

GEL7114

Module 5

Chapter 6: Introduction to FEC

Module 5

GEL7114

1

1

Topics covered

- Time coding and orthogonal codes ([Sec. 6.1-6.1.3.1](#))
- Channel models, parity, code rate and redundancy ([Sec. 6.3.1-6.3.3](#))
- Forward error correction ([Sec. 6.3.4](#))

Module 5

GEL7114

2

2

Coding

- To improve P_e
 - Accept expansion in bandwidth
 - Orthogonal, bi orthogonal, simplex
- To detect errors
 - Parity and retransmission
- To correct errors
 - Block codes
 - Convolutional codes
 - The best codes minimize bandwidth expansion

Module 5

GEL7114

3

3

Code rate

- Block encoder
- Block of k data bits enter
 - Block of n coded bits exit
- Code redundancy $\frac{n-k}{k}$
- Code rate $\frac{k}{n}$
- Bandwidth expansion $\frac{n}{k}$

Module 5

GEL7114

4

4

GEL7114

Hadamard Modulation

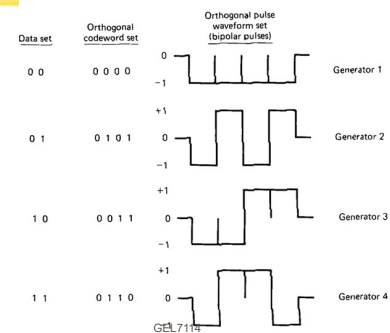
Module 5

GEL7114

5

5

Orthogonal Hadamard Modulation



Module 5

GEL7114

6

6

Cross correlation

- Piece-wise constant
that is, just +1 et -1

$$z_{ij} = \frac{1}{E} \langle s_i, s_j \rangle$$

$$= \frac{\# \text{ bits same} - \# \text{ bits differents}}{\# \text{ bits}}$$

bits same = # bits differents for Hadamard

Module 5

GEL7114

7

7

Orthogonal Hadamard Modulation

- Improved performance

$$P_e(M) = (M-1)Q\left(\sqrt{\frac{E_s}{N_0}}\right) = (M-1)Q\left(\sqrt{\frac{E \log_2 M}{N_0}}\right)$$

- BW inefficient

$$\eta_{\text{Hada}} = \frac{\log_2 M}{M} \text{ b/s/Hz}$$

k bits sans codage
 $M = 2^k$ bits avec codage

Module 5

GEL7114

8

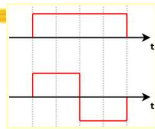
8

Orthogonal codes

- Hadamard matrices
- Binary case ($k=1, M=2^1=2$)

$$H_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

- First row 0 0 for logical zero
- Second row 0 1 for logical one



Module 5

GEL7114

9

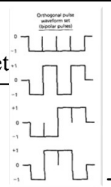
9

Data set

0 0
0 1
1 0
1 1

Orthogonal codeword set

$$H_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} H_1 & H_1 \\ H_1 & \bar{H}_1 \end{bmatrix}$$



Module 5

GEL7114

10

10

Hadamard Matrices

$$H_{n+1} = \begin{bmatrix} H_n & H_n \\ H_n & \bar{H}_n \end{bmatrix}$$

- For any pair of rows
 - #bits different = #bits same

$$z_{ij} = \frac{1}{E} \langle s_i, s_j \rangle = 0$$

Module 5

GEL7114

11

11

$\eta_{\text{ortho}} = \frac{\log_2 M}{M} \text{ b/s/Hz}$

Data set

0 0 0
0 0 1
0 1 0
0 1 1
1 0 0
1 0 1
1 1 0
1 1 1

Orthogonal codeword set

$$H_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} H_2 & H_2 \\ H_2 & \bar{H}_2 \end{bmatrix}$$

Module 5

GEL7114

12

12

Probability of error

- Orthogonal Codes

P_e = symbol error probability

$$P_e(M) = (M-1)Q\left(\sqrt{\frac{E_s}{N_0}}\right) = (M-1)Q\left(\sqrt{\frac{E_b \log_2 M}{N_0}}\right)$$

$$\frac{P_b}{P_e} = \frac{M/2}{M-1}$$

$$P_b \approx \frac{M}{2} Q\left(\sqrt{k \frac{E_b}{N}}\right)$$

Module 5

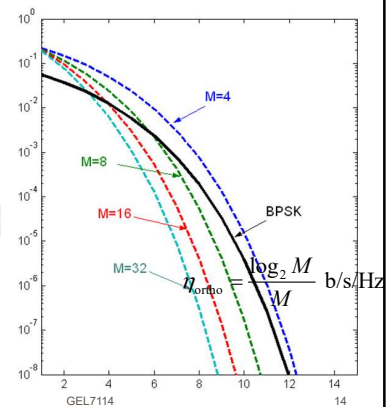
GEL7114

13

13

Performance

$$P_e(M) = (M-1)Q\left(\sqrt{\frac{E_b \log_2 M}{N_0}}\right)$$



Module 5

GEL7114

14

14

GEL7114

parity

Module 5

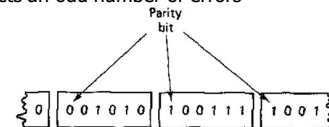
GEL7114

15

15

Parity

- Identify errors, but do not correct them
- ...
- Even and odd parity
- Rate $k/k+1$
 - Detects an odd number of errors



Module 5

GEL7114

16

16

Probability

- How many positions for j errors in a block of n bits

$$\binom{n}{j} = \frac{n!}{j!(n-j)!}$$

- Probability of j errors in n bits

$$\Pr(j, n) = \binom{n}{j} p^j (1-p)^{n-j}$$

- Probability of missing an error *i.e., even # of errors*

$$P_{nd} = \sum_{j=1}^{\lfloor n/2 \rfloor} \binom{n}{2j} p^{2j} (1-p)^{n-2j}$$

Module 5

GEL7114

17

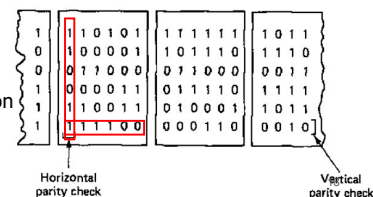
17

Vertical and horizontal parity

- Two-dimensional parity, block of $M \times N$ bits
- Add a bit to each row and column

$$\text{Rate} = \frac{MN}{(M+1)(N+1)} = \frac{MN}{MN + N + M + 1}$$

- Error correction possible



Module 5

Horizontal parity check

Vertical parity check

18

GEL7114

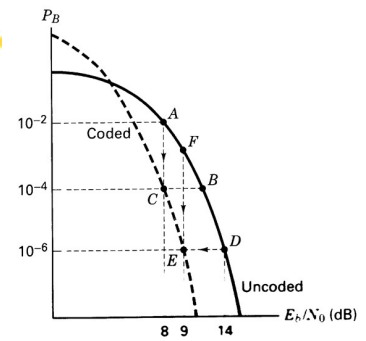
Forward error correction (FEC)

Module 5

GEL7114

19

Performance



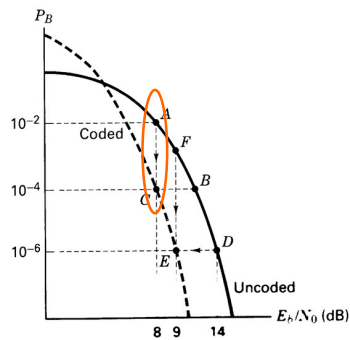
Module 5

GEL7114

20

Performance

Compromise:
less Power,
but less bandwidth
efficient



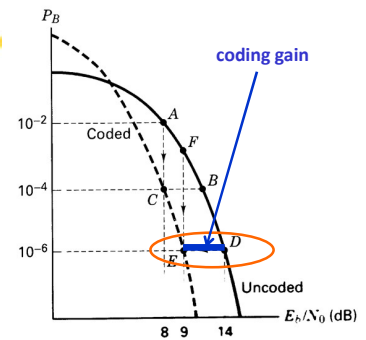
Module 5

GEL7114

21

Performance

Compromise:
less Power,
but less bandwidth
efficient



Module 5

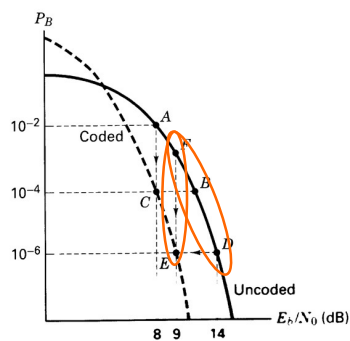
GEL7114

22

Performance

DEF with $R \uparrow$
FEE with $BW \uparrow$

Compromise:
less Power,
but less bandwidth
efficient



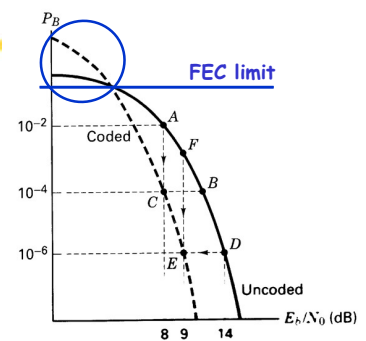
Module 5

GEL7114

23

???

How can coding
make performance
worse???



Module 5

GEL7114

24

After error correction

- BER for coded block $p_b = Q\left(\frac{\sqrt{2E_b}}{\sqrt{N_0}}\right) = 1.36 \times 10^{-4}$
- BER worsened before starting the correction process...
- A block with a single error will be corrected, but not others blocks $p_{be} = \sum_{j=1}^n \binom{n}{j} p^j (1-p)^{n-j} = \binom{n}{2} p^2 (1-p)^{n-2}$
- Our example $p_{be} = \binom{15}{2} (1.4 \times 10^{-4})^2 (1 - 1.4 \times 10^{-4})^{13} \approx 1.94 \times 10^{-8}$

this probability would have been terrible

that would result in a deterioration, not an improvement...

Module 5

GEL7114

31

31

GEL7114

Chapter 7

Module 5

GEL7114

32

32

Topics covered

- Convolutional encoding (Sec. 7.1)
- Representation of encoders (Sec. 7.2)
- The decoding algorithm (Sec. 7.3)
- Properties of convolutional codes (Sec. 7.4)

Module 5

GEL7114

33

33

Block vs. convolutional coding

- Same code rate k/n
- Block coding
 - isolated and independent block
 - no "memory" from one block to another
- Convolutional encoding
 - continuously encoded data
 - memory of ~~k~~1 blocks of k bits

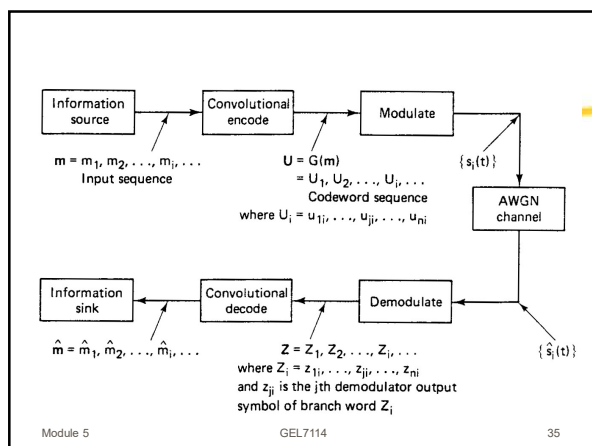
Constraint length

Module 5

GEL7114

34

34



35

Convolutional encoders

- Block diagram of shift registers
- Connection vectors and polynomials interpretation
- Impulse response
- State diagram
- Tree representation
- Trellis representation

Module 5

GEL7114

36

36

Why ...

- Physical implementations
- Mathematical manipulations
- Time behavior

Module 5

GEL7114

37

37

Block diagram of shift registers

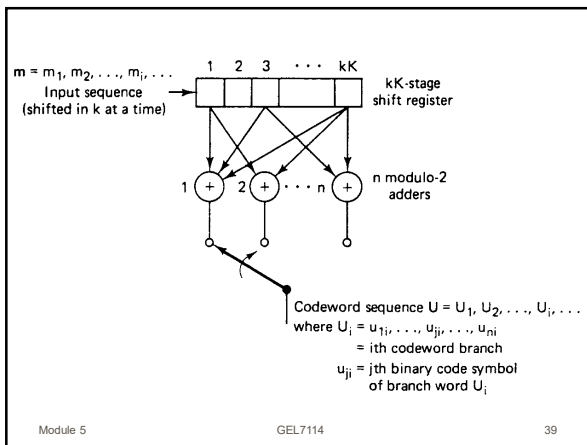
- $k \times K$ registers
- Blocks of k symbols enter and blocks of n symbols come out of the encoder
- Output a symbol function at the input
- Interconnections of shift registers determine the function
- We examine the case $k=1$

Module 5

GEL7114

38

38



Module 5

GEL7114

39

39

Connection vectors, polynomial interpretation

- The interconnections of the shift registers determines the code
- Let us use a vector of length $k \times K$ to specify the interconnections for each of the output symbols

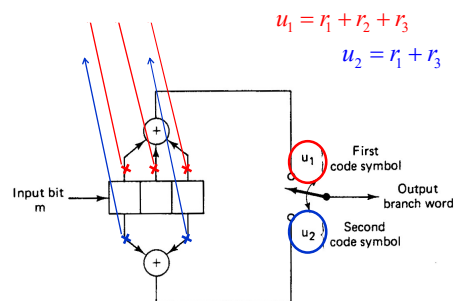
Module 5

GEL7114

40

40

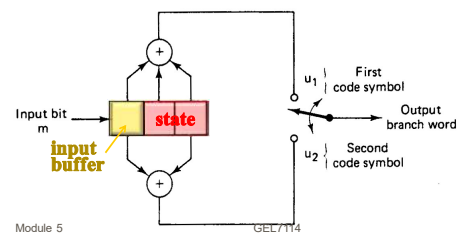
Exemple: $k=1, n=2, K=3$



41

41

Exemple: $k=1, n=2, K=3$



Module 5

GEL7114

42

42

Initialization

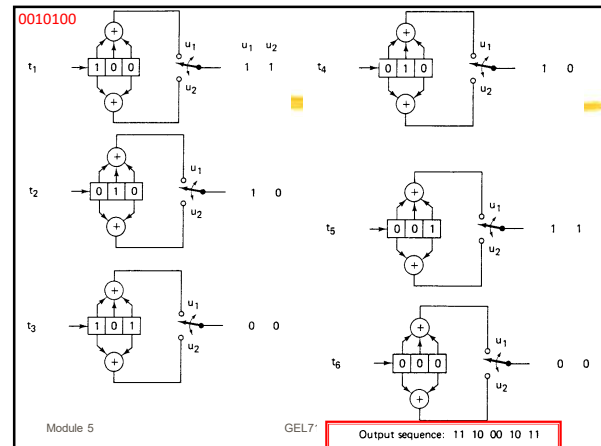
- Output is a function of the original state
- Typically assume that the registers are empty (0) to begin with
- Example : send message $m=101$
 - preceded by zeros
 - followed by zeroes

