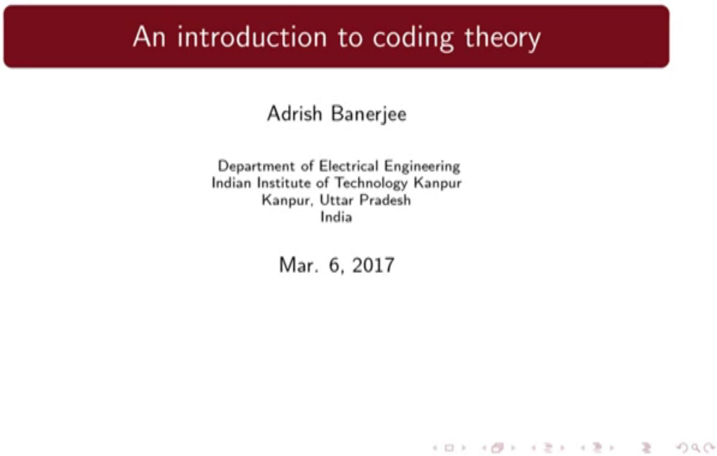
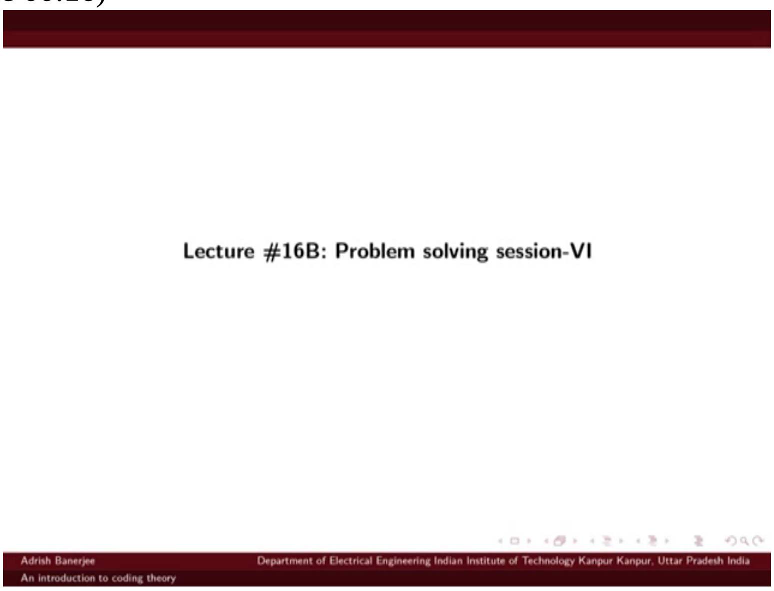
An Introduction to Coding Theory
Professor Adrish Banerji
Department of Electrical Engineering
Indian Institute of Technology, Kanpur
Module 07
Lecture Number 29
Problem Solving Session-VI

(Refer Slide Time 00:14)



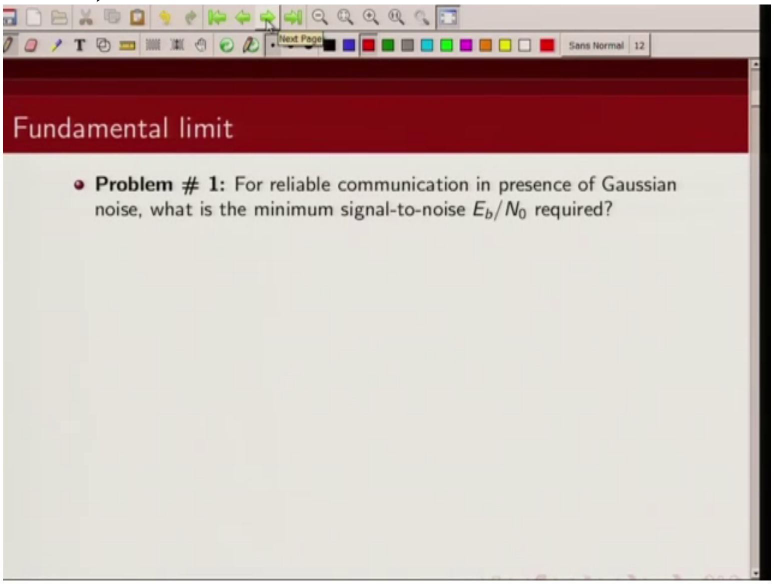
So in this lecture we will try to solve some problems

(Refer Slide Time 00:16)



related to convolutional code in general.

(Refer Slide Time 00:20)

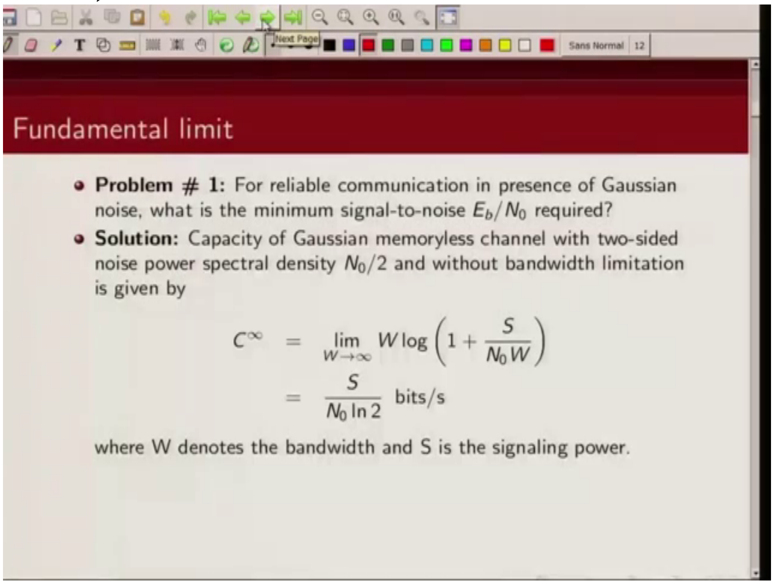


So the first question that we will try to answer is, if you want to do reliable communication in presence of additive white Gaussian noise channel and of course we have infinite bandwidth what is the minimum signal to noise ratio

(Refer Slide Time 00:37)

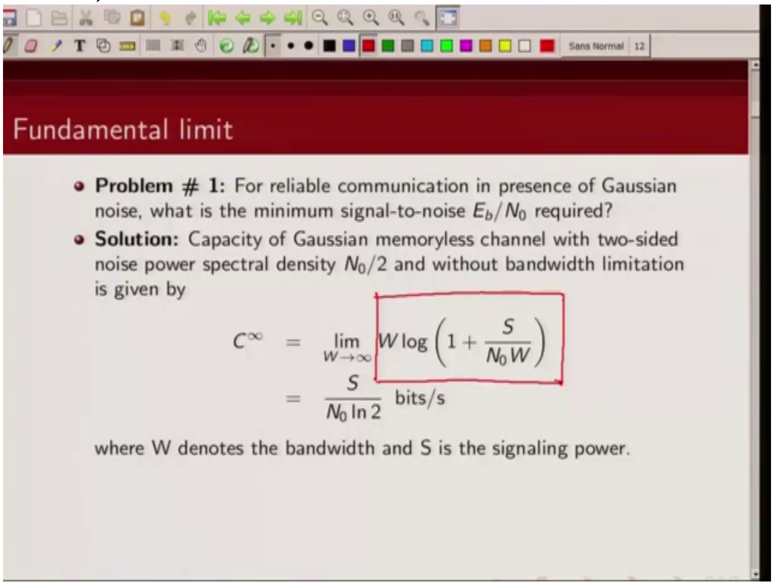
required?

(Refer Slide Time 00:41)



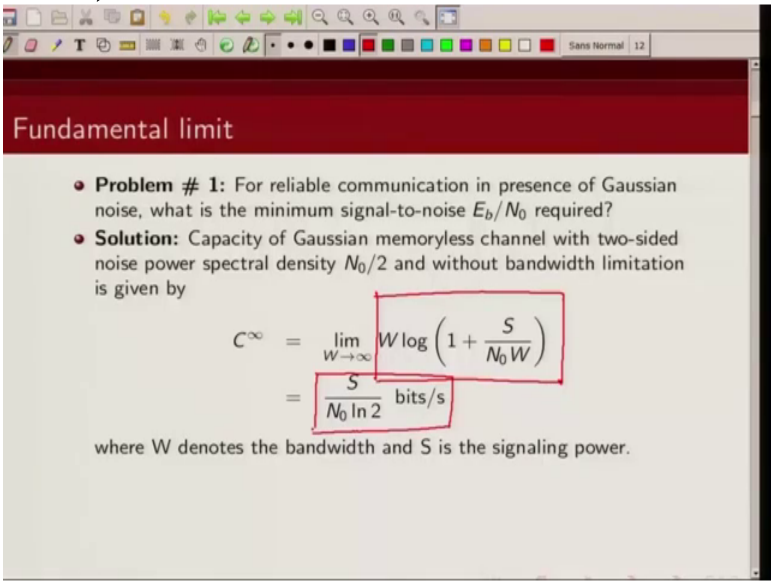
So for that we first need the expression for capacity of additive white Gaussian noise channel and the capacity of additive white Gaussian noise channel is given by this expression

(Refer Slide Time 00:58)



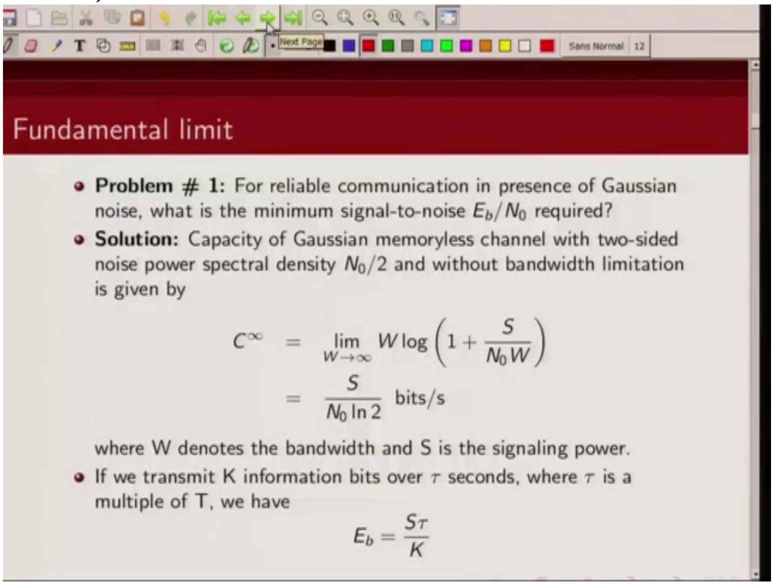
where w is the bandwidth, s is my signaling power, n naught by 2 is two sided power spectral density. Now, so we are considering when bandwidth is infinite. So this can be written, so this will be log 1 plus when w is infinite this will go to zero so log of 1 will be zero and 1 by w will also go to zero so zero by zero form. So we will differentiate and we can find out that the capacity when bandwidth is infinite is given by this expression,

(Refer Slide Time 01:44)



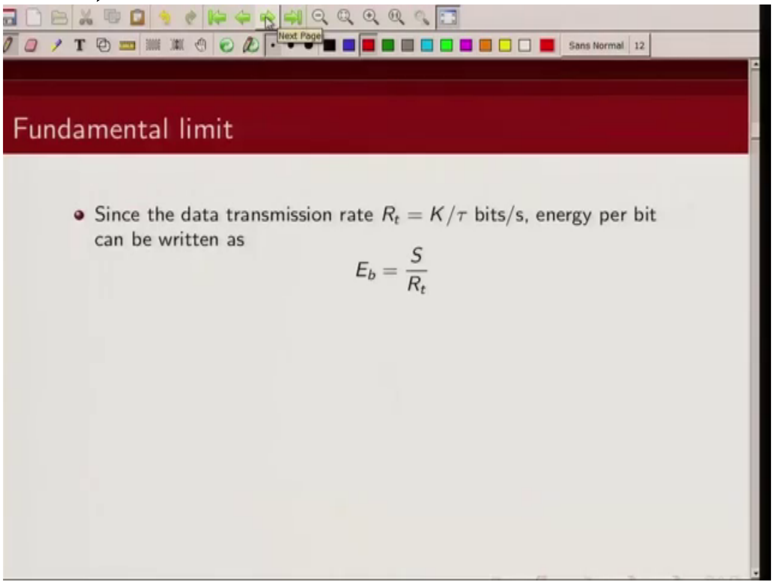
s by n naught, natural log of 2 bits per second. So we are interested

(Refer Slide Time 01:54)



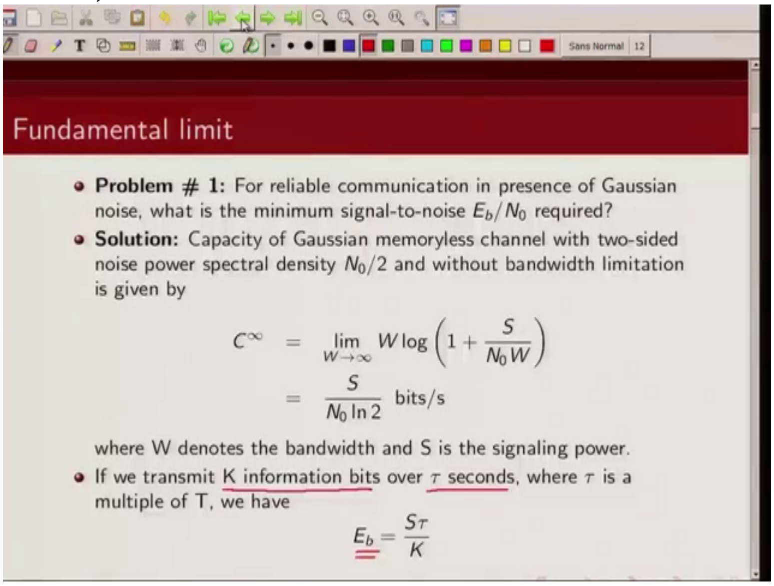
in transmitting k bits over tau seconds. So if we do that, where tau is the multiple of time period, so if we do that, our energy per bit is given by s tau by k. This s was my signaling power we are sending over time t, tau and total number of information bits was k. Energy per bit is s tau divided by k.

(Refer Slide Time 02:30)



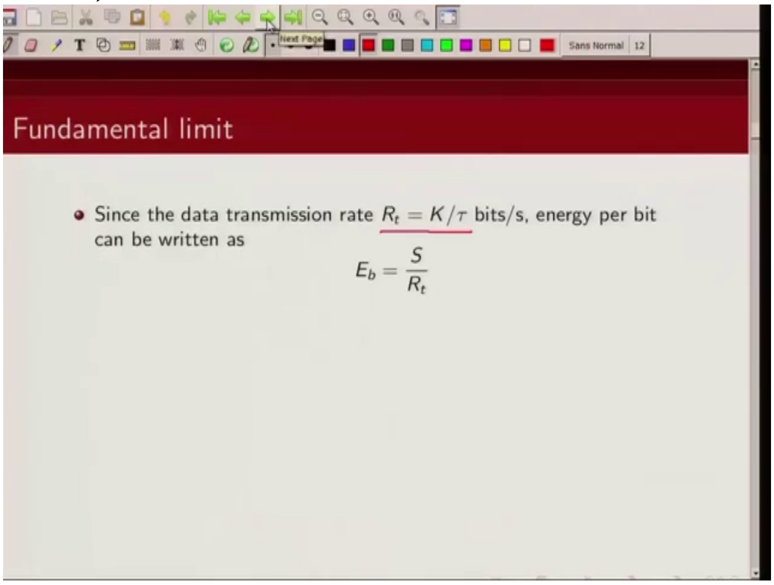
Now our transmission because you are transmitting k bits over time tau, so our transmission rate is k by tau bits per second. And our energy per bit that we wrote here

(Refer Slide Time 02:46)



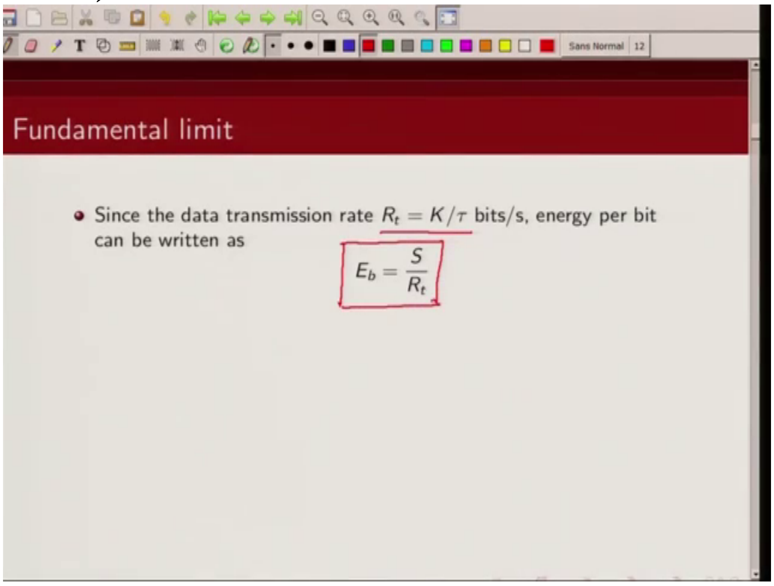
is basically s tau by k. And

(Refer Slide Time 02:51)



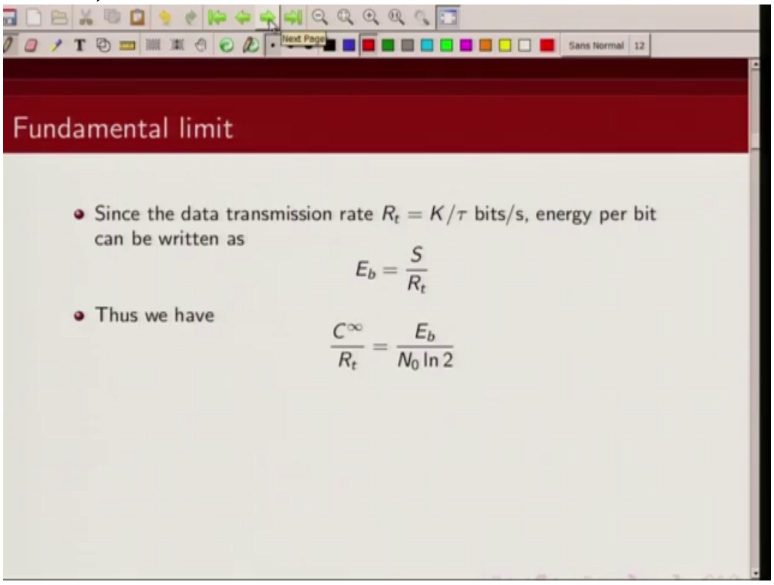
k by tau is R t. So we can write energy per bit in terms of signaling power

(Refer Slide Time 02:58)



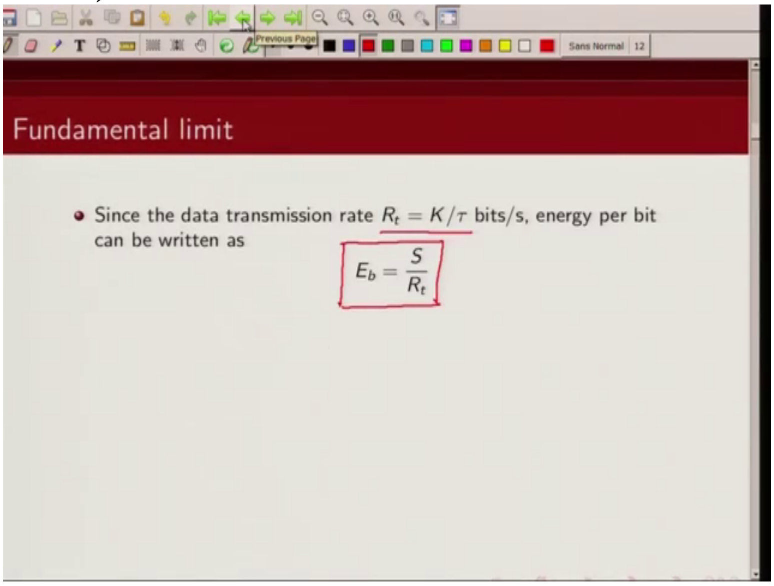
and transmission rate R of t. So

(Refer Slide Time 03:03)



