

**Indian Institute of Technology Kanpur
National Programme on Technology Enhanced Learning (NPTEL)**

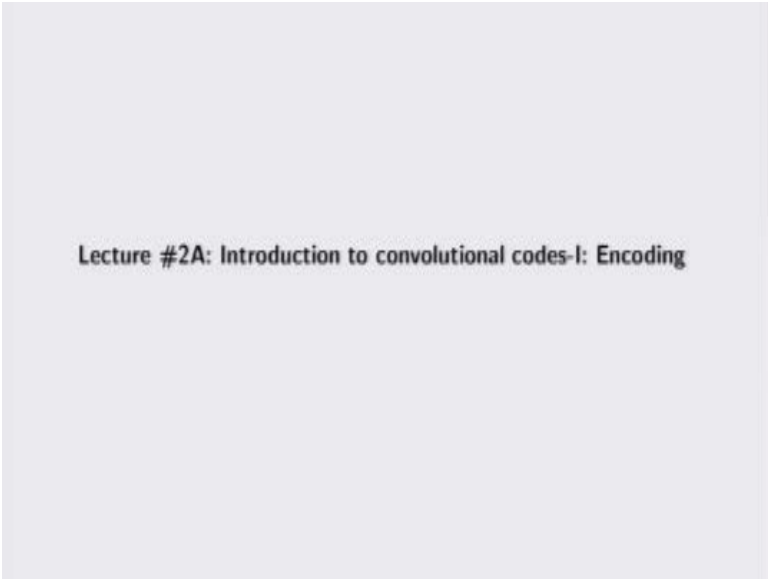
**Course Title
Error Control Coding: An Introduction to Convolution Codes**

**Lecture - 02
Introduction to Convolution Codes-I
Encoding**

**by
Prof. Adrsih Banerjee
Dept. Electrical Engineering Psychiatrist, IIT Kanpur**

Welcome to course on error control coding an introduction to convolutional code.

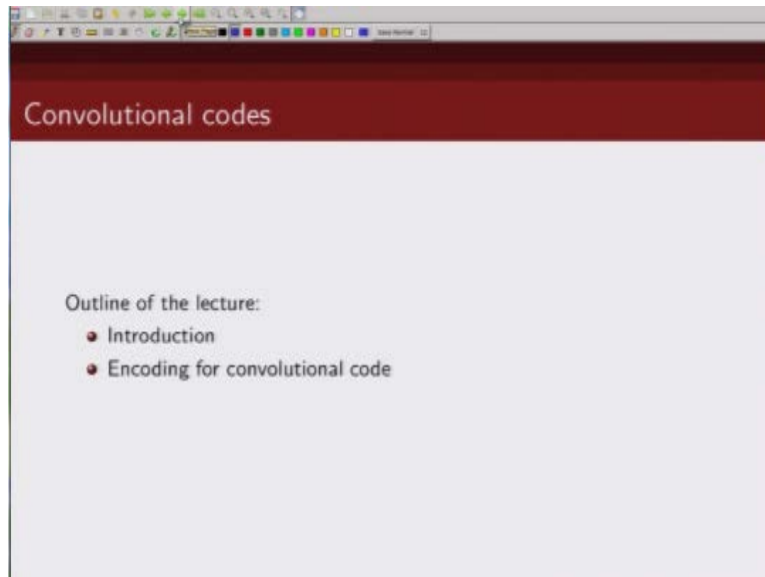
(Refer Slide Time: 00:19)



Lecture #2A: Introduction to convolutional codes-I: Encoding

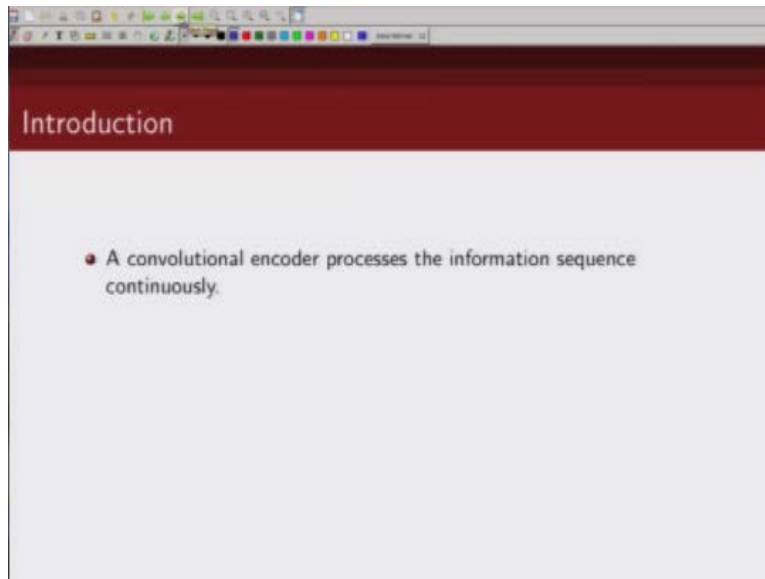
So we will start with introduction of convolutional code and today we are going to discuss how we can encode an information sequence using convolutional code.

(Refer Slide Time: 00:31)



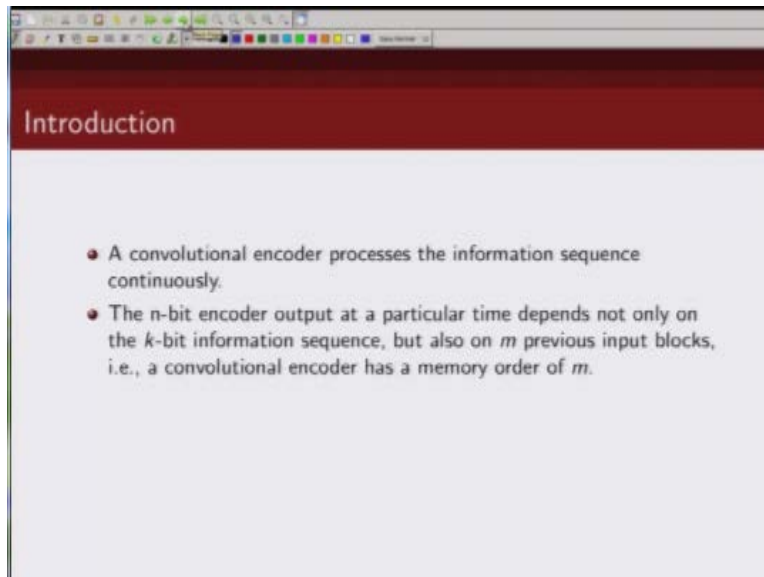
So today's topic of discussion is encoding of convolutional code and we will take a very simple example of a rate $1/n$ convolutional code.

(Refer Slide Time: 00:44)



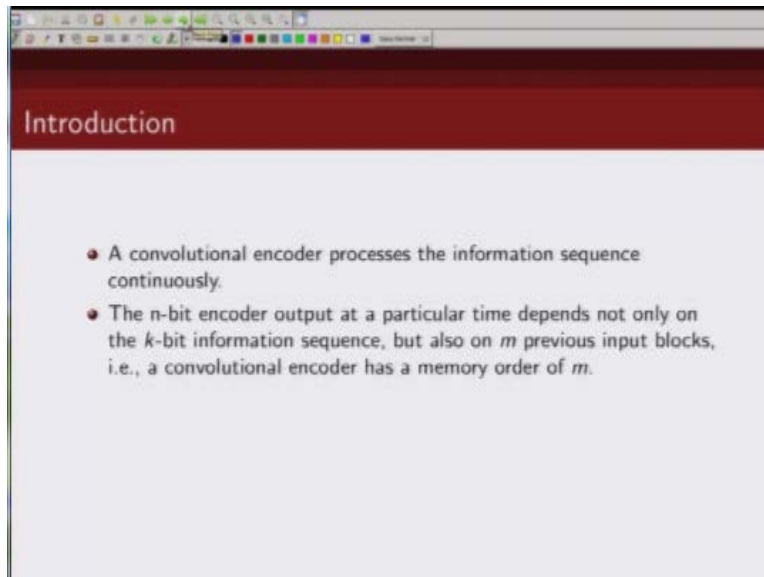
As you know a convolutional code process information sequence in a continuous fashion, so information bits come in and we can get continues output form a convolutional encoder.

(Refer Slide Time: 00:57)



We also know that the output of a convolutional encoder depends not only on the current input but it also depends on past inputs and past outputs depending up on the memory of the convolutional encoder.

(Refer Slide Time: 01: 16)

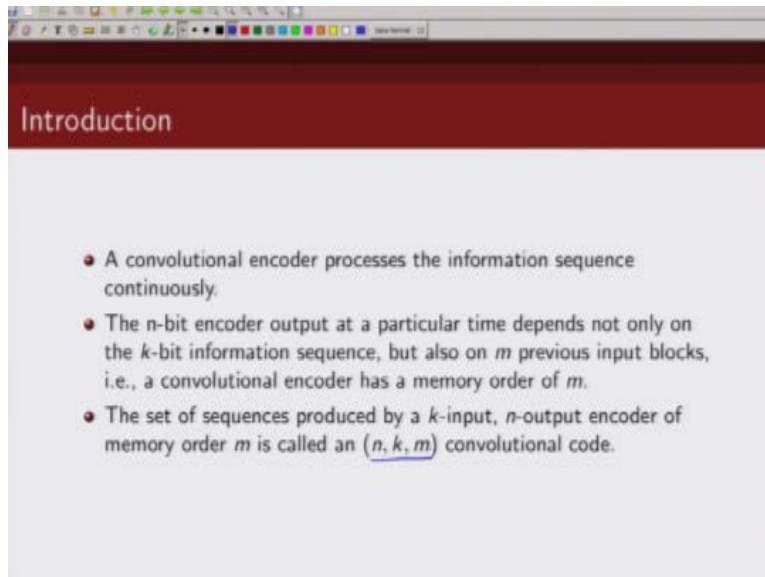


And as we have seen we can realize the convolutional code using shift registers.

(Refer Slide Time: 01: 22)

So we describe a convolutional code basically as an n, k convolutional code.

(Refer Slide Time: 01: 30)

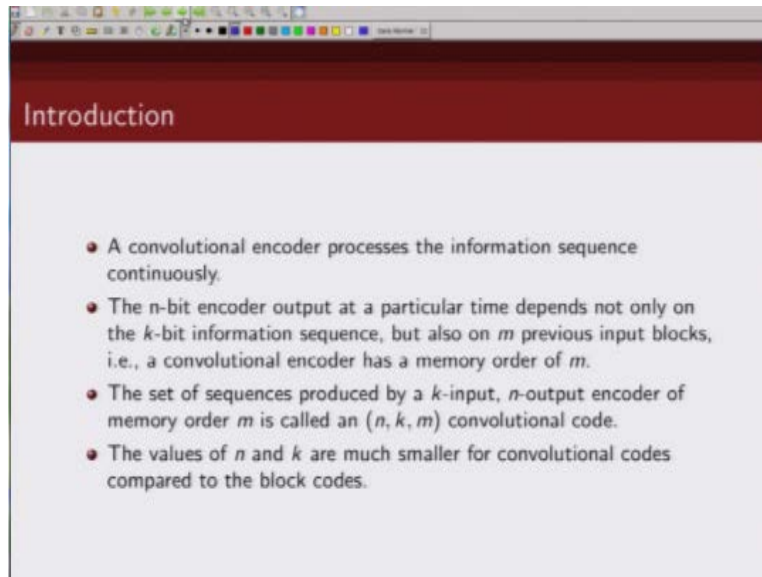


The image shows a presentation slide with a dark red header containing the word "Introduction". Below the header, there are three bullet points on a light gray background. The first bullet point states that a convolutional encoder processes the information sequence continuously. The second bullet point explains that the n -bit encoder output at a particular time depends not only on the k -bit information sequence, but also on m previous input blocks, meaning a convolutional encoder has a memory order of m . The third bullet point defines that the set of sequences produced by a k -input, n -output encoder of memory order m is called an (n, k, m) convolutional code.

- A convolutional encoder processes the information sequence continuously.
- The n -bit encoder output at a particular time depends not only on the k -bit information sequence, but also on m previous input blocks, i.e., a convolutional encoder has a memory order of m .
- The set of sequences produced by a k -input, n -output encoder of memory order m is called an (n, k, m) convolutional code.

With memory m .

(Refer Slide Time: 01: 33)

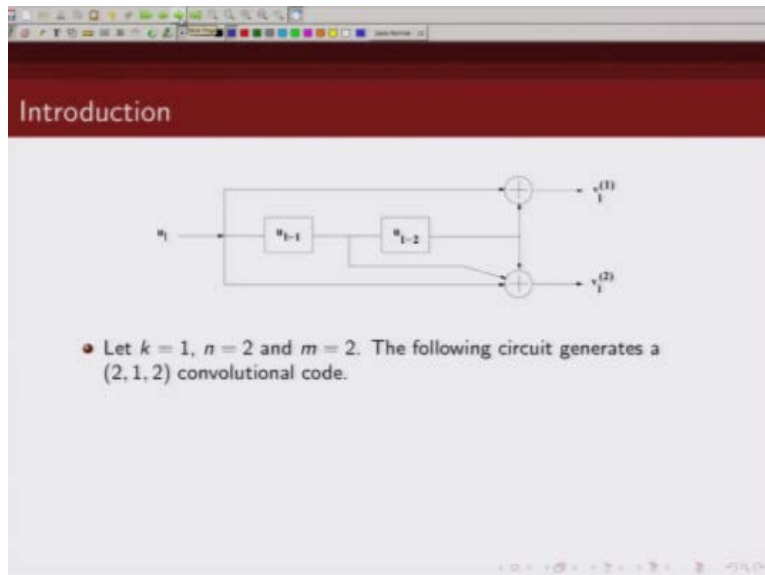


The image shows a presentation slide with a dark red header containing the word "Introduction" in white. Below the header, on a light gray background, are four bullet points, each starting with a red circular icon. The text of the slide is as follows:

- A convolutional encoder processes the information sequence continuously.
- The n -bit encoder output at a particular time depends not only on the k -bit information sequence, but also on m previous input blocks, i.e., a convolutional encoder has a memory order of m .
- The set of sequences produced by a k -input, n -output encoder of memory order m is called an (n, k, m) convolutional code.
- The values of n and k are much smaller for convolutional codes compared to the block codes.