Module 5

GEL7114

43

43



44

GEL7114

Impulse response and polynomial form

Module 5

GEL7114

45

45

Impulse response

1 1 1 0 1 1

46

Impulse response

	Register contents	Branch word	
		u_1	u_2
	1 0 0	1	1
	0 1 0	1	0
	0 0 1	1	1

Input sequence: 1 0 0
Output sequence: 1 1 1 0 1 1

Module 5

GEL7114

47

47

Linearity

Input m	Output					
1	1 1	1 0	1 1			
0		0 0	0 0	0 0		
1			1 1	1 0	1 1	
Modulo-2 sum:	1 1	1 0	0 0	1 0	1 1	

Output sequence: 11 10 00 10 11

Module 5

GEL7114

48

48

Polynomial multiplication

$$\begin{array}{lll} m=101 & g_1(X) = 1 + X + X^2 & g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ m(X)=1+X^2 & g_2(X) = 1 + X^2 & g_2 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \end{array}$$

49

Polynomial multiplication

$$\begin{array}{lll} m=101 & g_1(X) = 1 + X + X^2 & g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ m(X)=1+X^2 & g_2(X) = 1 + X^2 & g_2 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \end{array}$$

$$m(X)g_1(X) = (1 + X^2)(1 + X + X^2) = 1 + X + X^3 + X^4$$

$$m(X)g_2(X) = (1 + X^2)(1 + X^2) = 1 + X^4$$

50

Polynomial multiplication

$$\begin{array}{lll} m=101 & g_1(X) = 1 + X + X^2 & g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ m(X)=1+X^2 & g_2(X) = 1 + X^2 & g_2 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \end{array}$$

$$m(X)g_1(X) = (1 + X^2)(1 + X + X^2) = 1 + X + X^3 + X^4$$

$$m(X)g_2(X) = (1 + X^2)(1 + X^2) = 1 + X^4$$

$$m(X)g_1(X) = 1 + X + 0X^2 + X^3 + X^4$$

$$m(X)g_2(X) = 1 + 0X + 0X^2 + 0X^3 + X^4$$

51

Polynomial multiplication

$$\begin{array}{lll} m=101 & g_1(X) = 1 + X + X^2 & g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\ m(X)=1+X^2 & g_2(X) = 1 + X^2 & g_2 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \end{array}$$

$$m(X)g_1(X) = (1 + X^2)(1 + X + X^2) = 1 + X + X^3 + X^4$$

$$m(X)g_2(X) = (1 + X^2)(1 + X^2) = 1 + X^4$$

$$m(X)g_1(X) = 1 + X + 0X^2 + X^3 + X^4$$

$$\begin{array}{r} m \\ - \end{array} \begin{array}{l} \text{Output sequence: } 11 \ 10 \ 00 \ 10 \ 11 \\ \hline U = 1 \ 1 \quad 1 \ 0 \quad 0 \ 0 \quad 1 \ 0 \quad 1 \ 1 \end{array} \begin{array}{l} \\ \\ \\ \\ 1)X^4 \end{array}$$

52

Convolutional encoders

- Block diagram of shift registers
- Connection vectors and polynomials interpretation
- Impulse response
- State diagram
- Tree representation
- Trellis representation

Module 5

GEL7114

53

53

GEL7114

Finite state machine

Module 5

GEL7114

54

54

Finite state machine

- The output at time t is a function of the contents of the registers
 - "The state of the registers"
- Encoding is the transition from one state to another state
- The number of states is finite
- Finite state machine

Module 5

GEL7114

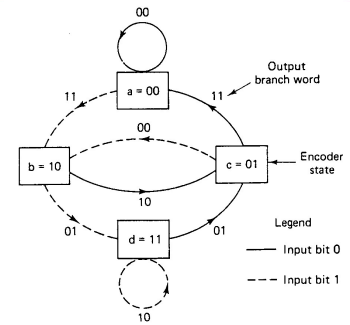
55

55

State diagram

$$g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$g_2 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$$

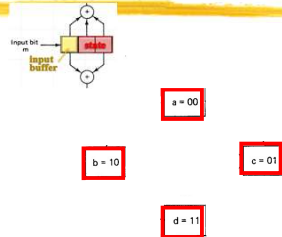


Module 5

56

State diagram

- Fix the 2^{K-1} states



Module 5

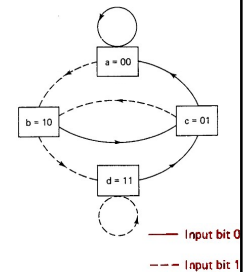
GEL7114

57

57

State diagram

- Fix the 2^{K-1} states
- Establish valid transitions



Module 5

GEL7114

58

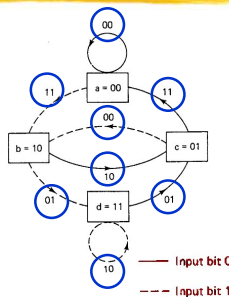
58

State diagram

- Fix the 2^{K-1} states
- Establish valid transitions
- Generate codes for each transition

$$g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$g_2 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$$



Module 5

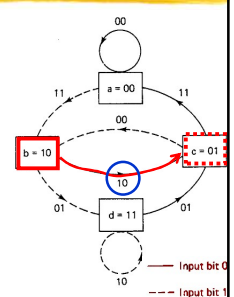
GEL7114

59

59

Example

- Let's start at the state b
 - $r = \begin{bmatrix} * & 1 & 0 \end{bmatrix}$
- Next bit 0
 - Next state $r = \begin{bmatrix} * & 0 & 1 \end{bmatrix}$
- Output
 - $r = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$
 - $u_1 = r_1 + r_2 + r_3 = 0 + 1 + 0 = 1$
 - $u_2 = r_1 + r_3 = 0 + 0 = 0$



Module 5

GEL7114

60

60

For the encoder shown in Figure 6.3, show the state changes and the resulting output codeword sequence U for the message sequence $m = 11011$, followed by $K - 1 = 2$ zeros to flush the register. Assume that the initial contents of the register are all zeros.

Solution

Input bit m_i	Register contents	State at time t_i	State at time t_{i+1}	Branch word at time t_i	
				u_1	u_2
—	0 0 0	0 0	0 0	—	—
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1

state t_i
state t_{i+1}

Output sequence: $U = 11 \ 01 \ 01 \ 00 \ 01 \ 01 \ 11$

61

State a

Input bit m_i	Register contents	State at time t_i	State at time t_{i+1}	Branch word at time t_i	
				u_1	u_2
—	0 0 0	0 0	0 0	—	—
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1

Output sequence: $U = 11 \ 01 \ 01 \ 00 \ 01 \ 01 \ 11$

62

Input: 1

Input bit m_i	Register contents	State at time t_i	State at time t_{i+1}	Branch word at time t_i	
				u_1	u_2
—	0 0 0	0 0	0 0	—	—
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1

Output sequence: $U = 11 \ 01 \ 01 \ 00 \ 01 \ 01 \ 11$

63

Output: 11

Input bit m_i	Register contents	State at time t_i	State at time t_{i+1}	Branch word at time t_i	
				u_1	u_2
—	0 0 0	0 0	0 0	—	—
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1

Output sequence: $U = 11 \ 01 \ 01 \ 00 \ 01 \ 01 \ 11$

64

State: $b = 10$

Input bit m_i	Register contents	State at time t_i	State at time t_{i+1}	Branch word at time t_i	
				u_1	u_2
—	0 0 0	0 0	0 0	—	—
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1

Output sequence: $U = 11 \ 01 \ 01 \ 00 \ 01 \ 01 \ 11$

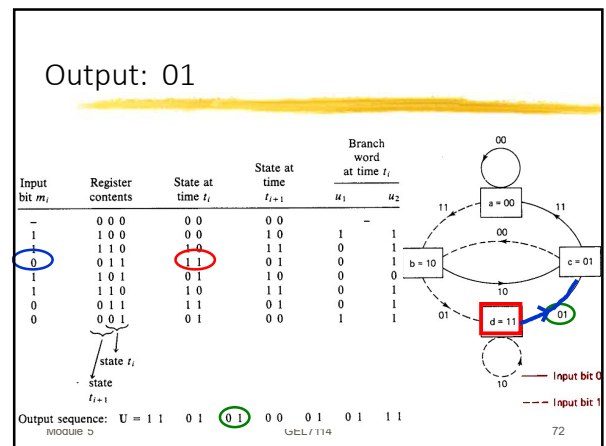
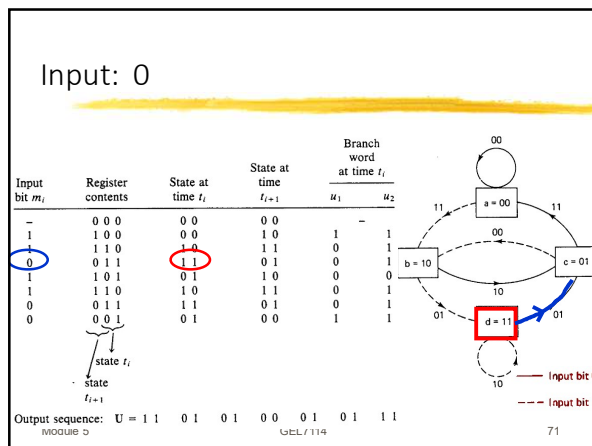
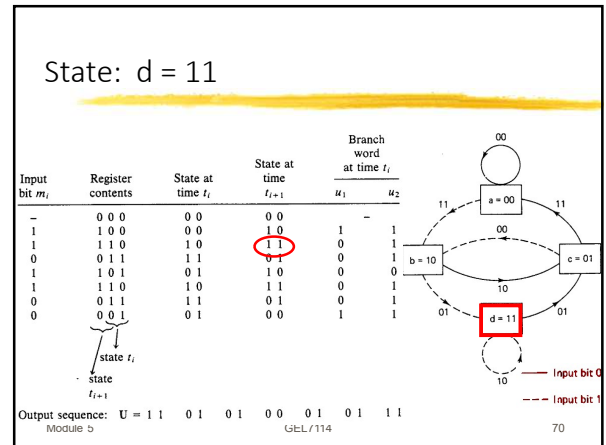
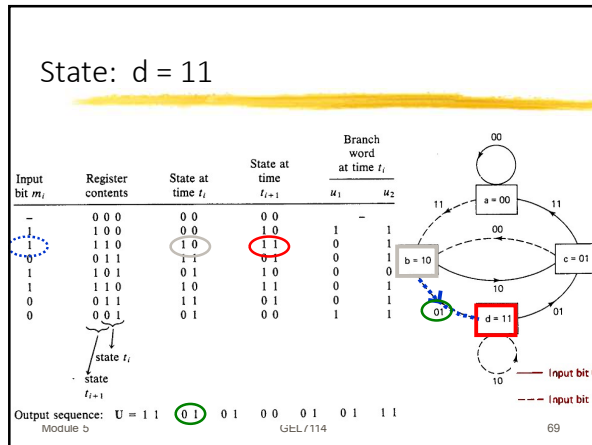
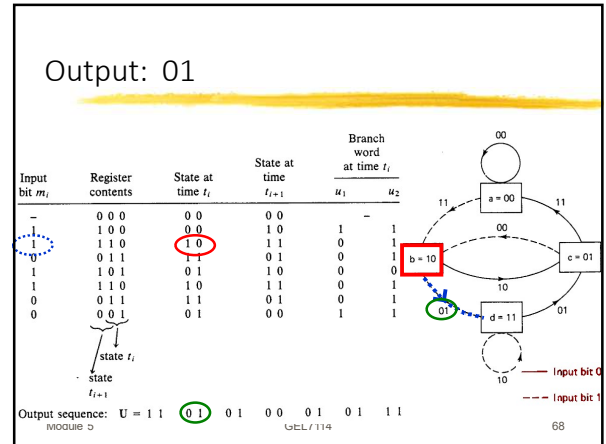
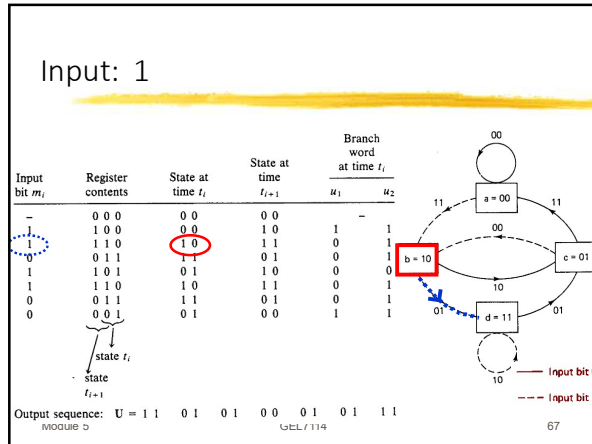
65

