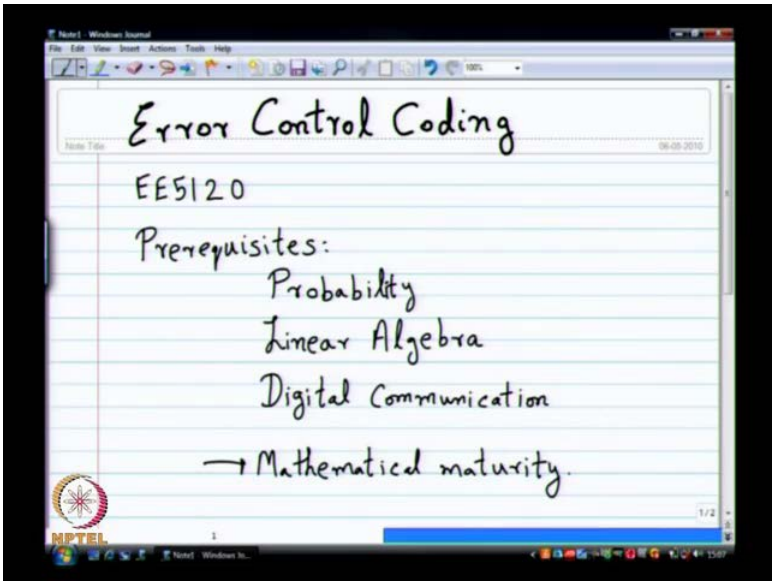
Coding Theory
Prof. Dr. Andrew Thangaraj
Department of Electronics and Communication Engineering
Indian Institute of Technology, Madras

Lecture - 1
Introduction to Linear Block Codes

(Refer Slide Time: 00:14)



So, we are all set. This is the first lecture in Error Control Coding which is EE 5120 according to our new numbering system. Hopefully, all of you are in the correct class. Yes, it is a recording class; so you would not have come wrong to this class. That is not a problem. So, what is this course about? This this course is about error control codes. If the whole, the entire, the terminology is new to you, if you seriously rethink doing this course assuming you have enough background to know what error control codes are.

So, those are some pre-requisites. So, I am going to start this course by talking about pre-requisites. This is particularly important. Please pay attention to this. So, the things that I am going to list down that I am going to list down now are very important for you to get a good grade in this course. So, by good grade, I mean anything other than u it is a good grade, so if you are if you are not sure if you are not done well in these courses in the past seriously, very seriously reconsider being in this class.

So, what is it that I will assume that you know I will assume you know probability not spelt it correctly see word. So, I will assume do you know probability random variables

random processes, so I will not do that you will need to know it as a fairly good level. So, for instance in the previous courses 3, 5, 6 for under graduate students and say 511 for the graduate students if you got less than a b, so very high chance you will get a u in this course. So, think about it very seriously before you take this course and then in the past it has happened particularly with B.Tech students taking this class and large numbers like it is happening.

Now, there are there were significant fraction of those students got a u and after lot of pleading it did not change anything so this is a very serious warning for those guys when you know you are doing several other courses you will not be very sincere. In this if you keeping doing tutorial sheets today's before the exam you are big trouble in this class it is not like previous classes many things you are doing in this course you would have never seen before. So, serious warning take that very seriously think other people be lastly, so it is not a problem probability is one thing and the other thing I will require is linear algebra, so this is one more thing that I will assume to a large extent.

So, usually it turns out that most people think that they know linear algebra let us see linear algebra. I know exactly what it is there is no problem usually that is good enough k more or less in this course at least k we would not do any very seriously in algebra, but it is good to know at least something very well.

For instance, if you done the mathematic linear algebra course you should be otherwise you might need some reading again. I would not have time to go into linear algebra at depth in this course some other thing will not do any specific questions on these two things. So, another desirable though we would not really need it extensively and I would not assume this too much is a course and digital communication.

So, I understand few of you might have already done this course may be a significant fraction might have already done this course in digital communication may be there is also a significant fraction which is may be doing it. Now, should be also if you are not doing digital communication at all what should happen is half the time you will not understand the motivation behind what I have taken about k. I just introduce something, and I will keep on talking it makes sense may be as an independent entity may be as an entity may not be able to relate it to practice of digital communication.

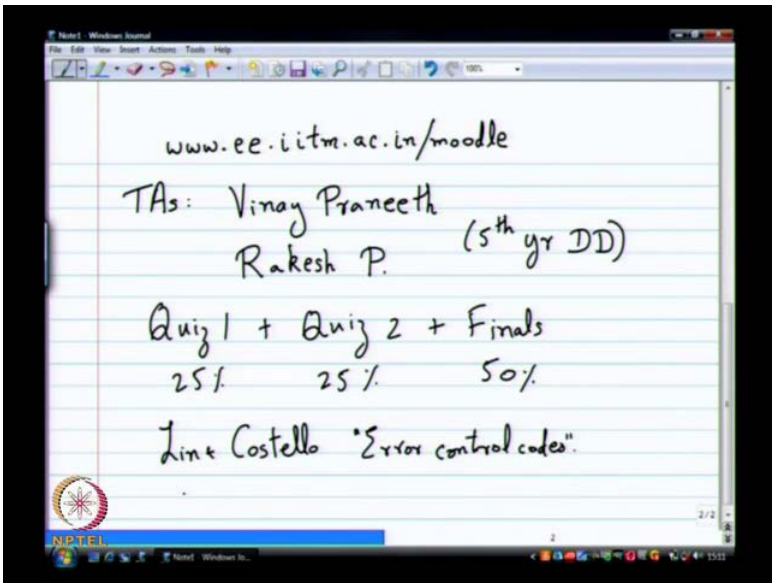
So, it is good to have done this course or you should be doing it this semester, so for the undergraduates I think there is a 419 which you should be doing or and for graduate students there is a 611 course. So, if you are doing this in parallel it should be more or less so that is pre requisites like I said I am starting with pre requisites not with the intention of scaring you a little bit. So, do not take it lightly, it is quiet important that should it is highly serious course and in general there are there are some aspects in this course which will require.

