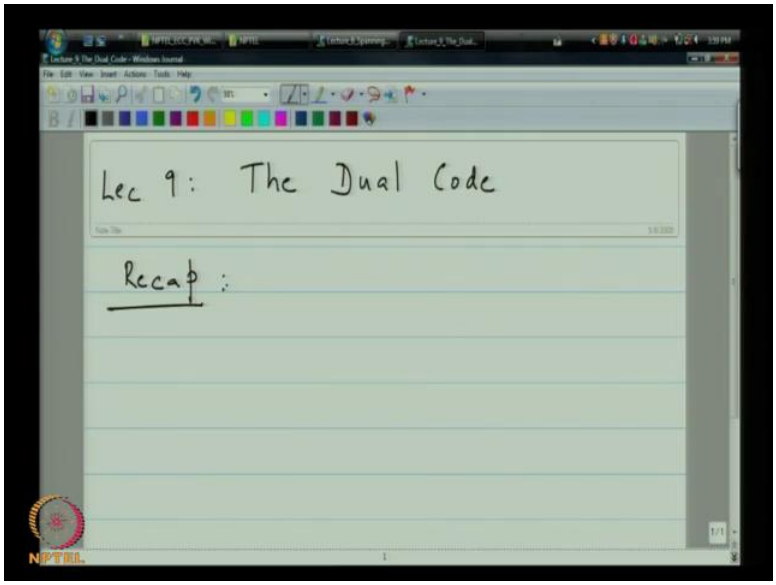
Error Correcting Codes
Dr. P. Vijay Kumar
Department of Electrical Communication Engineering
Indian Institute of Science, Bangalore

Lecture No. # 09
The Dual Code

(Refer Slide Time: 00:37)



Good after noon and welcome back again. So, we will continue our lecture series and if you are keeping count, this will be the ninth lecture, and I am going to quickly just put down a title for the talk and then will go on to recap, what we did in the last lecture. So, I am going to call this lecture, the dual code and let us take a quick recap of what we did in the last lecture. So, in the last lecture, we looked at the notion of spanning, the space span by a set of vectors, we looked at an example we looked at an example of a standard the standard basis and the space span by it.

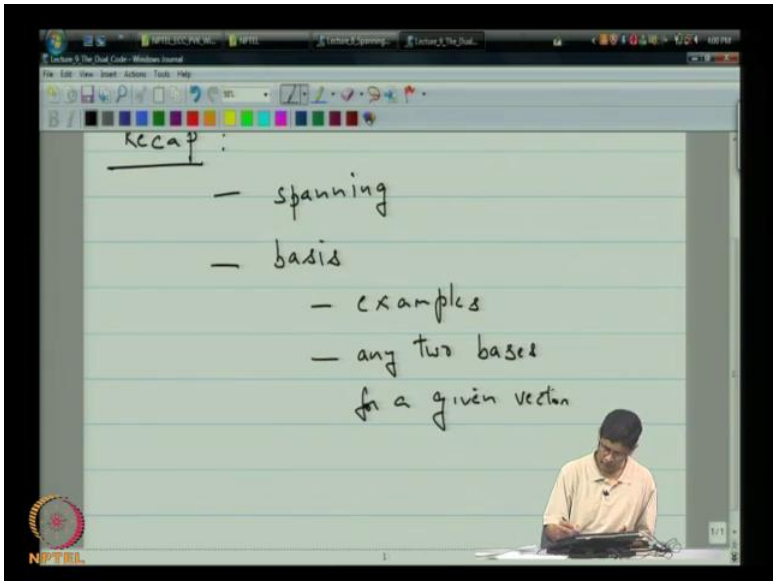
So, we introduced the… So, the standard basis in example of basis of course, and basis is defined as something that combines the concepts of linear independence and spanning. We define that we looked at examples, then we actually stated a theorem which said that supposing you have a vector space that contains a finite number of elements in its basis, then

we should that that number will be the same across different basis for the same vector space. And by counting the number of elements in that basis, you get the notion of dimension, and after that we defined what it means to talk about the dimension of a code and we looked at some notation, and that I think is about where we stopped.

So, today the plan is to actually continue and develop the notion of linearity a little bit further. So, we will introduce along with the code a notion of another code that strongly associated to the regional code and is called the dual code. So, let us just a limit, first begin with the quick overview of our last times lecture. Now, this is small for you to view, but hopefully by the time you are seeing this, you will have a hard copy in your hands.

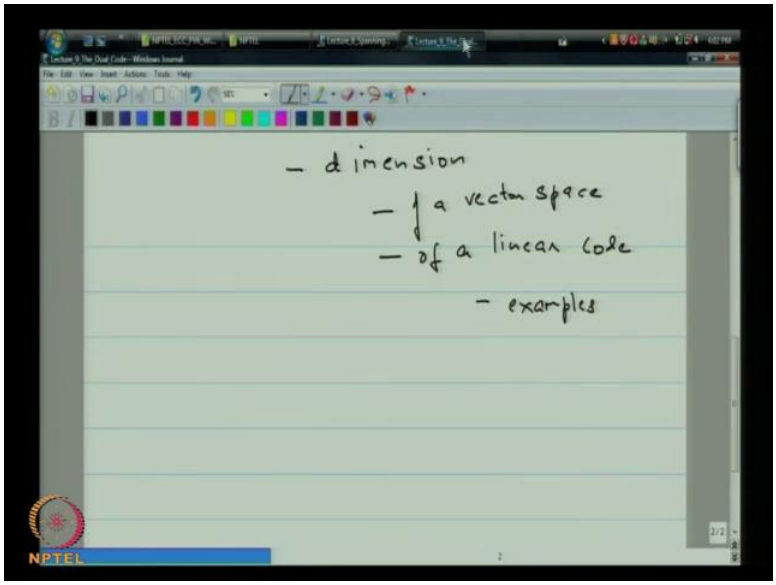
So, it should not be hard. So, here is where we finished the discussion on spanning then we talked about a basis. We prove that basis contains the same number of elements, we talked about the dimension of a vector space applied to a code defined, what is meant by dimension of a linear code and we define some notation in connection with it and took a quick look at an example.

(Refer Slide Time: 03:27)



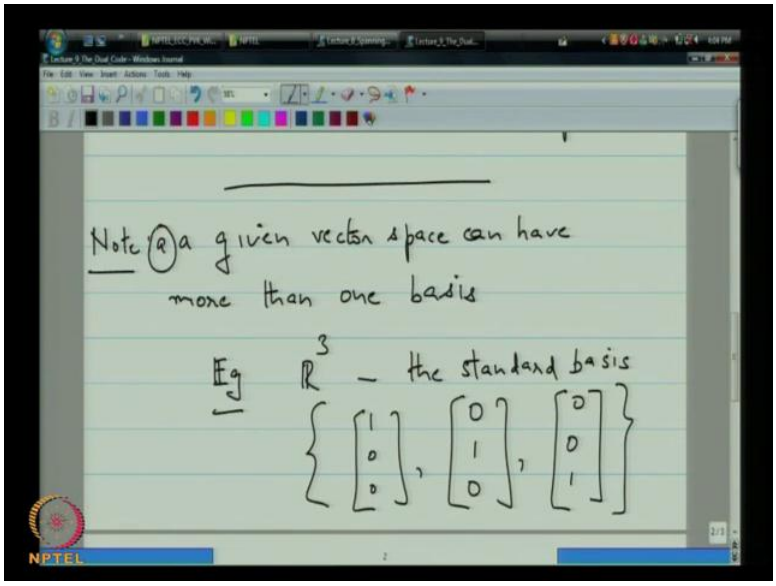
So, our recap will put down, that we in the we concluded our discussion on spanning and then we defined, what it means to talk about a basis for a vector space and here, we looked at examples and we also proved that roughly speaking any two bases for a given vector space contain the same number of elements .

