TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHÊN
KHOA ĐIỆN TỬ - ViỄN THÔNG

Bài 8: Mã hóa chập

Đặng Lê Khoa

Bộ môn Viễn thông – Mạng

Class 2

Introduction

- Trong mã hóa khối, bộ mã hóa đưa vào từng khối thông tin k-bit và phát ra từ mã n-bit $\Rightarrow$ dựa trên xử lý từng khối
- Bộ mã hóa cần phải có bộ đệm toàn bộ khối dữ liệu và phát ra từ mã
- Trong khi bit dữ liệu tới thường là nối tiếp hơn là song song, nên việc sử dụng bộ đệm là không mong muốn
- Mã hóa chập

Definitions

- Bộ mã hóa chập: một máy trạng thái hữu hạn gồm M tầng thanh ghi dịch, n bộ cộng modulo-2
- Chuỗi thông tin L-bit sinh ra một chuỗi n(L+M) bi
- Tốc độ mã:

$$r = \frac{L}{n(L+M)} \quad \text{(bits/symbol)}$$

- L>>M, vậy

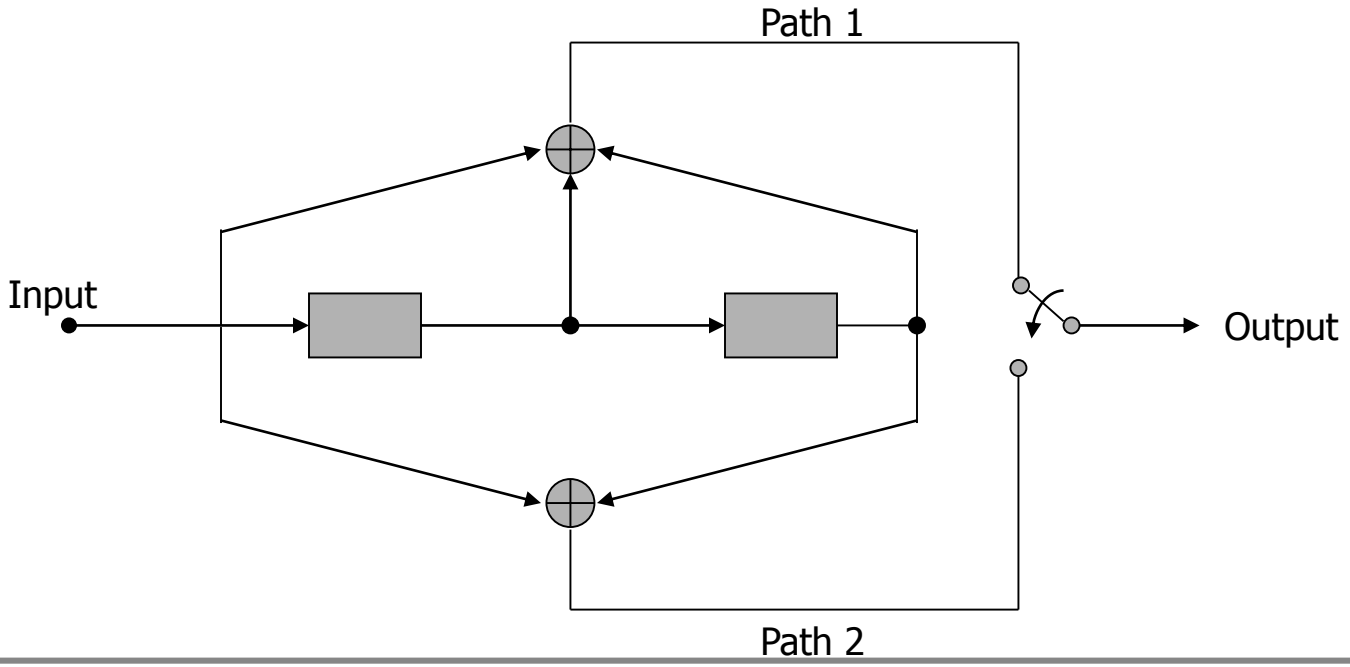
$$r = \frac{1}{n} \quad \text{(bits/symbol)}$$

Definitions

- *Chiều dài ràng buộc (Constraint length (K))*: số dịch của một bit đơn ảnh hưởng đến ngõ ra
- Thanh ghi dịch M tầng: cần M+1 dịch cho một bit từng lúc bắt đầu đến tới lúc ra khỏi mạch
- K=M+1

