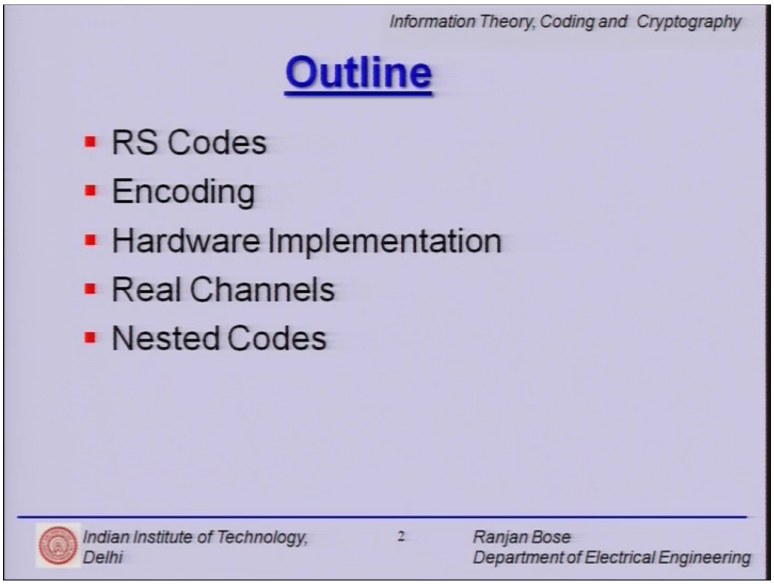
Information Theory, Coding and Cryptography
Dr. Ranjan Bose
Department of Electrical Engineering
Indian Institute of Technology, Delhi

Module - 25
Reed Solomon Codes
Lecture - 25

Hello, and welcome to our next a module on Reed Solomon Codes. Let us start with a brief outline of today's talk. We would understand what we mean by Reed Solomon Codes. Then, we will look at the encoding for Reed Solomon Codes.

(Refer Slide Time: 00:41)



Information Theory, Coding and Cryptography

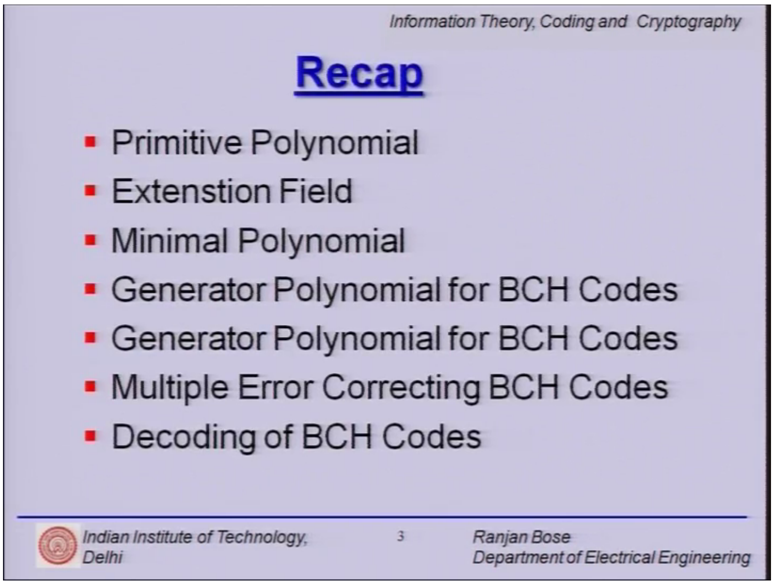
Outline

- RS Codes
- Encoding
- Hardware Implementation
- Real Channels
- Nested Codes

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

We will realise that Reed Solomon Codes are very very amenable to hardware implementation. We will look at how to do it efficiently using hardware, when we will look at how they perform over real channels. And finally we will spend some time on nested codes.

(Refer Slide Time: 01:04)



Information Theory, Coding and Cryptography

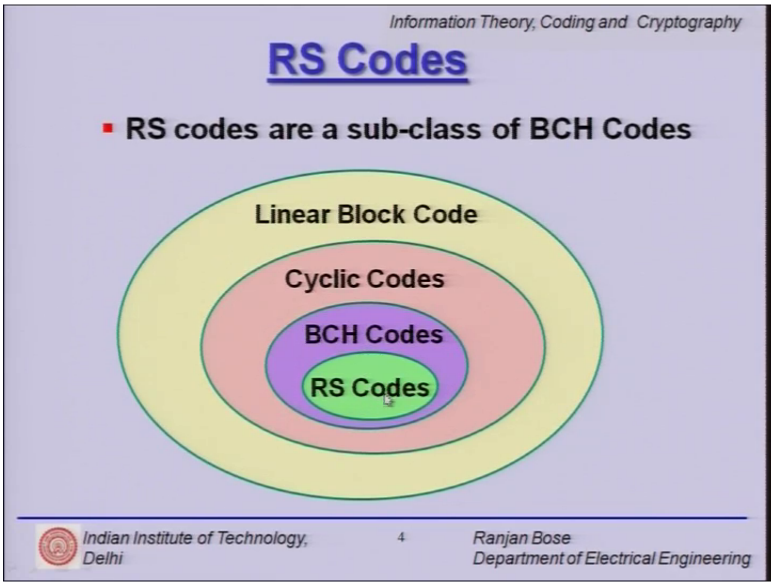
Recap

- Primitive Polynomial
- Extenstion Field
- Minimal Polynomial
- Generator Polynomial for BCH Codes
- Generator Polynomial for BCH Codes
- Multiple Error Correcting BCH Codes
- Decoding of BCH Codes

Indian Institute of Technology, Delhi 3 Ranjan Bose Department of Electrical Engineering

So, let us see what we have done so far. We have been looking at the general class of BCH codes where we have started with the primitive polynomial, looked at the extension fields, the notion of minimal polynomials, generating generator polynomials, and then finally the BCH codes.

(Refer Slide Time: 01:20)



Information Theory, Coding and Cryptography

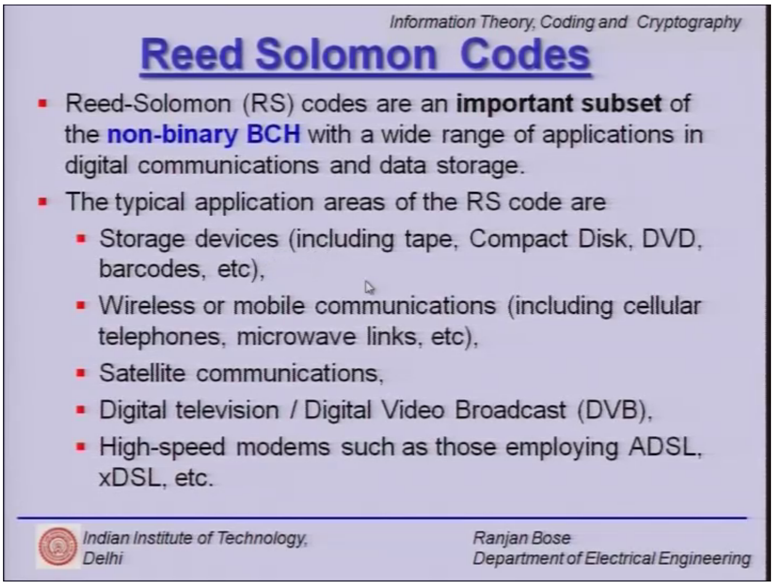
RS Codes

- RS codes are a sub-class of BCH Codes

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

Now, if you look at where we are in terms of Reed Solomon Codes, we clearly understand that Reed Solomon codes are a subclass of BCH codes. We pointed out in the earlier lecture that the umbrella code is the linear block code, cyclic codes form a subclass of linear block codes, BCH are special subclass of cyclic codes, and finally, we are going to study Reed Solomon Codes.

(Refer Slide Time: 01:56)



Information Theory, Coding and Cryptography

Reed Solomon Codes

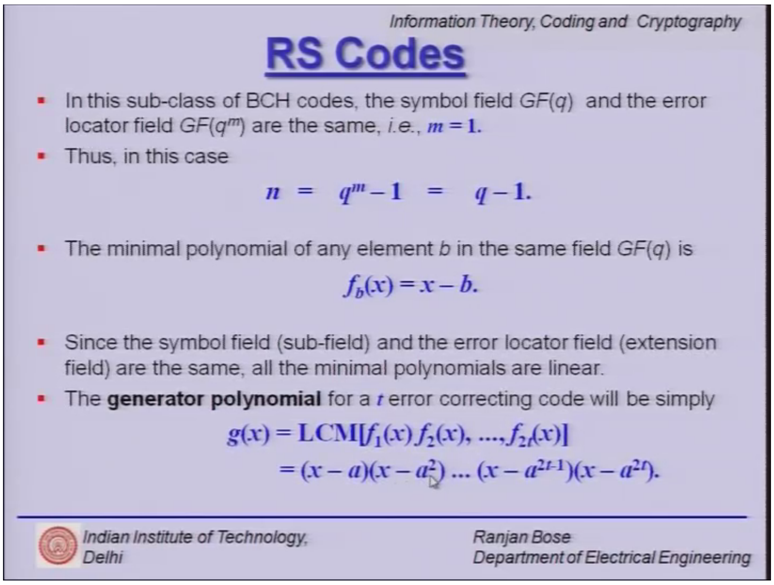
- Reed-Solomon (RS) codes are an **important subset** of the **non-binary BCH** with a wide range of applications in digital communications and data storage.
- The typical application areas of the RS code are
  - Storage devices (including tape, Compact Disk, DVD, barcodes, etc),
  - Wireless or mobile communications (including cellular telephones, microwave links, etc),
  - Satellite communications,
  - Digital television / Digital Video Broadcast (DVB),
  - High-speed modems such as those employing ADSL, xDSL, etc.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, they are an important subset of non-binary BCH codes. We will realise shortly that there is no point in forming binary Reed Solomon Codes. They will have no meaning. We will look at effective non-binary BCH codes and the applications are at many places including lot of storage devices using CDs, DVDs, barcode, they all use Reed Solomon Codes because of it is burst error correcting capability. Wireless channels and mobile communications also employee Reed Solomon Codes. So do satellite communication and deep space communications including we have this digital TV and digital video broadcasting, DVB standards and ADSL, xDSL also use some form of Reed Solomon Codes. So, they are present everywhere. There is very strong class of codes.

