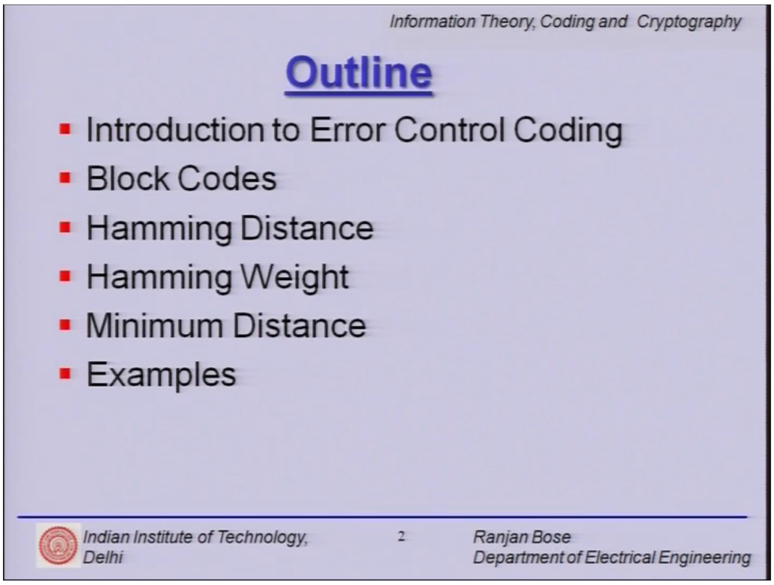
Information Theory, Coding and Cryptography
Dr. Ranjan Bose
Department of Electrical Engineering
Indian Institute of Technology, Delhi

Module – 13
Linear Block Codes
Lecture - 13

Hello and welcome to module 13, which is Linear Block Codes.

(Refer Slide Time: 00:34)



Information Theory, Coding and Cryptography

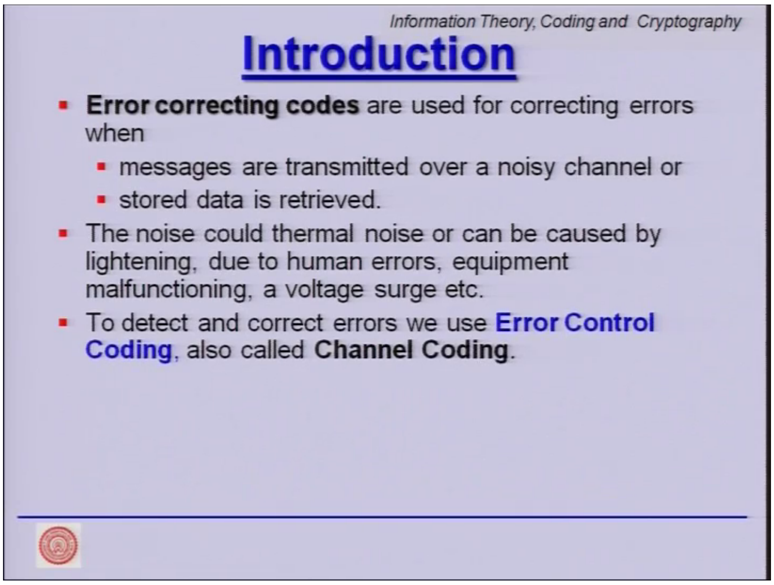
Outline

- Introduction to Error Control Coding
- Block Codes
- Hamming Distance
- Hamming Weight
- Minimum Distance
- Examples

Indian Institute of Technology, Delhi
2
Ranjan Bose
Department of Electrical Engineering

Let us look at a brief outline of today’s talk we will have an introduction to error control coding, which is a vast and exciting area within coding theory. Then we will look at block codes understand what do we mean by hamming distance? Then we go on to define hamming weight. Then we will talk about the minimum distance of codes and finally, we will look at some examples. So, that is the basic outline for today’s talk.

(Refer Slide Time: 01:08)



Information Theory, Coding and Cryptography

Introduction

- **Error correcting codes** are used for correcting errors when
  - messages are transmitted over a noisy channel or
  - stored data is retrieved.
- The noise could thermal noise or can be caused by lightening, due to human errors, equipment malfunctioning, a voltage surge etc.
- To detect and correct errors we use **Error Control Coding**, also called **Channel Coding**.

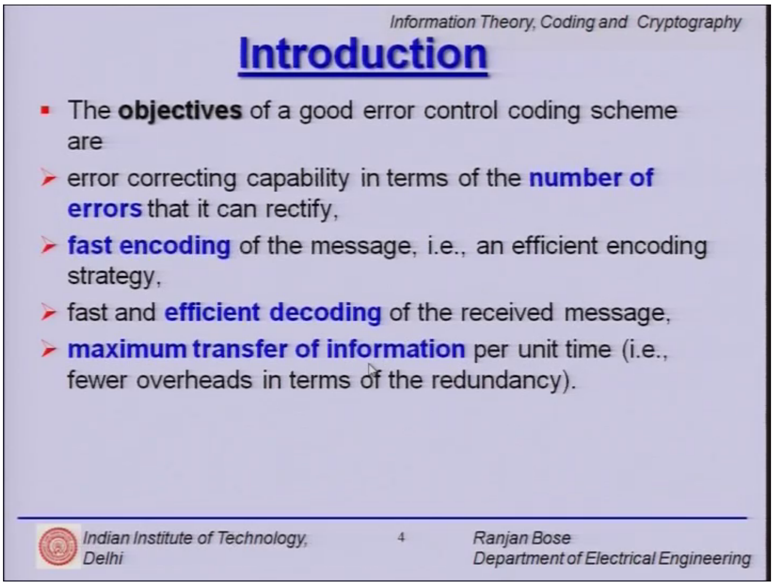
Let us start with a brief introduction our subject for today is error correcting codes. As the name suggests they are used for correcting errors when the messages are transmitted over noisy channels, which means that the noise in the channel are going to play havoc with the transmission, and some of the bits will eventually get flipped at the receiver.

On the other hand we also encounter errors when we store data and then retrieve them. For example, we stored data on a CD and the CD surface gets scratched; obviously, some errors would be introduced. Then also we might take help of error correcting codes to recover from errors. So, these are the primary 2 places where we will use error correcting codes.

Now, what do we mean by noise? Noise is present everywhere and then it could be a thermal noise; it could be caused by extraneous factors. It could be caused by human errors, equipment malfunction, voltage surge, lightening solar flares what have you. So, there could be many many causes of noise and which will eventually lead us to a certain number of errors.

Now, in order to detect and correct errors we will be using error control coding which is also called channel coding. So, let us understand these two things one is the detection problem, well we send bits we receive bits who knows which bits we have flipped. So, the first problem is the detection problem, and then I was smart enough to not only detect but also recover from the errors that are the problem of error correction. So, we will address both these issues in these lectures.

(Refer Slide Time: 03:09)



Information Theory, Coding and Cryptography

Introduction

- The **objectives** of a good error control coding scheme are
- error correcting capability in terms of the **number of errors** that it can rectify,
- **fast encoding** of the message, i.e., an efficient encoding strategy,
- fast and **efficient decoding** of the received message,
- **maximum transfer of information** per unit time (i.e., fewer overheads in terms of the redundancy).

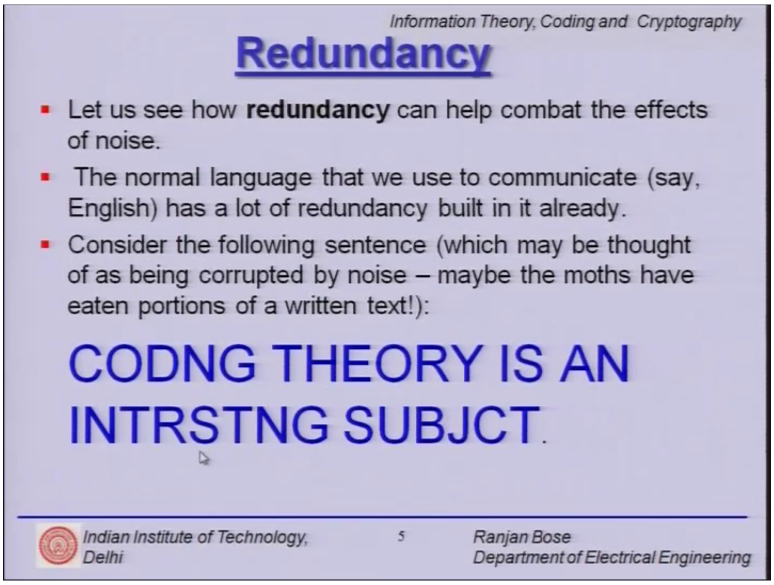
Indian Institute of Technology, Delhi 4 Ranjan Bose Department of Electrical Engineering

So, what are the objectives of good error correcting codes? Well the first and the foremost is it should be able to correct as many number of errors as possible ok. So, we talked about the error correcting capability of error correcting codes in terms of number of errors, and then we do not have all day to do things, it has to be fast in terms of the encoding process that is we are talking about an efficient encoding strategy.

And then just encoding and sending is not good enough at the receiver my friend must be able to decode it almost in real time and therefore, an efficient decoding is also required. And then it should be observed that we should be able to transfer maximum information per unit time, that it is not that we send a lot of coded message, but the actual information content is miniscule.

So, we must be able to send meaningful quantities of information when we do error correcting codes. So, all of these may not be satisfied at the same time, some of these constraints could be contradictory.

