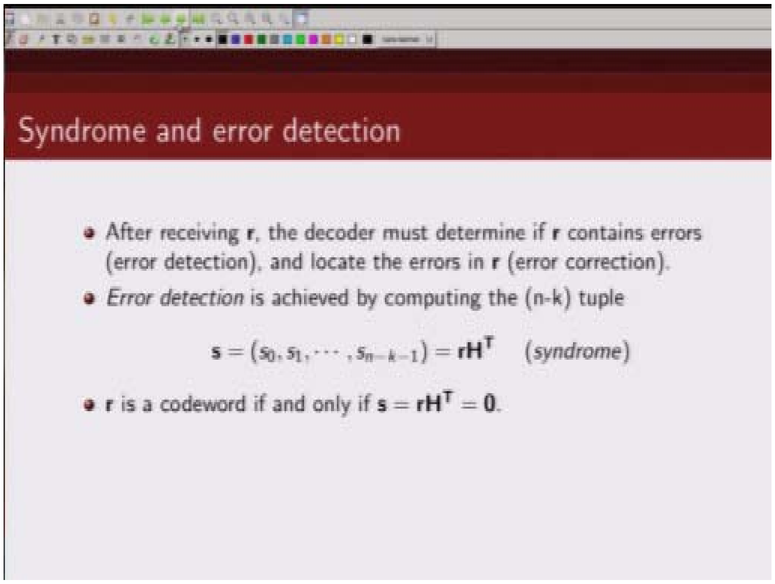
So we define a term which we call as syndrome.

(Refer Slide Time: 05:07)



What is a syndrome, syndrome is computed by computing this rH^T so is basically 1 x n vector, H is my n- k x n vector, so H^T n x n-k matrix. So this term rH^T is known as syndrome and this is when this is

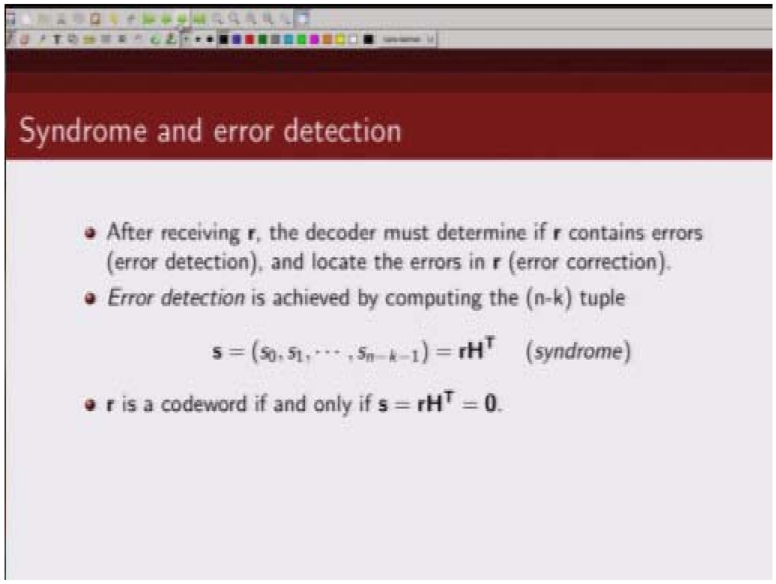
(Refer Slide Time: 05:37)



Non-zero it indicates that there is an error. So r is a code word if and only if the syndrome is 0. So whenever the syndrome is 0, basically if the syndrome is 0 then r is the code word if and only if syndrome is 0 and this is easy to check because we know that if v is a valid code word then vH^T will be 0.

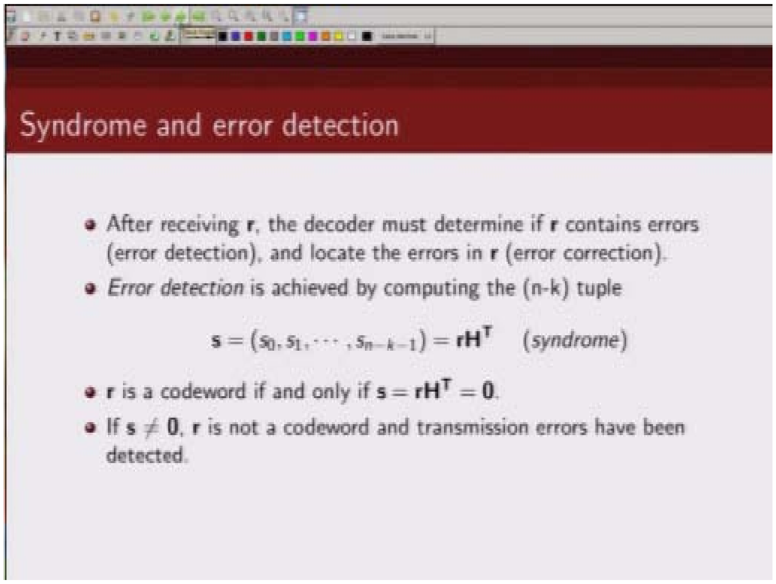
So if there is no error my receive sequence r would be just equal to v and then

(Refer Slide Time: 06:21)



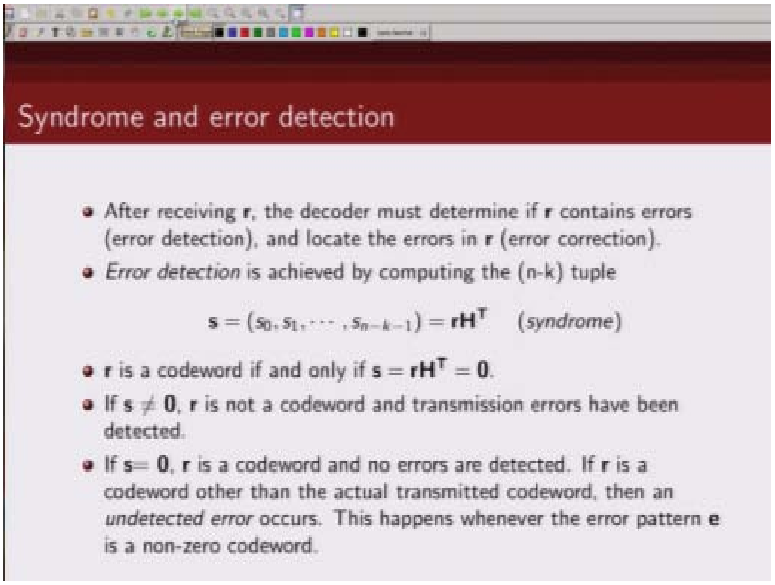
Syndrome would be vH^T which is 0.

