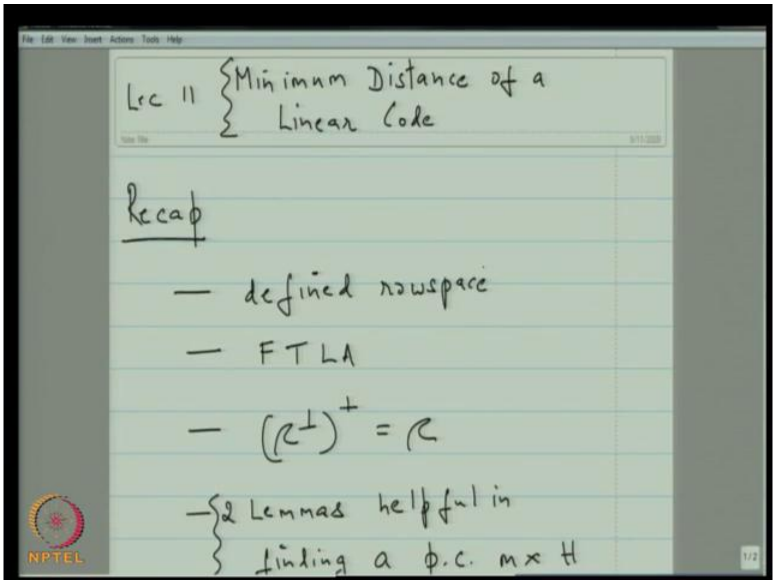
Error Correcting Codes
Dr. P. Vijay Kumar
Electrical Communication Engineering
Indian Institute of Science, Bangalore

Lecture No. #11
Minimum Distance of Linear Code

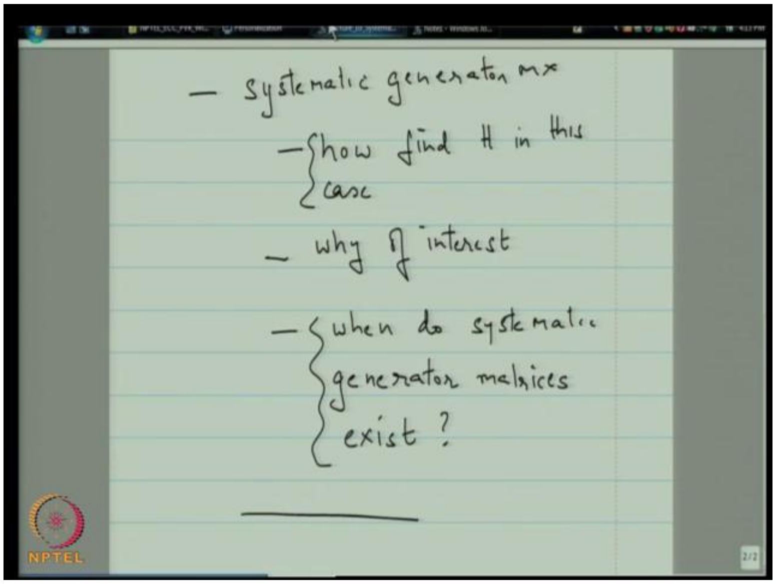
We will begin our eleventh lecture today, and I am going to give this lecture, the same title as the lecture I gave last time, because it turns out, that I did not get to what I wanted the main theme to be. So, I will go back and change the title from last time.

(Refer Slide Time: 00:36)



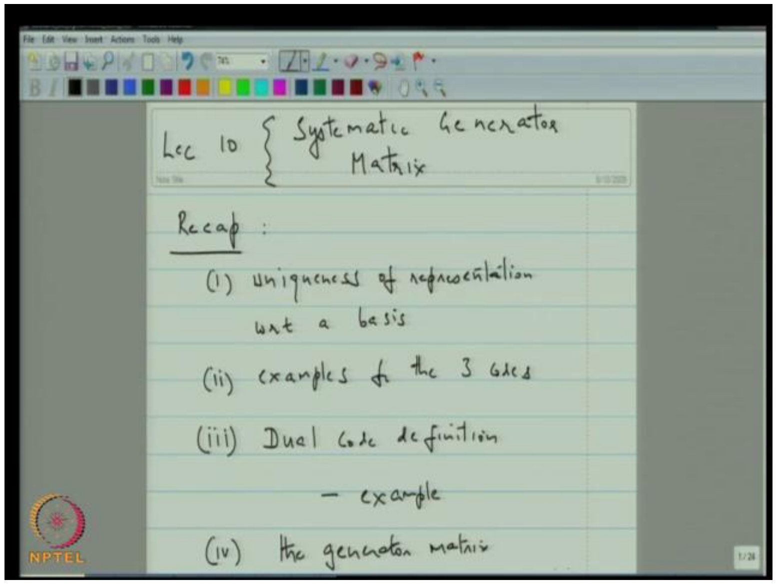
So, let us continue the title. So, we will call this the minimum distance, the minimum distance of a linear code. So, let me just begin with a recap; a recap. So first, we made a couple of observations relating to linear algebra, we defined what is meant by the row space. Then I went through the fundamental theorem of linear algebra; after this we proved that the double dual is the code itself. We next focused on lemmas, we we provided two lemmas helpful in finding parity check matrix H for the code.

(Refer Slide Time: 02:20)



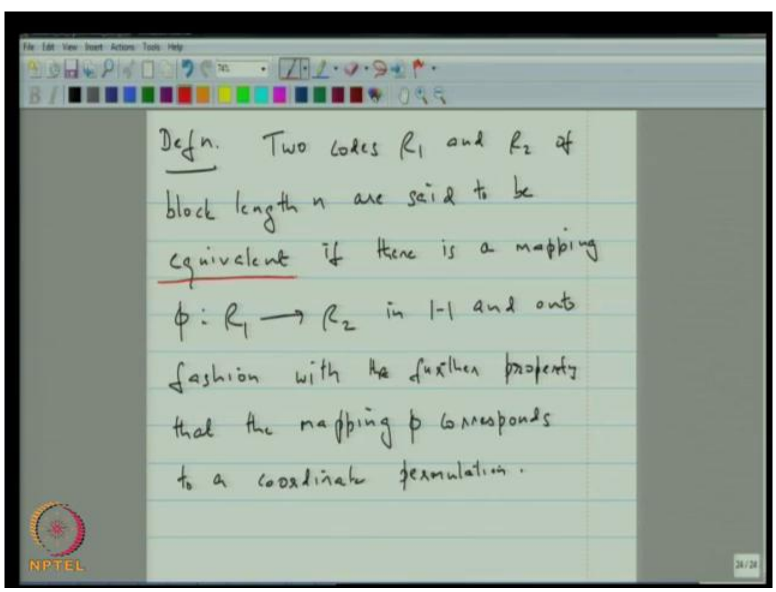
And then, we looked at what it means to talk about a systematic generator matrix. And so this is, for this particular case, we showed how find H in this case. Then we talked about why a systematic generator matrix is of interest in the first place; why is it of interest to have such (( ))? And then finally, towards end of the lecture we started talking about when do such matrices exist; so, when do systematic generator matrices exist? So, this is about may be (( )) and as I told you, I am going to go back and I want to change the title of our last lecture.