(Refer Slide Time: 04:35)



Information Theory, Coding and Cryptography

Redundancy

- Let us see how **redundancy** can help combat the effects of noise.
- The normal language that we use to communicate (say, English) has a lot of redundancy built in it already.
- Consider the following sentence (which may be thought of as being corrupted by noise – maybe the moths have eaten portions of a written text!):

CODNG THEORY IS AN INTRSTNG SUBJCT.

Indian Institute of Technology, Delhi

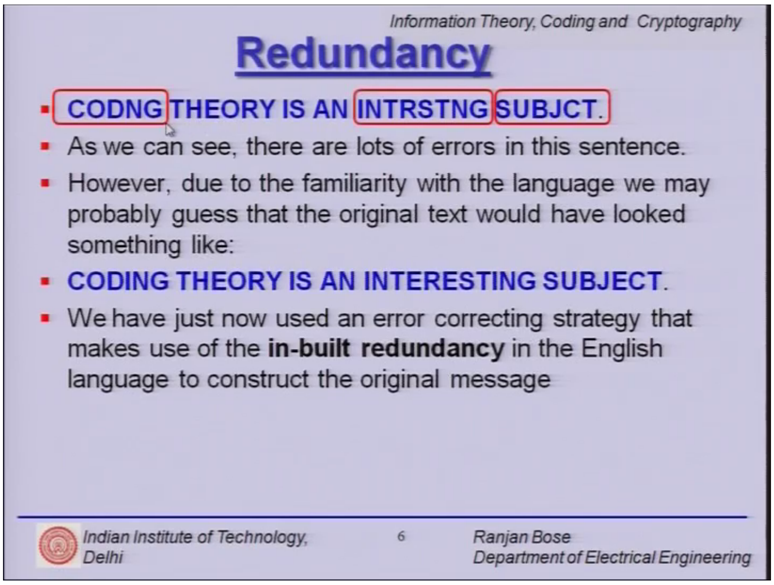
5

Ranjan Bose
Department of Electrical Engineering

Now, we start with one of the foremost requirements of an error correcting code. Let us see how redundancy can help combat the effects of noise or errors because noise would eventually lead to errors. So, let us begin with a simple example let us talk about the normal English language that we use this language that has evolved has a lot of redundancy built into it already.

We did encounter this redundancy of English language in our source coding section, where we saw that yes English could indeed be compressed to a great extent and therein lies the redundancy. So, consider the following sentence. So, I will not read it out, but you just have to observe and see what does it make any sense well it looks like English, it has the English alphabet and it has some words, but it appears that some of the words are in error ok.

(Refer Slide Time: 05:47)



Information Theory, Coding and Cryptography

Redundancy

- CODNG THEORY IS AN INTRSTNG SUBJCT.
- As we can see, there are lots of errors in this sentence.
- However, due to the familiarity with the language we may probably guess that the original text would have looked something like:
- CODING THEORY IS AN INTERESTING SUBJECT.
- We have just now used an error correcting strategy that makes use of the **in-built redundancy** in the English language to construct the original message

Indian Institute of Technology, Delhi 6 Ranjan Bose Department of Electrical Engineering

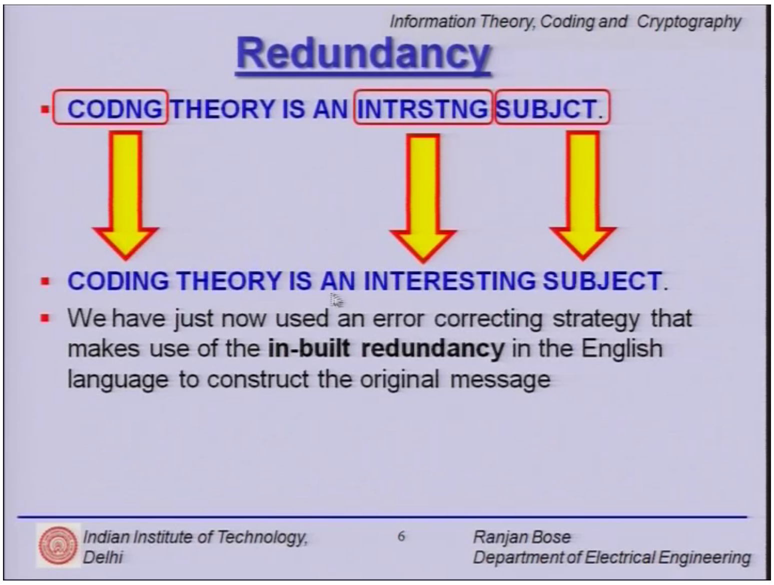
So, let us go and look at it a little bit more carefully it is the same sentence with some errors, and if we will try to make sense out of this seemingly meaningless, or at least erroneous sentence. So, due to our familiarity with a language we may probably start guessing. So, the first step that we did was detection. So, we detected there were errors.

Now the questions can be act smart and recover from these errors and we would probably try to guess the first word and it appears to be coding and second word is theory. So, it helped me combine these two words and I will get coding theory is an and this is again a difficult word, but I can guess it to be an interesting subject.

So, I come back with a recovered sentence like coding theory is an interesting subject here we use that everyday inbuilt redundancy of the English language to construct back the original message. It is a very simple example how redundancy helps us recover from errors.

Now, in our error control coding schemes we will use a more structured mathematically structured mechanism to recover from errors. But let us see what worked for this example. So, the 3 words that were really in error were the first word something to do with coding, then second word and the third word.

(Refer Slide Time: 07:40)



Information Theory, Coding and Cryptography

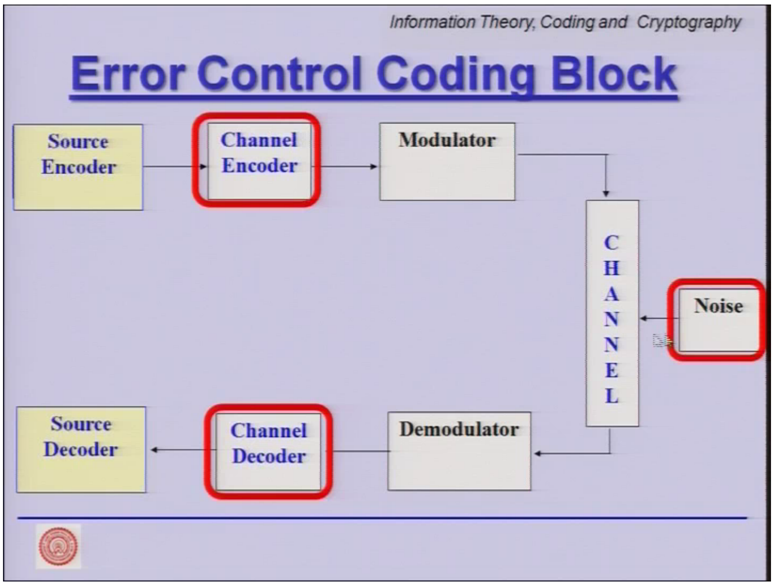
Redundancy

- CODNG THEORY IS AN INTRSTNG SUBJCT.
- CODING THEORY IS AN INTERESTING SUBJECT.
- We have just now used an error correcting strategy that makes use of the **in-built redundancy** in the English language to construct the original message

Indian Institute of Technology, Delhi 6 Ranjan Bose Department of Electrical Engineering

And when we looked at the first word either by itself or with relation to the next word we were able to get coding theory is an interesting subject. So, that is what we tried to do mentally. So, we had a strategy for decoding ok.

(Refer Slide Time: 07:57)



Information Theory, Coding and Cryptography

Error Control Coding Block

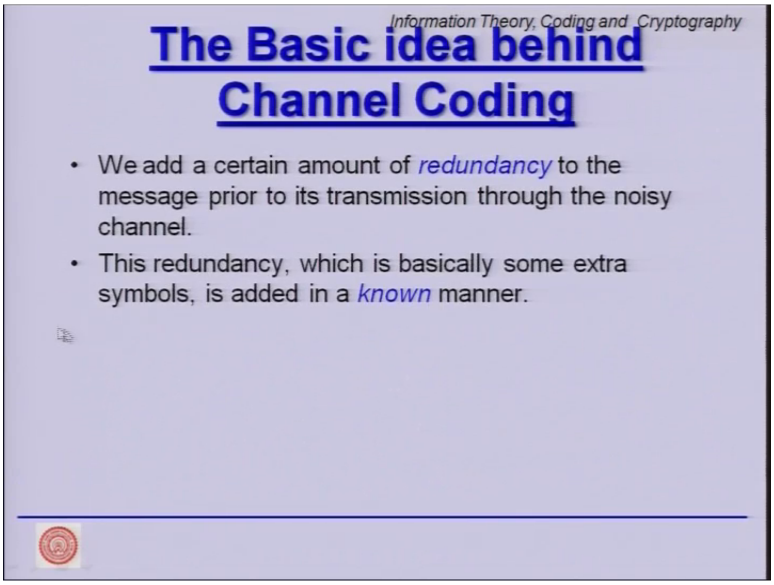
Now, let us look at the place of the error control coding block in the grand scheme of things. So, please remember we have already looked at source coding, where in we require the use of a source encoder. The job of the source encoder is to add redundancy oh I am sorry it is to recur reduce redundancy where the job of the channel encoder is to add redundancy.

So, first we remove redundancy and then these red circled blocks add redundancy in a known manner. So, the key is that we are not trying to undo the work of the source encoder which was removing redundancy ok. So, original language or information source has an inbuilt redundancy which is removed or reduced by the source encoder and then we add redundancy in a known manner using this channel encoder also called as the error control coding block. And then we must send the bit stream over the channel and to do, so we need a modulator.