(Refer Slide Time: 06:28)



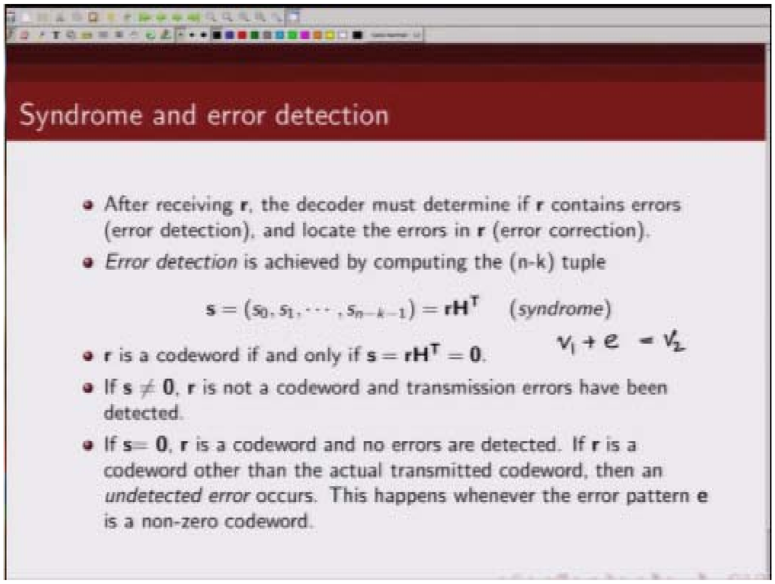
And if syndrome is not equal to 0 then it means there is an error. Now if the syndrome is 0 does it always mean that there is no error?

(Refer Slide Time: 06:47)



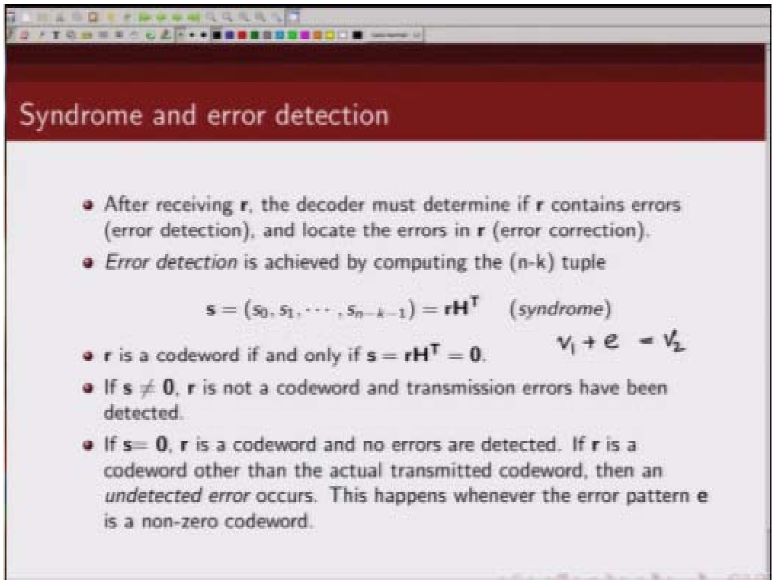
No there is a category of error which we call as undetected error. Now when does an undetected error happen?

(Refer Slide Time: 06:58)



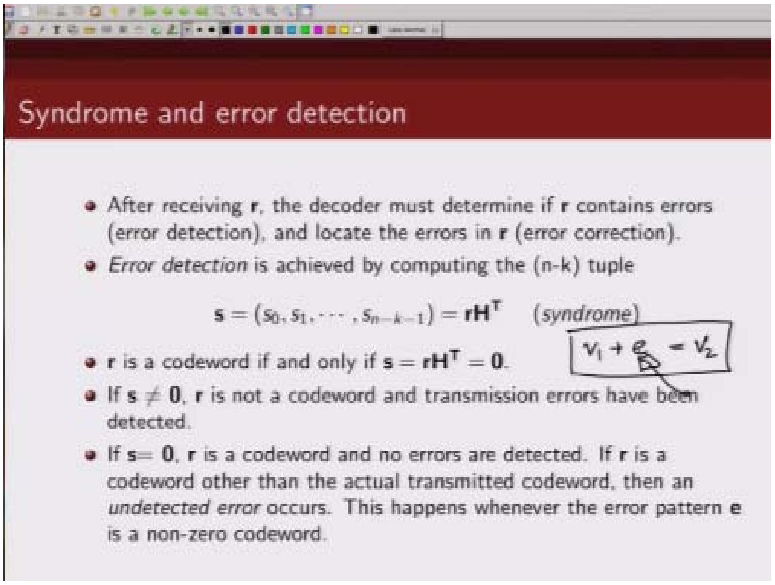
If your s is 0 and your receive sequence r is not the code word that you transmitted but some other code word, let us say I transmitted a code word v_1 and my error e was such that, that it transformed it into another code word v_2, so receive sequence is v_2. Now if I compute syndrome because v_2 is a valid code word then vH^T will be 0. So an undetected error happens when your error pattern is such that it transforms your one code word to some other code word.

(Refer Slide Time: 07:46)



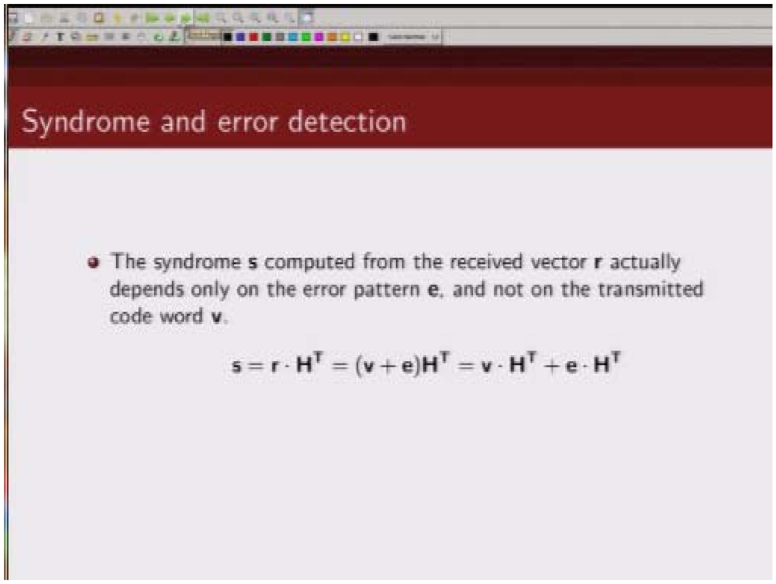
And that is what we have written here, so if s is 0 and r is a code word that means no errors are detected. However if r is a code word other than the actual code word transmitted then an undetected error had happened, and this happens when an error pattern is a non-zero code word, because we have said a property of linear block code, that sum of two code words is also a code word.

(Refer Slide Time: 08:18)



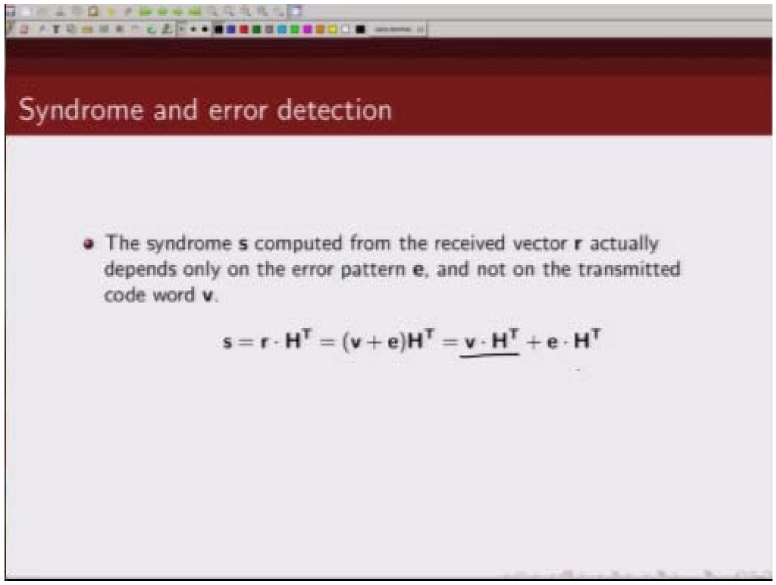
So for this scenario to happen this e has to be a valid non-zero code word.

