

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP VỀ NHÀ  
DATA – LINK**

**BỘ MÔN: MẠNG MÁY TÍNH  
GIẢNG VIÊN LÝ THUYẾT: HUỲNH THỤY BẢO TRÂN**

**NHÓM: 05  
THÀNH VIÊN:**

Lâm Khả Doãn  
Nguyễn Thị Ngọc Mai  
Lê Hùng Sơn  
Phan Đặng Diễm Uyên  
Lưu Đức Vũ

**Thành phố Hồ Chí Minh, tháng 12 năm 2020**

Ngày 23 – 12 – 2020  
Lớp 19CTT2  
Môn Mạng Máy Tính  
Nhóm 05

Danh sách thành viên nhóm

STT	MSSV	Họ và Tên	Ký tên	%
1	19120193	Lâm Khả Doãn		95
2	19120287	Nguyễn Thị Ngọc Mai		95
3	19120349	Lê Hùng Sơn		100
4	19120426	Phan Đăng Diễm Uyên		95
5	19120433	Lưu Đức Vũ		100

## A. PARITY 1 CHIỀU:

- Số bit parity: 1 bit
  - Chiều dài của dữ liệu cần gửi đi:  $d$  bit
- ⇒ Dữ liệu gửi đi sẽ có  $(d + 1)$  bit

### 1. Bài toán thuận: thông tin phía phát gửi ( $D + EDC$ )

#### Phương pháp gửi:

- Thêm 1 bit parity vào dữ liệu cần gửi đi
  - Mô hình chẵn (Even parity)  
Số bit 1 trong  $d + 1$  bit là một số chẵn
  - Mô hình lẻ (Odd parity)  
Số bit 1 trong  $d + 1$  bit là một số lẻ

Ví dụ: Dữ liệu cần gửi: 1101 0011 1100 1101 với mô hình parity chẵn

#### Giải

- Ta sẽ thêm 1 bit parity vào sau dữ liệu cần gửi đi
- Vì mô hình parity chẵn nên số bit 1 trong 17bit gửi đi phải là một số chẵn
- Dữ liệu gửi đi có 10bit 1 nên bit parity được thêm vào là 0

d bits	parity bit
1101 0011 1100 1101	0

Dữ liệu gửi: 1101 0011 1100 1101 0 ( $D = 1101 0011 1100 1101$ ,  $EDC = 0$ )

### 2. Bài toán ngược: Kiểm tra dữ liệu phía thu nhận được ( $D' + EDC'$ ) là đúng hay sai, sửa lỗi. Xác định dữ liệu thật cần gửi

#### Phương pháp kiểm tra:

- Nhận  $D'$  có  $(d + 1)$  bits
- Đếm số bit 1 trong  $(d + 1)$  bits =  $x$
- Mô hình chẵn: nếu  $x$  lẻ  $\Rightarrow$  error
- Mô hình lẻ: nếu  $x$  chẵn  $\Rightarrow$  error

**Ví dụ 1:** Dữ liệu nhận được: 1110 0011 0101 1101 1 với mô hình parity chẵn

**Giải**

- Ta có thể thấy  $D' = 1110\ 0011\ 0101\ 1101$ ,  $EDC' = 1$
- Số bit 1 trong 17 bits nhận được là 11
- Vì mô hình parity chẵn nên với số bit 1 trong 17 bits nhận được là 11, dữ liệu nhận được là sai, có lỗi xảy ra

**Ví dụ 2:** Dữ liệu nhận được: 1110 0011 0101 1101 1 với mô hình parity lẻ

**Giải**

- Ta có thể thấy  $D' = 1110\ 0011\ 0101\ 1101$ ,  $EDC' = 1$
- Số bit 1 trong 17 bits nhận được là 11
- Vì mô hình parity lẻ nên với số bit 1 trong 17 bits nhận được là 11, dữ liệu nhận được là đúng

## **B. PARITY 2 CHIỀU**

- Dữ liệu gửi đi được biểu diễn thành ma trận  $N \times M$
- Số bit parity:  $(N + M + 1)$  bit

### **1. Bài toán thuận: thông tin phía phát gửi ( $D + EDC$ )**

**Phương pháp gửi:**

- Biểu diễn dữ liệu cần gửi đi thành ma trận  $N \times M$
- Tính giá trị bit parity của từng dòng, từng cột