(Refer Slide Time: 02:52)



Information Theory, Coding and Cryptography

RS Codes

- In this sub-class of BCH codes, the symbol field $GF(q)$ and the error locator field $GF(q^m)$ are the same, *i.e.*, $m = 1$.
- Thus, in this case

$$n = q^m - 1 = q - 1.$$

- The minimal polynomial of any element b in the same field $GF(q)$ is

$$f_b(x) = x - b.$$

- Since the symbol field (sub-field) and the error locator field (extension field) are the same, all the minimal polynomials are linear.
- The **generator polynomial** for a t error correcting code will be simply

$$g(x) = \mathrm{LCM}[f_1(x) f_2(x), \ldots, f_{2t}(x)]$$
$$= (x - \alpha)(x - \alpha^2) \ldots (x - \alpha^{2t-1})(x - \alpha^{2t}).$$

Indian Institute of Technology, Delhi

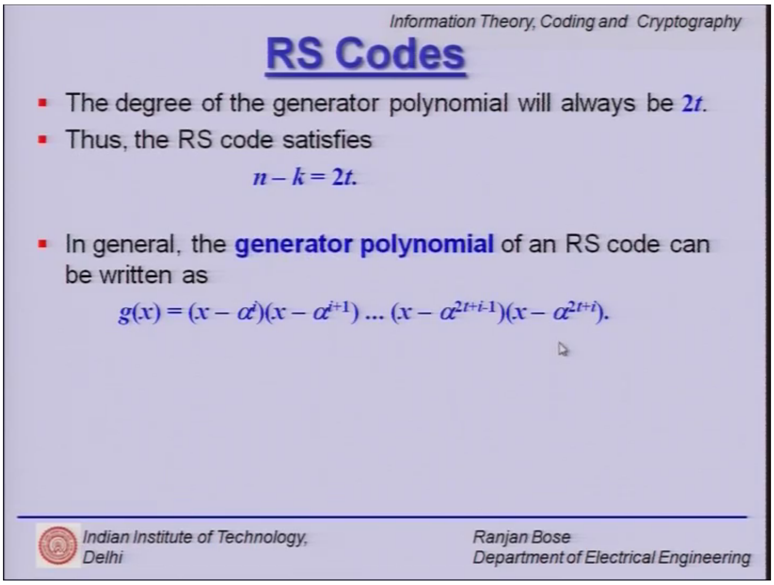
Ranjan Bose
Department of Electrical Engineering

And we are going to understand how they work today.

So, what is very important is that the BCH codes with n equal to 1 is the category of Reed Solomon Codes. That is the extension field G F q raise power m and G Fq, the sub base field they are the same. So, the field maps on to itself. So, n which is the primitive block length will be q raise power m minus 1 but m happens to be 1, therefore it is q minus 1. Now, we have already understood from BCH codes that the minimal polynomial of any element b in the same field G F q is a linear factor like this. So, the symbol field the subfield and the error locator field the extension field are the same since m is 1; all minimal polynomials are linear.

So, it is very easy to find the minimal polynomials for the Reed Solomon Codes and by that same notion, the generator polynomial g of x is nothing but LCM of f 1 x, f 2 x, up to f 2 t x. We have seen this general formulation for BCH codes but these are nothing but just x minus alpha, x minus alpha squared, x minus alpha raise power 2 t minus 1, up to x minus alpha 2 t. So, we really do not have to work hard to find the minimal polynomials, neither do we have to find the LCM because none of them repeat and it is quickly, it yields the g of x for Reed Solomon Codes.

(Refer Slide Time: 04:38)



Information Theory, Coding and Cryptography

RS Codes

- The degree of the generator polynomial will always be $2t$.
- Thus, the RS code satisfies

$$n - k = 2t.$$

- In general, the **generator polynomial** of an RS code can be written as

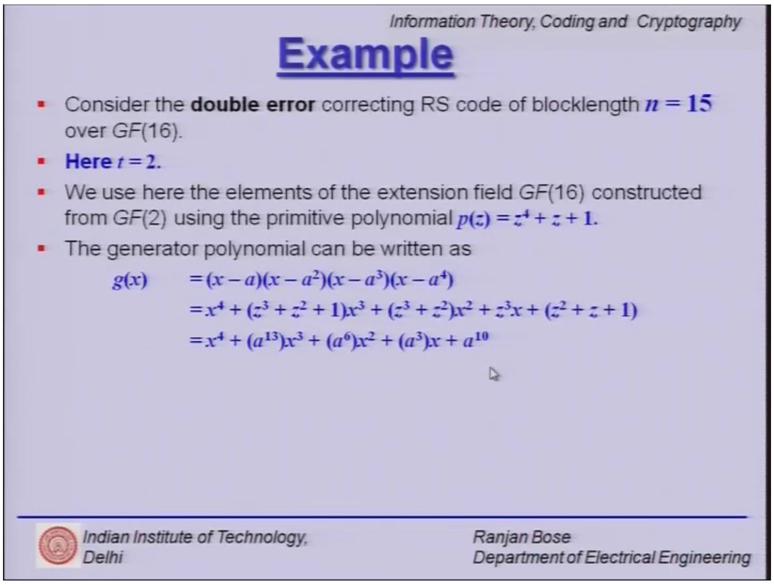
$$g(x) = (x - \alpha^{i})(x - \alpha^{i+1}) \ldots (x - \alpha^{2t+i-1})(x - \alpha^{2t+i}).$$

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, we are talking about the t error correcting Reed Solomon Codes and the degree of the generator polynomial we just saw will always be 2 raise power t from this previous example. Please note: that there are 2 t linear factors, so all the x is multiply. So, the highest power will be x raise power 2 t. So, we also know that for any BCH codes, the highest power of x, the degree of the polynomial g of x is n minus k. Therefore, n minus k equals to 2 t. And therefore, we can in general also write g of x as starting with x minus alpha raise power i, not necessarily an x raise power alpha raise power 1. It can start from anyone and go up to 2 t plus i. And this will be the generator polynomial in general for and R S Code.

(Refer Slide Time: 05:43)



Information Theory, Coding and Cryptography

Example

- Consider the **double error** correcting RS code of blocklength $n = 15$ over *GF*(16).
- **Here $t = 2$.**
- We use here the elements of the extension field *GF*(16) constructed from *GF*(2) using the primitive polynomial $p(z) = z^4 + z + 1$.
- The generator polynomial can be written as

$$\begin{aligned} g(x) &= (x-a)(x-a^2)(x-a^3)(x-a^4) \\ &= x^4 + (z^3 + z^2 + 1)x^3 + (z^3 + z^2)x^2 + z^3x + (z^2 + z + 1) \\ &= x^4 + (a^{13})x^3 + (a^6)x^2 + (a^3)x + a^{10} \end{aligned}$$

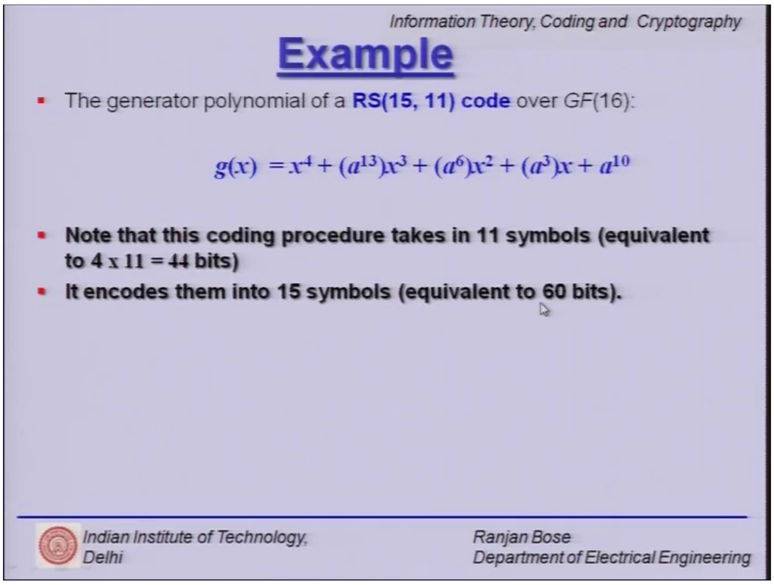
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Let us look at a quick example. We are interested in finding out a double error correcting Reed Solomon Code. We need the block length. Let n be equal to 15 and we need the Galva Field G F 16, needless to say the base field and extension field as the same. So, we have G F 16 as the working Galva Field.

Now, since there is a double error correcting code, we need to specify t equal to 2. Now we have to construct G F 16. So, we can use this a primitive polynomial to construct it and g x can be written as a linear factors. Now t is 2; so, 2 t is 4. So, we have x minus alpha, x x minus alpha squared multiplied with the x minus alpha cubed multiplied by the x minus alpha raise power 4 and this should yield to us the g of x. So, we can write it out in terms of alpha and alpha powers. And therefore, this is x raise power 4 plus alpha raise power 13 x cubed plus alpha raise power 6 x squared plus raise power 3 x plus alpha raise power 10. This is the generator polynomial for a double error correcting Reed Solomon Code.

(Refer Slide Time: 07:05)



Information Theory, Coding and Cryptography

Example

- The generator polynomial of a **RS(15, 11) code** over *GF*(16):

$$g(x) = x^4 + (a^{13})x^3 + (a^6)x^2 + (a^3)x + a^{10}$$

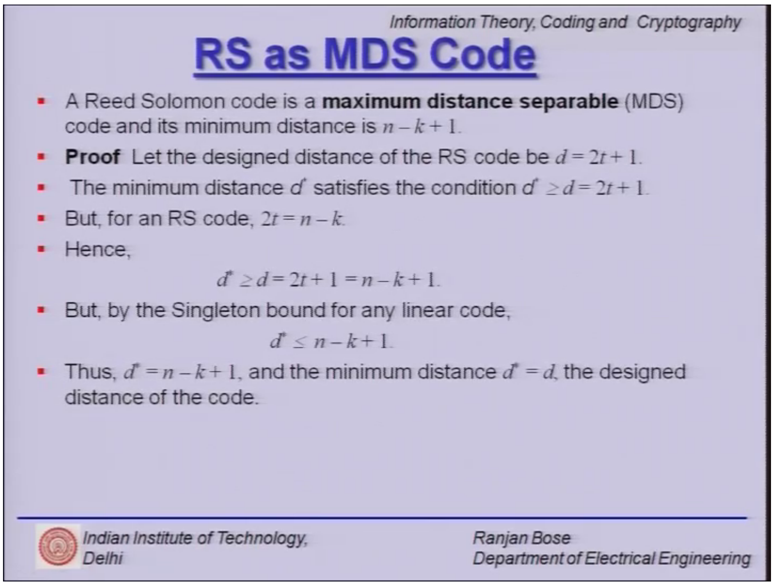
- **Note that this coding procedure takes in 11 symbols (equivalent to 4 x 11 = 44 bits)**
- **It encodes them into 15 symbols (equivalent to 60 bits).**

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, if you look at this generator polynomial, we make the following observation that this encoding procedure will take 11 symbols; why? Because it is a 15 comma 11 code, 15 comma 11 see n minus k is a highest power of g of x, n minus 4 is, n minus k is 4 n was specified to be 15 consequently, we obtained k is equal to 11. So, it takes an 11 symbols and this tantamounts to 44 bits because there are 4 bits per symbol, because we are working over G F 16. So, this encoder takes in 44 bits equivalently 11 symbols and converts them into 15 symbols which is equal to 60 bits.