(Refer Slide Time: 08:28)



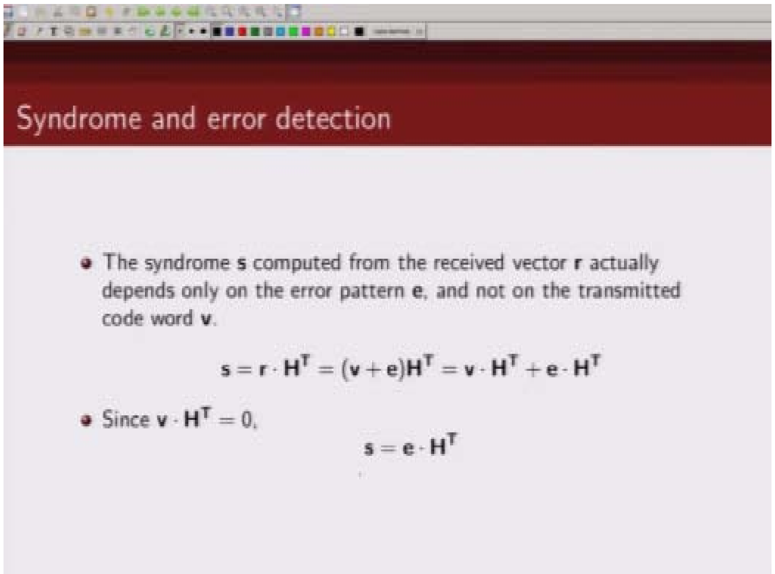
So as I said we compute the syndrome from the receive sequence r, the interesting part is syndrome depends only on the error pattern, it does not depend on what code word was transmitted. And this is easy to see.

(Refer Slide Time: 08:28)



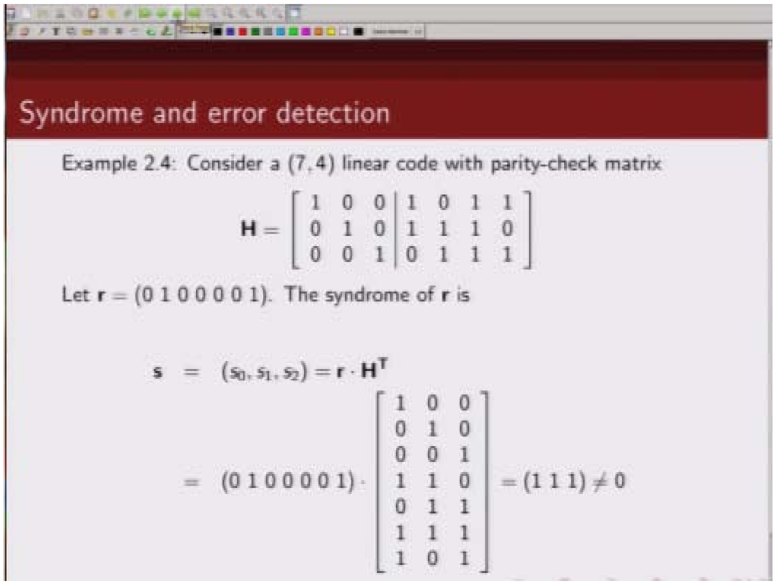
So syndrome is rH^T which we can write r is my transmitted code word plus error vector, this whole multiplied by H^T so this I can write as vH^T and eH^T. Now what is vH^T, vH^T is 0 because v is a valid code word, then syndrome is nothing but eH^T.

(Refer Slide Time: 09:16)



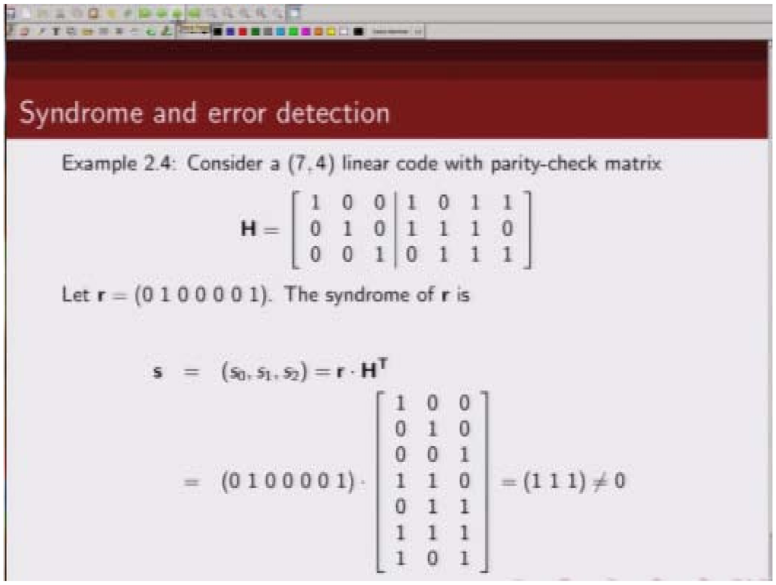
So syndrome does not depend on transmitted code word, it only depends on the error pattern.

(Refer Slide Time: 09:26)



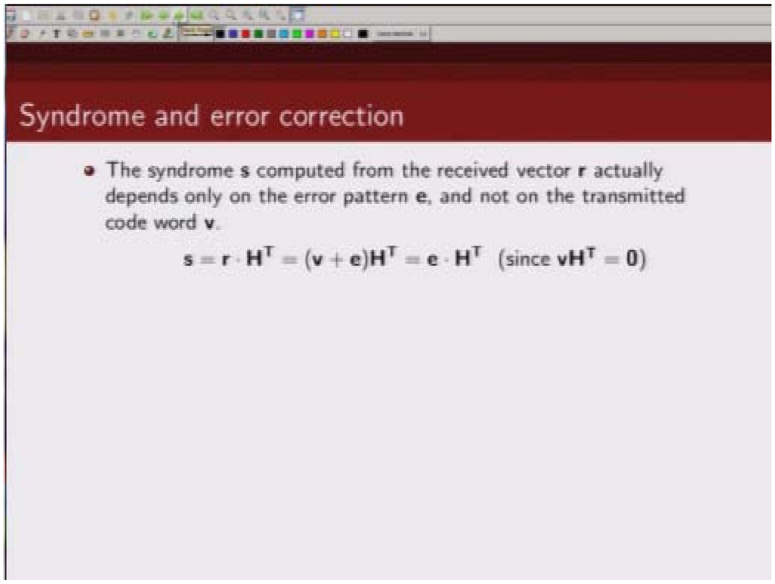
So I have an example here of a (7, 4) linear block code whose parity-check matrix is given this. Now let us say my receive sequence is, received coded sequence is this, and I am interested to find whether there is any error in this receive sequence. So how do I do that? So first I will compute the syndrome.