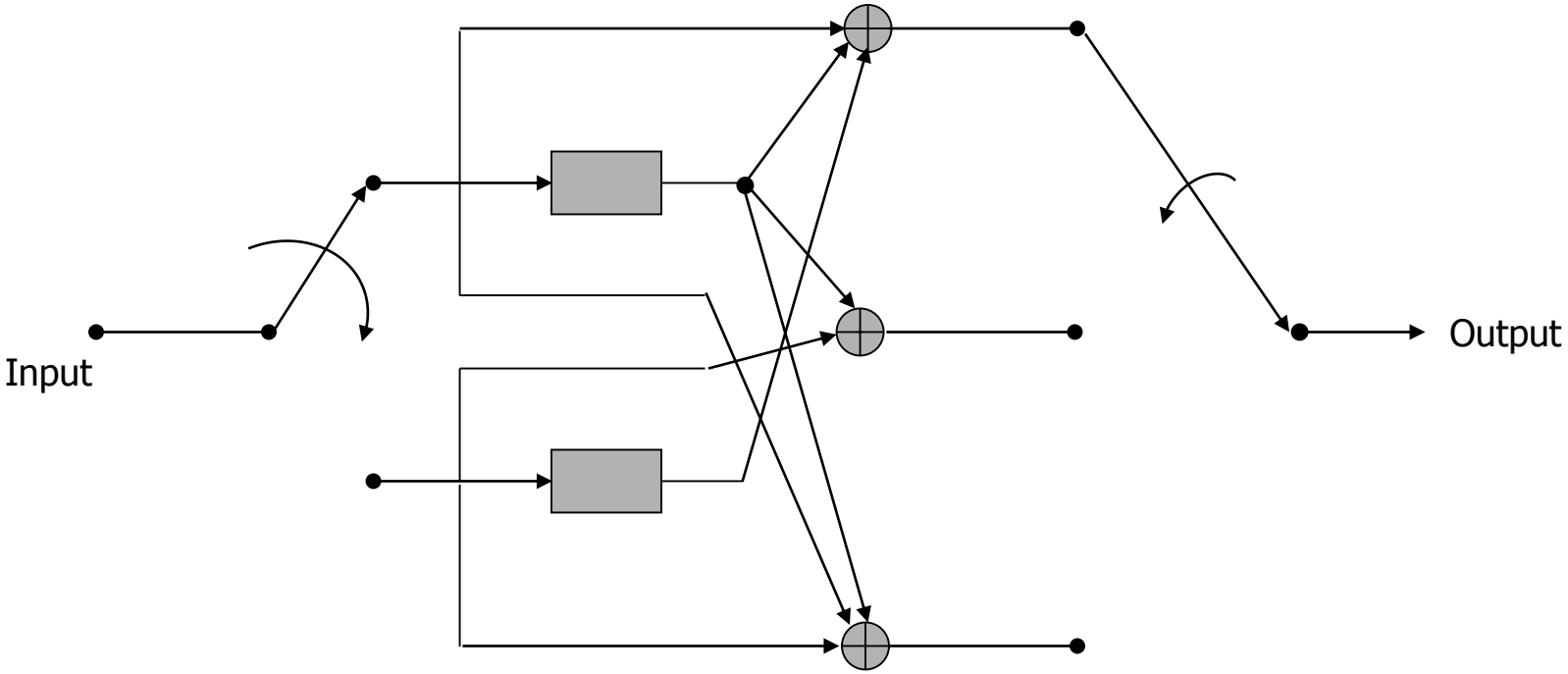
Example

- Mã hóa chập (2,1,2)
  - n=2: 2 bộ cộng modulo-2 và 2 ngõ ra
  - k=1: 1 ngõ vào
  - M=2: 2 tầng thanh ghi dịch (K=M+1=2+1=3)



Example

- Mã hóa chập (3,2,1)
  - n=3: 3 bộ cộng modulo-2 và 3 ngõ ra
  - k=2: 2 ngõ vào
  - M=1: 1 tầng của thanh ghi dịch (K=2)



Generations

- Mã chập là mã hóa không đối xứng
- Mỗi đường kết nối với ngõ ra với ngõ vào bởi một đặc tính đáp ứng xung (hay đa thức sinh)
- $(g_M^{(i)},...,g_2^{(i)},g_1^{(i)},g_0^{(i)})$ ký hiệu của đáp ứng xung của đường thứ i
- Đa thức sinh đường thứ i :

$$g^{(i)}(D) = g_M^{(i)}D^M + ... + g_2^{(i)}D^2 + g_1^{(i)}D + g_0^{(i)}$$

- D ký hiệu của biến độ trễ $\Rightarrow$ khác với X của mã vòng
- Mộ mã hóa chập hoàn chỉnh được diễn tả bởi tập đa thức sinh $\{ g^{(1)}(D), g^{(2)}(D),...,g^{(n)}(D)\}$

Example(1/8)

- Xét trường hợp (2,1,2)
- Đáp ứng xung của đường thứ 1 là (1,1,1)
- Đa thức sinh tương ứng là

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

- Đáp ứng xung của đường thứ 2 là (1,0,1)
- Đa thứ sinh tương ứng là

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

- Chuỗi thông tin: (11001)
- Trình bày đa thức: $m(D) = D^4 + D^3 + 1$

Example(2/8)