So, once we do that the message is sent over the channel where we have our friend or enemy depending upon how you look at it the noise, the noise introduces the error. Now we demodulate and then go for the channel decoder which is a reverse of the channel encoder it figures out the errors and tries to recover from the errors. And then we pass on the error free or seemingly error free bit stream to the source decoder which is the opposite effect of the source encoder.

So, this is the entire scheme of things how we perform source encoding, channel encoding, modulation over the channel back at the receiver demodulation, channel decoding, source decoding and back to the happy user. So, that is the sequence of events. Please note the most important character in this play is this noise, without the noise we have no need for this channel encoder and channel decoder.

(Refer Slide Time: 10:21)



Information Theory, Coding and Cryptography

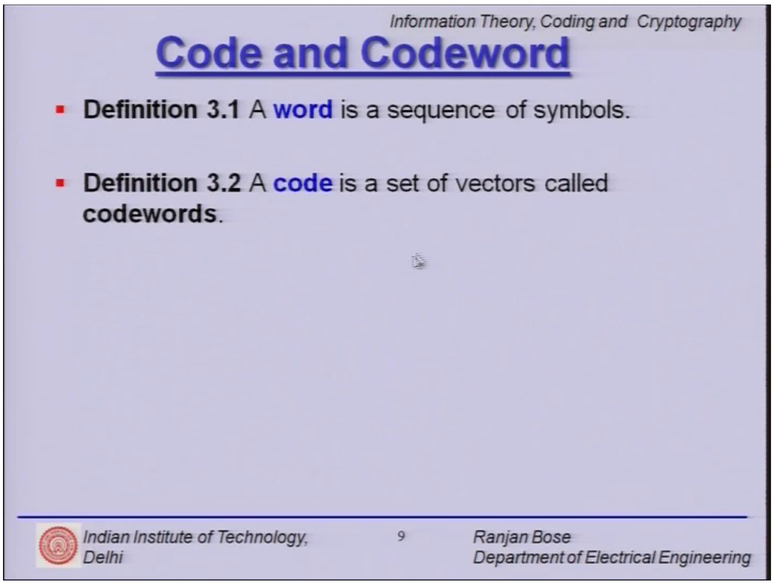
The Basic idea behind Channel Coding

- We add a certain amount of *redundancy* to the message prior to its transmission through the noisy channel.
- This redundancy, which is basically some extra symbols, is added in a *known* manner.

So, as we have observed just intuitively the basic idea behind channel coding is to add redundancy , but this redundancy is added in a known manner this is the critical word known manner known to whom well the transmitter and the friend receiver, the noise does not know the noise is random.

So, this is known we will probably put in an algebraic structure when we add redundancy, and that algebraic structure would be used at the decoder side to recover from errors ok. So, we add a certain amount of redundancy to the message prior to it is transmission though the noisy channel. This is the basic idea behind channel coding.

(Refer Slide Time: 11:12)



Information Theory, Coding and Cryptography

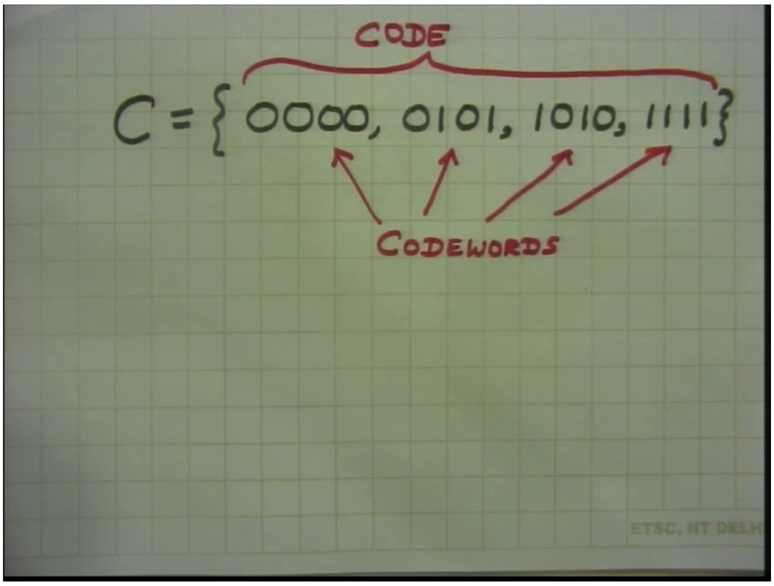
Code and Codeword

- **Definition 3.1** A **word** is a sequence of symbols.
- **Definition 3.2** A **code** is a set of vectors called **codewords**.

Indian Institute of Technology, Delhi — 9 — Ranjan Bose, Department of Electrical Engineering

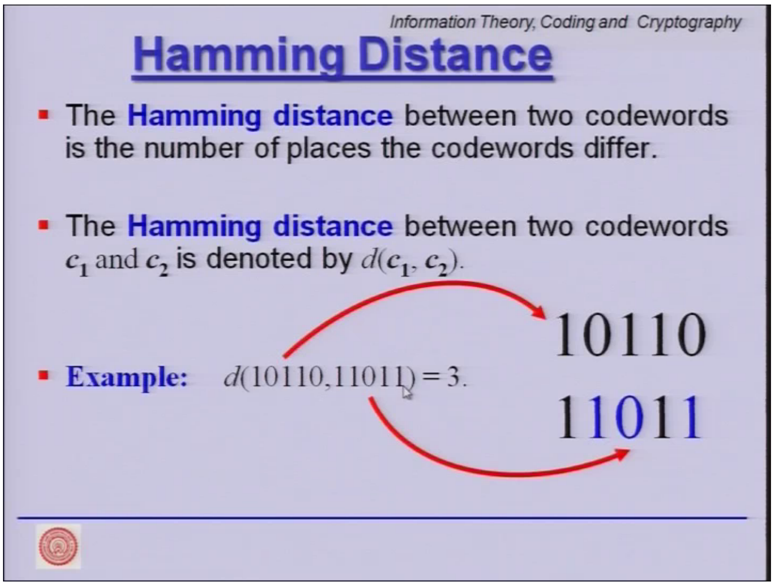
So, now let us look at the nuts and bolts of channel coding. How do we actually do it? So, we start with some definitions and let us start with a word. Word is a sequence of symbols we have come across this definition earlier, and a code is a set of vectors called code word ok. So, in general a code is a set of code words. So, if you look at some examples that we did in the previous classes let us refresh our memories.

(Refer Slide Time: 11:54)



So, I can always have a code which is a set. So, here I write out a set of code words. So, all of these 4 words are the code words and this whole set is the code. So, we go back to our slide and we start defining some properties of code words, so let us start with the first definition. So, you are looking at the slide now.

(Refer Slide Time: 12:51)



Information Theory, Coding and Cryptography

Hamming Distance

- The **Hamming distance** between two codewords is the number of places the codewords differ.
- The **Hamming distance** between two codewords c_1 and c_2 is denoted by $d(c_1, c_2)$.
- **Example:** $d(10110,11011) = 3$.

10110

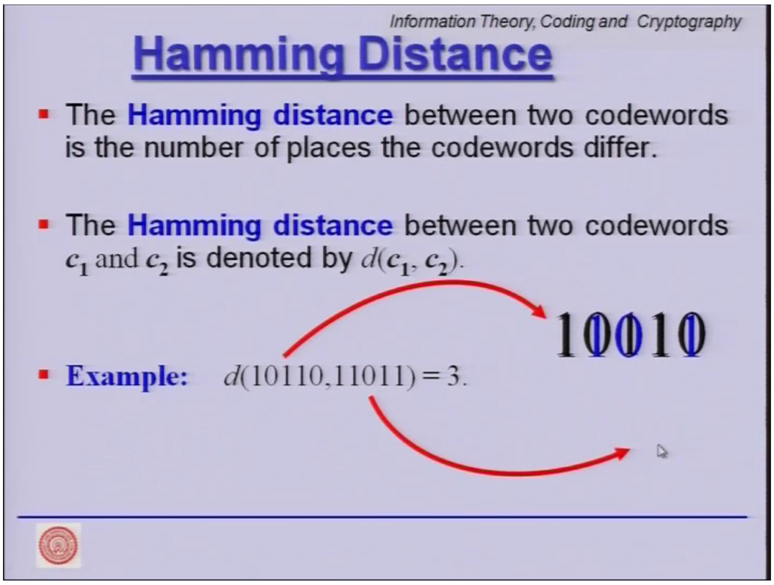
11011

We talked about the Hamming distance the hamming distance between two code words is a number of places the code words differ. So, please note it is a number of places hence it has no units, so this distance has no units. So, the hamming distance between two code words c 1 and c 2 can be denoted by d c 1 comma c 2, so that is the notation.

Let us look at an example so d that is the hamming distance between 1 0 1 1 0, and 11011 is 3 ok. How does it come? Well let us look at these two code words ok, so 10110 and 11011. We try to see whether the first location matches yes it does, the second location no there is a difference.

What are we trying to find the number of places the code words differ. So, I have marked the place number 2 with a blue because it differs and so does number 3 location number 3 it is a 0 here, it is a 1 here. So, again I put a blue and then 11 they are same. So, it is 1 and then again this last location is different. So, I just can superimpose and see that it differs in location 2, 3 and 5.