(Refer Slide Time: 04:17)



So, we will rename it now and again, this is making use of technology, so let us take advantage of that and I am going to call this in hindsight a systematic generator matrix because that turned out to be the focus for lecture. So, let me save that and get back to today's lecture, which has the title of finding a minimum distance, the minimum distance with linear block code.

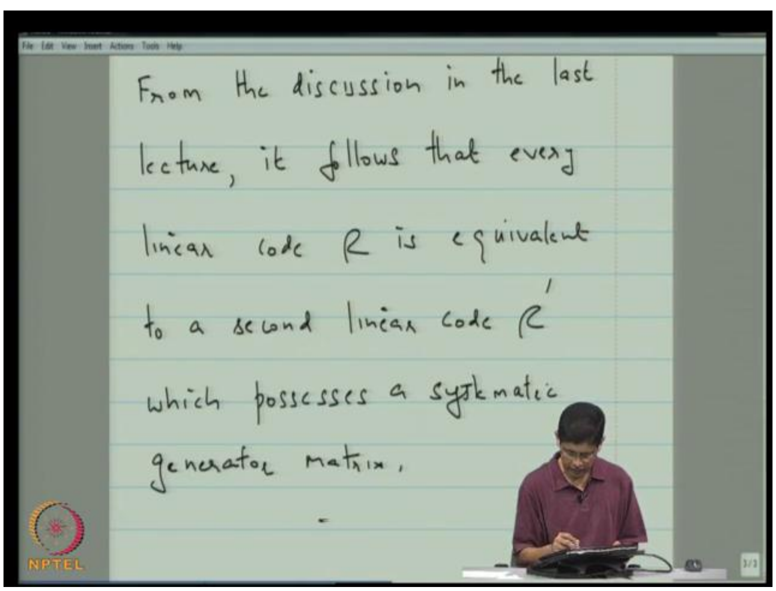
(Refer Slide Time: 05:22)



Now, so, just taking a look back again at the last lecture. Here, towards the end what we actually said was, in, we were face to the question, does every code possess a systematic generator matrix? And the answer we gave was, that if your matrix is k by n and if the first k columns are a full rank, then, then the, there, there exists a systematic generator matrix, but if not it is not like the code is hopeless in terms of being systematic because you can always find some other set of symbols with respect to which the generator matrix is systematic, meaning that you can actually shuffle the columns of the generator matrix and, and, arrange it into a form, so that the first k columns are linearly independent and then, that generated matrix would be systematic. However, it would not be a systematic generator matrix for the original code, but rather for one whose coordinates are shuffled. So, because of that I will just close the topic with this statement.

We will just look at the last slide of our last lecture, which said, that two codes of block length n are said to be equivalent if there is a map, which is one-to-one and on to and which corresponds to a shuffling of the coordinates or a permutation of the coordinates, alright.

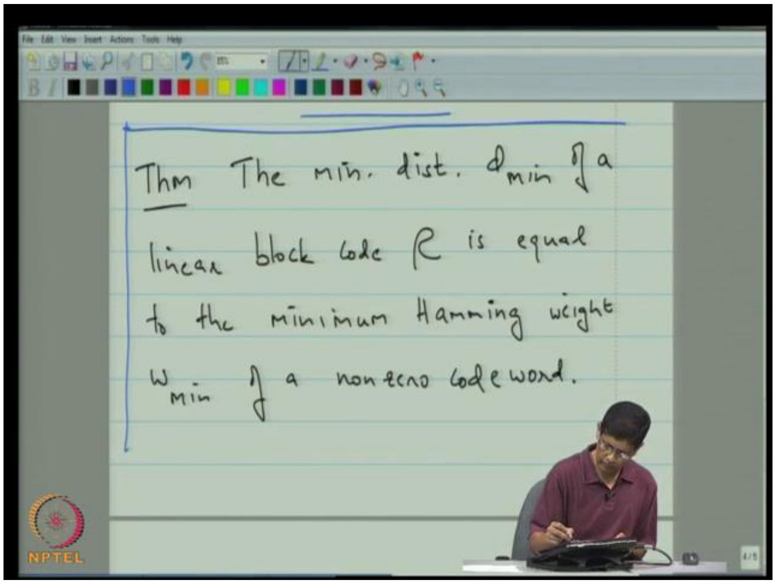
(Refer Slide Time: 06:58)



So, let us go back here and so we will conclude the discussion by saying, from, from the discussion in the last lecture it follows, that every linear, every linear code c is equivalent, is equivalent to a second linear code c prime, which possesses a systematic generator matrix.

So, that concludes the discussion on that particular topic. And now, let us move on to talking about how one would go about computing the minimum distance of a linear block code. I will just pull up my tool set, that I use here, which is very helpful.

(Refer Slide Time: 08:59)

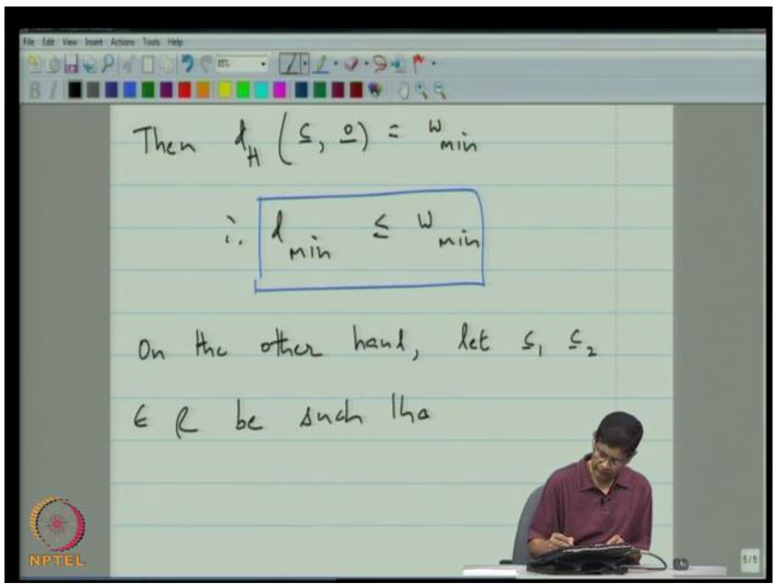


So, the minimum distance of, of a linear block code and we will begin straightaway with the theorem. The minimum distance d min of a linear block code c is equal to the, to the minimum hamming weight, to the minimum hamming weight w min of a non-zero code word. And the proof…

So, of course, let us say, that you have a code that contains ten code words and then, supposing you have to find the minimum distance of the code and you did it in a brute force fashion and say, let us say it was a linear code. What you could do is, you could look at all pairs of code words within the code, which should be ten choose to and then you would have to compute ten choose to, which is forty-five different pairs and then you would choose the minimum, but with linear codes. There is a major simplification, you do not have to do that, you can just simply find it by looking at these ten code words or rather nine because in a linear code word this is always going to be one, which is zero.