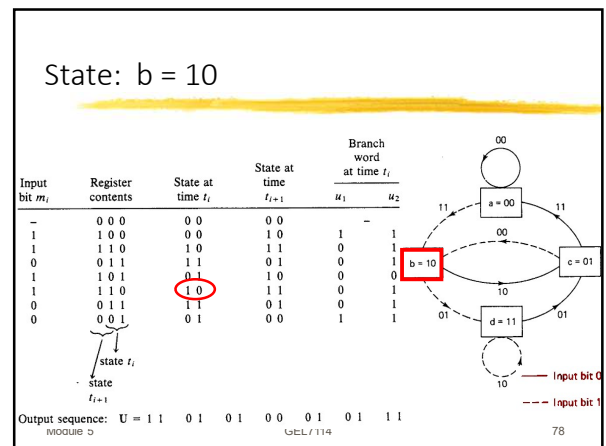
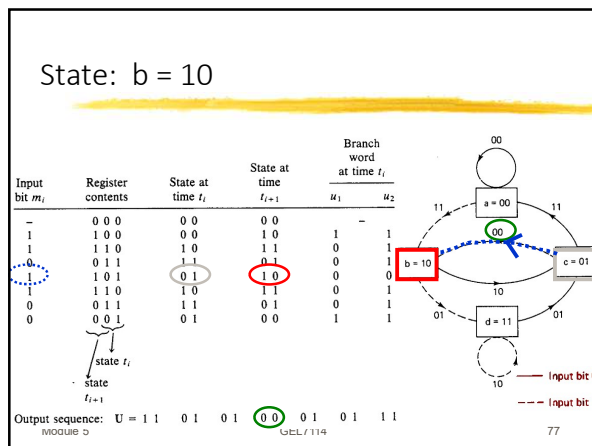
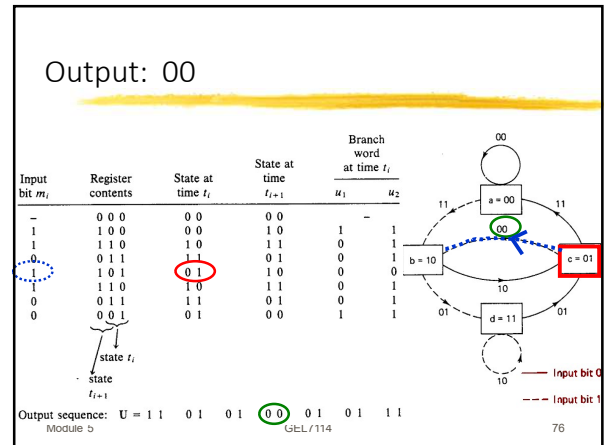
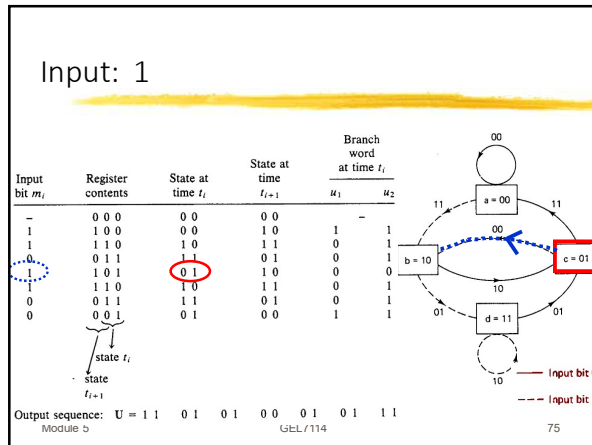
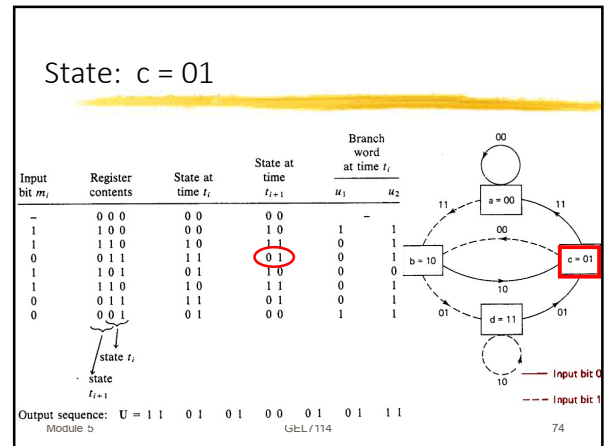
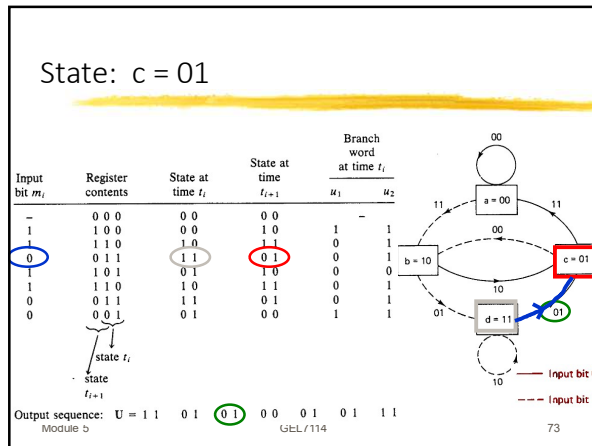
State: $b = 10$

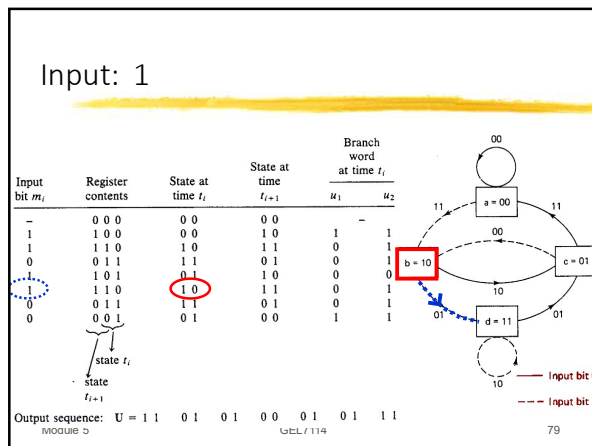
Input bit m_i	Register contents	State at time t_i	State at time t_{i+1}	Branch word at time t_i	
				u_1	u_2
—	0 0 0	0 0	0 0	—	—
1	1 0 0	0 0	1 0	1	1
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
1	1 0 1	0 1	1 0	0	0
1	1 1 0	1 0	1 1	0	1
0	0 1 1	1 1	0 1	0	1
0	0 0 1	0 1	0 0	1	1

Output sequence: $U = 11 \ 01 \ 01 \ 00 \ 01 \ 01 \ 11$

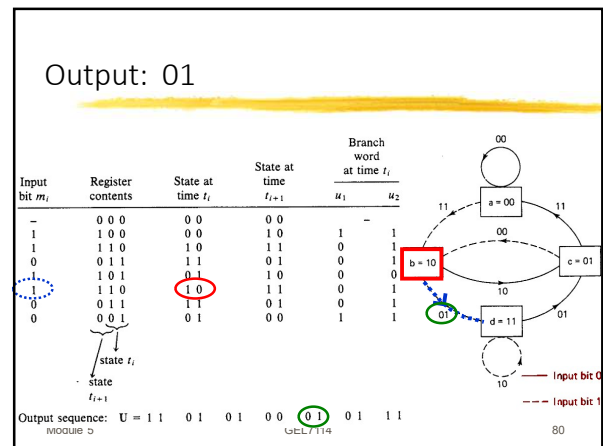
66



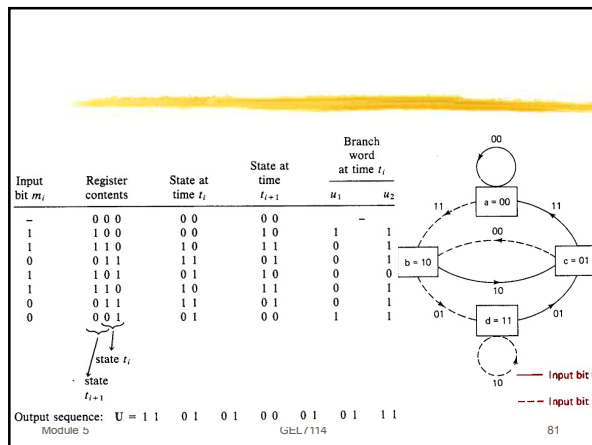




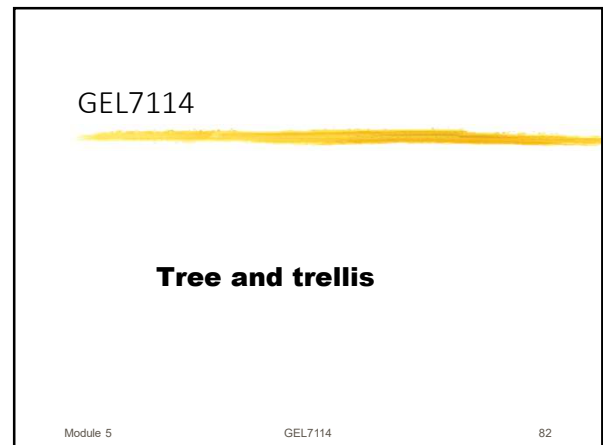
79



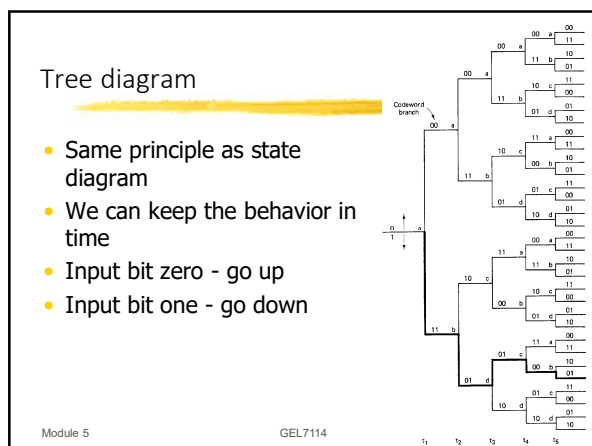
80



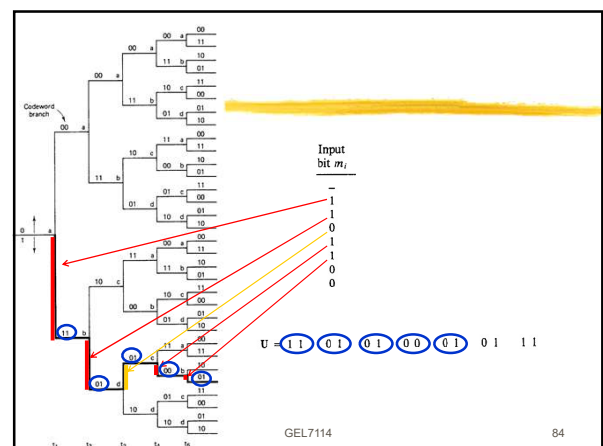
81



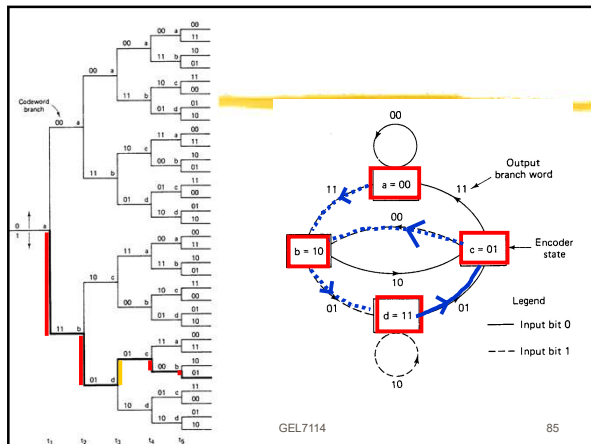
82



83



84



85

Trellis vs. tree

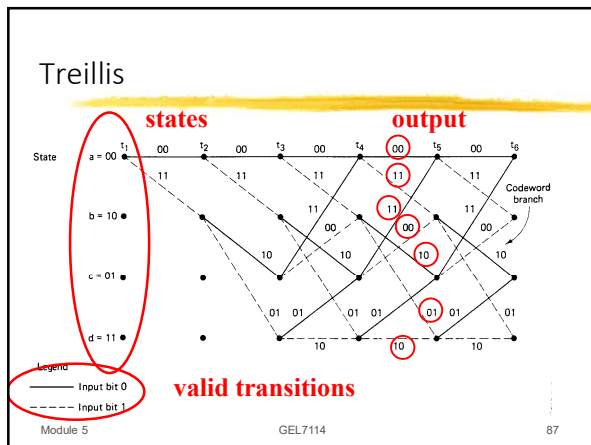
- Similar display that retains information about temporal behavior
- Trellis more compact than the tree
- Very important for decoding

Module 5

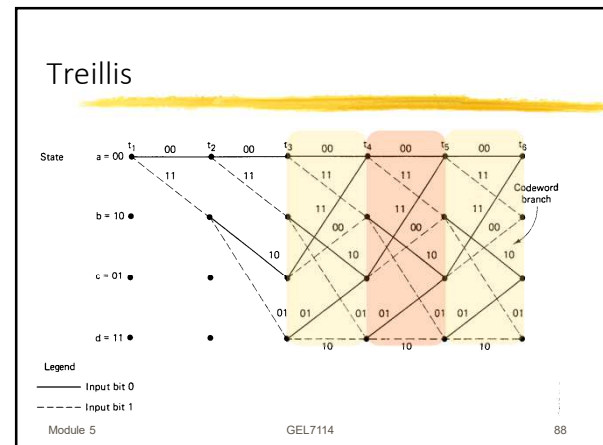
GEL7114

86

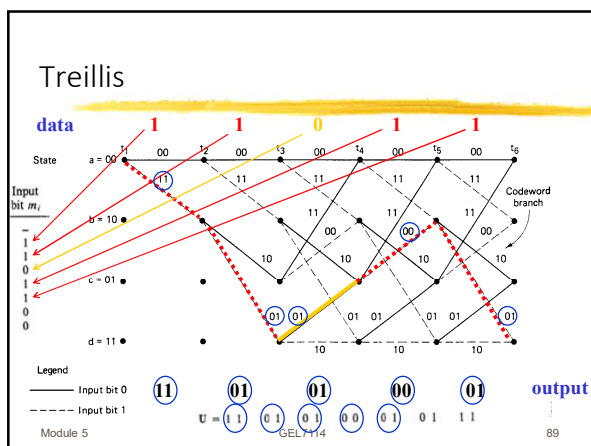
86



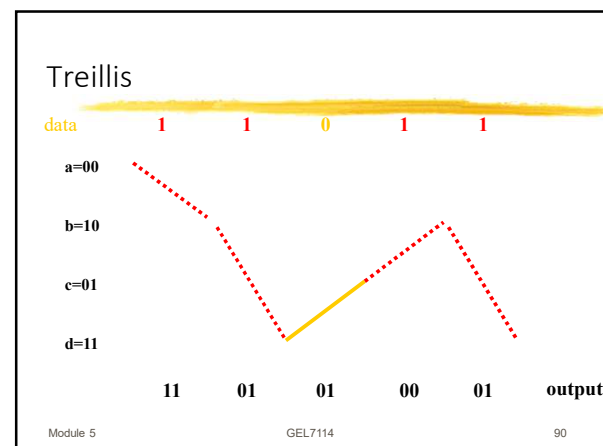
87



88



89



90

GEL7114

Decoding ...

Module 5

GEL7114

91

91

Topics covered

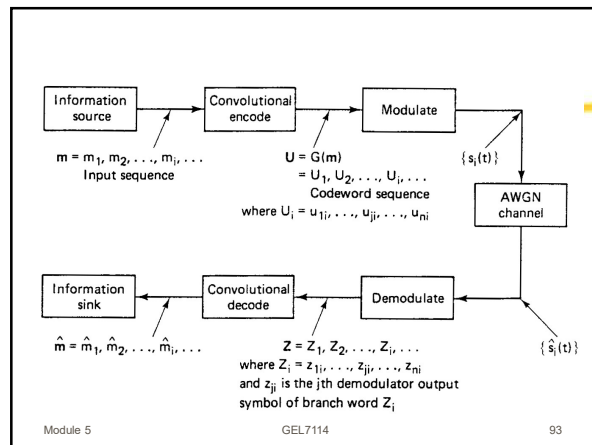
- Convolutional encoding (Sec. 7.1)
- Representation of encoders (Sec. 7.2)
- The decoding algorithm (Sec. 7.3)
- Properties of convolutional codes (Sec. 7.4)

Module 5

GEL7114

92

92



Module 5

GEL7114

93

93

Decisions

- **Hard decisions**
 - decisions coming from a finite set
 - e.g. binary 1 or -1
 - no information about confidence in the decision ...
- **Soft decisions**
 - Continuous decisions
 - For binary modulation .3 or -.9 instead of 1, -1
 - Keep information about confidence in the decision ...