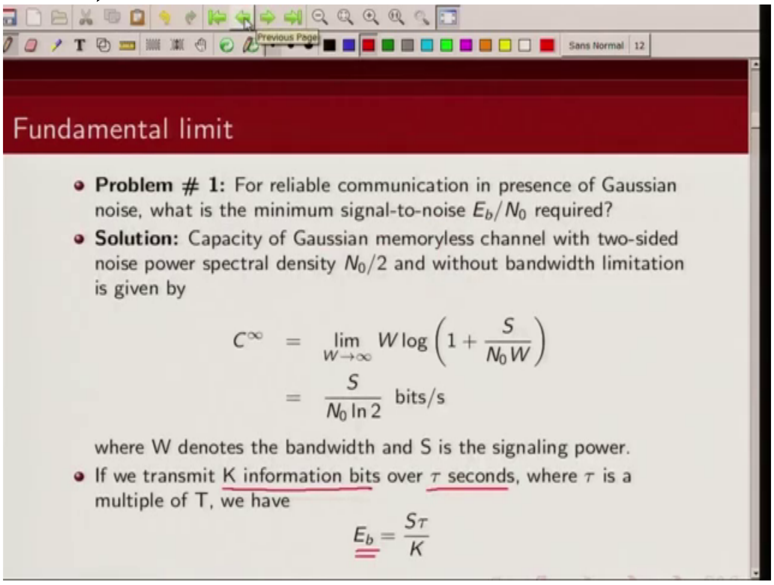
if we divide our

(Refer Slide Time 03:05)



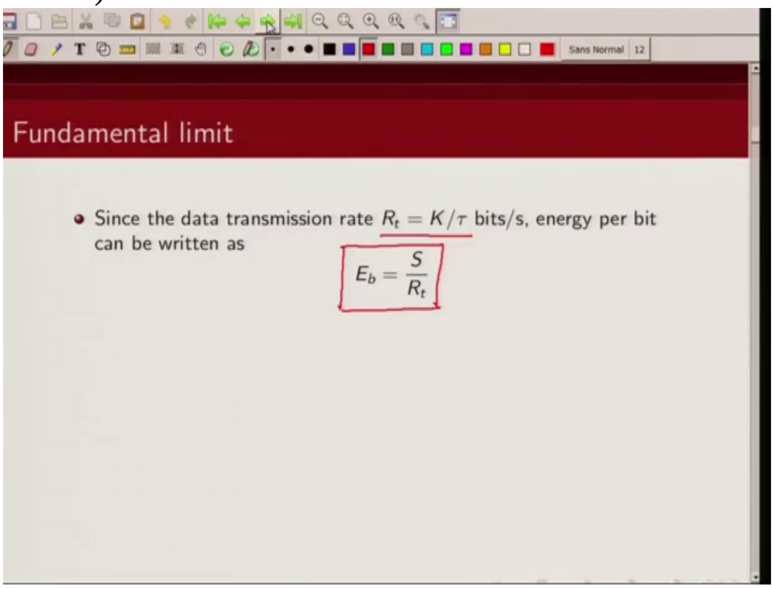
expression

(Refer Slide Time 03:07)



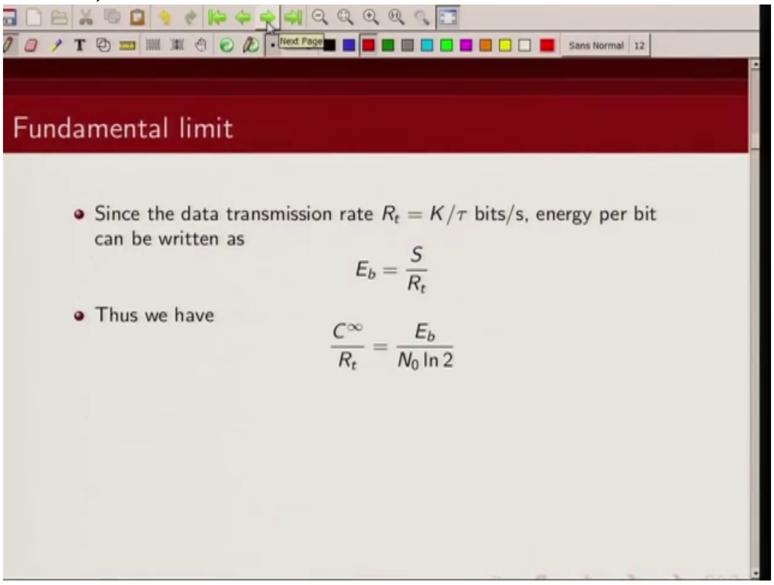
for channel capacity by R t what we

(Refer Slide Time 03:11)



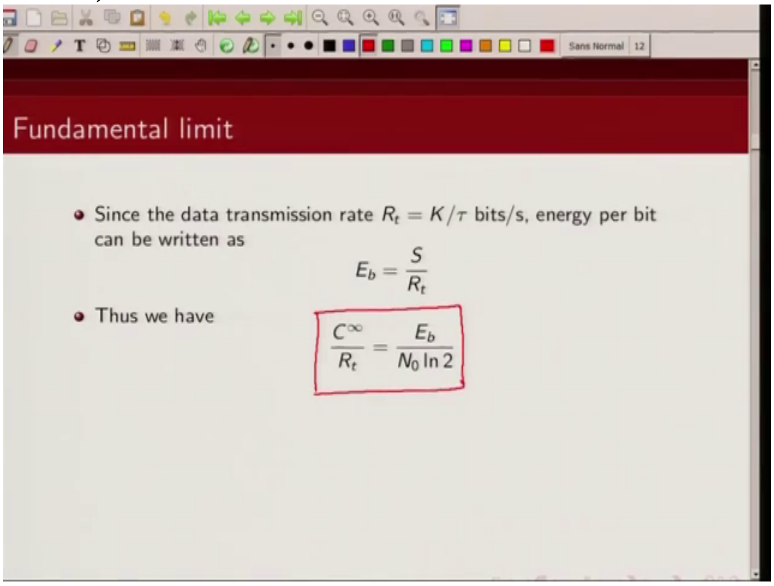
get is this

(Refer Slide Time 03:13)



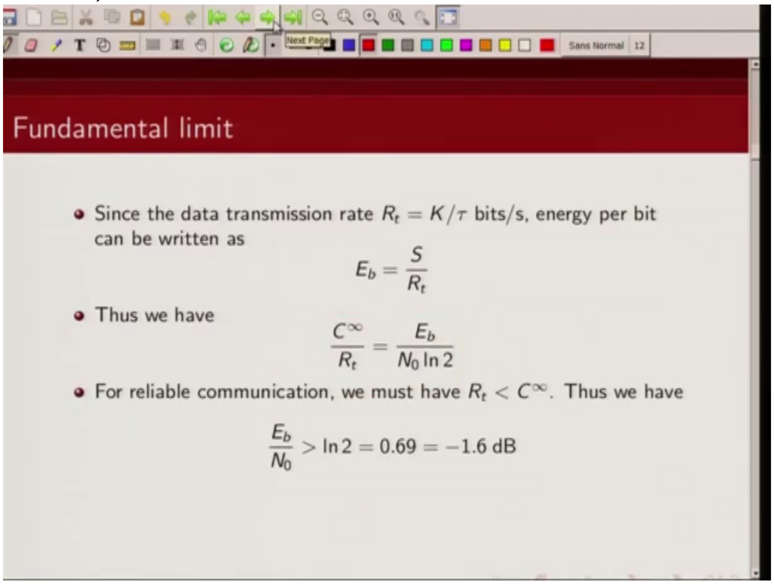
expression.

(Refer Slide Time 03:18)



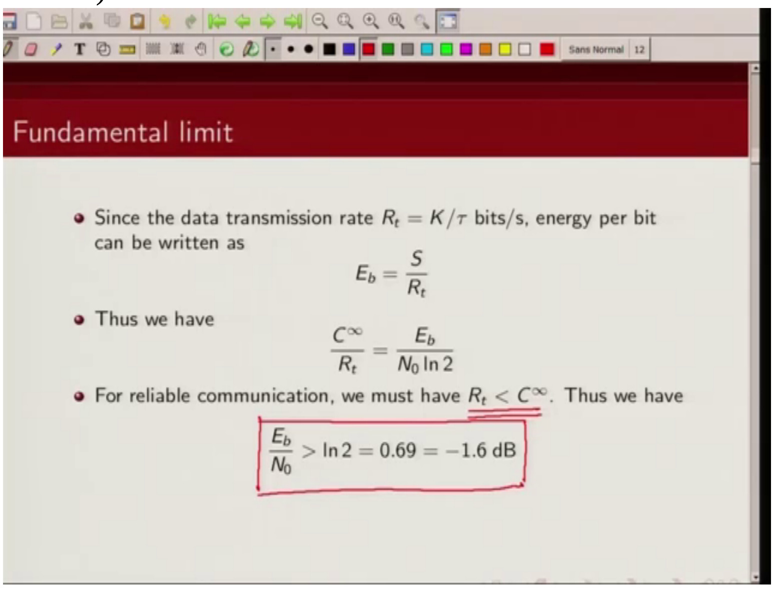
Now we know from Shannon Noisy channel coding theorem that as long as transmission rate is less than channel capacity we can reliably communicate over the communication channel. So we want R t to be less than this channel capacity.

(Refer Slide Time 03:40)



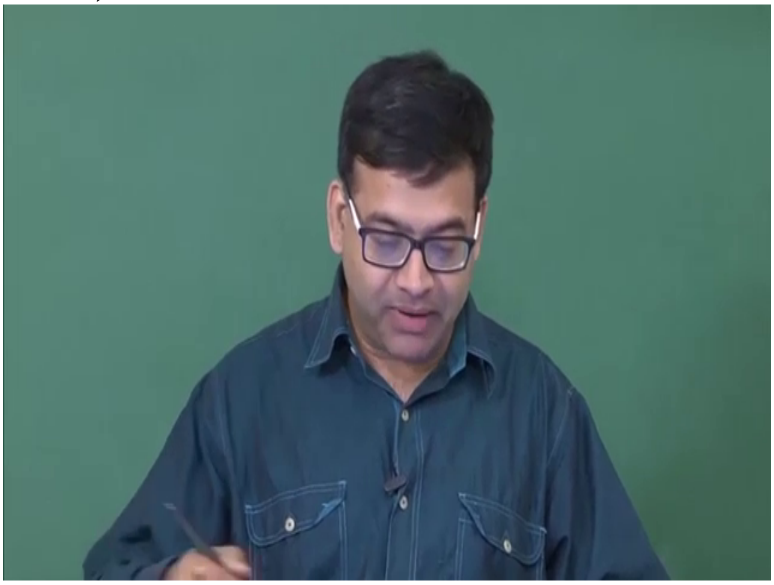
So if we want this to hold then we get a condition on signal to noise ratio. So we get this following condition on

(Refer Slide Time 03:53)



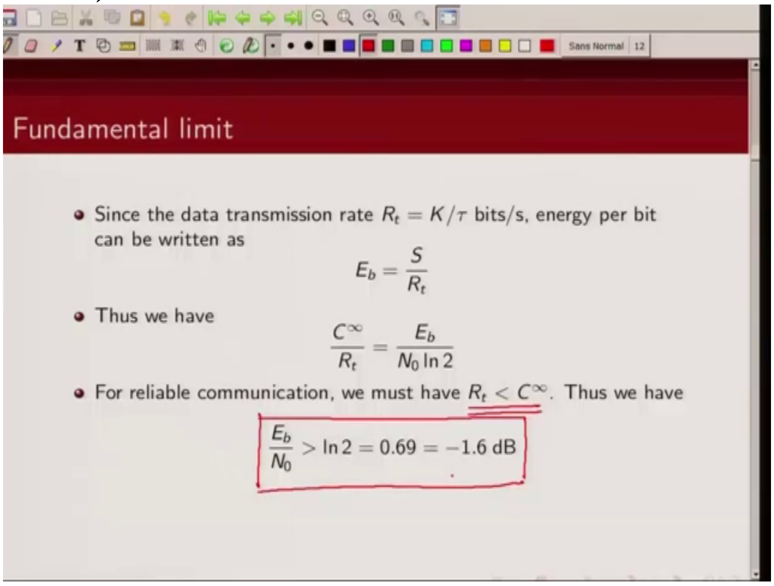
energy per bit. So s n R per information bit, we get this condition that s n R per information bit should be greater than 1, minus 1 point 6 D B. This is for the case when we have infinite

(Refer Slide Time 04:12)



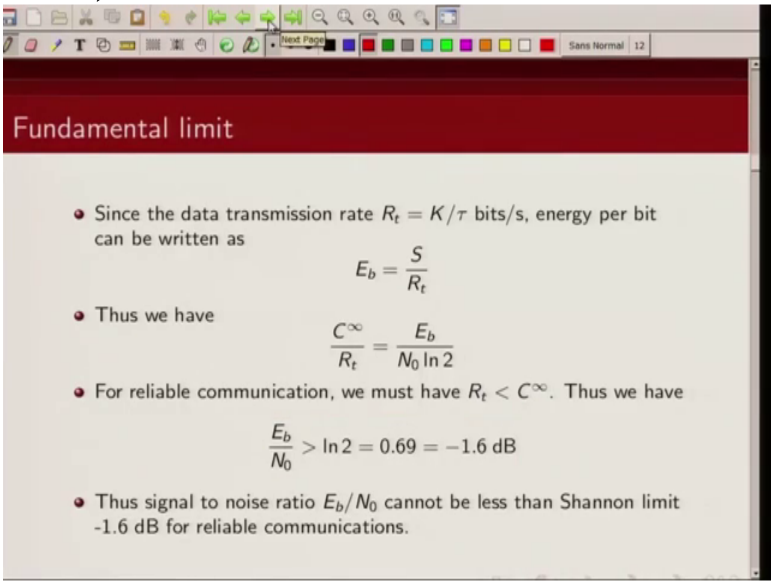
bandwidth and our code rate

(Refer Slide Time 04:14)



can actually go to zero. So

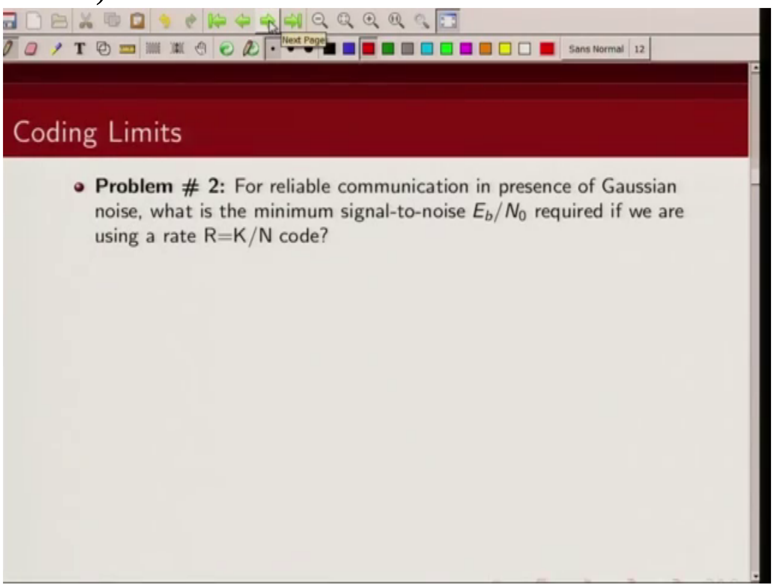
(Refer Slide Time 04:17)



signal to noise ratio then cannot be less than this limit which is minus 1 point 6 D B.

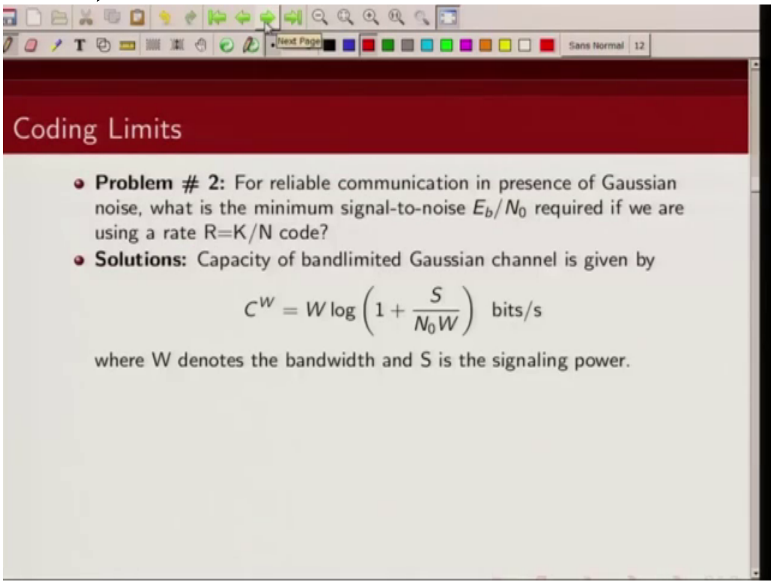
OK let's look at next problem. So

(Refer Slide Time 04:34)



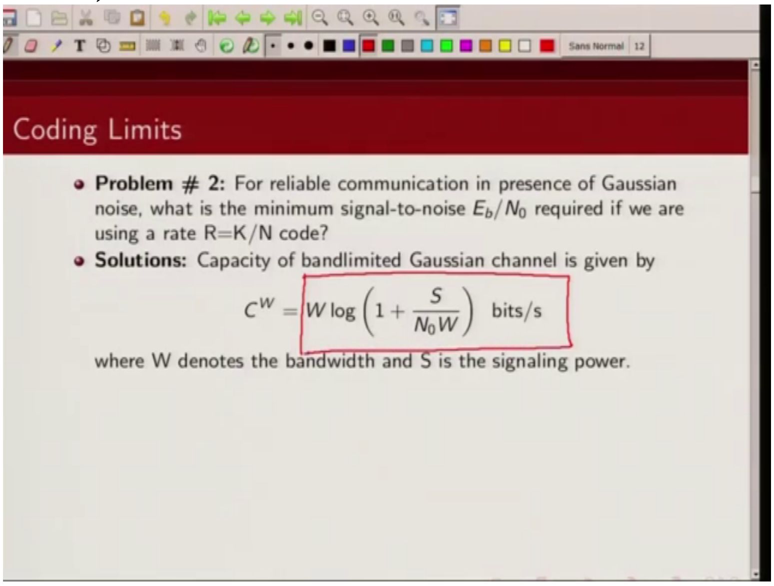
we are interested in reliable communication over additive white Gaussian noise channel and we are transmitting using a rate k by n code. So what is the affect of the

(Refer Slide Time 04:49)



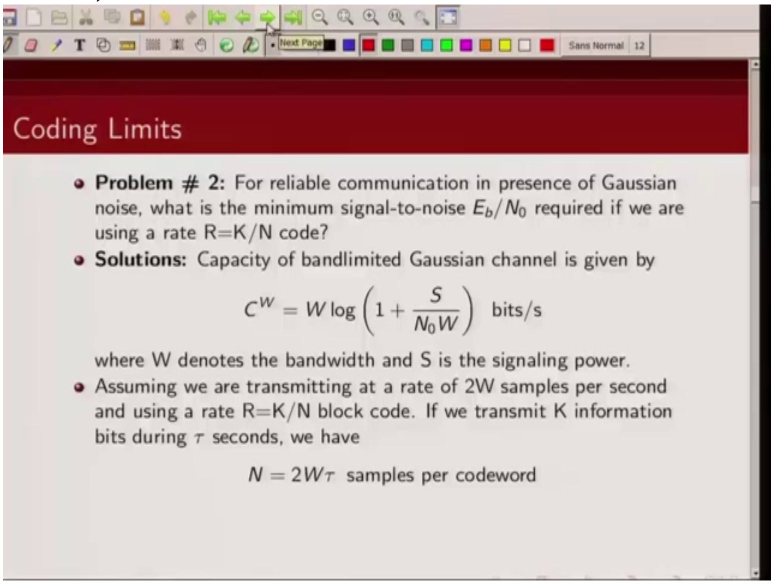
bandwidth w? Now we know for a band limited channel, the capacity is given by this expression.