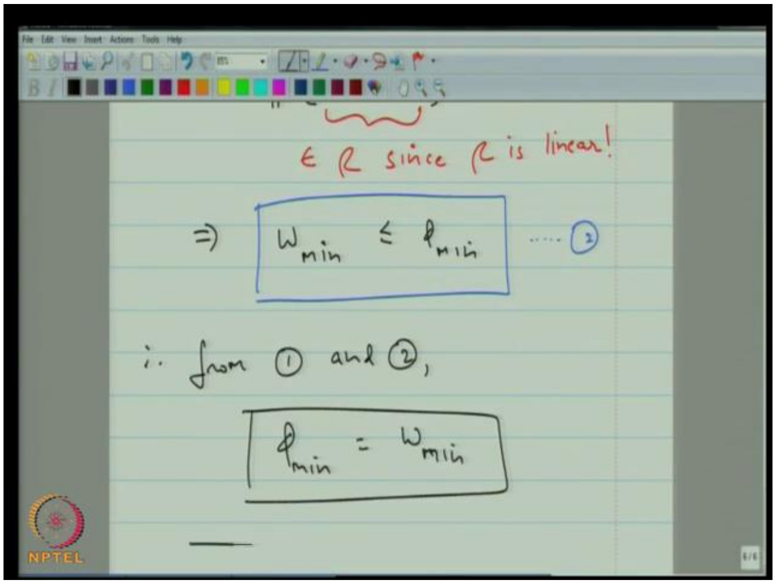
By the way, if you are thinking, that how can a linear code have ten code words you are absolutely right. A linear code word will, can only have code words, a number of code words equal to a power of two. So, let us say, that is, a linear code with sixteen code words, one of them is zero, so you only have to check the weight of the remaining fifteen and the minimum weight will then give you the minimum hamming distance. So, this is a big simplification, which holds only in the case of a linear code. So, let us see how we prove that and then we will quickly look at some examples.

(Refer Slide Time: 12:12)



So, proof, let c have hamming weight c equal to w min, then that means, that the hamming distance between c and 0 is equal to w min and so now, we found a pair of code words whose distance is w min. So, from this it follows, that therefore, the minimum distance of the code must be less than or equal to w min. So, that is the first observation. On the other hand, on the other hand, let c 1 and c 2 in c be such, that d H of c 1, c 2 is equal to d min.

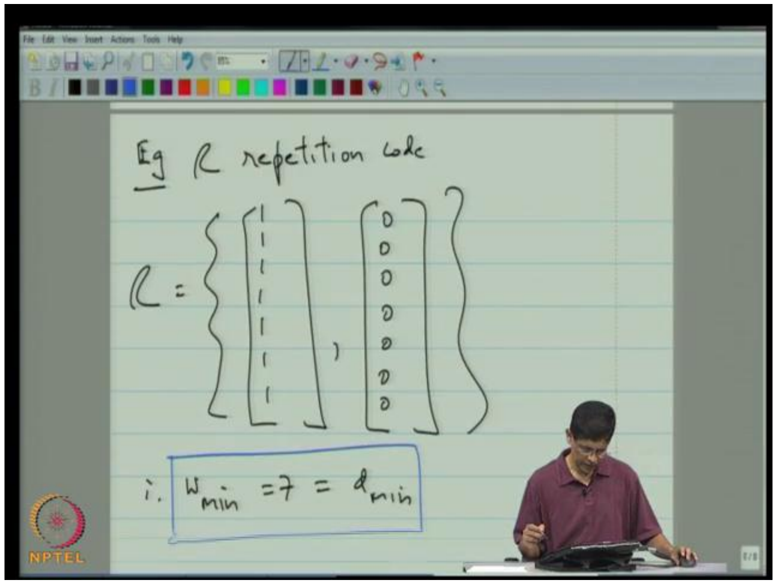
(Refer Slide Time: 13:38)



But this implies, that the hamming weight of c 1 plus c 2 is equal to d min, why is that? Because the minimum distance, the distance, hamming distance between a pair of code words is precisely equal to the hamming weight of the sum of the code words. So, that means, that you found and this belongs to the code, this belongs to c since c is linear. So, that implies. So, this implies, that w min is less than or equal to d min.

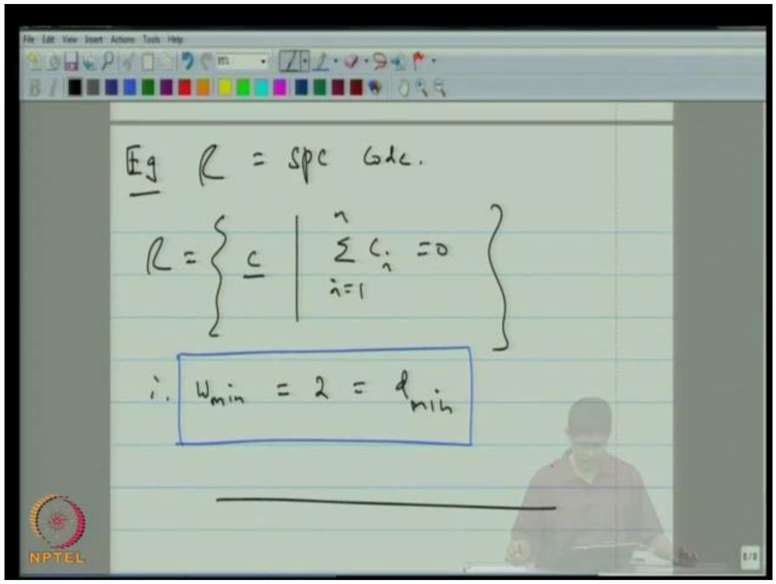
So, we have two equations, one which says and let us call this one, which says, that the minimum distance is less than or equal to the minimum weight and two, which gives us the reverse inequality. Therefore, from one and two we have, that d min is equal to w min and we are done. So, the minimum weight is equal to the minimum distance.

(Refer Slide Time: 15:15)



Let us apply that in an example. Supposing, we take c to be the repetition code, then we know, that c only contains two code words: the all 1 code word and the all 0 code word. So, when we compute w min of a code, of course, we must look at the minimum hamming weight of a non-zero code word. So, from that it follows, that therefore, w min is equal to 7 is equal to d min. And we have seen that before, but I just want to illustrate how much easier it is to find the minimum distance if you use this. Is t naught So, clear here. It will become little clearer in the next and perhaps even more clearer in other examples.

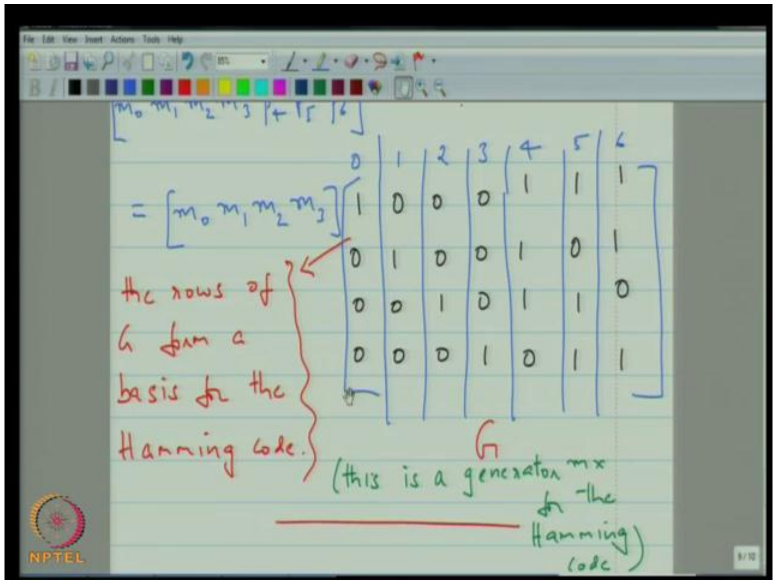
(Refer Slide Time: 16:28)



Let us look at another example. So, let us, c be the single parity check code. So, this is, so c is precisely the set of all code words having the property, that is, c i, i is equal to 1 to n is equal to 0. So, it is clear from this, that if you want the sum of all the symbols to be 0, then they must be at least to, once for this to happen, therefore, w min is equal to 2 is equal to d min. So, we have dealt with our two examples.