- Ngõ ra đa thức của đường thứ 1:

$$c^{(1)}(D) = m(D)g^{(1)}(D)$$
$$= (D^4 + D^3 + 1)(D^2 + D + 1)$$
$$= D^6 + D^5 + D^4 + D^5 + D^4 + D^3 + D^2 + D + 1$$
$$= D^6 + D^3 + D^2 + D + 1$$

- Chuỗi ra của đường thứ 1 (1001111)
- Ngõ ra đa thức của đường 2:

$$c^{(2)}(D) = m(D)g^{(2)}(D)$$
$$= (D^4 + D^3 + 1)(D^2 + 1)$$
$$= D^6 + D^4 + D^5 + D^3 + D^2 + 1$$

- Chuỗi ra của đường thứ 2 (1111101)

Example(3/8)

- m= (11001)
- $c^{(1)}$=(1001111)
- $c^{(2)}$=(1111101)
- Chuỗi đã mã hóa c=(11,01,01,11,11,10,11)
- Chiều dài thông tin L=5bits
- Chiều dài ngõ ra n(L+K-1)=14bits
- Một chuỗi cuối K-1=2 zeros được thêm vào các bit ngõ vào để khôi phục giá trị khởi tạo cho thanh ghi dịch

Example(4/8)

- Cách khác để tính ngõ ra:
- Đường 1:

m	111		output
001001	1		1
00100	11		0
0010	011		0
001	001	1	1
00	100	11	1
0	010	011	1
	001	0011	1

$c^{(1)}=(1001111)$

Example(5/8)

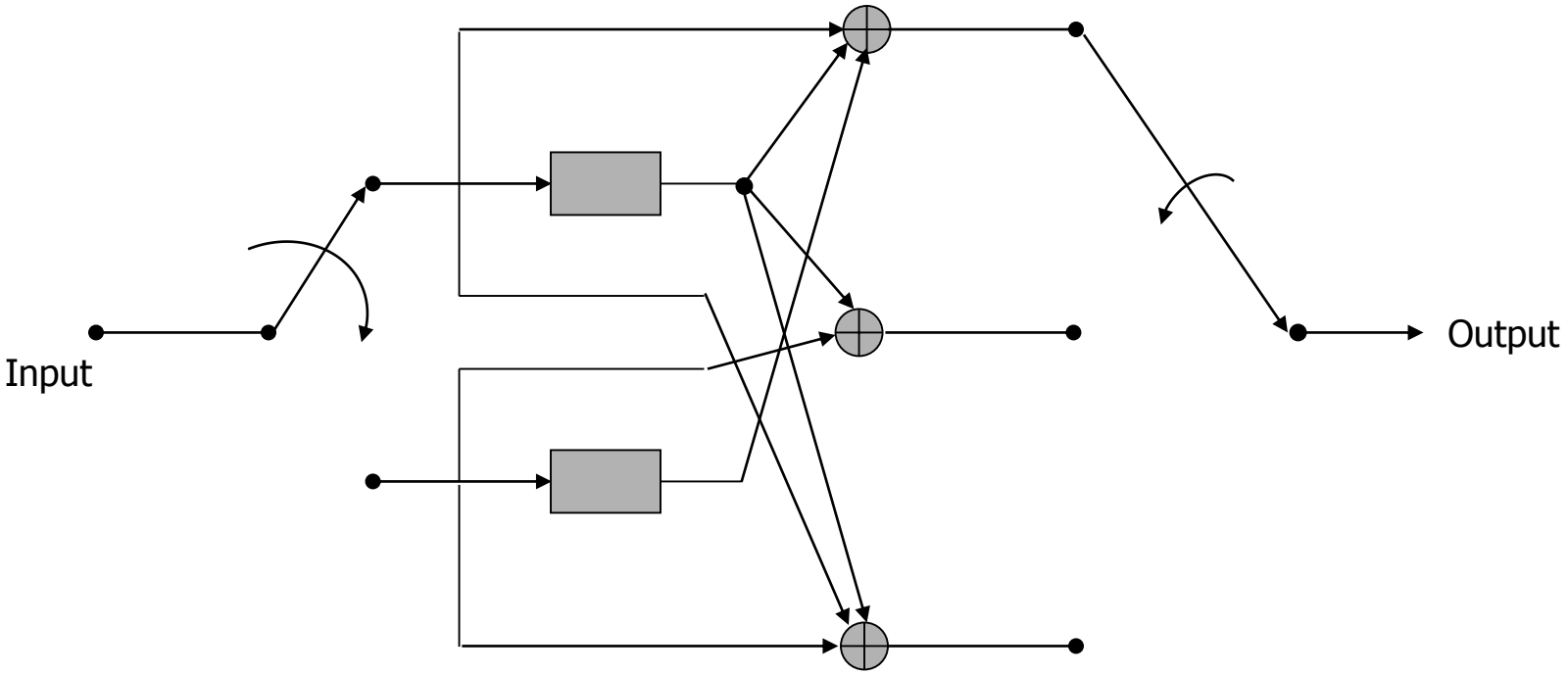
- Đường 2

m	101		output
001001	1		1
00100	11		1
0010	011		1
001	001	1	1
00	100	11	1
0	010	011	0
	001	0011	1