So, for want of a better term reasonable amount of mathematical maturity, it is impossible to define this re precisely, but I am assuming you will know what I mean. So, things like theorems and proofs should not scare you too much make a statement and try to prove it you should know why how the proof is working k more or less. For instance, one of the standard things that is time to show one set is equal to another set is what how do you show that. So, you start by showing one set is the sub set of the other then you show the other set is the sub set of this set which means these two are equal.

So, things like that are there are more complicated versions of this I am just giving you a simple example if you have never seen any proof where that was used before should be a little bit scared of this course. So, you might you might get seriously left out, so for instance the problems in quizzes will involve proofs like that and I will expect a precise proof you cannot just say tentatively this, so it is the answer, so not acceptable in the in the exams.

So, for instance in 356 for several questions people answer like that in probability you just tentatively these two are independent. So, you multiply and get the answer such things will not be possible, so you should get used to it primarily for the purpose of answering questions writing it precisely. What are the arguments, how one follows the other, so those things can I do in this course. I am assuming here comfortable all right everybody is with me on this, so that is all about preview questions we start fill down one page already go to the next page and I will give you some administered information.

(Refer Slide Time: 07:09)



First thing is the website for the course web page is hosted and can module the EE department's module. So, if you not heard of this website write it down go just visit this web page there will be a log in there you have to log in there you have to login with your mail login name and password. Then you have to click on EE 5120 add a control coding and you have to enroll, enrollment probably be open for the next 1 or 2 days after I close the enrollment. So, if you do not enroll within the next one or two days you will not be registered as someone who is doing this course. This means you would not get a grade, so make sure make sure you register it is important, I will put up some information here.

The assignments will be uploaded on this web site, the grades will be available through this web site what else, and there is a forum which you can use for posting messages. If you like, it is not the most popular forum in the world, but something you can think about in case you are interested the TAS will also be registered members of this web page, I cannot say the TAS. They might also be members of this web page as well so that brings me to TAS for this course one is Vinay, Praneeth both of them are not here, so hopefully they will show they have registration some system other person is Rakesh P K.

So, these two guys are the TAS for the course you free to approach them, I am assuming you would know how to contact them both of them are fifth year dual degree students. So, both of them are fifth year, it is like the display driver is giving me some problems, but it is fifth year dual degree students. So, they will be available, they will be in the lab,

also in case you need to contact them, you can e mail them so that is TAS and about exams. There will be two quizzes quiz 1 plus 2 two plus finals will be very simple, weight age 25 percent 25 percent 50 percent.

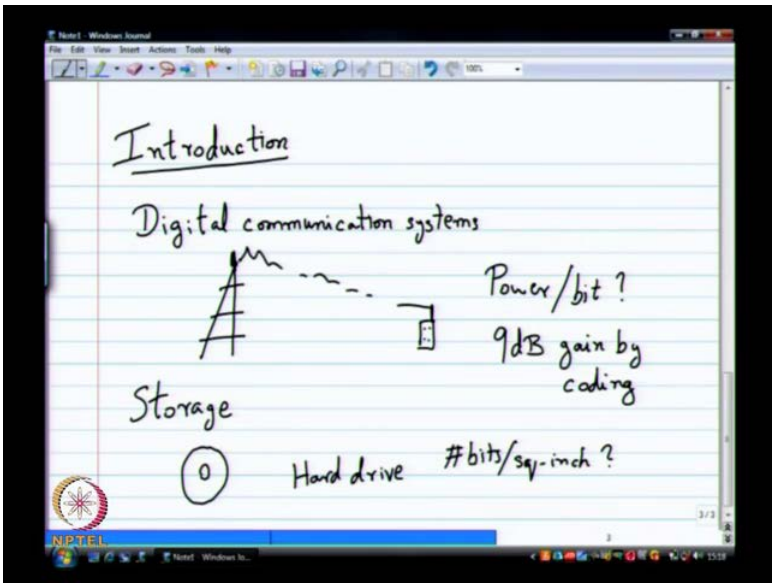
So, there are I mean for instance and the past I have given some tutorial exams to help out people who are not doing well this course there is no place for such things. So, I am assuming you are going to do well that is my assumption you are interested genuinely you will attend every class all these things you will do. So, those are assumptions that I made at the beginning of this class if you all are not going to be following that, then you will be in trouble in case of just 25 percent 25 percent 50 percent in the finals.

So, the module web page also has information on the syllabus outline lecture wise split of what I am going to do in each lecture it also has books. So, I will mention briefly the books that are a book by line and Costello which is probably available in cheap edition the first edition is available in cheap edition you might be able to buy it, I think the title is error control codes. If I am not wrong, check on this this is a good book to buy, it has enough information on it the other books are mentioned on the web page.

If you go to the web page, you will see the syllabus and outline the text books are listed there in case you can you can buy those most of the text books are expensive. You have pay approximate of 100 dollars for each of the text books, so if you do not feel like buying it you have to rely on class notes, means you have to attend every lecture k otherwise you will be lost.

So, that is about the text book and what else any questions about the administrative aspects exams grading how it is going to work etcetera no questions. So, this thing is doing all kinds of crazy stuffs, so I will if time is enough then we will start k any questions once again anything on how it is going to be conducted. So, the timings setting quiz I mailed people, so it is going to be Thursdays and Fridays 3 to 5 in this place. So, we will have two lectures happening any kind of 2, 1 hour lectures, the idea will hopefully have a break in the middle around four o clock for a few minutes. They will come and anything else I am forgetting something that is usually said in the beginning of the class.