So, the natural question is, well, what about the hamming code? Was not that our third code? It turns out, those in the case of hamming code there is an alternative method of finding the minimum distance and so, let us see how we handle the hamming code. I will take you, take us back to the last lecture where we might actually have, let us see if I can find it, see in this lecture or the one before that, here it is. So, I am going to, now I am going to select the page and copy it so that we can use it in our current lecture. So, let us put that down here.

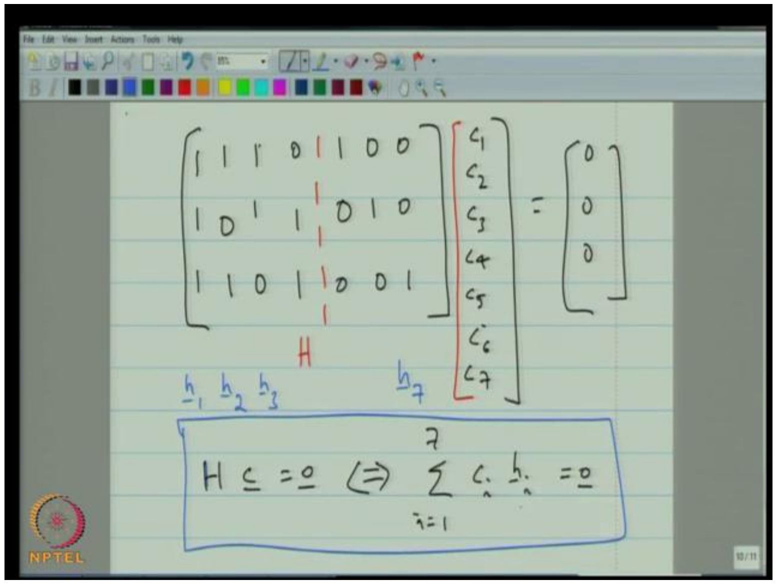
(Refer Slide Time: 18:36)



So, so from an earlier lecture we saw, that the matrix, that you see here in the middle of your page, this one, is actually a generator matrix for the Hamming code. So, now we also want to use the fact that this generator matrix is in the form of a systematic generator matrix. So, let me, let us remove some of these lines, which we do not need. Then, we see, that this part is the identity and this part is the p and from that we know, that we can actually write down in the parity check matrix of the hamming code.

So, perhaps just, so that people do not get confused. Let me erase all the redundant bits of information on this page, it is redundant for this lecture. So, this series is throwing to be a little temperamental here, but it is ok, alright. So, we have the generator matrix here and now based on this I can actually put down a parity check matrix for this code.

(Refer Slide Time: 20:44)



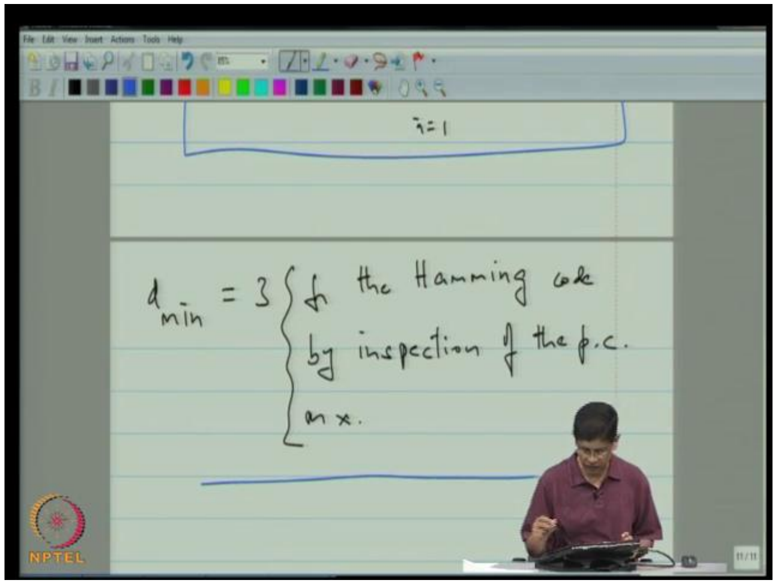
And therefore, H is equal to p transpose i n minus k, which in this case turns out to be, so I need to, to take the transpose I have to change that rows to columns. So, it is 1 1 1 1 0 1. So, 1 1 1 1 0 1 1 1 0 0 1 1 1 1 1 0 0 1 1 and then beyond that we just have the identity matrix. So, this is then our parity check matrix for the hamming code. And now, if I look at the parity check equation, so this is h then, which says, that H x is the H time c is equal to 0. And if I call these column vectors of the matrix, let us call them, maybe I should choose the different color, so let me write this in blue. h 1, h 2, h 3, all the way up to h 7. Then it follows it follows, that H times c equal to 0 if and only if sigma c i h i equal to 0, i is equal to 1 to 7. Now, this is an important equation because what it actually says is that every code word represents a linear dependence relationship amongst the columns of the h matrix. So, we are interested in finding out the code word, that has the minimum hamming weight and which is non-zero.

So, in other words that saying, well, what is this smallest set of columns within the parity check matrix of the hamming code I can find, which is dependent? So, let us go back to the parity check matrix and I am going to actually remove this red for a fear that will confuse, cause confusion. So, here is just the matrix itself and you notice, that there are seven column vectors and another thing. So, if you are in class I would have asked you this question. Would you notice about this matrix, is there anything special about the parity check matrix of the hamming code? And in fact, the observation that we will make in a second is an observation, that will allow to generalize from this particular hamming code to the general hamming code.

Now, going back to this matrix you see, that all non-zero three tuples, there are 2 to 3 minus 1, which is 7 non-zero three tuples and all them occur here, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0. So, all of them occur precisely once and if you are looking for the smallest set, which is dependent, you know that no two are dependent because all of them are distinct, but certainly, it is easy to find examples where three of them are dependent. So, for example, if you were to add, let us say, if you have to add this first column to the second column, right, you would get 0, 1, 0 and you would find that here. So, what that is telling you is that there is a code word in the hamming code, which has the following symbols, which has the 1, 1, 0, 0, 0, 1, 0.