(Refer Slide Time: 08:01)



Information Theory, Coding and Cryptography

RS as MDS Code

- A Reed Solomon code is a **maximum distance separable** (MDS) code and its minimum distance is $n - k + 1$.
- **Proof** Let the designed distance of the RS code be $d = 2t + 1$.
- The minimum distance d^* satisfies the condition $d^* \geq d = 2t + 1$.
- But, for an RS code, $2t = n - k$.
- Hence,

$$d^* \geq d = 2t + 1 = n - k + 1.$$

- But, by the Singleton bound for any linear code,

$$d^* \leq n - k + 1.$$

- Thus, $d^* = n - k + 1$, and the minimum distance $d^* = d$, the designed distance of the code.

Indian Institute of Technology, Delhi

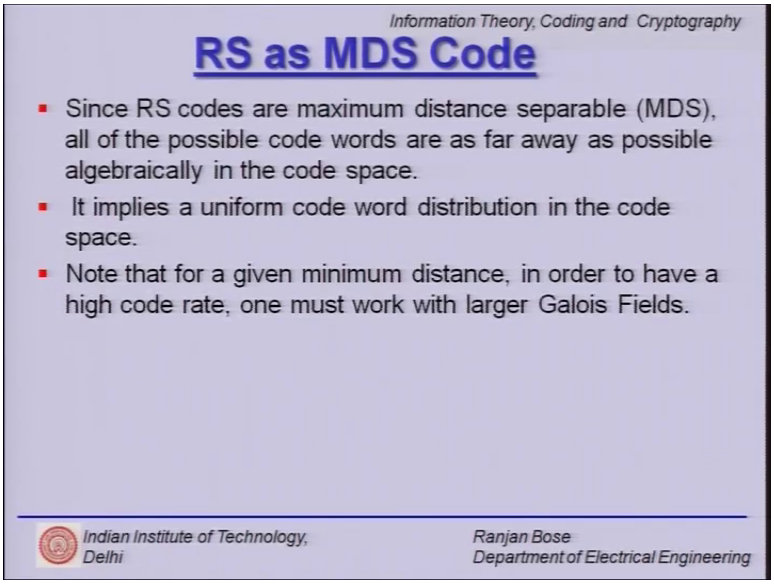
Ranjan Bose
Department of Electrical Engineering

So, let us look at some of the properties of Reed Solomon Codes. First and foremost to observe is that Reed Solomon Code are maximum distance separable codes. So, they are really good codes. Let us understand this. So, we claim that Reed Solomon Code is an M D S code and the minimum distance is n minus k plus 1. So, d star equals to n minus k plus 1; how do we prove that? So, let us say the design distance of the Reed Solomon Code, d is equal 2 t plus 1 because you know BCH codes are designer codes. We can start with the design distance that we want to make it for.

So, d star should be greater than or equal to the design distance 2 t plus 1, but you know that for Reed Solomon Code, the highest degree is 2 t for g of x which should be equal to n minus k; thus by the general understanding of cyclic codes and Reed Solomon Code forms a subclass of cyclic codes. Consequently, d is equal to 2 t plus 1 should be equal to n minus k plus 1 because 2 t is nothing but n minus k. But we have the singleton bound which says that d star should be less than or equal to n minus k plus 1, that the singleton bound. Here we have d star should be coming out to be greater than equal to n minus k plus 1, here d star is necessarily should be equal to n minus k plus 1.

So, from these two conditions, the only possible solution is d star is equal to n minus k plus 1. So, therefore, M D S code which requires d star to be n minus k plus 1 is the requirement that is satisfied and Reed Solomon Codes are M D S codes. So, in general, they are very good codes.

(Refer Slide Time: 10:05)



Information Theory, Coding and Cryptography

RS as MDS Code

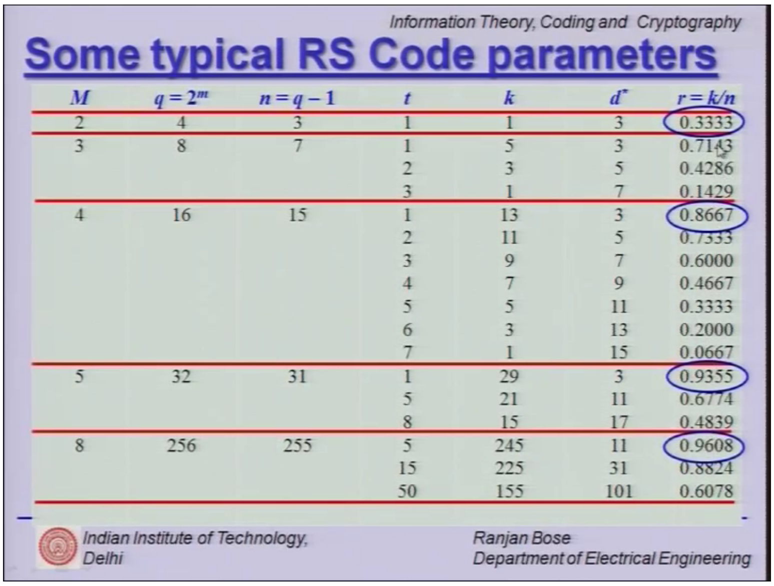
- Since RS codes are maximum distance separable (MDS), all of the possible code words are as far away as possible algebraically in the code space.
- It implies a uniform code word distribution in the code space.
- Note that for a given minimum distance, in order to have a high code rate, one must work with larger Galois Fields.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, what does this maximum distance separable code mean? Intuitively it say the code words are as far possible as possible algebraically in the code space. That is the maximum distance separable. You cannot separate them out any more. And this also implies that there is some kind of a uniform word distribution in the code space. Now we must observe one thing is that for a given minimum distance, in order to have a high code rate, one must work with larger Galva Fields. We will soon see this when we go to table of all good Reed Solomon Code and we will see how the code rate plays a role with respect to the error correcting capability.

(Refer Slide Time: 10:57)



Information Theory, Coding and Cryptography

Some typical RS Code parameters

M	$q = 2^m$	$n = q - 1$	t	k	d^*	$r = k/n$
2	4	3	1	1	3	0.3333
3	8	7	1	5	3	0.7143
			2	3	5	0.4286
			3	1	7	0.1429
4	16	15	1	13	3	0.8667
			2	11	5	0.7333
			3	9	7	0.6000
			4	7	9	0.4667
			5	5	11	0.3333
			6	3	13	0.2000
			7	1	15	0.0667
5	32	31	1	29	3	0.9355
			5	21	11	0.6774
			8	15	17	0.4839
8	256	255	5	245	11	0.9608
			15	225	31	0.8824
			50	155	101	0.6078

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

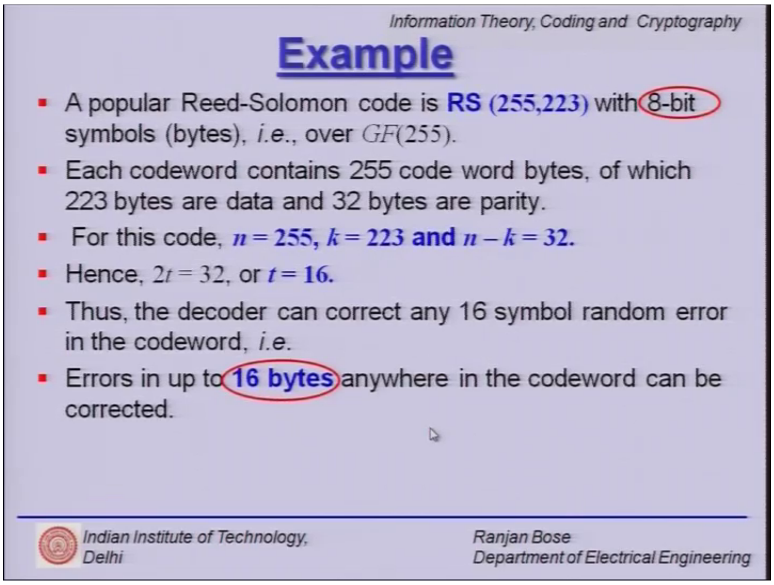
So, let us look at some typical Reed Solomon Code parameters. We have here q is equal to 2 raise power m. So, this is the Galva Field over which we are working. So, we start with non binary. So, G F 2 raise power 2, G F 2 raise power 3 and so and so forth. So, you have taken up to 256 but we can keep going. We have n values. So, you specify the n. Now, once you specify the n and the Galva Field over which we work, then what we have to do is specify the number of errors you need to correct.

So, next comes the specification of t. The moment you specify n and t for any Galva Field, rest of the job is mechanical because just take the linear factors and multiply them out. How many linear factors? 2 t; so in the first case you are now getting k equal to 1 and d star is equal to 3 and the code rate is 1 by 3 because you can see that k is equal to 1 and n is equal to 3. But you can go to higher and higher Galva Fields and for example, the moment you go to n is equal to 7 but you are in working over G F 8, you get immediately 1, 2 or even 3 error correcting code are possible because you need to have those many linear factors. Subsequently, if you go to G F 16 you can have up to 7 error correcting codes, 7 errors can be corrected in one code word.