(Refer Slide Time: 04:38)



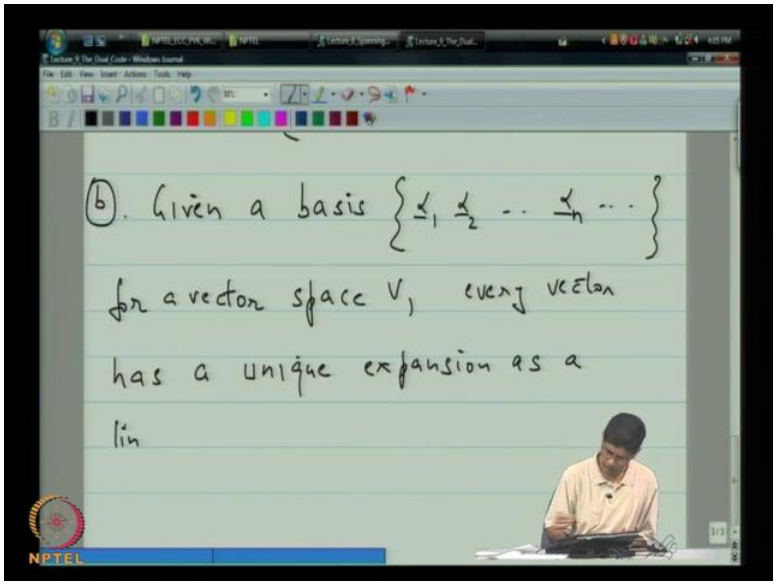
This led to the concept of dimension. First, of a vector space and then of a linear code and then we looked towards the end at examples. So, this this example is an example of a basis for the single parity check code which a losses to assign a dimension of six to the code. So, I just want to (( )) a couple of points as I begin the lecture.

(Refer Slide Time: 05:56)



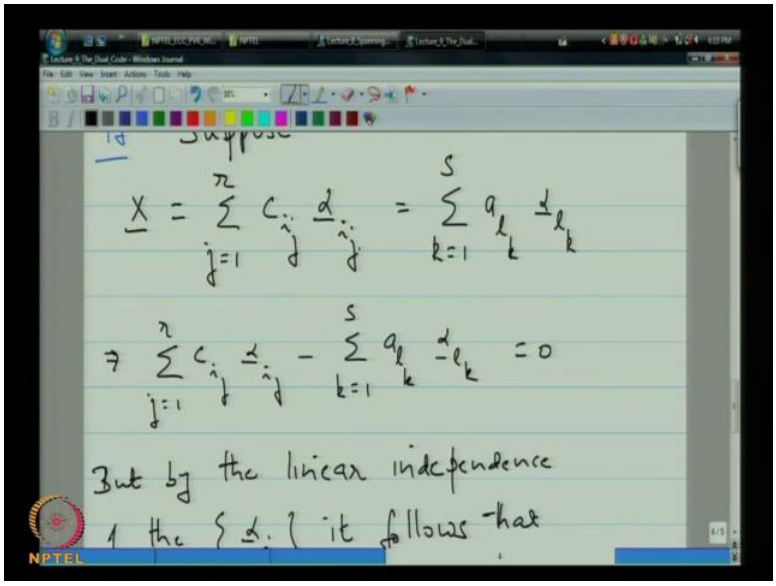
So, we will just cal them out as notes. Note, a given vector space can have more than one basis for example for example, if you look at R three then you have the standard basis which we looked at last time, but we also pointed out that any also any three vectors not on a plane also form a basis. So, which goes to suggest that a basis is for from being unique? On the other hand, there is a property of a basis. So, we are making notes. So, this was our first note. So, let me write a, and put a circle around it.

(Refer Slide Time: 08:08)



And the second note is that, given given a basis alpha one, alpha two, alpha n. For a vector space actually let me make this a little bit more general, I will allow the vector space to have an infinite basis. So, given a basis alpha one, alpha two, dot dot dot for a vector space V, every vector has a unique expansion as a linear combination as a linear combination of elements of the basis. And So, the key point is it as a unique expansion and also as a linear combination. So, the proof is straight forward.

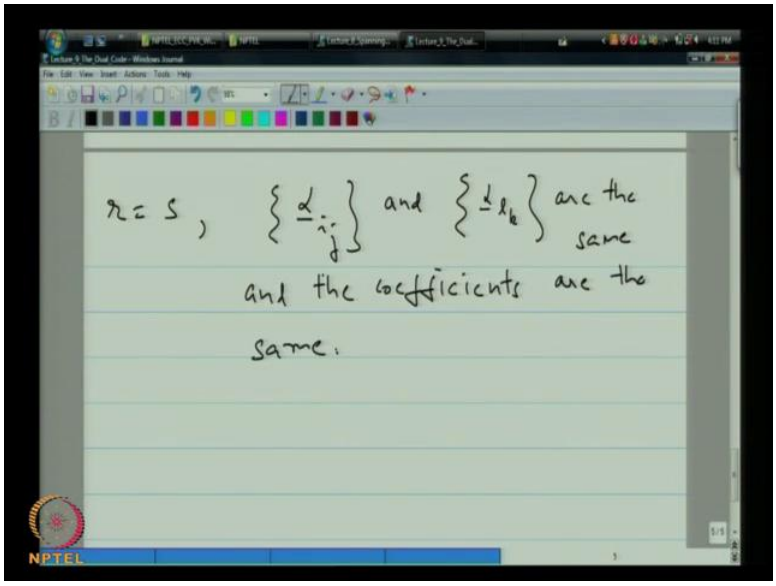
(Refer Slide Time: 09:55)



Suppose and now, supposing you have a vector x, which on the one hand is sigma c i j, alpha i j, j is equal to 1 to r and it is also given by let say, k is equal to 1 to s, c a i k alpha I k, but this would imply this would implies that the sum j is equal to 1 to r, c i j alpha I j minus the sum k is equal to 1 to s, a i k alpha i k is equal to 0, but but but by the linear independence of the alpha i, it follows is that, this can happen if and only if, r is equal to s, the alpha i j are the same.