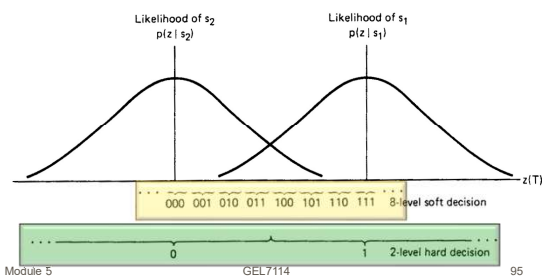
Module 5

GEL7114

94

94

Soft v. Hard Decisions



Module 5

GEL7114

95

95

Two metrics $dist(z_i, u_i)$

- **Hamming distance**
 - For hard decisions
 - $dist(z_i, u_i) = \# \text{ of different bits}$
- **Euclidian distance**
 - For soft decisions
 - $dist(z_i, u_i) = z_i^T \cdot u_i = \sqrt{(z_{i,1} - u_{i,1})^2 + \dots + (z_{i,n} - u_{i,n})^2}$

Module 5

GEL7114

96

96

Binary Symmetric Channel

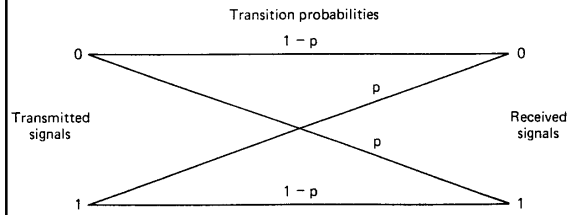


Figure 6.9 Binary symmetric channel (hard-decision channel).

Module 5

GEL7114

97

97

Decoding

- Correlation introduced between the code words
- Decoder must exploit this correlation
- Maximum Likely Detector (ML) is a **Maximum likelihood sequence estimation (MLSE)**
- Effective algorithm for MLSE: Viterbi

Module 5

GEL7114

98

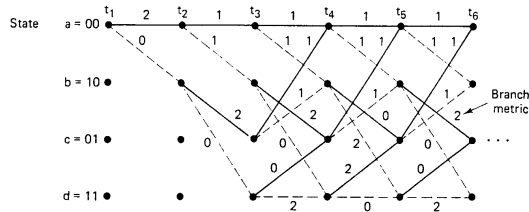
98

Branch metrics

Input data sequence m : 1 1 0 1 1 ...

Transmitted codeword U : 11 01 01 00 01 ...

Received sequence Z : 11 01 01 10 01 ...

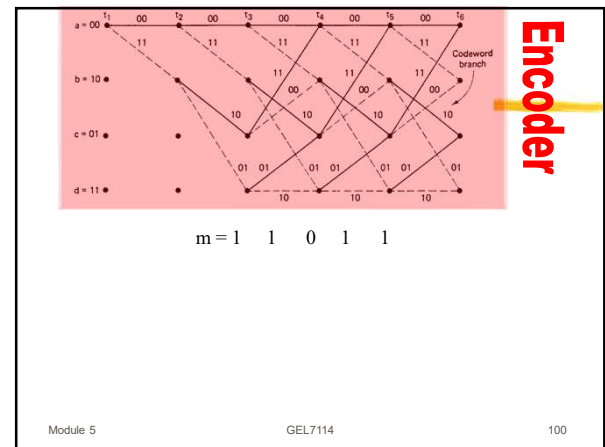


Module 5

GEL7114

100

99

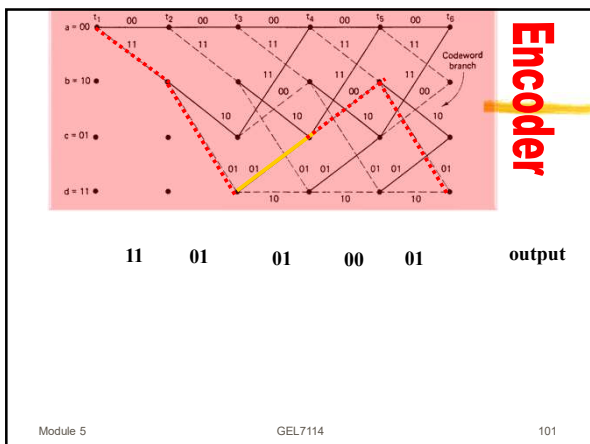


Module 5

GEL7114

100

100

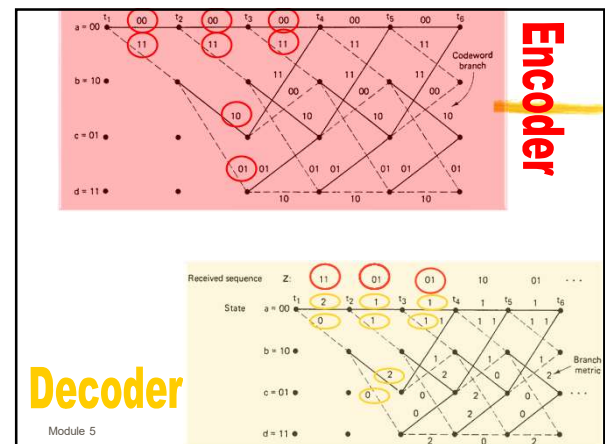


Module 5

GEL7114

101

101

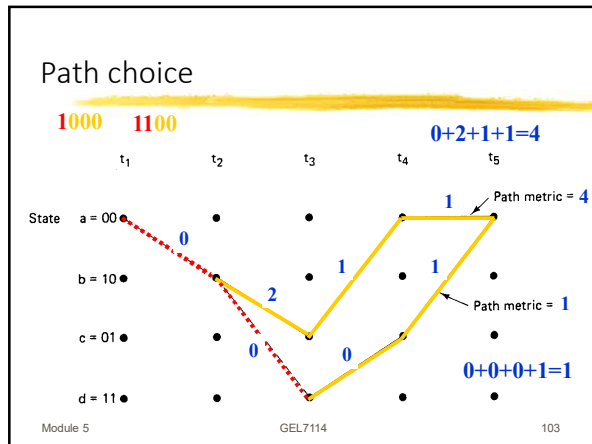


Module 5

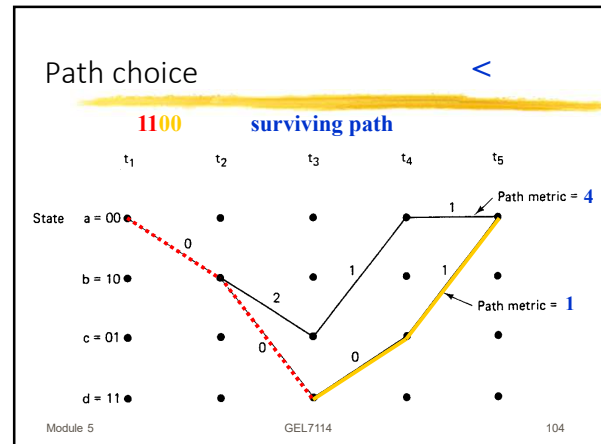
GEL7114

102

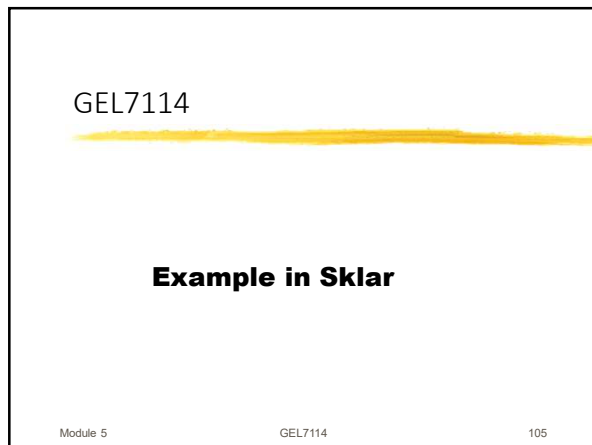
102



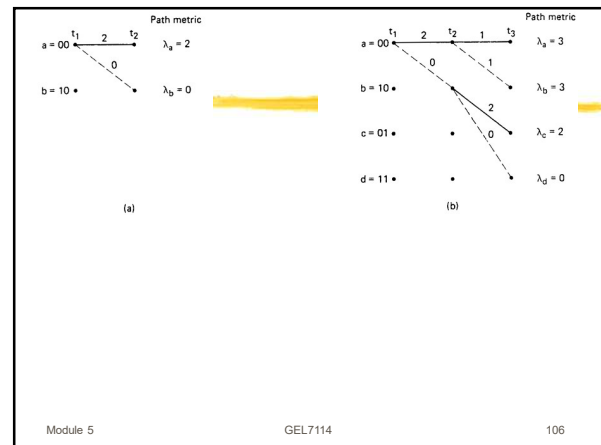
103



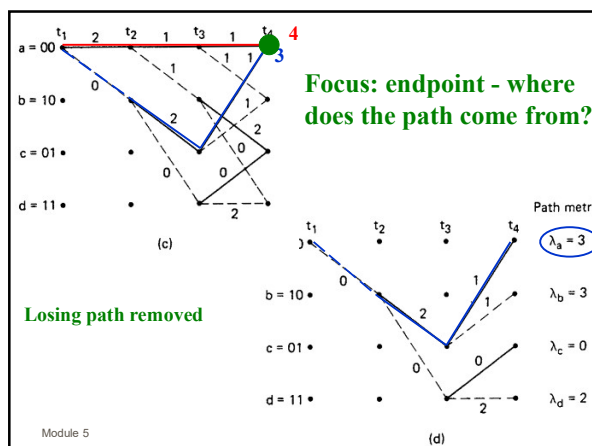
104



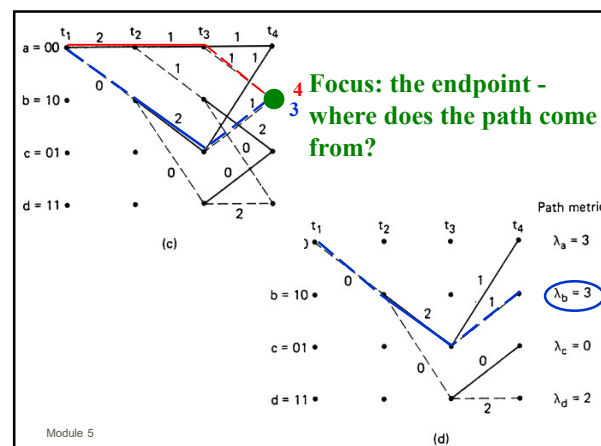
105



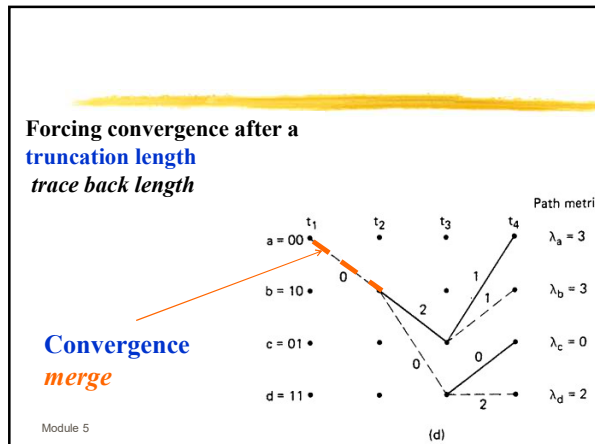
106



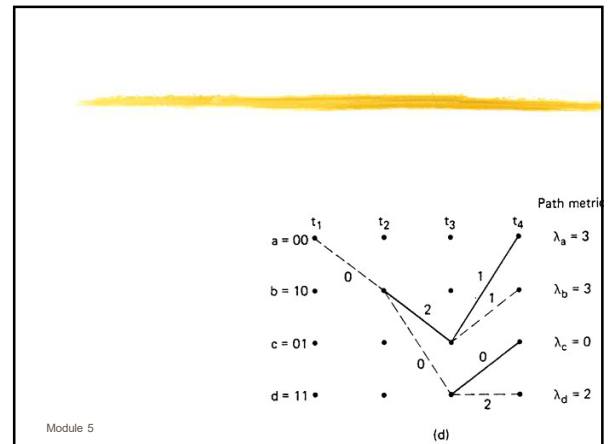
107



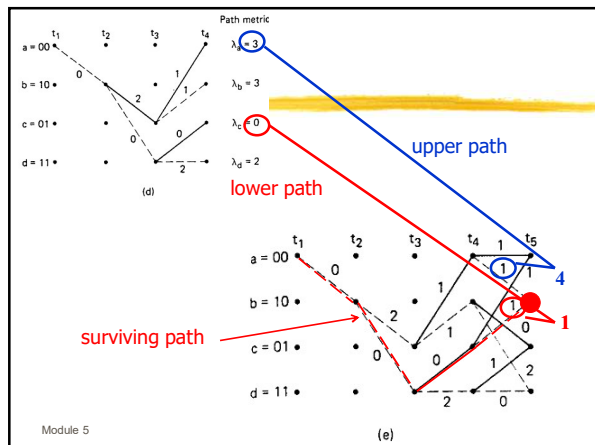
108



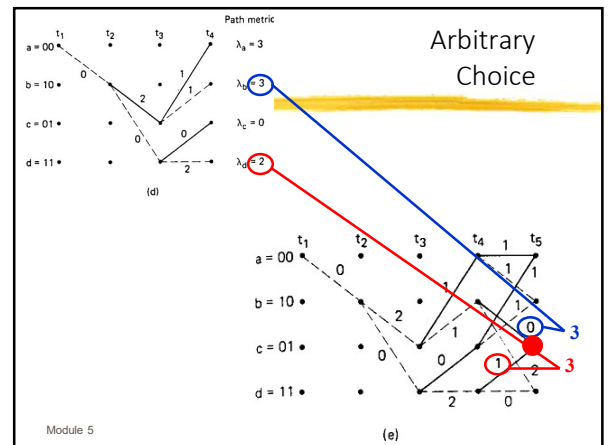
109



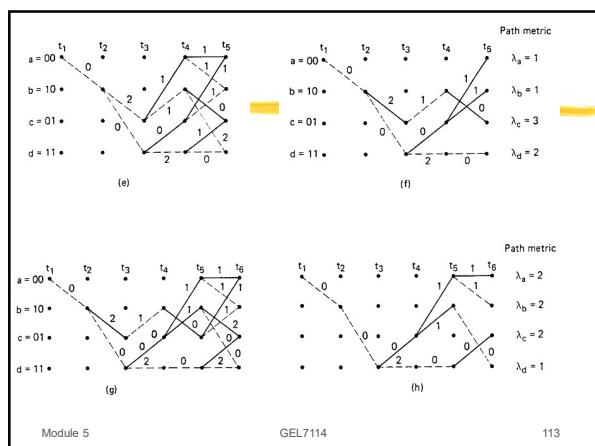
110



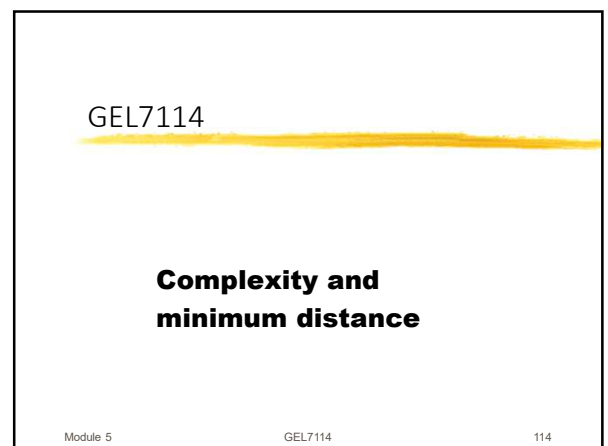
111



112



113



114

Complexity

- Operations
 - Add-compare-select (ACS)
- Memory
 - h – maximum length of a path
 - force a convergence (*merge*)
 - typically 4 or 5 $\times$ the constraint length K
 - $h2^{K-1}$
 - path length $\times$ # of states
 - bit sequence for each path
 - 2^{K-1} path metrics

Module 5

GEL7114

115

115

Final exam 2006

Considérons l'algorithme de Viterbi pour décoder un code convolutif. Supposons que la longueur de contrainte est K , qu'il y a un bit ($k=1$) qui entre dans les K registres à chaque intervalle, qu'il y a n bits de parité, que nous utilisons des décisions fermes, et que nous avons choisi de forcer une décision après $h=5K$ bits.