(Refer Slide Time: 09:50)



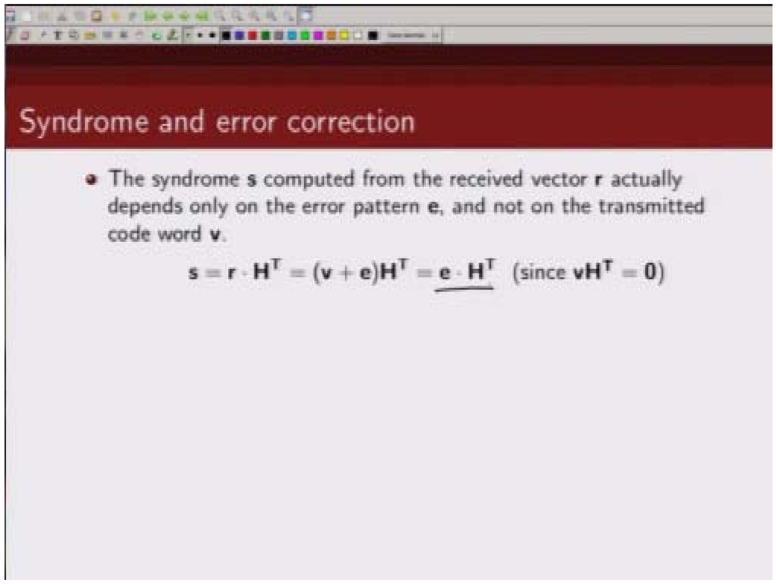
So what is syndrome of r? This syndrome is rH^T so r is this, and H is given, so H^T is basically 100,010,001, so this my H^T and when I multiply this I multiply this by this, multiply this by this, multiply this by this, what I get is 111 which is not 0. That means there is an error in my received sequence.

(Refer Slide Time: 10:31)



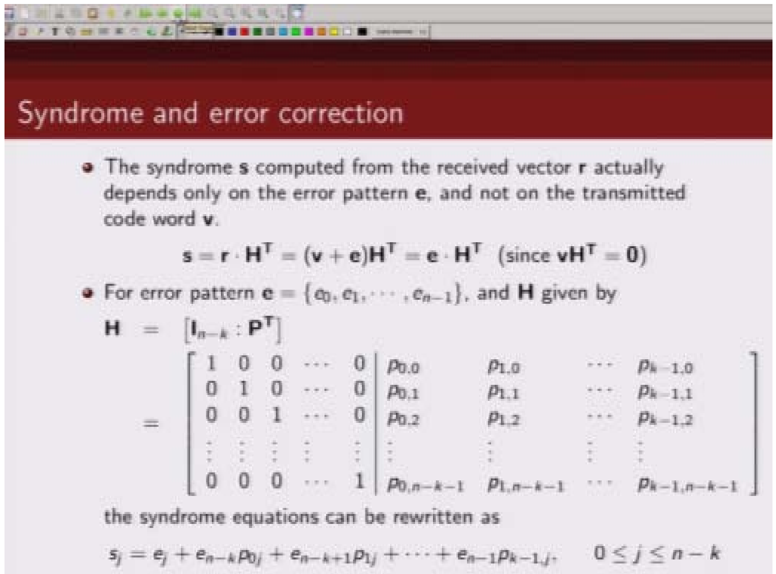
As I said the syndrome only depends on the

(Refer Slide Time: 10:33)



Error pattern, it does not depend on transmitted

(Refer Slide Time: 10:37)

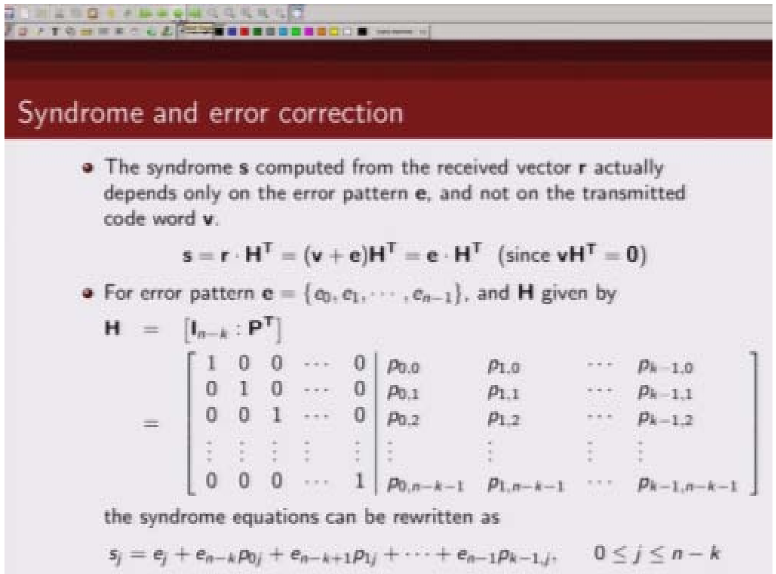


Code word, and if you have your parity-check matrix in a systematic form then you can write your syndrome equations in terms of your error patterns like this. So these are your basically n-k a syndrome equations.

(Refer Slide Time: 10:57)

So to recap what we have said so far we have said that for error detection we need to compute the syndrome and if the syndrome is non zero that means an error has occurred. Now we will move into error correction and we will talk about how we can use the syndrome for error correction. So as we saw basically

(Refer Slide Time: 11:33)

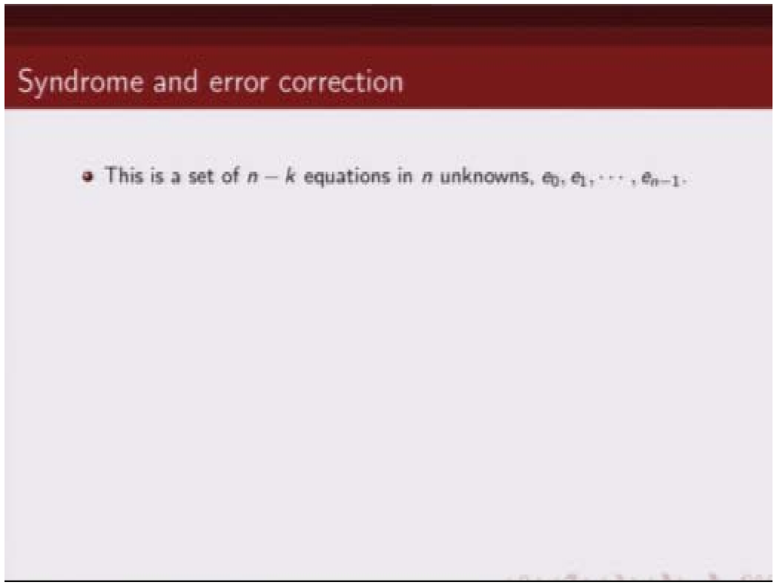


From the syndrome equations we had basically this was 1 x n and H^T was n x n-k so we essentially had

(Refer Slide Time: 11:44)