(Refer Slide Time 05:00)



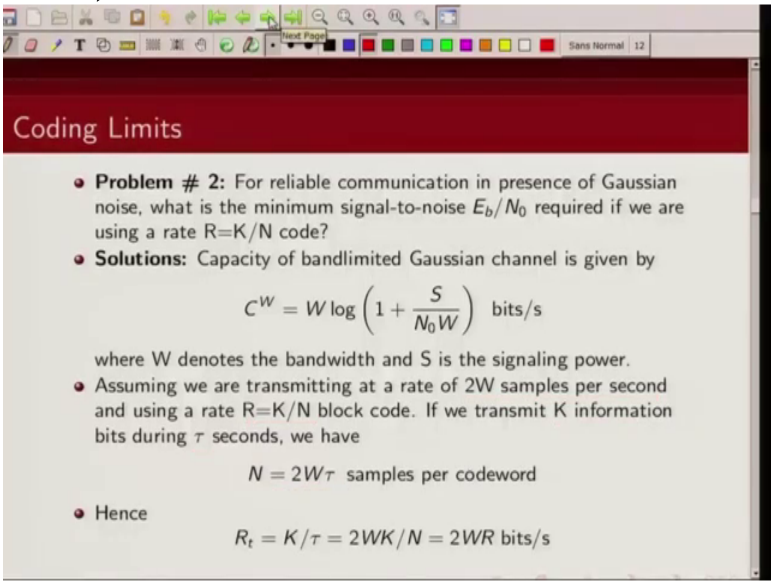
So

(Refer Slide Time 05:03)



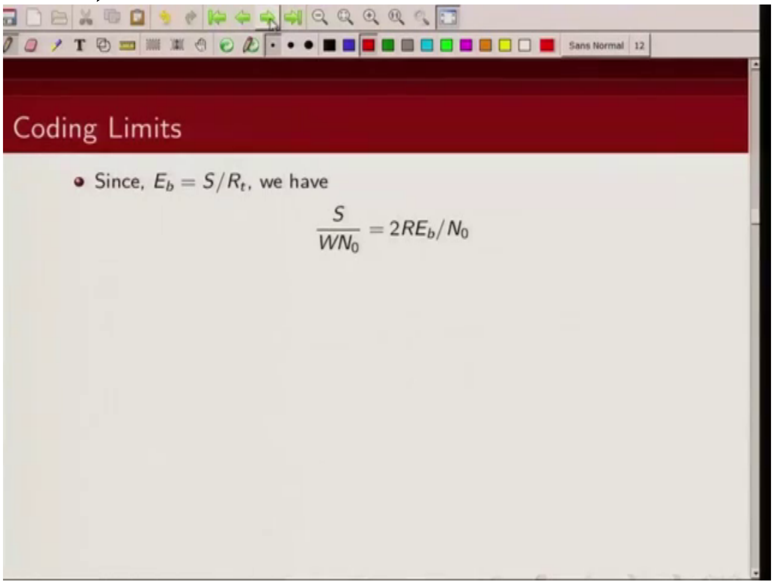
let's say we are transmitting at Nyquist rate. So we are transmitting at 2 w samples per second. And we are using a rate k by n code. So if we transmit k information bits during time tau then number of samples per codeword we are sending is 2 w tau.

(Refer Slide Time 05:32)



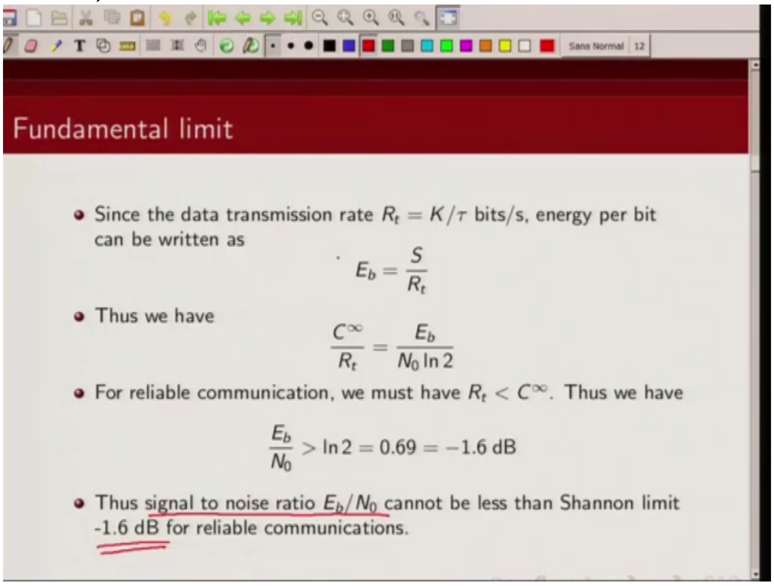
Now our transmission rates is k information bits over tau time. So our transmission rate is k by tau and this we can write in terms of these samples per codeword n which is given by this expression and k by n is nothing but our code rate. So this transmission rate is given by 2 w R where R is my code rate. So this many bits per second is my transmission rate. Now

(Refer Slide Time 06:09)



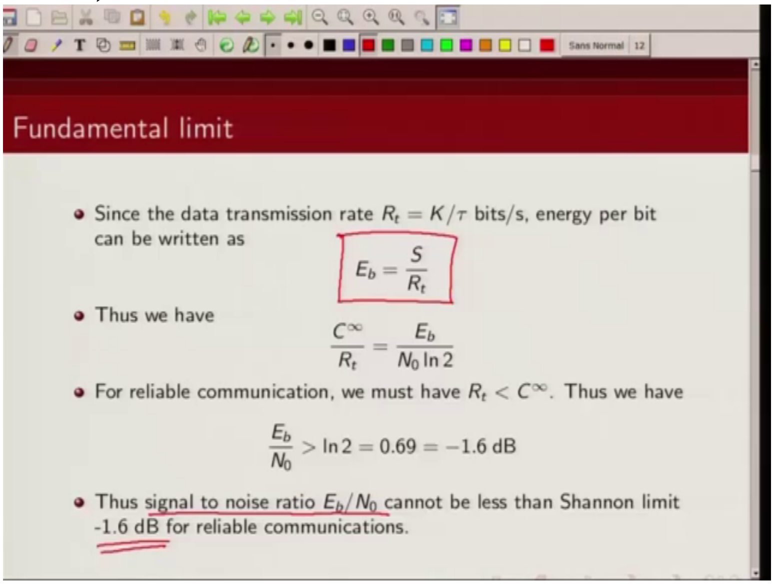
we know that energy per bit is given by signaling power by transmission rate, this we have done in the last example.

(Refer Slide Time 06:25)



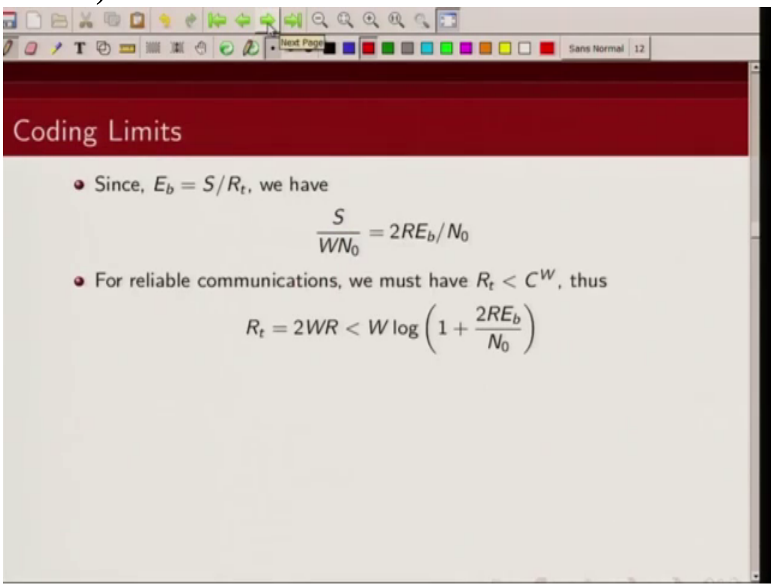
We had this, right

(Refer Slide Time 06:28)



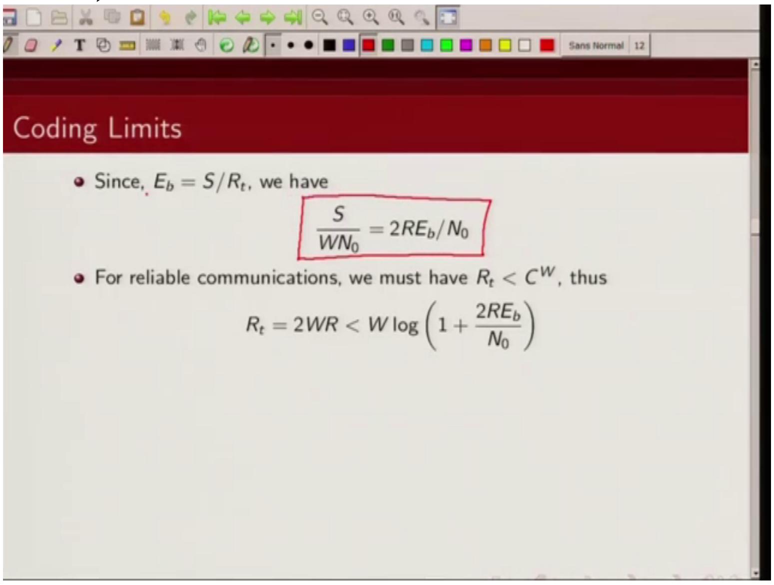
so using this, then

(Refer Slide Time 06:33)



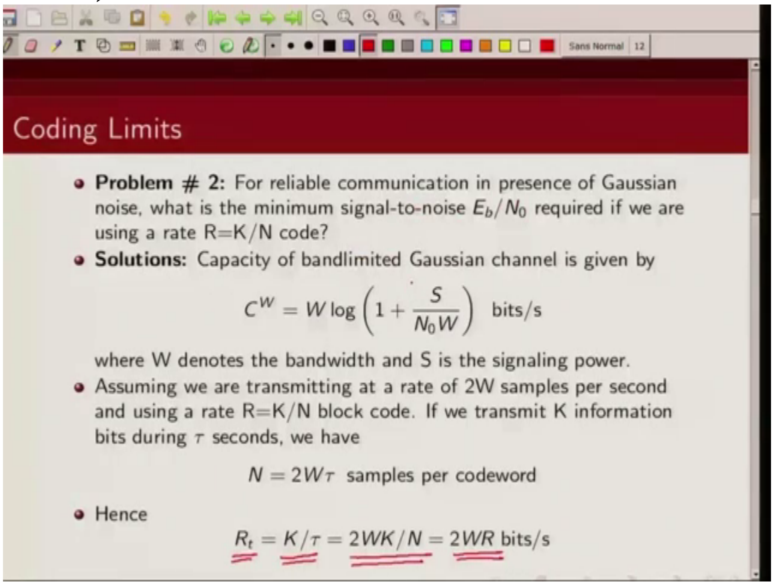
so we do that we can write s by w n naught in terms of

(Refer Slide Time 06:41)



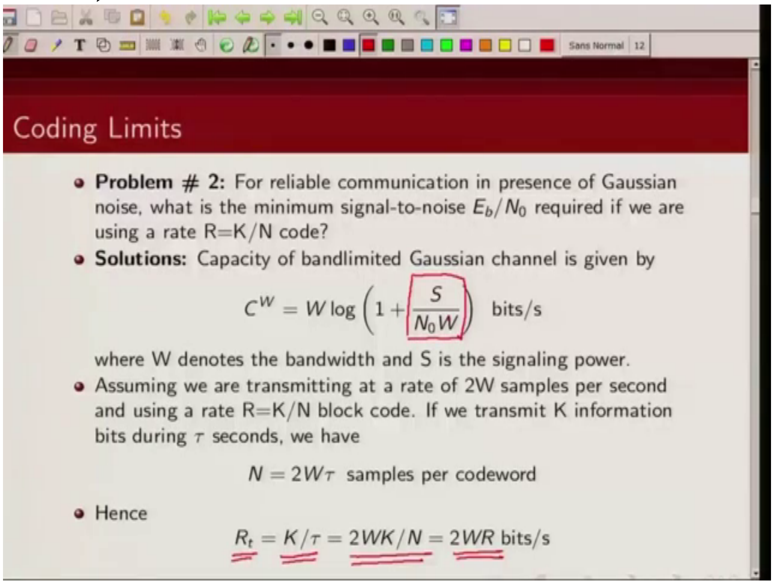
E b by N naught, this is equal to 2 times code rate by, into s n R per information bit. So in this expression of

(Refer Slide Time 06:55)



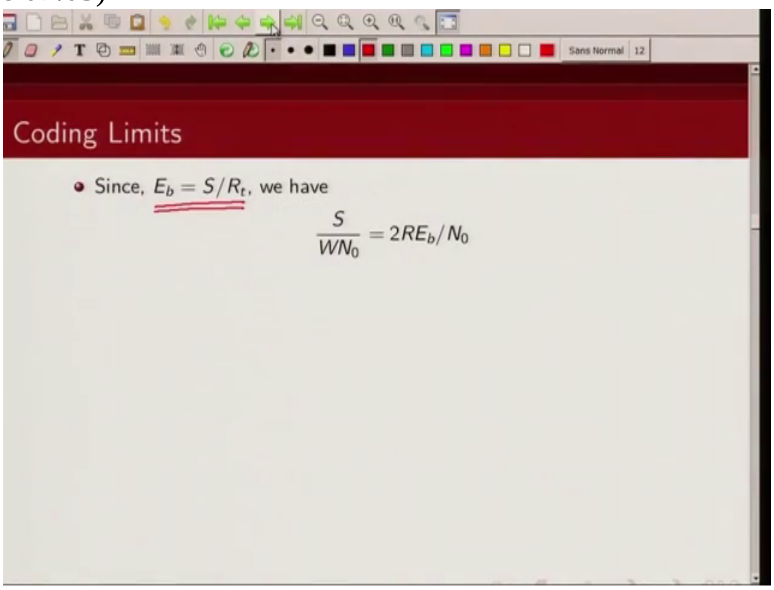
capacity of band limited channel, we can replace this

(Refer Slide Time 07:01)



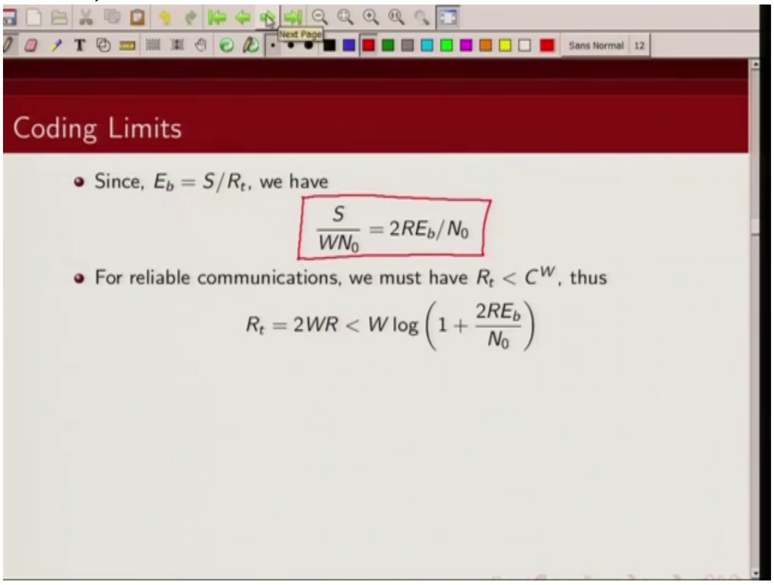
expression by

(Refer Slide Time 07:03)



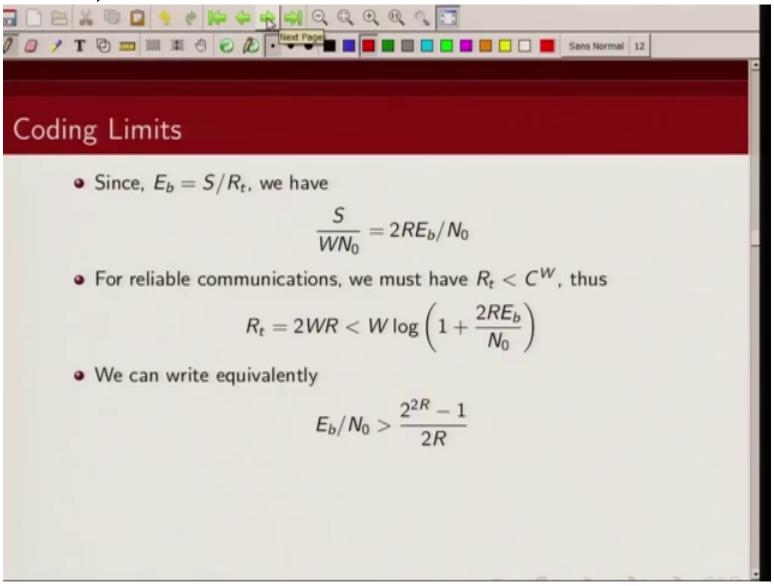
this, fine? Now

(Refer Slide Time 07:11)



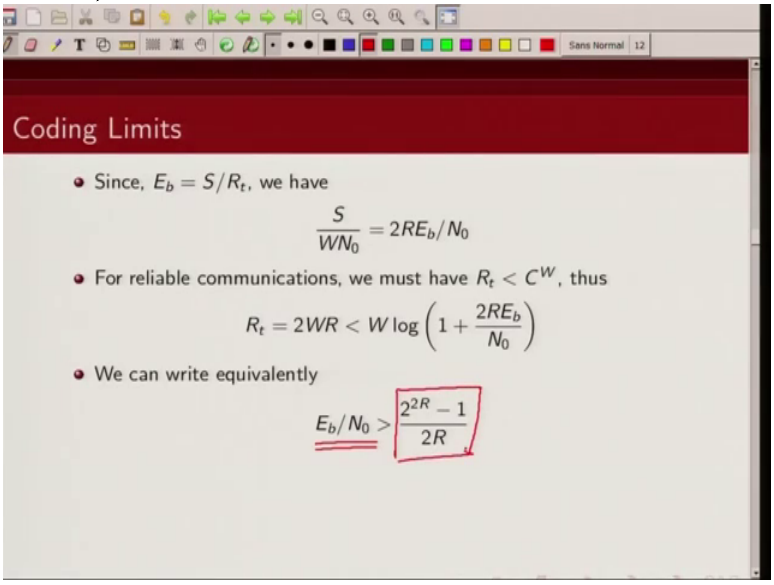
for reliable communication we know our transmission rate should be less than channel capacity. And what's channel capacity? That's for the band limited channel; it is w log of 1 plus s by 2 N naught which is basically given by this expression. So transmission rate which is 2 times w R should be less than w log of 1 plus 2 R E b by N naught.

(Refer Slide Time 07:43)



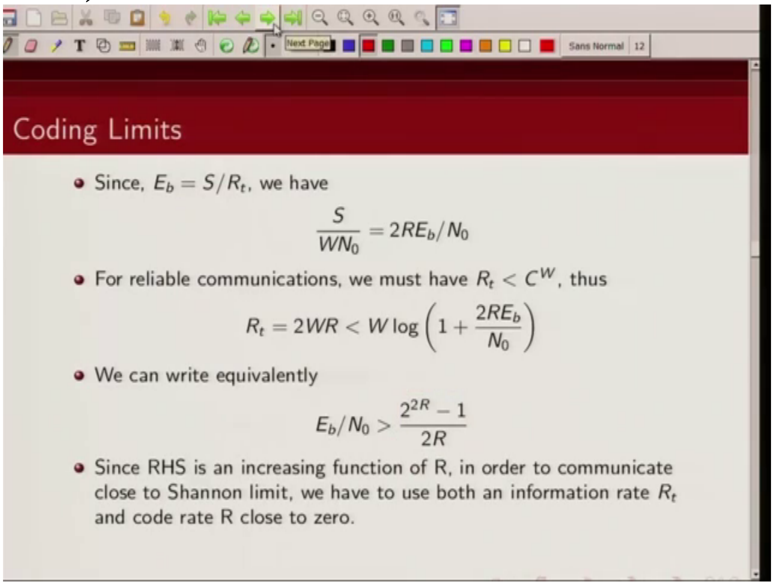
So from here then we can write down the expression for minimum s n R information bit, energy per information bit by noise power density so s n R per information bit we can write it as, this should be greater than equal to 2 raised to power 2 R minus 1 divided by 2 R. Now this right hand side term is an increasing function of R.