(Refer Slide Time: 12:50)



So, let us begin, so this class will mostly be introduction will also get started on some course, but I would like to give a brief introduction to y coding is needed etcetera. So, two very popular engineering systems use error control codes as one of the key ingredients. The first one is thing most people will guess very immediately is communication systems particularly digital communication systems. So, those systems use error control coding that is the first application of error control codes, so on what way do they use error control coding is something that is of importance we will touch up on it. So, before we see it and grade detail, I want to briefly mention how it works from a very higher level. So, how does the digital communication system look, so the all kinds of pictures you can draw.

So, since it is an introductory class you can draw a picture and then I ask you what it is what is this it is a cell phone tower, so looks like latest example, then what is this cellular phones obviously both of them are not scare you know just drawing it. So, as antennas here antennas here then there is a antenna here one more eventually makes its way. So, important concern when you build a digital communication system as how do you reduce the power that you need to transmit one bit of information reliable that is main.

That is an interesting question you can ask, so what is the question I gave, how I needed power bit, so clearly I would like to decrease this. I would like to decrease this, this is power that is a thing i am sure is nice name and mathematics for such kind of logic why

would I like to reduce. We cannot just remove the battery in five minutes what is the point cannot keep on recharging point of being mobile is lost and eve on the wasting end.

Even though you might have power, there are several situations why am I going to conserve power may not have power at sometimes that happens a lot, so power per bit is an important issue. So, turns out if you do coding you can get up to 9 dB improvement in power per bit. This depends on specific type of channel etcetera, I am just giving you a 9 dB number term may be you have digital communication course, you will learn more about how this works out. So, this is the big number you know nine d b is large gain ten times reduction power, so you have to imagine that is the first application of error control coding.

So, by the end of this class this course you should know how this works how is it that in digital communication system you can get up to 9 dB of coding in what is the secret we had. So, one tenth of the power how do you transmit a bit reliably is the key word here though they have not mentioned it here I guess you need reliability another system where it is used very extensively storage.

It is a storage such as I am not going to draw a picture it just one favorite picture of mine that time. So, let me draw it just that one picture, then I will not draw anymore pictures what is this some distance DVD, VCD, CD, Bluray and all. So, these are say hard drive compute a hard drive, so here the unit of interest is number of bits per let us say square inch k so this is the kind of metric that you want to optimize. So, what would I like to do to this metric it should be increase, so I want this to be as large as I can possibly make it k.

So, means there are obvious reasons why you might want to do it, but to give you a simple example if you have a really long movie the HD version might want to fit it into one DVD supposed to 10 DVD. So, that that clearly depends on number of bits per square inch you might not want to do that same thing with the hard drive today hard drives are in every device a in fact even in mobile device. So, you want to put as many bits as possible k and i do not have a similar nine d b number here, but significant gains are possible by coordinator. So, the number of the amount of improvement you can do because of coding the number of bits per square is very significant in these two things.

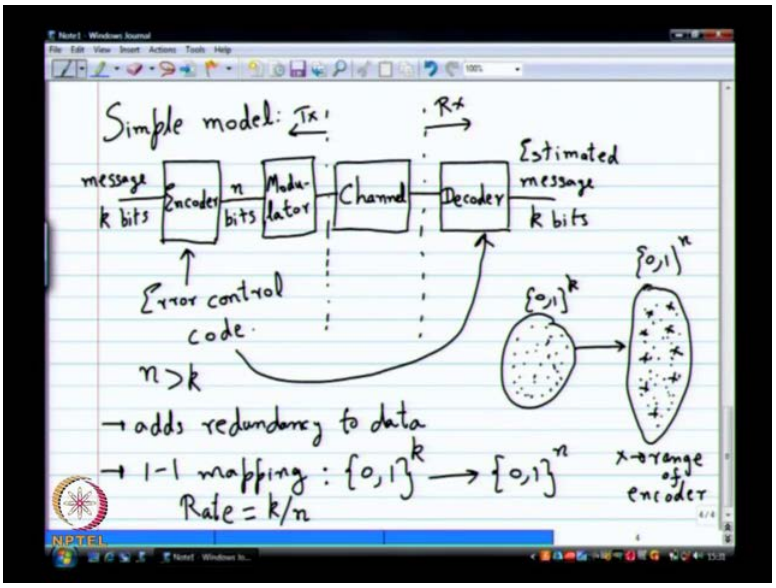
So, if you just look around this room or anywhere you go how many storage devices and how many communication devices do you think you get there. Everybody has it right, everybody has some mobile phone the room has at least two three computers and even in the home you have computers. So, the point I want to make is error control codes are everywhere. So, there where one of the most is of technology they find anywhere you go in most systems. So, of course communication is everywhere wherever is communication there is error control codes, where ever storage also there is error control codes.

So, you cannot get whatever you do k so it is good to know nice technology to know, so this a good picture it looks very rosy and nice, but to really understand how code work you have to do a far amount of mathematics. So, it is an entrance area since you have to know how the how several things look. So, after this we will plunge into some of the mathematics slowly, but anyways good to know in the first class at least that from a very high level it is nice to know about error control codes.

So, I am not going to ask for any questions at this point let us just two way at this point we will go to something more precise and then I will pose you questions, so that is applications. So, I think this is so probably the only slide I will have this kind of nice diagrams case and you are interested in modern art you might feel, so how does coding work. So, here is a here is a here is a very simplified picture of how a communication system looks and how error control codes work in a communication system. So, you might say why I am saying communication system and why not storage system it turns out storage also is like communication.