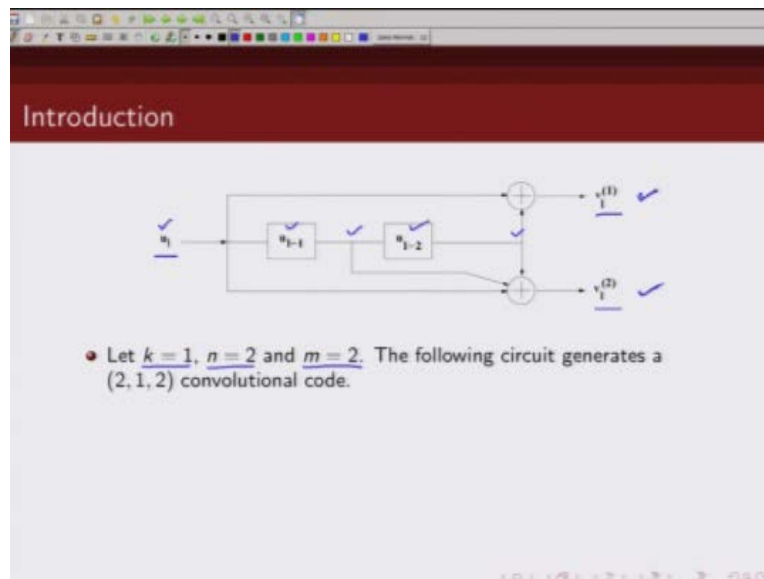
Now as we have said before our post to block codes typically the value of n and k for convolutional code is much smaller like k may be one, two, three and similarly n will be may be two, three, four like that.

(Refer Slide Time: 01: 51)



So this is one example of a memory to convolutional encoder

(Refer Slide Time: 01: 59)



We can see this is out input u_i an output $v^{(1)}_1$ and $v^{(2)}_1$ now note that the output $v^{(1)}_1$ and $v^{(2)}_1$ depends not only on the current input but also depends on the past inputs as indicated by content here and content here and this is $k=1$ because there is already one input here that is why k is 1 there are 2 output this one and this one , so that is why n is 2 and since the output depends on two memory elements this and this elements to so this a $2, 1, 1$ convolutional code.

(Refer Slide Time: 02: 53)

Introduction

• Let $k = 1$, $n = 2$ and $m = 2$. The following circuit generates a $(2, 1, 2)$ convolutional code.

• Input: u_t

As we said the input is u_1

(Refer Slide Time: 02: 57)

Introduction

• Let $k = 1$, $n = 2$ and $m = 2$. The following circuit generates a $(2, 1, 2)$ convolutional code.

• Input: u_l

• Outputs:

$$v_l^{(1)} = u_l + u_{l-2}$$
$$v_l^{(2)} = u_l + u_{l-1} + u_{l-2}$$

And the output is $v^{(1)}$ and $v^{(2)}$ and how is $v^{(1)}$ and $v^{(2)}$ depend on $u(l)$ and the pass values this is given by the interconnection.

(Refer Slide Time: 03: 09)

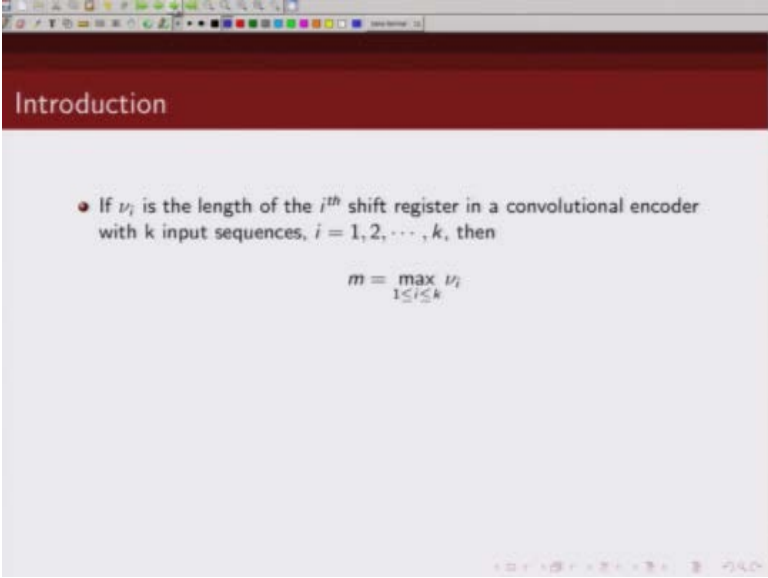
Introduction

- Let $k = 1$, $n = 2$ and $m = 2$. The following circuit generates a $(2, 1, 2)$ convolutional code.
- Input: u_i
- Outputs:

$$\begin{aligned} v_i^{(1)} &= u_i + u_{i-2} \quad \checkmark \\ v_i^{(2)} &= u_i + u_{i-1} + u_{i-2} \quad \checkmark \end{aligned}$$

So we can see for $v^{(1)}_1$ it depends on input u_1 as given by this and it depends on u_{1-2} as given by this link so $v^{(1)}_1$ is given by u_1 and u_{1-2} so it depends on the current input and the input which was there two time is this earlier similarly $v^{(1)}_{12}$ depends on u_1 as given by this interconnection u_{1-1} as given by this interconnection and u_{1-2} as given by this interconnection. So this is our $v^{(1)}_2$ so these are the two outputs and this is how they are related to the input so we can see that whether a particular input appears in the output that is basically given by these interconnections. These interconnections tell us whether that particular bit is taking part in the output or not.

(Refer Slide Time: 04: 17)



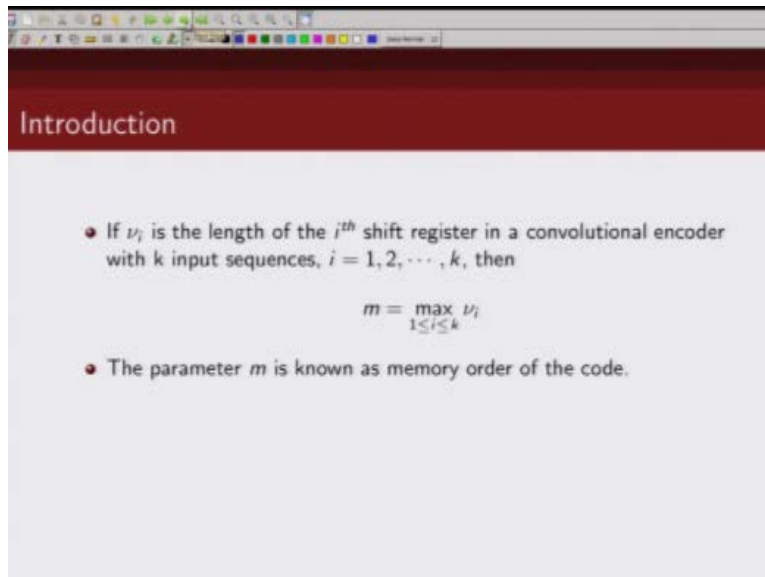
Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \dots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$

So if we denote by ν_i the length of the i^{th} shift register in a convolutional encoder then we defined a memory order as the maximum of maximum length of the shift register among the k shift registers use to represent the convolutional encoder.

(Refer Slide Time: 04: 39)



The slide is titled "Introduction" and contains the following text:

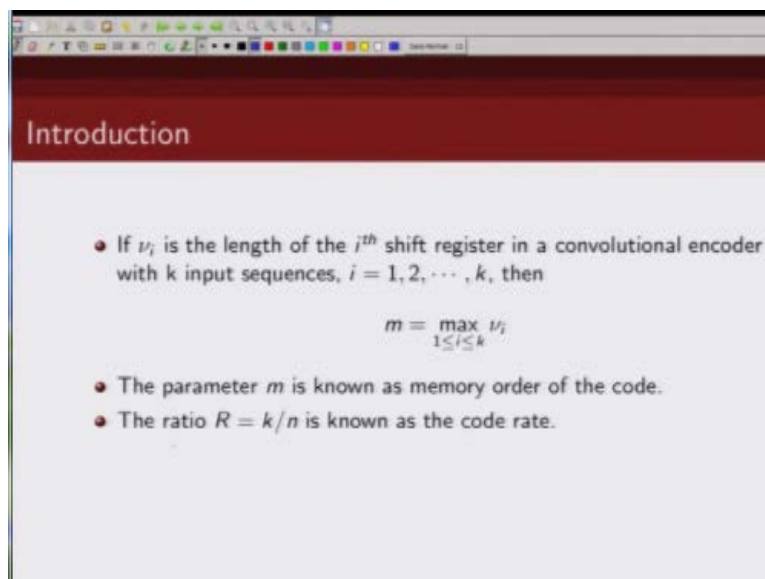
- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \dots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$

- The parameter m is known as memory order of the code.

And this parameter m is also known as memory order of the convolutional code.

(Refer Slide Time: 04: 50)



The slide is titled "Introduction" and contains the following text:

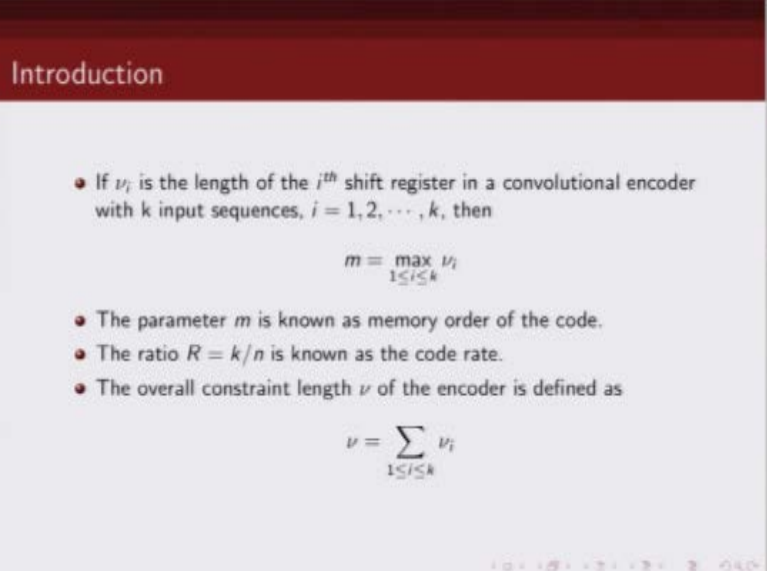
- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \dots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$

- The parameter m is known as memory order of the code.
- The ratio $R = k/n$ is known as the code rate.

As we know this ratio of information bits to coded bits k/n is known as code rate which we denote by R .

(Refer Slide Time: 05: 01)

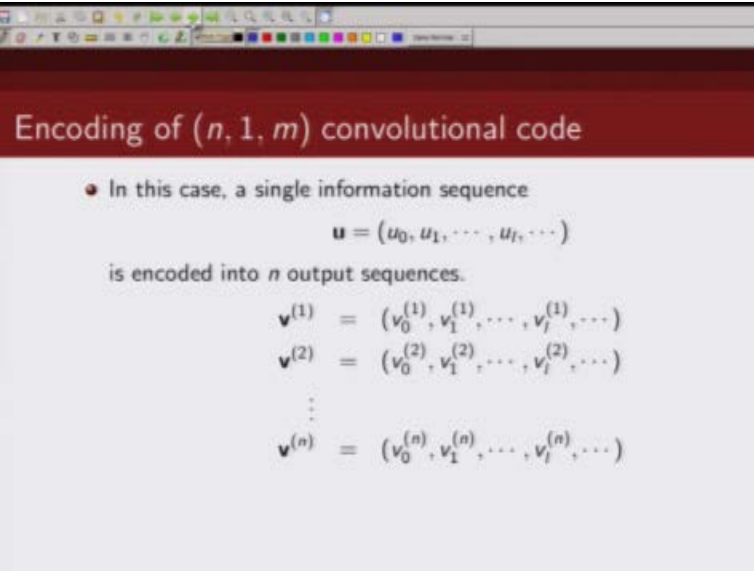


Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \dots, k$, then
$$m = \max_{1 \leq i \leq k} \nu_i$$
- The parameter m is known as memory order of the code.
- The ratio $R = k/n$ is known as the code rate.
- The overall constraint length ν of the encoder is defined as
$$\nu = \sum_{1 \leq i \leq k} \nu_i$$

And overall constrained length is defined as Σ of length of all the k shift register that is the overall constrained length.

(Refer Slide Time: 05: 16)

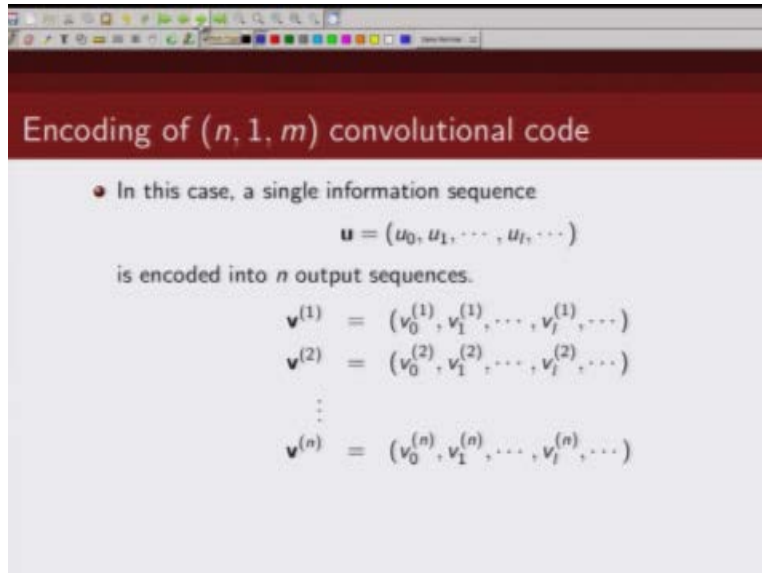


Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence
$$\mathbf{u} = (u_0, u_1, \dots, u_l, \dots)$$
is encoded into n output sequences.
$$\begin{aligned} \mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \dots, v_l^{(1)}, \dots) \\ \mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \dots, v_l^{(2)}, \dots) \\ &\vdots \\ \mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \dots, v_l^{(n)}, \dots) \end{aligned}$$

Now we are going to show how we can encode a convolutional an informational sequence using a rate $1/n$ convolutional encoder.