(Refer Slide Time: 14:33)



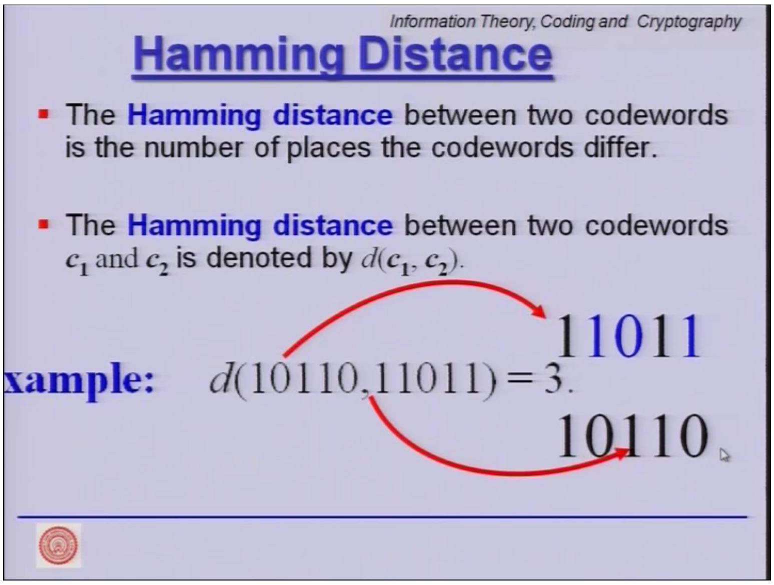
Information Theory, Coding and Cryptography

Hamming Distance

- The Hamming distance between two codewords is the number of places the codewords differ.
- The Hamming distance between two codewords c_1 and c_2 is denoted by $d(c_1, c_2)$.
- Example: $d(10110,11011) = 3.$

So, now I separate it out and I can say that the hamming distance is actually 3.

(Refer Slide Time: 14:42)



Information Theory, Coding and Cryptography

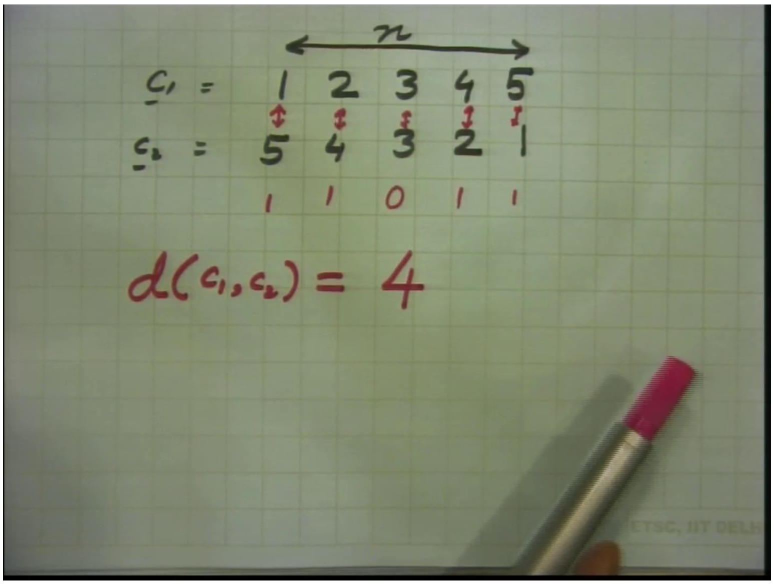
Hamming Distance

- The Hamming distance between two codewords is the number of places the codewords differ.
- The Hamming distance between two codewords c_1 and c_2 is denoted by $d(c_1, c_2)$.

xample: $d(10110,11011) = 3.$

So, I have been able to find out the number of places these two code words differ. Please note for these two have any meaning, the length of these two vectors must be the same. So, when we are talking about code words all the code words are of the same length called the block lengths. So, we have no problems, but hamming distance can be found between any 2 vectors of the same length.

(Refer Slide Time: 15:25)



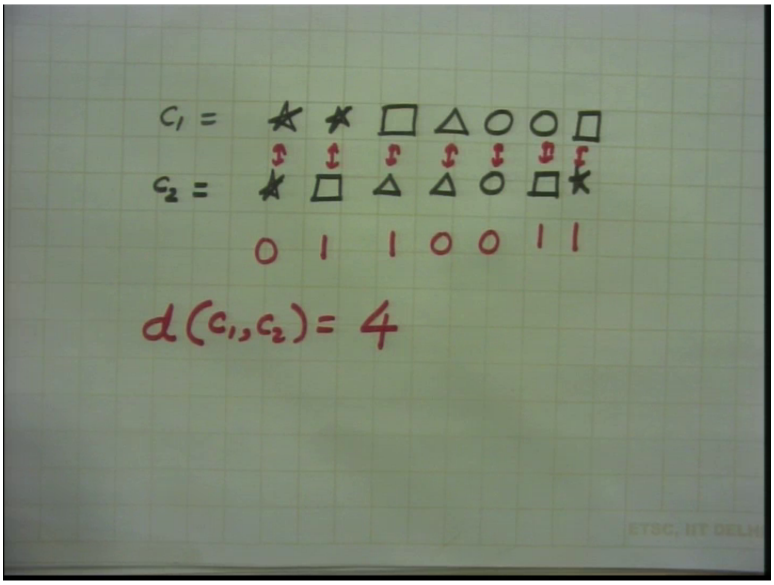
$$\begin{array}{lccccc} & \multicolumn{5}{c}{\xleftrightarrow{\quad n \quad}} \\ \underline{c}_1 = & 1 & 2 & 3 & 4 & 5 \\ & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow \\ \underline{c}_2 = & 5 & 4 & 3 & 2 & 1 \\ & 1 & 1 & 0 & 1 & 1 \end{array}$$

$$d(c_1, c_2) = 4$$

So, before we go to the next definition let us look at some other examples that will help us understand it. Suppose I have vector 1 as 1, 2, 3, 4, and 5. So, I have vector 1 c 1, vector 2 c 2, and then they are of equal length, I would like to find out the hamming distance between these two.

So, I compare the first one and I say oh this is different, so 1 this is different, 1 this is same so 0, this is different 1, this is different 1. So, the hamming distance d between c 1, and c 2 is equal to 4; the point that has being made is that it does not have to be a binary bit stream for you to compute the hamming distance.

(Refer Slide Time: 16:50)

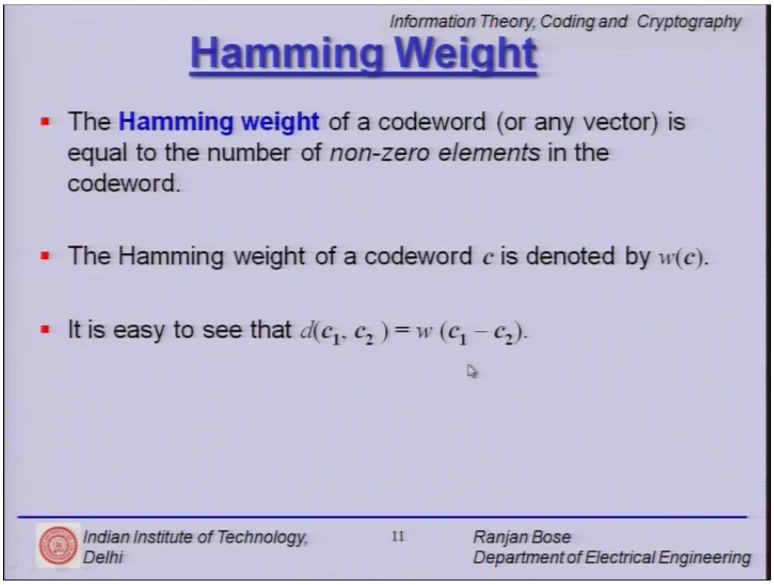


Let us look at another example let us say my c 1 is star, star, square, triangle, circle, circle, square. And let us say my c 2 is equal to star, square, triangle, triangle, circle, square, star, well. This is a vector well these are not numbers, but these are symbols this could be voltages, this could be light amplitudes, these could be star, square, circle, like we have drawn.

So, first we would like to count the vector lengths are the same or not 1, 2, 3, 4, 5, 6, 7, for both. So, we can find out the hamming distance between these two. So again we compare same, different, different, same, same, different, different.

So, if you compute the hamming distance between c 1 and c 2 you get again 4. So, these two vectors differ at 4 places. So, we have been able to find out the hamming distance between c 1 and c 2 in this case. So, the definition is pretty general.

(Refer Slide Time: 18:35)



Information Theory, Coding and Cryptography

Hamming Weight

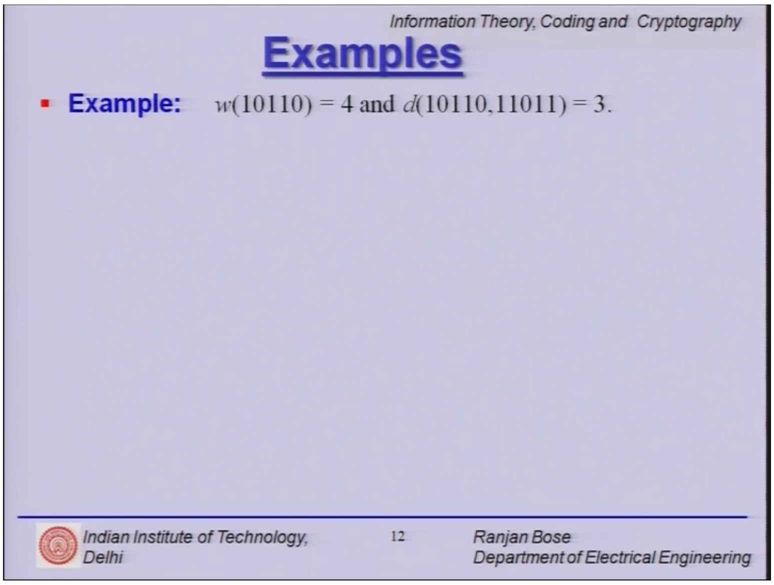
- The **Hamming weight** of a codeword (or any vector) is equal to the number of *non-zero elements* in the codeword.
- The Hamming weight of a codeword c is denoted by $w(c)$.
- It is easy to see that $d(c_1, c_2) = w(c_1 - c_2)$.