**Ví dụ:**

- Mô hình parity chẵn
- $N = 4, M = 5$
- Dữ liệu cần gửi: 11001 11010 00100 10001

**Giải**

Ta biểu diễn dữ liệu cần gửi đi dưới dạng ma trận 4x5:

1	1	0	0	1	1
1	1	0	1	0	1
0	0	1	0	0	1
1	0	0	0	1	0
<hr/>					
1	0	1	1	0	1

Vậy dữ liệu gửi đi: 110011 110101 001001 100010 101101

**2. Bài toán ngược:** kiểm tra dữ liệu phía thu nhận được ( $D' + EDC'$ ) là đúng hay sai, sửa lỗi. Xác định dữ liệu thật cần gửi

**Phương pháp kiểm tra:**

- Biểu diễn dữ liệu nhận thành ma trận  $(N + 1) \times (M + 1)$
- Kiểm tra tính đúng đắn của từng dòng/cột
- Đánh dấu các dòng /cột dữ liệu bị lỗi
- Bit lỗi: bit tại vị trí giao giữa dòng và cột bị lỗi

**Ví dụ 1:**

- Mô hình parity chẵn
- Dữ liệu nhận được: 110001 110101 001001 100010 101101

**Giải**

- Ta biểu diễn dữ liệu thành ma trận 5x6

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

Vị trí bit lỗi

- Ta thực hiện kiểm tra tính đúng đắn của từng dòng, từng cột:

Cột 1: tổng số bit 1 ở cột 1 là 3. Vì mô hình parity chẵn nên bit parity tại vị trí cột 1 là 1 ⇒ Đúng

Cột 2: tổng số bit 1 ở cột 2 là 2. Vì mô hình parity chẵn nên bit parity tại vị trí cột 2 là 0 ⇒ Đúng

Cột 3: tổng số bit 1 ở cột 3 là 1. Vì mô hình parity chẵn nên bit parity tại vị trí cột 3 là 1 ⇒ Đúng

Cột 4: tổng số bit 1 ở cột 4 là 1. Vì mô hình parity chẵn nên bit parity tại vị trí cột 4 là 1 ⇒ Đúng

Cột 5: tổng số bit 1 ở cột 5 là 1. Vì mô hình parity chẵn nên bit parity tại vị trí cột 5 là 1 ⇒ Sai

Dòng 1: tổng số bit 1 ở dòng 1 là 2. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 1 là 0 ⇒ Sai

Dòng 2: tổng số bit 1 ở dòng 2 là 3. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 2 là 1 ⇒ Đúng

Dòng 3: tổng số bit 1 ở dòng 3 là 1. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 3 là 1 ⇒ Đúng

Dòng 4: tổng số bit 1 ở dòng 4 là 2. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 4 là 0 ⇒ Đúng

Dòng bị lỗi: dòng 1; cột bị lỗi: cột 5

Bit lỗi nằm tại vị trí giao giữa dòng và cột bị lỗi: bit lỗi nằm tại dòng 1 cột 5

Tại vị trí đó bit = 0 ⇒ Ta sửa số 0 thành số 1

Vậy dữ liệu thật gửi là 11001 11010 00100 10001 10110

### Ví dụ 2:

- Mô hình parity chẵn
- Dữ liệu nhận được: 111001 001001 111111 101011 100100

### Giải

- Ta biểu diễn dữ liệu thành ma trận 5x6

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

- Ta thực hiện kiểm tra tính đúng đắn của từng dòng, từng cột:

Cột 1: tổng số bit 1 ở cột 1 là 3. Vì mô hình parity chẵn nên bit parity tại vị trí cột 1 là 1 ⇒ Đúng

Cột 2: tổng số bit 1 ở cột 2 là 2. Vì mô hình parity chẵn nên bit parity tại vị trí cột 2 là 0 ⇒ Đúng

Cột 3: tổng số bit 1 ở cột 3 là 4. Vì mô hình parity chẵn nên bit parity tại vị trí cột 3 là 1 ⇒ Đúng

Cột 4: tổng số bit 1 ở cột 4 là 1. Vì mô hình parity chẵn nên bit parity tại vị trí cột 4 là 1 ⇒ Đúng

Cột 5: tổng số bit 1 ở cột 5 là 2. Vì mô hình parity chẵn nên bit parity tại vị trí cột 5 là 0 ⇒ Đúng