(Refer Slide Time: 05: 31)

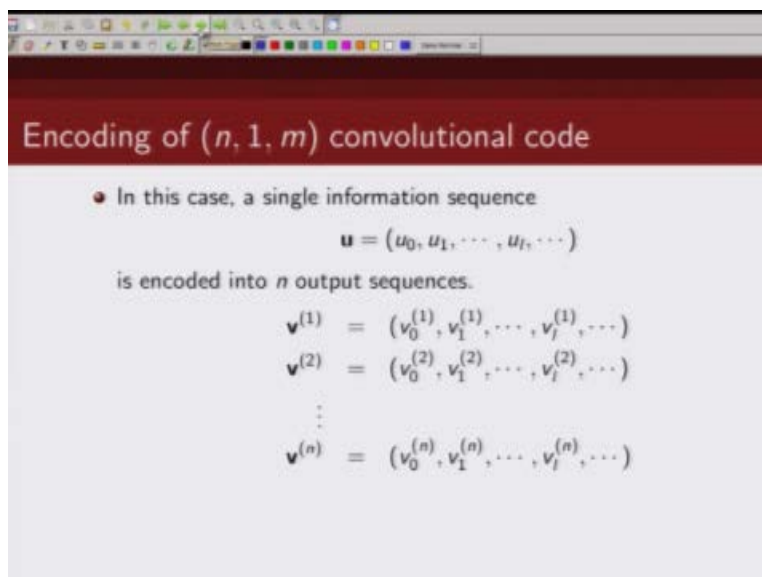


Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence
$$\mathbf{u} = (u_0, u_1, \dots, u_l, \dots)$$
is encoded into n output sequences.
$$\mathbf{v}^{(1)} = (v_0^{(1)}, v_1^{(1)}, \dots, v_l^{(1)}, \dots)$$
$$\mathbf{v}^{(2)} = (v_0^{(2)}, v_1^{(2)}, \dots, v_l^{(2)}, \dots)$$
$$\vdots$$
$$\mathbf{v}^{(n)} = (v_0^{(n)}, v_1^{(n)}, \dots, v_l^{(n)}, \dots)$$

So k is 1 n number of coded bits are send so there is one input coming in and there are n output and the maximum length of this one shift register used to represent this rate $1/n$ code is m so this shift register as m memory elements.

(Refer Slide Time: 05: 56)



Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence
$$\mathbf{u} = (u_0, u_1, \dots, u_l, \dots)$$
is encoded into n output sequences.
$$\mathbf{v}^{(1)} = (v_0^{(1)}, v_1^{(1)}, \dots, v_l^{(1)}, \dots)$$
$$\mathbf{v}^{(2)} = (v_0^{(2)}, v_1^{(2)}, \dots, v_l^{(2)}, \dots)$$
$$\vdots$$
$$\mathbf{v}^{(n)} = (v_0^{(n)}, v_1^{(n)}, \dots, v_l^{(n)}, \dots)$$

So let us take our input which we denote by

(Refer Slide Time: 06: 00)

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence $\mathbf{u} = (u_0, u_1, \dots, u_i, \dots)$ is encoded into n output sequences.

$$\begin{aligned} \mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \dots, v_i^{(1)}, \dots) \\ \mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \dots, v_i^{(2)}, \dots) \\ &\vdots \\ \mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \dots, v_i^{(n)}, \dots) \end{aligned}$$

\mathbf{u} to be $u_0, u_1, u_2, \dots, u_{i-1}$ since is a rate $1/n$ convolutional code so what we would get is corresponding to one input we are going to get n outputs and we denote these n outputs by $v^{(1)}, v^{(2)}, v^{(3)}, \dots, v^{(n)}$ where each of this $v^{(i)}$'s can be return like this

(Refer Slide Time: 06: 28)

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence $\mathbf{u} = (u_0, u_1, \dots, u_i, \dots)$ is encoded into n output sequences.

$$\begin{aligned} \mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \dots, v_i^{(1)}, \dots) \\ \mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \dots, v_i^{(2)}, \dots) \\ &\vdots \\ \mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \dots, v_i^{(n)}, \dots) \end{aligned}$$

- The n output sequences are interleaved to form a single code sequence.

$$\mathbf{v} = (\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_i, \dots), \text{ where } \mathbf{v}_i = (v_i^{(1)}, v_i^{(2)}, \dots, v_i^{(n)})$$

So the output at a particular incidence then is so corresponding to

(Refer Slide Time: 06: 34)

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence
 $\mathbf{u} = (u_0, u_1, \dots, u_l, \dots)$
is encoded into n output sequences.

$$\begin{aligned} \mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \dots, v_l^{(1)}, \dots) \\ \mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \dots, v_l^{(2)}, \dots) \\ &\vdots \\ \mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \dots, v_l^{(n)}, \dots) \end{aligned}$$

- The n output sequences are interleaved to form a single code sequence.

$$\mathbf{v} = (\underline{v_0}, \underline{v_1}, \dots, \underline{v_l}, \dots), \text{ where } \underline{v_l} = (v_l^{(1)}, v_l^{(2)}, \dots, v_l^{(n)})$$

U_0 then what is the output these are the n bits output corresponding to this input u_0 similarly corresponding to u_1 my output is this corresponding to u_1 my output is this n bit output okay so I write the output by $v^{(0)}, v^{(1)}, v^{(2)}, v^{(1)}$ where this $v^{(0)}$ is a n bit.

(Refer Slide Time: 07: 10)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$
$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$
$$\vdots$$
$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$

Now how we generate these n bit vector from this one input and if we just go

(Refer Slide Time: 07: 20)

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence $\mathbf{u} = (u_0, u_1, \dots, u_i, \dots)$ is encoded into n output sequences.

$$\mathbf{v}^{(1)} = (v_0^{(1)}, v_1^{(1)}, \dots, v_i^{(1)}, \dots)$$
$$\mathbf{v}^{(2)} = (v_0^{(2)}, v_1^{(2)}, \dots, v_i^{(2)}, \dots)$$
$$\vdots$$
$$\mathbf{v}^{(n)} = (v_0^{(n)}, v_1^{(n)}, \dots, v_i^{(n)}, \dots)$$

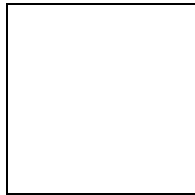
- The n output sequences are interleaved to form a single code sequence.

$$\mathbf{v} = (\underline{v_0}, \underline{v_1}, \dots, \underline{v_i}, \dots), \text{ where } \underline{v_i} = (v_i^{(1)}, v_i^{(2)}, \dots, v_i^{(n)})$$

Back to our example

(Refer Slide Time: 07: 20)

(Refer Slide Time: 07: 23)



That we had shown look at this example how did we, generate 2 coded bits corresponds to one information sequence. How did we generate these 2 coded bits these coded bits where generated by various combination of input and these past inputs and whether a particular bit appears in eth output that is governed by these interconnections. Whether there is a line connecting this part to the output or not that determines whether that particular bit it's participating in the output bit.

(Refer Slide Time: 08: 07)

Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \dots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$
- The parameter m is known as memory order of the code.
- The ratio $R = k/n$ is known as the code rate.
- The overall constraint length ν of the encoder is defined as

$$\nu = \sum_{1 \leq i \leq k} \nu_i$$

So what we can

(Refer Slide Time: 08: 10)

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence

$$\underline{\mathbf{u}} = (u_0, u_1, \dots, u_l, \dots)$$
 is encoded into n output sequences.

$$\underline{\mathbf{v}}^{(1)} = (v_0^{(1)}, v_1^{(1)}, \dots, v_l^{(1)}, \dots)$$

$$\underline{\mathbf{v}}^{(2)} = (v_0^{(2)}, v_1^{(2)}, \dots, v_l^{(2)}, \dots)$$

$$\vdots$$

$$\underline{\mathbf{v}}^{(n)} = (v_0^{(n)}, v_1^{(n)}, \dots, v_l^{(n)}, \dots)$$

Conclude form here is.

(Refer Slide Time: 08: 11)

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence

$$\mathbf{u} = (u_0, u_1, \dots, u_i, \dots)$$

is encoded into n output sequences,

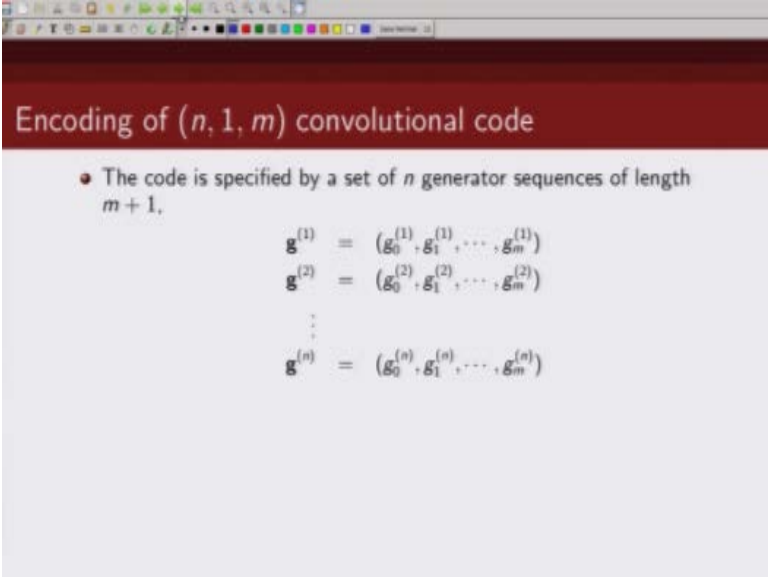
$$\begin{aligned} \mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \dots, v_i^{(1)}, \dots) \\ \mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \dots, v_i^{(2)}, \dots) \\ &\vdots \\ \mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \dots, v_i^{(n)}, \dots) \end{aligned}$$

- The n output sequences are interleaved to form a single code sequence.

$$\mathbf{v} = (\underline{v}_0, \underline{v}_1, \dots, \underline{v}_i, \dots), \text{ where } \underline{v}_i = (v_i^{(1)}, v_i^{(2)}, \dots, v_i^{(n)})$$

Basically.

(Refer Slide Time: 08: 12)



Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$
$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$
$$\vdots$$
$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$

We can completely specify a code by this set of n generator sequence of length $m+1$ where each of these generator sequences is basically of length $m+1$ and what are these $g^{(0)}_1, g^{(1)}_1, g^{(1)}_2, g_m$ so you can see so this super script that you see one, two, three and n this corresponds to each of the output sequence so the first output sequence is specified by this generator sequence g_1 the second output sequence is specified by this generator sequence $g^{(2)}$ and the n yet output sequence is specified by this output sequence $g^{(n)}$ and what are these $g^{(i)}$'s now note that the memory order of our convolutional encoder is m so there are so if let's say.

(Refer Slide Time: 09: 22)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$
$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$

⋮

$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$

Just take an example

(Refer Slide Time: 09: 24)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$.

$$g^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$

$$g^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$

$$\vdots$$

$$g^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$

$m=2$ so if we take $m=2$ let's say two memory order so then basically and this say this my input $u(1)$ and my output I can take from some interconnections form this let say this is my example that I had this was my $v^{(1)}1$ this was my $v^{(1)}2$ now note that these interconnections are specifying whether a particular bit is participating in the output code sequence or not so if we look at the first coded bit u_1 now this as memory order m so they are possible $m+1$ connection what are those possible $m+1$ connection 1 first one is corresponding to whether u_1 is participating in the output bit or not.

Second one corresponding to whether u_{1-1} is participating or not is this point. Three one is this point whether u_{1-2} is participating or not similarly the second coded sequence whether u_1 is participating or not whether u_{1-1} participating or not whether u_{1-2} is participating or not so we can see that the output here let us take the first output sequence that is completely specified by whether u_1 is participating were u_{1-1} is participating u_{1-2} is participating.

So in this example $g^{(0)}1, g^{(1)}1, g^{(2)}m$ completely specifies what inputs are participating in generating our code sequence similarly look at the second bit her also these $m+1$ connections will completely specify whether a particular bit or the pass bit are taking part in the output coded bit. So you can see if you have rate $1/n$ code whose memory is m then we can completely specify that code using a set of n generator sequence where each of this n generator sequence correspond to one of the output sequences and

(Refer Slide Time: 11: 55)

The slide is titled "Encoding of $(n, 1, m)$ convolutional code". It contains the following text and diagram:

- The code is specified by a set of n generator sequences of length $m + 1$.
- $g^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$
- $g^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$
- \vdots
- $g^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$

To the right of the equations is a block diagram of a convolutional encoder. An input u_k is fed into n parallel branches. Each branch contains a shift register and a multiplier. The outputs of these branches are summed to produce the encoded output v_k .

Each of the generator sequence is of length $m+1$ specifying the interconnections of u_1, u_{1-1}, u_{1-2} up to u_{1-m} so then what are these $g^{(0)}_1$ and $g^{(1)}_1$ if $g^{(0)}_1$ and $g^{(1)}_1$ are either one or zero. One means they are participating zero means it does not participate for example in this example

(Refer Slide Time: 12: 24)'

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$.

$$\begin{aligned} g^{(1)} &= (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)}) \\ g^{(2)} &= (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)}) \\ &\vdots \\ g^{(n)} &= (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)}) \end{aligned}$$

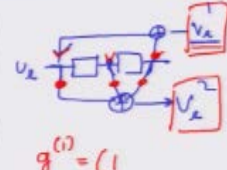
$g^{(1)} =$

What is g_1 is u_1 participating in the output sequence of $v^{(1)}$ yes it is so then

(Refer Slide Time: 12: 36)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m+1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$
$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$
$$\vdots$$
$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$


$g^{(1)} = (1)$

$g^{(0)}$ will be one is u_{i-1} participating in the output sequence $v^{(1)}$ no so then this will be.

(Refer Slide Time: 12: 43)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$
$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$
$$\vdots$$
$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$

u_k

$g^{(1)} = (10)$

So then this will zero what about u_{k-2} it is participating the output sequence so it will be

(Refer Slide Time: 12: 50)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$g^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$
$$g^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$
$$\vdots$$
$$g^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$

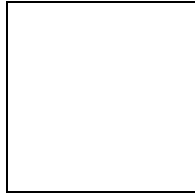
$g^{(1)} = (101)$

One so $g^{(1)}$ is 101 similarly.