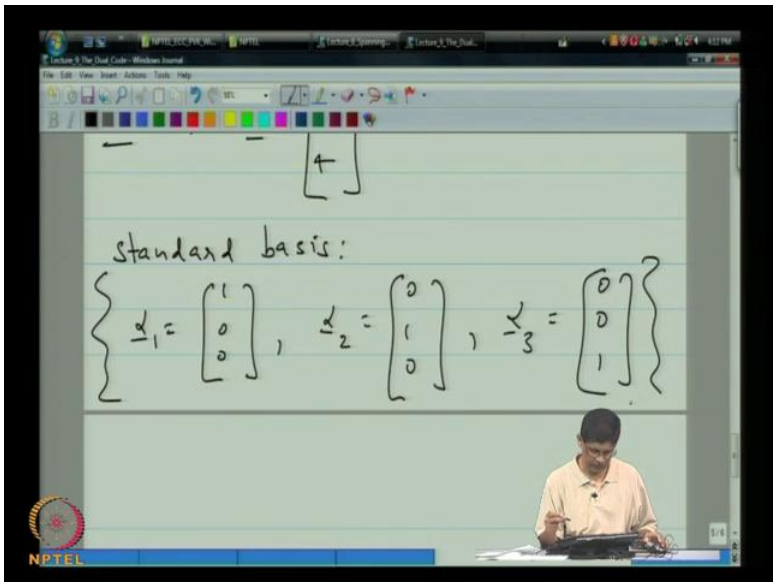
I let me just make one change here; I see that there is the possibility of a confusion here. I want to change the I sub case to l sub case So, that, we do not we leave the possibility leave up in the possibility that the vectors on this side are different from those on the right side So, sorry about that limit just quickly make that correction. So, here this is l k, l k, l k, l k. So, I made these So, that these, such the vectors could extensibly in the beginning at outside could be different, but the only way this can happen is a fall the coefficients in front of every alpha are 0.

(Refer Slide Time: 13:43)



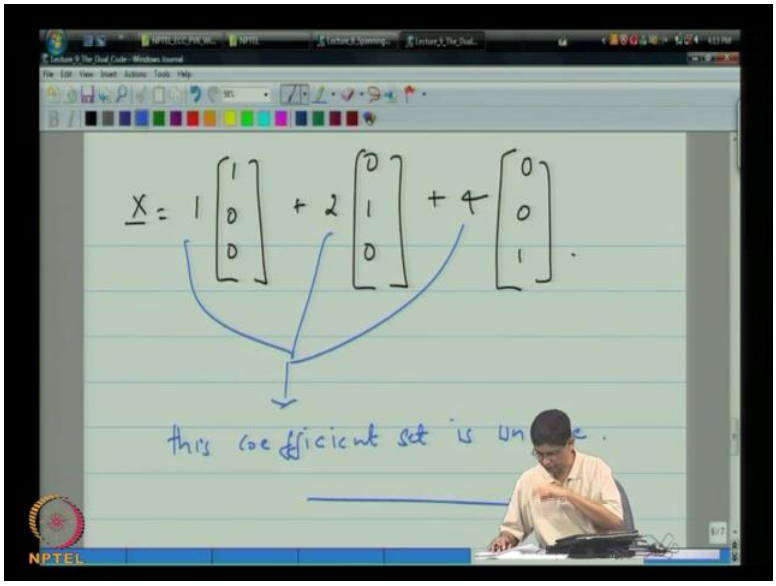
And that can only happen only, if and only if, r is equal to s and alpha i j and alpha and the alpha l k are the same, you have the same set and the coefficients are the same. So, what that proves is that, every vector has a unique expansion with respect to a basis.

(Refer Slide Time: 14:46)



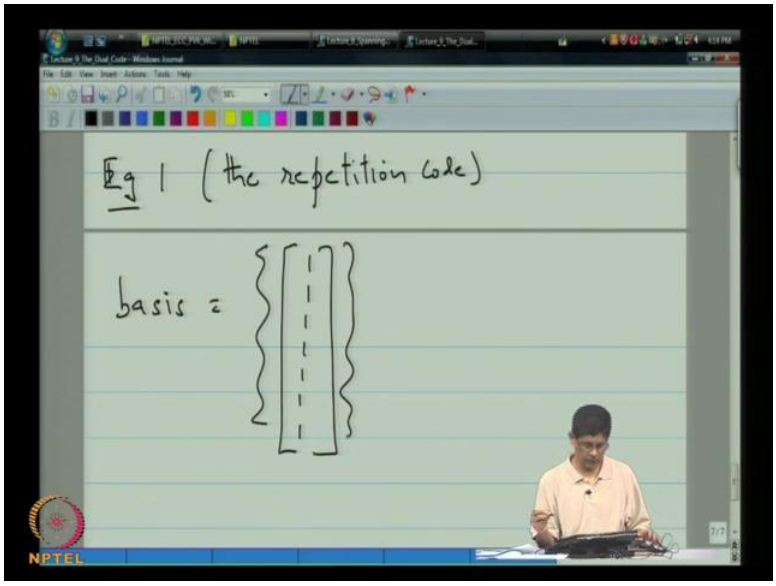
Let us take a quick example, supposing you are in three-dimensional space and you are looking at the vector 1 2 4, let say then we know that, this only one way and let say that a basis is the basis is the standard basis which is that alpha 1 is 1 0 0, alpha 2 is 0 1 0, alpha 3 is 0 0 1. So, this is your basis.

(Refer Slide Time: 15:43)



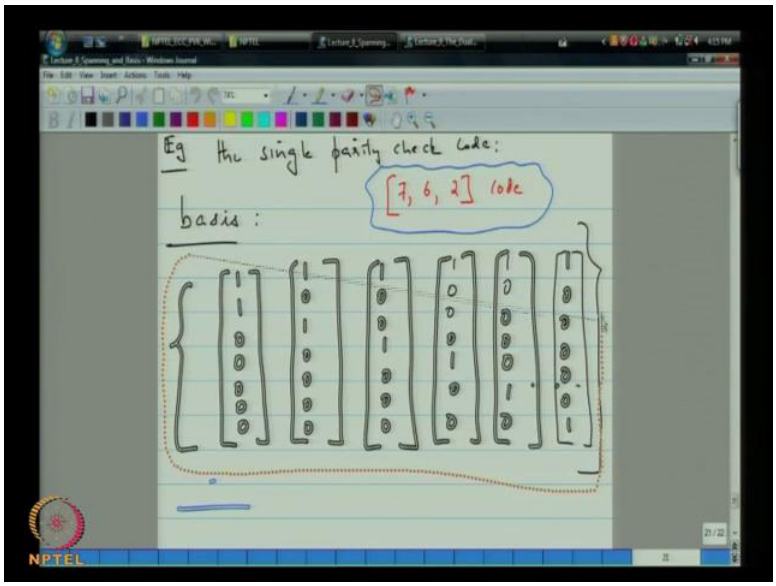
Then it is clear that, x is 1 times 1 0 0 is 1 times 1 0 0 plus 2 times 0 1 0 plus 4 times 0 0 1 and that this is the only way of writing it. So, our point is that these coefficients are unique. This coefficient set is unique right. Now, so you made two remarks concerning a basis.