So, you write at one time and then you read a another time, so it is like a communication so they are suppose to happening at the same time at different points in space. You are at the same pole place, but you reading and writing at different times, so both of them can be written down as communication system that is what we will do in here also.

(Refer Slide Time: 20:07)



So, here is a very simplified model once again to understand at the very high level what happens with the error control codes. So, you have a channel which you going to communicate you leave I mean there is lots of distill communication single processing you have to do a lot of engineering you have to do to work with the channel like this. I am going to capture all of that and in a very simple blog diagram and simply say simply say what some model 8 this will say I do not have any space to write model 8 here.

So, let me write let me write it in two lines, so this modulator is basically something which does single processing to communicate into the channel. So, what error control codes do is you would have something called an encoder at the transmitter, hopefully you can see what I am writing is reasonably all right. My hand writing is not the best in the world, but at least I think it is reasonably legitimate, there is not no too much room for confusing.

So, the encoder sends at the transmitter k, so this comes from some error control code k we have not seen what it is said, but I want to show you how it fits into the system some error control code which will define an encoder is just a thing in the transmitter. At the receiving end at the receiving end receiver case remember this is the transmitter. So, everything to decide is a transmitter everything to this side is the transmitter let y everything, so this side is the receiver.

So, the decoder you would do the opposite you would have something called at the receiver am sorry you have the decoder. So, this would undo whatever the error control code encode at and enter the transmitter, so this also plays a role here k so what does the from a very big picture point of you what happens here. Suppose, you have a message which is let us say some k bits, the encoder would do some coding and convert it into n bits and this n typically would be greater than k most case would be greater.

Then, these n bits would be sent on the channel k so modulation is something which will do some second crossing to convert those bits into some signals which will go through the channel. They will be received on the other side and you to do a lot of crossing also the receiver I am not showing any of that directly putting on a decoder.

So, you going to do a decoding and you get back your message case you would get let us say estimated message. So, I have just put everything at the receiver as decoder clearly that is a very simplified view, you have to do lot of crossing if you can run decode, so essentially what does an error control code, it takes k bits converts.

It into n bits and you take have n s greater than k, so let us see if you just look at this what is the encoder doing, lots of technical terms the people are throwing for instance one of the complicated is discussing. Describing this is to say that it acts redundantly to the data, so not be very complicated, but it adds redundancy to the data. So, once again what is the point of adding redundancy remember my original motivation I want to decrease the power per bit.

So, that is the motivation, ultimately this will reduce the power per bit that is what we want to show, we will show that much later on the course that is the idea this is how the error control codes work. They would not add redundancy to the bits that you are sending and then so that finally, the power bit required after the redundancy is still lower than what to do what it would have been without coder so that is the idea of the coder. So, if you have never seen how encoders look this is how they would function in a digital communication system.

So, you take a message which is k bits so you have to remember some something like let us say thousand or something like that put k there imagine numbers like that. So, imagine like that then n would be 2000 it is something like that, so it is a lot of bits, so take bits and what it tend to end k usually. Even though that is redundancy, this will be one to one

mapping almost always unless you want to do something very extraordinary at the transmitter, this will be one to one mapping, so that is what the encoder would do.

So, if you have to come with an error control code or if you have to design an error control code or specify an error control code what should you do what should one do it is enough if you specify the encoder. Hopefully, I said the error control code specifies the encoder, so if you specify the encoder the operation is clear man how do you add the redundancy is all you have to specify. So, that is what we will see soon enough k how you specify how this redundancy is added.

So, that is the that is the way you specify an error control code and let me ask some quick questions based on just this encoder how many possible message s are there 2 power k messages. So, this one to one mapping will be from where to where 2 power k, k bit vectors to what 0 1, k 2, 0 one m k what do I mean by this is just fancy notation to say this is the set of all k bit factors.

So, it is the partition product to zero one with its self repeated k times in case if you do that you would get the set of all k bit factors. I have one to one mapping from the set of all k, k bit factors which has to include all the k bit factors. I cannot just drop some message right some message comes I cannot say I would not transmit. So, I have to include all of them, so how many vectors will there be in the range 2 power k again.

So, clearly it will not be what is called an on to mapping, so there will be vectors in 0 1 end which will not have a counterpart, so that is what it means really to add redundancy. So, if you this picture in the graphic rotation, if I say this is 0 1 all the message here all the points here are messages, after I do my encoding what happens I get a larger 0 1 n what happens in the 0 1 n, I have once again a lot of points.

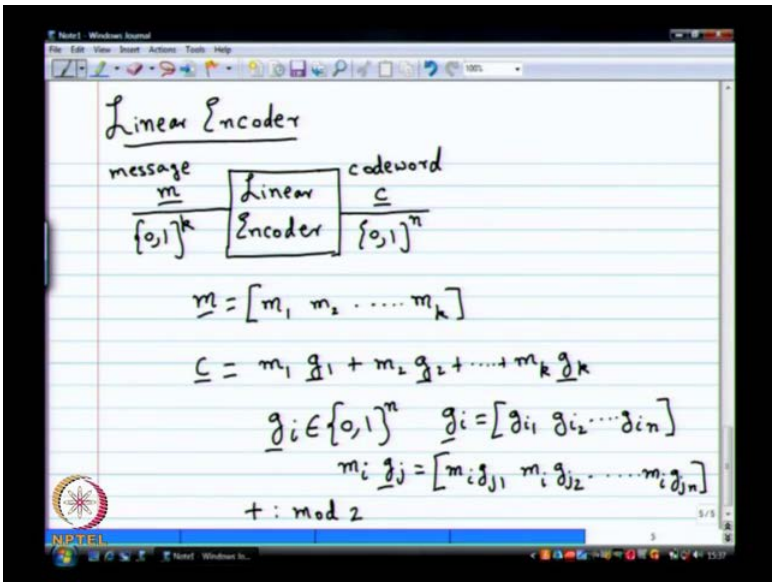
All these points are not are not are not really included only how many of them are included two part these are the range of encoder. So, this picture is a nice picture to keep in your mind, what does code you said adds redundancy to the data. So, after it adds redundancy to the data how does the whole picture look previously when you had no redundancy you had the space 0 1. All the points in the space where being used now after encoding you have actually a larger space 0, 1 and in which you are not using all the points k, how many to use 2 per k of them.