(Refer Slide Time 08:16)



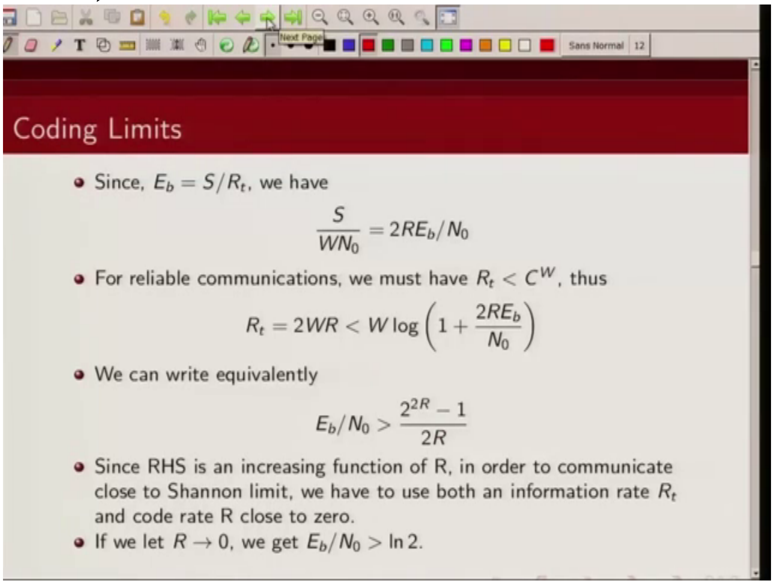
of R. So

Refer Slide Time 08:17)



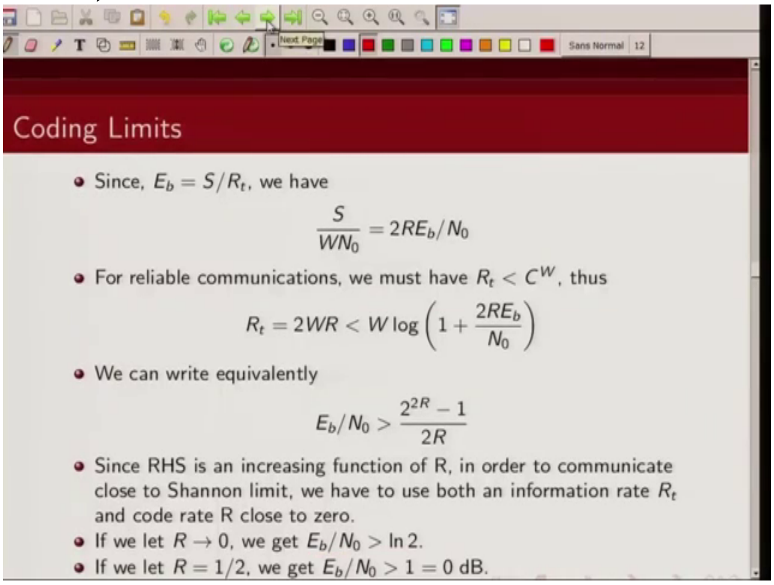
So if you want to communicate close to the channel limit then your information rate R as well as code rate R should be close to zero. In fact if you

(Refer Slide Time 08:33)



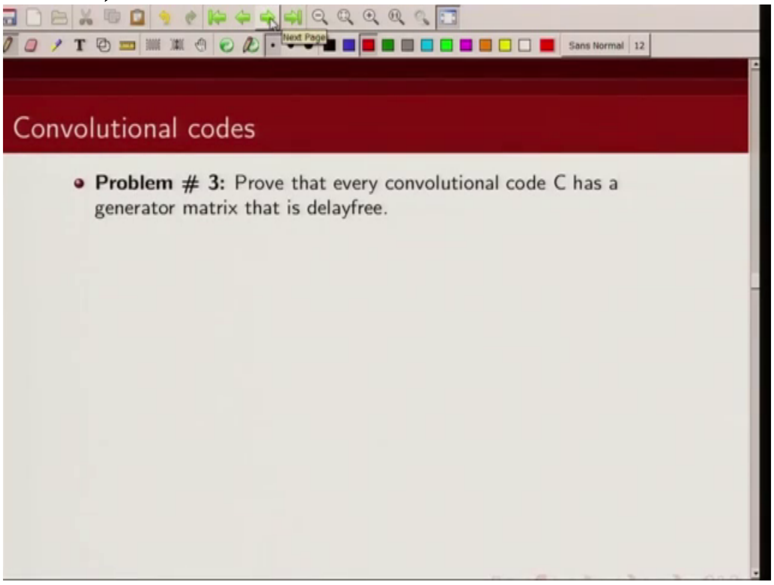
if you let R go to zero, what you will see is you will get the limit that we had just talked about in the previous problem which was minus 1 point 6 d B and let's plug in some practical values of R. Let's say R is half,

(Refer Slide Time 08:53)



we put value of R to be half then e b by n naught should be more than 0 d B. So you can see for any rate which is away from zero then this minimum s n R required for transmission is also more, Ok.

(Refer Slide Time 09:16)

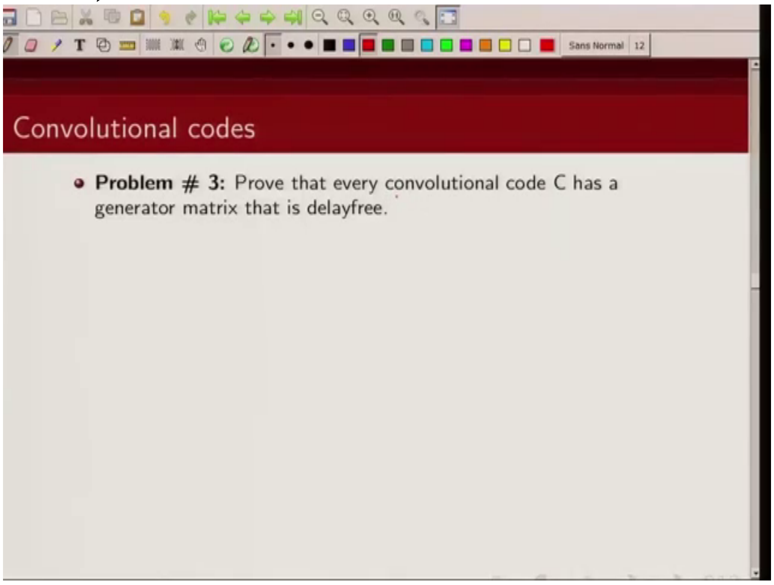


Now the next question that we

(Refer Slide Time 09:19)

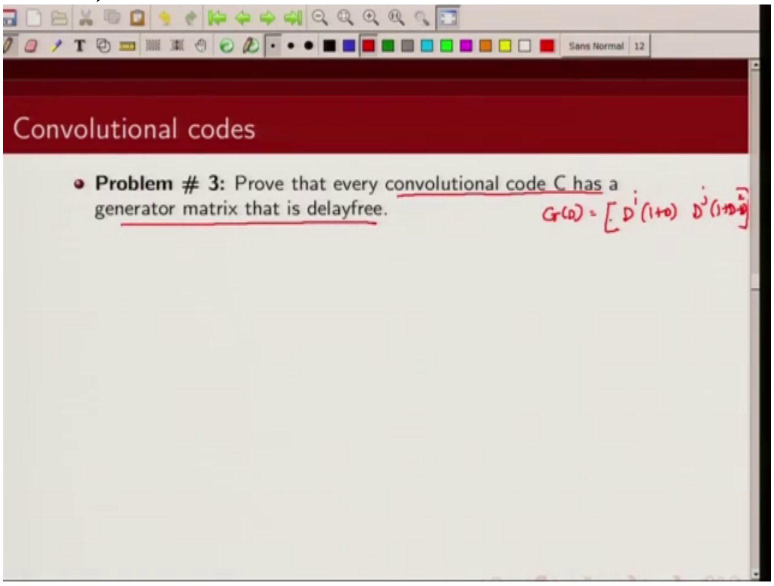
will solve is to prove that

(Refer Slide Time 09:22)



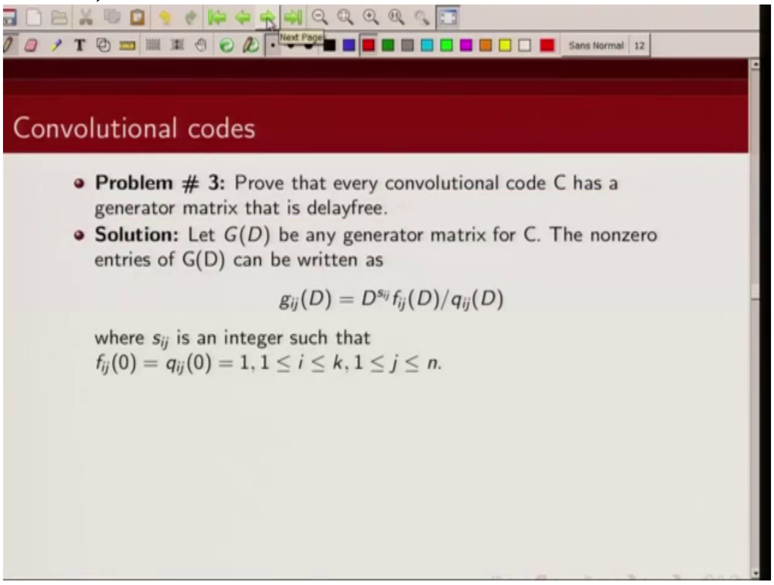
for every convolutional code has a generator matrix which is delay free. Now delay free meaning basically we, if we have a generator matrix let's say of the form D i and then something like 1 plus D and some D j, 1 plus D plus D square or something like that, we can always write

(Refer Slide Time 09:48)



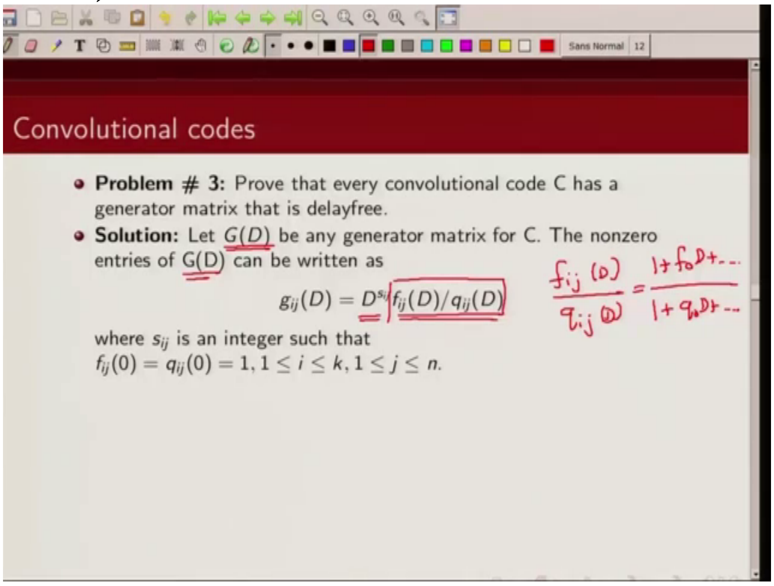
equivalent generator matrix which will be free of these, like delay terms. I will talk about that.

(Refer Slide Time 09:56)



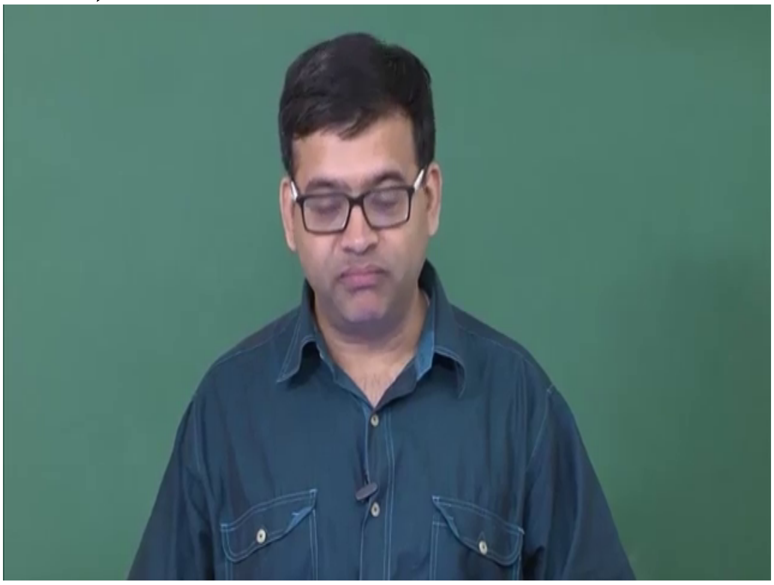
So let's say we have a generator matrix G of D, Ok. And its non zero entries can be written in this particular form. So there is some delay term, we are calling it D s t raised to power s i j. And then we have this rational function. We have this rational function which is f i j D by q i j D. And f i j D and q i j D is of the form 1 plus some f naught D plus dah dah dah and this is one plus q naught D plus something something,

(Refer Slide Time 10:42)



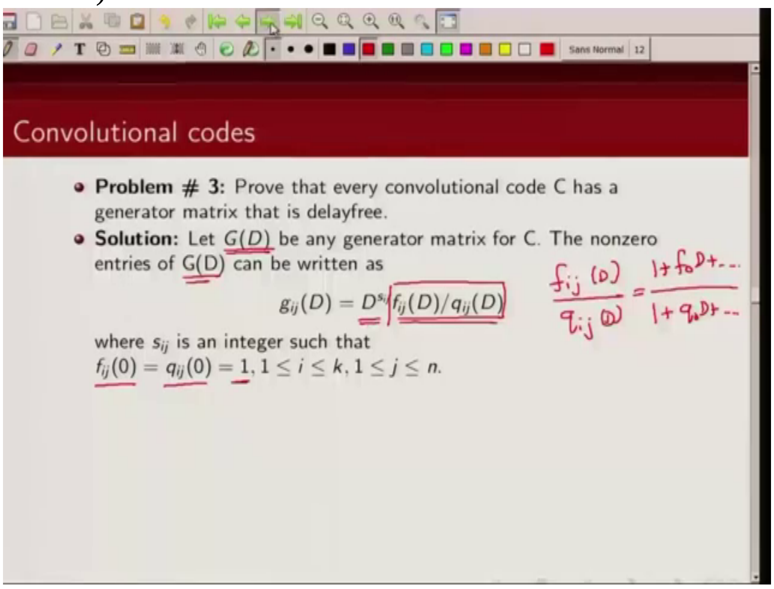
Ok. So that's what I meant f 0 and q y j zero is 1.So any entry in the generator matrix can be written in this form,

(Refer Slide Time 10:54)

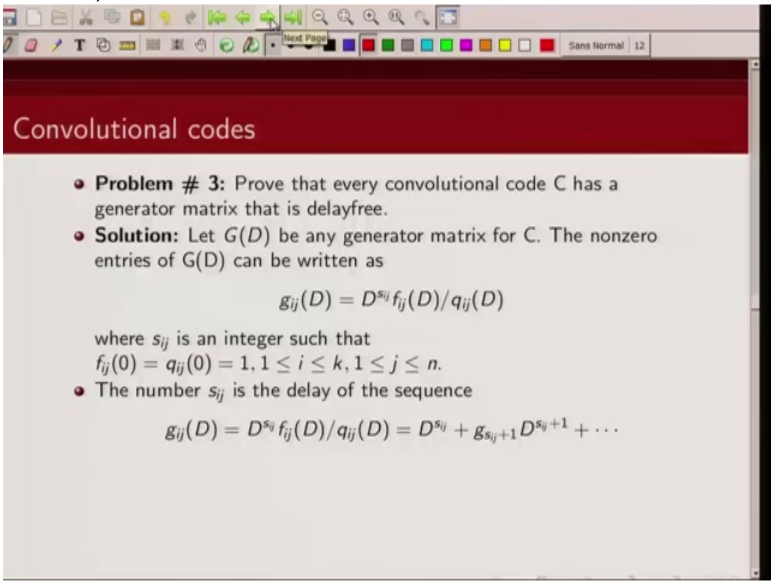


Ok. Now

(Refer Slide Time 10:58)

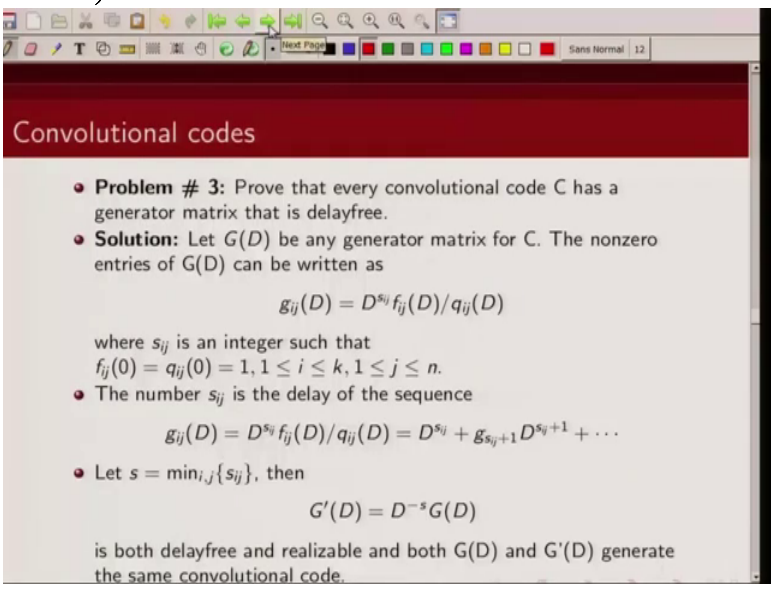


(Refer Slide Time 10:59)



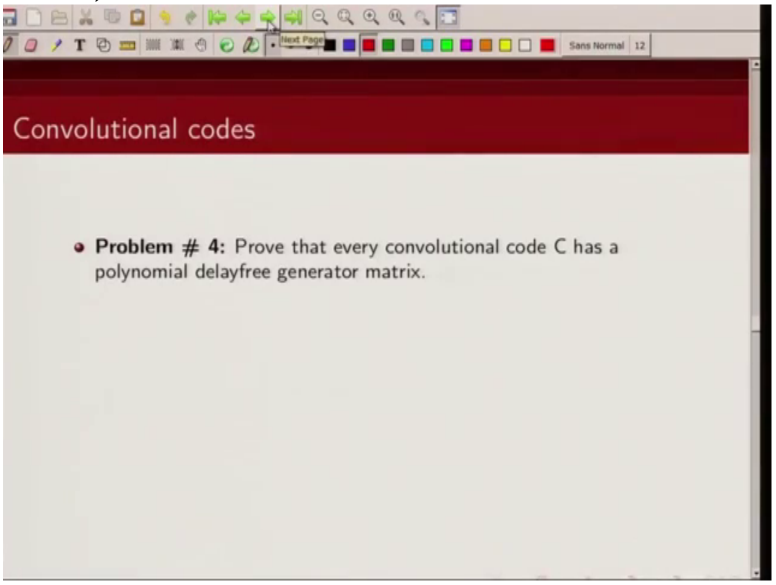
what is this term s i j? Essentially it's a delay term. So if we have some term like s i j so what you will get is terms of this particular form. Now

(Refer Slide Time 11:12)



if we consider s which is the minimum s i j over all i and j, then we can find an equivalent generator matrix which would be G dash of D which we can write as G minus s G D and this G i D will be delay free and it is of course realizable and it also generates the same set of codewords. So whenever we have a generator matrix which has a delay term we can always find an equivalent generator matrix which is delay free.

(Refer Slide Time 11:55)

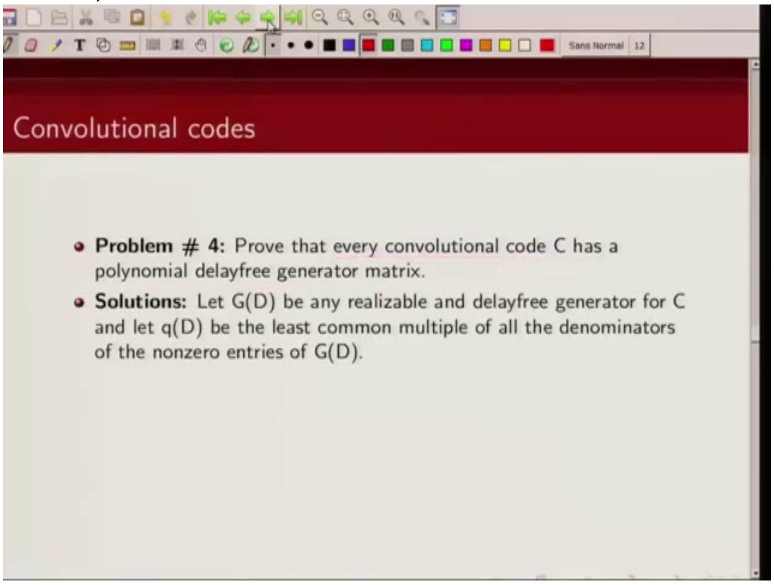


Next question, prove that every convolutional code has a polynomial delay free generator matrix. So for any convolutional code,

(Refer Slide Time 12:09)

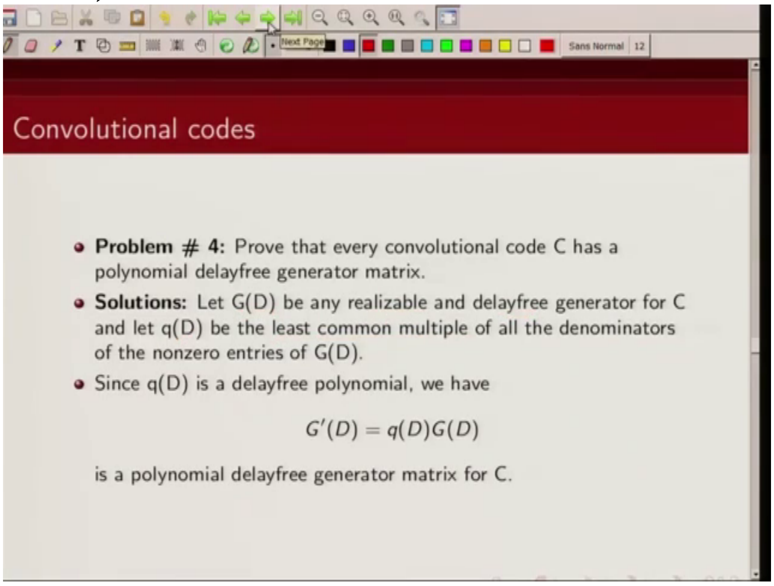
we have an equivalent polynomial delay free generator matrix.