Dòng 1: tổng số bit 1 ở dòng 1 là 3. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 1 là 1 ⇒ Đúng

Dòng 2: tổng số bit 1 ở dòng 2 là 1. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 2 là 1  $\Rightarrow$  Đúng

Dòng 3: tổng số bit 1 ở dòng 3 là 5. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 3 là 1  $\Rightarrow$  Đúng

Dòng 4: tổng số bit 1 ở dòng 4 là 3. Vì mô hình parity chẵn nên bit parity tại vị trí dòng 4 là 1  $\Rightarrow$  Đúng

Ta có thể thấy dữ liệu nhận được hoàn toàn không có lỗi  $\Rightarrow$  Dữ liệu nhận được là đúng

Vậy dữ liệu thật gửi là 11100 00100 11111 10101 10010

## C. CHECK SUM

### 1. Bài toán thuận: thông tin phía phát gửi ( $D + EDC$ )

#### Phương pháp gửi:

- d bits trong dữ liệu gửi đi được xem như gồm N số k bits:  $x_1, x_2, \dots, x_N$
- Tính tổng  $X = x_1 + x_2 + \dots + x_N$
- Tính bù 1 của X  $\Rightarrow$  giá trị checksum  
 $\Rightarrow$  Dữ liệu gửi: d bits + checksum

Ví dụ: Dữ liệu cần gửi 1001 1001 1110 0010 và  $k = 4$

#### Giải

16 bits trong dữ liệu gửi đi được xem như gồm 4 số 4 bits:

$$x_1 = 1001, x_2 = 1001, x_3 = 1110, x_4 = 0010$$

Ta tính tổng X:  $\text{sum} = x_1 + x_2 + x_3 + x_4 = 1001 + 1001 + 1110 + 0010$

Đầu tiên, ta tính  $1001 + 1001$

$$\begin{array}{rcccc} & 1 & 0 & 0 & 1 \\ + & 1 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 \end{array}$$



Vì  $k = 4$  nên ta đem 1 xuống và cộng với các số còn lại, ta được tổng

$$\begin{array}{r} 0 \quad 0 \quad 1 \quad 0 \\ + \quad \quad \quad 1 \\ \hline 0 \quad 0 \quad 1 \quad 1 \end{array}$$

Tiếp theo, ta tính  $1001 + 1001 + 1110 = 0011 + 1110$

$$\begin{array}{r} 0 \quad 0 \quad 1 \quad 1 \\ + \quad 1 \quad 1 \quad 1 \quad 0 \\ \hline 1 \quad 0 \quad 0 \quad 0 \quad 1 \end{array}$$

Ta đem 1 xuống và cộng với các số còn lại, ta được tổng

$$\begin{array}{r} 0 \quad 0 \quad 0 \quad 1 \\ + \quad \quad \quad 1 \\ \hline 0 \quad 0 \quad 1 \quad 0 \end{array}$$

Cuối cùng ta thực hiện phép cộng  $1001 + 1001 + 1110 + 0010 = 0010 + 0010$

$$\begin{array}{r} 0 \quad 0 \quad 1 \quad 0 \\ + \quad 0 \quad 0 \quad 1 \quad 0 \\ \hline 0 \quad 1 \quad 0 \quad 0 \end{array}$$

Vậy  $\text{sum} = 1001 + 1001 + 1110 + 0010 = 0100$

Lấy bù 1 của sum, ta được checksum = 1011

Vậy dữ liệu gửi: 1001 1001 1110 0010 **1011**

( $D = 1001 \ 1001 \ 1110 \ 0010$ ,  $EDC = 1011$ )

**2. Bài toán ngược:** kiểm tra dữ liệu phía thu nhận được ( $D' + EDC'$ ) là đúng hay sai, sửa lỗi. Xác định dữ liệu thật cần gửi

**Phương pháp kiểm tra:**

- Tính tổng cho tất cả giá trị nhận được (kể cả giá trị checksum)
- Nếu tất cả các bit là 1, thì dữ liệu nhận được là đúng; ngược lại, có lỗi xảy ra

**Ví dụ 1:** Dữ liệu nhận được: 1110 0110 0101 1000 1100 và  $k = 4$

**Giải**

$D' = 1110\ 0110\ 0101\ 1000$ ,  $EDC' = 1100$

Ta tính tổng tất cả giá trị nhận được:  $1110 + 0110 + 0101 + 1000 + 1100$

Nếu kết quả tổng bằng 1111, dữ liệu nhận được là đúng; ngược lại, dữ liệu nhận được sai, có lỗi xảy ra.