(Refer Slide Time: 12: 55)

$g^{(2)}$ will be 111 because u_1 , u_{1-1} and u_{1-2} they are all participating in the output coded sequence
okay so if I specify these generator sequence then mu convolutional code is completely specified

(Refer Slide Time: 13: 18)



And what is my output then my output is nothing but

(Refer Slide Time: 13: 23)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$

$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$

$$\vdots$$

$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$
- The output sequence is the discrete convolution of the information sequence \mathbf{u} and the generator sequence $\mathbf{g}^{(i)}$, i.e.

$$\mathbf{v}^{(i)} = \mathbf{u} * \mathbf{g}^{(i)}, \quad 1 \leq i \leq n,$$
 and

$$\mathbf{v}_i^{(i)} = u_i g_0^{(i)} + u_{i-1} g_1^{(i)} + \dots + u_{i-m} g_m^{(i)}$$

$$= \sum_{j=0}^m u_{i-j} g_j^{(i)}$$

Is a discrete convolution of the information sequence with this generator sequence? So if my generator sequence if my code as memory m then basically I can write this discrete convolution in this particular position and that is basically my output sequence which is discrete convolution of the input sequence with this generator sequence. Now let us take an example

(Refer Slide Time: 13: 59)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$

$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$

$$\vdots$$

$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$
- The output sequence is the discrete convolution of the information sequence \mathbf{u} and the generator sequence $\mathbf{g}^{(i)}$, i.e.

$$\mathbf{v}^{(i)} = \mathbf{u} * \mathbf{g}^{(i)}, \quad 1 \leq i \leq n,$$
 and

$$\mathbf{v}_i^{(i)} = u_i g_0^{(i)} + u_{i-1} g_1^{(i)} + \dots + u_{i-m} g_m^{(i)}$$

$$= \sum_{j=0}^m u_{i-j} g_j^{(i)}$$

This is the same example that we are considering this rate one of code with

(Refer Slide Time: 14: 07)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1 \ 0 \ 1),$$

$$g^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes in red:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

Memory two so you can see $v_l^{(1)}$ this is basically again discrete convolution of input with these generator sequence which we can write as $u_l + u_{l-2}$ and this $v_l^{(2)}$ can be written as $u_l + u_{l-1} + u_{l-2}$ if you go back

(Refer Slide Time: 14: 36)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$

$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$

$$\vdots$$

$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$
- The output sequence is the discrete convolution of the information sequence \mathbf{u} and the generator sequence $\mathbf{g}^{(i)}$, i.e.

$$\mathbf{v}^{(i)} = \mathbf{u} * \mathbf{g}^{(i)}, \quad 1 \leq i \leq n,$$
 and

$$\mathbf{v}_i^{(i)} = u_i g_0^{(i)} + u_{i-1} g_1^{(i)} + \dots + u_{i-m} g_m^{(i)}$$

$$= \sum_{j=0}^m u_{i-j} g_j^{(i)}$$

Our output is this if you can

(Refer Slide Time: 14: 36)

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m + 1$,

$$\mathbf{g}^{(1)} = (g_0^{(1)}, g_1^{(1)}, \dots, g_m^{(1)})$$

$$\mathbf{g}^{(2)} = (g_0^{(2)}, g_1^{(2)}, \dots, g_m^{(2)})$$

$$\vdots$$

$$\mathbf{g}^{(n)} = (g_0^{(n)}, g_1^{(n)}, \dots, g_m^{(n)})$$
- The output sequence is the discrete convolution of the information sequence \mathbf{u} and the generator sequence $\mathbf{g}^{(i)}$, i.e.

$$\mathbf{v}^{(i)} = \mathbf{u} * \mathbf{g}^{(i)}, \quad 1 \leq i \leq n,$$
 and

$$\mathbf{v}_i^{(i)} = u_i g_0^{(i)} + u_{i-1} g_1^{(i)} + \dots + u_{i-m} g_m^{(i)}$$

$$= \sum_{l=0}^m u_{i-l} g_l^{(i)} = u_i g_0^{(i)} + u_{i-1} g_1^{(i)} + u_{i-2} g_2^{(i)}$$

Expand it for this particular example this will be $u_l g_0^{(i)} + u_{l-1} g_1^{(i)} + u_{l-2} g_2^{(i)}$ and for the first coded sequence this g is $g^{(0)}, g^{(1)}, g^{(2)}$ was 101 and the second sequence was 111 that is why the first.

(Refer Slide Time: 15: 24)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1\ 0\ 1),$$

$$g^{(2)} = (1\ 1\ 1),$$

Handwritten notes in red:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

First coded sequence is $u_l + u_{l-2}$ and the second coded sequence is $u_l + u_{l-1} + u_{l-2}$.

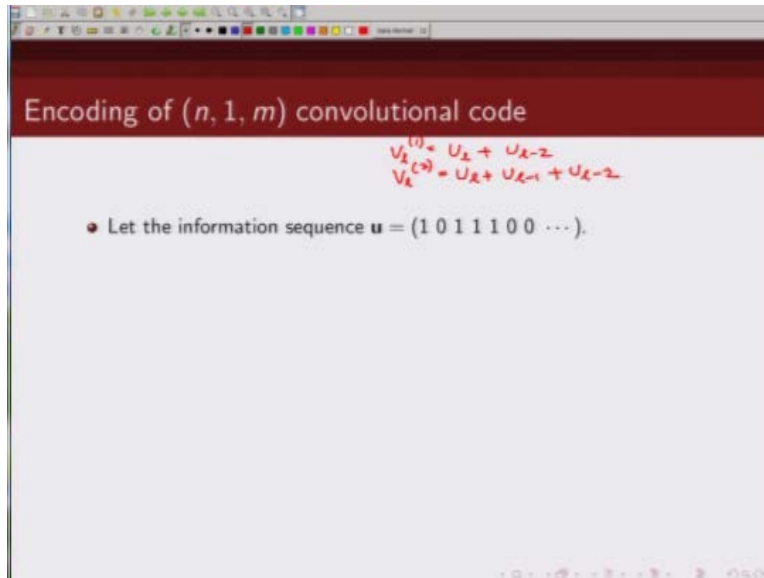
(Refer Slide Time: 15: 38)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

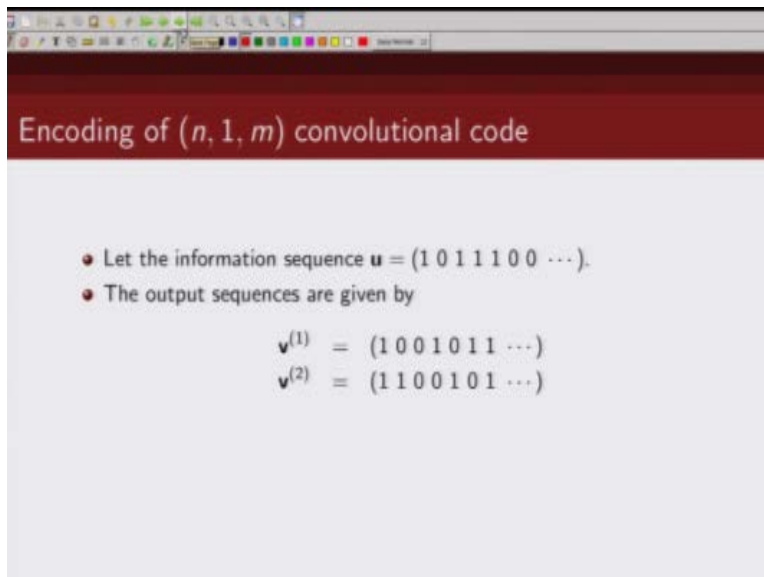
So if we have information sequence this, what was our output sequence

(Refer Slide Time: 15: 43)



We had $v_1^{(1)}$ is $u_1 + u_{1-2}$ and $v_2^{(1)}$ is $u_1 + u_{1-1} + u_{1-2}$

(Refer Slide Time: 16: 03)



Now we can show that our output coded sequence will be given by this now this can easily verified so let us says what was our output

(Refer Slide Time: 16: 15)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

Coded sequence $v^{(1)}$ was $u_l + u_{l-2}$ and $v^{(2)}$ is $u_l + u_{l-1} + u_{l-2}$ now note when the first input u_l which is one comes what is the output now to specify the output we need to specify what the initial contents of u_{l-1} and u_{l-2} so initially we will assume that the convolutional encoder was in all zero state now what do we mean by all zero so we are assuming that initially the contents of the shift registers were all zero in other words u_{l-2} and u_{l-1} they were both zero okay.

If both were zero initially and if $u^{(1)}$ is 1 what will be

(Refer Slide Time: 17: 15)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (\underline{1} \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

V11 this is 1+0 which is 1 so you can see.

(Refer Slide Time: 17: 21)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (\underline{1} 0 1 1 1 0 0 \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (\underline{1} 0 0 1 0 1 1 \dots)$$

$$\mathbf{v}^{(2)} = (\underline{1} 1 0 0 1 0 1 \dots)$$

This is one and what is $v_1^{(2)}$ is $1+0+0$ so that is also 1 next what happens next if you go back

(Refer Slide Time: 17: 36)

Encoding of $(n, 1, m)$ convolutional code

$$V_k^{(1)} = u_k + u_{k-2}$$

$$V_k^{(2)} = u_k + u_{k-1} + u_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

(Refer Slide Time: 17: 37)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1\ 0\ 1),$$

$$g^{(2)} = (1\ 1\ 1),$$

Handwritten equations:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

This one which was here when you apply a clock this one moves here and a new bit comes here so now in the next time instance u_{l-1} becomes 1 and what is u_{l-2} since u_{l-1} initially was zero so this zero will come here so the new contents of the shift register will be now one and zero.

(Refer Slide Time: 18: 03)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (\underline{1} \underline{0} \underline{1} \underline{1} \underline{1} \underline{0} \underline{0} \dots)$.
- The output sequences are given by

$$v^{(1)} = (\underline{1} \underline{0} \underline{0} \underline{1} \underline{0} \underline{1} \underline{1} \dots)$$

$$v^{(2)} = (\underline{1} \underline{1} \underline{0} \underline{0} \underline{1} \underline{0} \underline{1} \dots)$$

So what we have is now u_{l-1} is 1 and u_{l-2} is zero. Now the next input is zero so next input is zero so what is are next output this is zero and u_{l-2} is zero so this will be zero you can see this is zero what about this now $u^{(l)}$ is zero u_{l-1} is one and u_{l-2} is zero so $0+1 +0$ that will be one and that is I given by this okay next what happens.

(Refer Slide Time: 18: 44)

Encoding of $(n, 1, m)$ convolutional code

$$V_k^{(1)} = U_k + U_{k-2}$$
$$V_k^{(2)} = U_k + U_{k-1} + U_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

(Refer Slide Time: 18: 45)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1 \ 0 \ 1),$$

$$g^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes in red:

$$V_l^{(1)} = u_l + u_{l-2}$$

$$V_l^{(2)} = u_l + u_{l-1} + u_{l-2}$$

Again go back to this diagram you are input zero here so now this zero will move here and you had a one here so this is one will move here so the new contents which shift register will be zero and one okay.

(Refer Slide Time: 19: 02)

Encoding of $(n, 1, m)$ convolutional code

$$V_k^{(1)} = U_k + U_{k-2}$$
$$V_k^{(2)} = U_k + U_{k-1} + U_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

If that happens

(Refer Slide Time: 19: 02)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (\underline{1} \underline{0} 1 1 1 0 0 \dots)$.
- The output sequences are given by

$$v^{(1)} = (\underline{1} \underline{0} 0 1 0 1 1 \dots)$$

$$v^{(2)} = (\underline{1} \underline{1} 0 0 1 0 1 \dots)$$

Then next input is one so if this is one.

(Refer Slide Time: 19: 07)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

So if this one what is u_{l-2} u_{l-2} was one so $1+1$ that is zero and $u^{(1)}$ is one u_{l-2} is 1 and u_{l-1} is zero so $1+0+1$ that is zero. So like that you can basically write down the output coded sequence so then what is my final output so corresponding to this one and what is my coded sequence that is given by this corresponding to this zero my coded sequence is given by this okay.

(Refer Slide Time: 19: 45)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by
$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$
$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$
- The code sequence can be written as
$$\mathbf{v} = (11, 01, 00, 10, 01, 10, 11, \dots)$$

So then I can write my final output as so corresponding to input one I get

(Refer Slide Time: 20: 01)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$
- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

11 that is give by this corresponding to zero I get 01 that I give this corresponding I get 00 as given by this so this is how I can write my output coded sequence

(Refer Slide Time: 20: 17)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m & & \\ & g_0 & g_1 & \dots & \dots & g_m & \\ & & & \ddots & & & \ddots \\ & & & & & & & \ddots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)} \ g_i^{(1)} \ \dots \ g_i^{(n)}), \ 0 \leq i \leq m$$

Now the same thing I can write in the matrix form so.

(Refer Slide Time: 20: 24)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m \\ & g_0 & g_1 & \dots & \dots & g_m \\ & & \dots & \dots & \dots & \dots \\ & & & & & \dots \end{bmatrix}$$

where

$$g_i = (g_i^{(1)} \ g_i^{(2)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

$\mathbf{u} = [u_0 \ u_1 \ u_2 \ \dots \]$

I define this generator matrix G which generate this codes code word so the output code word can be written as input times this generator matrix G okay and this generator matrix is of the form like this so let us just expand it and may be try to explain why the generator form as this semi infinite kind of form for a convolutional code so let us say u is $u^{(0)}, u^{(1)}, u^{(2)} \dots$ Is continuing set of sequence like this right now what is your, output sequence. Output sequence so initially what happens if you go back to this.

(Refer Slide Time: 21: 18)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\begin{aligned} \mathbf{v}^{(1)} &= (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots) \\ \mathbf{v}^{(2)} &= (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots) \end{aligned}$$
- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

Diagram.