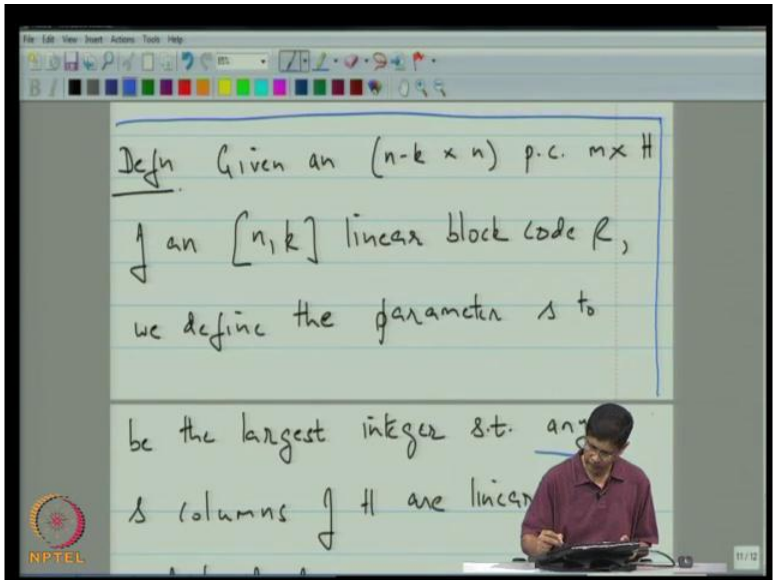
So, you see that you can go directly from a dependence relationship amongst the columns of the H matrix to a code word. So, that means, that when, so again let me repeat what I had said earlier. If you are interested in finding out non-zero code word of minimum hamming weight, that is equivalent to asking the question, what is the smallest size of a dependence set within the columns of a hamming parity check matrix, and the answer clearly is three here.

(Refer Slide Time: 26:39)



So, from this point of view the minimum distance of the hamming code is 3, so I will just write this very briefly by saying, that d min is equal to 3 for the hamming code by, by inspection of the parity check matrix. You might say, well, that is too brief an explanation, but that is fine because we will actually explain this in detail later. So, this is in some sense a preview of a general theorem, that will actually proof, which is, which, which is going to take us to the second method of finding the minimum distance of a linear code. The first method says, work, find the minimum hamming weight of a non-zero code word. The second method is telling us look at the columns of the parity check matrix and from that sometimes it is easy to find the minimum distance or at least it is feasible.

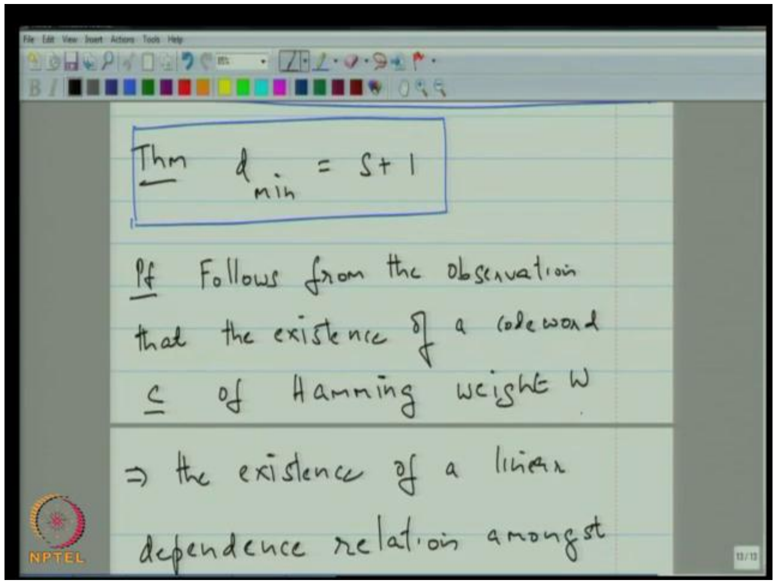
(Refer Slide Time: 27:46)



So, for that we, we need a definition. So, we are going to introduce a parameter s. So, given an n minus k by n, parity check matrix of an, lets also give this a name, so for parity check matrix H of an n, k linear block code c, we define the parameter s to be, to be the largest, to be the largest integer such that any, and this is important, such that any s columns of H are linearly independent.

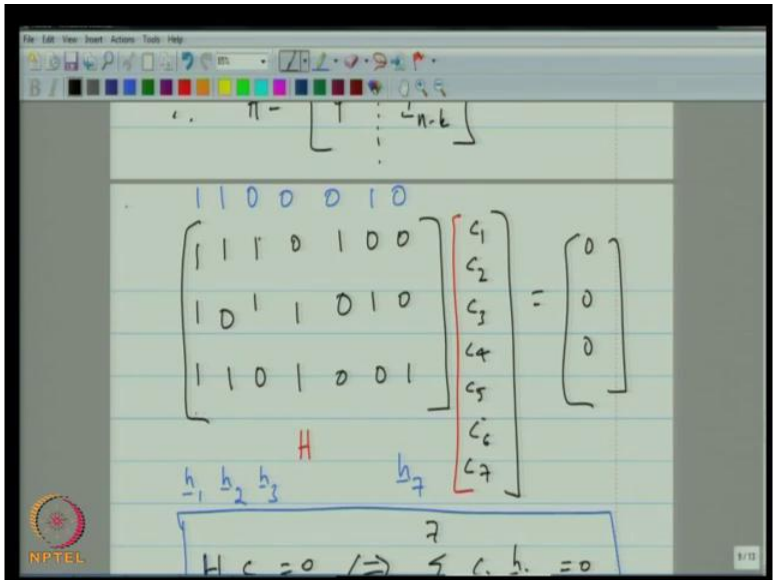
Alright, now you might say, wait a minute; I thought you said that the minimum distance was length to the smallest number of columns that is dependent. Well, that is very closely related because if you find the largest set, which is always linearly independent in you increment, that by one, then you will get the smallest set, which is dependent and that leads to our next theorem, theorem d min is equal to S plus 1.

(Refer Slide Time: 30:42)



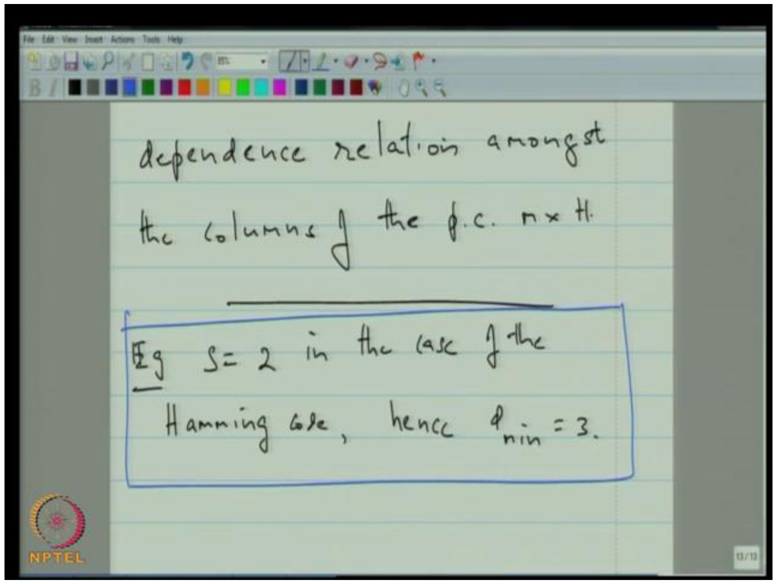
So, very simply stated and the proof is very simple, follows from the observation, that the existence of, of a code word c of hamming weight w implies the existence of a linear dependence relation amongst the columns of the parity check matrix H. So, that is the proof. And let us do one thing, just to make the proof clearer let us go back and apply to the hamming code. So, here is the hamming codes parity check matrix and the theorem says that d min S plus 1. So, all that we need to do compute S.