Đầu tiên, ta tính tổng  $1110 + 0110$

$$\begin{array}{r} 1110 \\ + 0110 \\ \hline 10100 \end{array}$$

Vì  $k = 4$  nên ta đem 1 xuống cộng với các số còn lại, ta được tổng

$$\begin{array}{r} 0100 \\ + 1 \\ \hline 0101 \end{array}$$

Tiếp theo, ta tính tổng  $1110 + 0110 + 0101 = 0101 + 0101$

$$\begin{array}{r} 0101 \\ + 0101 \\ \hline 1010 \end{array}$$

Tiếp theo, ta tính tổng  $1110 + 0110 + 0101 + 1000 = 1010 + 1000$

$$\begin{array}{r} 1010 \\ + 1000 \\ \hline 10010 \end{array}$$

Ta đem 1 xuống cộng với các số còn lại, ta được tổng

$$\begin{array}{r} 0010 \\ + 1 \\ \hline 0011 \end{array}$$

Cuối cùng, ta tính tổng  $1110 + 0110 + 0101 + 1000 + 1100 = 0011 + 1100$

$$\begin{array}{rcccc}
 & 0 & 0 & 1 & 1 \\
 + & 1 & 1 & 0 & 0 \\
 \hline
 & 1 & 1 & 1 & 1
 \end{array}$$

Vì  $1110 + 0110 + 0101 + 1000 + 1100 = 1111$  nên dữ liệu nhận được là đúng

**Ví dụ 2:** Dữ liệu nhận được: 1110 0111 0101 1000 1100 và  $k = 4$

### Giải

$D' = 1110\ 0111\ 0101\ 1000$ ,  $EDC' = 1100$

Ta tính tổng tất cả các giá trị nhận được  $1110 + 0111 + 0101 + 1000 + 1100$

Nếu kết quả tổng bằng 1111, dữ liệu nhận được là đúng; ngược lại, dữ liệu nhận được là sai, có lỗi xảy ra.

Đầu tiên ta tính tổng  $1110 + 0111$

$$\begin{array}{rcccc}
 & 1 & 1 & 1 & 0 \\
 + & 0 & 1 & 1 & 1 \\
 \hline
 1 & 0 & 1 & 0 & 1
 \end{array}$$

Vì  $k = 4$  nên ta đem 1 xuống cộng với các số còn lại, ta được tổng

$$\begin{array}{rcccc}
 & 0 & 1 & 0 & 1 \\
 + & & & & 1 \\
 \hline
 & 0 & 1 & 1 & 0
 \end{array}$$

Tiếp theo ta tính tổng  $1110 + 0111 + 0101 = 0110 + 0101$

$$\begin{array}{rcccc}
 & 0 & 1 & 1 & 0 \\
 + & 0 & 1 & 0 & 1 \\
 \hline
 & 1 & 0 & 1 & 1
 \end{array}$$

Tiếp theo ta tính tổng  $1110 + 0111 + 0101 + 1000 = 1011 + 1000$

$$\begin{array}{rcccc}
 & 1 & 0 & 1 & 1 \\
 + & 1 & 0 & 0 & 0 \\
 \hline
 1 & 0 & 0 & 1 & 1
 \end{array}$$

Ta đem 1 xuống cộng với các số còn lại, ta được tổng

$$\begin{array}{r}
 0 \quad 0 \quad 1 \quad 1 \\
 + \quad \quad \quad 1 \\
 \hline
 0 \quad 1 \quad 0 \quad 0
 \end{array}$$

Cuối cùng ta tính tổng  $1110 + 0111 + 0101 + 1000 + 1100 = 0100 + 1100$

$$\begin{array}{r}
 0 \quad 1 \quad 0 \quad 0 \\
 + \quad 1 \quad 1 \quad 0 \quad 0 \\
 \hline
 1 \quad 0 \quad 0 \quad 0 \quad 0
 \end{array}$$

Ta đem 1 xuống cộng với các số còn lại, ta được tổng

$$\begin{array}{r}
 0 \quad 0 \quad 0 \quad 0 \\
 + \quad \quad \quad 1 \\
 \hline
 0 \quad 0 \quad 0 \quad 1
 \end{array}$$