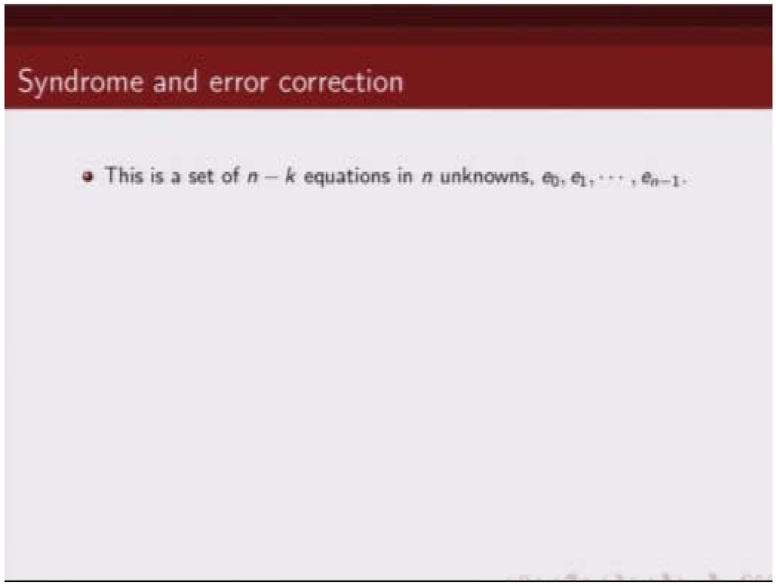
n-k syndrome equations, so we had total n-k equations and how many unknowns we have, we have n unknowns from location error at location 1 to error at location n. So we have n unknown

(Refer Slide Time: 12:01)

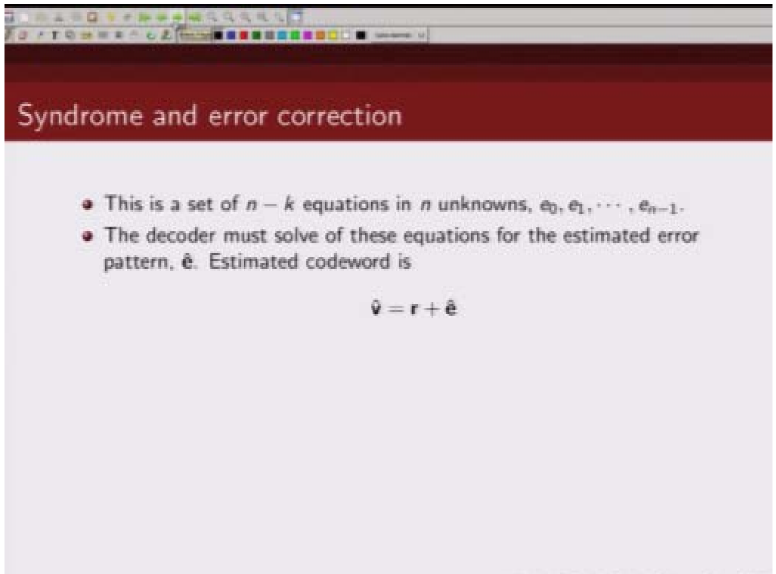


e_0, e_1, e_2 e_{n-1} whereas we have only n- k equations so we have less equations, more unknown.

(Refer Slide Time: 12:13)

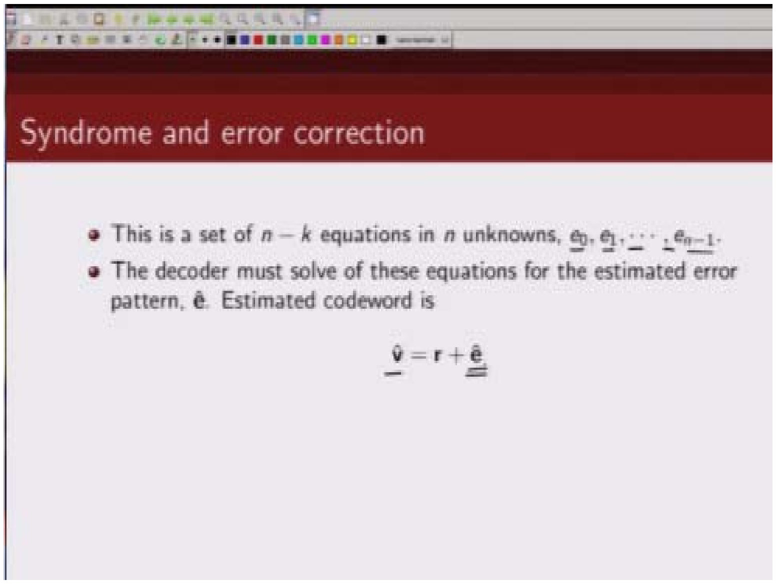


(Refer Slide Time: 12:15)



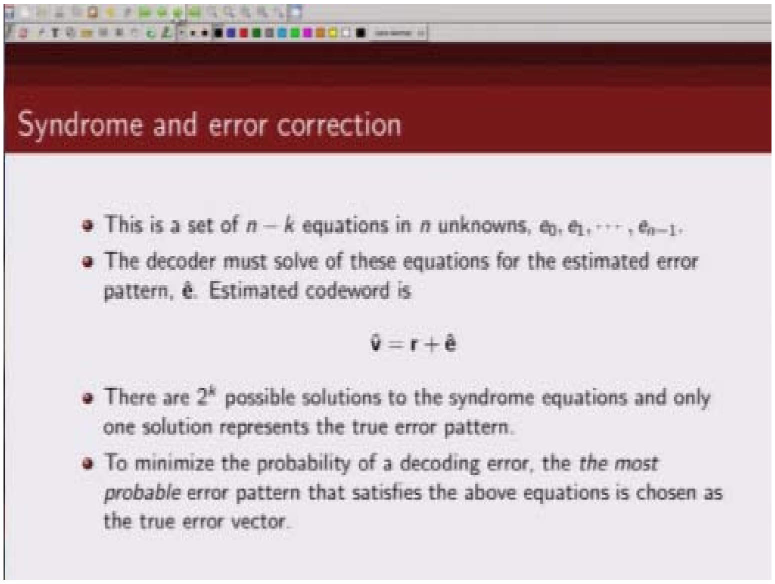
And what we need to do is we need to solve these set of equations to find out what these unknown quantities are, because to find out the error pattern we need to know

(Refer Slide Time: 12:30)



What these e_i's are okay, so we need to solve these set of n-k equations to get back our corrected sequence, and now what would be our corrected sequence? So our estimated code word would be nothing but r, receive sequence plus estimated error pattern. So we need to when we do error correction essentially what we are trying to do is we are trying to estimate what the error pattern is.

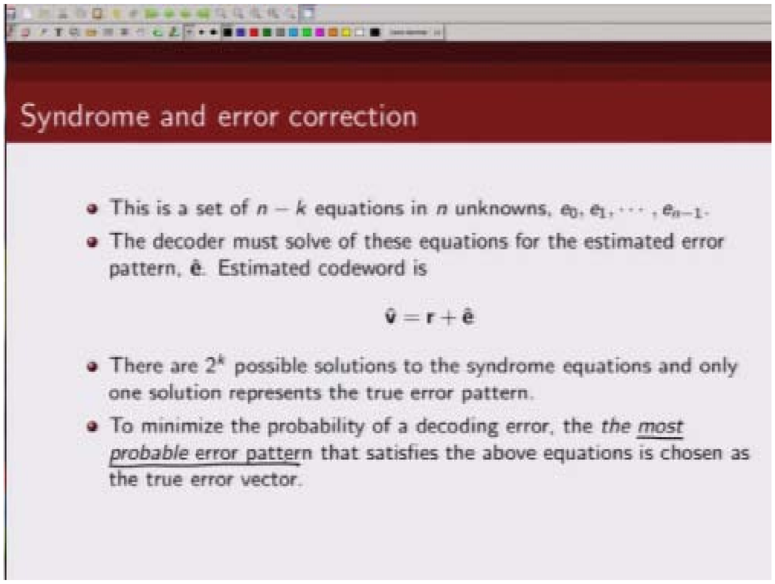
(Refer Slide Time: 12:30)



So as we can see from here because we have n-k equations and n unknowns we have total 2^K solutions of these n-k equations. So there are total two k solutions to these syndrome equation and out of them there is only one which is the correct error pattern, out of those 2k solutions there is only one error pattern which is correct. Now how do we choose the most likely pattern from these 2^k solutions, how do we choose the most likely pattern that is basically what our objective is.