(Refer Slide Time 12:15)



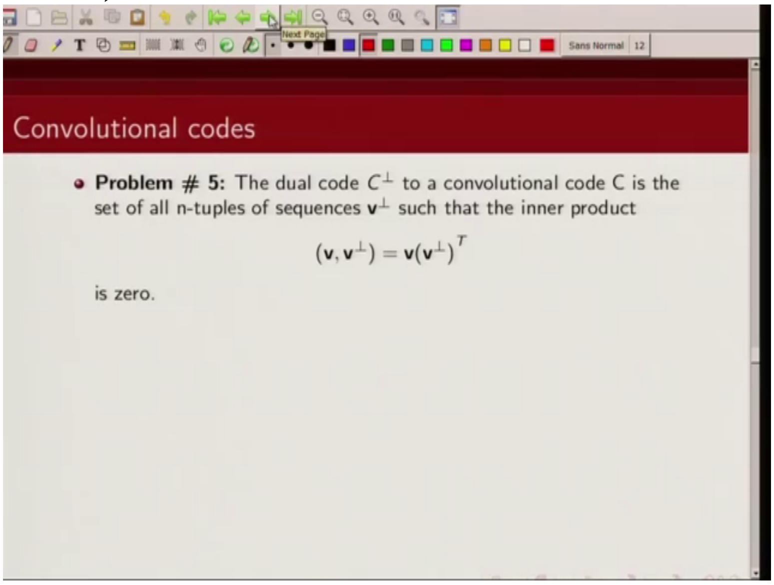
So let us say we have a generator matrix which is given by G D and this be any realizable and delay free generator for this convolutional code c and let q be the least common multiple of all the denominators of the non zero entries of G D. So G D basically has rational form and q D is the l c m of the denominator terms of the non zero entries of G D. Now if we have such

(Refer Slide Time 13:02)



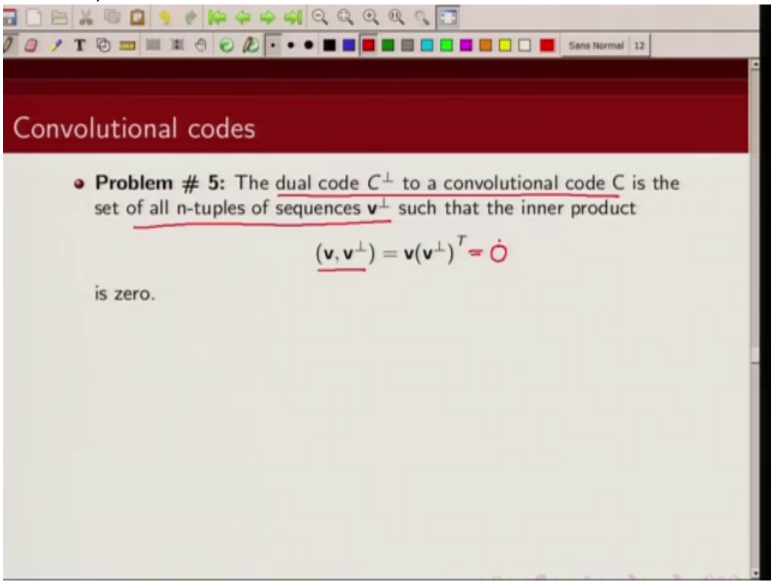
an, and we will have a q D which is basically delay free. So if we multiply our original generator matrix by this q D we will end up with a new generator matrix G dash of D which will be polynomial and delay free.

(Refer Slide Time 13:26)



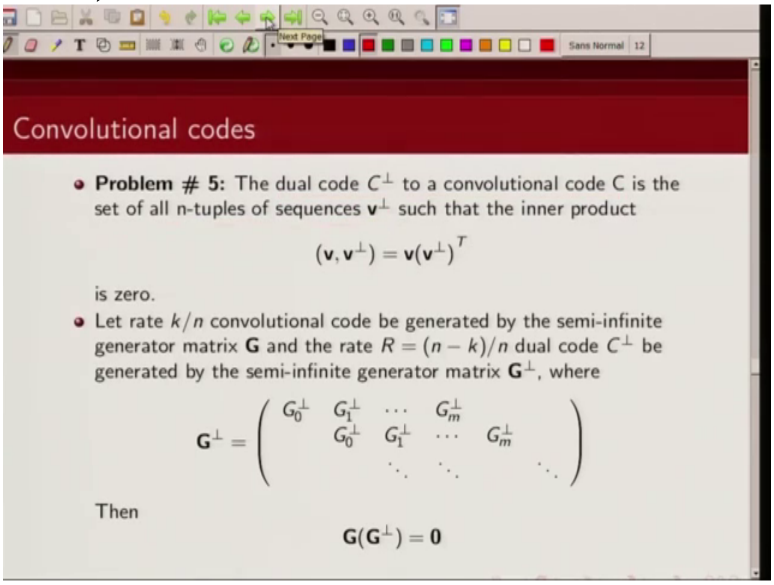
Ok, so let's first define a dual of a convolutional code. So we define a dual of a convolutional, convolutional code c as a set of all end tuples of sequence v dual such that the inner product between these which is defined by v and v dual transpose, this is essentially zero. So

(Refer Slide Time 13:59)



if you have a dual to the original convolutional code then if you take set of code n tuple sequence from the original code and the dual code their inner product will be zero.

(Refer Slide Time 14:14)

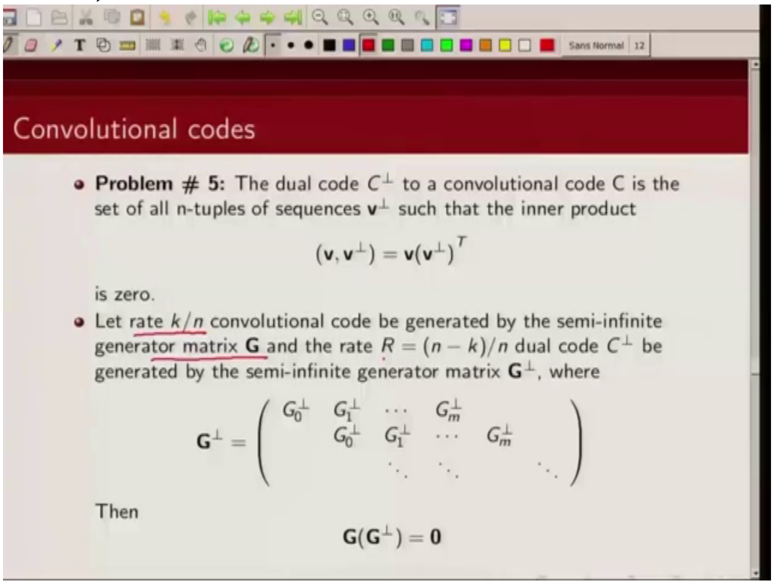


So let us define a rate k by n convolutional code which is generated by a generator matrix G and you know that we can write a generator

(Refer Slide Time 14:26)

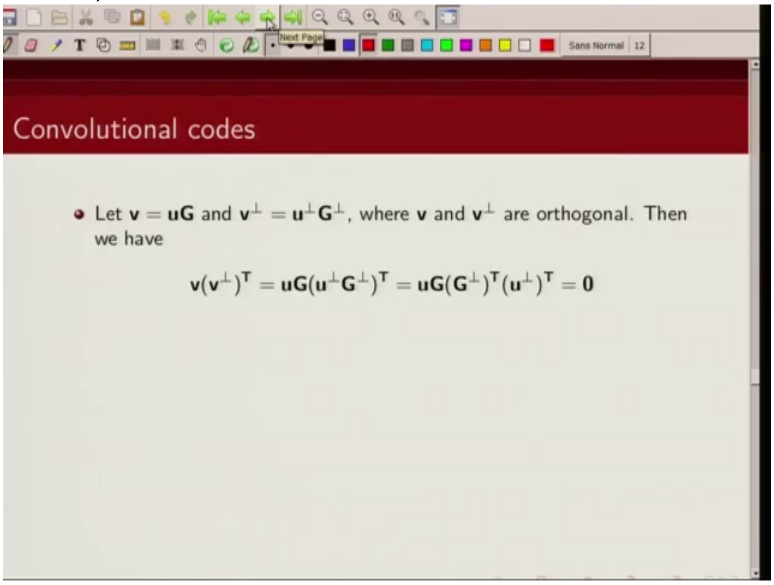
matrix of convolutional code in a semi infinite fashion. And let's say

(Refer Slide Time 14:33)



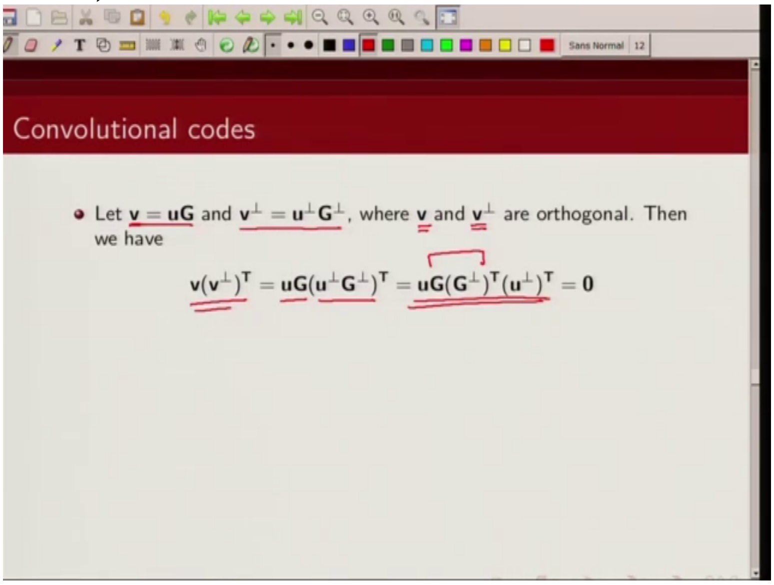
the dual code is this which has rate n minus k by n and it is generated by this semi infinite generator matrix then show that G G transpose basically will be zero where as G is the generator matrix of the original code and this is the generator matrix of the dual code.

(Refer Slide Time 15:05)



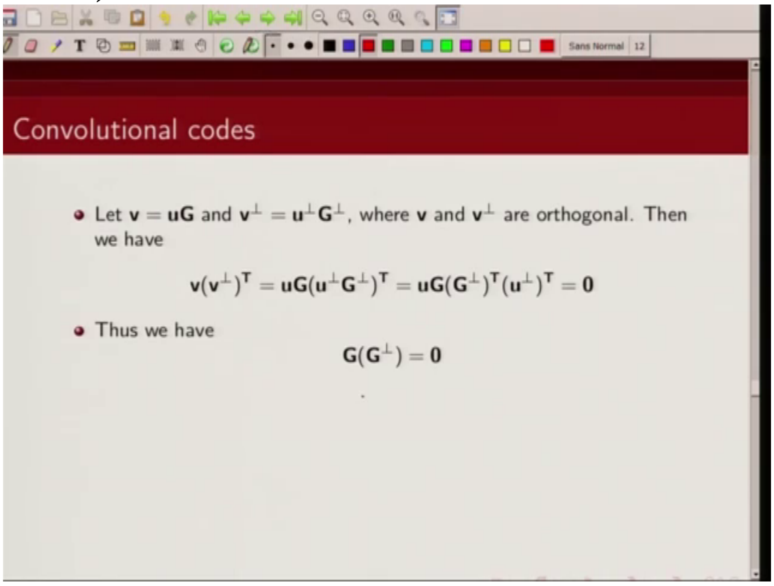
So how does this follows? So since v v transpose is 0, so what is v? v is u times j. And v, and v dual is u dual G dual. Now we know that they will be dual of each other if v v dual basically is orthogonal. So if this inner product is 0, now if inner product is zero, so then u G and this transpose should be zero. So this we can write as u G G transpose. Now this term will be zero only if this term is zero

(Refer Slide Time 15:53)



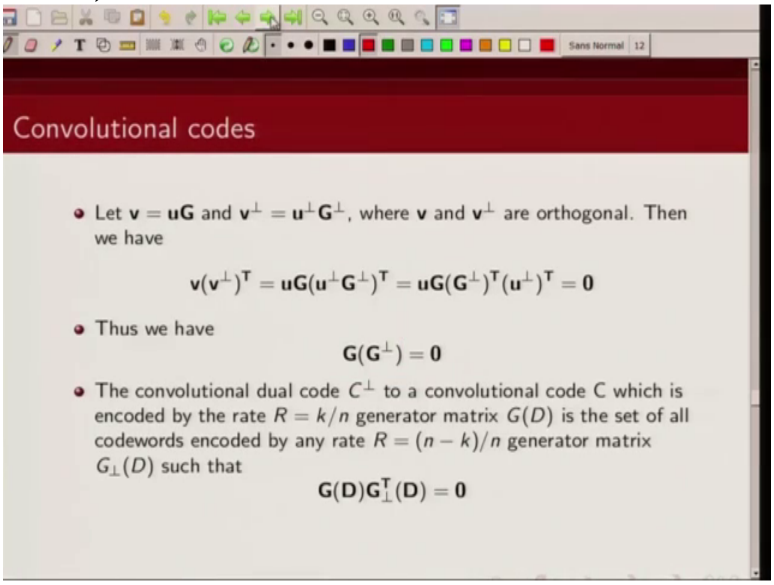
for all u, Ok and hence

(Refer Slide Time 16:00)



we get this condition that this should be zero, Ok.

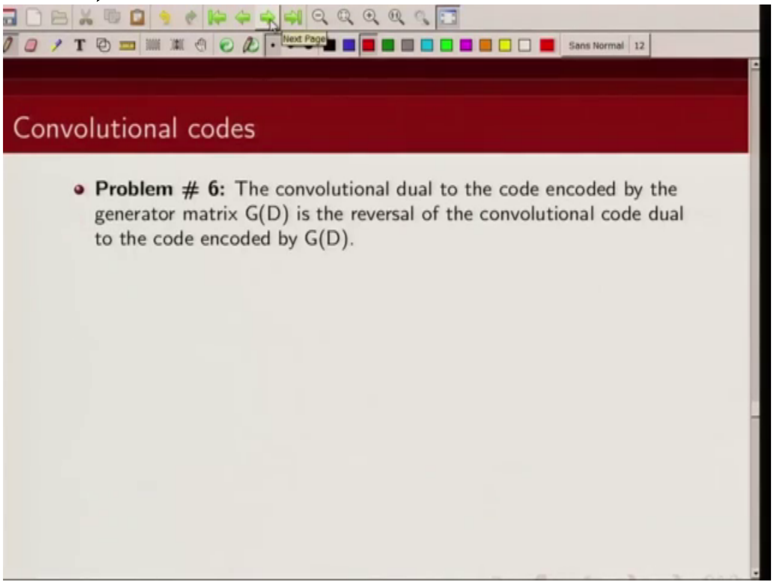
(Refer Slide Time 16:11)



So a convolutional dual to a convolutional code c which is encoded by a rate k by n generator matrix G D is set of all codewords encoded by rate n minus k with generator matrix this such that G D and the generator matrix of the dual should be 0. So the dual of the code is defined by this.

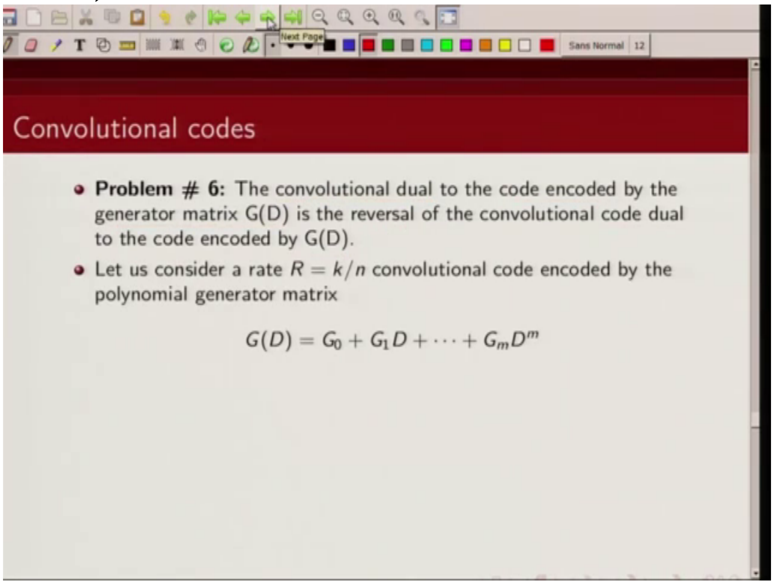
Next,

(Refer Slide Time 16:47)



so here we will show that the convolutional dual to the code encoded by the generator matrix G D is nothing but reversal of the convolutional code dual to the code generated by G of D. So

(Refer Slide Time 17:11)

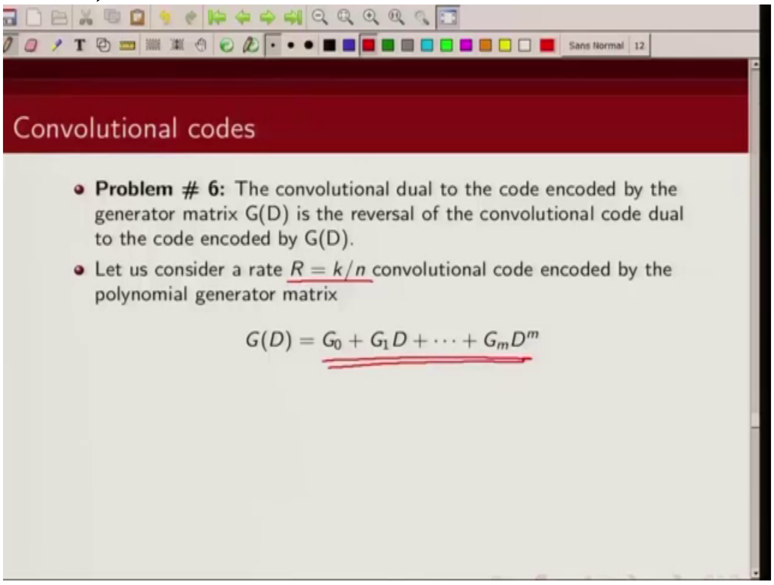


let us consider a rate k by n convolutional code which is encoded by a polynomial generator. We have already shown that we can find an equivalent polynomial delay free representation of any convolutional code, right. So let's see this is our

(Refer Slide Time 17:35)

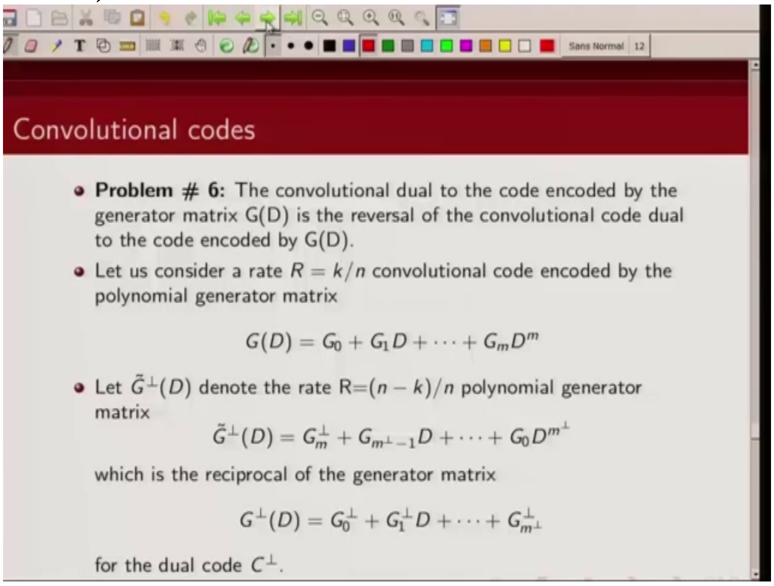
G D for original

(Refer Slide Time 17:37)



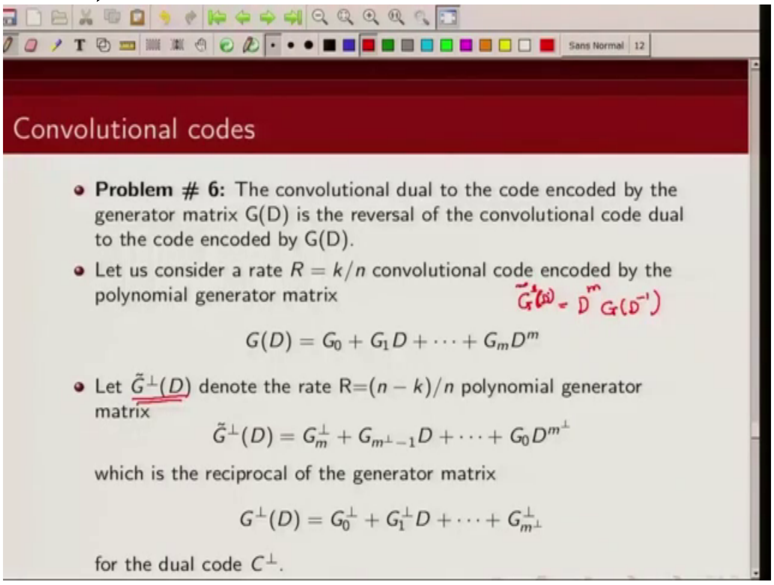
code c.