Vì  $1110 + 0111 + 0101 + 1000 + 1100 = 0001 \neq 1111$  nên dữ liệu nhận được là sai, có lỗi xảy ra

## D. HAMMING CODE KIỂU 1

Mỗi hamming code:

- Có M bit, đánh số từ 1 đến M
- Bit parity:  $\log_2 M$  bits, tại các vị trí lũy thừa của 2
- Dữ liệu thật được đặt tại các vị trí không là lũy thừa của 2

### **1. Bài toán thuận: thông tin phía phát gửi ( $D + EDC$ )**

#### **Phương pháp gửi:**

- Chia dữ liệu cần gửi đi thành các khối dữ liệu (với số bit là số vị trí có thể đặt vào Hamming Code).

- Với mỗi khối dữ liệu thì tạo một Hamming Code.
  - Đặt các bit dữ liệu cần gửi vào các vị trí không phải là lũy thừa của 2 trong Hamming Code (các vị trí được đánh số từ 1 đến M).
  - Tính lấy bits.
  - Tính giá trị của bit parity.

**Ví dụ:**

Ta cần gửi gói dữ liệu: 11100100 với mô hình parity chẵn,  $M = 12$

**Giải**

- Gói dữ liệu có 8 bits, đặt p là số bit parity của gói dữ liệu. Ta có  $p = M - 8 = 12 - 8 = 4$
- Đánh số thứ tự từ 1 đến 12 và đặt các bits cần gửi vào các vị trí không phải là lũy thừa của 2-như bảng biểu dữ liệu sau:

		1		1	1	0		0	1	0	0
$2^0$	$2^1$	3	$2^2$	5	6	7	$2^3$	9	10	11	12

- Bảng check bits:

3			$2^1$	$2^0$
5		$2^2$		$2^0$
6		$2^2$	$2^1$	
7		$2^2$	$2^1$	$2^0$
9	$2^3$			$2^0$
10	$2^3$		$2^1$	
11	$2^3$		$2^1$	$2^0$
12	$2^3$	$2^2$		

- Theo như bảng check bits ta có:
  - Cột  $2^0$  có các vị trí 3, 5, 7, 9, 11. Kết hợp cùng bảng dữ liệu ta có được số bit 1 trong các vị trí trên là 2. Vì ta dùng mô hình parity chẵn nên bit ở vị trí  $2^0$  sẽ là bit 0.
  - Cột  $2^1$  có các vị trí 3, 6, 7, 10, 11. Kết hợp cùng bảng dữ liệu ta có được số bit 1 trong các vị trí trên là 3. Vì ta dùng mô hình parity chẵn nên bit ở vị trí  $2^1$  sẽ là bit 1.
  - Cột  $2^2$  có các vị trí 5, 6, 7, 12. Kết hợp cùng bảng dữ liệu ta có được số bit 1 trong các vị trí trên là 2. Vì ta dùng mô hình parity chẵn nên bit ở vị trí  $2^2$  sẽ là bit 0.
  - Cột  $2^3$  có các vị trí 9, 10, 11, 12. Kết hợp cùng bảng dữ liệu ta có được số bit 1 trong các vị trí trên là 1. Vì ta dùng mô hình parity chẵn nên bit ở vị trí  $2^3$  sẽ là bit 1.

- Ta có bảng dữ liệu cần gửi đi là:

0	1	1	0	1	1	0	1	0	1	0	0
$2^0$	$2^1$	3	$2^2$	5	6	7	$2^3$	9	10	11	12

Vậy dữ liệu được gửi đi là: 011011010100.

## 2. **Bài toán ngược:** kiểm tra dữ liệu phía thu nhận được ( $D' + EDC'$ ) là đúng hay sai, sửa lỗi. Xác định dữ liệu thật cần gửi

### **Phương pháp kiểm tra:**

Với mỗi Hamming Code

- Điền các bit Hamming Code nhận vào các vị trí từ 1 đến M
- Tính check bit
- Kiểm tra các bit parity
  - Nếu tại bit  $2^i$  phát hiện sai  $\Rightarrow$  Đánh dấu Error, hệ số  $k_i = 1$
  - Ngược lại, đánh dấu No Error = 0, hệ số  $k_i = 0$
- Vị trí bit lỗi:  $pos = \sum 2^i * k_i$