Indian Institute of Technology, Delhi 11 Ranjan Bose Department of Electrical Engineering

Now we go back to our slide and we look at the next definition which is that of hamming weight. The hamming weight of a code word or any vector for that matter is equal to the number again we are talking about a number here number of nonzero elements in that code word.

So, again the hamming weight has no units it is just a number of nonzero elements, hamming weight of a code word c is denoted by w of c. So, you can easily observe that the hamming distance between code word c 1 and c 2 is nothing, but the weight of c 1 minus c 2. In fact, the previous 2, 3 examples we are only computing the difference and then finding the weight.

(Refer Slide Time: 19:33)



Information Theory, Coding and Cryptography

Examples

- **Example:** $w(10110) = 4$ and $d(10110, 11011) = 3$.

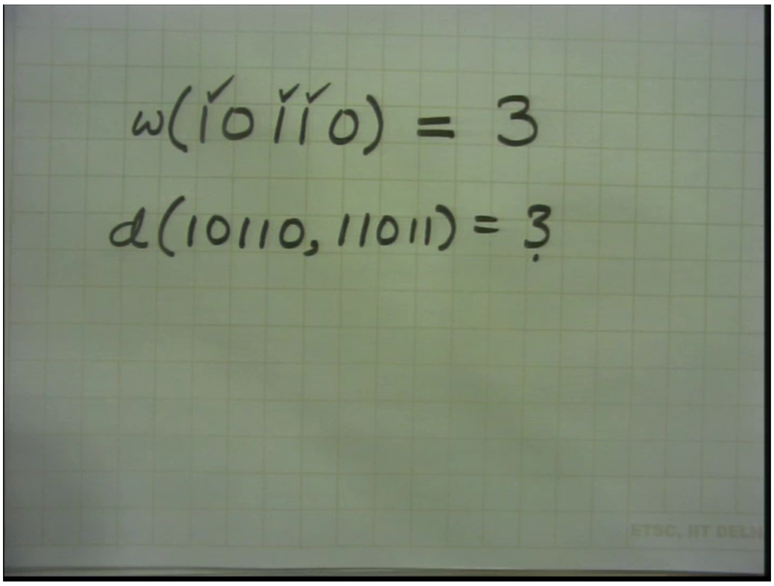
Indian Institute of Technology,
Delhi

12

Ranjan Bose
Department of Electrical Engineering

Let us look at some more examples. So, let us look at this vector 10110 well, this should be 3. Because the weight the number of nonzero elements is 3, so this should be 3. So, if you look at this example we can write it here.

(Refer Slide Time: 20:07)

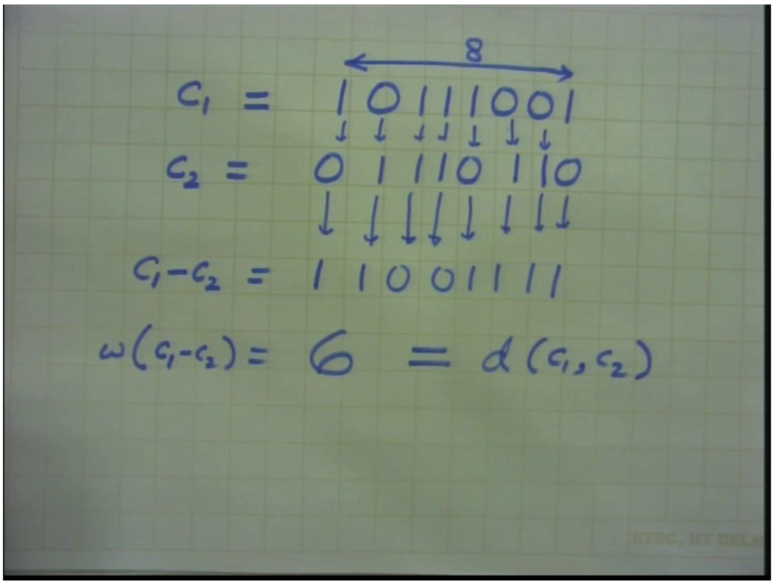


$$w(\overset{\checkmark}{1}0\overset{\checkmark}{1}\overset{\checkmark}{1}0) = 3$$

$$d(10110, 11011) = 3$$

So, we go back to our drawing board and look at w 10110, we count non zero, non zero, non zero, 3 right. And hamming distance we have already computed between any 2 vectors and find out the hamming distance. So, if you look at another example.

(Refer Slide Time: 21:00)



$$c_1 = \overset{\xleftrightarrow{\quad 8 \quad}}{10111001}$$

$$c_2 = 01110110$$

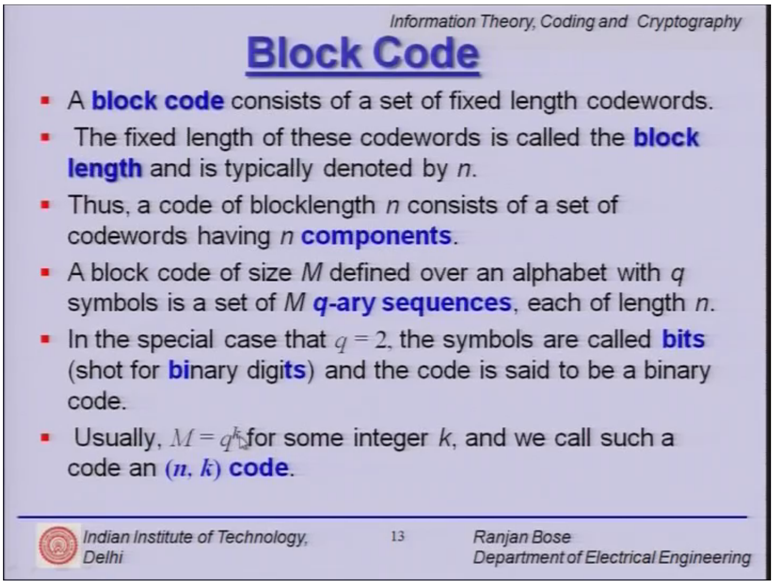
$$c_1 - c_2 = 11001111$$

$$\omega(c_1 - c_2) = 6 = d(c_1, c_2)$$

Let us say we have c 1 is equal to 10111001, and c 2 is equal to 0111011 we have to make sure 1, 2, 3, 4, 5, 6, 7, 8; 1, 2, 3, 4, 5, 6, 7, and 8. So, they are of the equal length if you find out c 1 minus c 2, it is simply a binary subtraction and here we get 1100. So, we are going right and then 111 and 1.

So, if you see that the difference actually tells you the number of places they are individually different each of this is an element of this vector. So, if you find out the weight of this vector c 1 minus c 2 is 1, 2, 3, 4, 5, 6, then nonzero elements should be the same as the distance between c 1 and c 2. So, these are what we mean by the hamming distance between c 1 and c 2 is nothing, but the weight of the difference of the 2 vectors.

(Refer Slide Time: 22:59)



Information Theory, Coding and Cryptography

Block Code

- A **block code** consists of a set of fixed length codewords.
- The fixed length of these codewords is called the **block length** and is typically denoted by *n*.
- Thus, a code of blocklength *n* consists of a set of codewords having *n* **components**.
- A block code of size *M* defined over an alphabet with *q* symbols is a set of *M* **q-ary sequences**, each of length *n*.
- In the special case that $q = 2$, the symbols are called **bits** (shot for **bi**nary digi**ts**) and the code is said to be a binary code.
- Usually, $M = q^k$ for some integer *k*, and we call such a code an **(n, k) code**.

Indian Institute of Technology, Delhi
13
Ranjan Bose
Department of Electrical Engineering

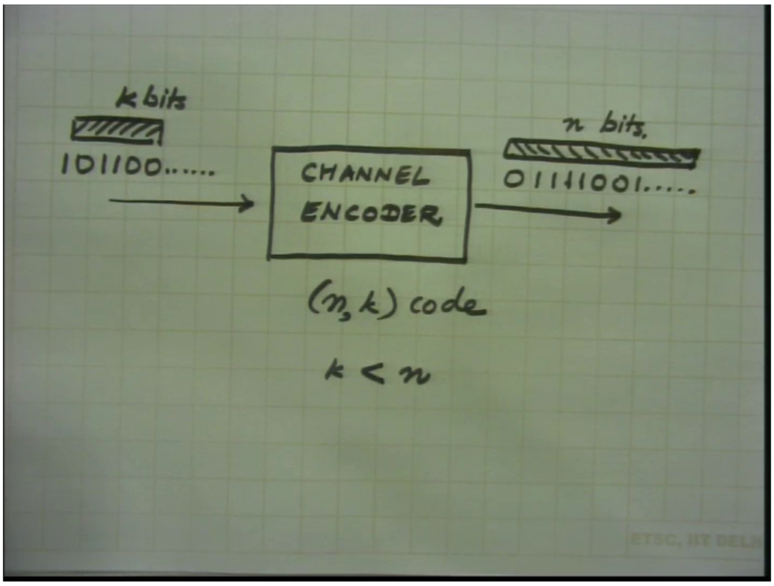
Now, let us look at something called as a block code. Because now we are entering the domain of error control coding and we would need these mathematical tools. A block code consists of a set of fixed length code words. So, clearly we already know that code is a set of code words now we are introduced this new adjective block.

So, block code consists of a set of fixed length code words and what is this length called this length is called the block length and it is typically denoted by n that is a code of block length n consists of a set of code words each having n components. Basically it means each code word is n symbols long, if it is binary then it is n bits long, but nothing forces us to make it only binary.