$c^{(2)}=(1111101)$

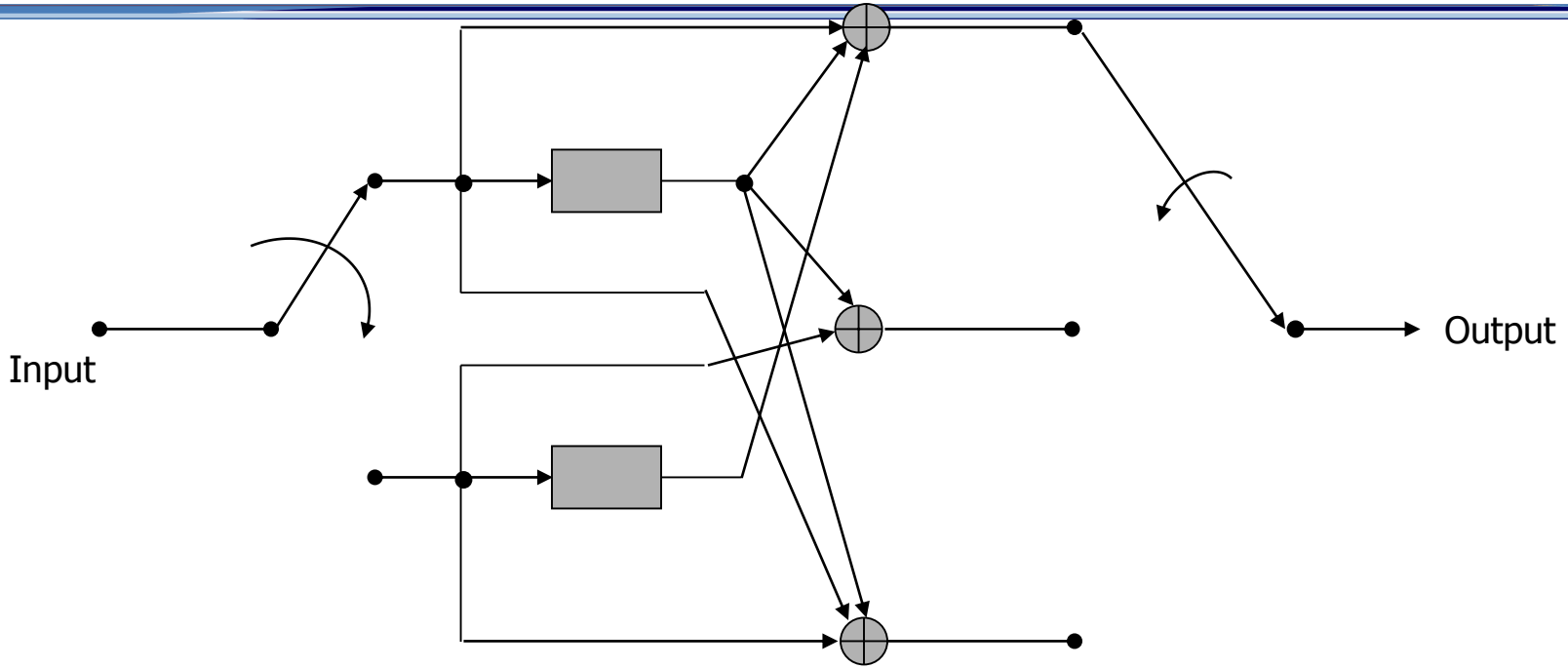
Example(6/8)

- Xét trường hợp (3,2,1)



- $g_i^{(j)} = (g_{i,M}^{(j)}, g_{i,M-1}^{(j)}, \ldots, g_{i,1}^{(j)}, g_{i,0}^{(j)})$ ký hiệu cho đáp ứng xung đường thứ j tương ứng với ngõ vào thứ i

Example(7/8)



$$g_1^{(1)} = (11) \Rightarrow g_1^{(1)}(D) = D + 1$$

$$g_2^{(1)} = (01) \Rightarrow g_1^{(1)}(D) = 1$$

$$g_1^{(2)} = (01) \Rightarrow g_1^{(2)}(D) = 1$$

$$g_2^{(2)} = (10) \Rightarrow g_2^{(2)}(D) = D$$

$$g_1^{(3)} = (11) \Rightarrow g_1^{(1)}(D) = D + 1$$

$$g_2^{(3)} = (10) \Rightarrow g_1^{(1)}(D) = D$$

Example(8/8)