But what we can do is make some basic observations. Let us separate the first of all the different Galva Fields and then make an observation. So, the best in the class code rate is if you can see, improve some 0.33 going up to 0.86 for Galva Field G F 16 up to 0.93, 32 and up to 256.96. So, we are going closer and closer to 1. We are getting very very efficient code; efficiency is simply because we are going to higher and higher Galva Field. But within that same class, so if you are looking at G F 256, we can have 5, 15 or even 50 error correcting codes but you can see that the code rate would drop subsequently. But you have enough possibilities to have very efficient codes here.

Let us look at another thing, suppose, we are intruded only 5 errors been corrected per code word. So, we have G F 16, 5, but the code rate is 0.33. You have got G F 32 again t is 5, so, we have 0.67; again for 256 t error correcting code t equal to 5 but the efficiency is 0.96. So, you can see a pattern, you can we have much munch fore. So, invariably, any useful practical Reed Solomon Coded are above G F 256 or even higher in real life.

(Refer Slide Time: 14:32)



Information Theory, Coding and Cryptography

Example

- A popular Reed-Solomon code is **RS (255,223)** with 8-bit symbols (bytes), *i.e.*, over $GF(255)$.
- Each codeword contains 255 code word bytes, of which 223 bytes are data and 32 bytes are parity.
- For this code, $n = 255$, $k = 223$ and $n - k = 32$.
- Hence, $2t = 32$, or $t = 16$.
- Thus, the decoder can correct any 16 symbol random error in the codeword, *i.e.*
- Errors in up to **16 bytes** anywhere in the codeword can be corrected.

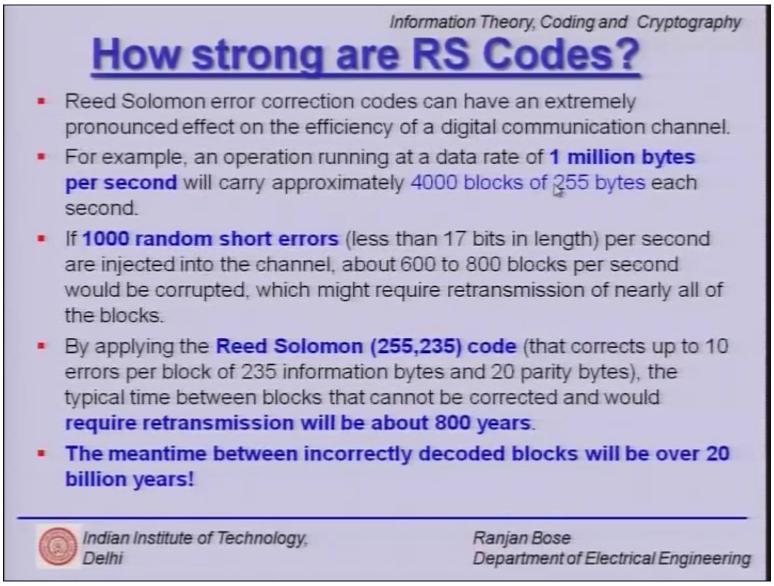
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Let us look at an example of a Reed Solomon 255 comma 223, this is a popular Reed Solomon Code with 8 bit symbols. So, each symbol is 1 byte and Galva Field is G F 255.

So, what it means is, it takes in 223 bytes of data, pads up with 32 parity bytes and then it yields 256 byte long code word. So, n minus k is 32. So, a 2 t is 32 or t is 16. So, we worked it out the other way round and so, immediately by looking at R S 255 comma 223, I can just looking at this n comma k, I can predict that t is equal to 16. So, it can correct 16 errors right but these are 16 symbol random errors. So, consequently, this code can correct up to 16 bytes anywhere in the code word. So, this is pretty strong.

(Refer Slide Time: 15:42)



Information Theory, Coding and Cryptography

How strong are RS Codes?

- Reed Solomon error correction codes can have an extremely pronounced effect on the efficiency of a digital communication channel.
- For example, an operation running at a data rate of **1 million bytes per second** will carry approximately 4000 blocks of 255 bytes each second.
- If **1000 random short errors** (less than 17 bits in length) per second are injected into the channel, about 600 to 800 blocks per second would be corrupted, which might require retransmission of nearly all of the blocks.
- By applying the **Reed Solomon (255,235) code** (that corrects up to 10 errors per block of 235 information bytes and 20 parity bytes), the typical time between blocks that cannot be corrected and would **require retransmission will be about 800 years**.
- **The meantime between incorrectly decoded blocks will be over 20 billion years!**

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

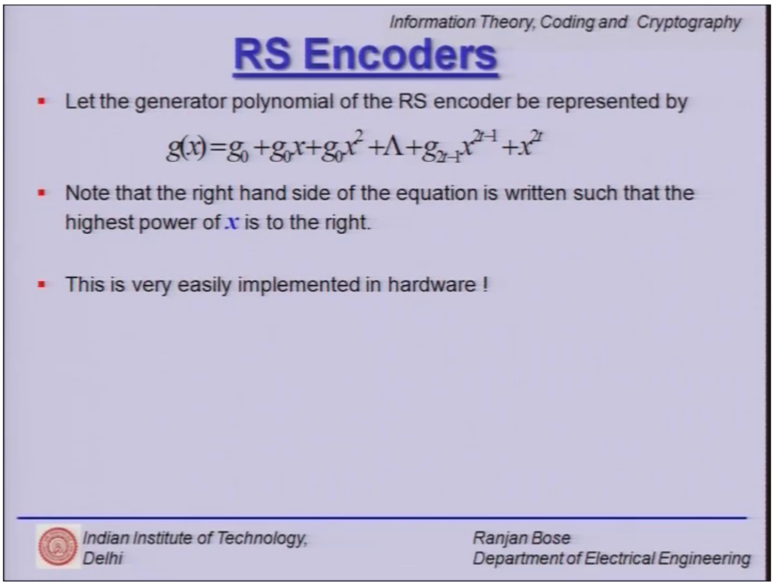
So, how strong really are Reed Solomon Codes? That is the question we need to ask ourselves. So, they are definitely extremely pronounced effect on the efficiency of a digital communication channel because of their strong error correcting capability. So, let us take a very simple calculation back of the envelope calculation. Suppose we have operation which is transmitting 1 million bytes per second ok. So, this is a data rate of a high speed communication link. So, approximately, we have about 4000 blocks of 255 bytes each per second ok. So, this if you multiply them, it is roughly of the order of 1 million bytes per second. So, we have consequently 4000 blocks of 255 bytes. So, these are code words. So, 4000 code words per second are being sent right.

Now, suppose, 1000 random short errors less than 17 bits in length per second are injected into the channel, this is just an example. So, about 600 to 800 blocks per second would be corrupted right. So, if this was not protected, then all the blocks are most of the blocks would require retransmission. But let us suppose we want to apply Reed Solomon

Codes right, then what we do it for every 255 bytes, we pack them up and we get this 255 byte long code words and so, we are padding up with 20 parity bytes. If you employ this and look at the error correcting and detecting capability, we would see that reed transmission will not be required for 800 years at the rate of 1 million bytes per second. That is how strong it is, that is the part of the story. And this R S code will also make errors in detection but the meantime between in correctly decoded blocks will be over 20 billion years.

So, in a real practical world, this application of this 255 comma 235 code practically, makes it a very very useful almost error communication, even I working at this high rate of 1 million bytes per second and this efficiency is also not too bad. Of course, we can improve the efficiency the code rate of this code by going to higher Galva Fields. So, the point is that Reed Solomon Codes are very strong practical codes.

(Refer Slide Time: 18:30)



Information Theory, Coding and Cryptography

RS Encoders

- Let the generator polynomial of the RS encoder be represented by

$$g(x)=g_0+g_0x+g_0x^2+\Lambda+g_{2t-1}x^{2t-1}+x^{2t}$$

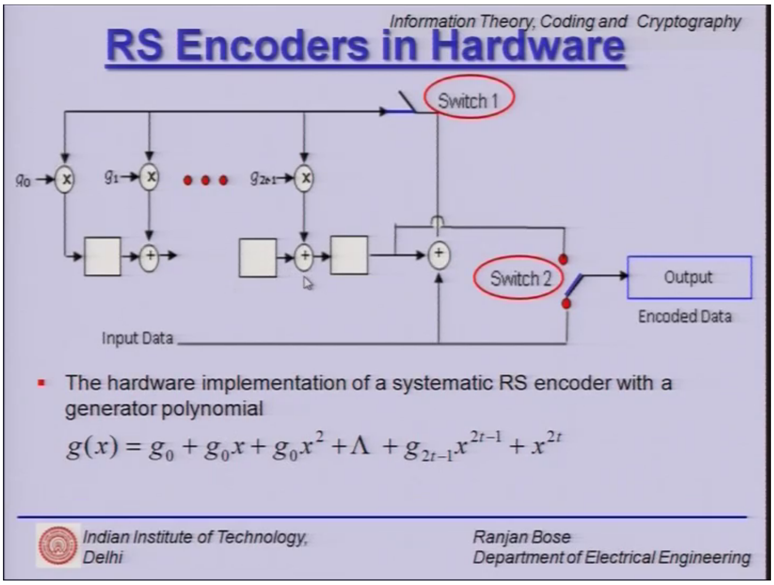
- Note that the right hand side of the equation is written such that the highest power of x is to the right.
- This is very easily implemented in hardware !

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us look at the encoder representations and we would like to encode the following generator polynomial. So, this is the epsilon g nought, g 1, g 2, up to dot, dot, 2 t minus 1 up to this monic, x raise power 2 t. So please note: we have arranged it. So, that the highest power of x is to the right. This is a very easily implementable form in hardware.