Now, block code of size M defined over an alphabet with q symbols. So, here we are talking about non binary block code is a set of n q-ary sequences each of length n alright. So, if it is ternary then your code words will consists of 0, 1 and 2; if it is quaternary it will consist of 0, 1, 2 and 3 ok. It could be hexadecimal it could be decimal. So, you can have q ary sequences representing the code words. Binary case q is equal to 2, and therefore it is called bits. Bits is like bi for binary, and ts for digits and therefore, this together forms the bits and therefore, it is called the binary code.

So, in general we can have a q ary sequence, but one of the most popular forms is the binary code and most of the time we will look at binary codes. But please remember they are not the only kind of codes possible. So, M is equal to q raise to power k, what is this k is some integer and we will explain the physical significance shortly, and this is called an n comma k code.

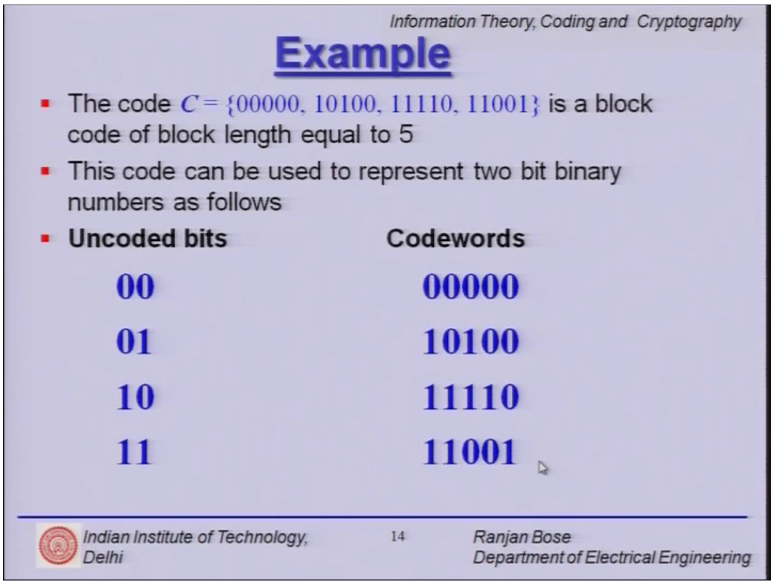
(Refer Slide Time: 25:43)



So, let us go back to our drawing board and where we have our error control correcting block. So, we called as a channel encoder, the job of this channel encoder is to add redundancy. So, input is of course, a bit stream an output is also a bit stream, but it has additional bits involved.

Suppose this channel encoder starts with q bits, and after adding redundancy it makes it n bits. Then it is using an n comma k code to do. So, so an n comma k channel code takes in k bits, and converts it into n bits. So, typically k is less than n.

(Refer Slide Time: 27:03)



Information Theory, Coding and Cryptography

Example

- The code $C = \{00000, 10100, 11110, 11001\}$ is a block code of block length equal to 5
- This code can be used to represent two bit binary numbers as follows
- **Uncoded bits** **Codewords**

Uncoded bits	Codewords
00	00000
01	10100
10	11110
11	11001

Indian Institute of Technology, Delhi 14 Ranjan Bose Department of Electrical Engineering

Let us look at an example let us talk about this code c which is a set of code words, it is a block code. Now what is the block length, so we count 1, 2, 3, 4, and 5, each one has a length 5, so the block length n equal to 5 for this code.

Now, what is the practical significance of this? So, what if it is a set of code words what can we do with it? Well this code can be used to represent 2 bit binary numbers as follows.

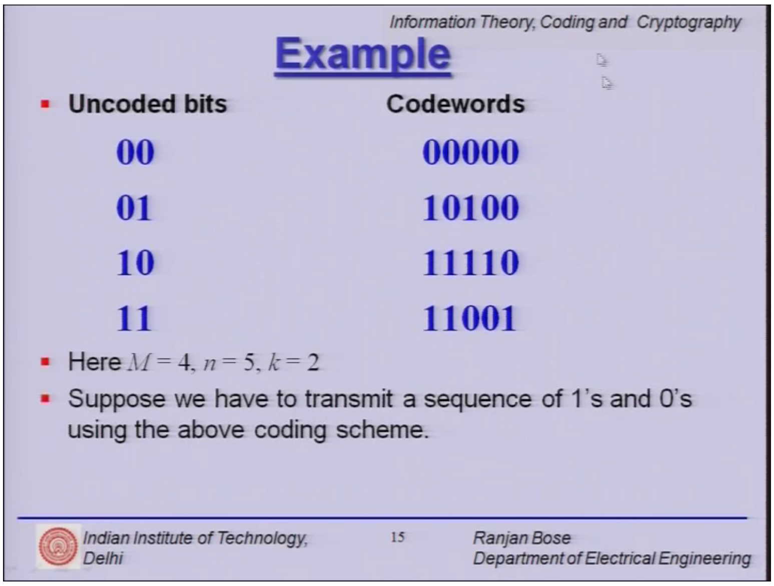
So, this error control code takes in 2 bits 2 uncoded bits, and maps them to the code words as follows. So, 00 is rewritten as 00000, so 2 bits become 5 bits, sure we have added redundancy 01 could be 10100 who told me to do this well this is my particular code. Similarly, 10 becomes 1110 and 11 becomes 11001.

Now let us make certain observations then the set the size is 4. There are 4 possible code words and hence log to the base 2 4 equals 2 and so it can only take 2 bits at a time, and you can have 4 unique code words. The first thing is they should be unique; the aim of the receiver is to go back from the received code words and try to decode and declare the original bits.

So, ultimately it is a lookup table input output and this is the mapping done by the error control coding block. So, one thing we can clearly observe is there are several possible sets which are of length 5. So, for n is equal to 5 we can have actually 2 raised to the power 5; 32 possible choices or which we have picked 4 this should be noted, even though the possible inputs are only four 00011011.

The mere fact that the length of the code word is 5 we have a possible set of 32 vectors of with respect to 5 4. Now, which 4 are the best is the subject of further study. But here we actually represent an n comma k where n is equal to 5 and k is equal to 2. So, here k is equal to 2, and n is equal to 5, I take 2 bits make it into 5, 2 bits make it into 5.

(Refer Slide Time: 30:25)



Information Theory, Coding and Cryptography

Example

Uncoded bits	Codewords
00	00000
01	10100
10	11110
11	11001

- Here $M = 4$, $n = 5$, $k = 2$
- Suppose we have to transmit a sequence of 1's and 0's using the above coding scheme.

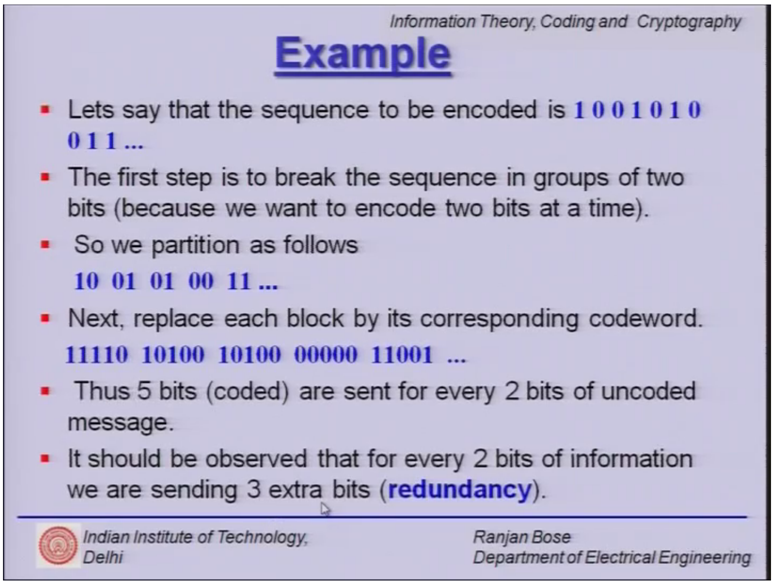
Indian Institute of Technology, Delhi

15

Ranjan Bose
Department of Electrical Engineering

So, this if this is my lookup table I will say M is equal to 4, there is the size n is equal to 5 and k is equal to 2. Now, that we have the lookup table how do we implement it in real life. Suppose we have to transmit a long sequence of 1's and 0's using this coding scheme. So, we already have our first toy example we have a basic block code linear block code with n is equal to 5 and k is equal to 2. And we want to use this simple example to encode a long bit stream how do we do that?

(Refer Slide Time: 31:08)



Information Theory, Coding and Cryptography

Example

- Lets say that the sequence to be encoded is 1 0 0 1 0 1 0 0 1 1 ...
- The first step is to break the sequence in groups of two bits (because we want to encode two bits at a time).
- So we partition as follows
  10 01 01 00 11 ...
- Next, replace each block by its corresponding codeword.
  11110 10100 10100 00000 11001 ...
- Thus 5 bits (coded) are sent for every 2 bits of uncoded message.
- It should be observed that for every 2 bits of information we are sending 3 extra bits (**redundancy**).