**Ví dụ 1:**

- $M = 7$
- Dùng parity chẵn
- Thông tin nhận 1001010

**Giải**

- Đánh số thứ tự từ 1 đến 7 và đặt các bit nhận được vào vị trí như bảng biểu dữ liệu sau:

1	0	0	1	0	1	0
$2^0$	$2^1$	3	$2^2$	5	6	7

- Bảng check bits:

3		$2^1$	$2^0$
5	$2^{2b}$		$2^0$
6	$2^2$	$2^1$	
7	$2^2$	$2^1$	$2^0$

**Vị trí  $2^0$ :** Xét cột  $2^0$  trong bảng check bits

- Các vị trí có bit 1 là 3, 5, 7
  - Giá trị dữ liệu tương ứng với các vị trí trên là 0, 0, 0
- ⇒ Số bit 1 tại các vị trí trên là 0

Vì mô hình parity chẵn nên bit parity tại vị trí  $2^0$  là 0 ⇒ E : 1

**Vị trí  $2^1$ :** Xét cột  $2^1$  trong bảng check bits

- Các vị trí có bit 1 là 3, 6, 7
  - Giá trị dữ liệu tương ứng với các vị trí trên là 0, 1, 0
- ⇒ Số bit 1 tại các vị trí trên là 1

Vì mô hình parity chẵn nên bit parity tại vị trí  $2^1$  là 1 ⇒ E : 1

Vị trí  $2^2$ : Xét cột  $2^2$  trong bảng check bits

- Các vị trí có bit 1 là 5, 6, 7
- Giá trị dữ liệu tương ứng với các vị trí trên là 0, 1, 0  
⇒ Số bit 1 tại các vị trí trên là 1

Vì mô hình parity chẵn nên bit parity tại vị trí  $2^2$  là 1 ⇒ NE : 0

Vị trí bit lỗi pos =  $\sum 2^i * k_i = 2^0 * k_0 + 2^1 * k_1 = 1 * 1 + 2 * 1 = 3$

⇒ Dữ liệu sai ở bit 3

⇒ Bit 3 đang là 0, sửa 0 thành 1

⇒ Thông tin nhận đúng: 10**1**0**1**0

Vậy dữ liệu thật là 1010

### Ví dụ 2:

- M = 12
- Dùng parity lẻ
- Thông tin nhận 101000111010

- Đánh số thứ tự từ 1 đến 12 và đặt các bit nhận được vào vị trí như bảng biểu dữ liệu sau:

1	0	1	0	0	0	1	1	1	0	1	0
$2^0$	$2^1$	3	$2^2$	5	6	7	$2^3$	9	10	11	12



- Bảng check bits:

3			$2^1$	$2^0$
5		$2^2$		$2^0$
6		$2^2$	$2^1$	
7		$2^2$	$2^1$	$2^0$
9	$2^3$			$2^0$
10	$2^3$		$2^1$	
11	$2^3$		$2^1$	$2^0$
12	$2^3$	$2^2$		