(Refer Slide Time: 17:09)



So, now let us look at a basis for our example codes for the three example codes example 1, the repetition code. So, in this case it is easy to see that a basis has (( )) single element namely the all one code. So, this is a singleton basis.

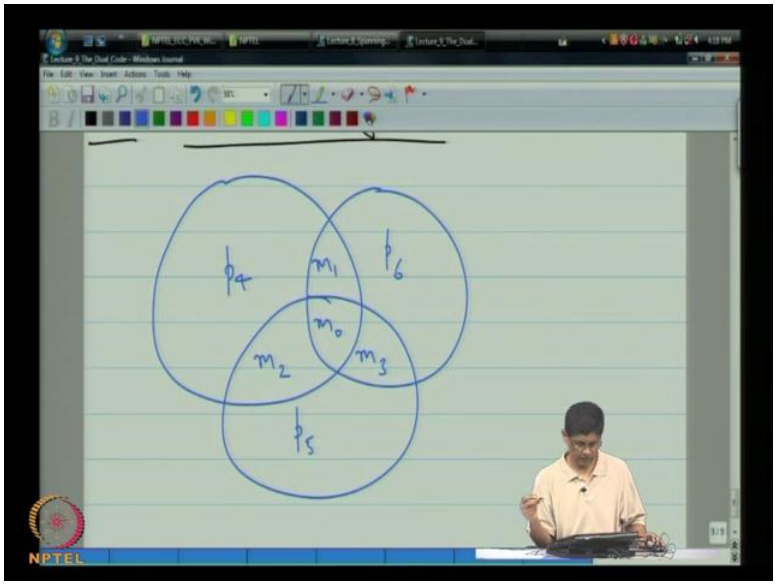
(Refer Slide Time: 18:24)



Example 2, if we look at the single parity check code then a basis for this is now, we have actually looked at that before. So, we will any use take advantage of the technology that we have to bring that up

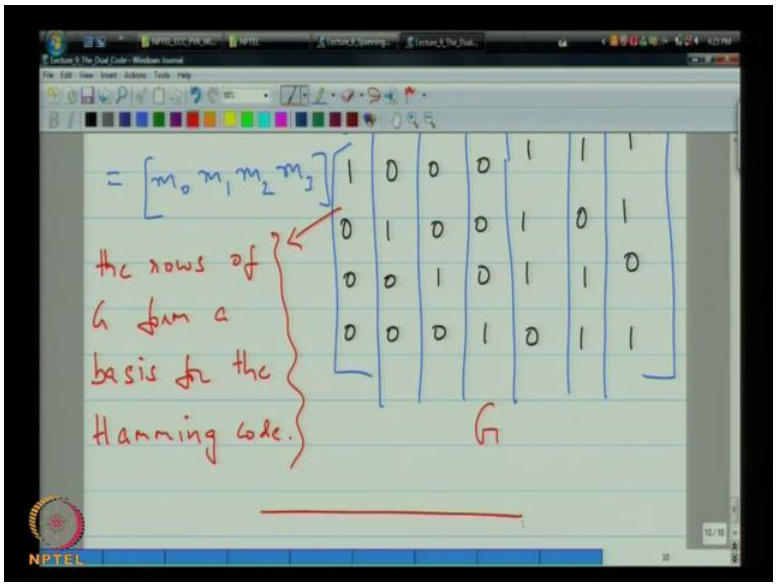
And I am going to paste that over here. So, this is a basis for the example parity check code. So, let me a the collection of six vectors below, I am not providing too much by way of justification, because I think it is fairly straight forward, it is clear that this span the space and these vectors are linearly independent. So, the meet the criterion, so it is fairly clear that they are a basis.

(Refer Slide Time: 20:40)



The last example, the hamming code perhaps needs a little bit more of work. So, example 3, the hamming code now, So, let us begin with our three circles diagram and the symbols are m 0, m 1, m 2, m 3 and you have the parity check symbols p is p 4, p 5 and p 6 right. So, this is the code and I am going to write the code words in a certain fashion which will encounter again just a little bit.

(Refer Slide Time: 22:04)



So, I am going to write the in tag code word like this, m 0, m 1, m 2, m 3, p 4, p 5, p 6. So, this is a row vector which I am going to express in matrix form and I mean draw some lines to help keep me organized here, So, this is three and So, this is 0, 1, 2, 3, 4, 5 and 6 and I am going to what I am going to do here is, my goal is to actually fill up, my goal is to fill up this matrix, keeping in mind that the vector are there are like to generate is this vector and so let see how we do that. So, my goal is to actually generate this code word over here and I want to express it in terms of the m I's. I am going to think of the m I as message symbols and I am going to fill in this matrix accordingly.

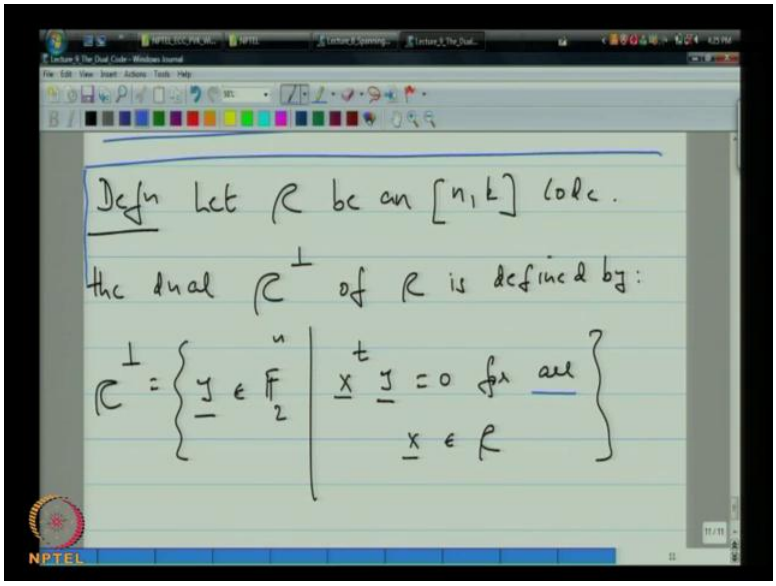
So, from the so from that it is not too hard to see that, the correct way to fill it up is like this.

Now, at this point I have expressed the first new symbols. Now, I need to look at how to generate p 4, p 4 is a linear combination of m 0, m 1 and m 2. So, what I do here is I put down 1 1 1 and is 0 and in a similar fashion to generate p 5, it is m 0, m 2 and m 3. So, that is 1 0 1 1 and for the last symbol it is m 0, m 1 and m 3. So, I have written down in expression which tells you, how the code words in a hamming code are generated, but this

particular matrix is the matrix, that we will frequently encounter. It is called a generator matrix, but before you get to that, this is notice is couple of each of this matrix, excuse me.