So, there is a very related definition which is called the rate of the code what you think is the rate of the code k divided by k, so this is the rate of the code and typically rate is going to be less than 1, so because n is greater than k, so you have all kinds of rates. So, initially we are going to talk a lot about the encoder, so as many a few can easily guess the encoder is presumably an easier problem, what would be a more difficult problem decoding part of it would be more difficult.

So, precisely for that is an look at decoders later, so we will start with the an encoder, well I will define a very simple encoder which is very commonly used. Then we will see how to study it properly is that any questions or comments at this point not really said when I said lot of power is for bit, it is number of bits total number of message not code numbers we divide by k is still hopefully the question.

So, he was saying the same power per bit what if I divide by n as opposite to k he is asking see if we divide by n that is cheating sorry divide by k because k is the main bits that you understand I have to divide by k. So, once again this pictures this simple graphical picture you have on this granular point fully being used. Suppose, to few points in a large space, so very important picture to keep in mind it require again and again k, so conceptually it is easy to understand course.

(Refer Slide Time: 31:27)



What is this simple encoder, I am going to describe it is called a linear encoder k, it is not the only way to encode I mean obviously it is not the only way to encode. So, many

other ways of doing it, but for starting now we will look at what is called a linear encode, so once again like I said we going to look at the encoder in more detail in the beginning. Then we will slowly move on to the decode k, so what does the encode I have, so it is it is going to take in k bits here.

So, what does a linear encoder do, it is what with us it means let us draw the picture it has a message m it seems the message m which is from 0 1 k and their outputs are never called the output. I have never said what the outputs of an encoder the outputs of an encoder at p are called code words k. So, code word c from 0 1 n k, so there are various ways of describing a linear encoder, so what I am going to do is take a very simple description. So, what the linear encoder would do is the following k the message m is going to be k bits, so I am going to call the k bits as m 1 m 2 so on till m k.

So, what the linear encoders going to do is going to compute the code word c by doing the following it is going to take m 1 multiplied by some vector which I will call g 1 and then to it will add m 2 times another vector g 2 so on till k times another vector g k k. So, I have to tell you a lot of things it is not done yet, but this is the general principle and you can see why it is called linear k k is linear because it is doing linear combinations of some fixed vectors.

So, that is why it is called linear this is the important principle to understand, but I have not explained what the g i's are, what are the g i’s are from 0 1 m k. So, they are fixed vectors which we will we can think of them as say bases vectors case we can talk think of them as some kind bases vectors. These are some vectors that give think I had of time, so then what else should I specify. Let me see think or needs to be specified everything is clear what is m 1 times g 1 bar what is that what is the operation scalar multiplication of people are I have, but anyways let me say one that is k.

So, each g i like I said is from 0 1 and so maybe it is g i 1, g i 2 so on till g i what n, so we can denote by each vector is like that k what is m i times g j what is this operation to each element of g j. I have to multiply by what is multiplication its regular integral multiplication think of the bits 0 and 1 as the integers 0 and the integer 1.

You are multiplying by the bit 0 or 1 which is again an integer 0 one just do integer multiplication basically 0 times 0, 0, 0 times 1 as 1, 1 times 1 is 1 again that is what you do. So, this is would m i g j one m i g j two so until m i g j n, so that is the multiplication,

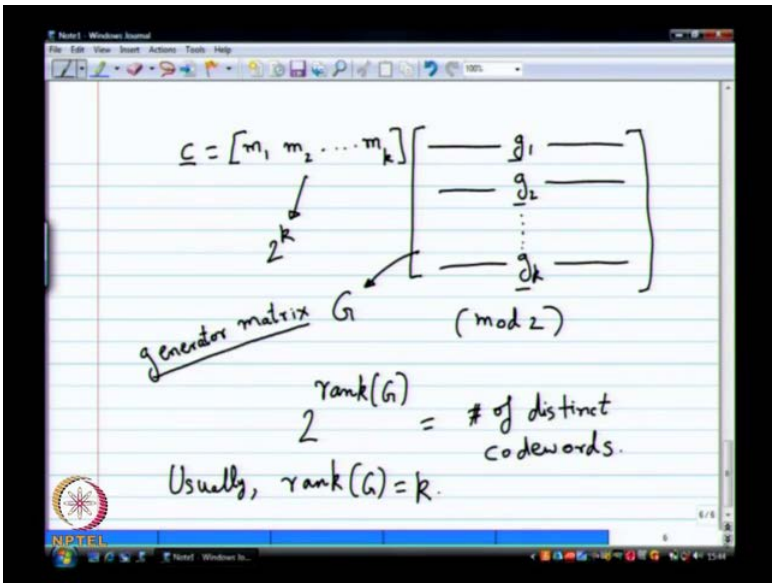
so if you are familiar with vector spaces linear vector spaces over fields. This is obviously the scalar multiplication in vector spaces, so I am not come to that yet we will see that slowly, but this is the operation that happens.

The next operation is plus what is plus, so I have to add, but then there is a problem here, I have already put it as equal to c has to belong to 0 1 n. If I add 1 plus 1 what will happen it will go to 2, so you are not supposed to do that you are supposed to do all these additions modulo 2, so that is the next thing so plus is modulo 2 addition. So, it is like binary hexa, so zero plus one would be 1 1 plus 0 would be 1, again 0 plus 0 is of course, 0 1 plus 1 would be 0.