(Refer Slide Time 17:40)



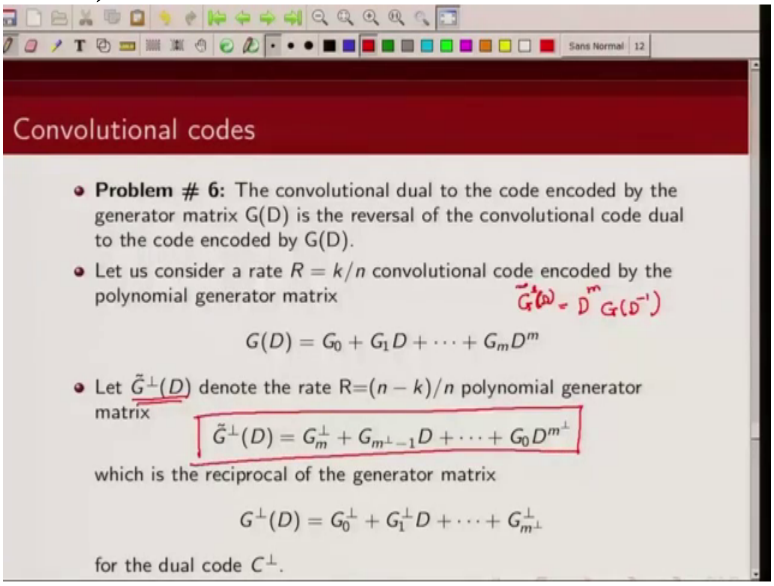
Now we define this as the reciprocal of the generator matrix. Now how do we come up with a reciprocal? So we do G D minus 1 and we multiply by the maximum degree which is, in this case, m. This is how we get the reciprocal. This is how we get the reciprocal, Ok. Now

(Refer Slide Time 18:15)



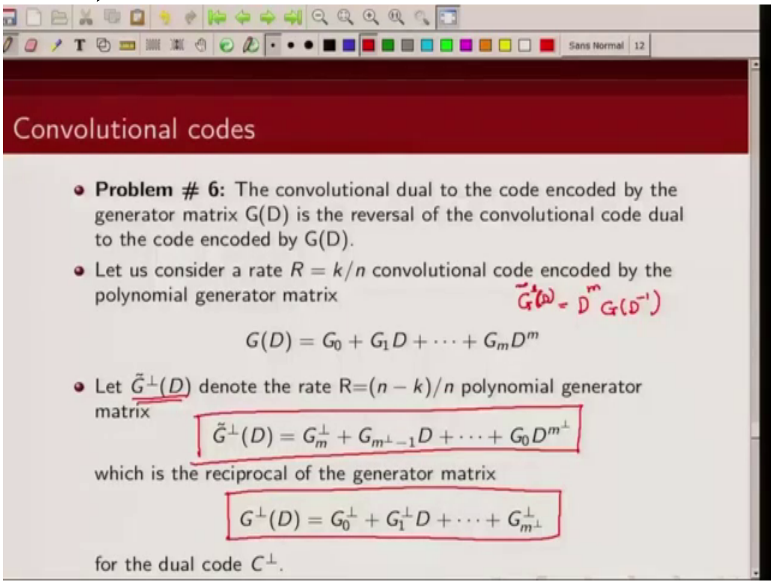
what I have shown you here is the reciprocal of

(Refer Slide Time 18:22)



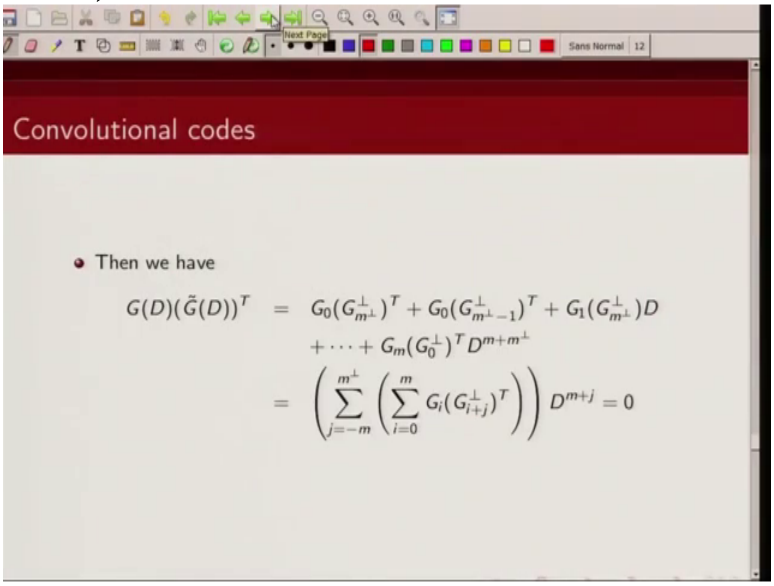
the generator matrix for the dual.

(Refer Slide Time 18:28)



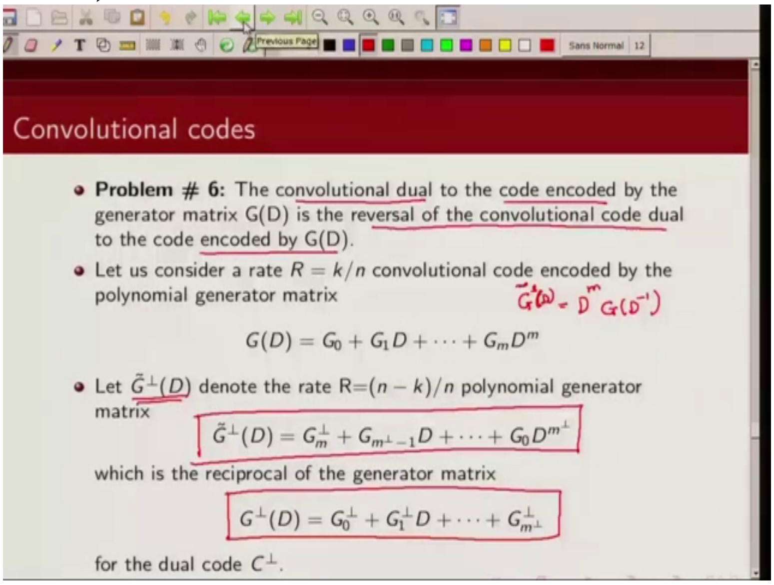
This is the generator matrix of the dual code. And this is the reciprocal of the generator matrix of the dual code. Now note we have to show that the convolutional dual to the code encoded by G D is nothing but the reversal of the convolutional dual coded by G D. So if that's the case then G D, dot product of G D with this should be zero. So

(Refer Slide Time 19:08)



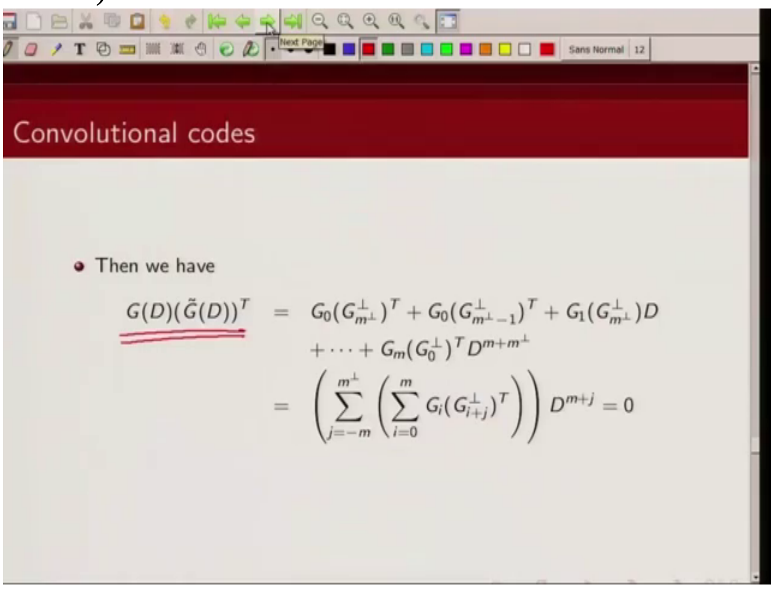
let's try to find G D and transpose of the reversal of the

(Refer Slide Time 19:20)



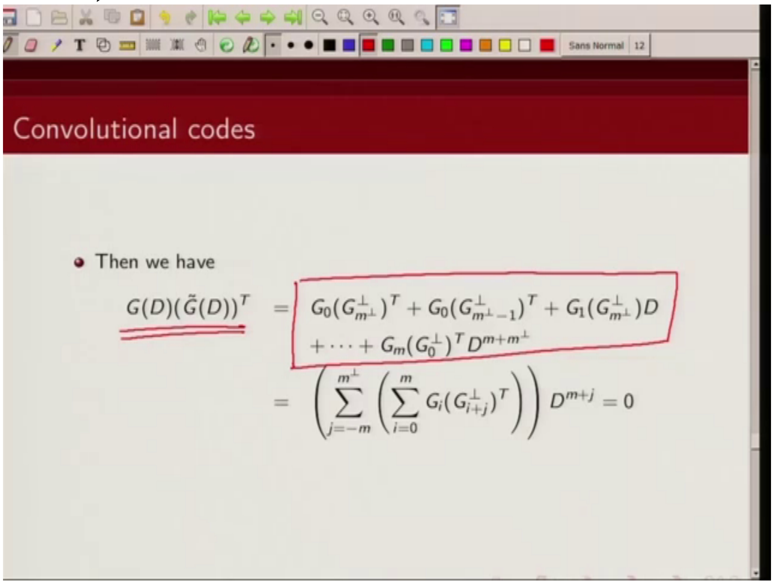
dual matrix. See; again pay attention to the question. Basically what we are saying is the dual of the code encoded by the generator matrix is given by the reversal of the convolutional code dual. So reversal of the convolutional code dual is this, right. So if the dual of the convolutional code is given by reversal of the convolutional code, then what is the property, the, this generator matrix should satisfy? This and this transpose should be zero. So we have to show that G D and this generator matrix transpose is zero.

(Refer Slide Time 20:07)



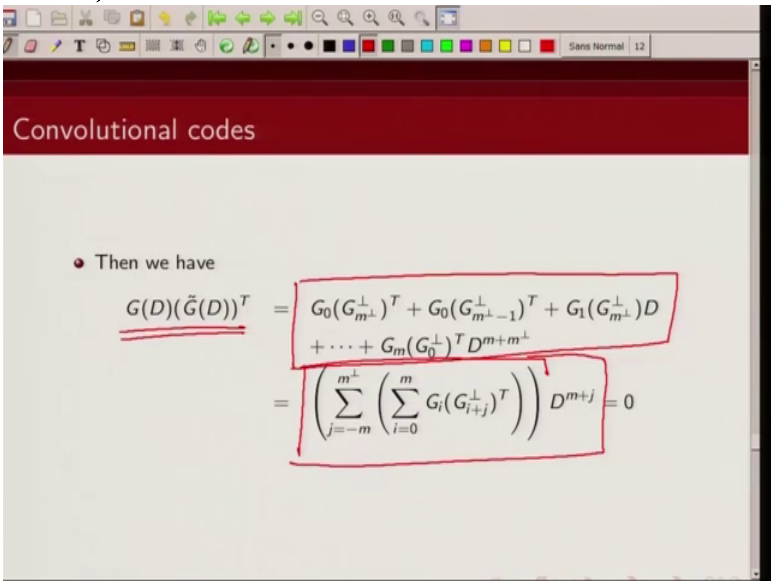
So we do this. So this can be written as this term, Ok?

(Refer Slide Time 20:16)



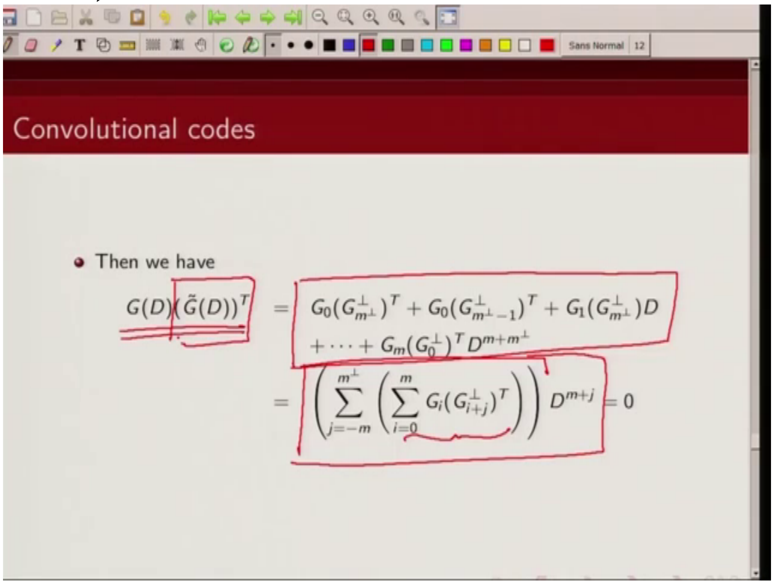
Further we can write this as double summation

(Refer Slide Time 20:24)



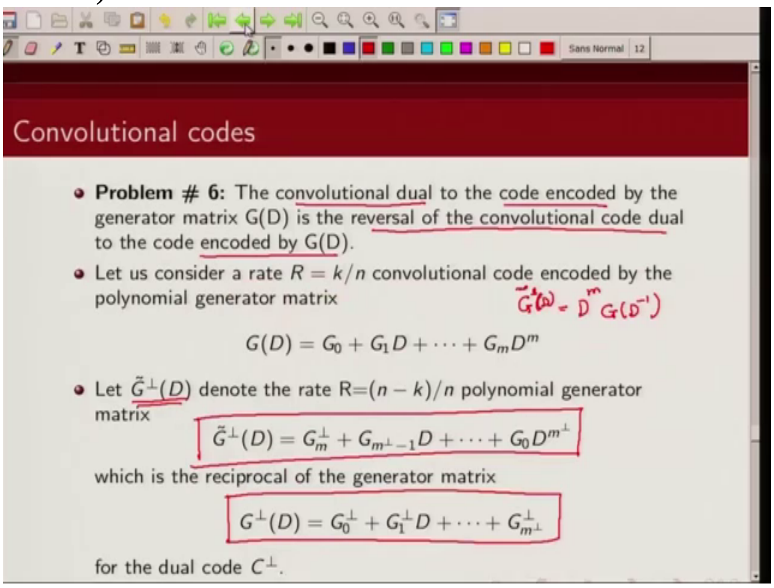
and what is this? This is the dual of, generator matrix is dual of this, so this, this G D G transpose should be zero, right? So this whole summation would be also zero. So what we have shown is then the dual of the, dual of this code

(Refer Slide Time 20:50)



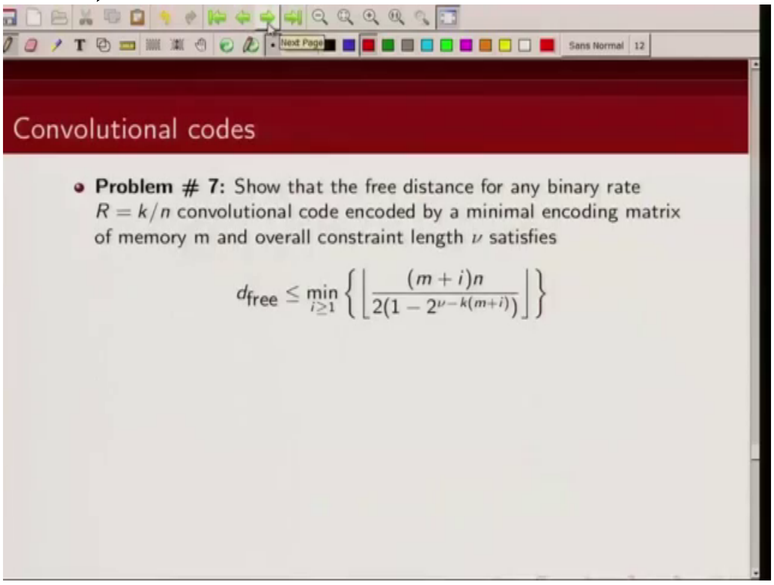
which has generator matrix G

(Refer Slide Time 20:53)



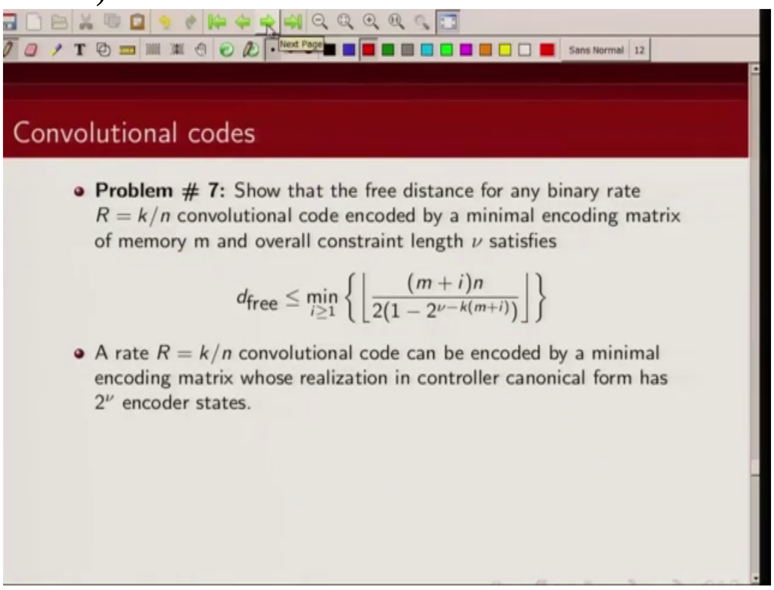
is nothing but the reversal of convolutional code dual because this generator matrix is nothing but it is the reversal of the generator matrix of the dual code of c, Ok.