(Refer Slide Time: 19:03)



Information Theory, Coding and Cryptography

RS Encoders in Hardware

- The hardware implementation of a systematic RS encoder with a generator polynomial

$$g(x) = g_0 + g_0 x + g_0 x^2 + \Lambda + g_{2t-1} x^{2t-1} + x^{2t}$$

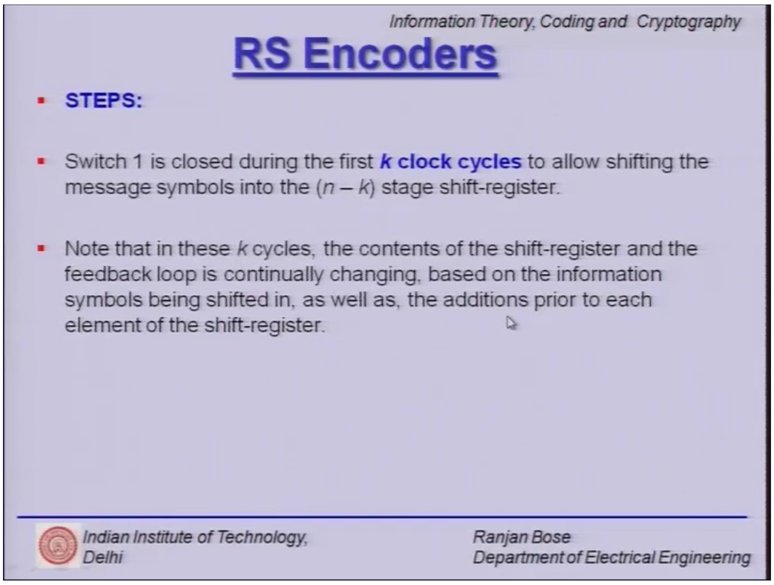
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, how does the hardware encoder for Reed Solomon look like? So, please note there is a shift register, there is some coefficients which will pump in the values for g of x and then you have two switches; switch 1 and switch 2. And the way to work is the first there will be k clock cycles wherein this will be a systematic codes which consequently means that the input data goes directly to the output.

So please, coming to this hardware. So, for every time, I clock in the input, it goes out. So, the first k symbols are the same because it is systematic. The rest n minus k symbols, I will close this switches and get the encoded version which is multiplied with g of x.

(Refer Slide Time: 20:04)



Information Theory, Coding and Cryptography

RS Encoders

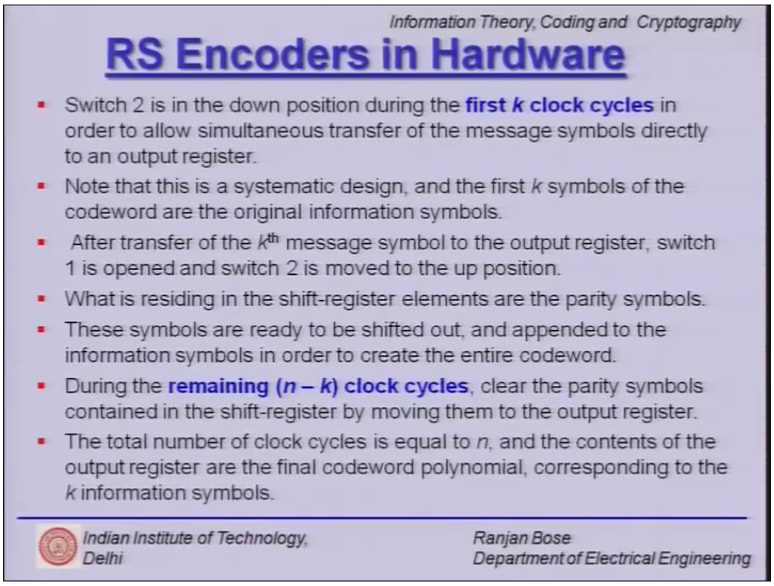
- **STEPS:**
- Switch 1 is closed during the first ***k* clock cycles** to allow shifting the message symbols into the ($n - k$) stage shift-register.
- Note that in these *k* cycles, the contents of the shift-register and the feedback loop is continually changing, based on the information symbols being shifted in, as well as, the additions prior to each element of the shift-register.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, if you look at the steps; switch 1 is closed in using the first k clock cycles to allow the shifting of the message right; n minus k shift registers. Now, this k cycles the contents of the shift register and the feedback loop is continuously changing. This shows the information symbols being shifted in as well as a addition prior to each elements of the shift register. So, what has happened is the switch 1 is closed and switch 2 is connected here. So, by the time this data is being clocked, please note these value of the shift registers are also continuously changed.

(Refer Slide Time: 20:49)



Information Theory, Coding and Cryptography

RS Encoders in Hardware

- Switch 2 is in the down position during the **first *k* clock cycles** in order to allow simultaneous transfer of the message symbols directly to an output register.
- Note that this is a systematic design, and the first *k* symbols of the codeword are the original information symbols.
- After transfer of the k^{th} message symbol to the output register, switch 1 is opened and switch 2 is moved to the up position.
- What is residing in the shift-register elements are the parity symbols.
- These symbols are ready to be shifted out, and appended to the information symbols in order to create the entire codeword.
- During the **remaining ($n - k$) clock cycles**, clear the parity symbols contained in the shift-register by moving them to the output register.
- The total number of clock cycles is equal to *n*, and the contents of the output register are the final codeword polynomial, corresponding to the *k* information symbols.

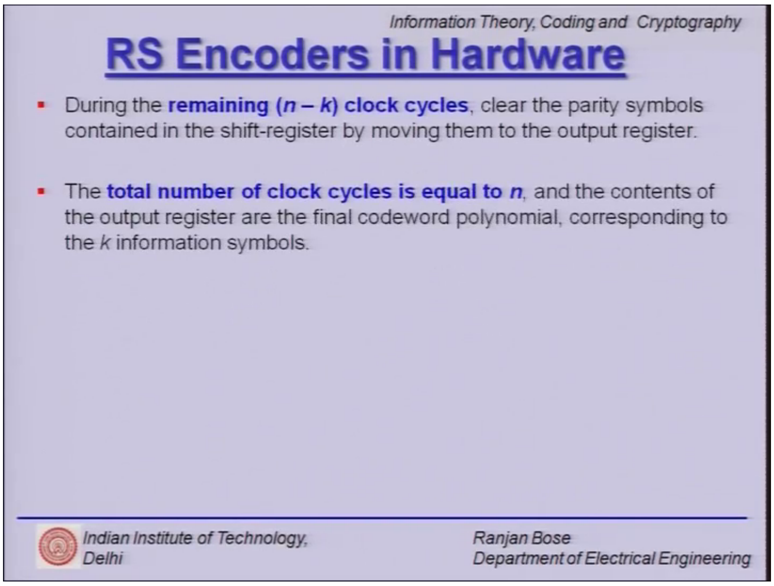
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, the switch 2 is down for the first k cyclic k is clock cycles allowing the simultaneous transfer of the message symbol directly to the output register leading to the systematic Reed Solomon Code. After the transfer of the k-th massage symbol to the output register, switch 1 is opened and switch 2 is moved to the up position. So, what we do is in after this k symbol, switch 2 is move to the up position because we would like to read out what is present in this shift register right and this one is open. So, no more further changes will be happening.

So, what is residing the shift register elements are actually the parity symbols. This symbol ready to be shifted out and appended to the information symbols already stored in a buffer which will create the entire code word. So, please note the very efficient hardware implementation, first k clock cycles and second k clock cycle things happening in parallel right. So, during the remaining n minus k clock cycles, we clear the parity symbols contained in the shift register by moving them to the output register and we are ready to encode the next code word. So, total number of clock cycles is equal to n and the contents of the output registers are actually the final code word polynomial corresponding to the k information symbols.

(Refer Slide Time: 22:23)



Information Theory, Coding and Cryptography

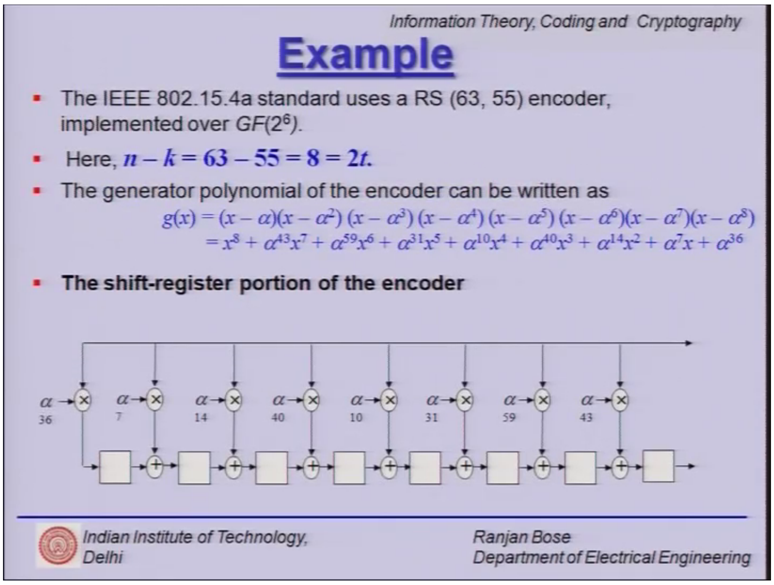
RS Encoders in Hardware

- During the **remaining ($n - k$) clock cycles**, clear the parity symbols contained in the shift-register by moving them to the output register.
- The **total number of clock cycles is equal to n**, and the contents of the output register are the final codeword polynomial, corresponding to the k information symbols.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

(Refer Slide Time: 22:26)



Information Theory, Coding and Cryptography

Example

- The IEEE 802.15.4a standard uses a RS (63, 55) encoder, implemented over $GF(2^6)$.
- Here, $n - k = 63 - 55 = 8 = 2t$.
- The generator polynomial of the encoder can be written as

$$g(x) = (x-\alpha)(x-\alpha^2)(x-\alpha^3)(x-\alpha^4)(x-\alpha^5)(x-\alpha^6)(x-\alpha^7)(x-\alpha^8)$$
$$= x^8 + \alpha^{43}x^7 + \alpha^{59}x^6 + \alpha^{31}x^5 + \alpha^{10}x^4 + \alpha^{40}x^3 + \alpha^{14}x^2 + \alpha^7 x + \alpha^{36}$$