(Refer Slide Time: 21: 20)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (\underline{1}\ \underline{0}\ \underline{1}\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\begin{aligned} \mathbf{v}^{(1)} &= (\boxed{1}\ \boxed{0}\ 0\ 1\ 0\ 1\ 1\ \dots) \\ \mathbf{v}^{(2)} &= (\boxed{1}\ \boxed{1}\ 0\ 0\ 1\ 0\ 1\ \dots) \end{aligned}$$

(Refer Slide Time: 21: 20)

Encoding of $(n, 1, m)$ convolutional code

$$V_i^{(1)} = U_i + U_{i-2}$$

$$V_i^{(2)} = U_i + U_{i-1} + U_{i-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

(Refer Slide Time: 21: 20)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\mathbf{g}^{(1)} = (1\ 0\ 1),$$

$$\mathbf{g}^{(2)} = (1\ 1\ 1),$$

$$V_i^{(1)} = U_i + U_{i-2}$$

$$V_i^{(2)} = U_i + U_{i-1} + U_{i-2}$$

Initially you are assuming that the encoder is an all zero state correct so what will be the first output that you will get here that is $U^{(0)}$ times $g^{(0)}$ what is $g^{(0)}$.

(Refer Slide Time: 21: 45)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\mathbf{g}^{(1)} = (1\ 0\ 1),$$

$$\mathbf{g}^{(2)} = (1\ 1\ 1).$$

Handwritten notes:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

g_1 this g_2 is this. This interconnection which is connecting $u^{(i)}$ to the output so at first time instance.

(Refer Slide Time: 21: 57)

Encoding of $(n, 1, m)$ convolutional code

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

What you would get is

(Refer Slide Time: 21: 59)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\begin{aligned} \mathbf{v}^{(1)} &= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots) \\ \mathbf{v}^{(2)} &= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots) \end{aligned}$$

$$\begin{aligned} v_x^{(1)} &= u_x + u_{x-2} \\ v_x^{(2)} &= u_x + u_{x-1} + u_{x-2} \end{aligned}$$

(Refer Slide Time: 22: 00)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by
$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$
$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$
- The code sequence can be written as
$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

The output is nothing but.

(Refer Slide Time: 22: 02)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$u = [u_0 \ u_1 \ u_2 \ \dots]$

$$\underline{v} = \underline{uG}$$

$$G = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m \\ & g_0 & g_1 & \dots & \dots & g_m \\ & & \ddots & & & \ddots \end{bmatrix}$$

where

$$g_i = (g_i^{(1)} \ g_i^{(1)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

$U^{(0)}$ times.

(Refer Slide Time: 22: 04)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{u}\mathbf{G}$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \dots & \dots & \mathbf{g}_m \\ & \mathbf{g}_0 & \mathbf{g}_1 & \dots & \mathbf{g}_m \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & \ddots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)} \ g_i^{(2)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

Handwritten notes in red:

- $\mathbf{v} = [v_0 \ v_1 \ v_2 \ \dots \]$
- $\underline{v_0 \ g_0}$

$g^{(0)}$ this is the output that you will get at first time instance what is a output that you will get in the second time

(Refer Slide Time: 22: 10)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$

- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

Encoding of $(n, 1, m)$ convolutional code

$$v_x^{(1)} = u_x + u_{x-2}$$
$$v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$$

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$
$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

(Refer Slide Time: 22: 13)

Encoding of $(n, 1, m)$ convolutional code

$$V_1^{(1)} = U_x + U_{x-2}$$

$$V_2^{(2)} = U_x + U_{x-1} + U_{x-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

Now.

(Refer Slide Time: 22: 14)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\mathbf{g}^{(1)} = (1\ 0\ 1),$$

$$\mathbf{g}^{(2)} = (1\ 1\ 1),$$

$$V_1^{(1)} = U_x + U_{x-2}$$

$$V_2^{(2)} = U_x + U_{x-1} + U_{x-2}$$

Whatever u_0 you had now that u_0 has moved here correct and a new bit which is u_1 have come here u_1 so what is a output at this time it is u_1 times g_0 + u_0 times g_1 so I can write.

(Refer Slide Time: 22: 35)

Encoding of $(n, 1, m)$ convolutional code

$$V_k^{(1)} = U_k + U_{k-2}$$

$$V_k^{(2)} = U_k + U_{k-1} + U_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

(Refer Slide Time: 22: 33)

Encoding of $(n, 1, m)$ convolutional code

$$V_k^{(1)} = U_k + U_{k-2}$$

$$V_k^{(2)} = U_k + U_{k-1} + U_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$

And the second time.

(Refer Slide Time: 22: 37)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$
- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

Instance.

(Refer Slide Time: 22: 38)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m \\ & g_0 & g_1 & \dots & \dots & g_m \\ & & \dots & \dots & \dots & \dots \\ & & & \dots & \dots & \dots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)}\ g_i^{(2)}\ \dots\ g_i^{(n)}),\ 0 \leq i \leq m$$

$\mathbf{u} = [u_0\ u_1\ u_2\ \dots]$

$u_0 g_0$

My output is give by

(Refer Slide Time: 22: 40)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\underline{v} = \underline{uG}$$

$$G = \begin{bmatrix} g_0 & g_1 & \cdots & \cdots & g_m \\ & g_0 & g_1 & \cdots & g_m \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & \ddots \end{bmatrix}$$

where

$$g_i = (g_i^{(1)} \ g_i^{(2)} \ \cdots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

$u = [u_0 \ u_1 \ u_2 \ \dots]$

$u_0 g_0$
 $u_1 g_0 + u_0 g_1$

u_1 times $g^{(0)}$ + u_0 times $g^{(1)}$ fine next time instance what is my output

(Refer Slide Time: 23: 00)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\underline{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$v^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$v^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

(Refer Slide Time: 23: 01)

Encoding of $(n, 1, m)$ convolutional code

$V_1^{(1)} = U_2 + U_{2-2}$
 $V_2^{(2)} = U_2 + U_{2-1} + U_{2-2}$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

(Refer Slide Time: 23: 01)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$\mathbf{g}^{(1)} = (1\ 0\ 1)$
 $\mathbf{g}^{(2)} = (1\ 1\ 1)$

$V_1^{(1)} = U_2 + U_{2-2}$
 $V_2^{(2)} = U_2 + U_{2-1} + U_{2-2}$

Now what is going to happen is this u_1 will move here so this will be now u_2 this will become u_1 and this will become u_0 so what is my output now it is u_2 times $g^{(1)} + u_1$ times $g^{(1)} + u_0$ times $g^{(2)}$.

(Refer Slide Time: 23: 42)

Encoding of $(n, 1, m)$ convolutional code

$$v_k^{(1)} = u_k + u_{k-2}$$

$$v_k^{(2)} = u_k + u_{k-1} + u_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.

(Refer Slide Time: 23: 43)

Encoding of $(n, 1, m)$ convolutional code

$$v_k^{(1)} = u_k + u_{k-2}$$

$$v_k^{(2)} = u_k + u_{k-1} + u_{k-2}$$

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$

(Refer Slide Time: 23: 43)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by

$$\begin{aligned} \mathbf{v}^{(1)} &= (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots) \\ \mathbf{v}^{(2)} &= (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots) \end{aligned}$$
- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

(Refer Slide Time: 23: 44)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m & & \\ & g_0 & g_1 & \dots & \dots & g_m & \\ & & \dots & \dots & \dots & \dots & \dots \\ & & & & & & \dots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)}\ g_i^{(2)}\ \dots\ g_i^{(n)}),\ 0 \leq i \leq m$$

$\mathbf{u} = [u_0\ u_1\ u_2\ \dots]$
 $u_0 g_0$
 $u_1 g_0 + u_0 g_1$

So go back to so what would be my output here it is

(Refer Slide Time: 23: 48)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m \\ & g_0 & g_1 & \dots & g_m \\ & & \dots & \dots & \dots \\ & & & \dots & \dots \end{bmatrix}$$

where

$$g_i = (g_i^{(1)} \ g_i^{(2)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

Handwritten notes:

$$u = [u_0 \ u_1 \ u_2 \ \dots]$$

$$u_0 g_0$$

$$u_1 g_0 + u_0 g_1$$

$$u_2 g_0 + u_1 g_1 + u_0 g_2$$

u_2 times $g^{(0)} + u_1 g^{(1)} + u_0$ times $g^{(2)}$ what happens next

(Refer Slide Time: 24: 02)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$
- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

(Refer Slide Time: 24: 03)

Encoding of $(n, 1, m)$ convolutional code

$v_k^{(1)} = u_k + u_{k-2}$
 $v_k^{(2)} = u_k + u_{k-1} + u_{k-2}$

↓
 • Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
 • The output sequences are given by

$v^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$
 $v^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$

(Refer Slide Time: 24: 03)

Encoding of $(n, 1, m)$ convolutional code

$v_k^{(1)} = u_k + u_{k-2}$
 $v_k^{(2)} = u_k + u_{k-1} + u_{k-2}$

• Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.

(Refer Slide Time: 24: 03)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\mathbf{g}^{(1)} = (1 \ 0 \ 1),$$

$$\mathbf{g}^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes in red:

$$V_i^{(1)} = U_i + U_{i-2}$$

$$V_i^{(2)} = U_i + U_{i-1} + U_{i-2}$$

This u_0 moves out

(Refer Slide Time: 24: 10)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\mathbf{g}^{(1)} = (1 \ 0 \ 1),$$

$$\mathbf{g}^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes in red:

$$V_i^{(1)} = U_i + U_{i-2}$$

$$V_i^{(2)} = U_i + U_{i-1} + U_{i-2}$$

Here what we will get is u_1

(Refer Slide Time: 24: 18)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2, (2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1 \ 0 \ 1),$$

$$g^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

This will be u_1 what about this. This will become u_2

(Refer Slide Time: 24: 26)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2, (2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1 \ 0 \ 1),$$

$$g^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

So this u_2 and this will become u_3 .

(Refer Slide Time: 24: 26)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\mathbf{g}^{(1)} = (1 \ 0 \ 1),$$

$$\mathbf{g}^{(2)} = (1 \ 1 \ 1),$$

Handwritten notes:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

So this is u_3 so what will be the output now it is u_3 times $g^{(0)}$ + u_2 times $g^{(1)}$ + u_1 times $g^{(2)}$ and u_0 does not appear because of memory order of this course was 2 so what is the output in this case.

(Refer Slide Time: 24: 51)

Encoding of $(n, 1, m)$ convolutional code

Handwritten notes:

$$V_l^{(1)} = U_l + U_{l-2}$$

$$V_l^{(2)} = U_l + U_{l-1} + U_{l-2}$$

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.

(Refer Slide Time: 24: 51)

Encoding of $(n, 1, m)$ convolutional code

$v_x^{(1)} = u_x + u_{x-2}$
 $v_x^{(2)} = u_x + u_{x-1} + u_{x-2}$

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

(Refer Slide Time: 24: 52)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$.
- The output sequences are given by

$$\mathbf{v}^{(1)} = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ \dots)$$

$$\mathbf{v}^{(2)} = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \dots)$$

- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

(Refer Slide Time: 24: 52)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m \\ & g_0 & g_1 & \dots & \dots & g_m \\ & & \ddots & \ddots & \ddots & \ddots \\ & & & & & \ddots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)} \ g_i^{(1)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

Handwritten notes:

$$\mathbf{u} = [u_0 \ u_1 \ u_2 \ \dots \]$$

$$u_0 g_0$$

$$u_1 g_0 + u_0 g_1$$

$$u_2 g_0 + u_1 g_1 + u_0 g_2$$

Was so what is the output in this case thirds instance this will be

(Refer Slide Time: 24: 56)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{uG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & \dots & g_m \\ & g_0 & g_1 & \dots & \dots & g_m \\ & & \ddots & \ddots & \ddots & \ddots \\ & & & & & \ddots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)} \ g_i^{(1)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

Handwritten notes:

$$\mathbf{u} = [u_0 \ u_1 \ u_2 \ \dots \]$$

$$u_0 g_0$$

$$u_1 g_0 + u_0 g_1$$

$$u_2 g_0 + u_1 g_1 + u_0 g_2$$

$$u_3 g_0 + u_2 g_1 + u_1 g_2$$

u_3 times $g^{(0)} + u_2$ times $g^{(1)} + u_1$ times $g^{(2)}$ now if you write this same thing in a matrix form so what is \mathbf{v} is basically

(Refer Slide Time: 24: 15)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

• Matrix form $[v_0 v_1 v_2 \dots] = U = [u_0 u_1 u_2 \dots] G$

$v = uG$

$$G = \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ g_0 & g_1 & g_2 & \dots & \dots & g_m \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{bmatrix}$$

where $g_i = (g_i^{(1)} g_i^{(2)} \dots g_i^{(n)}), 0 \leq i \leq m$

$u_0 g_0$
 $u_1 g_0 + u_0 g_1$
 $u_2 g_0 + u_1 g_1 + u_0 g_2$
 $u_3 g_0 + u_2 g_1 + u_1 g_2$

$G = \begin{bmatrix} g_0 & g_1 & g_2 & \dots & g_m \\ & g_0 & g_1 & g_2 & \dots & g_m \\ & & g_0 & g_1 & g_2 & \dots & g_m \\ & & & g_0 & g_1 & g_2 & \dots & g_m \\ & & & & g_0 & g_1 & g_2 & \dots & g_m \end{bmatrix}$

v at time zero time 1 time 2 if you write this in this particular form is equal to u times this matrix G now you compare this equation with this equation so at first times is this that output is $u_0 g^{(0)}$ so that is what so this is u_0 times $g^{(0)}$ so this is $g^{(0)}$ second is this what is my output my output is u_0 times $g^{(1)}$ this term and then $g^{(0)}$ times u_1 which is this term so the second entry of this generator matrix is this okay now what is this third entry here you can see u_0 times $g^{(2)}$ so u_0 times this is $g_2 + u_1$ times g_1 that is this so this is g_1 and then this u_2 times g_0 so you can see then feather if you look at this what we get here is so u_0 times 0 will get here and the uv will get u_1 times $g^{(2)}$ u_3 times so this will be like zero $g^{(0)}$, $g^{(1)}$ and $g^{(2)}$.so say in this case a memory order was m that I why we are getting like this so you can see here our generator matrix is of the form of semi infinite form we are basically our G is something like this.

so you have $g^{(0)}$, to g_m now this becomes zero now this is new $g^{(0)}$, and this g_m and then this is 00 this is $g^{(0)}$, so it is like in this way diagonally my generator sequence is moving and that is what I have written here so if try to write it in the form of generator matrix then I can my generator matrix in this case is a semi infinite from and through this.