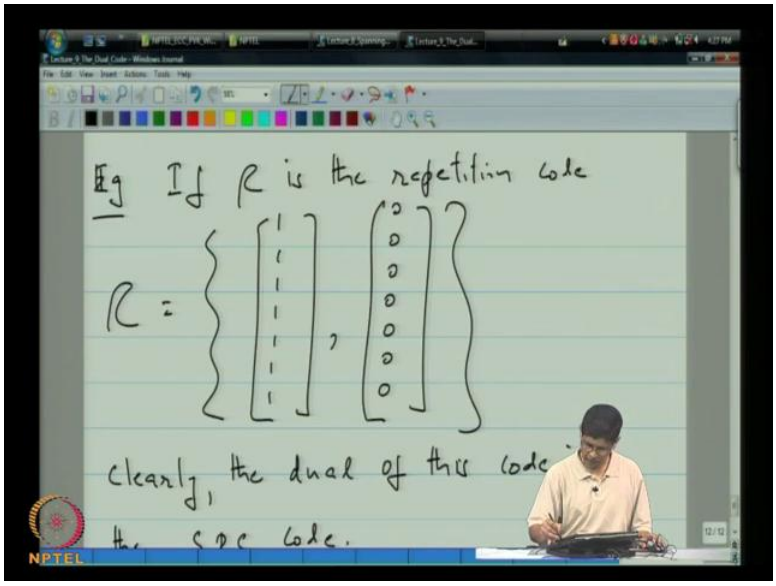
So, first of all getting down to the matrix, you can see that, every code word can be expressed as linear combination of the rows of this matrix. So, I am going to call this matrix G, we call this our matrix G and this times, this is short for generator matrix. So, every code word can be express is a linear combination of the rows of this matrix. So, it is certainly it is true that the rows span the code and it is also clear that they are linearly independent; hence the rows of this matrix form a basis for the hamming code. So, that is the conclusion, the rows of G form a basis for the hamming code with that we now, ready to move on to the main topic of this lecture which is the dual code.

(Refer Slide Time: 27:03)



So, definition let c be an n, k code, if we look all from the last lecture, we explain that this is notation for a code for a linear code whose block length is n and its dimension is k. The dual the dual c perp of c is defined by c perp is equal to the to set of all vectors y, such that x transpose y is equal to zero for all, x in the code. It is important that, this is for all. So, this is a what is meant by the dual of the code.

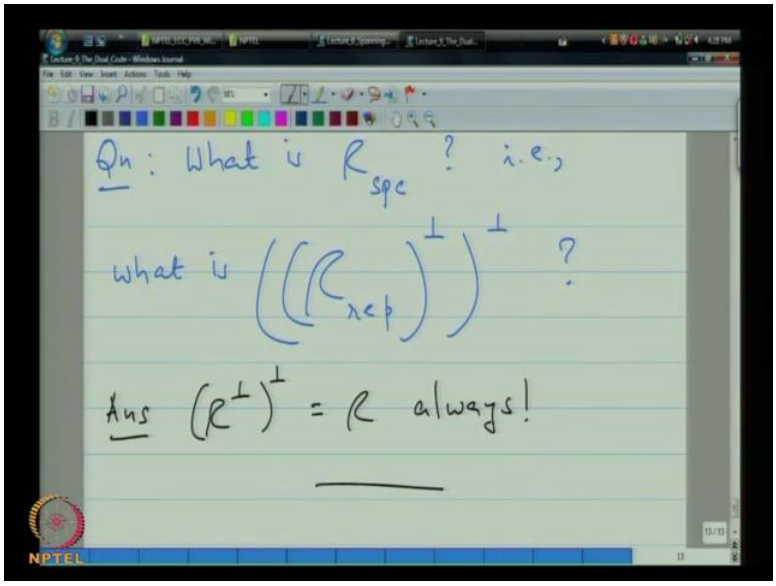
(Refer Slide Time: 29:06)



Next topic example, example, if c is the repetition code that is c consists of the vectors will you call that, the dual code is the set of all vectors which is which are such that, when you take the inner product with the either of this code words will get 0. So, certainly you will always get 0, if you take the inner product will 0. So, only requirement is that the inner product with the all one vector is 0 which is equivalent to the saying that the vector has even parity.

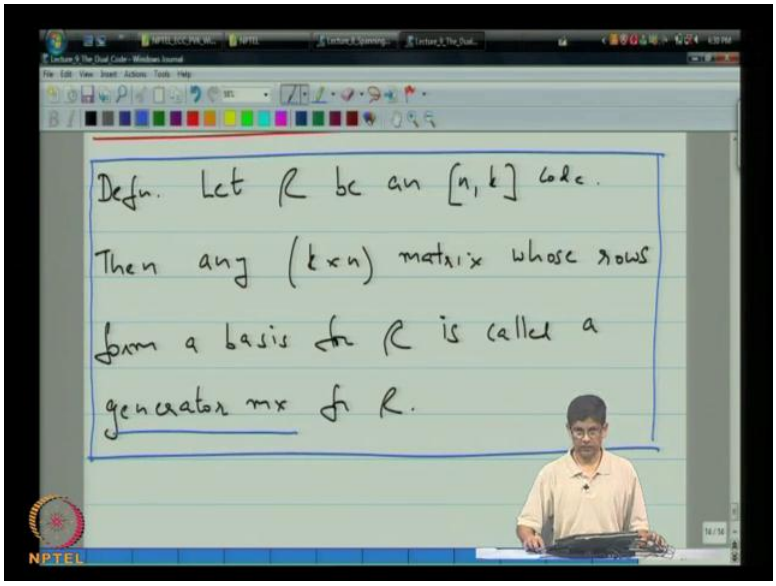
So, from this it follows that the dual of this code is nothing but the single parity check code. So, will note that, clearly, the dual of this code is the single parity check code right and so we can summarize these and say i e, if you take the repetition code, its dual is the single parity check code. So, an interesting question that you might ask is, what is the dual of the single parity check code?

(Refer Slide Time: 31:09)



So, will put that down as a question, what is the dual of the single parity check code or in other words i e, what is the dual of the dual of the repetition code. Now, some of you may actually guess the answer, the answer terms out to be that, c dual dual is c always not just in this instance, this always the guess. So, our next goal is to actually prove this. So, but along the way to providing a proof for this remember just few minutes ago, we introduced a generate a matrix of the hamming code. So, what will actually do is, will define a matrix G associate to every code.