(Refer Slide Time: 32:46)



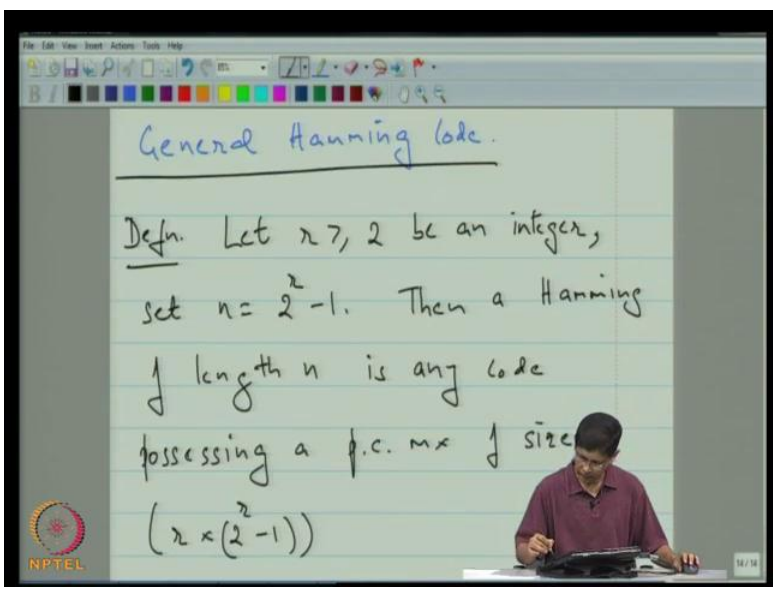
Now, s is the largest integer such that any S columns of the parity check matrix are linearly independent and you can see by inspection, that any two columns are linearly independent because they are dependent only if they are the same and none of them, other than S cannot be 3, because you can find subsets of three columns, which have dependent. For example, the 1st, the 2nd and the 6th are dependent, as we saw earlier because the sum of the 1st and the 2nd is precisely the 6th column, alright.

(Refer Slide Time: 33:38)



So, example S is equal to 2 in the case of the hamming code, code, hence d min equals 3. Now, we have seen that earlier, so this is an alternative and perhaps the simplest way of actually finding the minimum distance of a hamming code and you can actually use this to define a family of codes and this family is called the family of hamming codes.

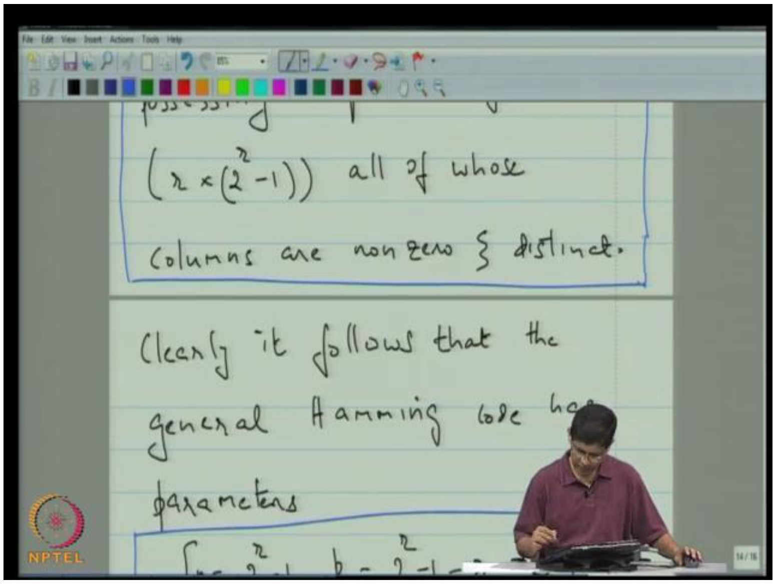
(Refer Slide Time: 34:40)



So, I will just simply call this the general hamming code meaning, that the code, that we looked at earlier was the specific hamming code because it had the specific (( )) seven. So, in general we would have the following definition. Let r greater than or equal to 2 be an integer and then set, set n equal to 2 to the r minus 1, then a hamming code of length n is any code, possessing a parity check matrix of size r by 2 to the r minus 1. All of whose columns are non-zero and distinct.

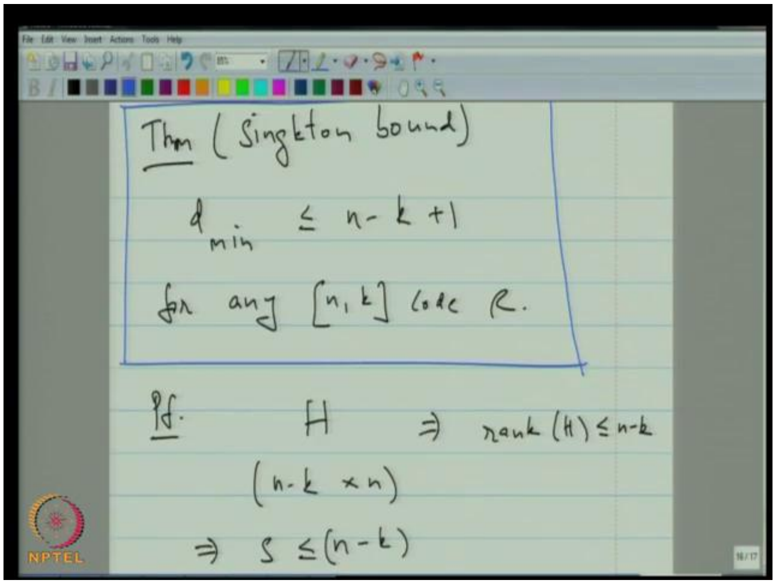
Alright, so, so when, again going back to the example code here, so here rth parameter, r is equal to 3, so the length is 2 to the 3 minus 1, which is 7 and all the columns are distinct. So, if, for example, you have to replace r equal to 3 by r equal to 4, then the hamming code would be defined in terms of a parity check matrix, whose, which is 4 by 15 matrix since r is 4 2 to the r minus 1 is 15. So, it would be a 4 by 15 parity check matrix and the columns would be all distinct. It did not, it does not matter, which order and distinct and non-zero and does not matter what order you put the columns and it will still be a hamming code. And so, next, so that leads us to the question, what are the parameters of the hamming code? So, I will just, this does not require a proof.