(Refer Slide Time 21:14)



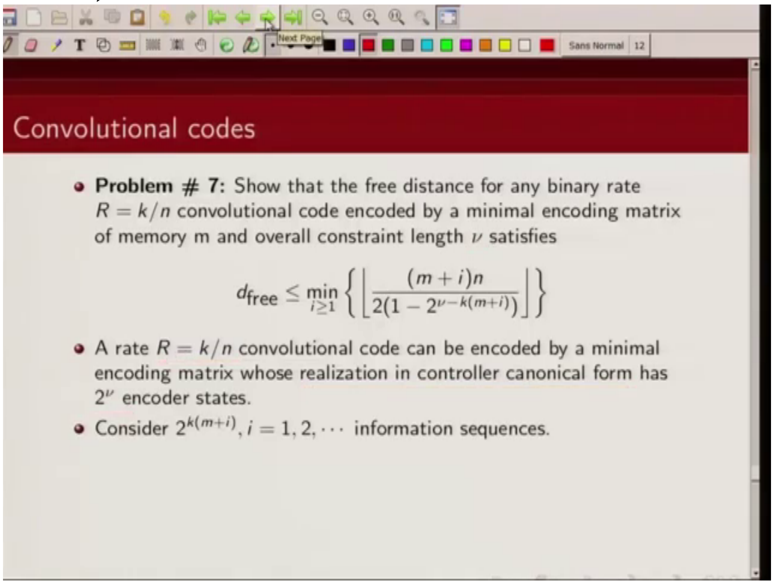
Next we are going to show a bound on free distance of convolutional code. So show that if you have a rate k by n convolutional code whose encoding matrix is memory n and overall constraint length mu then free distance is upper bounded by this.

(Refer Slide Time 21:43)



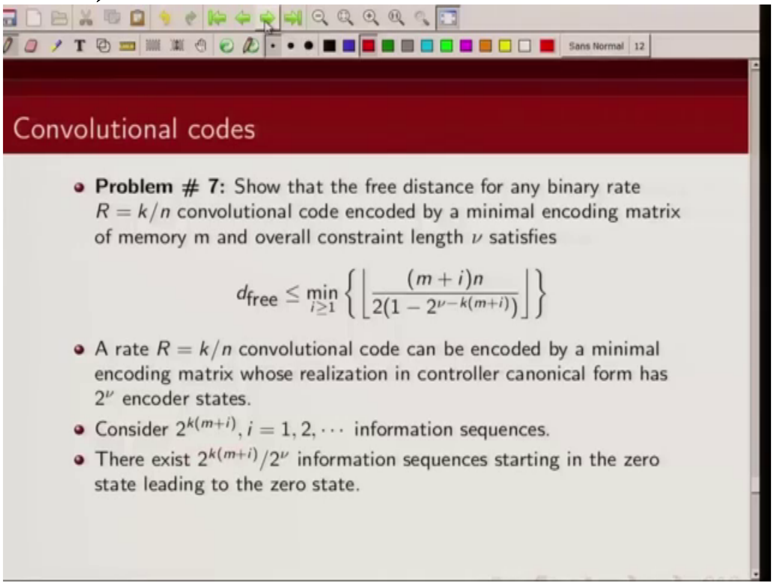
So if you have a rate k by n code, we can realize it using controller canonical form using 2 mu encoder state because overall constraint length is 2.

(Refer Slide Time 22:00)



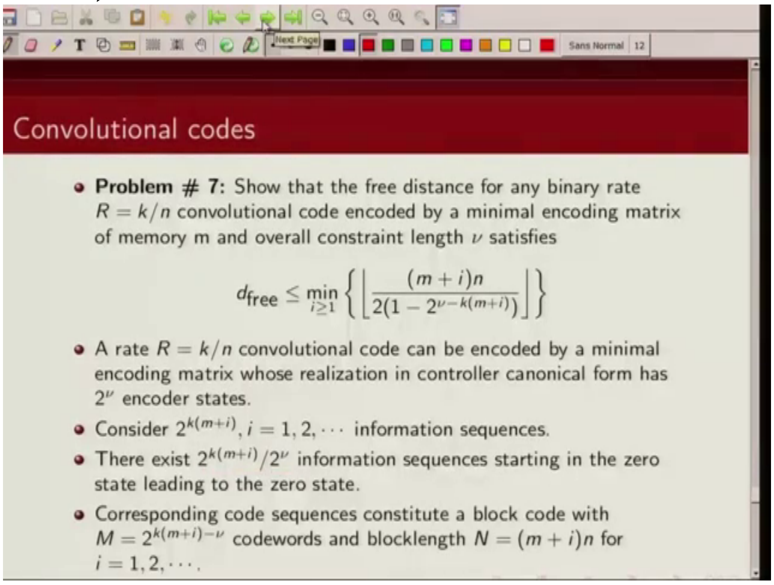
Now if we consider 2 raised to power k into m plus i information sequences

(Refer Slide Time 22:08)



then there exist 2 k m plus i divided by 2 mu information sequence starting at zero state leading to all zero state. So

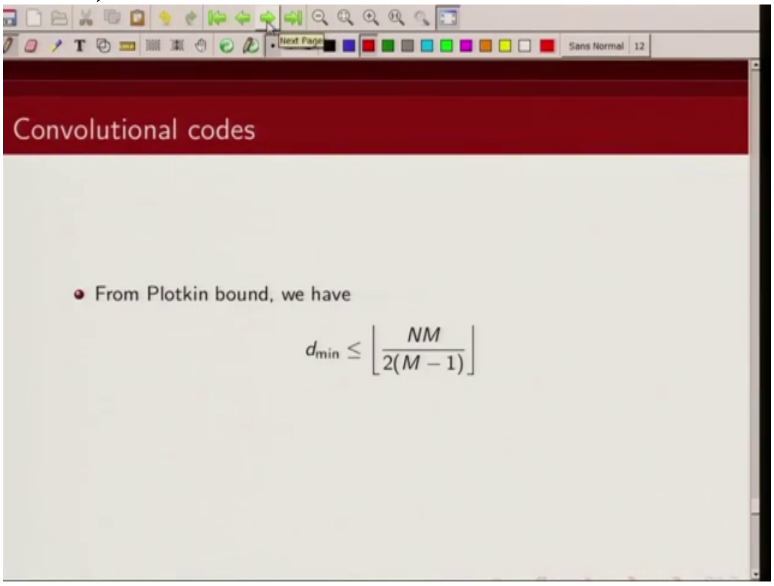
(Refer Slide Time 22:20)



if we count number of such codewords, then number of such codewords m is given by 2 raised to power k into m plus i minus mu and what is the length of this codeword? Length of this codeword would be length plus i into n, because it is a rate k by n code, right?

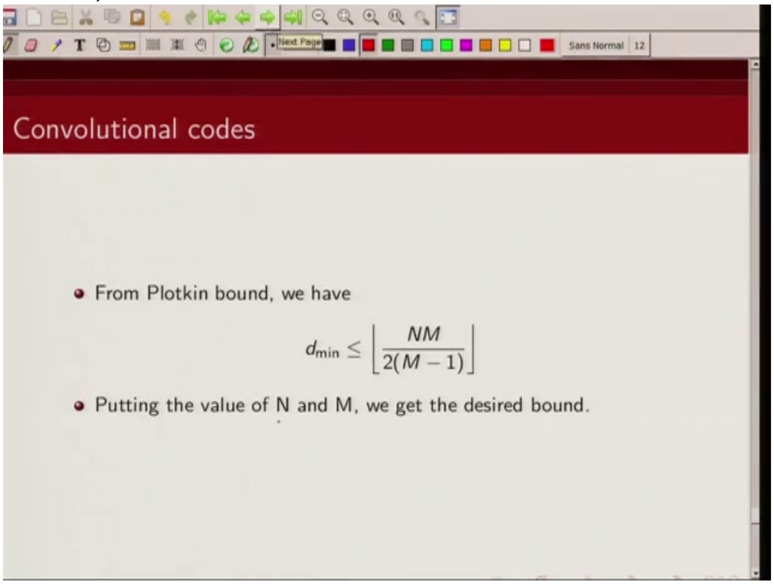
Next we are going to use

(Refer Slide Time 22:49)



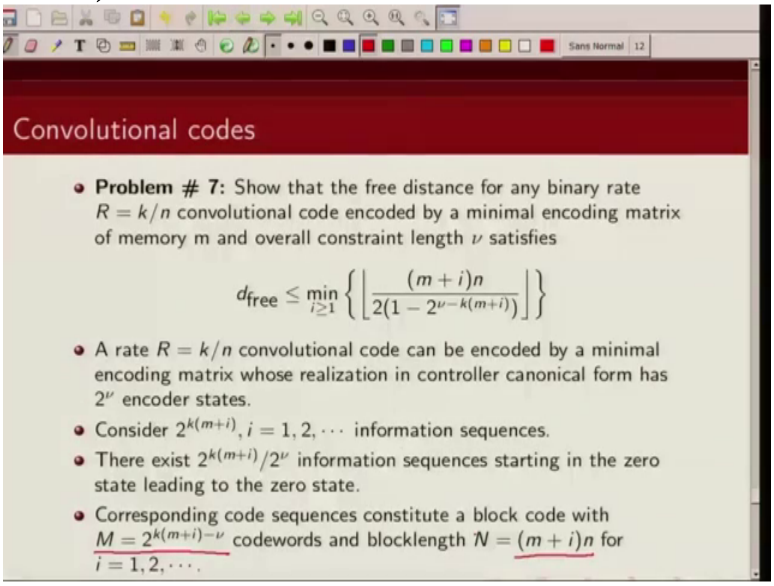
Plotkin's bound. The Plotkin bound says it upper bounds the minimum distance as floor of, this is codewords length, number of codewords, 2 into number of codewords minus 1. So in this example what is our

(Refer Slide Time 23:07)



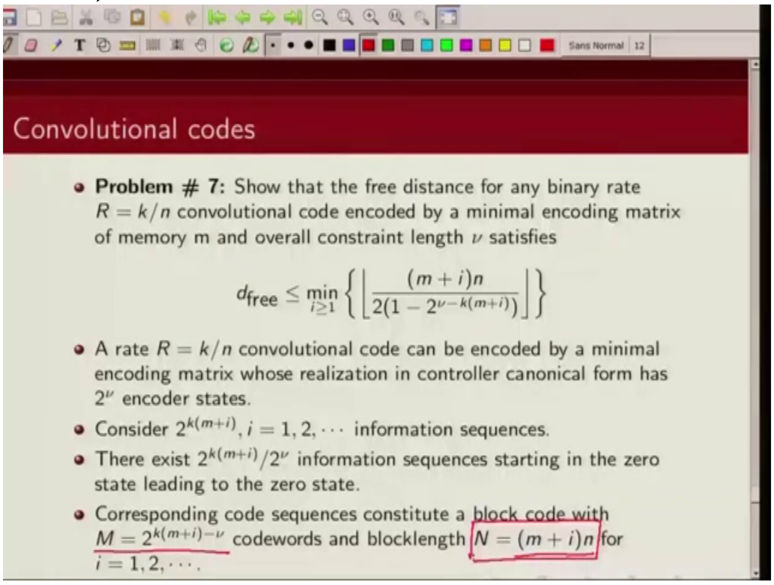
n? Our n is

(Refer Slide Time 23:10)



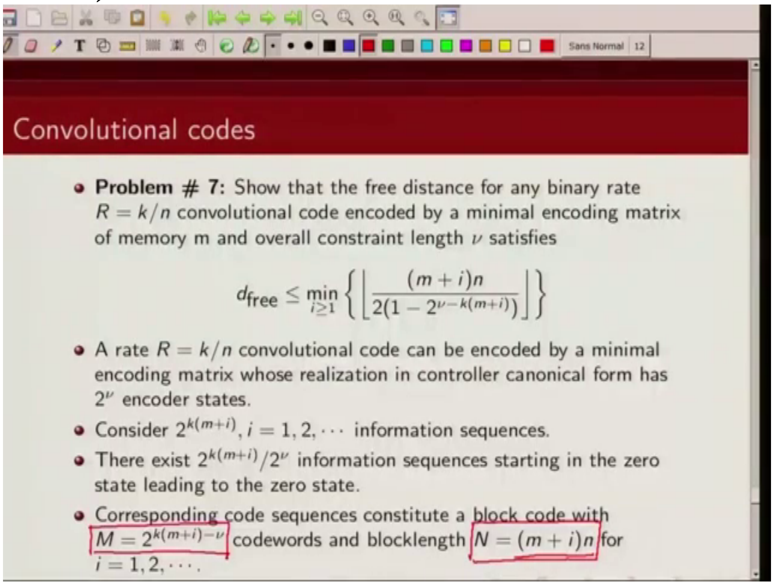
this, Ok?

(Refer Slide Time 23:14)



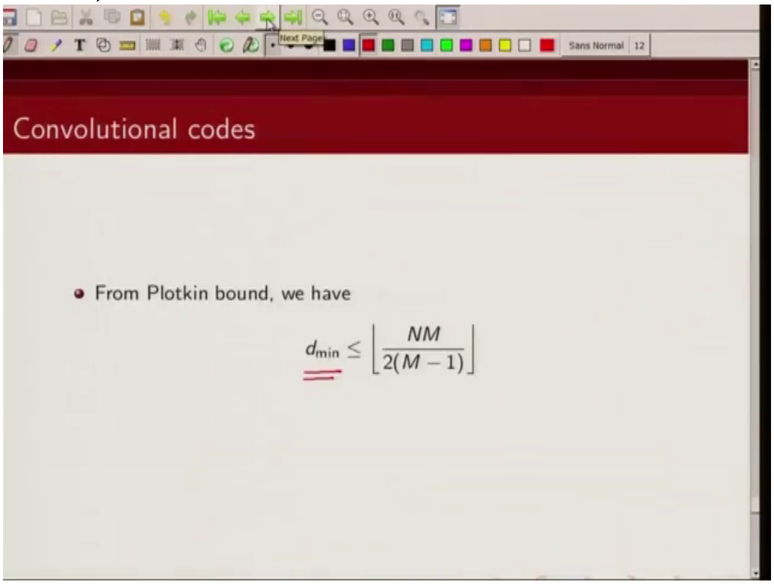
This is our n. And what is our m? Our m is this.

(Refer Slide Time 23:19)



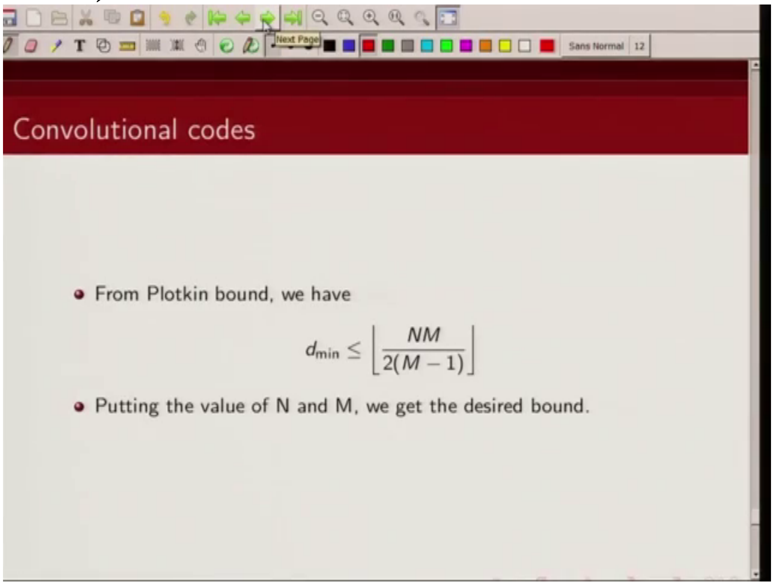
So we plug in this value of m and n in our

(Refer Slide Time 23:24)



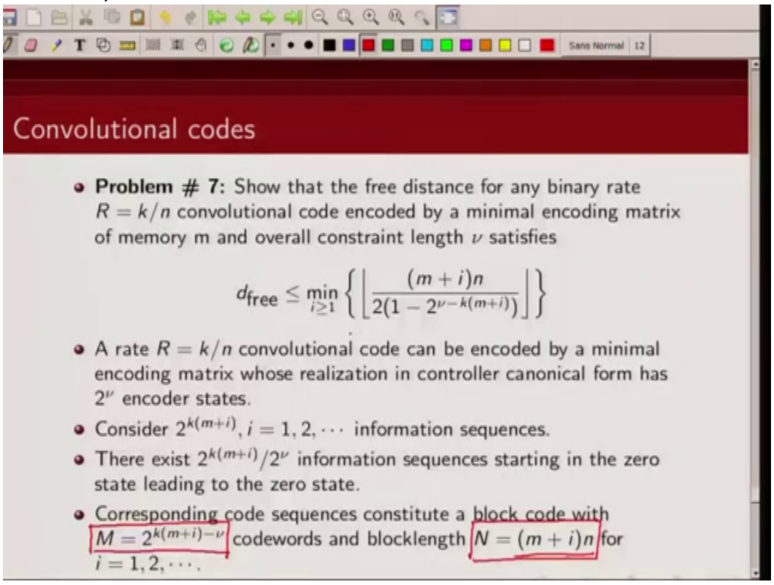
Plotkin bounds what we will get is

(Refer Slide Time 23:27)



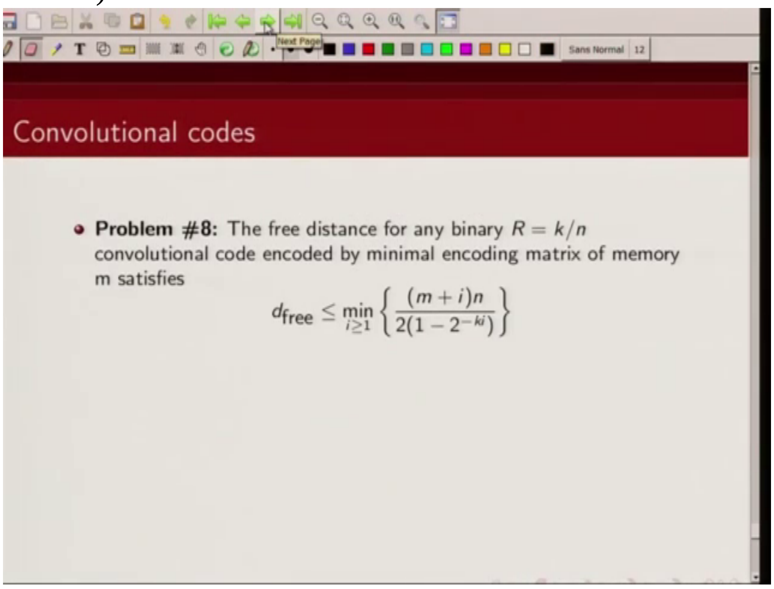
our desired bound which is this result

(Refer Slide Time 23:31)



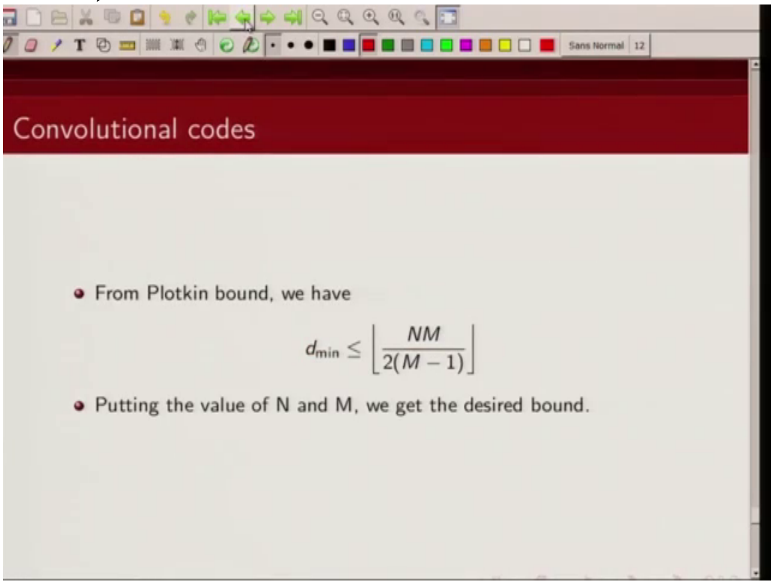
, Ok

(Refer Slide Time 23:35)



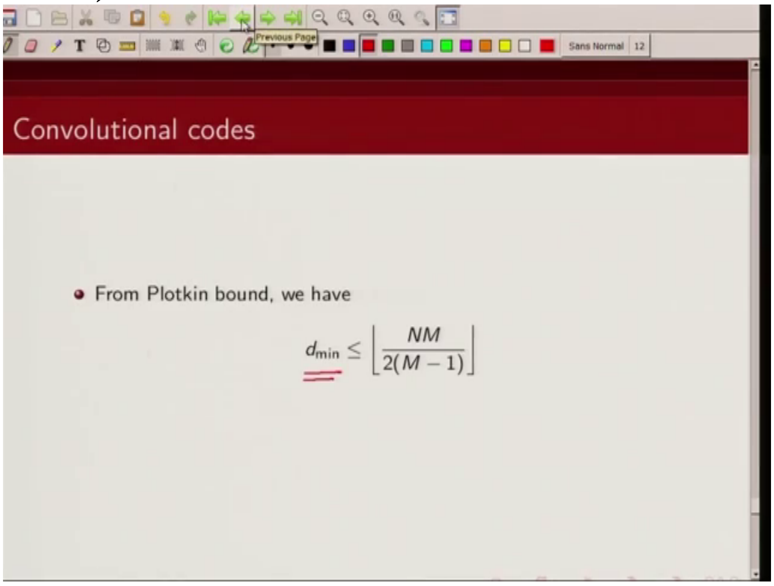
So the next problem we will solve is to show that free distance of a rate k by n convolutional code is upper bounded by this. Now this result can be obtained from