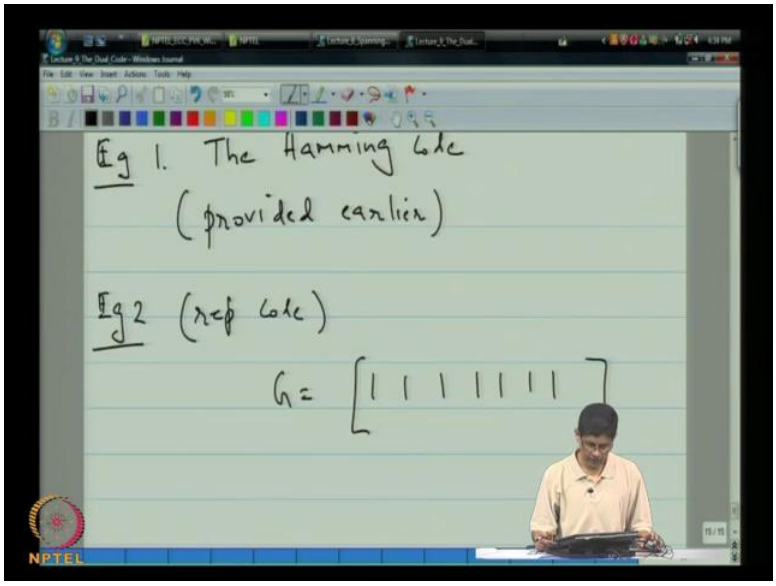
(Refer Slide Time: 32:24)



So, definition let c be an n, k code then then any k by n matrix matrix whose rows form a basis for c is called a generator matrix. Now, an obvious common because remember, when we will talking about a basis, I mention to you that given a vector space a basis for the vector space is not unique, you can have several and it follows therefore, that given a linear code they could be several basis a several generator matrices for the code .

So, note every code can in general or rather not every code let say a code more exactly, more preciously a code can in general have more than one generator matrix right. Now as for is examples go, we have already seen a generator matrix for that hamming code. So, in (( )) since this is (( )) I should use a different color, let me write in green. So, I will put down here, this is a generator matrix matrix for the hamming code.

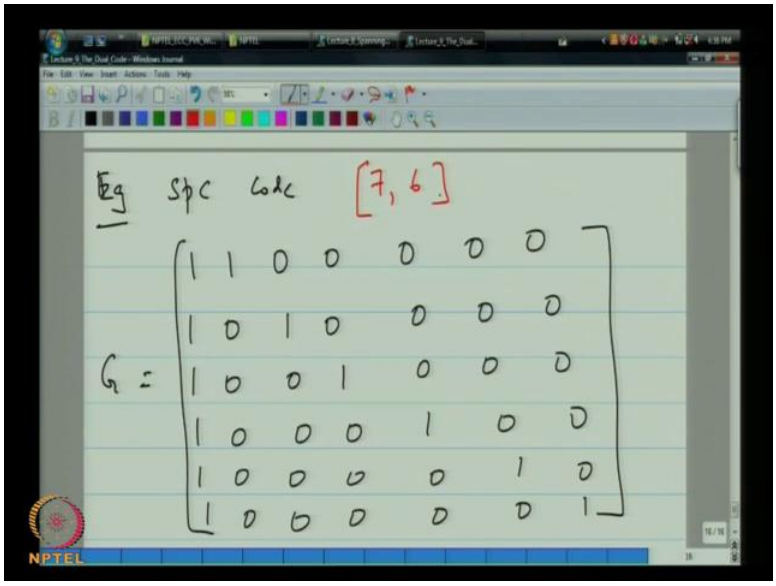
(Refer Slide Time: 36:39)



So, example one one the hamming code. So, the example was provided earlier. Example two for the repetition code, it is not had to see that the generator matrix is very simply a row of all ones and the one one remark here is an order. Now, when I talk about the repetition code here and I put down vectors, I am putting them down as column vectors however, when you view them in terms of the generator matrix code words appear along the rows.

So, you have (( )) for that and its standard terminology in a coding theory to define generator matrices, the weight have defined it. So, there for whatever historical reason, it is the rows where appear as code words, but still my notation will be that, by default a vector is a column vector. So, this is a basis for the generator matrix, so the repetition code.

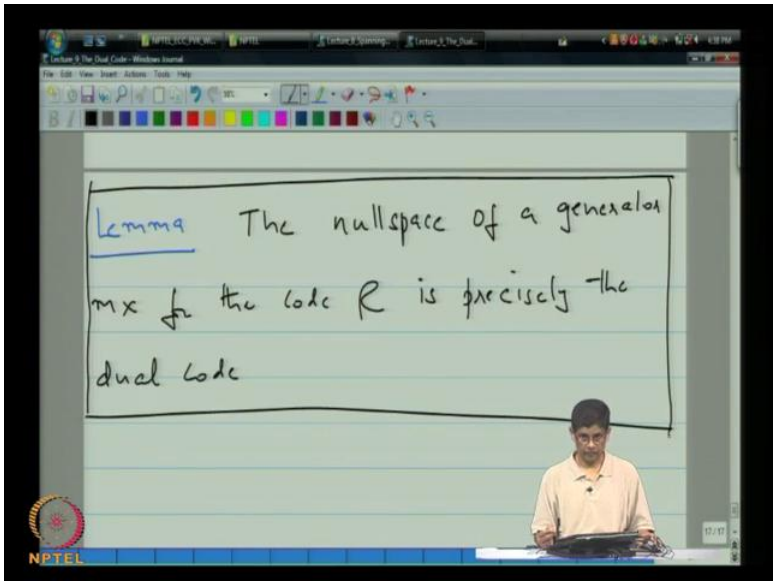
(Refer Slide Time: 32:24)



And for the single parity check code case, a generator matrix this simply obtain by simply putting down in the form of rows, the vectors that we looked at earlier. So, let see if again pull that up. So, these vectors which we breaking down as column vectors, if you write them down as rows of a matrix then you will obtain preciously a generator matrix for the code. So, that is what I am going to do.

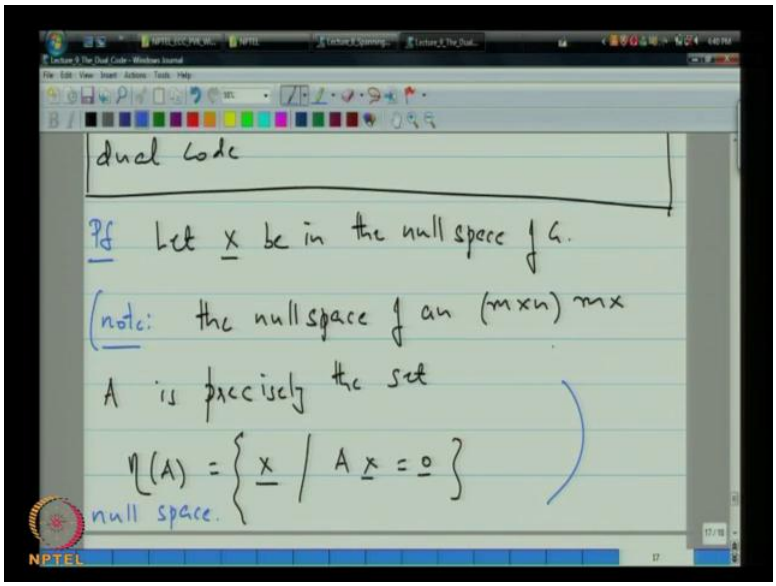
So, this is a generator matrix and will always use the symbol G to generate a to denote a generator matrix and a notice that the matrix will always have. So, since the single parity check code has parameters 7, 6 is a 7, 6 code. So, this matrix is therefore, 6 by 7. So, in general, the generator matrix will always be a k by n matrix. Now, so what you done up to now, we have introduce the generator matrix, I am mean, we first introduce the null code, the dual code and then I introduce the generator matrix. So, next what will do is, will make a connection between the two.