Indian Institute of Technology, Delhi

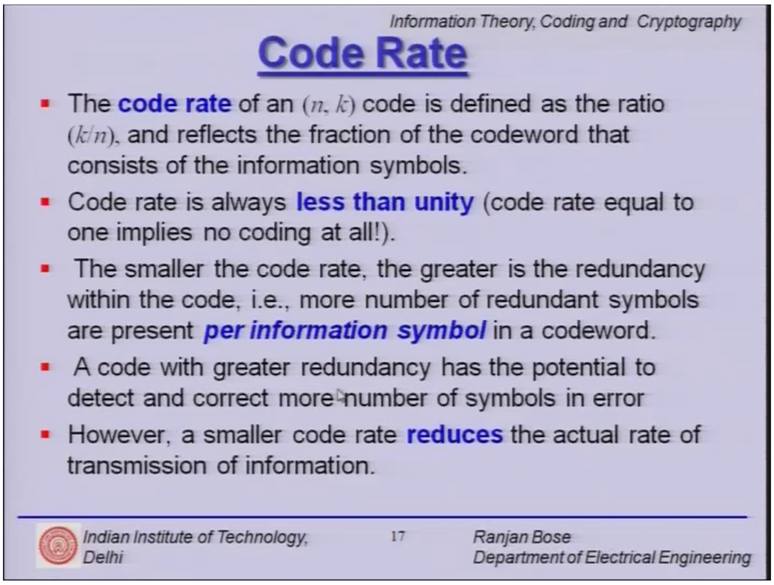
Ranjan Bose
Department of Electrical Engineering

Suppose the sequence to be encoded is some random 1001010 dot dot dot dot, so we can we have enough of those bits coming in. The first step is to break the sequence into groups of two bits why two bits? Because the encoder takes k bits at a time and k is equal to 2 it cannot handle more than 2 bits at a time.

So, the long bit sequence that we have we partition it as 10 01 01 00 11 these are the 10 01 01 and so on and so forth. And then each one is mapped as per the lookup table, so 10 will be replaced by it is 11110; 01 will be replaced as 10100 and so and so forth till you get the entire thing.

So, for the first 1, 2, 3, 4, 5, 10 bits, the first 10 bits we will have 25 bits of data being sent out. So, it is a lot of redundancy that has been added here. So, for every 5 coded bits that we want to send out 2 input bits are taken of the uncoded met message. So, we have added 3 extra bits for every 2 information bits. So, we have information bit and redundant bits, 2 information bits and 3 redundant bits constitute one code word that is the general plan in this example.

(Refer Slide Time: 32:54)



Information Theory, Coding and Cryptography

Code Rate

- The **code rate** of an (n, k) code is defined as the ratio (k/n), and reflects the fraction of the codeword that consists of the information symbols.
- Code rate is always **less than unity** (code rate equal to one implies no coding at all!).
- The smaller the code rate, the greater is the redundancy within the code, i.e., more number of redundant symbols are present ***per information symbol*** in a codeword.
- A code with greater redundancy has the potential to detect and correct more number of symbols in error
- However, a smaller code rate **reduces** the actual rate of transmission of information.

Indian Institute of Technology, Delhi 17 Ranjan Bose Department of Electrical Engineering

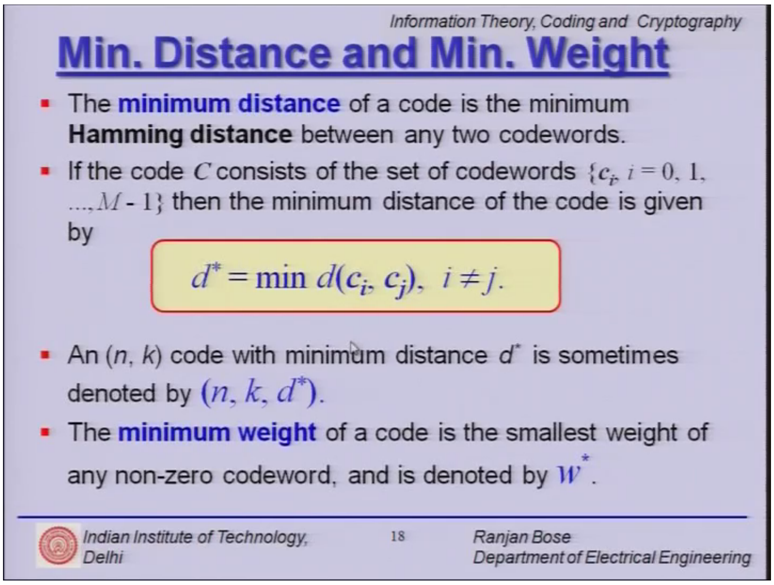
So, code rate of an n comma k code is defined as a ratio k over n clearly k is less than n. So, code rate must necessarily be less than 1 ok. If we have code rate equal to unity that is n is equal to k we have added no redundancy and obviously, we have not done any coding at all, but that will be a useless trivial case.

The smaller is the code rate the greater amount of redundancy within the code that is the more number of redundant symbols are present per information symbol. So, this is not a good news, because if you had too much redundancy you are sending the overhead bits rather than the information symbols, so we have to be careful about it.

A code with greater redundancy has the potential to detect and correct more errors. Because you pay the price and you get the return but the challenges that you should have as small additional bits added. So, the code rate should be as close to 1 as possible. So, code rate in some sense you know tells us about the efficiency of the code.

A smaller code rate reduces the actual rate of transmission of information this is one of our original objectives of a good error correcting code that it should be efficient not only in terms of encoding procedure, or decoding procedure, but also efficient in terms of the overheads that we use to transfer the data.

(Refer Slide Time: 34:37)



Information Theory, Coding and Cryptography

Min. Distance and Min. Weight

- The **minimum distance** of a code is the minimum **Hamming distance** between any two codewords.
- If the code C consists of the set of codewords $\{c_i, i = 0, 1, \ldots, M - 1\}$ then the minimum distance of the code is given by

$$\boxed{d^* = \min d(c_i, c_j),\ i \neq j.}$$

- An (n, k) code with minimum distance d^* is sometimes denoted by (n, k, d^*).
- The **minimum weight** of a code is the smallest weight of any non-zero codeword, and is denoted by w^*.

Indian Institute of Technology, Delhi 18 Ranjan Bose Department of Electrical Engineering

Now, let us look at certain properties of the linear block codes, and these will be used to characterize how good or bad these codes are. So, we talked about 2 very important properties the minimum distance of a code and the minimum weight of a code. So, let us define them.

So, the minimum distance of a code is a minimum hamming distance between any 2 code words in that set. And if the set consists of M code words then the minimum distance of the code is defined as d star, is minimum over all possible distances.

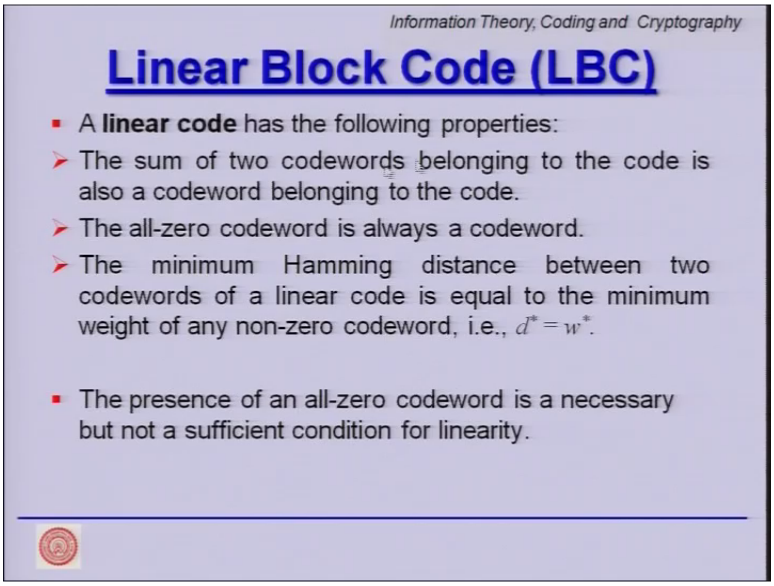
So, d distance between ci and cj where i is not equal to j clearly when c i equal to cj that they are identical code words the distance will be 0, the distance will be 0. So, if they are non-identical, then you will be conclude all the distances and whatever is the smallest that minimum distance is called the minimum distance of the code.

So, it is one of the properties of the code. So, sometimes an n comma k code is denoted by n comma k comma d star, so it is a property. So, it takes k bits as inputs transforms it is into n bits and provides a minimum distance of d star.

Now what is a minimum weight the minimum weight of a code is a smallest weight of any nonzero code word and it is denoted by w star. So, if there are M code words in this code remember code is a set of code words then we find out all the weights hamming weights of the code words other than the nonzero code word.

And the minimum of those weights is called the w star we will show that d star is equal to double star for linear block codes shortly , but what are linear block codes. So, this is again a very interesting and a very important area and let us focus on it.

(Refer Slide Time: 37:15)



Information Theory, Coding and Cryptography

Linear Block Code (LBC)

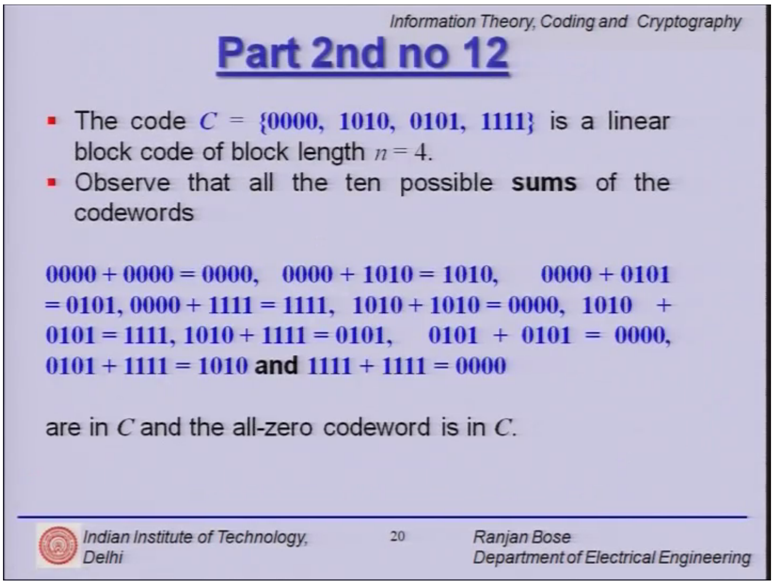
- A **linear code** has the following properties:
  - The sum of two codewords belonging to the code is also a codeword belonging to the code.
  - The all-zero codeword is always a codeword.
  - The minimum Hamming distance between two codewords of a linear code is equal to the minimum weight of any non-zero codeword, i.e., $d^* = w^*$.
- The presence of an all-zero codeword is a necessary but not a sufficient condition for linearity.