Vị trí  $2^0$ : Xét cột  $2^0$  trong bảng check bits

Các vị trí có bit 1 là 3, 5, 7, 9, 11

Giá trị dữ liệu tương ứng với các vị trí trên là 1, 0, 1, 1, 1

⇒ Số bit 1 tại các vị trí trên là 4

Vì mô hình parity lẻ nên bit parity tại vị trí  $2^0$  là 1 ⇒ NE : 0

Vị trí  $2^1$ : Xét cột  $2^1$  trong check bit

Các vị trí có bit 1 là 3, 6, 7, 10, 11

Giá trị dữ liệu tương ứng với các vị trí trên là 1, 0, 1, 0, 1

⇒ Số bit 1 tại các vị trí trên là 3

Vì mô hình parity lẻ nên bit parity tại vị trí  $2^1$  là 0 ⇒ NE : 0

Vị trí  $2^2$ : Xét cột  $2^2$  trong check bit

Các vị trí có bit 1 là 5, 6, 7, 12

Giá trị dữ liệu tương ứng với các vị trí trên là 0, 0, 1, 0

⇒ Số bit 1 tại các vị trí trên là 1

Vì mô hình parity lẻ nên bit parity tại vị trí  $2^2$  là 0 ⇒ NE : 0

Vị trí  $2^3$ : Xét cột  $2^3$  trong check bit

Các vị trí có bit 1 là 9, 10, 11, 12

Giá trị dữ liệu tương ứng với các vị trí trên là 1, 0, 1, 0

⇒ Số bit 1 tại các vị trí trên là 2

Vì mô hình parity lẻ nên bit parity tại vị trí  $2^3$  là 1 ⇒ NE : 0

***Ta có thể thấy không có vị trí bit lỗi***

⇒ Thông tin nhận đúng: 10**100111010**

Dữ liệu thật là 10011010

## **E. HAMMING CODE KIỂU 2**

Số bit parity ràng buộc theo công thức:  $2^p \geq d + p + 1$  với p là số bit parity, d là số bit data

### **1. Bài toán thuận: thông tin phía phát gửi ( $D + EDC$ )**

#### **Phương pháp gửi:**

Tạo từ mã Hamming Code như sau:

- Đánh dấu tất cả vị trí lũy thừa của 2 là bit parity (vị trí 1, 2, 4, 8, 16, 32, ...)
- Tất cả vị trí còn lại là bit dữ liệu sẽ được mã hóa (vị trí 3, 5, 6, 7, 9, 10, 11, 12, 13, ...)
- Giá trị bit parity được tính dựa vào tính chẵn lẻ của một số bit dữ liệu trong từ mã. Vị trí bit parity sẽ xác định luân phiên chuỗi các bit được tính hay bỏ qua
  - Vị trí 1: lấy 1 bit, bỏ qua 1 bit ... (1, 3, 5, 7, 9, 11, 13, ...)
  - Vị trí 2: lấy 2 bits, bỏ qua 2 bits ... (2, 3, 6, 7, 10, 11, 14, 15 ...)
  - Vị trí 4: lấy 4 bits, bỏ qua 4 bits ... (4, 5, 6, 7, 12, 13, 14, 15, 20, 21, 22, 23, ...)
  - Vị trí 8: lấy 8 bits, bỏ qua 8 bits ... (8 – 15, 24 – 31, 40 – 47, ...)
- Bit parity có giá trị 1 khi tổng số bit 1 ở những vị trí nó kiểm tra là số lẻ (odd) và ngược lại.

**Ví dụ:** Dữ liệu cần gửi: **1011 0111**

### **Giải**

Đặt  $p$  là số bit parity của gói dữ liệu

Ta có số bit parity thỏa mãn công thức  $2^p \geq d + p + 1 \Rightarrow d = 8, p = 4$

Xếp dữ liệu lần lượt vào các ô không phải là lũy thừa của 2 như bảng sau:

1	2	3	4	5	6	7	8	9	10	11	12
		1		0	1	1		0	1	1	1
p1	p2		p4				p8				
		d3		d5	d6	d7		d9	d10	d11	d12

- Xác định **p1** (lấy 1 bỏ 1)

Lấy các vị trí: 1, 3, 5, 7, 9, 11

Ta có giá trị tại các vị trí trên là **p1 1 0 1 0 1**

Tổng số bit 1 tại các vị trí trên là 3  $\Rightarrow$  Tổng số lẻ  $\Rightarrow$  **p1 = 1**

- Xác định **p2** (lấy 2 bỏ 2)

Lấy các vị trí: 2, 3, 6, 7, 10, 11

Ta có giá trị tại các vị trí trên là **p2 1 1 1 1 1**

Tổng số bit 1 tại các vị trí trên là 5  $\Rightarrow$  Tổng số lẻ  $\Rightarrow$  **p2 = 1**

- Xác định **p4** (lấy 4 bỏ 4)

Lấy các vị trí: 4, 5, 6, 7, 12

Ta có giá trị tại các vị trí trên là **p4 0 1 1 1**

Tổng số bit 1 tại các vị trí trên là 3  $\Rightarrow$  Tổng số lẻ  $\Rightarrow$  **p4 = 1**

- Xác định **p8** (lấy 8 bỏ 8)

Lấy các vị trí: 8, 9, 10, 11, 12

Ta có giá trị tại các vị trí trên là **p8 0 1 1 1**

Tổng số bit 1 tại các vị trí trên là 3  $\Rightarrow$  Tổng số lẻ  $\Rightarrow$  **p8 = 1**

1	2	3	4	5	6	7	8	9	10	11	12
1	1	1	1	0	1	1	1	0	1	1	1
p1	p2		p4				p8				
		d3		d5	d6	d7		d9	d10	d11	d12