A. (15 points) Quelle information doit être sauvegardée pour chaque intervalle de bit et quelle est la quantité d'information à sauvegarder?

Module 5

GEL7114

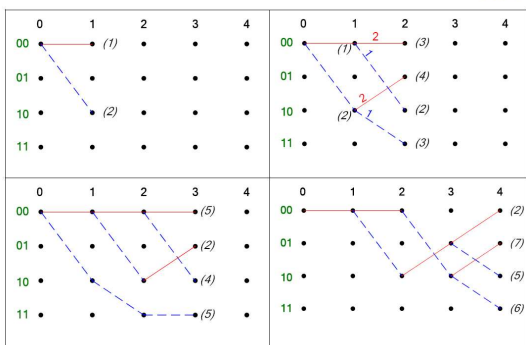
116

116

B. (10 points) Donnez l'information sauvegardée pour les quatre intervalles de bits illustrés dans la table suivante.

Donnée 0 : ———

Donnée 1 : - - - -



17

117

$K=3$ $2^{3-1}=4$ états $h=5 \times 3=15$

Chaque état A : - - - - - 01 dist état A : 2

$b=1$

B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

$b=2$

A	-	-	-	-	-	00	-	-	-	-	-	-	-	-	-	-	-	-	-
B	-	-	-	-	-	10	-	-	-	-	-	-	-	-	-	-	-	-	-
C	-	-	-	-	-	01	-	-	-	-	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	11	-	-	-	-	-	-	-	-	-	-	-	-	-

$b=3$

A	-	-	-	-	-	-	000	-	-	-	-	-	-	-	-	-	-	-	-
B	-	-	-	-	-	-	010	-	-	-	-	-	-	-	-	-	-	-	-
C	-	-	-	-	-	-	001	-	-	-	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	111	-	-	-	-	-	-	-	-	-	-	-	-

$b=4$

A	-	-	-	-	-	-	-	0100	-	-	-	-	-	-	-	-	-	-	-
B	-	-	-	-	-	-	-	0010	-	-	-	-	-	-	-	-	-	-	-
C	-	-	-	-	-	-	-	0101	-	-	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	0011	-	-	-	-	-	-	-	-	-	-	-

118

118

Distance between code words

- We are still talking about codeword sequences for convolutional codes.
- Consider all possible sequences, and the distance between each pair
- Minimum distance
 - minimum among all pairs of sequences
- Linear code
 - minimum distance between zero paths and other sequences is equivalent

Module 5

GEL7114

119

119

Minimum distance

- Procedure
 1. Starts in state a
 2. Finish in state a
 3. Smallest distance d_f
 - length = free distance
- The minimum distance is a measure of the code's ability to correct an erroneous path

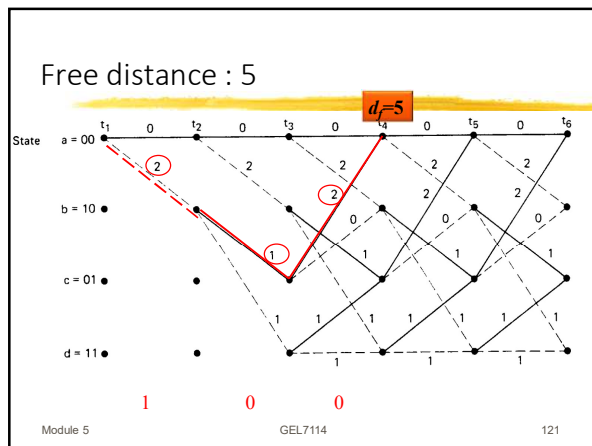
$$t = \left\lfloor \frac{d_f - 1}{2} \right\rfloor$$

Module 5

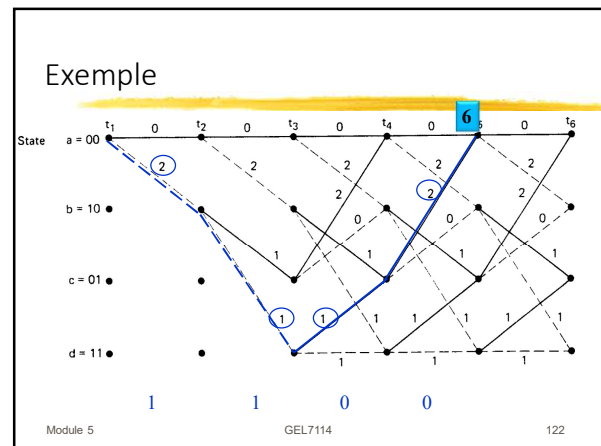
GEL7114

120

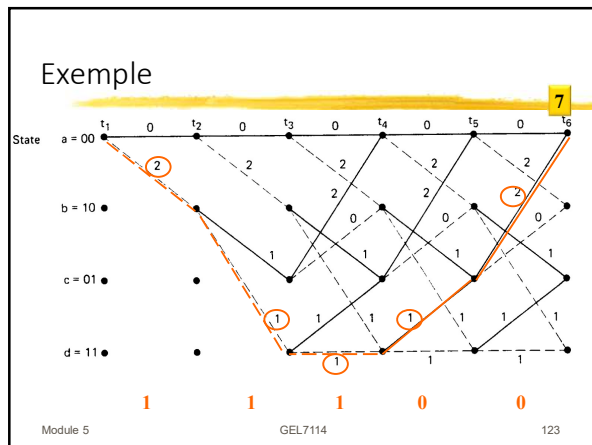
120



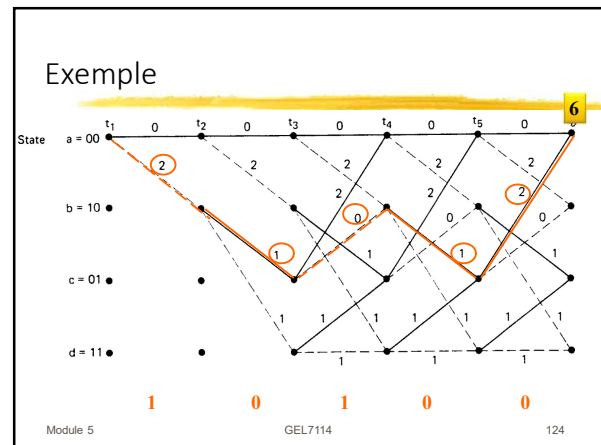
121



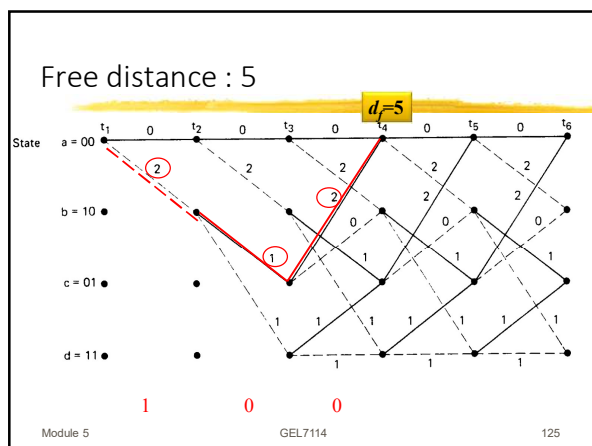
122



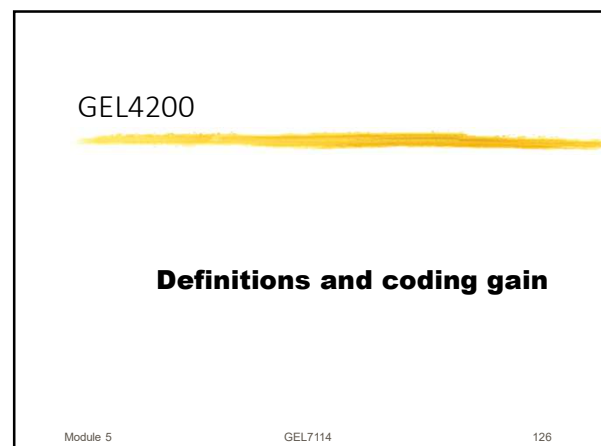
123



124



125



126

Definitions

- Systematic code
 - Data readable in code
- Cyclical code
 - Each code word is a cyclic rotation of another code word
- Catastrophic code
 - Code where finite # of errors in the code can generate an infinite # of errors in the data
- Perfect code
 - $t = \frac{d_f - 1}{2}$

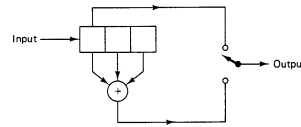
Module 5

GEL7114

127

127

Systematic codes



- Less efficient than non-systematic codes for convolutional codes

TABLE 6.1 Comparison of Systematic and Nonsystematic Free Distance, Rate $\frac{1}{2}$

Constraint length	Free distance systematic	Free distance nonsystematic
2	3	3
3	4	5
4	4	6
5	5	7
6	6	8
7	6	10
8	7	10

Source: A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*, McGraw-Hill Book Company, New York, 1979, p. 251.

Module 5

GEL71

128

Possible gain

$$10 \log_{10} r d_f$$

TABLE 6.2 Coding Gain Upper Bounds for Some Convolutional Codes

Rate $\frac{1}{2}$ codes			Rate $\frac{1}{3}$ codes		
K	d_f	Upper bound (dB)	K	d_f	Upper bound (dB)
3	5	3.97	3	8	4.26
4	6	4.76	4	10	5.23
5	7	5.43	5	12	6.02
6	8	6.00	6	13	6.37
7	10	6.99	7	15	6.99
8	10	6.99	8	16	7.27
9	12	7.78	9	18	7.78

Source: V. K. Bhargava, D. Haccoun, R. Matyas, and P. Nuspi, *Digital Communications by Satellite*, John Wiley & Sons, Inc., New York, 1981.

Module 5

GEL7114

129

129

Gain for given SNR

TABLE 6.3 Basic Coding Gain (dB) for Soft Decision Viterbi Decoding AWGN

Uncoded E_b/N_0 (dB)	P_B	code rate $\frac{1}{2}$							
		K	7	8	5	6	7	6	8
6.8	10^{-3}	4.2	4.4	3.3	3.5	3.8	2.9	3.1	2.6
9.6	10^{-5}	5.7	5.9	4.3	4.6	5.1	4.2	4.6	3.6
11.3	10^{-7}	6.2	6.5	4.9	5.3	5.8	4.7	5.2	3.9
Upper bound		7.0	7.3	5.4	6.0	7.0	5.2	6.7	4.8

Source: I. M. Jacobs, "Practical Applications of Coding," *IEEE Trans. Inf. Theory*, vol. IT20, May 1974, pp. 305-310.

Module 5

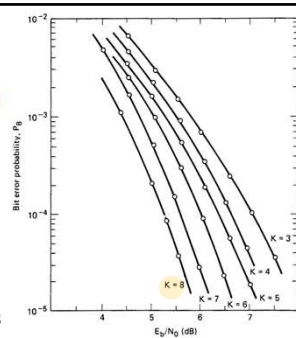
GEL7114

130

130

BER

- Fixed rate, therefore fixed bandwidth
- BER improved by adding complexity in the decoder
 - Memory
 - Number of calculations per symbol



BER vs. E_b/N_0 , rate $\frac{1}{2}$, coherent BPSK, BSC, Viterbi hard decision, 32-bit path memory

Module 5

GEL7114

131

131

Optimal codes

- Exhaustive research
- Eliminate catastrophic codes
- Minimize
 - Free distance d_f
 - # Paths @ d_f
 - # Paths @ d_f+1
 - # Paths @ d_f+2 , etc.

TABLE 6.4 Optimum Short Constraint Length Convolutional Codes (Rate $\frac{1}{2}$ and Rate $\frac{1}{3}$)

Rate	Constraint length	Free distance	Code vector
$\frac{1}{2}$	3	5	111 101
$\frac{1}{2}$	4	6	1111 1011
$\frac{1}{2}$	5	7	10111 11001
$\frac{1}{2}$	6	8	101111 110101
$\frac{1}{2}$	7	10	1001111 1101101
$\frac{1}{2}$	8	10	10011111 1100101
$\frac{1}{2}$	9	12	11010111 100011101
$\frac{1}{2}$	3	8	111 111 101
$\frac{1}{2}$	4	10	1111 1011 1101
$\frac{1}{2}$	5	12	11111 11011 10101
$\frac{1}{2}$	6	13	101111 110101 111001
$\frac{1}{2}$	7	15	1001111 1101011 1101101
$\frac{1}{2}$	8	16	11010111 10011011 1011001

Typical values

- Decisions
 - Hard
 - Soft with 3 quantization bits
- Constraint length : $3 \leq K \leq 9$
- Code rate : $r \geq 1/3$
- Maximum path : $h \leq 5K$

Module 5

GEL7114

133

133

Chapter 8

- Reed-Solomon code
 - Cyclic code, non-binary
 - Maximum free distance
 - Very strong against « burst » noise
- Interleaving
- Concatenated codes
 - Two levels of coding
 - Example: audio CD
- Turbo codes
 - Decoding that exploits concatenated codes

Module 5

GEL7114

134

134

Shannon Capacity

Module 5

GEL7114

135

135

Shannon's Law

- The capacity of a channel with signal-to-noise ratio SNR and bandwidth W

$$C = W \log_2(1 + SNR)$$

- Ability to send C b/s with as small a P_e as you want...