(Refer Slide Time: 27: 23)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $[v_0 v_1 v_2 \dots] = \mathbf{v} = \mathbf{uG}$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ & g_0 & g_1 & g_0 & \dots & g_m \\ & & g_1 & g_1 & g_0 & \dots \\ & & & g_2 & g_1 & \dots \\ & & & & g_2 & \dots \end{bmatrix}$$

where $g_i = (g_i^{(1)} g_i^{(2)} \dots g_i^{(n)})$, $0 \leq i \leq m$

Handwritten annotations show the calculation of the output vector \mathbf{v} as a sum of shifted products of the input vector $\mathbf{u} = [u_0 u_1 u_2 \dots]$ and the rows of \mathbf{G} :

- $u_0 g_0$
- $u_1 g_0 + u_0 g_1$
- $u_2 g_0 + u_1 g_1 + u_0 g_2$
- $u_3 g_0 + u_2 g_1 + u_1 g_2 + u_0 g_3$

A diagram below the matrix shows the overlapping nature of these products, with circles and arrows indicating the shifting of bits.

Sample for a memory to code we should that this is \mathbf{G} is of the form this okay and where each of this $g^{(i)}$, are basically these will represent what are these and bit output.

(Refer Slide Time: 27: 41)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $g^{(1)} = (1 0 1)$, and $g^{(2)} = (1 1 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & & & \\ & 11 & 01 & 11 & & \\ & & 11 & 01 & 11 & \\ & & & 11 & 01 & 11 \\ & & & & 11 & 01 & 11 \\ & & & & & \dots & \dots \end{bmatrix}$$

So let us continue with example that we are considering so far so we are continuing with our rate $1/2$ code who is memory orders is two and we know.

(Refer Slide Time: 27: 53)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1\ 0\ 1)$, and $\mathbf{g}^{(2)} = (1\ 1\ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & \dots & \dots \\ 00 & 11 & 01 & 11 & 00 & \dots \\ & & 11 & 01 & 11 & \dots \\ & & & 11 & 01 & 11 & \dots \\ & & & & 11 & 01 & 11 & \dots \end{bmatrix}$$

$$\begin{aligned} g_0 &= (1\ 1) \\ g_1 &= (0\ 1) \\ g_2 &= (1\ 1) \end{aligned}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & g_2 & 0 & 0 & \dots \\ & g_0 & g_1 & g_2 & 0 & \dots \\ & & g_0 & g_1 & g_2 & 0 & \dots \end{bmatrix}$$

Our generator sequence for the first code sequences given by 101 because my out $v^{(1)}$ is $u_1 + u_{1-2}$ similarly the generator sequence for the second code word is give by 11 okay then can I write basically what is my $g^{(0)}$, $g^{(1)}$, and $g^{(2)}$, so $g^{(0)}$, is given by now there are two outputs so $g^{(0)}$, will have two terms the first term corresponding to the fist coded sequence so higher this is one and what about the second code is sequence that is one so g_0 is 11 $g^{(1)}$, is this is zero so this zero and this one.

So g_1 is 01 and what about $g^{(2)}$, $g^{(2)}$, is this is one and this is one so $g^{(2)}$, is 11. So I can then write my generator matrix which is of the form \mathbf{G} is of the form $g^{(0)}$, $g^{(1)}$, $g^{(2)}$, and the rest all of these are basically zero these are zero these are zero this is g_0 , g_1 g_2 and then these are all zero. So what is $g^{(0)}$, $g^{(0)}$, is 11 so that is what I have written here g_1 is 01 that is what I have written here and $g^{(2)}$, is 11 the rest all these entries are zero.

Similarly this is 00 and then I have $g^{(0)}$, $g^{(1)}$, $g^{(2)}$, and then these are all zeros okay so this how I can write a generator matrix.

(Refer Slide Time: 29: 44)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$\begin{aligned} g_0 &= (1 \ 1) \\ g_1 &= (0 \ 1) \\ g_2 &= (1 \ 1) \end{aligned}$$

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & & & & \\ 00 & 11 & 01 & 11 & 00 & \dots & \\ & & 11 & 01 & 11 & & \\ & & & 11 & 01 & 11 & \\ & & & & 11 & 01 & 11 \\ & & & & & \dots & \dots \end{bmatrix}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & g_2 & 0 & 0 & \dots \\ & g_0 & g_1 & g_2 & 0 & \dots \end{bmatrix}$$

(Refer Slide Time: 29: 59)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by
$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$
$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$
- The code sequence can be written as
$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

Our coded sequence corresponding to this information sequence now let us try using this generator.

(Refer Slide Time: 30: 10)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $[v_0 v_1 v_2 \dots] = v = uG$ where $u = [u_0 u_1 u_2 \dots]$

$$G = \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ & g_0 & g_1 & g_2 & \dots & g_m \\ & & \ddots & \ddots & \ddots & \vdots \\ & & & g_1 & g_2 & \dots \\ & & & & g_2 & \dots \end{bmatrix}$$

where $g_i = (g_i^{(1)} g_i^{(2)} \dots g_i^{(n)}), 0 \leq i \leq m$

Handwritten notes show the expansion of the matrix multiplication:

$$v_0 = u_0 g_0$$

$$v_1 = u_1 g_0 + u_0 g_1$$

$$v_2 = u_2 g_0 + u_1 g_1 + u_0 g_2$$

$$v_3 = u_3 g_0 + u_2 g_1 + u_1 g_2 + u_0 g_3$$

A diagram shows the generator matrix G with columns $g_0, g_1, g_2, \dots, g_m$ and rows of zeros shifted to the right.

(Refer Slide Time: 30: 11)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $g^{(1)} = (1 \ 0 \ 1)$, and $g^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$G = \begin{bmatrix} 11 & 01 & 11 & \dots & 00 & \dots \\ 00 & 11 & 01 & 11 & 00 & \dots \\ & & 11 & 01 & 11 & \dots \\ & & & 11 & 01 & 11 & \dots \\ & & & & 11 & 01 & 11 & \dots \end{bmatrix}$$

Handwritten notes show the generator vectors:

$$g_0 = (1 \ 1)$$

$$g_1 = (0 \ 1)$$

$$g_2 = (1 \ 1)$$

A diagram shows the generator matrix G with columns g_0, g_1, g_2 and rows of zeros shifted to the right.

See if you use

(Refer Slide Time: 30: 12)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & \dots & \dots \\ 00 & 11 & 01 & 11 & 00 & \dots \\ & & 11 & 01 & 11 & \dots \\ & & & 11 & 01 & 11 & \dots \\ & & & & 11 & 01 & 11 & \dots \end{bmatrix}$$

$$\begin{aligned} \mathbf{g}_0 &= (1 \ 1) \\ \mathbf{g}_1 &= (0 \ 1) \\ \mathbf{g}_2 &= (1 \ 1) \end{aligned}$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & 0 & 0 & \dots \\ & \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & 0 & \dots \end{bmatrix}$$

(Refer Slide Time: 30: 43)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $[v_0 \ v_1 \ v_2 \ \dots] = \mathbf{v} = \mathbf{u} \mathbf{G}$

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \dots & 0 & \dots & \mathbf{g}_m \\ \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_0 & \dots & \dots & \mathbf{g}_m \\ & & \mathbf{g}_1 & \mathbf{g}_0 & \dots & \mathbf{g}_m \\ & & & \mathbf{g}_2 & \dots & \mathbf{g}_m \\ & & & & \dots & \mathbf{g}_m \end{bmatrix}$$

$$\mathbf{v} = \mathbf{u} \mathbf{G}$$

where

$$\mathbf{g}_i = (\mathbf{g}_i^{(1)} \ \mathbf{g}_i^{(2)} \ \dots \ \mathbf{g}_i^{(n)}), \quad 0 \leq i \leq m$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \dots & \mathbf{g}_m \\ & \mathbf{g}_0 & \mathbf{g}_1 & \dots & \mathbf{g}_m \\ & & \mathbf{g}_1 & \mathbf{g}_0 & \dots & \mathbf{g}_m \\ & & & \mathbf{g}_2 & \dots & \mathbf{g}_m \\ & & & & \dots & \mathbf{g}_m \end{bmatrix}$$

(Refer Slide Time: 30: 44)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1011100 \dots)$.
- The output sequences are given by

$$\begin{aligned} \mathbf{v}^{(1)} &= (1001011 \dots) \\ \mathbf{v}^{(2)} &= (1100101 \dots) \end{aligned}$$
- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

So we are getting the same output sequence

(Refer Slide Time: 30: 49)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $[v_0 \ v_1 \ v_2 \ \dots] = \mathbf{u} \mathbf{G}$

$$\mathbf{G} = \begin{bmatrix} \underline{g_0} & g_1 & \dots & 0 & \dots & g_m \\ g_0 & g_1 & g_0 & \dots & \dots & g_m \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \end{bmatrix}$$

where $\mathbf{g}_i = (g_i^{(1)} \ g_i^{(1)} \ \dots \ g_i^{(n)}), \ 0 \leq i \leq m$

Handwritten notes on the slide show the expansion of the matrix multiplication:

$$\mathbf{v} = \mathbf{u} \mathbf{G} \Rightarrow [v_0 \ v_1 \ v_2 \ \dots] = [u_0 \ u_1 \ u_2 \ \dots] \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ g_0 & g_1 & g_0 & \dots & \dots & g_m \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \end{bmatrix}$$

Resulting in:

$$v_0 = u_0 g_0$$

$$v_1 = u_1 g_0 + u_0 g_1$$

$$v_2 = u_2 g_0 + u_1 g_1 + u_0 g_2$$

$$v_3 = u_3 g_0 + u_2 g_1 + u_1 g_2 + u_0 g_3$$

(Refer Slide Time: 30: 50)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$\mathbf{g}_0 = (1 \ 1)$
 $\mathbf{g}_1 = (0 \ 1)$
 $\mathbf{g}_2 = (1 \ 1)$

$$\mathbf{G} = \begin{bmatrix} \underline{11} & \underline{01} & \underline{11} & & & & \\ \underline{00} & \underline{11} & \underline{01} & \underline{11} & \underline{00} & \dots & \\ & & \underline{11} & \underline{01} & \underline{11} & & \\ & & & \underline{11} & \underline{01} & \underline{11} & \\ & & & & \underline{11} & \underline{01} & \underline{11} \\ & & & & & \dots & \dots \end{bmatrix}$$

$\mathbf{G}_2 = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & 0 & 0 & \dots \\ & \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & 0 & \dots \end{bmatrix}$

As before.

(Refer Slide Time: 30: 52)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & & & & & & & \\ & 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & & & & & & \\ & & 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & & & & & \\ & & & 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & & & & \\ & & & & 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & & & \\ & & & & & 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & & \\ & & & & & & 1 \ 1 \ 0 \ 1 \ 1 \ 1 & & & & \\ & & & & & & & \ddots & & & \\ & & & & & & & & \ddots & & \\ & & & & & & & & & \ddots & \end{bmatrix}$$

- For $\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$,

$$\mathbf{v} = \mathbf{uG} = (\underline{11}, \underline{01}, \underline{00}, \underline{10}, \underline{01}, \underline{10}, \underline{11}, \underline{00}, \underline{00}, \dots)$$

(Refer Slide Time: 30: 56)

Encoding of $(n, 1, m)$ convolutional code

- Polynomial representation:

$$\mathbf{v}^{(i)}(D) = \mathbf{u}(D)\mathbf{g}^{(i)}(D), \quad 1 \leq i \leq n$$

$$\mathbf{v}(D) = \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{v}^{(n)}(D^n)$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1 \ 0 \ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1 \ 1 \ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} * \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} * \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

Now we are going to give a polynomial representation of these generator sequence which I very convenient in case of convolutional codes so I'm introducing

(Refer Slide Time: 31: 10)

Encoding of $(n, 1, m)$ convolutional code

- Polynomial representation:

$$\mathbf{v}^{(i)}(D) = \mathbf{u}(D)\mathbf{g}^{(i)}(D), \quad 1 \leq i \leq n$$

$$\mathbf{v}(D) = \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{v}^{(n)}(D^n)$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1 \ 0 \ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1 \ 1 \ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} * \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} * \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

A delay operator D so if you have one memory element delay that will be

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$\mathbf{v}^{(i)}(D) = \mathbf{u}(D)\mathbf{g}^{(i)}(D), \quad 1 \leq i \leq n$$

$$\mathbf{v}(D) = \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{v}^{(n)}(D^n)$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1 \ 0 \ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1 \ 1 \ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} * \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$\quad = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\quad = 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} * \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$\quad = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$\quad = 1 + D + D^4 + D^6$

D if have delay of 2 it will be D^2 if you have delay of three D^3 so the exponent of D is going to specify how much delay okay so what I'm going to show you is that I can write my output sequence in this polynomial notation as u times D into gi times D so every output codes sequence can be represent as product of this information sequence using this delay operator multiplied by this generator sequence in the delay operator farm work and the overall code sequence when we have a rate $1/n$ code can be given by this expression.

So let us first try to write each of these terms in terms of this delay operator polynomial representation and then we will show that this time domain representation where we where computing the output using convolutional discrete convolutional can be similarly obtained using just this operation in the delay domain which we are calling cross from domain operation so we will take the same example that we were considering so I will

(Refer Slide Time: 31: 32)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1\ 0\ 1)$, and $\mathbf{g}^{(2)} = (1\ 1\ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & & & & \\ & 11 & 01 & 11 & & & \\ & & 11 & 01 & 11 & & \\ & & & 11 & 01 & 11 & \\ & & & & 11 & 01 & 11 \\ & & & & & \ddots & \ddots \end{bmatrix}$$

$\mathbf{g}_0 = \begin{pmatrix} 1 & 1 \end{pmatrix}$
 $\mathbf{x}_1 = \begin{pmatrix} 0 & 1 \end{pmatrix}$

- For $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$,

$$\mathbf{v} = \mathbf{uG} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, 00, 00, \dots)$$

(Refer Slide Time: 32: 33)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$\mathbf{g}_0 = (1 \ 1)$
 $\mathbf{g}_1 = (0 \ 1)$
 $\mathbf{g}_2 = (1 \ 1)$

$$\mathbf{G} = \begin{bmatrix} \underline{11} & \underline{01} & \underline{11} & \dots & \dots & \dots \\ \underline{00} & \underline{11} & \underline{01} & \underline{11} & \underline{00} & \dots \\ & & \underline{11} & \underline{01} & \underline{11} & \dots \\ & & & \underline{11} & \underline{01} & \underline{11} \\ & & & & \underline{11} & \underline{01} & \underline{11} \\ & & & & & \dots & \dots \end{bmatrix}$$

$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & 0 & 0 & \dots \\ & \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & 0 & \dots \end{bmatrix}$

Back and show you again.