So, you will get this so that is the idea behind this so the entire encoder is specified by what in this scenario the vectors g. So, which we can think of as the basis vectors, so the generating vectors or these are all names you can give it to them give it to them that is the that is was the vectors which defined the code c. So, as you can imagine your fundamental parameters of interest they reducing the power per bit etcetera will totally depend on how you choose g.

So, if you choose g when you get a good code, then it will reduce your power if you choose a very bad g, then nothing will happen because that code is not doing anything that is the idea this is what a the linear encoder would do. I am going to now write this operation in a slightly different way using matrices and vectors which is a very common way of writing linear encoders. Once again, all this will be very simple if you done linear algebra, but you have not done, it is worth it is worth seeing it.

(Refer Slide Time: 37:49)



So, essentially what is happening is matrix multiplication which is you can think of the messages m 1, m 2, m k multiplying a matrix which has what which has g 1 in its first row g 2 in its second row and so on g k in its last row. So, once again what is hidden in this multiplication is what how I multiply 2 bits and how I add 2 bits, that is hidden in this multiplication what am I supposed to do? Everything is done treating 0's and 1's as integers, but modulo 2 that is the main thing to keep in mind, so everything is done modulo 2.

So, whenever I deal with bits and do multiplication and addition this modulo 2 is will almost always be implied in this course and I will not write it too often. So, I will assume you are familiar with when you have done digital systems and courses like that and you would know what it means. So, this is done modulo 2, now I am going to ask a very difficult question it is particularly difficult question in this class, already you people are expecting difficult question how many code words will I get if I do this.

If I take all possible 2 per k here how many code words will I get how many distinct code words will I get will I get 2 per k distinct code words always all of I check take all these vectors as 0. You are assuming a lot when we say 2 per k right you are assuming something when you say 2 per k. So, you should be aware of what that assumption is, so if you know linear algebra this will be easy enough, but if you do not know this is something to work out.

So, if I vary this m one through m k through all 2 per k possibilities under what conditions will I get 2 per k code words. So, if all these vectors are linearly independent, so how do you guarantee how do you check that this characters are linearly independent are not what should you do to compute the rank how do you compute the rank. So, we have to reduce it into whatever row reduced to take along form how do you do that what is the crucial step Gaussian elimination.

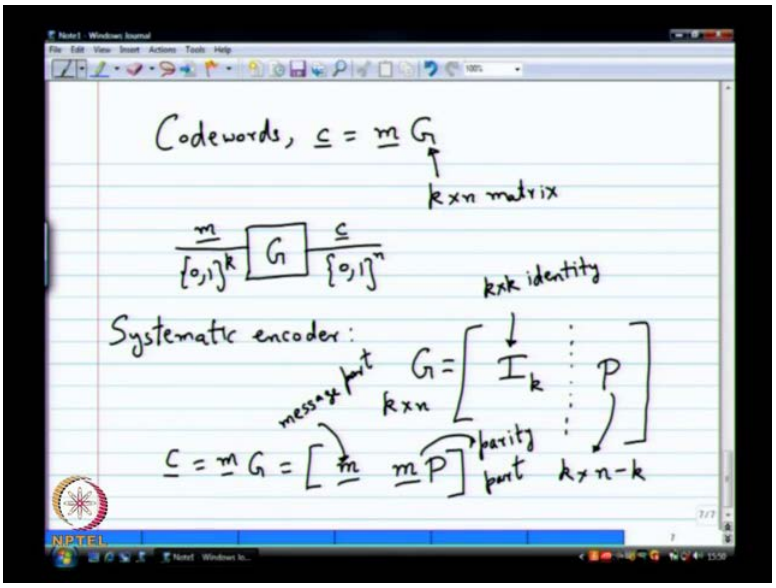
So, turns out you can do an operation combining the rows in a clever way which is called Gaussian elimination it does not change the rank it does not change the dependence properties. So, you keep doing it you will get to this remember once again when you do Gaussian elimination how will you add you have to add modulo 2. So, that is a twist in the process if you are not used to do in coming Gaussian elimination modulo 2, it will confuse you a little bit you may not have done it before.

So, we will see some examples later on, but the thing I want to point out is once you decide your vectors g 1 through g k. You can do Gaussian elimination and figure out how many of them are linearly independent and how many of them are linearly dependent on the rest. So, based on that you can find what is called the rank of the matrix the rank once again by standard linear algebra, you can sure will be at most what at most k will be can be equal to k it can never be greater thank.

So, that will be the rank, so if the rank is k, then what will you get you will get two per k in different code words distinct code words if the rank is less than k what will happen you get two power rank of k rank of g if this entire matrix is called G. So, if you denote this matrix as g in fact such a matrix is very popular that it is called in fact the G matrix, so you can call it as g matrix it is called what is called the generator matrix.

So, the code generator matrix g its mean by may be define this later on, but people know that this collegian made a matrix g and the rank of g clearly plays an important role, what role does this play, rank of g 2 power rank of g equals number of distinct code words. So, just purely based on intuition when you pick g as a k by n matrix and you want to send k bits you should pick rank of G to be k. If we pick rank of G less than k clearly you are giving up in the beginning its self when they are make making too many errors not you will not get anything see if can know usually rank of k is rank of G is equal to k.

(Refer Slide Time: 43:50)



So, a code words for a linear encoder a code words c generated by m times g where this g is a k by n matrix which is called a generator matrix so this is the idea of a linear encode, so this is the good thing to remember. So, I can for instance make a picture like this let draw m basically put g here we will get a code word c, so this is a generator I mean. So, another main aspect of this course is how you pick this G, so code design can be reduced to the problem of picking a suitable G.