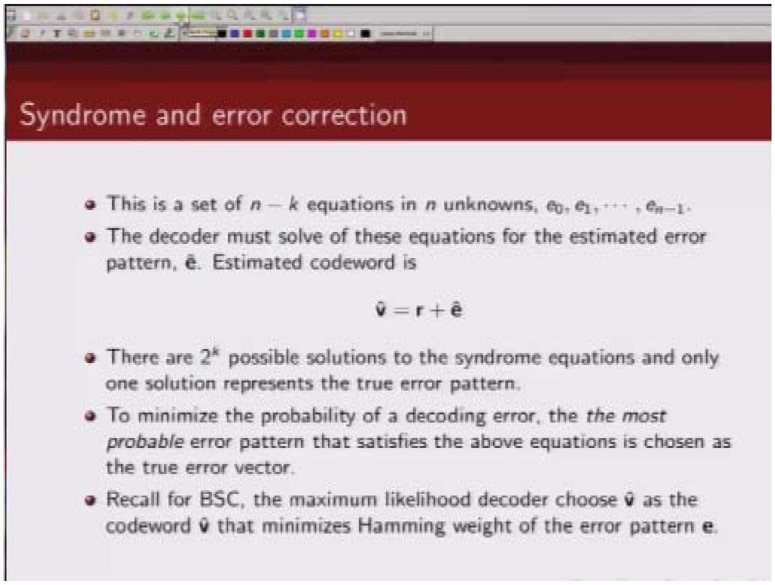
So when we try to minimize probability of error.

(Refer Slide Time: 14:00)



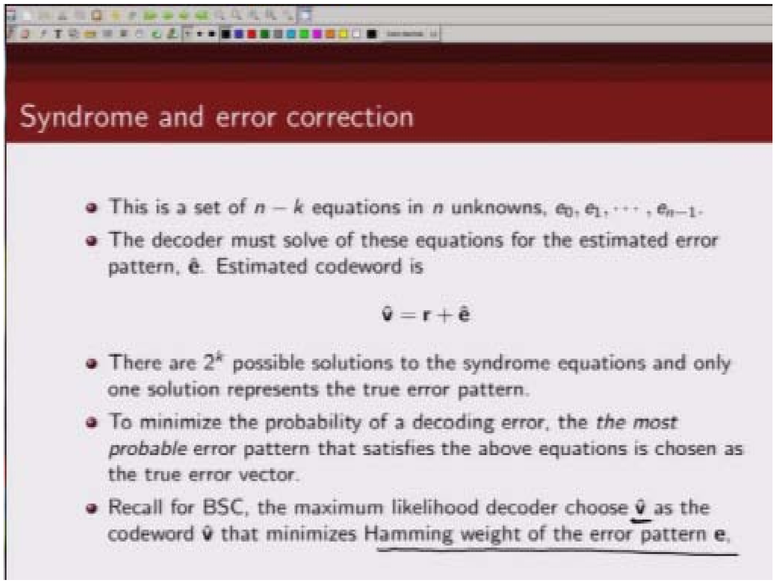
We want to choose the most probable error pattern, we want to choose the most probable error pattern from those 2^k solutions of this set of equations, and as we said we did an exercise

(Refer Slide Time: 14:19)



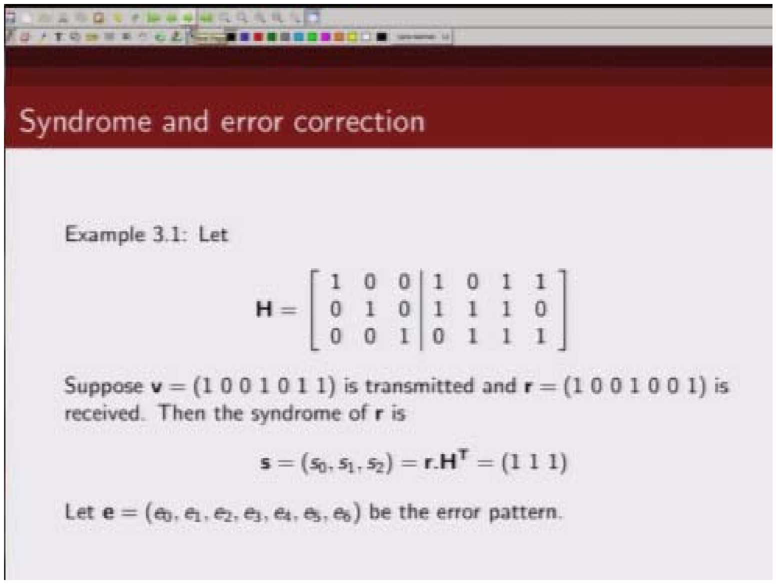
You know in previous lectures we have shown the decoding rule for maximum-likelihood decoding rule for a binary symmetric channel and we have shown that for maximum-likelihood decoder we will choose

(Refer Slide Time: 14:38)



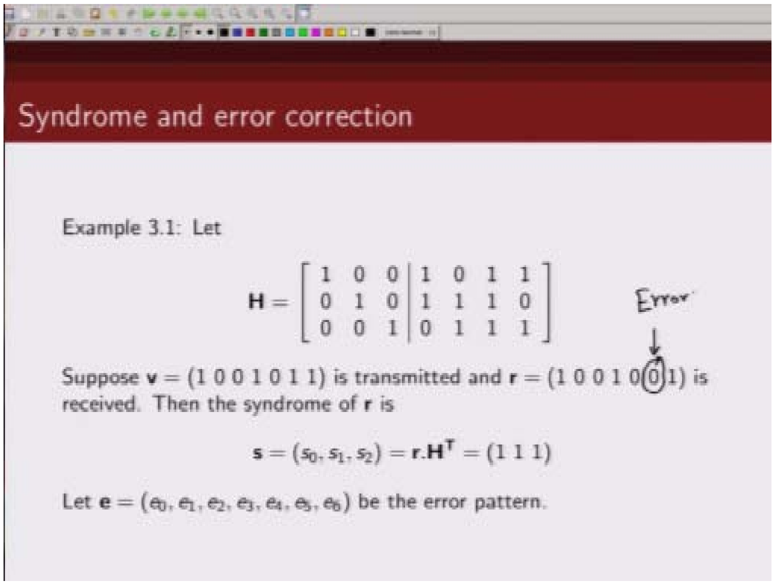
Our estimated code word as one that will minimize the hamming distance between the receive code word and the transmit code, in other word which would minimize the hamming weight of the error pattern e. So from the 2^k different solutions basically the one which has the least hamming weight that is the best solution for a maximum likelihood decoding rule for a binary symmetric channel.