- Assume that:
  - $m^{(1)}=(101) \Rightarrow m^{(1)}(D)=D^2+1$
  - $m^{(2)}=(011) \Rightarrow m^{(1)}(D)=D+1$
- Outputs are:
- $c^{(1)}=m^{(1)}*g_1^{(1)}+m^{(2)}*g_2^{(1)}$

  $= (D^2+1)(D+1)+(D+1)(1)$

  $=D^3+D^2+D+1+D+1=D^3+D^2 \Rightarrow c^{(1)}=(1100)$
- $c^{(2)}=m^{(1)}*g_1^{(2)}+m^{(2)}*g_2^{(2)}$

  $= (D^2+1)(1)+(D+1)(D)$

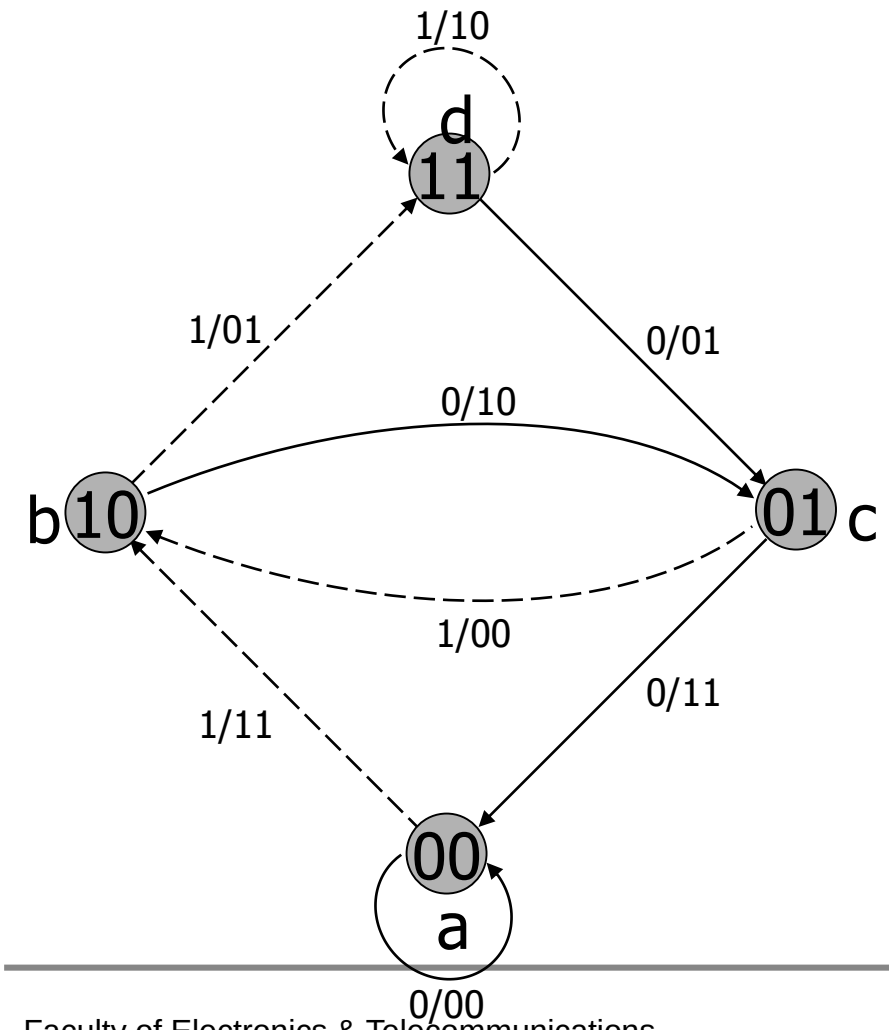
  $=D^2+1+D^2+D=D+1 \Rightarrow c^{(2)}=(0011)$
- $c^{(3)}=m^{(1)}*g_1^{(3)}+m^{(2)}*g_2^{(3)}$

  $= (D^2+1)(D+1)+(D+1)(D)$

  $=D^3+D^2+D+1+D^2+D=1=D^3+1 \Rightarrow c^{(3)}=(1001)$
- Output c=(101,100,010,011)

State diagram

Xét mã chập (2,1,2)