(Refer Slide Time: 32: 33)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $[v_0 v_1 v_2 \dots] = \mathbf{u} \mathbf{G}$

$$\mathbf{v} = \mathbf{u} \mathbf{G}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ g_0 & g_1 & g_2 & \dots & \dots & g_m \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ g_0 & g_1 & g_2 & \dots & \dots & g_m \end{bmatrix}$$

where $\mathbf{g}_i = (g_i^{(1)} g_i^{(2)} \dots g_i^{(n)}), 0 \leq i \leq m$

Handwritten notes on the slide show the calculation of the first three rows of the code vector \mathbf{v} :

- Row 1: $v_0 = u_0 g_0$
- Row 2: $v_1 = u_1 g_0 + u_0 g_1$
- Row 3: $v_2 = u_2 g_0 + u_1 g_1 + u_0 g_2$

A diagram below the matrix shows the shift of the generator polynomial \mathbf{g}_i to the right in each row of the matrix, with arrows indicating the shift.

The

(Refer Slide Time: 32: 35)

Encoding of $(n, 1, m)$ convolutional code

$$v_k^{(1)} = u_k + u_{k-2}$$

$$v_k^{(2)} = u_k + u_{k-1} + u_{k-2}$$

- Let the information sequence $\mathbf{u} = (1 0 1 1 1 0 0 \dots)$.

The slide shows a diagram of a convolutional encoder with two parallel shift registers. The top register has a feedback loop and a tap at the second stage. The bottom register has a tap at the second stage. The outputs are summed to produce the code vector \mathbf{v} .

Convolutional encoder.

(Refer Slide Time: 32: 36)

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$g^{(1)} = (1 \ 0 \ 1),$$

$$g^{(2)} = (1 \ 1 \ 1),$$

Handwritten equations:

$$v_l^{(1)} = u_l + u_{l-2}$$

$$v_l^{(2)} = u_l + u_{l-1} + u_{l-2}$$

That we are considering we have one input we have two outputs output depends on past two inputs so basically memory order is two $g^{(1)}$ is give by this $g^{(2)}$ is give by this these are my output $v_l^{(1)}$, $v_l^{(2)}$ these are my output sequences.

(Refer Slide Time: 32: 59)

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$\mathbf{v}^{(i)}(D) = \mathbf{u}(D)\mathbf{g}^{(i)}(D), \quad 1 \leq i \leq n$$

$$\mathbf{v}(D) = \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{v}^{(n)}(D^n)$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1 \ 0 \ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1 \ 1 \ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} * \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$\quad = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\quad = 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} * \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$\quad = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$\quad = 1 + D + D^4 + D^6$

Okay so let us look this so

(Refer Slide Time: 33: 04)

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$\mathbf{v}^{(i)}(D) = \mathbf{u}(D)\mathbf{g}^{(i)}(D), \quad 1 \leq i \leq n$$

$$\mathbf{v}(D) = \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{v}^{(n)}(D^n)$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1 \ 0 \ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1 \ 1 \ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} * \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} * \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

$g^{(1)}$ is 101 now what does this one corresponds to one corresponds to this connection $g^{(0)}$ which is linking my input $u(1)$ so that would be $u(1)$ without any delay so that would be 1 what was this corresponds to $g^{(1)}$ that is input delayed by one so this will be representing using D so D time zero will be zero and this will be this will corresponds to $g^{(2)}$ basically and this is delay of two so this will be represented using D^2 so this $g^{(1)}$ in this transform domain using this delay operator can be written as $1+D^2$ similarly this $g^{(2)}$ which is 111 can be written as $1+D+D^2$ so this my $g^{(2)}$ of D .

now the information sequence also I can write in this D delay notation since is $u^0 u^1 u^2 u^3$ so this is a informative sequence I'm getting at this time this after one delay element two, three, four so then this will be $1+D^2+D^3+D^4$ and that's I basically my informative sequence now the discrete convolutional of information sequence is $g^{(1)}$ is basically given by this and this if I write in a delay operator form will be what $1+D+D^2+D^3+D^4+D^5+D^6$ and what is $u(D)$ $u(D)$ ia given by this D^1 is given by this so let us multiply these two so what do we get .

so if we multiply $u(D)/g^{(1)}(D)$ so one time this will be $1+D+D^2+D^3+D^4+D^2$ rimes one is D^2 this will be D^4 this will be D^5 and this will be D^6 SO D^2+D^2 is zero D^4+D^4 is zero so what we are left with $1+D^3+D^5+D^6$ this is prissily what you get here okay so you can see basically these two in representation is equivalent similarly we can write $u^{(2)}$ which is give by this and you can verify for yourself $u^{(2)}D$ is given by this.

Now once you have these individual sequence how do you write the overall output sequence so note that for one input sequence you are getting an outputs okay so this is taken care by.

(Refer Slide Time: 33: 12)

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$v^{(i)}(D) = u(D)g^{(i)}(D), \quad 1 \leq i \leq n$$

$$v(D) = v^{(1)}(D^n) + Dv^{(2)}(D^n) + \dots + D^{n-1}v^{(n)}(D^n)$$

Time Domain	Transform Domain
$g^{(1)} = (1 \ 0 \ 1)$	$g^{(1)}(D) = 1 + D^2$
$g^{(2)} = (1 \ 1 \ 1)$	$g^{(2)}(D) = 1 + D + D^2$
$u = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$u(D) = 1 + D^2 + D^3 + D^4$
$v^{(1)} = u * g^{(1)}$	$v^{(1)}(D) = u(D)g^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$v^{(2)} = u * g^{(2)}$	$v^{(2)}(D) = u(D)g^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

$D^2 \quad D^3$

$1 + D^2 + D^3 + D^4$
 $+ D^2 + D^4 + D^5$
 $+ D^6$

$1 + D^3 + D^5 + D^6$

This so if $v^{(i)}D$ is going to give me output sequence corresponding top each of this output n output outputs now if I can combine this n outputs in this particular section so I take the first output note that I have made a D over in because the if the rate 1/n code the first output will appear after every n bits. The first bit is from the first coded sequence then after n bits it will again repeat it will come meaning so that is.

(Refer Slide Time: 36: 50)

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$v^{(i)}(D) = u(D)g^{(i)}(D), \quad 1 \leq i \leq n$$

$$v(D) = v^{(1)}(D^n) + Dv^{(2)}(D^n) + \dots + D^{n-1}v^{(n)}(D^n)$$

Time Domain	Transform Domain
$g^{(1)} = (1 \ 0 \ 1)$	$g^{(1)}(D) = 1 + D^2$
$g^{(2)} = (1 \ 1 \ 1)$	$g^{(2)}(D) = 1 + D + D^2$
$u = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$u(D) = 1 + D^2 + D^3 + D^4$
$v^{(1)} = u * g^{(1)}$	$v^{(1)}(D) = u(D)g^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$v^{(2)} = u * g^{(2)}$	$v^{(2)}(D) = u(D)g^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

$D^2 \quad D^3$

Why I have made it $D(n)$ now how do I combine these n sequences so note I'm take this is output $v^{(1)}$ $D(n)$ is output from the first code sequence this is the output from the second sequence that is I delayed by one the output from third sequence is delayed by D^2 as similarly the output from the enough sequence it will delayed by $n-1$ go back here.

(Refer Slide Time: 37: 19)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1\ 0\ 1)$, and $\mathbf{g}^{(2)} = (1\ 1\ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 1\ 1 & 0\ 1 & 1\ 1 & & & \\ & 1\ 1 & 0\ 1 & 1\ 1 & & \\ & & 1\ 1 & 0\ 1 & 1\ 1 & \\ & & & 1\ 1 & 0\ 1 & 1\ 1 \\ & & & & 1\ 1 & 0\ 1 & 1\ 1 \\ & & & & & \ddots & \ddots \end{bmatrix}$$

- For $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$,

$$\mathbf{v} = \mathbf{uG} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, 00, 00, \dots)$$

(Refer Slide Time: 37: 20)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by $R = \frac{1}{2}$

$$\begin{aligned} \mathbf{v}^{(1)} &= (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots) \quad \checkmark \\ \mathbf{v}^{(2)} &= (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots) \quad \checkmark \end{aligned}$$

- The code sequence can be written as

$$\mathbf{v} = (\underline{11}, \underline{01}, \underline{00}, 10, 01, 10, 11, \dots)$$

These are the two individual outputs how where we getting the final output so note here I'm taking first bit from here that is one second bit I'm taking from her that is this the third bit is this which is this fourth bit is this which is this so what is what am I doing in this case rate was one half so after every you can see in the output every second bit is coming from this so this is my one which is appearing here this is my zero which is this is my one which is appearing this so note this appearing every second bit and that is I why what we did.

(Refer Slide Time: 38: 07)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & \dots & \dots & \dots \\ 00 & 11 & 01 & 11 & 00 & \dots \\ & & 11 & 01 & 11 & \dots \\ & & & 11 & 01 & 11 & \dots \\ & & & & 11 & 01 & 11 & \dots \end{bmatrix}$$

Handwritten notes on the slide:

- $g_0 = (1 \ 1)$
- $g_1 = (0 \ 1)$
- $g_2 = (1 \ 1)$
- $\mathbf{G} = \begin{bmatrix} g_0 & g_1 & g_2 & 0 & 0 & \dots \\ g_0 & g_1 & g_2 & 0 & 0 & \dots \end{bmatrix}$

(Refer Slide Time: 38: 08)

Encoding of $(n, 1, m)$ convolutional code

Polynomial representation:

$$\mathbf{v}^{(i)}(D) = \mathbf{u}(D)\mathbf{g}^{(i)}(D), \quad 1 \leq i \leq n$$

$$\mathbf{v}(D) = \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{v}^{(n)}(D^n)$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1 \ 0 \ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1 \ 1 \ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} * \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} * \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

Handwritten notes on the slide:

- $D^2 \ D^3$
- $1 + D^2 + D^3 + D^4 + D^5 + D^6$
- $1 + D^3 + D^5 + D^6$
- $1 + D + D^4 + D^6$

Combined we made it each of this coded bit we made it D forward in next.

(Refer Slide Time: 38: 08)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1 \ 0 \ 1)$, and $\mathbf{g}^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 11 & 01 & 11 & \dots & \dots & \dots \\ 00 & 11 & 01 & 11 & 00 & \dots \\ & & 11 & 01 & 11 & \dots \\ & & & 11 & 01 & 11 \\ & & & & 11 & 01 & 11 \\ & & & & & \dots & \dots \end{bmatrix}$$

$$\begin{aligned} g_0 &= (1 \ 1) \\ g_1 &= (0 \ 1) \\ g_2 &= (1 \ 1) \end{aligned}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & g_2 & 0 & 0 & \dots \\ g_0 & g_1 & g_2 & 0 & 0 & \dots \end{bmatrix}$$

If you look here.

(Refer Slide Time: 38: 20)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $\mathbf{v} = \mathbf{uG}$

$$\mathbf{v} = [v_0 \ v_1 \ v_2 \ \dots]$$

$$\mathbf{u} = [u_0 \ u_1 \ u_2 \ \dots]$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ g_0 & g_1 & g_2 & \dots & \dots & g_m \\ & & & g_1 & & \\ & & & & g_2 & \\ & & & & & \dots \end{bmatrix}$$

where

$$g_i = (g_i^{(1)} \ g_i^{(2)} \ \dots \ g_i^{(n)}), \quad 0 \leq i \leq m$$

$$\mathbf{G} = \begin{bmatrix} g_0 & \dots & g_m \\ 0 & g_1 & \dots & g_m \\ & 0 & g_2 & \dots & g_m \\ & & & \dots & \dots \end{bmatrix}$$

$$\begin{aligned} &u_0 g_0 \\ &u_1 g_0 + u_0 g_1 \\ &u_2 g_0 + u_1 g_1 + u_0 g_2 \\ &u_3 g_0 + u_2 g_1 + u_1 g_2 \end{aligned}$$

(Refer Slide Time: 38: 21)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by $R = \frac{1}{2}$

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots) \quad \checkmark$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots) \quad \checkmark$$
- The code sequence can be written as
$$\mathbf{v} = (11\ |01\ |00\ |10\ |01\ |10\ |11\ | \dots)$$

The first coded sequence is

(Refer Slide Time: 38: 21)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by $R = \frac{1}{2}$

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots) \quad \checkmark$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots) \quad \checkmark$$
- The code sequence can be written as
$$\mathbf{v} = (11\ |01\ |00\ |10\ |01\ |10\ |11\ | \dots)$$

this one this is the output from the first code sequence and what is the output form the second sequence which is this one so what are you doing when your combining these output sequence which is $v^{(1)}$ and $v^{(2)}$ so your taking $v^{(1)}$ like as it is only things is, is this spread out after every second bit and

(Refer Slide Time: 38: 52)

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \dots)$.
- The output sequences are given by $R = \frac{1}{2}$

$$\mathbf{v}^{(1)} = (1\ 0\ 0\ 1\ 0\ 1\ 1\ \dots)$$

$$\mathbf{v}^{(2)} = (1\ 1\ 0\ 0\ 1\ 0\ 1\ \dots)$$
- The code sequence can be written as
$$\mathbf{v} = (11\ 01\ 00\ 10\ 01\ 10\ 11, \dots)$$

$v^{(2)}$ is delayed by one and it is also spread out this is one this one is appearing here this zero is appearing here this is appearing here so every second bit is also from this encoded sequence and note that this is delayed by one corresponding to $v^{(1)}$ so that is what we are doing.