Module 5

GEL7114

136

136

Alternative form

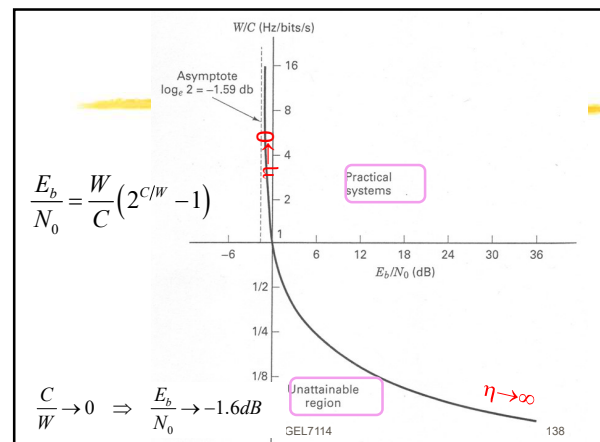
$$\begin{aligned} \left. \frac{C}{W} \right|_{C=R} &= \log_2 \left(1 + \frac{S}{WN_0} \right) \bigg|_{C=R} = \log_2 \left(1 + \frac{S}{CN_0} \frac{C}{W} \right) \bigg|_{C=R} \\ &= \log_2 \left(1 + \frac{S}{RN_0} \frac{C}{W} \right) \\ &= \log_2 \left(1 + \frac{ST_b}{N_0} \frac{C}{W} \right) = \log_2 \left(1 + \frac{E_b}{N_0} \frac{C}{W} \right) \\ \frac{E_b}{N_0} &= \frac{W}{C} (2^{C/W} - 1) \quad \frac{C}{W} \rightarrow 0 \Rightarrow \frac{E_b}{N_0} \rightarrow -1.6\text{dB} \end{aligned}$$

Module 5

GEL7114

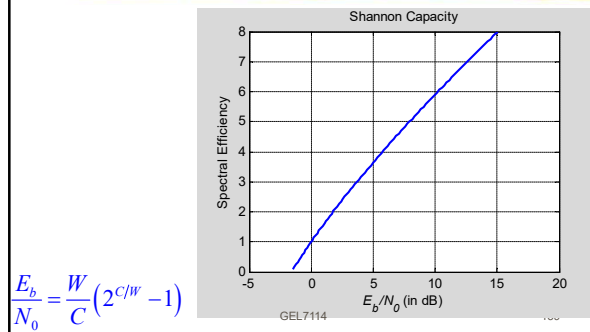
137

137



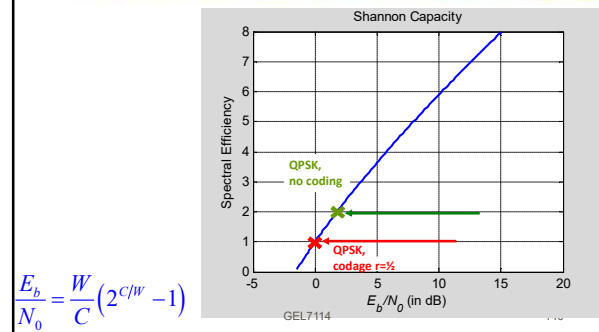
138

Linear scale



139

Example QPSK



140

Example – Digital Video Broadcast

Module 5 GEL7114 141

141

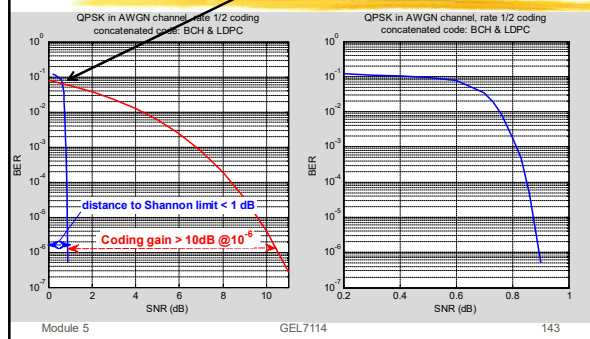
DVB-S2

- Satellite video
- Concatenated codes
 - BCH (32400,32208) & LDPC (64800,32400)
 - Interleaving
- Example Matlab demo
 - QPSK & coding rate 1/2
- Close to Shannon limit