(Refer Slide Time: 15:11)



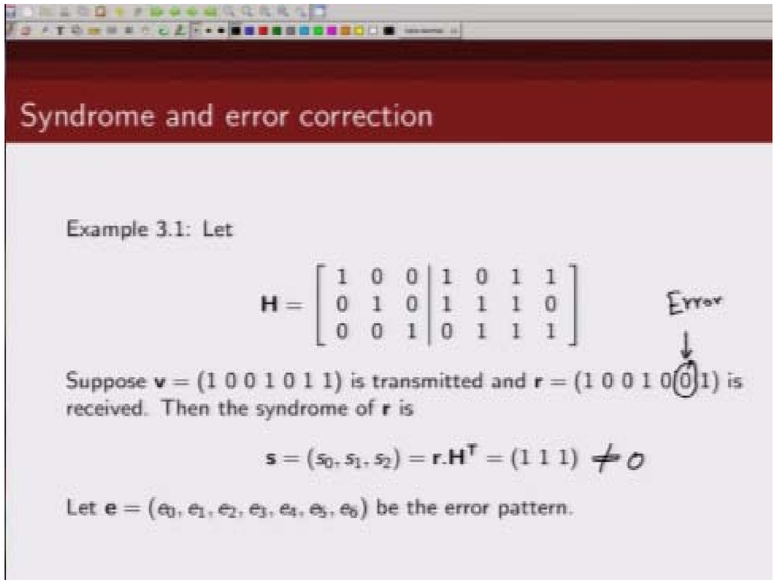
So let us take an example now, so we have a 7, 4 code whose parity check matrix is given by this and our transmitted code word is this and received code word is this, we can see that

(Refer Slide Time: 15:28)



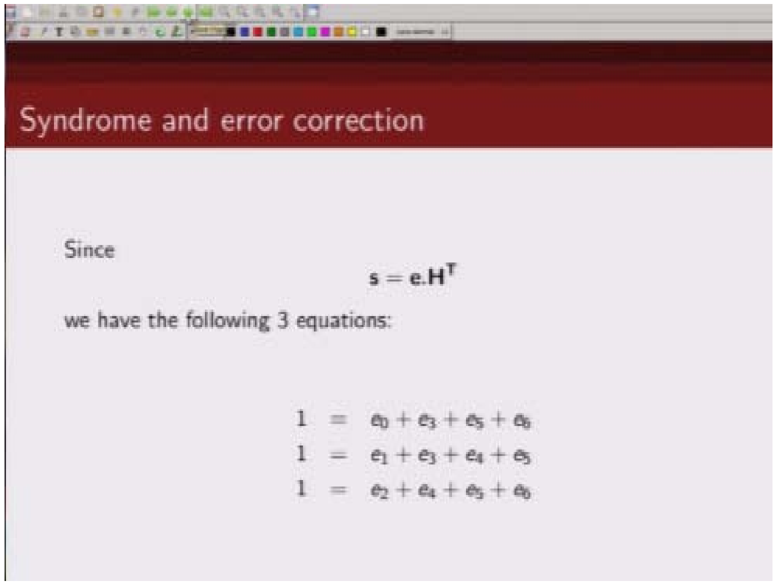
There is an error in this location, this was transmitted as one and this was received as zero, so there is an error in this location. Now how do we find out that there is an error and there is an error in this location? So first thing that we will do is we will compute the syndrome, when we compute the syndrome which is rH^T

(Refer Slide Time: 15:55)



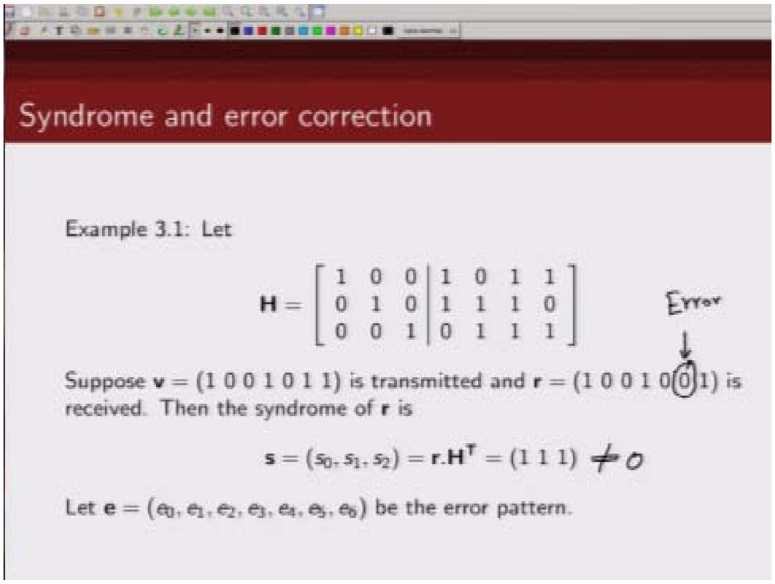
What we get is a non zero and since this is a non zero this means there is an error. Now next step is to find out where the error has occurred. So in this case

(Refer Slide Time: 16:14)



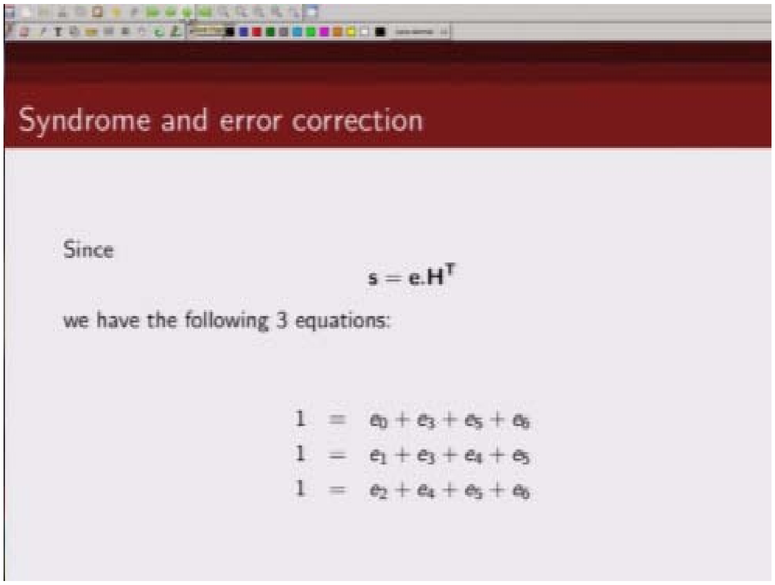
We can write the syndrome in terms of error bits, so we have three equations and we have total 7 unknown's right and these equations are basically given by this.