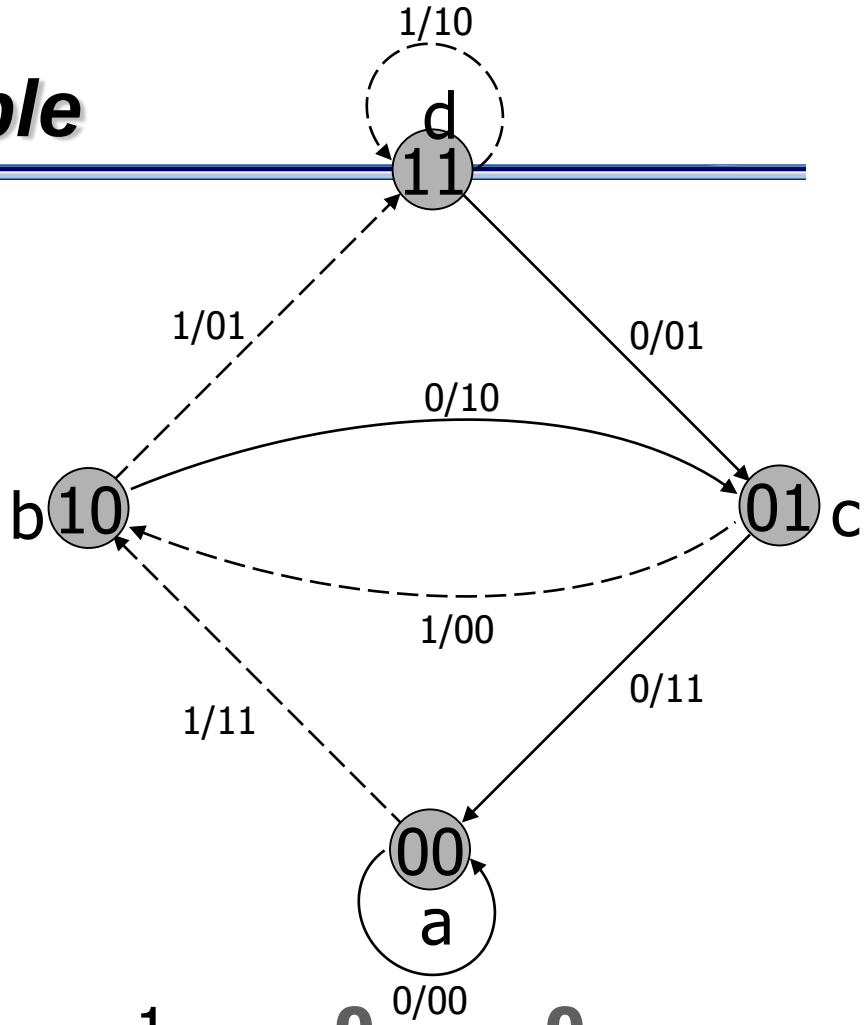


state	Binary description
a	00
b	10
c	01
d	11

- 4 trạng thái có thể
- Mỗi nối có 2 nhánh đến và 2 nhánh đi
- Một chuyển trạng thái tưng ứng với ngõ vào 0 tưng ứng với đường liền nét, và 1 tương ứng với đường đứt nét
- Ngõ ra được gán nhãn trên đường truyền