(Refer Slide Time: 38:05)



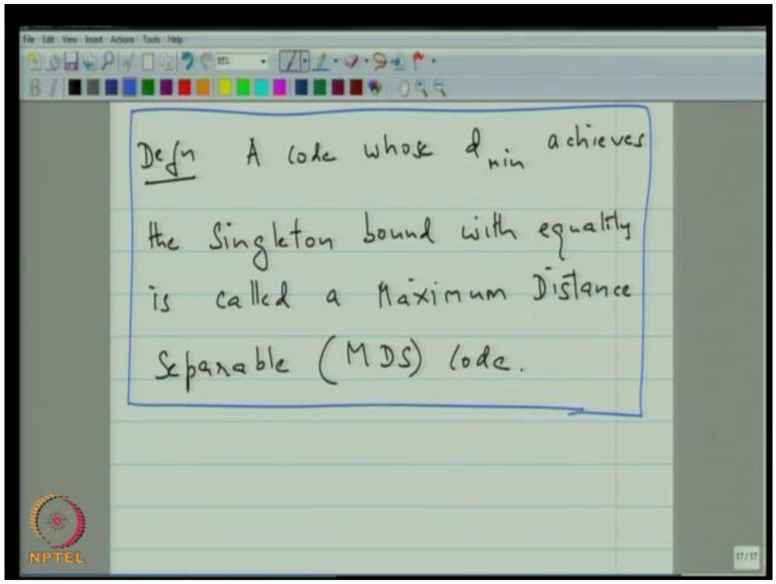
So, clearly it follows, that that the general hamming code, code has parameters n equals 2 to the r minus 1 k equals 2 to the r minus 1 minus r and d min equal to 3. So, if this is not clear to you, then i would, so just you, to work this out because it is simply a consequence of the nature of the parity check matrix of the hamming code. Now, this observation about the minimum distance of a code being equal to s plus 1 where s is the largest integer such that any s columns of H are linearly independent, leads to a bound, which is called the singleton bound.

(Refer Slide Time: 39:38)



So, theorem singleton bound d min is less than or equal to n minus k plus 1 for any n, k, code c and where is that, well, that simply because your H, a parity check matrix H is n minus k by n. So, it follows, that from the size of H, that s cannot be any bigger than n minus k because you know, that the rank of H is less than or equal to n minus k. So, from this it follows, that d min is at most s plus 1. So, it is at most this, so that is the proof.

(Refer Slide Time: 41:24)

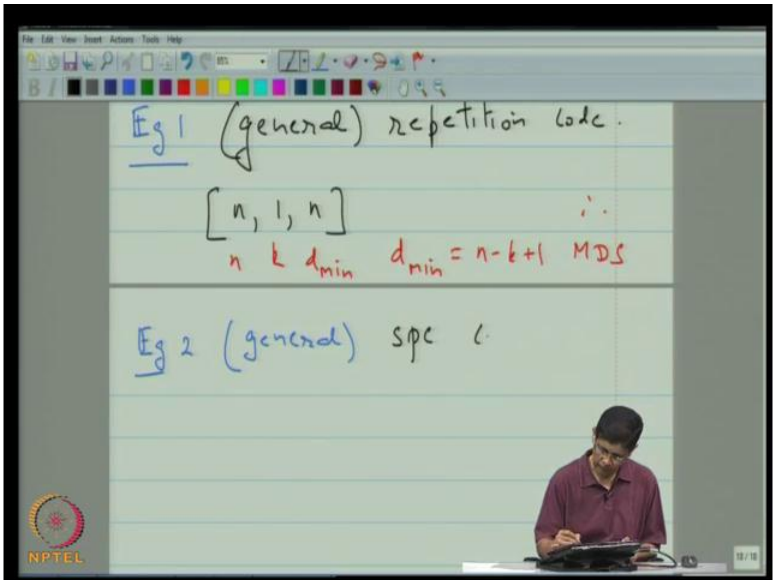


Now, and so, here is a definition a code whose d min achieves the singleton bound with equality is called a maximum distance separable code and that is abbreviated MDS code. Now, the reason for this term is quite clear because what we are saying is that if a code is MDS, then the minimum distance is as large as possible, which means, there in some sense you are trying to keep the code words in the code as far separated as you possibly can, of course.

So, that leads to the question, is it possible to construct these maximum distance separable codes? The answer is yes and no; yes, it is possible, but the codes, that you can construct in the binary case or not very interesting. Excuse me, it turns out there in the binary case, the only possible MDS codes or either, the repetition code or the single parity check code is no other code, that is, MDS. However, it turns out, that if you look at non-binary codes, which we have not studied up to now, that in the case of non-binary codes it is possible to find MDS codes. In fact, even though we have not talked about these, I am sure many of you heard of the class of (( )) codes. So, these are the codes that I used in which are, in bytes produce. For example, in compact disc and in a deep space communication, for example, these are well known and very powerful family of non-binary codes, which happen to meet the singleton bound with the quality and they are called therefore, they are the examples of

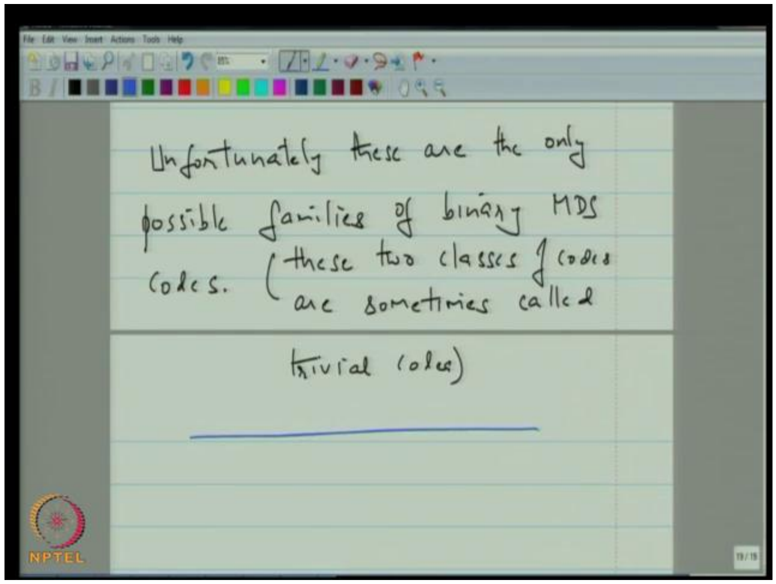
MDS codes. So, let us go ahead and see, why it is, that how it is that the repetition anomaly a, the single parity check code meet the bound with equality.

(Refer Slide Time: 44:25)



So, example one, so we will take the repetition code. So, in general, so this is let us say, the general repetition code. So, let us say with this is the general repetition code. So, this, such a code has parameters n, 1, n. So, since this is n k d, it means, that d min, this is d min of course, is equal to n minus k plus 1. Therefore, it is an MDS code. So, the general repetition code for all the symbols are the same and they are still only two code words is an example of an MDS code.