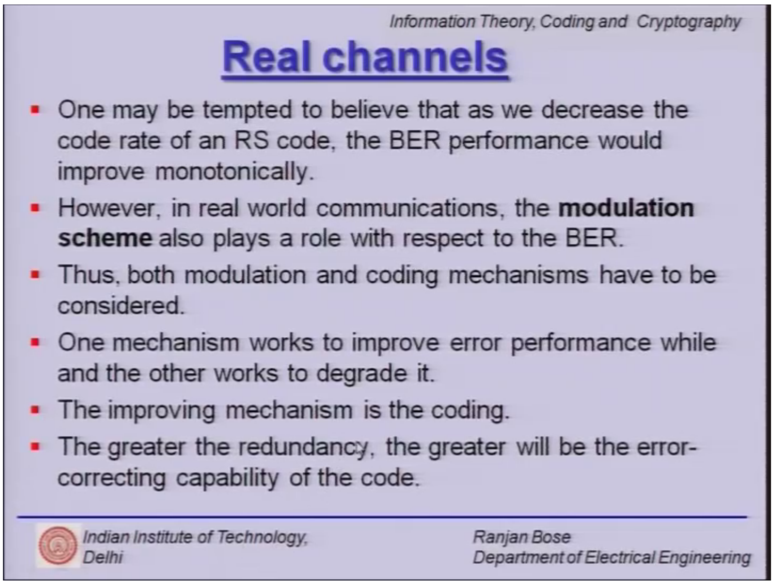
- **The shift-register portion of the encoder**

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us look at a very simple example. This is a from a standard. IEEE to do dot 15 dot 4 a standard, it is an R S 63 comma 55 encoder. So, it can be very easily implemented in on an G F a. So, here n minus k is 63 minus 55 equal to 8 is equal to 2 t consequently it is a 4 symbol error correcting code. So, we multiply it out very easily, x minus alpha, x minus alpha squared so and so forth up to x minus alpha raise power 2 t, 2 t happens to be 8. So, we have these 8 linear factors multiply it out and immediately, you have the g of x and the corresponding shift register portion of the hardware encoder is given here. These are the coefficients of your g of x. So, this is a simple example, how we can include in R, R S encoder this notion of the generator polynomial.

(Refer Slide Time: 23:35)



Information Theory, Coding and Cryptography

Real channels

- One may be tempted to believe that as we decrease the code rate of an RS code, the BER performance would improve monotonically.
- However, in real world communications, the **modulation scheme** also plays a role with respect to the BER.
- Thus, both modulation and coding mechanisms have to be considered.
- One mechanism works to improve error performance while and the other works to degrade it.
- The improving mechanism is the coding.
- The greater the redundancy, the greater will be the error-correcting capability of the code.

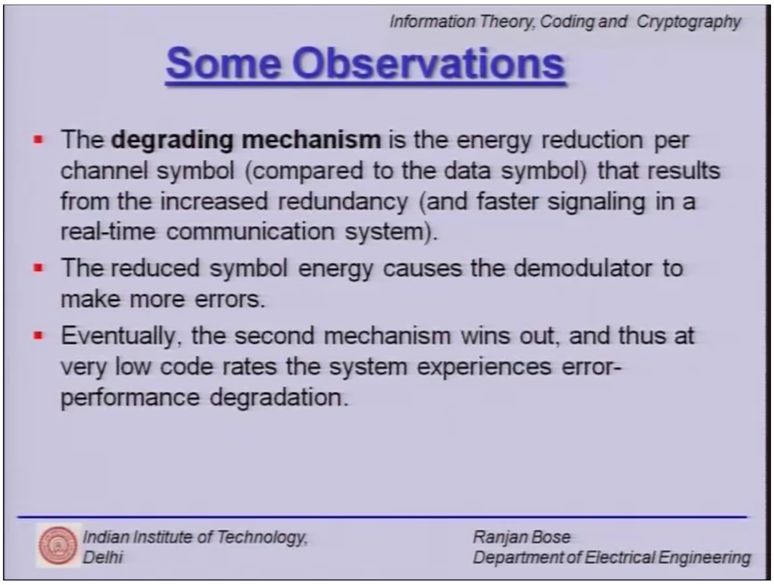
Indian Institute of Technology, Delhi
Ranjan Bose
Department of Electrical Engineering

Now, we come to the question of real channels. How do they perform over real channels? So, word of caution, one may be tempted to believe that as we decrease the code rate of a Reed Solomon Code, the B E R performance which improve monotonically because you would believe that you have you have padding an extra bits. They are overhead bits hopefully, your B E R performance which improve.

However, in real world communication, the modulation scheme also plays a role with respect to B E R. Thus both modulation and coding mechanisms have to be considered. So, we are now touching base with reality just because you have encoded a symbols using an Reed Solomon Encoder, offer that matter any encoder is does not mean that your bitter rate will keep going down. If you increase the redundancy, we must consider the modulation part. So, you have to know that one of the mechanisms improves the error performance while the other would work to degrade it. So, modulation, we have to high modulation scheme would cause more errors to happen and this edition of additional parity bits reduces the B E R. So, they are working opposite to each other.

So, let us see, the improving mechanism is coding the greater the redundancy, the greater will be the error correcting capability of the code but if you pack in too many redundancy, you have to switch to higher modulation schemes which will make it counterproductive.

(Refer Slide Time: 25:23)



Information Theory, Coding and Cryptography

Some Observations

- The **degrading mechanism** is the energy reduction per channel symbol (compared to the data symbol) that results from the increased redundancy (and faster signaling in a real-time communication system).
- The reduced symbol energy causes the demodulator to make more errors.
- Eventually, the second mechanism wins out, and thus at very low code rates the system experiences error-performance degradation.

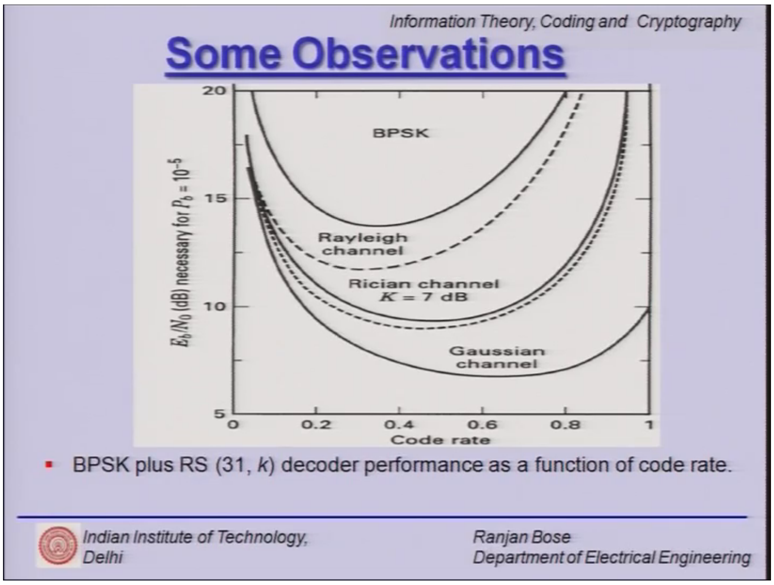
Indian Institute of Technology,
Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us make some observation that the degrading mechanism is the energy reduction per channel symbol compared to the data symbol. That results from increased redundancy and faster signalling in a real time communication system. So, your energy per symbol also degrades. The reduce symbol energy causes demodulator to make errors ok. So, just packing in additional bits, padding it with additional parity bits would overall reduce the symbol per the energy symbol and causing it to have a poor performance.

So, there is a trade-off and somewhere, there will be an optimal solution and before which the second mechanism wins out and very low rate codes would again have poor performance. So, there is a trade off mechanism happening.

(Refer Slide Time: 26:20)



Information Theory, Coding and Cryptography

Some Observations

- BPSK plus RS (31, *k*) decoder performance as a function of code rate.

Indian Institute of Technology, Delhi

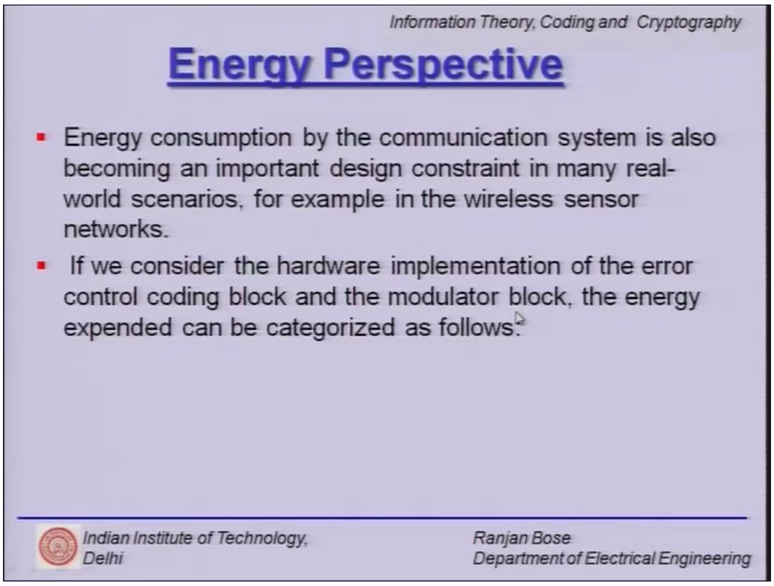
Ranjan Bose
Department of Electrical Engineering

Let us look at this. So, if you look at this curve, on the x axis, we have the code rate ok, the efficiency of the code and on the y axis, we have the energy per E b over N naught. So, it is the normalised S N R and we looking at a typical error rate of 10 raise power minus 5. So, what does it take to achieve this P b equal to 10 raise power minus 5 and if you look at, we are looking at Reed Soloman 31 comma k code.

So, if you change the k, your code rate changes. So, by changing the different values of k for this Reed Solomon 31 comma k, I move along the x axis and we can eventually compute E b over N naught and get plot these performances over different kinds of channels. So, we have a Gaussian channel, we have a Rician channel, we have a Rayleigh channel. These 2 are fading channels, Rician channels when you have a direct line of sight, Rayleigh channels when you do not have a direct line of sight from the transmitter to the receiver and Gaussian channel as you know is a standard e w g n channel.

So, if you just consider the Gaussian channel, the optimal point where you have the minimum E b over N naught because your performance be measured in terms of E b right 10 raise power minus 5. So, all this curves are for the same performance level, but here, you required the minimum E b over N naught energy is the minimum S N R requirement for this code rate. If you go beyond that again, you have to pump in more energy per symbol. So, all of them go through a minimum. So, this is an important observation from a real world channel. I will agree we would have believed that as the code rate goes down your performance would necessarily increase with the E b over N naught.