(Refer Slide Time: 39: 12)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form $[v_0 v_1 v_2 \dots] = v = [u_0 u_1 u_2 \dots] G$

$$v = uG$$

$$G = \begin{bmatrix} g_0 & g_1 & \dots & 0 & \dots & g_m \\ g_0 & g_1 & g_2 & \dots & \dots & g_m \\ & g_1 & g_2 & g_3 & \dots & \dots \\ & & g_2 & g_3 & g_4 & \dots \\ & & & g_3 & g_4 & g_5 & \dots \\ & & & & g_4 & g_5 & g_6 & \dots \end{bmatrix}$$

where $g_i = (g_i^{(1)} \ g_i^{(2)} \ \dots \ g_i^{(n)}), 0 \leq i \leq m$

Handwritten notes on the slide show the expansion of the matrix multiplication:

- $u_0 g_0$
- $u_1 g_0 + u_0 g_1$
- $u_2 g_0 + u_1 g_1 + u_0 g_2$
- $u_3 g_0 + u_2 g_1 + u_1 g_2$

A diagram below the matrix shows the shift of generator vectors g_i and their overlap in the matrix structure.

(Refer Slide Time: 39: 13)

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $g^{(1)} = (1 \ 0 \ 1)$, and $g^{(2)} = (1 \ 1 \ 1)$, the generator matrix is given by

$$G = \begin{bmatrix} 11 & 01 & 11 & \dots & \dots \\ 00 & 11 & 01 & 11 & 00 & \dots \\ & & 11 & 01 & 11 & \dots \\ & & & 11 & 01 & 11 & \dots \\ & & & & 11 & 01 & 11 & \dots \end{bmatrix}$$

Handwritten notes on the slide show the generator vectors:

- $g_0 = (1 \ 1)$
- $g_1 = (0 \ 1)$
- $g_2 = (1 \ 1)$

A diagram below the matrix shows the structure of the generator matrix G with columns labeled $g_0, g_1, g_2, 0, 0, \dots$.

(Refer Slide Time: 39: 13)

Encoding of $(n, 1, m)$ convolutional code

- Polynomial representation:

$$\begin{aligned} v^{(i)}(D) &= u(D)g^{(i)}(D), \quad 1 \leq i \leq n \\ v(D) &= v^{(1)}(D^n) + Dv^{(2)}(D^n) + \dots + D^{n-1}v^{(n)}(D^n) \end{aligned}$$

Time Domain

$$g^{(1)} = (1 \ 0 \ 1)$$

$$g^{(2)} = (1 \ 1 \ 1)$$

$$u = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$$

$$v^{(1)} = u * g^{(1)}$$

$$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$$

$$v^{(2)} = u * g^{(2)}$$

$$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$$

Transform Domain

$$g^{(1)}(D) = 1 + D^2$$

$$g^{(2)}(D) = 1 + D + D^2$$

$$u(D) = 1 + D^2 + D^3 + D^4$$

$$v^{(1)}(D) = u(D)g^{(1)}(D)$$

$$= 1 + D^3 + D^5 + D^6$$

$$v^{(2)}(D) = u(D)g^{(2)}(D)$$

$$= 1 + D + D^4 + D^6$$

If you combine this consider this combine output sequence there are n coded sequence $v^{(1)}$ $v^{(2)}$ $v^{(3)}$ $v^{(n)}$ now for sequence we just take $v^{(1)} D^n$ second is $v^{(2)} D^n$ third is $v^{(3)} D^n$ and then each one of them are delayed by 1 1 so this is no delay this is delay of one delay of two and this delay of $n-1$ so over all code sequence will be give by this

(Refer Slide Time: 39: 44)

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$v^{(i)}(D) = u(D)g^{(i)}(D), \quad 1 \leq i \leq n$$

$$v(D) = v^{(1)}(D^n) + Dv^{(2)}(D^n) + \dots + D^{n-1}v^{(n)}(D^n)$$

Time Domain	Transform Domain
$g^{(1)} = (1 \ 0 \ 1)$	$g^{(1)}(D) = 1 + D^2$
$g^{(2)} = (1 \ 1 \ 1)$	$g^{(2)}(D) = 1 + D + D^2$
$u = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$u(D) = 1 + D^2 + D^3 + D^4$
$v^{(1)} = u * g^{(1)}$	$v^{(1)}(D) = u(D)g^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$v^{(2)} = u * g^{(2)}$	$v^{(2)}(D) = u(D)g^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

$D^2 \ D^3$

Expression so I hope I made it clear why this is D^n and why each the parity bits re delayed by 1 D, D^2, D^3, D^{n-1} so following this basically we.

(Refer Slide Time: 40: 03)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,
$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$

where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$

Basically we can also write our encoding sequence in this

(Refer Slide Time: 40: 08)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\underline{v(D) = u(D^n)g(D)}$$

where

$$\underline{g(D) \triangleq g^{(1)}(D^n) + Dg^{(2)}(D^n) + \dots + D^{n-1}g^{(n)}(D^n)}$$

Particular form where output sequence is give by $u(D^n)$ times $g(D)$ where $g(D)$ is this a generator sequence for the first coded sequence the generator sequence form the second delayed by one generator sequence of the third delayed by two generator sequence of the n delayed by D^n so the overall encoding sequence can be equivalently written like this .

(Refer Slide Time: 40: 34)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,
$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$
where
$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$
- Example: Time Domain
$$\mathbf{v} = \mathbf{u}\mathbf{G} = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$$

And we can again go back to the same example our output sequence in a time domain was given by this and if we follow the same procedure.

(Refer Slide Time: 40: 34)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$
- where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$
- Example: Time Domain

$$\mathbf{v} = \mathbf{u}\mathbf{G} = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$$
- Example: Transform Domain

$$\begin{aligned} \mathbf{v}(D) &= \mathbf{u}(D^2)\mathbf{g}(D) = (1 + D^4 + D^6 + D^8)(1 + D + D^3 + D^4 + D^5) \\ &= 1 + D + D^3 + D^6 + D^9 + D^{10} + D^{12} + D^{13} \end{aligned}$$

Of $\mathbf{v}(D)$ should be $\mathbf{u}(D^2)$ times $\mathbf{g}(D)$ where $\mathbf{g}(D)$ is $g^{(1)}D^2 + Dg^{(2)}D^2$ so $\mathbf{u}(D^2)$ is

(Refer Slide Time: 41: 00)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as, $u(D) =$

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$
- where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$
- Example: Time Domain

$$\mathbf{v} = \mathbf{u}\mathbf{G} = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$$
- Example: Transform Domain

$$\begin{aligned} \mathbf{v}(D) &= \mathbf{u}(D^n)\mathbf{g}(D) = (1 + D^4 + D^6 + D^8)(1 + D + D^3 + D^4 + D^5) \\ &= 1 + D + D^3 + D^6 + D^9 + D^{10} + D^{12} + D^{13} \end{aligned}$$

What is $u(D)$ go back the example

(Refer Slide Time: 41: 04)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,
$$\underline{v(D) = u(D^n)g(D)}$$

where

$$g(D) \triangleq \underline{g^{(1)}(D^n)} + D\underline{g^{(2)}(D^n)} + \dots + D^{n-1}\underline{g^{(n)}(D^n)}$$

(Refer Slide Time: 41: 05)

Encoding of $(n, 1, m)$ convolutional code

• Polynomial representation:

$$v^{(i)}(D) = u(D)g^{(i)}(D), \quad 1 \leq i \leq n$$

$$v(D) = v^{(1)}(D^n) + Dv^{(2)}(D^n) + \dots + D^{n-1}v^{(n)}(D^n)$$

Time Domain	Transform Domain
$g^{(1)} = (1 \ 0 \ 1)$	$g^{(1)}(D) = 1 + D^2$
$g^{(2)} = (1 \ 1 \ 1)$	$g^{(2)}(D) = 1 + D + D^2$
$u = (1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$u(D) = 1 + D^2 + D^3 + D^4$
$v^{(1)} = u * g^{(1)}$	$v^{(1)}(D) = u(D)g^{(1)}(D)$
$= (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D^3 + D^5 + D^6$
$v^{(2)} = u * g^{(2)}$	$v^{(2)}(D) = u(D)g^{(2)}(D)$
$= (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots)$	$= 1 + D + D^4 + D^6$

Handwritten notes: $D^2 \ D^3$ above the polynomial representation; $1+D^2+D^3+D^4$ above the transform domain $u(D)$; $1+D^3+D^5+D^6$ above the transform domain $v^{(1)}(D)$; $1+D+D^4+D^6$ above the transform domain $v^{(2)}(D)$.

$U(D)$ is $1+D^2+D^3+D^4$ so this is.

(Refer Slide Time: 41: 11)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,
$$\underline{v(D) = u(D^n)g(D)}$$
where
$$\underline{g(D) \triangleq g^{(1)}(D^n) + Dg^{(2)}(D^n) + \dots + D^{n-1}g^{(n)}(D^n)}$$

(Refer Slide Time: 41: 04)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$v(D) = u(D^n)g(D)$$

where

$$g(D) \triangleq g^{(1)}(D^n) + Dg^{(2)}(D^n) + \dots + D^{n-1}g^{(n)}(D^n)$$

Example: Time Domain

$$v = uG = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$$

Example: Transform Domain

$$v(D) = u(D^n)g(D) = (1 + D^4 + D^5 + D^8)(1 + D + D^3 + D^4 + D^5)$$

$$= 1 + D + D^3 + D^5 + D^9 + D^{10} + D^{12} + D^{13}$$

11010010011011

$u(D)$ is $1+D^2+D^3+D^4$ so $u(D^2)$ will be $1+D^4+D^6+D^8$ so that is what we have written here okay and what is $g^{(1)}$ $g^{(2)}$ $g^{(1)}$ $D^2 + D$ times $g^{(2)}$ D^2 and what is $g^{(1)}$ D and $g^{(2)}$ D $v^{(1)}$ D is $1+D^2$ and this was $1+D+D^2$ so $g^{(1)}$ D^2 will be $1+D^4$ and this will be so this will be $1+D^4$ $1+D^4$ $g^{(2)}$ D will be $g^{(2)}$ D^2 this is so this term is give by this I hope this is clear so $g^{(1)}$ D is given by this what we are interested is $g^{(1)}$ D^2 so $g^{(1)}$ D^2 will b given by this expression and we are interest in $g^{(2)}$ D^2 so this will be given by $1+D^2+D^4$ now what is our overall $g(D)$ this is given by $g^{(1)}$ $D^2 + D$ times $g^{(2)}$ D^2 so then this will be $g^{(1)}$ D^2 is $1+D^4+D$ times this okay so this can be written as $1+D^4+D+D^3+D^5$

so this is $1+D$ this is $1+D+D^3+D^4+D^5$ okay so this our $g(D)$ now if you multiply all of them what we get is this and we can write this what is 1 D is this D^2 is zero D^3 is one D^4 is zero D^5 is zero D^6 is 1 D^7 is zero D^8 zero D^9 1 D^{10} one d^{11} zero d^{12} one d^{13} one so this is our output sequence now compare with this what we got in time domain 11,11,01,01,00,00,10,10,01,01 so you see basically we are getting same sequence 10, 10, 11, 11 and rest are all zeros it also we are getting all zeros so the point to take is these generator sequence that we wrote using this time domain representation we can similarly represent them using this display domain representation and it is not more covenant to write it in this particular notation because then the output sequence.

(Refer Slide Time: 44:40)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$
 where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$
- Example: Time Domain

$$\mathbf{v} = \mathbf{u}\mathbf{G} = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$$
- Example: Transform Domain

$$\begin{aligned} \mathbf{v}(D) &= \mathbf{u}(D^n)\mathbf{g}(D) = (1 + D^4 + D^5 + D^8)(1 + D + D^3 + D^4 + D^5) \\ &= 1 + D + D^3 + D^6 + D^9 + D^{10} + D^{12} + D^{13} \end{aligned}$$

Is just product of the input sequence and this generator sequence in this domain?

(Refer Slide Time: 44:40)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$
 where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$
- Example: Time Domain

$$\mathbf{v} = \mathbf{u}\mathbf{G} = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$$
- Example: Transform Domain

$$\begin{aligned} \mathbf{v}(D) &= \mathbf{u}(D^n)\mathbf{g}(D) = (1 + D^4 + D^5 + D^8)(1 + D + D^3 + D^4 + D^5) \\ &= 1 + D + D^3 + D^6 + D^9 + D^{10} + D^{12} + D^{13} \end{aligned}$$

So with this I'm going to conclude this lecture I just want to make another point that these generator matrix that we saw

(Refer Slide Time: 44:59)

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as, $g^{(1)}(D) = (101)$
 $g^{(2)}(D) = (111)$
 $R = \frac{1}{2} (5, 7)$

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$

where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \dots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$

- Example: Time Domain
 $\mathbf{v} = \mathbf{u}\mathbf{G} = (11, 01, 00, 10, 01, 10, 11, 00, 00, \dots)$
- Example: Transform Domain
 $\mathbf{v}(D) = \mathbf{u}(D^2)\mathbf{g}(D) = (1 + D^4 + D^6 + D^8)(1 + D + D^3 + D^5)$
 $= 1 + D + D^3 + D^6 + D^9 + D^{10} + D^{12} + D^{13}$

For example this $g^{(1)}(D)$ which we wrote as 101 and $g^{(2)}(D)$ which is 111 it is typically represented using octal notations so in many books when they will describe the convolutional encoder for this they will write it as five seven code because octal notation of this is five and octal notation of this is seven so in many places they will say it's a rate $\frac{1}{2}$ five seven code and what it means is they are specifying the generator sequence using this octal notation. Thank you.

Acknowledgement

Ministry of Human Resource & Development

Prof. Satyaki Roy

Co-ordinator, NPTEL IIT Kanpur

NPTEL Team

Sanjay Pal

Ashish Singh

Badal Pradhan

Tapobrata Das

Ram Chandra

Dilip Tripathi
Manoj Shrivastava
Padam Shukla
Sanjay Mishra
Shubham Rawat
Shikha Gupta
K.K. Mishra
Aradhana Singh
Sweta
Ashutosh Gairola
Dilip Katiyar
Sharwan
Hari Ram
Bhadra Rao
Puneet Kumar Bajpai
Lalty Dutta
Ajay Kanaujia
Shivendra Kumar Tiwari

an IIT Kanpur Production

©copy right reserved