So, the linear block code has a has the following properties. Property number 1 the sum of any 2 code words belonging to the code is also a valid code word belonging to the code. So, this is very interesting it is inclusive. So, take any 2 code words at random from the set add them up. What do you mean by adding them up well the addition operation needs to be defined for binary let us say it is just XOR operation which is the addition operation. So, sum of any 2 code words is also a valid code word.

Property number 2 of a linear block code the all-zero code word must be a valid code word. And finally, the minimum hamming distance between 2 code words of a linear code. So, the minimum distance of the code word is equal to the minimum weight of the code. So, d star is w star for a linear block code. So, these are the three properties of a linear code.

Please note that the presence of an all zero code word is a necessary, but not a sufficient condition for linearity ok. So, if you are required to verify whether a given set of vectors forms a linear block code you need to check these three properties.

(Refer Slide Time: 38:53)



Information Theory, Coding and Cryptography

Part 2nd no 12

- The code C = {0000, 1010, 0101, 1111} is a linear block code of block length $n = 4$.
- Observe that all the ten possible **sums** of the codewords

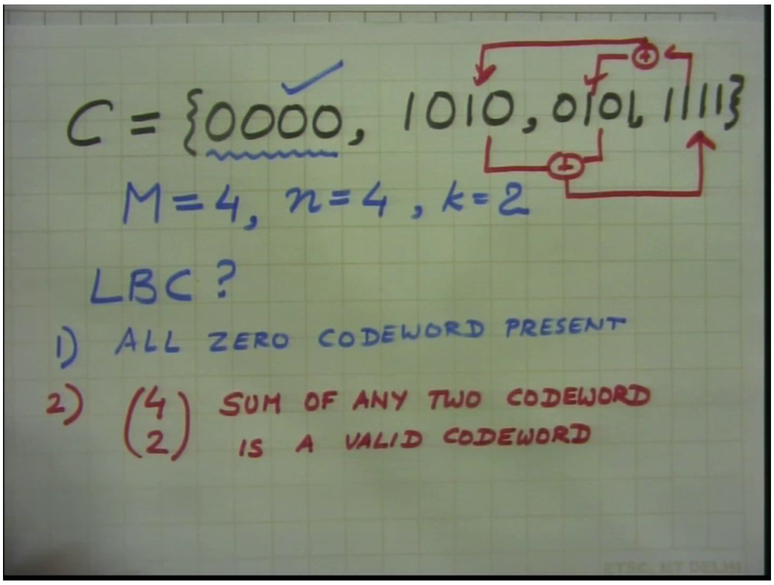
0000 + 0000 = 0000, 0000 + 1010 = 1010, 0000 + 0101 = 0101, 0000 + 1111 = 1111, 1010 + 1010 = 0000, 1010 + 0101 = 1111, 1010 + 1111 = 0101, 0101 + 0101 = 0000, 0101 + 1111 = 1010 and 1111 + 1111 = 0000

are in C and the all-zero codeword is in C.

Indian Institute of Technology, Delhi 20 Ranjan Bose Department of Electrical Engineering

So, let us look at this example. So, let us go back to the drawing board.

(Refer Slide Time: 39:15)



$C = \{0000, 1010, 0101, 1111\}$

$M=4, n=4, k=2$

LBC ?

1) ALL ZERO CODEWORD PRESENT

2) $\binom{4}{2}$ SUM OF ANY TWO CODEWORD IS A VALID CODEWORD

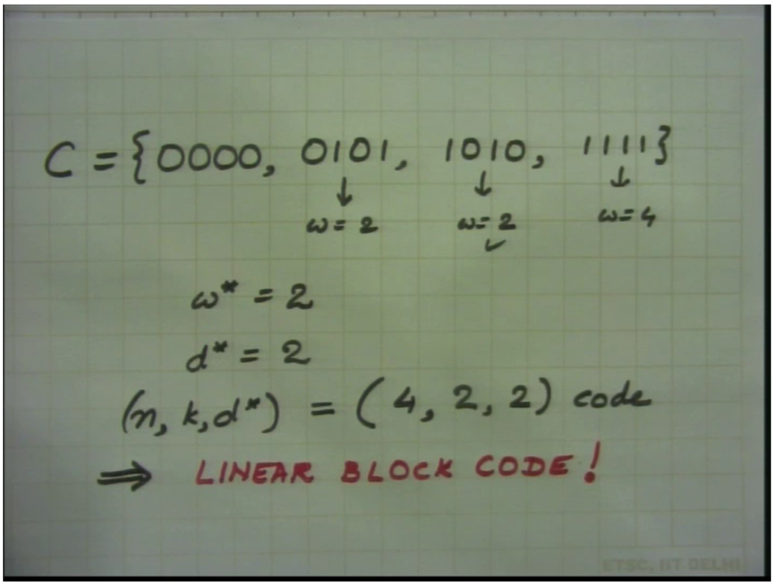
And look at this code C equal to. So, if you look at this code M is equal to 4, n is equal to 4, and k is equal to 2. Now we would like to verify whether it is a linear block code or not that is our question? So, first let us see whether the all zero code word forms one of the valid code words it is true, so property number 1 all zero code word present ok.

So, now, we look at the second property which says that sum of any 2 code words is a valid code word. Now, how many sums are possible well this plus this plus this. So, basically you have 4 choose 2 right so we have to now check all of them and find out whether each one each of the sums is actually a valid code word.

So, if you look at all zero code word if you add it to any one of them they will remain unaltered. So, they automatically belong to the set, but let us say this and this. So, if you had this plus this you get this, so this, plus this, you get this. On the other hand if I add this plus this I get this, so I can add any one and I can verify that it falls back into one of these 4 vectors.

So, we have sum of any 2 code words is valid code word and then we can carry out the final check. So we looked at our code again 0000, 0101, 1010, 1111 and if you look at the minimum weight of the code, which is the weight of the nonzero elements.

(Refer Slide Time: 42:36)



$C = \{0000, \underset{\overset{\downarrow}{w=2}}{0101}, \underset{\overset{\downarrow}{w=2}}{1010}, \underset{\overset{\downarrow}{w=4}}{1111}\}$

$w^* = 2$

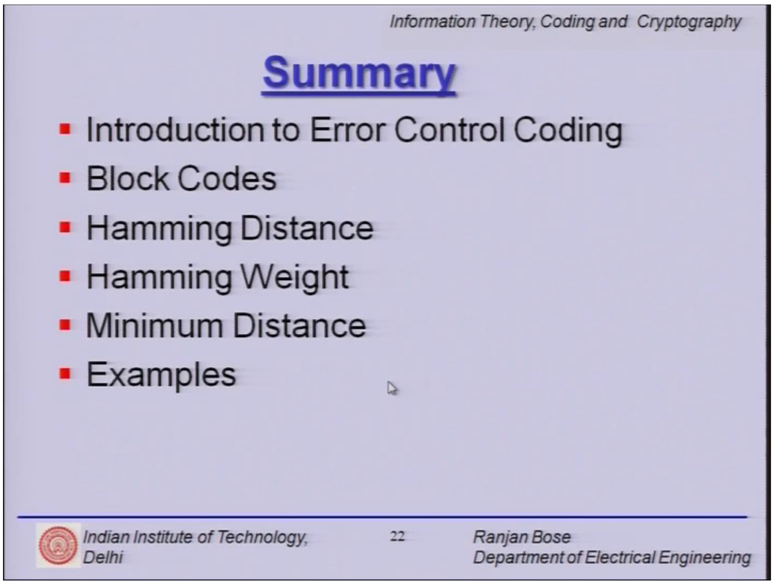
$d^* = 2$

$(n, k, d^*) = (4, 2, 2)$ code

$\Rightarrow$ LINEAR BLOCK CODE!

So, this has weight equal to 2; weight equal to 2; weight equal to 4. So, minimum is 2 so w star is equal to 2 and if you find out the differences and you will get d star also equal to 2. So, this code n k d star can be written as 4, 2, 2 code. And we have verified that it is indeed a linear block code.

(Refer Slide Time: 44:06)



Information Theory, Coding and Cryptography

Summary

- Introduction to Error Control Coding
- Block Codes
- Hamming Distance
- Hamming Weight
- Minimum Distance
- Examples

Indian Institute of Technology, Delhi 22 Ranjan Bose Department of Electrical Engineering

So, now, we come to the summary of our talk. We started with a brief introduction to error control coding why is it important? What are the good properties expected from error control codes. We then looked at what are block codes. We defined what do we mean by hamming distance and hamming weight? And then we looked at how they can be used to determine the minimum distance of the code.

In subsequent lectures, we will talk about how important is this minimum distance of a code and it actually tells us how good the code is, what is the error correcting capability of the code all of it will get linked to the minimum distance. So, this quantity will become one of the single most important parameters to judge how good the code is the other one will be of course, the code rate. And then we looked at several examples to better understand these concepts. So, with this we come to the end of this lecture.