(Refer Slide Time 23:54)



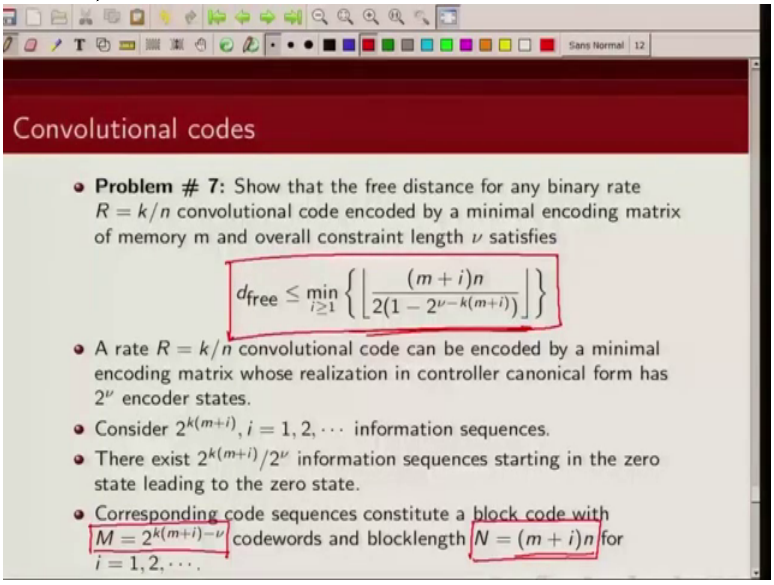
the bound that we had

(Refer Slide Time 23:57)



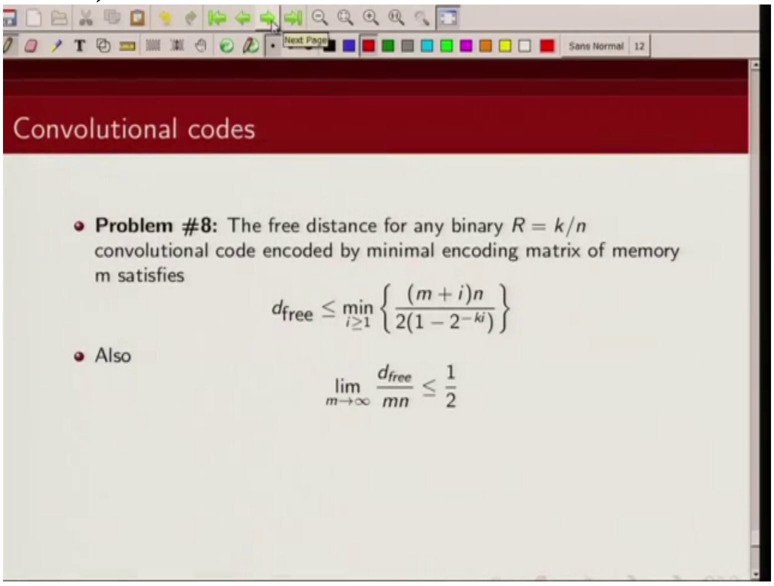
derived earlier, this bound,

(Refer Slide Time 24:02)



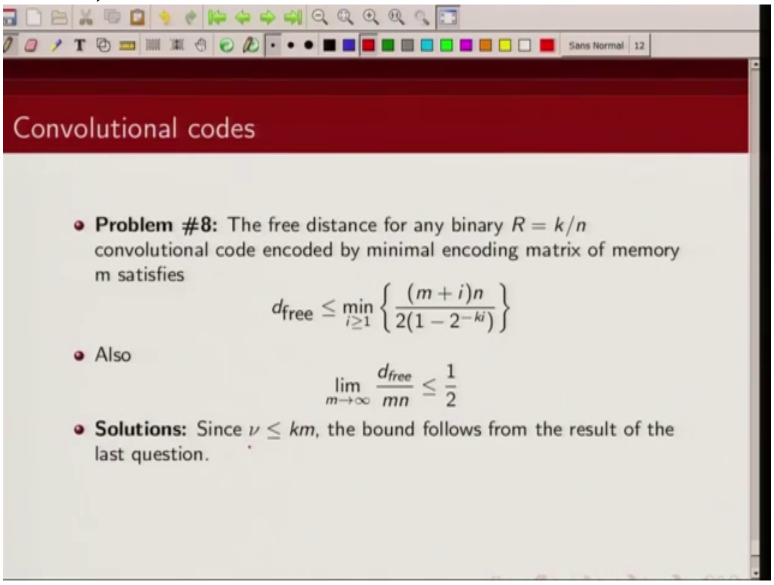
Ok.

(Refer Slide Time 24:04)



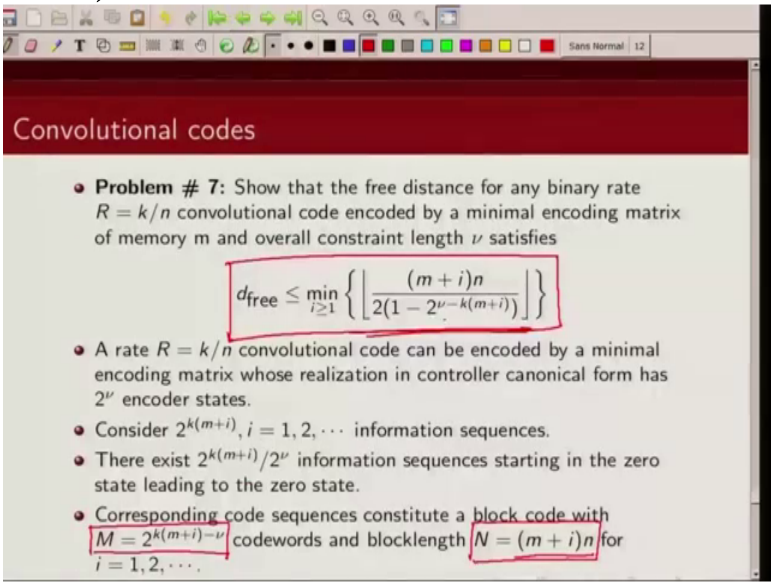
And also we will show that this relation holds. So since

(Refer Slide Time 24:10)



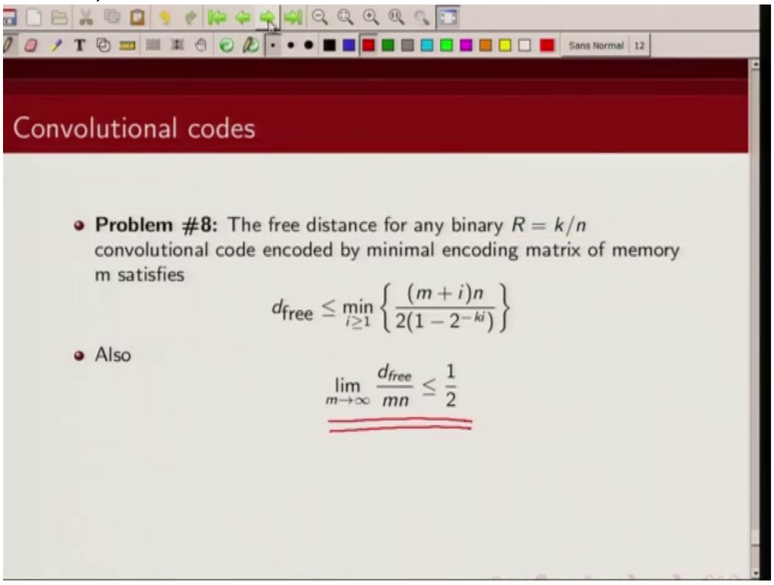
overall constraint length is less than k times m, then if you go back to this expression

(Refer Slide Time 24:19)

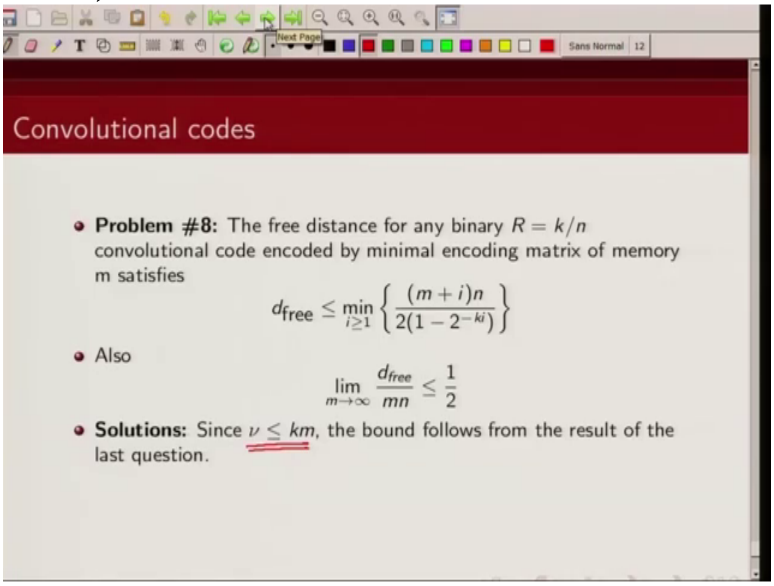


v minus k m is actually less than zero so we can then upper bound this by just 2 raised to power k i. If we do that we get this expression, Ok.

(Refer Slide Time 24:38)

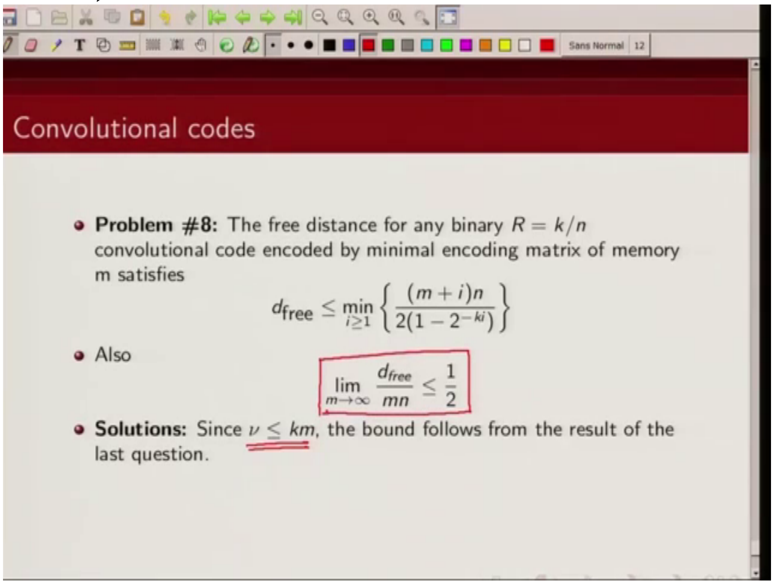


(Refer Slide Time 24:39)



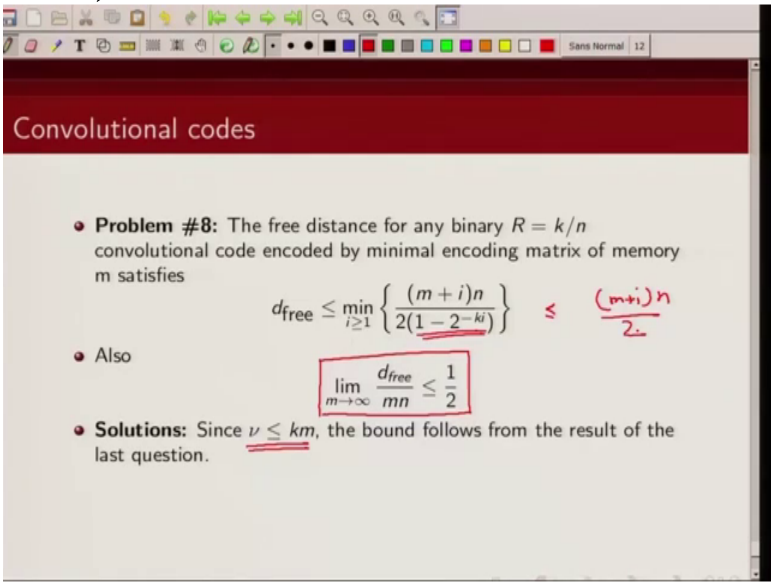
Now how do we get this expression?

(Refer Slide Time 24:46)



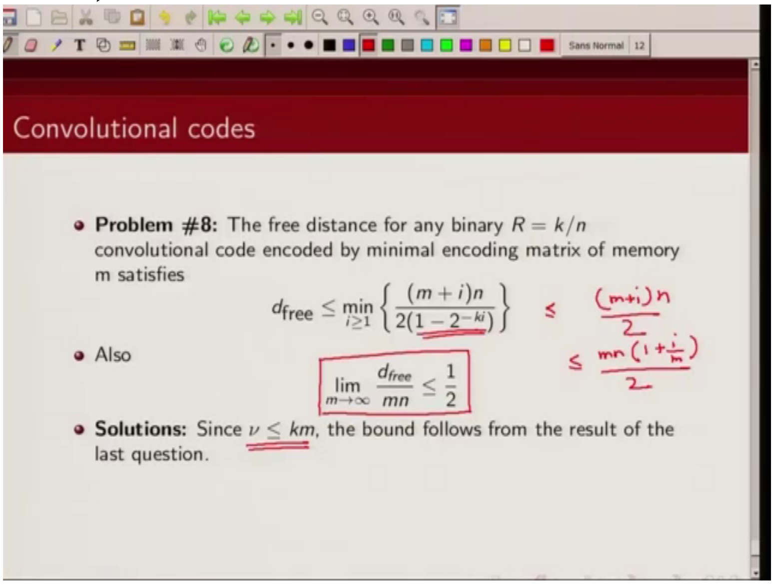
So we know that this term is less than 1, and so this we can just upper bound by m plus i by n by 2, and if I

(Refer Slide Time 25:00)



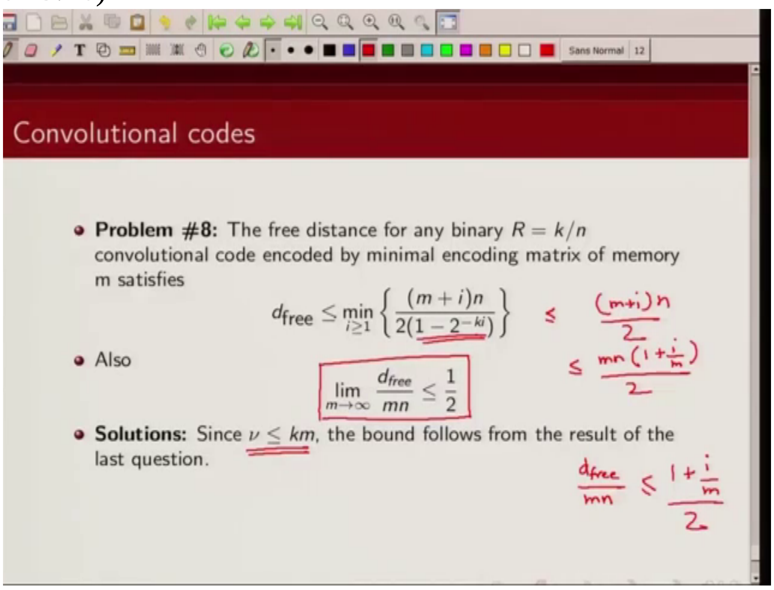
take m and n out, this will be 1 plus i times m by 2.

(Refer Slide Time 25:10)



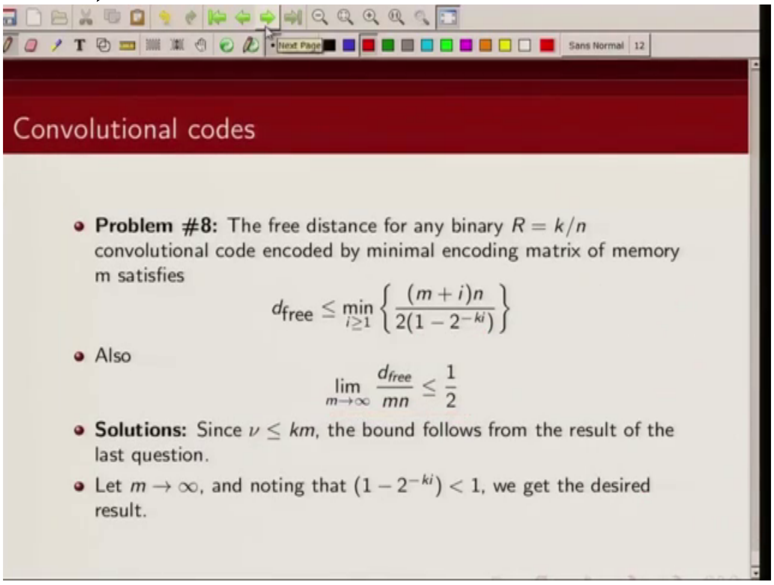
So D free by m of n will then be, so if I do further, D free by m of n would be upper bounded by 1 plus i by m by 2

(Refer Slide Time 25:26)



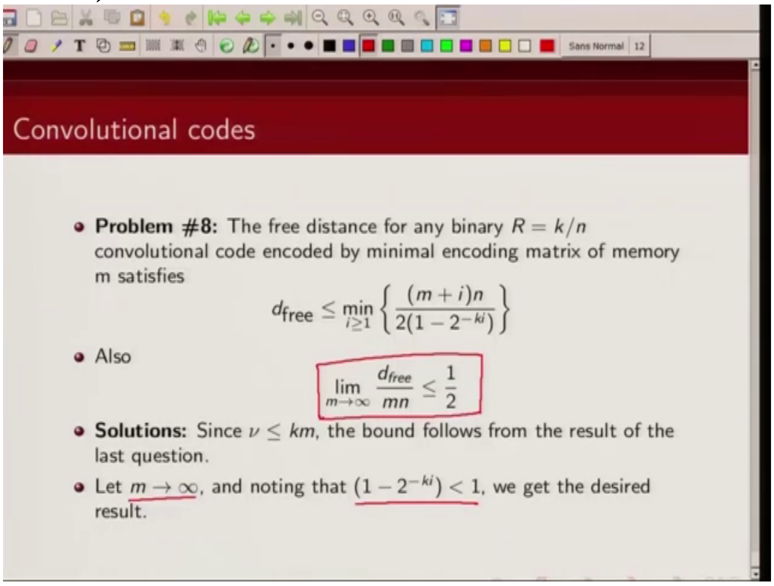
and if we let m go to infinity this would go to zero so that would be upper bounded by half. So that's

(Refer Slide Time 25:34)

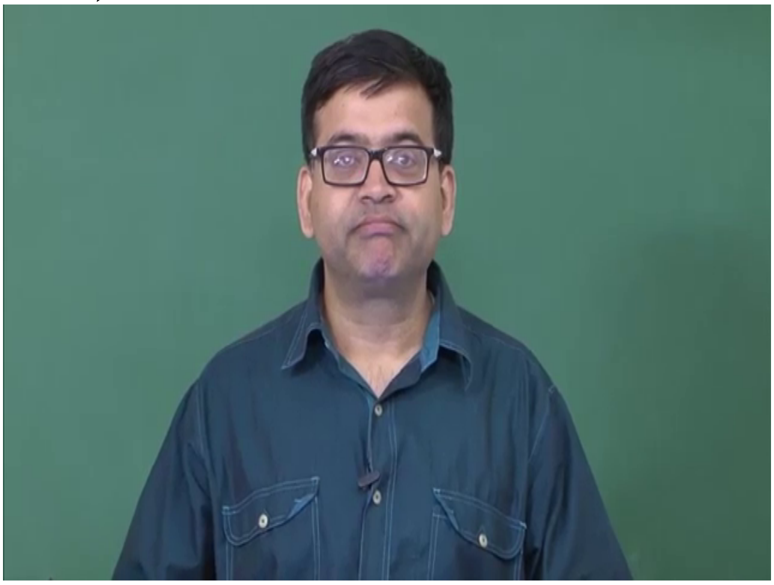


the proof. So you let m go to infinity and since this is less than 1, what you will get is this,

(Refer Slide Time 25:44)



(Refer Slide Time 25:45)



thank you