(Refer Slide Time: 40:37)



So, I will put that down as a theorem a perhaps is lemma, the null space of a generator matrix matrix for the code c is precisely the dual code. So, how do you prove that?

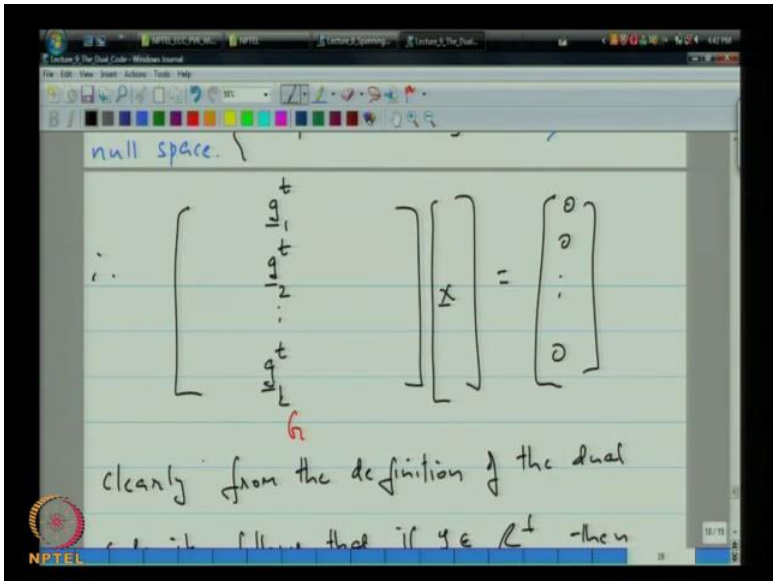
(Refer Slide Time: 41:43)



So, proof. Now… so, let x be in the null space of g. So, perhaps it is worth making a note for those one or two familiar with the terminology of a null space. Note, the null space of an m

by n matrix, A is precisely the set, something to write script n (A) to denote the null space is the set of all vectors x is was that A times x is 0. So, this is terminal notation for the null space. So, we want to proof that the null space of the generator matrix is precisely the dual code. So, let us x been the null space of g.

(Refer Slide Time: 43:44)

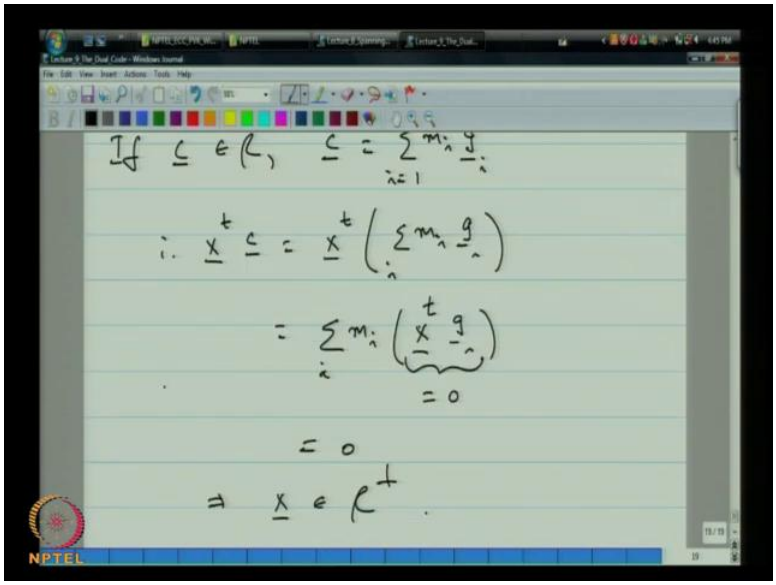


Therefore, it follows that g 1 transpose g 2 transpose g k transpose times, the vector x is 0. So, let means that the vector the inner product of the vector x with each of the rows of the generator matrix is 0, if something is in null space. Now, if you think about the rows of the generator matrix. So, let me write down g, just remind you that, this is our generator matrix, the rows of the generator matrix are a basis for the set of all code words. So, if anything is in the dual code, it must be it must have inner product 0 with every one of the elements of the basis.

So, in other words, if anything in the dual code must line the null space. So, that much is clear, clearly by definition of the dual code. So, let us make a note of that, clearly from the definition of of the dual code, it follows that, if y belongs to the dual code then y belongs to the null space of g. So, that much is clear. On the other hand, if something belongs to the null space then it is inner product with every one of these rows is 0 and from that which is

easy to see there is inner product with any linear combination of the rows is 0; and therefore, it belongs to the dual code.

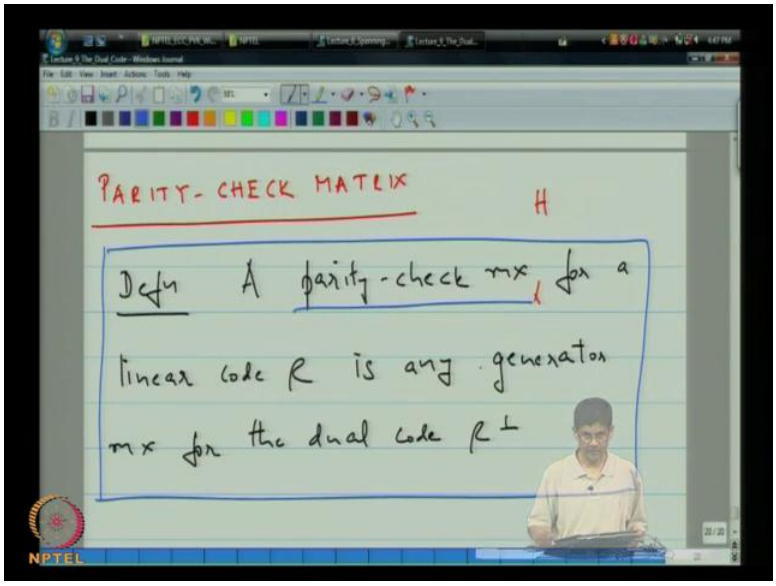
(Refer Slide Time: 46:15)



So, will put that down in writing, on the other hand, if x belongs to the null space of g then x transpose g i is equal to 0, if c is the code word then c can be written as sigma m i g i, i is equal to 1 to k. Now, therefore, therefore x transpose c is equal to x transpose sigma i m i g i which is sigma i m i x transpose g i. And we know the, we know this we know this to be 0 we know this to be 0. So, this entire thing is 0 which shows that x belongs to the dual code.