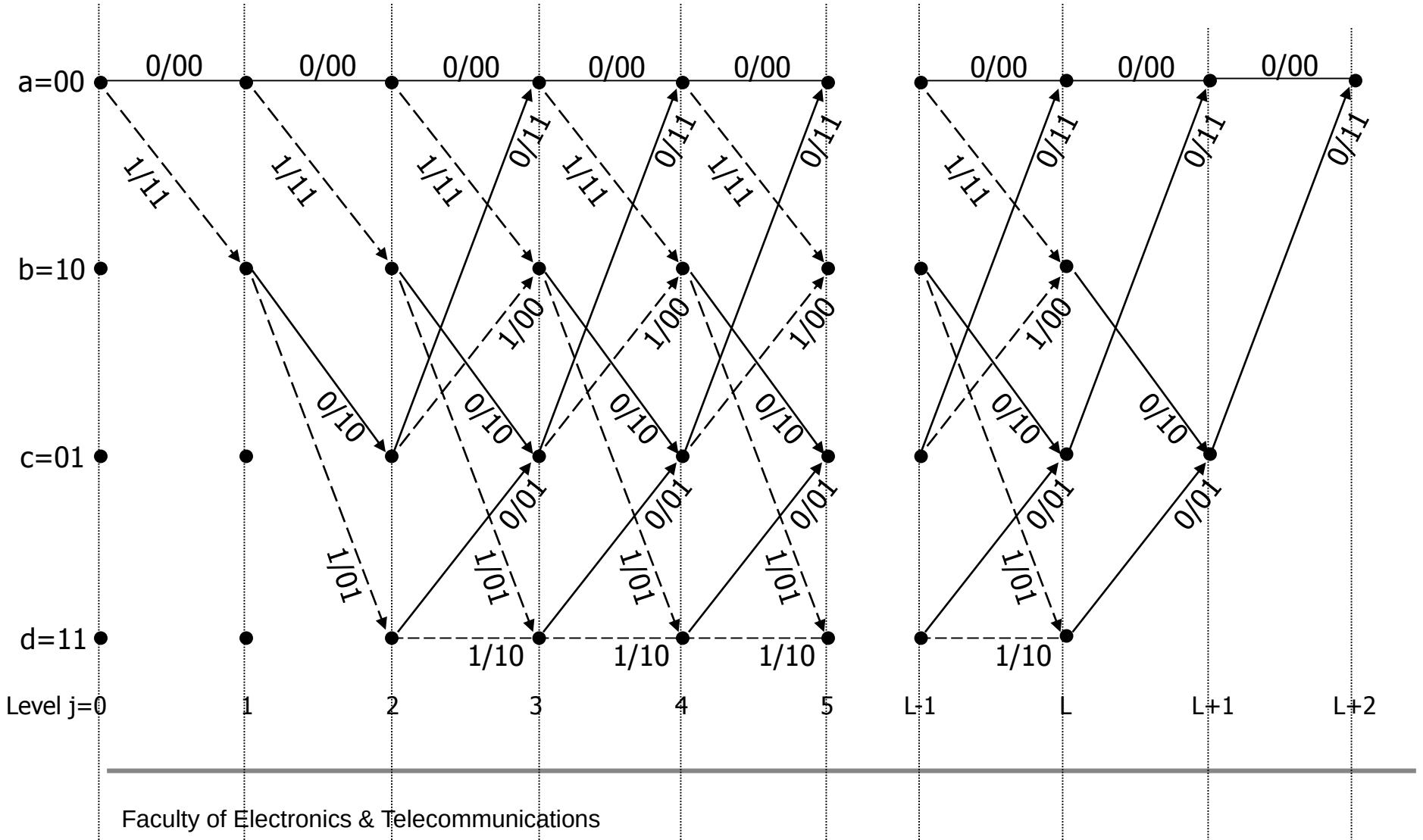
Example

- Chuỗi bit 11001
- Bắt đầu trạng thái a
- Đi theo giản đồ trạng thái để có dữ liệu ngõ ra



Input	1	1	0	0	1	0	0	
State	00	10	11	01	00	10	01	00
	a	b	d	c	a	b	c	a
Output	11	01	01	11	11	10	11	

Trellis(1/2)