Module 5 GEL7114 142

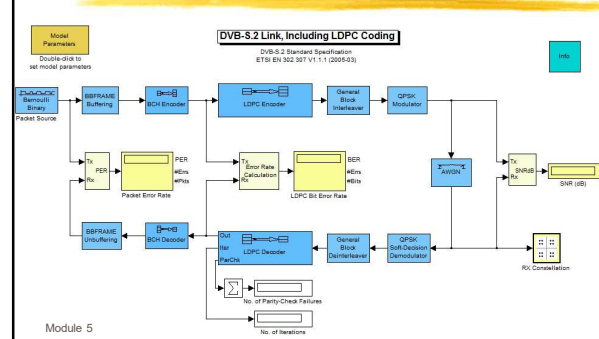
142

FEC limit very high: .05

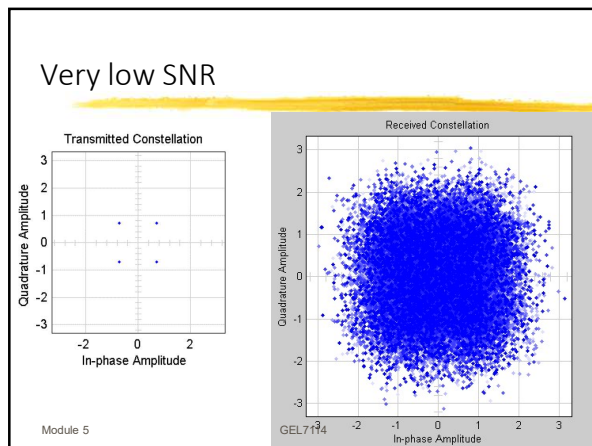


143

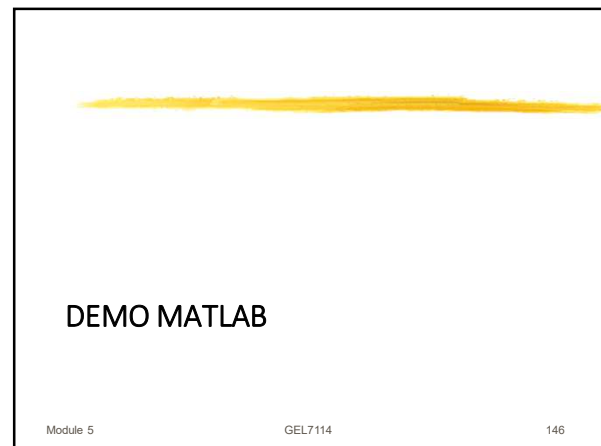
Matlab demo



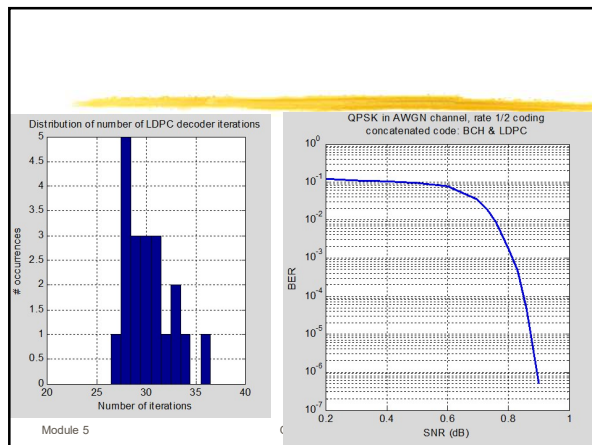
144



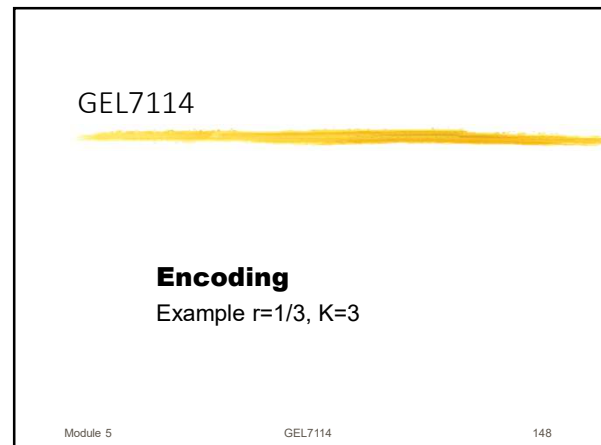
145



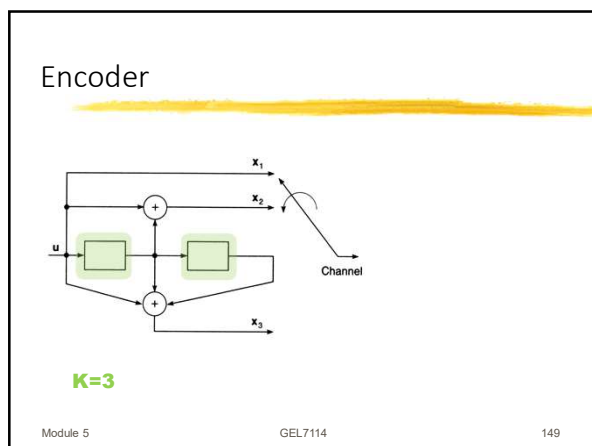
146



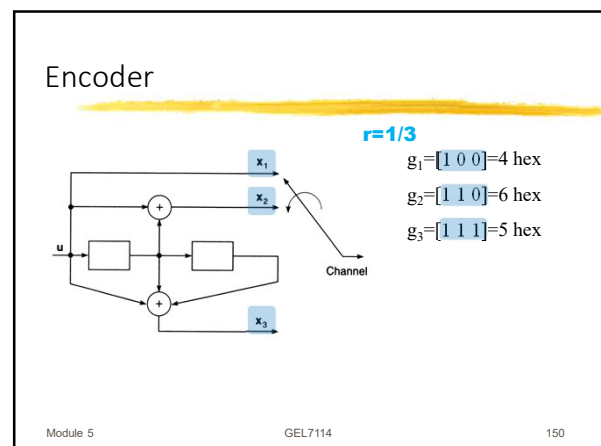
147



148

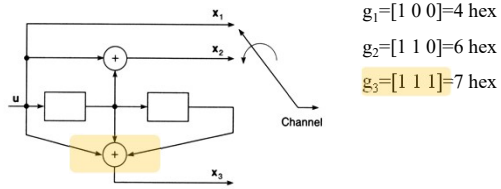


149



150

Encoder



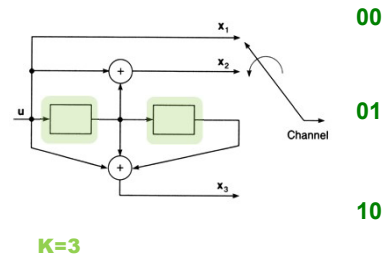
Module 5

GEL7114

151

151

Encoder



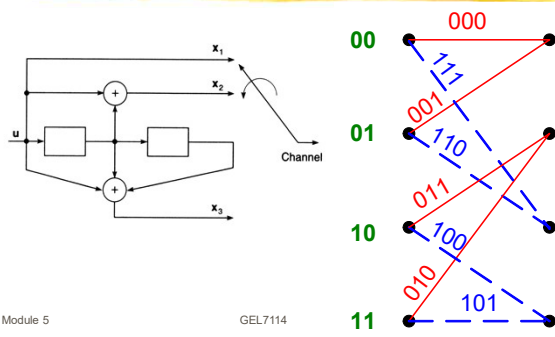
Module 5

GEL7114

152

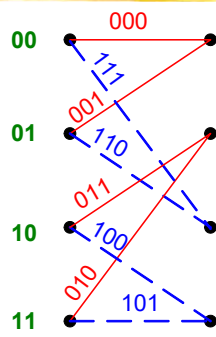
152

Encoder



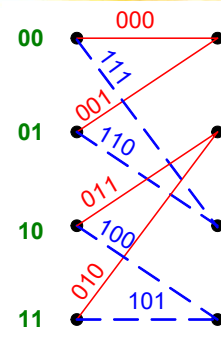
Module 5

GEL7114



153

Trellis encoder

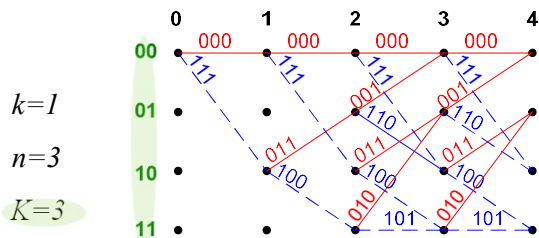


Module 5

154

154

All possible transitions



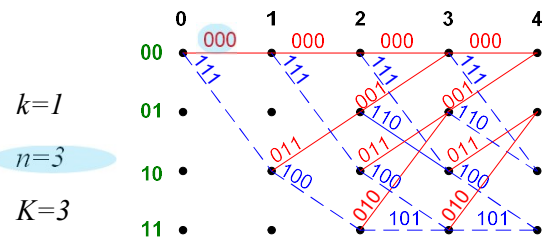
Module 5

GEL7114

155

155

All possible transitions



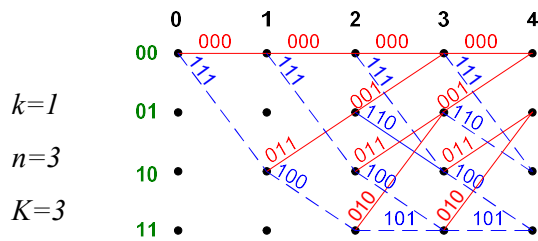
Module 5

GEL7114

156

156

Encoder



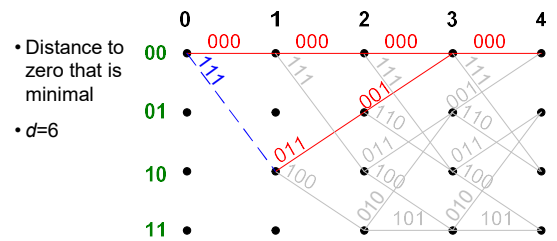
Module 5

GEL7114

157

157

Minimum distance



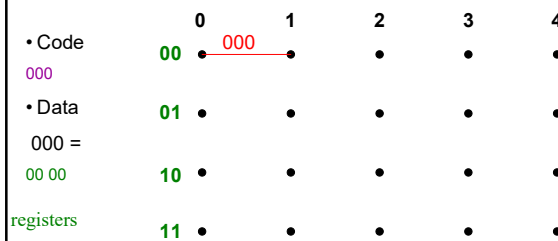
Module 5

GEL7114

158

158

Example - Encoding



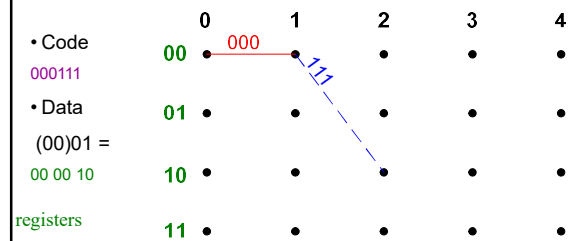
Module 5

GEL7114

159

159

Exemple - encodage



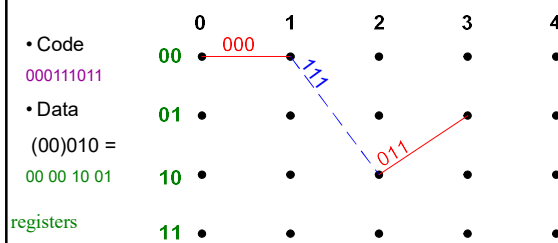
Module 5

GEL7114

160

160

Exemple - encodage



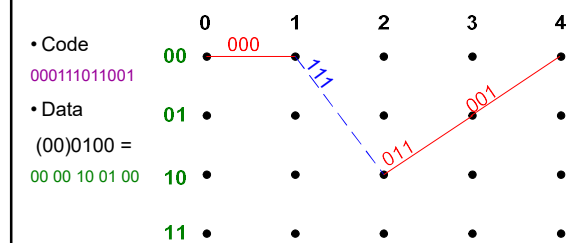
Module 5

GEL7114

161

161

Exemple - encodage

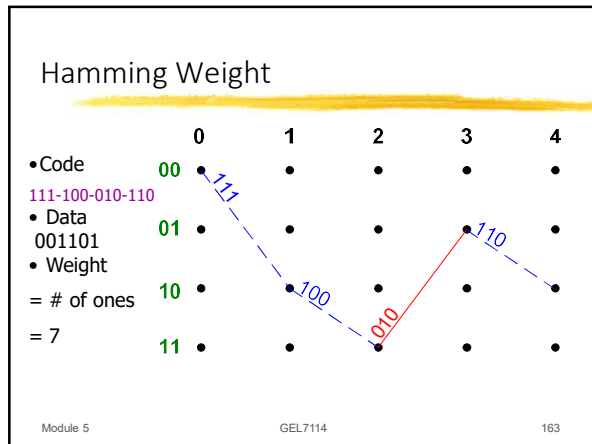