(Refer Slide Time: 16:34)



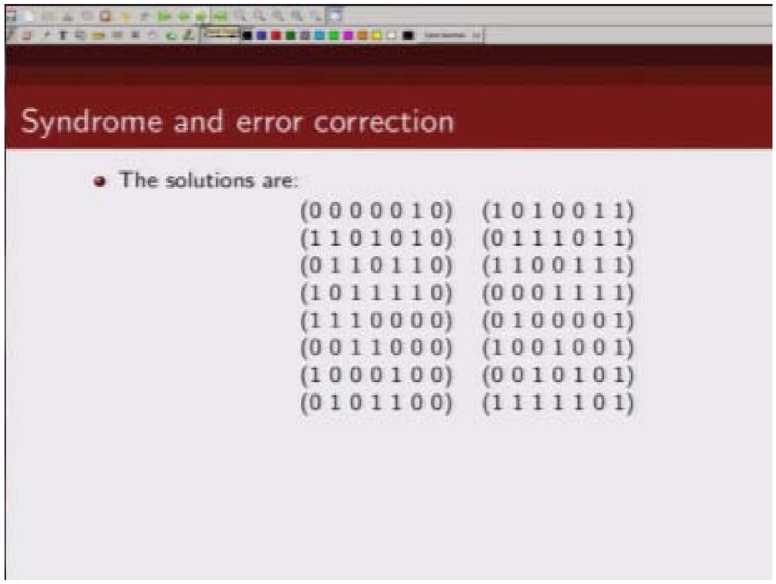
We have our H matrix is given by this, this is our error pattern so when we

(Refer Slide Time: 16:40)



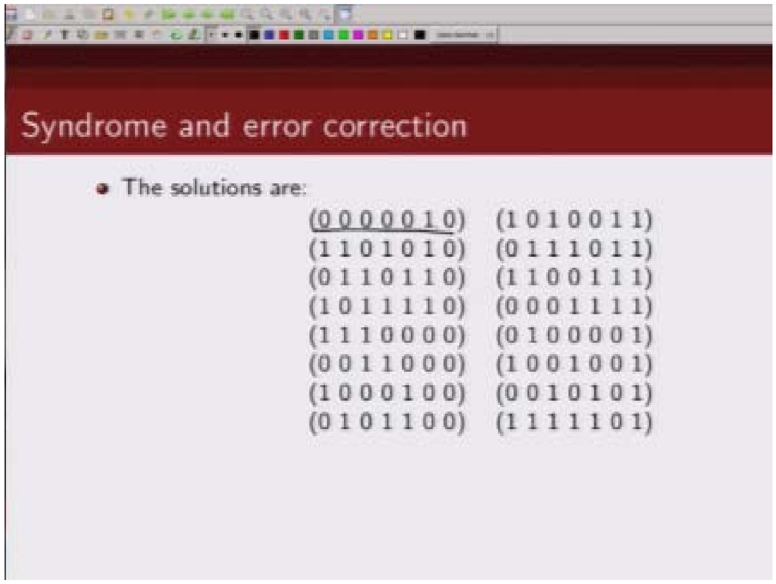
When we compute eH^T we get these set of three equations.

(Refer Slide Time: 16:46)



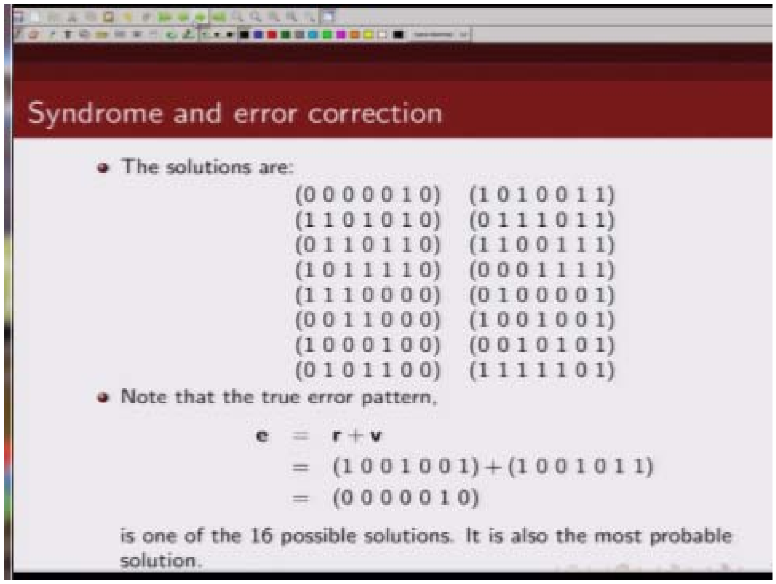
And there are sixteen solutions to this set of equations this set of three equations where there seven unknowns there are total 2^4 different solutions and these are the sixteen different solutions and as we said the maximum likelihood decoding rule for a binary symmetric channel will choose an error pattern that as minimum Hamming weight.

(Refer Slide Time: 17:20)



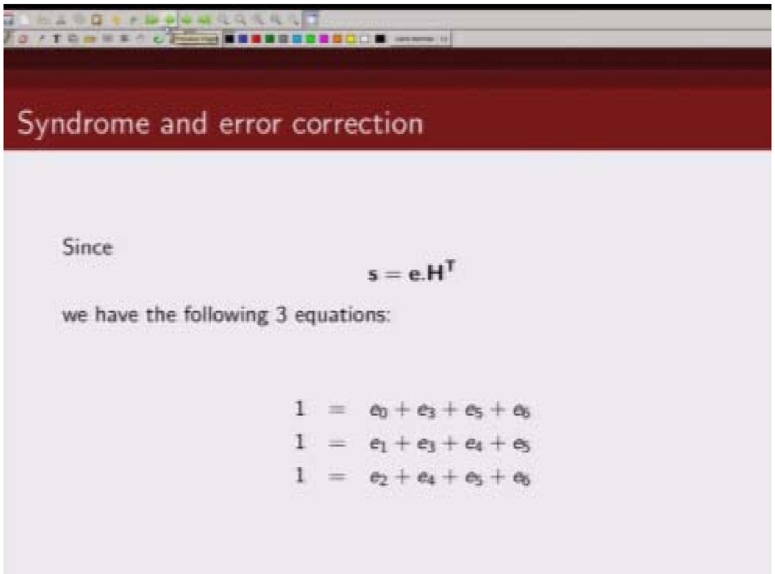
So which has minimum number of ones, so you can see among these sixteen solutions the one that has minimum hamming weight minimum number of ones is this, all other have this has four ones, this have four ones, the five ones, three ones, two ones, you can check basically this one have the least number of ones, so this error pattern has the least of hamming weight so this is out of those 2^4 solutions. This is the most likely solution and

(Refer Slide Time: 17:55)



This we can verify also because we were given the receive sequence and we knew the what were the transmit sequence was error pattern indeed was this which we found out from by solving this set of equations. So to recap then now when we want to do error correction what do we need to do, we need to solve for

(Refer Slide Time: 18:26)



This syndrome equations and there are n-k equations but there will be n unknowns, so this will have 2^k solutions and we will have to pick the most probable solutions from these set of 2^k solutions. The next lecture we are going to talk about a general decoding algorithm for a linier block code, thank you.

Acknowledgement
Ministry of Human Resource & Development

Prof. Satyaki Roy
Co-ordinator, NPTEL IIT Kanpur

NPTEL Team
Sanjay Pal
Ashish Singh
Badal Pradhan
Tapobrata Das
Ram Chandra
Dilip Tripathi
Manoj Shrivastava
Padam Shukla
Sanjay Mishra
Shubham Rawat
Shikha Gupta
K. K. Mishra
Aradhana Singh
Sweta

Ashutosh Gairola
Dilip Katiyar
Sharwan
Hari Ram
Bhadra Rao
Puneet Kumar Bajpai
Lalty Dutta
Ajay Kanaujia
Shivendra Kumar Tiwari

an IIT Kanpur Production