(Refer Slide Time: 28:33)



Information Theory, Coding and Cryptography

Energy Perspective

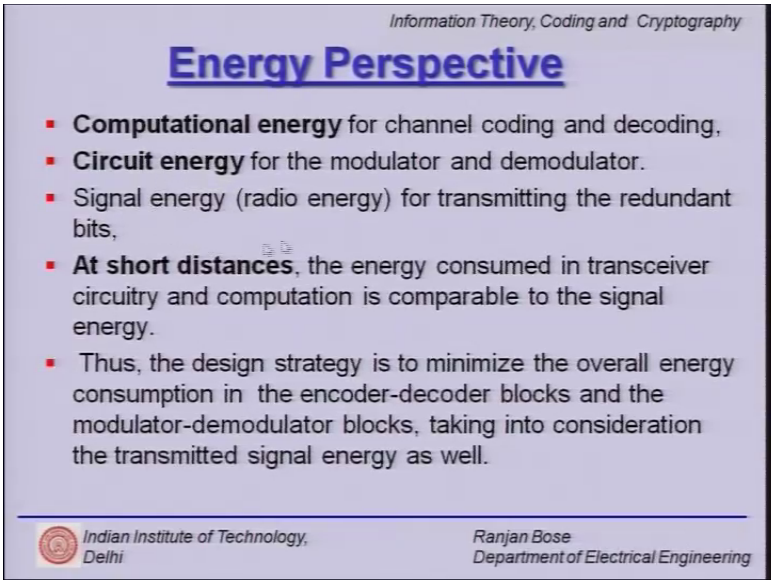
- Energy consumption by the communication system is also becoming an important design constraint in many real-world scenarios, for example in the wireless sensor networks.
- If we consider the hardware implementation of the error control coding block and the modulator block, the energy expended can be categorized as follows:

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, let us spend some time from the Energy Perspective. Today, energy is essential in the design aspect of all communication systems, beat for the green telecom perspective, increase battery life or better performance, energy consumption by communication system is a interesting and important design constraint. If you look at situation like wireless sensor networks where we are necessarily limited by the energy available, we must take it into consideration. If you consider the hardware implementation of the error control coding block and the modulator block, the energy expended can be categorised as follows.

(Refer Slide Time: 29:20)



Information Theory, Coding and Cryptography

Energy Perspective

- **Computational energy** for channel coding and decoding.
- **Circuit energy** for the modulator and demodulator.
- Signal energy (radio energy) for transmitting the redundant bits.
- **At short distances**, the energy consumed in transceiver circuitry and computation is comparable to the signal energy.
- Thus, the design strategy is to minimize the overall energy consumption in the encoder-decoder blocks and the modulator-demodulator blocks, taking into consideration the transmitted signal energy as well.

Indian Institute of Technology, Delhi

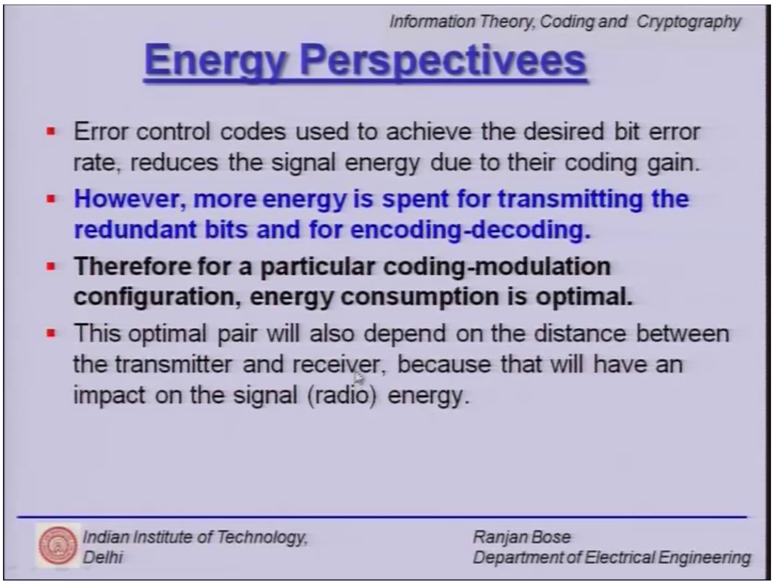
Ranjan Bose
Department of Electrical Engineering

There is a computation energy for channel encoding and decoding. This is important. So far, we just believe that multiplying over various Galva Fields, adding, do all that computation requires negligible energy, but that may not be true. You could possibly end up spending a lot of energy just doing the computation; encoding and decoding. I mean, it is possible that you are spending more energy encoding and decoding, than saving in terms of getting that same better rate. Then there is a circuit energy required for modulation and demodulation.

And then you have the signal energy the radio energy which is actually used for transmitting all of those bits including the redundant bits. So, at short distances, the energy consumed in the transceiver circuitry and computation is comparable to the signal energy; why? Because the required radio energy, the actual energy for the electromagnetic radiation is very small and it becomes comparable to what we spend inside the circuit in terms of the circuit energy and the computational energy.

So, the overall design objective should be to minimise the total energy including computational energy, circuit energy and the radio energy. If you take all of them together and maximize the performance or minimise the total energy, then we have an energy perspective to the whole thing.

(Refer Slide Time: 30:54)



Information Theory, Coding and Cryptography

Energy Perspectivees

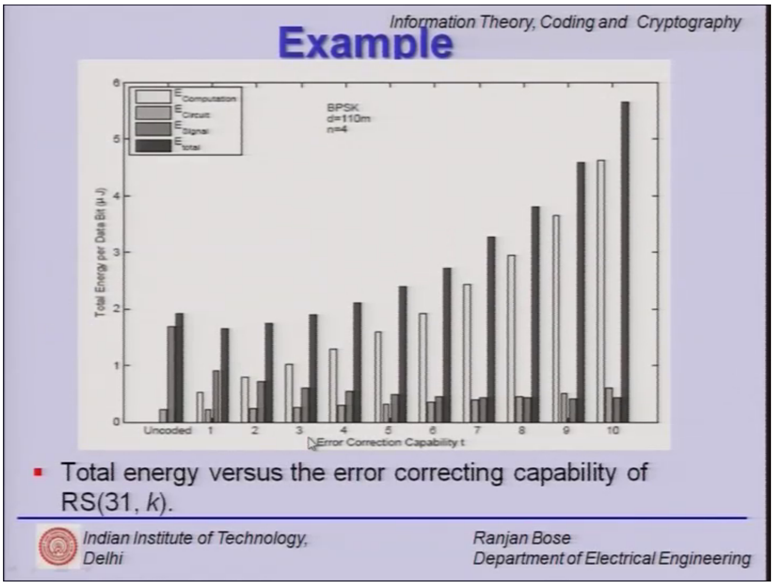
- Error control codes used to achieve the desired bit error rate, reduces the signal energy due to their coding gain.
- **However, more energy is spent for transmitting the redundant bits and for encoding-decoding.**
- **Therefore for a particular coding-modulation configuration, energy consumption is optimal.**
- This optimal pair will also depend on the distance between the transmitter and receiver, because that will have an impact on the signal (radio) energy.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, they have to make sure that for a particular coding and modulation configuration, we have to optimize the energy consumption. So, the optimal pair of energy of coding and modulation will also depend on the distance between the transmitter and receiver; why is that; because the radio energy depends on the distance between the transmitter and receiver.

(Refer Slide Time: 31:23)



Information Theory, Coding and Cryptography

Example

- Total energy versus the error correcting capability of RS(31, *k*).

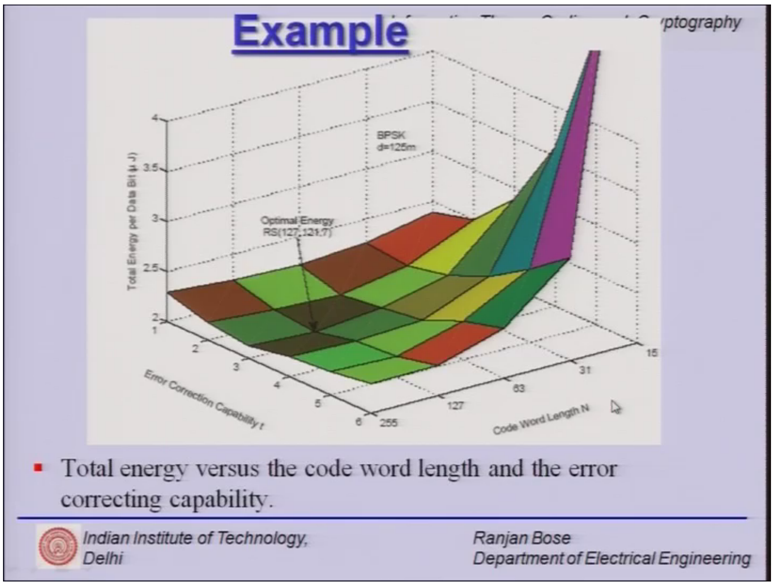
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us look at this example. This is an example using a Reed Solomon 31 comma k code. So, k can be changed and there by changing the error correcting capability. So, in this graph, you have on the x axis the error correction capability t and the y axis, the total energy per data bit in micro joules.

Now, we have by this differentiates that the different components; what are the components of energy; the computation energy, the circuit energy and the signal energy which is the radio energy. So, the darkest is the signal energy followed by the circuit energy; the lightest is the computation energy. So, if you see that the total dark line is the total energy; so the different components there and as we increase the error correcting capability all of them.

(Refer Slide Time: 32:25)



ptography

Example

- Total energy versus the code word length and the error correcting capability.

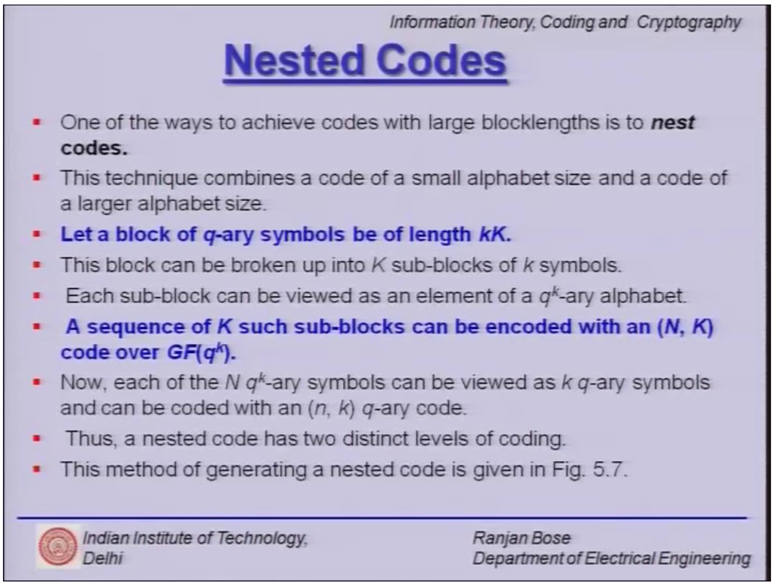
Indian Institute of Technology,
Delhi

Ranjan Bose
Department of Electrical Engineering

Increase consequently leading to an increased total energy per data bit, in terms of micro joules. Now, if you look at it from another perspective, suppose on this axis you have the code word length, on this axis you have the error correcting capability and on the z axis you have the total energy per data bit in terms of the micro joules. So, we are plotting the total energy verses the code word length and the error correcting capability together. So, you can get a feel that for example the optimal energy comes here for R S 127 comma 121.