So, at this point a good question to ask is, are we giving up anything by restricting our self to linear encode that is a good question to ask. We will postpone that question as far as this course is concerned we will pretty much look at only linear encoders. If you have taken information theory, you all of may be read enough about the information theory or some other course like that there people show that linear encoders are good enough for all most all situations.

So, it is enough to look at this special type of encoders they are quite good for communication purposes, so all most all purposes linear encoders are more than good enough so we will exclusively see such things any questions. So, I want to see some show you some examples, but before that we will introduce one more idea which is which is fairly which is fairly important it makes the examples much easier. So, there is something called a systematic encoder, so like I said in this course we have only going to talk about linear encoders, so I will drop the term linear.

So, let us assume that its linear, so it is something called a systematic encoder in the systematic encoder what you do is you pick g as follows remember g is k by n you pick g as I k that I k is what yes as you can guess its k by k by k identity matrix. These are just dots, these are not one, so this will just separate us, and then you would have a matrix p which is what would be the dimensions k by n minus k. So, this is the generator matrix in systematic form so this is systematic form, so that is why it is called systematic encoder so all these things all this terms are used interchangeably.

So, this will generate a matrix in systematic form, so one thing that this does is what is what is the one immediate simplification when you have a generator matrix in systematic form yeah rank is k I mean you just do not worry about rank anymore. So, otherwise you have to worry about doing, so this is also the row reduced take a long form kind of thing, so it is like that, so the rank is immediately obvious from the structure it is k.

So, you do not worry about it what is the other thing that you can notice on the systematic form first k bits of each code word will be equal to k. So, that is the other idea which is interested in showing in the systematic encoder c when you multiply m with g will take the form m multiplied by I k which would be m itself and then m multiplied by p, so this is called as a systematic form for the code word.

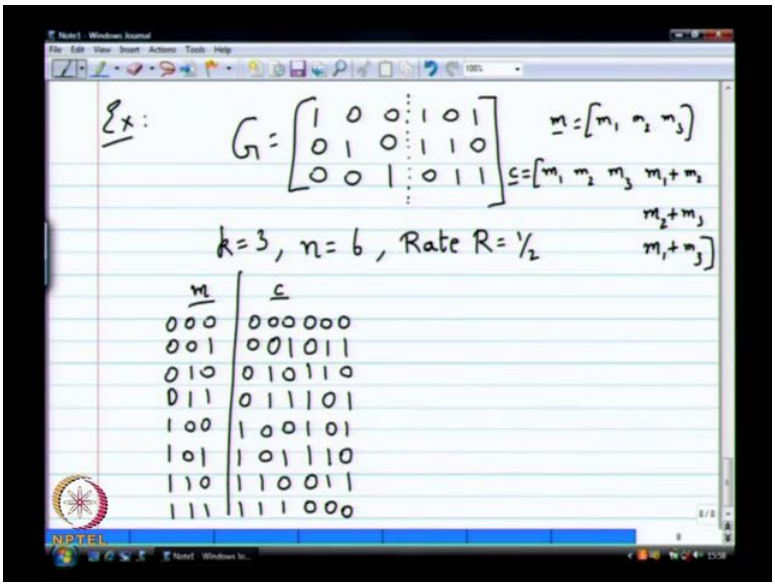
So, one nice thing which we will may be eventually see after a few examples is that any code any linear code can be put into systematic form. So, you are not anything by doing systematic, so that is the first thing which is very nice, so it is very simple property we can easily do that which you do by row reduction by which is very easy to prove also. So, we are not loosing anything by doing systematic any code can be put in systematic form.

So, it is not a problem, but systematic form is also particularly nice from, so many points other points of view in practice in practice in in single processing and all that systematic form is really nice and useful. So, that is why it is another reason why it is useful, so for one yeah so maybe we will see that later on its useful in practice it is all most of the course that you will see out there will be in systematic form as its being implemented. May be there are very few exceptions 1 or 2 here and there which might be in non systematic form most codes would be in systematic form.

So, in systematic form every code will have a message part message part and then the remaining part is called parity so this is called the parity part. So, the message part would be k bits and the parity part would be n minus k bits, so once you write the linear encoder in systematic form you can think of the linear encoder as explicitly adding redundancy to your data. You have the message which comes out by itself and then you add n 1 minus k parity bits to the data how are the parity bits formed why are they called parity bits are formed.

So, I will may be at this point it is not clear to you immediately why it is called as why it is called as parity. So, basically this m times p right while we have written it like this it is actually a its actually some subsets of the bits in m and doing the exhort of all of them we will see that in an example. Then it will become clearer later, so that is why it is called parity you are taking sub sets of the message bits and doing parity of that that is the idea there. So, let us see example in systematic form then we will see in non systematic form and reduce it to systematic form, so that is the two things we will try.

(Refer Slide Time: 51:00)



So first example we will take g as 1 0 0 0 1 0 0 0 1, then I have this fictions separate up which is clearly not the part of the matrix you just do it in the separate form 1 0 1 1 0 1 0 1. So, what are the various you can infer by looking at this generator matrix systematic what are the other parameters develop interest to me, sorry k is 3 and n is 6, good that some people know. So, that is what I want case all amount is very simple observation, so

you observe that k equal to 3 and n equals 6, so in particular the rate is going to be rate is denoted r several times, so say rate r equals 1 by 2, so rate half code.

So, one good thing to do always is to list all the code words of a code particularly for small codes you can do that if k is very large it is very difficult to do that. So, we would not do that but for small k it is good to list all the code words of the code, so remember my picture where I drew a circle with all the three k bit vectors and then code words as a sub set of the points in the n bit vectors.