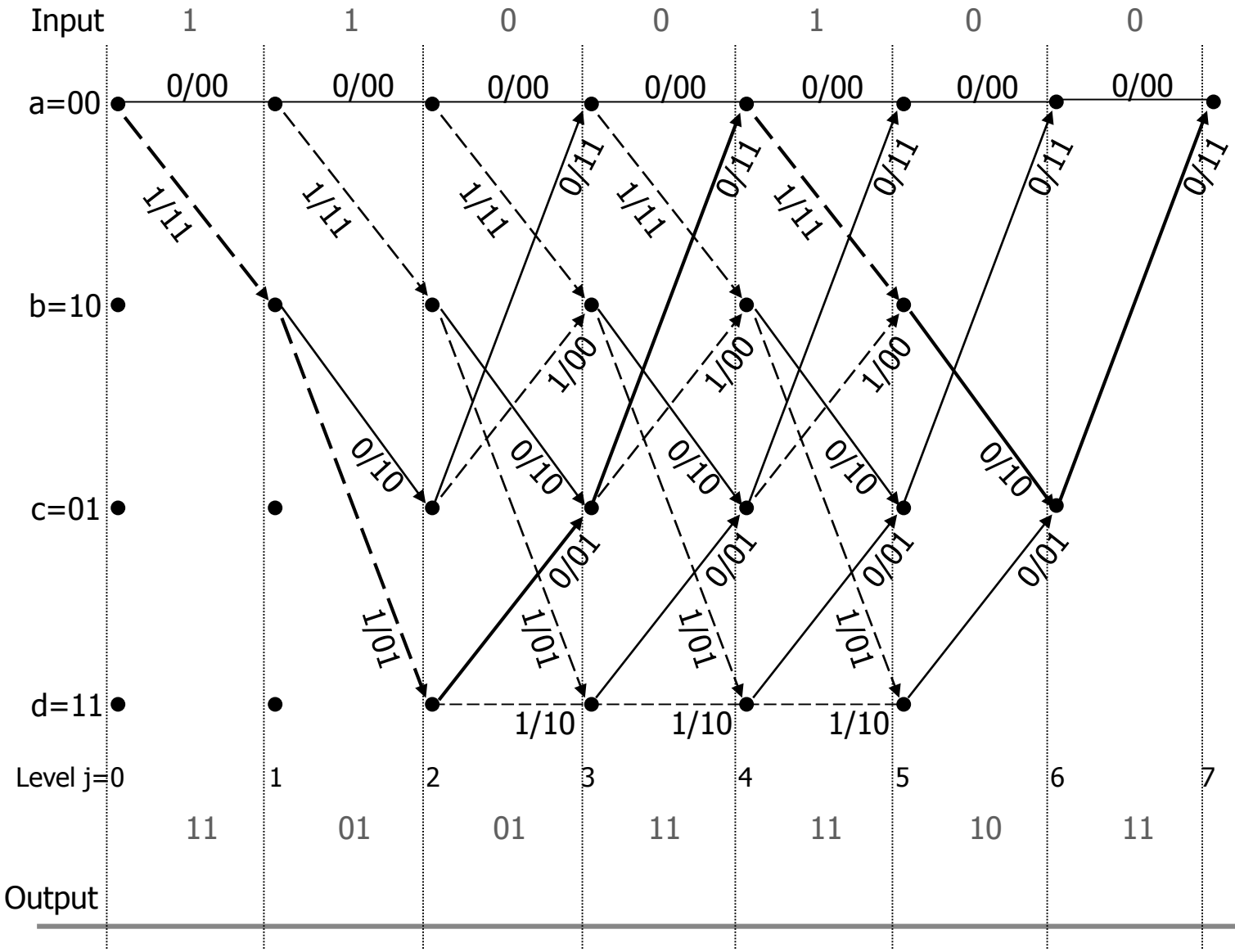

a=00
b=10
c=01
d=11
0/00
1/11
0/11
1/00
0/10
0/01
1/01
1/10
Level j=0
1
2
3
4
5
L-1
L
L+1
L+2

Trellis(1/2)

- Một lưới chứa (L+K) mức
- Được gán nhãn j=0,1,…,L,…,L+K-1
- Mức đầu tiên (K-1) tương ứng với bộ mã hóa ba91t đầu từ trạng thái khởi tạo *a*
- Mức cuối cùng (K-1) tương ứng với bộ mã hóa trở về trạng thái *a*
- Đối với mức thứ j nằm trong khoảnK-1≤j≤L, tất cả các trạng thái có thể đến

- Message 11001

Example



Maximum Likelihood Decoding of Convolutional codes

- *m* ký hiệu là vector thông tin
- *c* ký hiệu vector mã tương ứng
- *r* ký hiệu vector nhận
- Khi có *r* , bộ giải mã yêu cầu ước lượng vector thông tin $\hat{m}$ tương ứng với ước lượng vector từ mã $\hat{c}$
- $\hat{m} = m$ only if $\hat{c} = c$ nếu không, sẽ xuất hiện lỗi
- *Qui luật giải mã* được gọi là tối ưu khi xác xuất lỗi là tối thiểu
- Bộ giải mã cực đại khả năng hoặc qui luật quyết định được diễn tả như sau:
  - Chọn ước lượng $\hat{c}$, với hàm cực đại khả năng log*p(r/c)* đạt cực đại

Maximum Likelihood Decoding of Convolutional codes

- Kênh nhị phân đối xứng: cả c và r là chuỗi nhị phân có chiều dài N

$$p(r \mid c) = \prod_{i=1}^{N} p(r_i \mid c_i)$$

$$\Rightarrow \log p(r \mid c) = \sum_{i=1}^{N} \log p(r_i \mid c_i)$$

$$\text{with} \quad p(r_i \mid c_i) = \begin{cases} p & \text{if } r_i \neq c_i \\ 1-p & \text{if } r_i = c_i \end{cases}$$

- r khác với c là d vị trí, hoặc d là khoảng cách Hamming giữa r và c

$$\Rightarrow \log p(r \mid c) = d \log p + (N-d)\log(1-p)$$

$$= d \log\left(\frac{p}{1-p}\right) + N \log(1-p)$$

Maximum Likelihood Decoding of Convolutional codes

- Qui luật giải mã phát biểu lại như sau:
  - Chọn ước lượng $\hat{c}$, mà tối thiểu khoảng cách Hamming giữa vector nhận *r* và vector truyền *c*
- Vector nhận *r* thì so với mỗi vector mã tương ứng *c*, và cái gần nhất với *r* được chọn như là vector từ mã truyền đã sửa lỗi
- Chọn một đường trong lưới mà chuỗi mã khác với chuỗi nhận có ít nhất các vị trí khác nhau

The Viterbi algorithm

- Thuật toán hoạt động bằng cách tính toán khoảng cách mỗi đường trong lưới
- Số đo là khoảng cách Hamming giữa chuỗi mã được diễn tả bằng các đường và chuỗi nhận
- Đối với mỗi node, hai đường đi vào một node, đường thấp nhất được sống sót. Những cái khác loại bỏ
- Tính toán lặp lại mỗi mức j trong khoảng $K-1 \leq j \leq L$
- Số sống sót mỗi mức $\leq 2^{K-1}=4$

The Viterbi algorithm

- c=(11,01,01,11,11,10,11),r=(11,0**0**,01,11,1**0**,10,11)