So, it is not obvious just by looking at the n and k of a Reed Solomon Code, what is the best code Reed Solomon Code to pick. It is not just the error correcting capability because in an earlier table, we had plotted with respect to the error correcting capability t; that table is not itself sufficient if you are considering the energy perspective. You have to look at the other aspects of the energy as well.

(Refer Slide Time: 33:44)



Information Theory, Coding and Cryptography

Nested Codes

- One of the ways to achieve codes with large blocklengths is to ***nest* codes.**
- This technique combines a code of a small alphabet size and a code of a larger alphabet size.
- **Let a block of *q*-ary symbols be of length *kK*.**
- This block can be broken up into *K* sub-blocks of *k* symbols.
- Each sub-block can be viewed as an element of a q^k-ary alphabet.
- **A sequence of *K* such sub-blocks can be encoded with an (*N*, *K*) code over *GF*(q^k).**
- Now, each of the *N* q^k-ary symbols can be viewed as *k* *q*-ary symbols and can be coded with an (*n*, *k*) *q*-ary code.
- Thus, a nested code has two distinct levels of coding.
- This method of generating a nested code is given in Fig. 5.7.

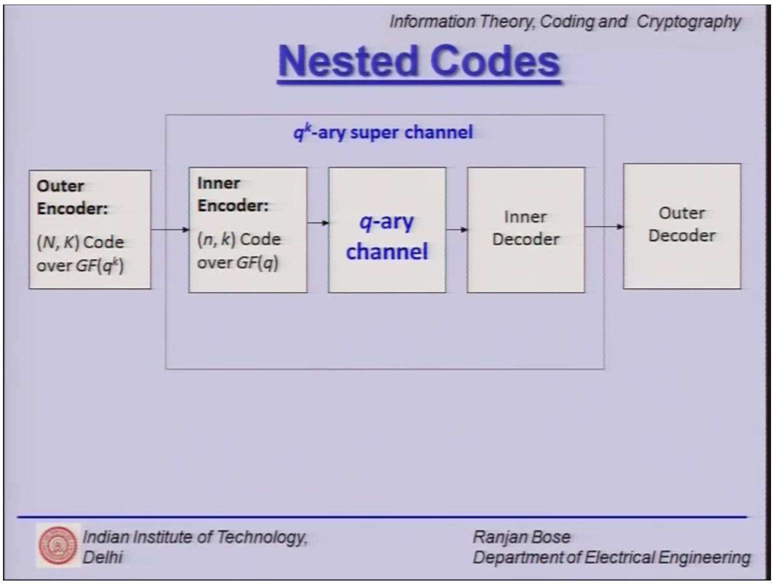
Indian Institute of Technology, Delhi *Ranjan Bose Department of Electrical Engineering*

Now, we come to the notion of Nested Codes. So, one of the ways to achieve codes with large block length is to nest codes. What do we mean by that? In this technique, we combine a code of small alphabet size and a code of a larger alphabet size. So, we are combining two codes and nest them. We will have a notion of an inner code and an outer code.

So, let a block of q-ary symbols be of the length small k, capital K. So, k K is a length of a block. Now, these two k would correspond to the two different codes; one with a small alphabet size and one with a large alphabet size. So, you can write this as k sub blocks, capital K sub blocks of small k symbols making it k into capital K symbols all together. So, each sub block can be viewed as an element of a q raise power k q-ary alphabet alright. So, a sequence of capital K such sub blocks can be further encoded with an capital N comma capital K code over G Fq k, alright.

So, we with two step process; now, each of the N q raise power k q-ary symbols can be viewed as a k q-ary symbols that can be encoded with an n comma k q-ary code. So, therefore, you nesting it out. So, the nested code has two distinct levels of coding. So, it is coding over coding.

(Refer Slide Time: 35:40)



Information Theory, Coding and Cryptography

Nested Codes

q^k-ary super channel

Outer Encoder: (N, K) Code over $GF(q^k)$

Inner Encoder: (n, k) Code over $GF(q)$

q-ary channel

Inner Decoder

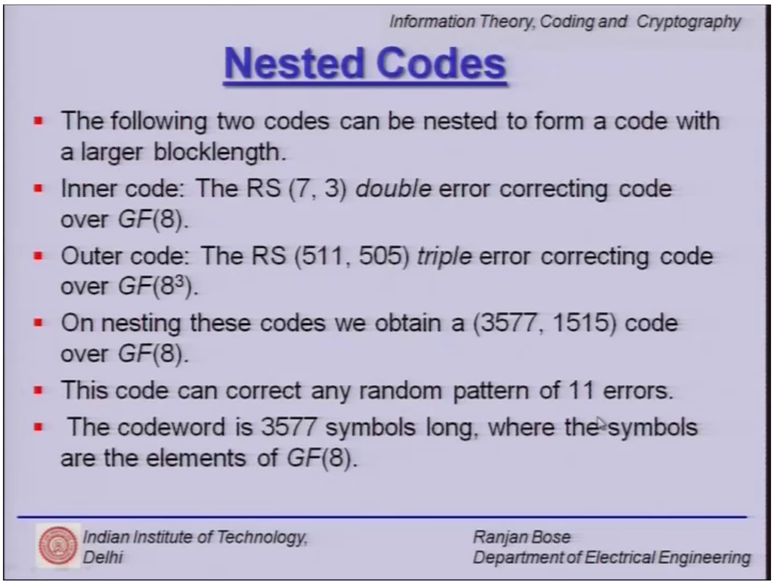
Outer Decoder

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us look at how we are trying to do it. At the centre of it is the q-ary channel, so, you have a notion of the inner code. What does the inner encoder do? It takes n comma k code over G Fq. So, it is q sorry k symbols and converts it into n symbols. And then, this guy is encoded further using this n k code. So, together, this is the outer code and inner code form the nested code. You have to reverse the process on the decoding side. So, you have a inner decoder which decodes it small n comma small k code and then there is a outer decoder which decodes capital N a comma capital K codes. So, you have a coding over coding and this is also used in practice.

(Refer Slide Time: 36:38)



Information Theory, Coding and Cryptography

Nested Codes

- The following two codes can be nested to form a code with a larger blocklength.
- Inner code: The RS (7, 3) *double* error correcting code over $GF(8)$.
- Outer code: The RS (511, 505) *triple* error correcting code over $GF(8^3)$.
- On nesting these codes we obtain a (3577, 1515) code over $GF(8)$.
- This code can correct any random pattern of 11 errors.
- The codeword is 3577 symbols long, where the symbols are the elements of $GF(8)$.

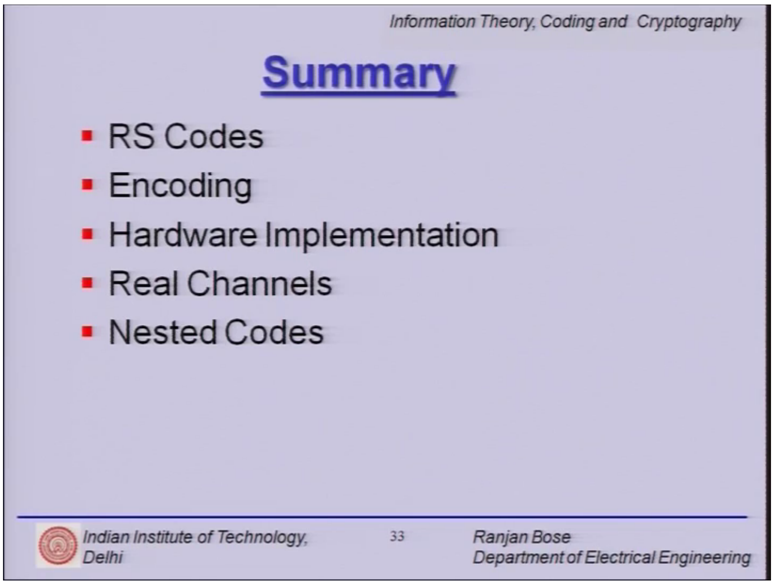
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us look at some examples. So, suppose, we have an inner code the R S 7 comma 3 double error correcting code over G F 8 and we have an outer code the Reed Solomon 511 comma 505 triple error correcting code over G F 8 raise power 3. So, if you nest these codes ok, so this is a small k 3 this is a capital K 505. So, small k into capital K should be 1515. So, the nesting code the super code is 3577 comma 1515 code over G F 8. Now is it good; well this code can connect any random pattern of 11 errors and the code word is 3577 symbol long. And what is the symbol? Symbol is an element of G F 8.

So, this is a simple example of nesting R S 7 comma 3 with R S 51 comma 505 leading us to a big nested code of 3577 comma 1515 code over G F 8.

(Refer Slide Time: 38:02)



Information Theory, Coding and Cryptography

Summary

- RS Codes
- Encoding
- Hardware Implementation
- Real Channels
- Nested Codes

Indian Institute of Technology, Delhi 33 Ranjan Bose Department of Electrical Engineering

So, now let us summarise what we have done so far. We introduced in this lecture the concept of Reed Solomon Codes which is the subclass of BCH codes. We looked at the encoding procedure. Specifically, we looked at the hardware implementation, it is very easy to encode Reed Solomon Codes using shift registers and two switches. Then we moved on to real channels and the performance over real channels for Reed Solomon Codes. We also considered the importance of overall energy optimisation while designing the error control codes. And finally, we talked about nested codes.

With that we come to the end of this lecture.