So, it is good to draw that a couple of times and see what is actually happening when we do something like that. So, we will do we will do that here for the simple example we may not do it later on, but it is good that once to see to see a list of all the code words. So, let us make a table the table will have two columns, first column will have all possible messages then next to it we will write all possible code words.

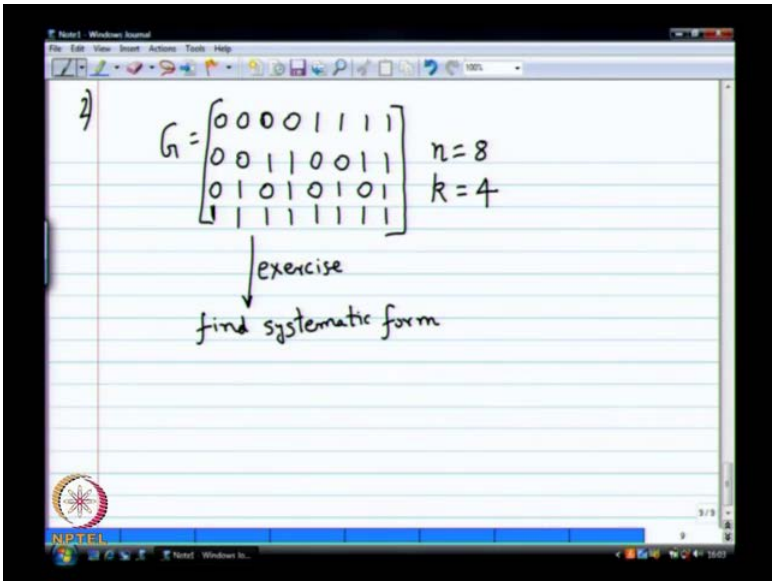
So, one nice thing to help you do this is to first write how the code word is formed in the general case suppose you have m as m 1, m 2, m 3 what will be the corresponding code word it is going to be m 1, m 2, m 3 that was easy enough. How will the force between be formed its m 1 plus m 2 it is good that everybody says m 1 plus m 2 because by now we are used to the mod 2.

So, this are going to be the sets of such m 1 plus m 2 what will the fifth bit m 2 plus m 3 what will be the sixth bit m 1 plus m 3. Now, I guess my previous comment about why it is called parity is little bit more clear, what is happening to the parity bits, how are the parity bits produced you take a sub set of the message bits. Then do the exhort of them, so how are the subsets specified by the columns of p, so where ever the ones are ones in the columns of p, then you which bits are getting exhort.

So, it is a very simple operation, but all these things are good to keep in mind sometimes it will help you visualize what is happening. So, I will give you couple of minutes when to go ahead and fill out all the message and code word that is quite easy to do, but it is good to do. Now, we will know major surprises in this list, so as you can see that there are 8 messages all 8, 3 bit vectors are messages. Then how many code words do you have eight of them all of them are distinct what did I do 0 0 1.

So, there are only eight code words, so how many 6 bit vectors are there 64 only eight of them are code words, so you have a lot of route. So, that is the redundancy that people talk about, so you just use eight possible 6 bit vectors out of this huge 64 possibilities there is lots of redundancy there and what else can you observe from here. So, I think I do not want to say anything more beyond this, so this example is probably clear enough we will come back may be more properties on this later on, now this example is any question on this example how we got it, etcetera.

(Refer Slide Time: 57:39)



So, the next 1 is a g in which now I am going to give a slightly more complicated example it is what have I done here did you remind you of something we must have done in a digital systems course. You must have written down something like this when you made two tables right so just listing out all possibilities in a different angle that is a correct from left to right.

Finally, I have added all once just to complete things, so what is k and n for this, so four you have to be a little bit careful you know I mean the reason is usually want to take k to be rank of g. So, once you know that the rank of g is 4, then you can say four otherwise it is always good to take k to be equal to rank of g. You know I wanted to take something greater than k n is clearly 8. I agree k in fact is four for this problem, but you should find the rank explicitly before you can be sure. I will give you a couple of minutes to get used

to this, but then we will see we will quickly see the row reduced form and see how it works this is g k we will determine.

So, the row operations I mean you have to do quite a few row operations here, so the first thing to keep in mind is you have to a bring a one to the first position and then eliminate all the other 1's. So, here you bring the one to the first position what should you do swap the last column with the first column. Then we do not have to do anything else right and then what you will do get a 1 to the second 1 then do some next additions row operations, it is a series of steps and I would highly recommend that all of you try this.

So, you go back tonight in the evening and then try it, so it is quite important if you have not seen something like this before. So, it can confuse you a little bit, so couple of things to keep in mind is you can do any number of row inter changes. When you do this operation nothing will happen to the code, but if you are forced to do a column interchange what will happen yeah something will change in the code. The code will change, but from an error control code point of view from a communication system point of view.

It does not make any difference why does not it make any difference, the only change is the order in which the bits go out, so usually what happens is column change column swaps are also allowed when you do this Gaussian elimination. So, you can do column swaps as long as you remember that the sequence of bits going out as now changed. So, to reduce it to systematic form you can do cooperation's definitely, but you can do column swaps, but what it that you cannot do you is cannot add two columns.

So, that does not make much sense from a Gaussian elimination point of view adding two columns definitely not allowed you cannot do that. So, an exercise for you is to convert it into systematic form k you see k equals 4, then you try to list it out you will get sixteen different code words it will quite a few of them. So, you can try to write down if you like that this is a good exercise to figure out how to find systematic form for the code for the generator matrix. So, any number of examples you can now come up with its not too difficult this one is a particularly famous code.

So, it is a very famous code we will may be come back and see it later previous one we had nothing very special about it, but this one has may be previous one also has something special, but I do not know this one is quite special. So, it is it is called lots of

history, so I am going to pause for about four minutes, we will commence again at four ten, and you cannot wait.