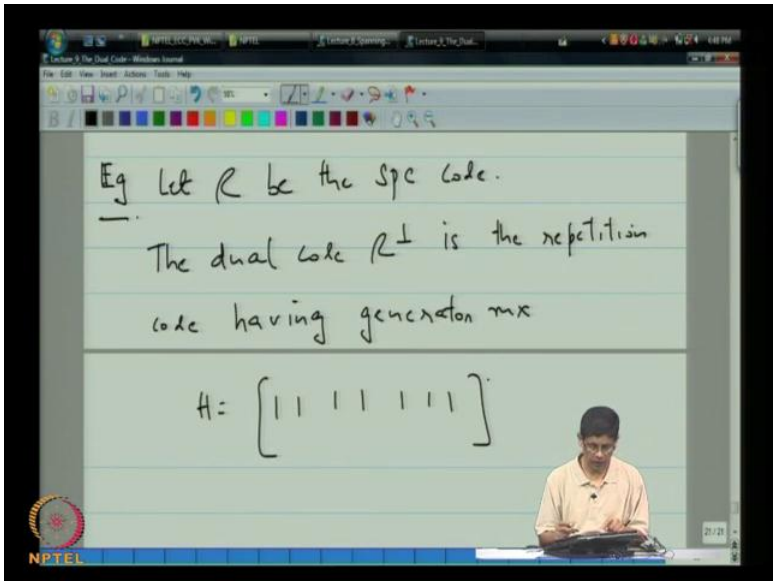
So, this is just by linearity, if then inner product of x is 0 with every row of the generator matrix, it follows that is inner product is 0 with every code word. So, what that proves is that, if x is in the null space belongs to the dual code and earlier we showed that, it something belongs to the dual code, it belongs to the null space. So, that proves with here.

(Refer Slide Time: 48:50)



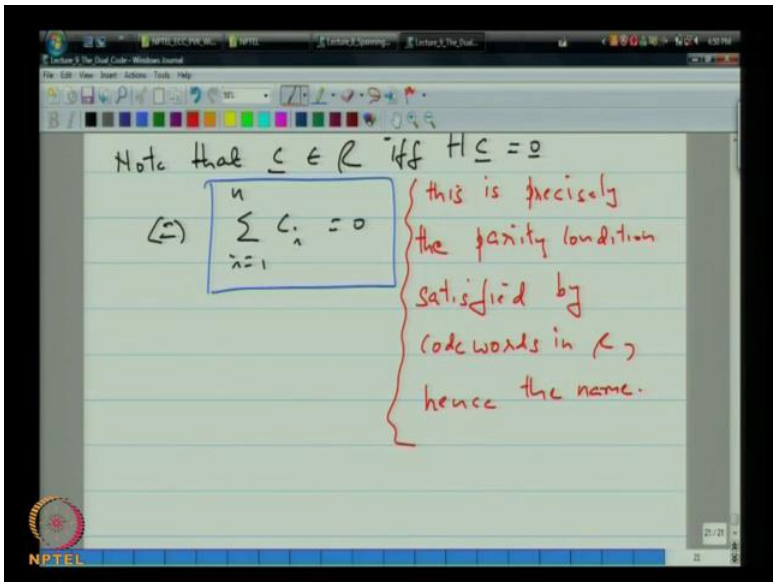
Now, we ready to introduce a second matrix. So, I will term this section the parity check matrix, definition the parity check matrix or excuse me, let say A parity check matrix because again they can be many a parity check matrix for a linear code c is any and will give this a name. So, will write each is any generator matrix is any generator matrix for the dual code c perp. So, one of the first thing that must (( )) is that, if I want a matrix that is connected to the code c, why is that after go to the dual code, but I think that will become clear from an example.

(Refer Slide Time: 50:52)



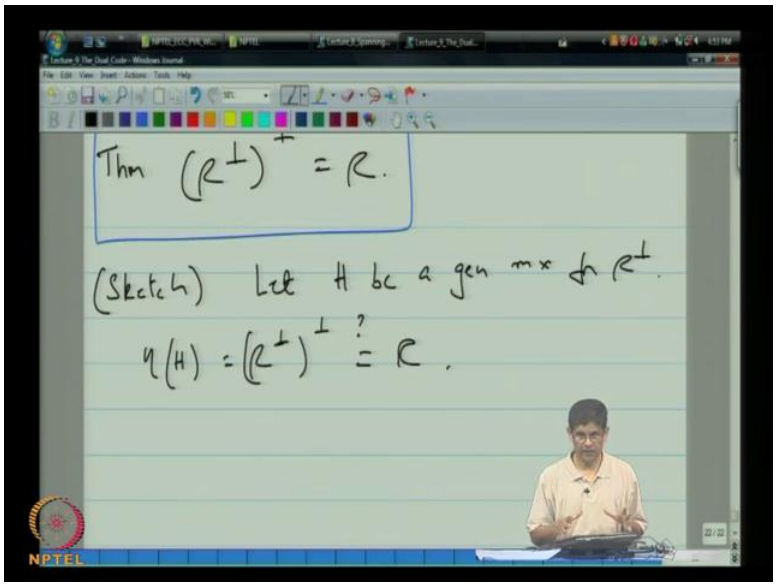
So, supposing we look at let c be the single parity check code, the dual code code c perp is is the repetition code is the repetition code having generator matrix H which is 1 1 1 1 1 1 1 and evolve we seen that. Now, note one thing, I am calling this the matrix H, because although it is a generator matrix for the dual code or focuses on the original code and we already reserve the symbol H to denote the generate a matrix of the dual code. So, that is the reason for calling at H. Now, still what we did, what I am trying to explain through this example is y z, that this matrix is called a parity check matrix of this code.

(Refer Slide Time: 52:31)



Now, the key point is that, the code in relation to this matrix, the code appears is the null space of this matrix. So, let us note make a note of that, note that c belongs to the code, if and only if, H time c is equal to zero which is another way of saying that sigma c i, i is equal to one is equal to 0. So, what this matrix is actually doing is it is slaying down, the parity check condition satisfied by all the code words in the single parity check code. So, this is precisely this is precisely the parity check, the parity condition satisfied by code words in c, hence the name. So, this is the region sorry this is the reason for calling each the parity check matrix of the code.

(Refer Slide Time: 54:21)



Now, what we like to show next is that, some not show, if you go enough time to finish, but let us give it to try today. So, theorem c dual dual is c that is the dual of the dual is the original code. So, note that c, so I thing perhaps this theorem is the best postponed until the next lecture, (( )) the writing down of it. So, let me just go over the arguments, we want to show that the, the dual of the dual gets you back to the code. So, rather than calling this the proof let me just call this a sketch of the proof. So, this is I will just, I will formally put down the proof next time the sketch is that.

So, let H be a generator matrix for c perp. We know that, the null space of H is the dual to the dual. So, the question really is, is this equal to the code? So, what will show next time is that by making an argument with regard to dimension which is essentially accounting argument will actually show that the two other same which will establish that the dual of the dual is the original code itself. Once you have done that then we look at means of actually finding in the generator matrix of a code is easy enough to find. So, we look at how to find parity check matrices. So, I think this is the good place to stop. Thank you.