Module 5

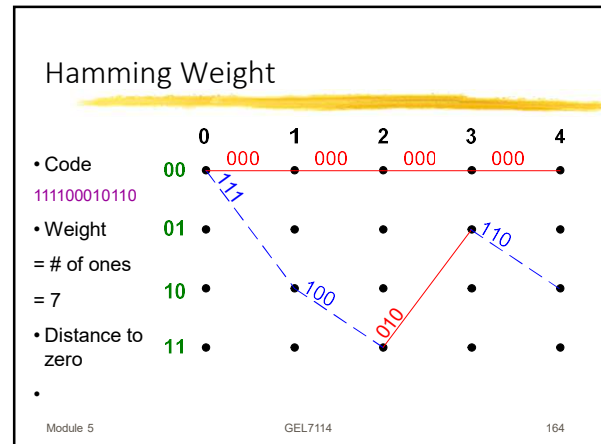
GEL7114

162

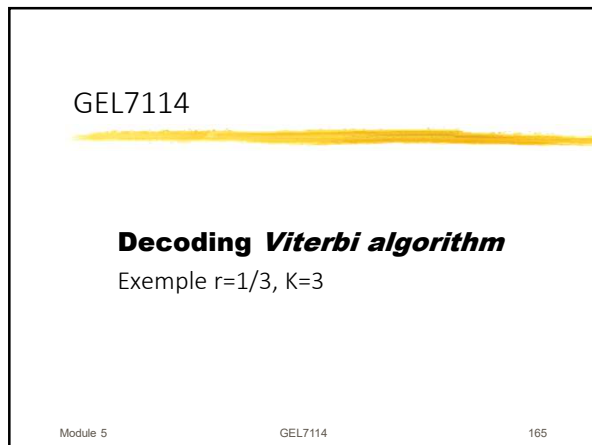
162



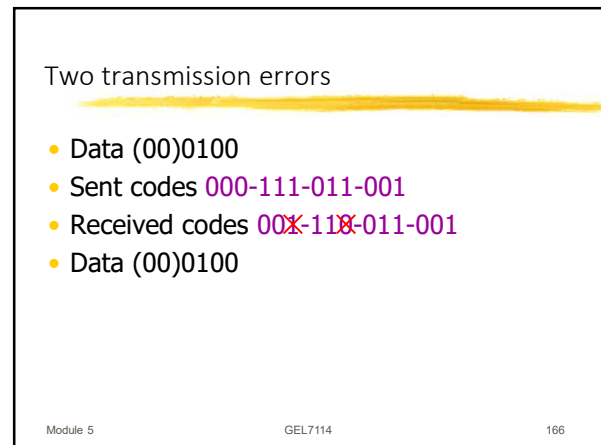
163



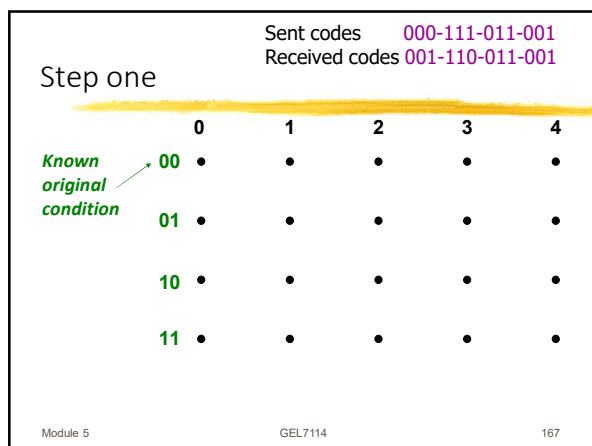
164



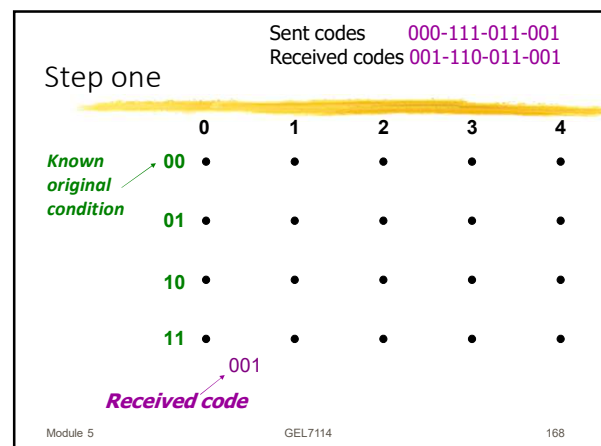
165



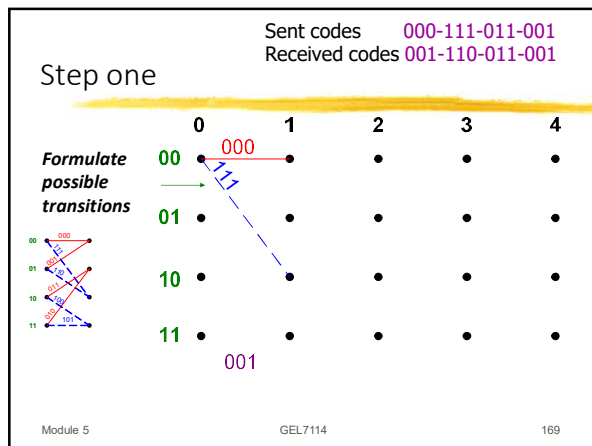
166



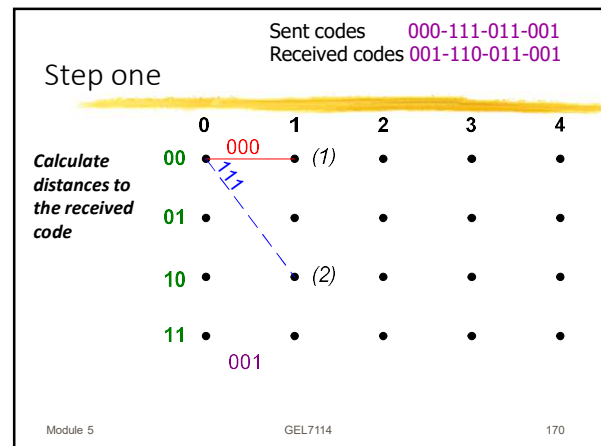
167



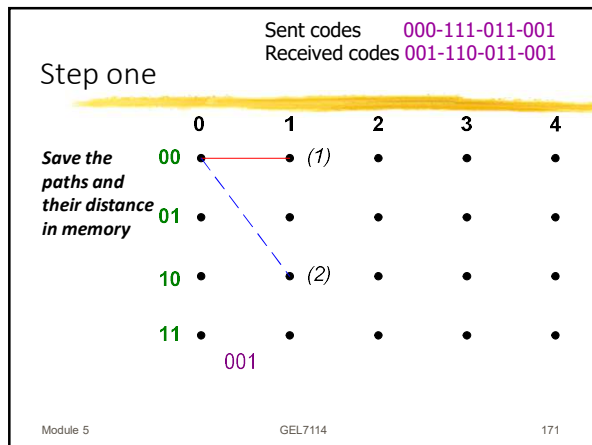
168



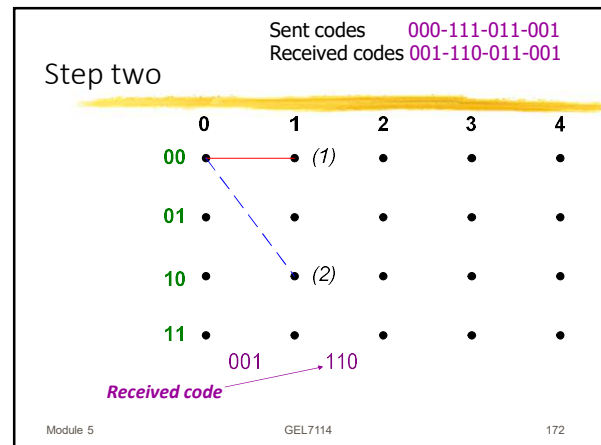
169



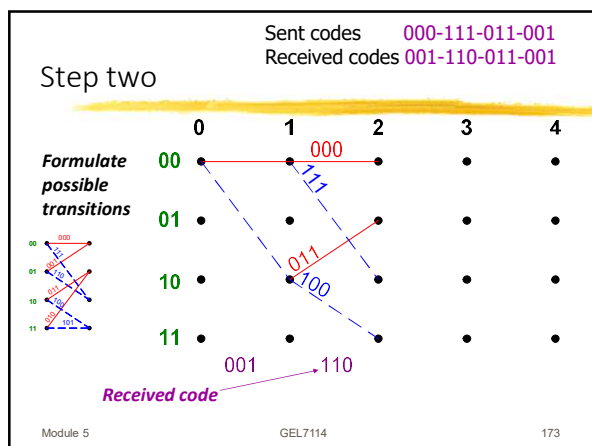
170



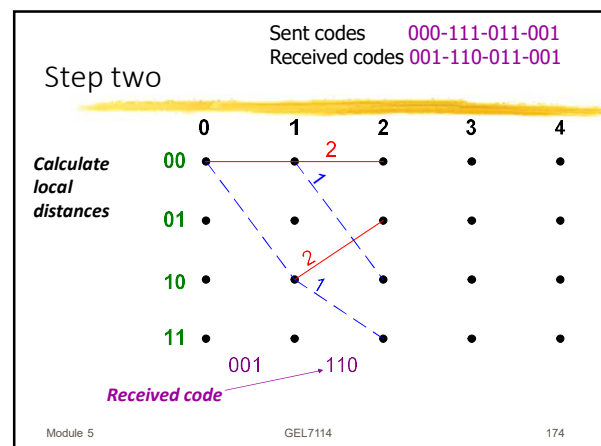
171



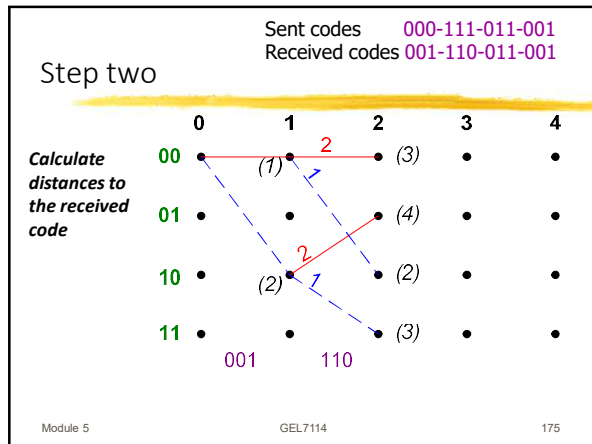
172



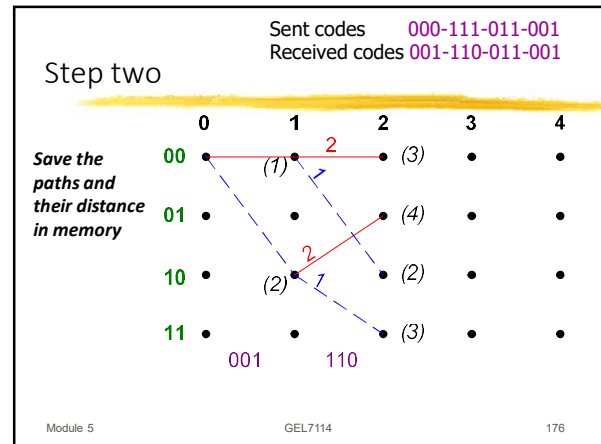
173



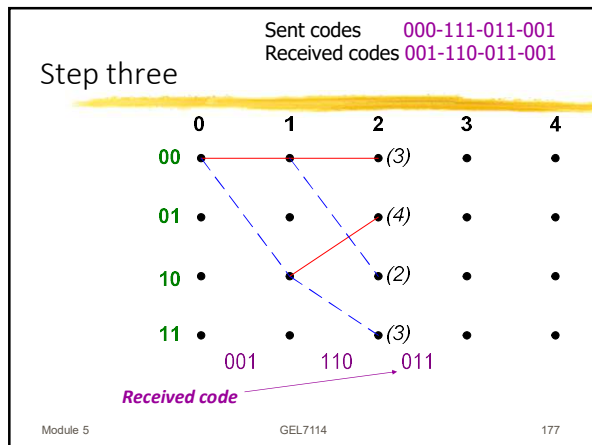
174



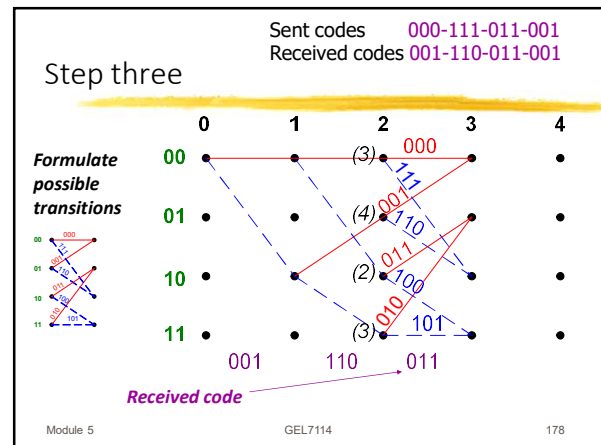
175



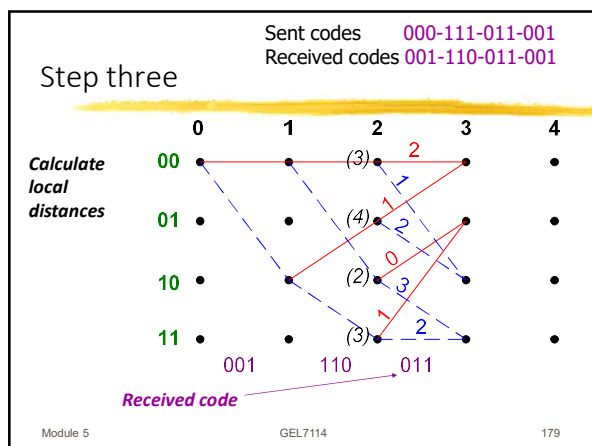
176



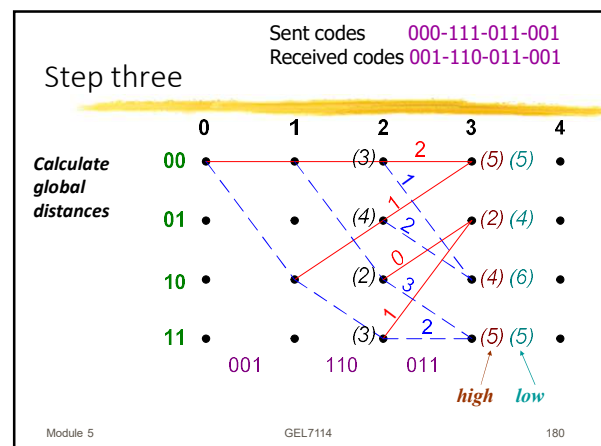
177



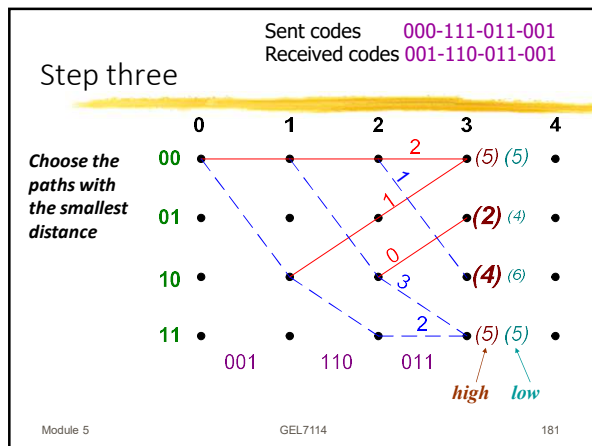
178



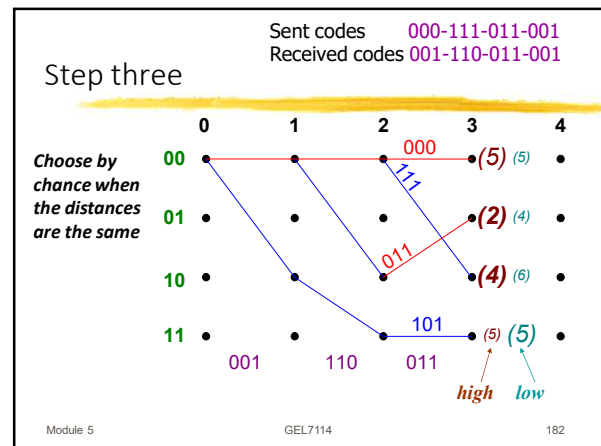
179



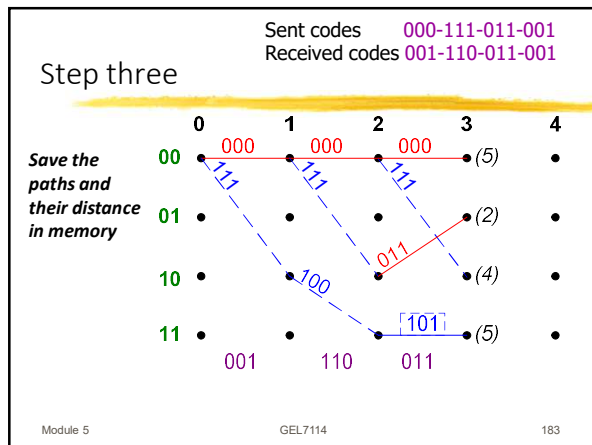
180



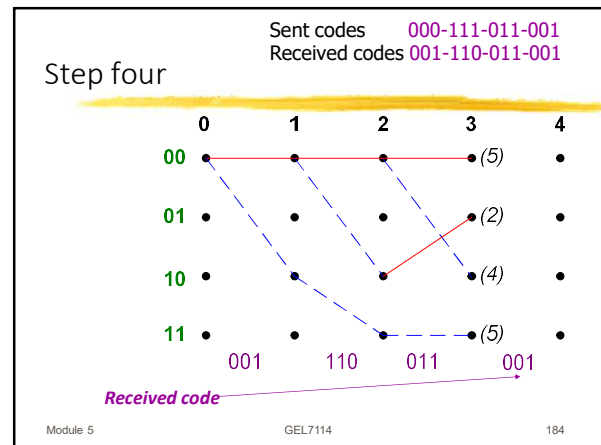
181



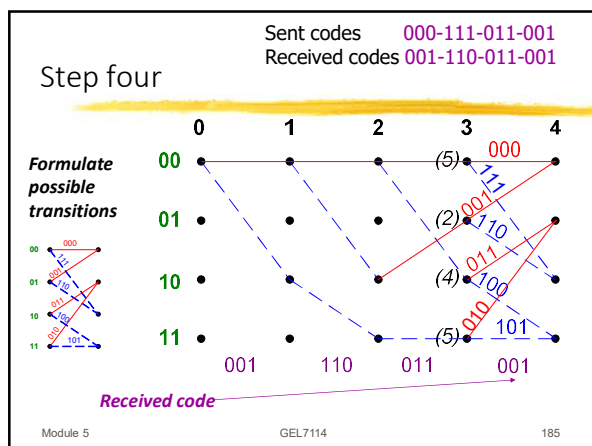
182



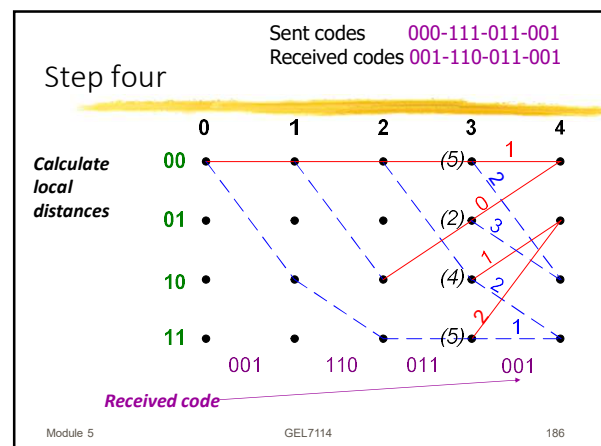
183



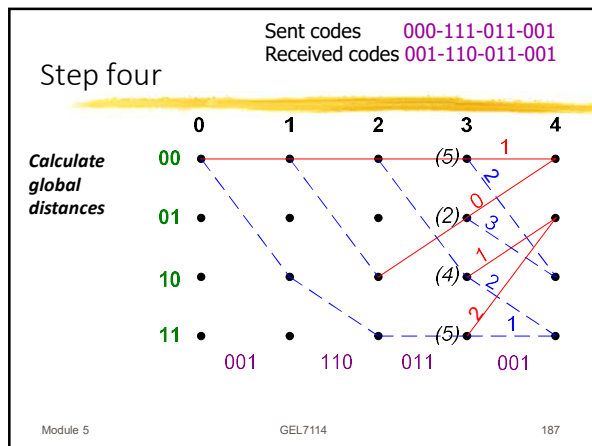
184



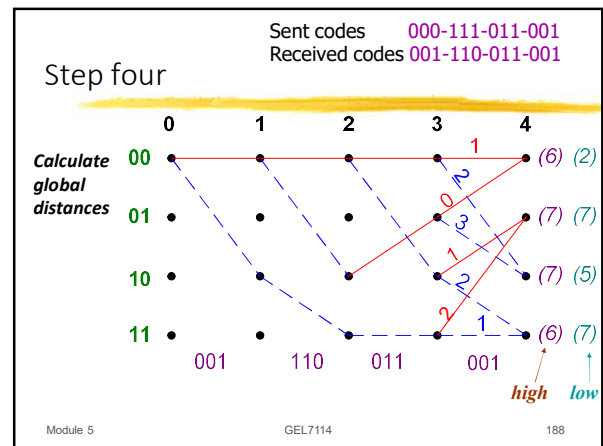
185



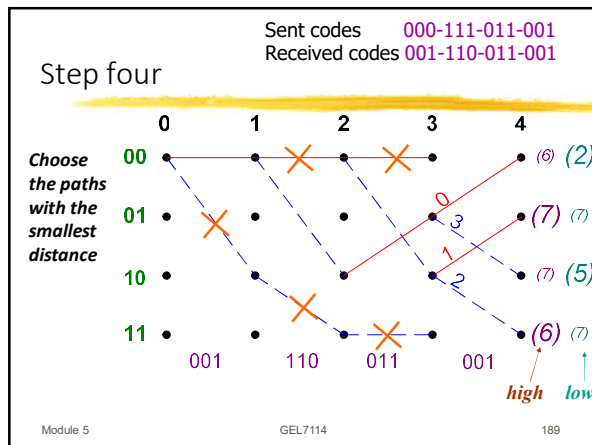
186



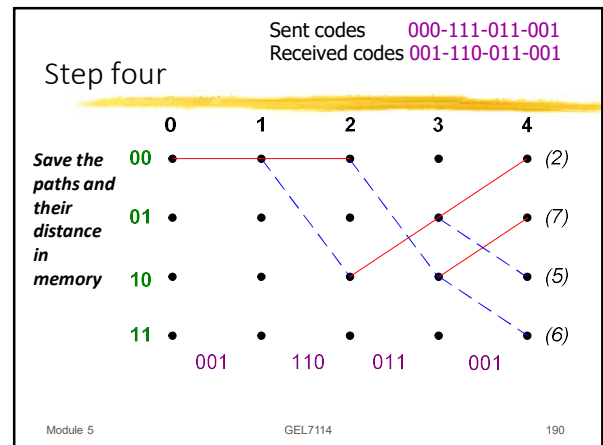
187



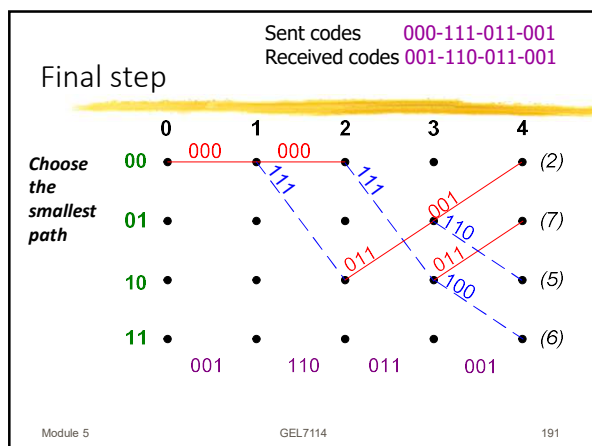
188



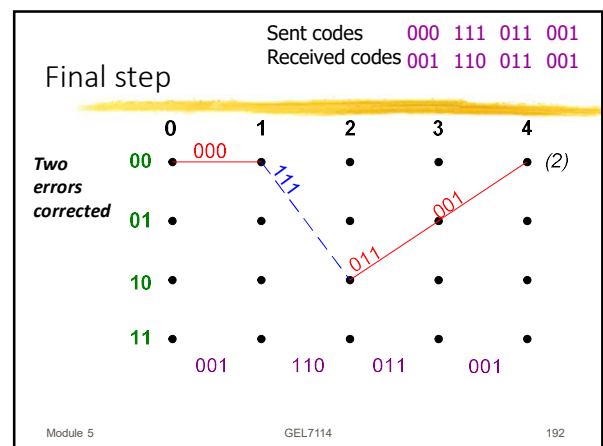
189



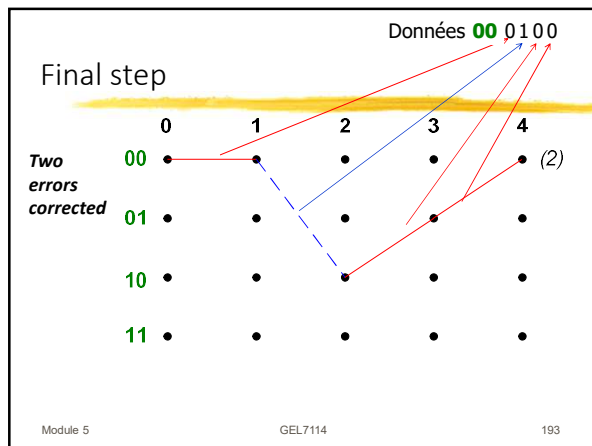
190



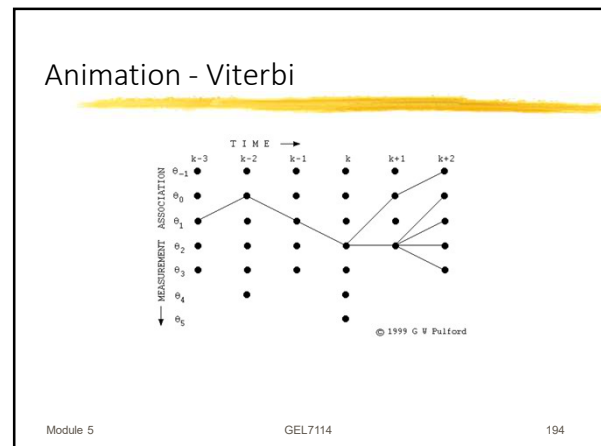
191



192



193



194