(Refer Slide Time: 46:10)



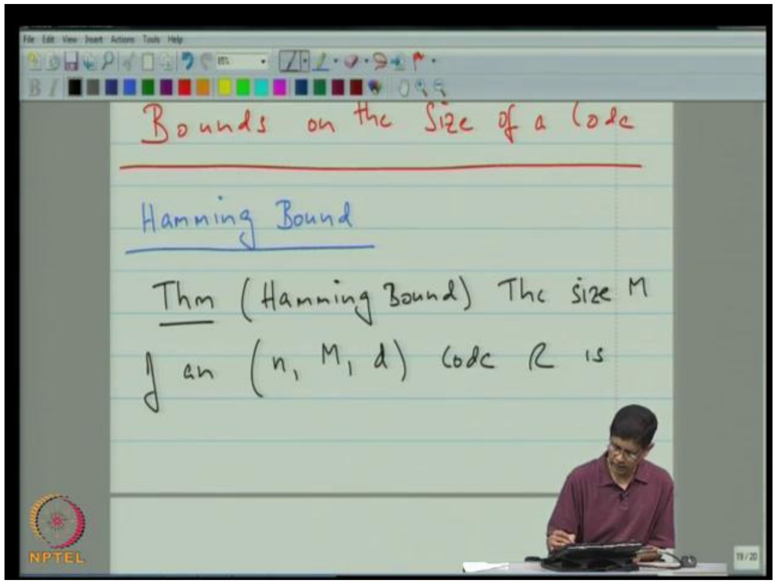
Now, similarly, if you look at the example of a, of the general single parity check code, then its parameters are length is n, the dimension is n minus 1, the minimum distance is 2, simply because if you take a look at the set of all n tuples, there are 2 to the n of them and half of them are even parity, half of them are odd parity. So, the number is 2 to the n minus 1. So, the dimension is n minus 1 and the minimum distance is still 2 because the minimum hamming weight of a non-zero vector is still 2. So, here, again this is n, this is k, this is d min. So, here again d min is equal to n minus k plus 1.

Therefore, MDS and from this you might be tempted to conclude, that perhaps it is easy to construct MDS codes, but the fact of the matter is, there exist no other classes of MDS, binary MDS codes. These are the only possible families of binary MDS codes and since there minimum distance is, either the minimum distance is very small or there dimension is small, these codes are called trivial codes; these two classes of codes are sometimes called trivial codes.

Now, trivial does not mean, that they are of no use. It just means that from an academic, academic point of view they are not very interesting, alright. So, that concludes our quick discussion of bounds of of, sorry, of the minimum distance of a code and related topics.

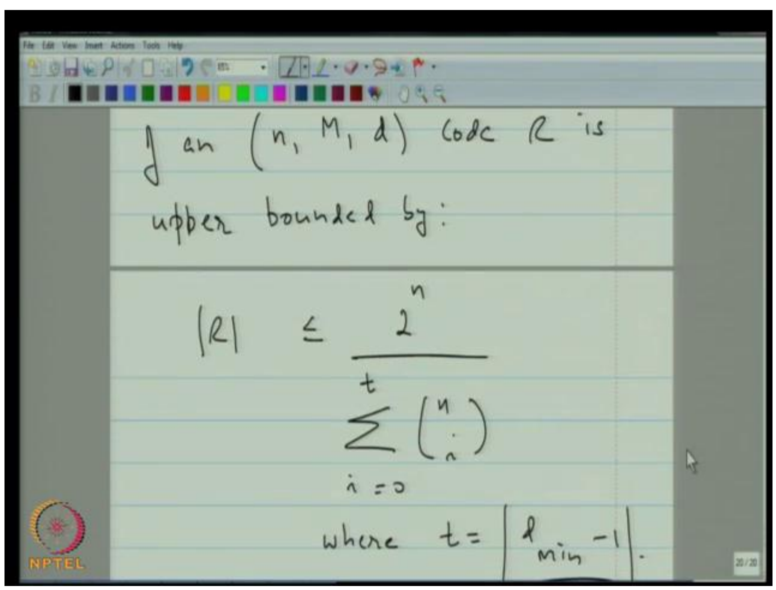
Now, we will go on to talk about, will introduce a new topic, which in a way you could view as continuing this theorem. This theorem provided as a bound, which relates some of the parameters of a linear block code. So, now, we are going to give other bounds.

(Refer Slide Time: 49:24)



So, bounds on the size of a code and there are two bounds in particular, that we will actually discuss. The first bound is the hamming bound.

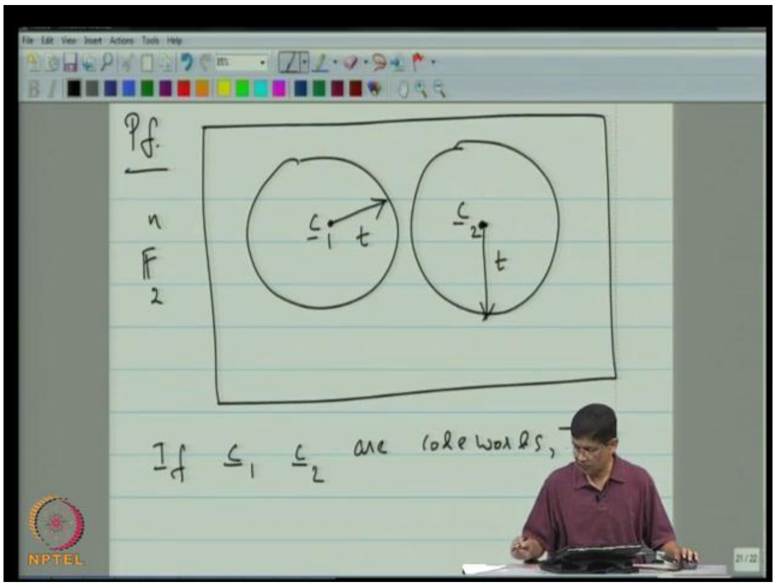
(Refer Slide Time: 51:05)



And the hamming bound says, that says, that the size M of an n, M, D, code c is a upper bounded by c less than or equal to 2 to the n times the sum, I goes from 0 to t n, choose i where t is equal to d min minus 1 upon two flow. So, this is what the hamming code says.

Now, so, how does one go about proving that? The idea is very simple. By the way when I say, I guess this notation here, this notation here is the floor function, which means the t is the largest integer, which is less than or equal to d min minus 1 upon 2 and assume most are familiar with that.

(Refer Slide Time: 52:35)



So, how do you prove that? The proof proceeds as follows. Let us say that this is an abstract depiction of the set of all n tuples and there are 2 to the n of them. So, they are scattered, allow a in here and now supposing, so let me erase that since i do not want these to, sure. Supposing I have two code words, supposing I have a code word c 1 and have a code word c 2, and have other code words as well. If I draw a sphere, a hamming sphere of radius t surrounding each of these two code words, then we know from a lemma, that we showed early on, that these two fears are disjoint if c 1, c 2 are code words.

Then so I see, if that we just have about a little over a minute left, so perhaps I should just quickly summarize and if we do not have time to finish the proof, I will continue this next time, it is not a problem. Perhaps it is best we will summarize in what we looked at today. So, the main goal was to actually show means of actually computing the minimum distance of a code. So, we showed that you can compute it by computing the minimum hamming weight and then, by computing the parameter s and saying d min is s plus 1 that led us to the general hamming code, as well as, to the singleton bound and to MDS codes. After that we begin our new topic studying other bounds on codes. We started the hamming bound, and will continue from that point onwards in the next class.

Thank you.