Vậy dữ liệu gửi: **1111 0111 0111**

**2. Bài toán ngược:** kiểm tra dữ liệu phía thu nhận được ( $D' + EDC'$ ) là đúng hay sai, sửa lỗi. Xác định dữ liệu thật cần gửi

**Ví dụ 1:** Dữ liệu nhận được: **1111 0111 0111**

V	V		V				V				
1	2	3	4	5	6	7	8	9	10	11	12
1	1	1	1	0	1	1	1	0	1	1	1
p1	p2		p4				p8				
		d3		d5	d6	d7		d9	d10	d11	d12

Kiểm tra **p1** (lấy 1 bỏ 1)

Lấy các vị trí 1, 3, 5, 7, 9, 11

Giá trị tương ứng tại các vị trí trên: **1 1 0 1 0 1**

Tổng số bit 1 tại các vị trí trên là 4  $\Rightarrow$  Tổng số bit 1 chẵn  $\Rightarrow$  Đúng

Kiểm tra **p2** (lấy 2 bỏ 2)

Lấy các vị trí 2, 3, 6, 7, 10, 11

Giá trị tương ứng tại các vị trí trên: **1 1 1 1 1 1**

Tổng số bit 1 tại các vị trí trên là 6  $\Rightarrow$  Tổng số bit 1 chẵn  $\Rightarrow$  Đúng

Kiểm tra **p4** (lấy 4 bỏ 4)

Lấy các vị trí 4, 5, 6, 7, 12

Giá trị tương ứng tại các vị trí trên: **1 0 1 1 1**

Tổng số bit 1 tại các vị trí trên là 4  $\Rightarrow$  Tổng số bit 1 chẵn  $\Rightarrow$  Đúng

Kiểm tra **p8** (lấy 8 bỏ 8)

Lấy các vị trí 8, 9, 10, 11, 12

Giá trị tương ứng tại các vị trí trên: **1 0 1 1 1**

Tổng số bit 1 tại các vị trí trên là 4  $\Rightarrow$  Tổng số bit 1 chẵn  $\Rightarrow$  Đúng

Vậy dữ liệu gửi đúng và là: **1111 0111 0111**

Dữ liệu thật là: **1011 0111** (bỏ các bit parity **p1, p2, p4, p8**)

Ví dụ 2: Dữ liệu nhận được: **1111 0011 0111**

V	X		X				V				
1	2	3	4	5	6	7	8	9	10	11	12
<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>
<b>p1</b>	<b>p2</b>		<b>p4</b>				<b>p8</b>				
		d3		d5	d6	d7		d9	d10	d11	d12

Kiểm tra **p1** (lấy 1 bỏ 1)

Lấy các vị trí 1, 3, 5, 7, 9, 11

Giá trị tương ứng tại các vị trí trên: **1 1 0 1 0 1**

Tổng số bit 1 tại các vị trí trên là 4  $\Rightarrow$  Tổng số bit 1 chẵn  $\Rightarrow$  Đúng

Kiểm tra **p2** (lấy 2 bỏ 2)

Lấy các vị trí 2, 3, 6, 7, 10, 11

Giá trị tương ứng tại các vị trí trên: **1 1 0 1 1 1**

Tổng số bit 1 tại các vị trí trên là 5  $\Rightarrow$  Tổng số bit 1 lẻ  $\Rightarrow$  Sai

Kiểm tra **p4** (lấy 4 bỏ 4)

Lấy các vị trí 4, 5, 6, 7, 12

Giá trị tương ứng tại các vị trí trên: **1 0 0 1 1**

Tổng số bit 1 tại các vị trí trên là 3  $\Rightarrow$  Tổng số bit 1 lẻ  $\Rightarrow$  Sai

Kiểm tra **p8** (lấy 8 bỏ 8)

Lấy các vị trí 8, 9, 10, 11, 12

Giá trị tương ứng tại các vị trí trên: **1 0 1 1 1**

Tổng số bit 1 tại các vị trí trên là 4  $\Rightarrow$  Tổng số bit 1 chẵn  $\Rightarrow$  Đúng

**p2 và p4 sai  $\Rightarrow$  Vị trí dữ liệu sai là  $2 + 4 = 6$**

**$\Rightarrow$  d6 đang là 0, sửa 0 thành 1**

Vậy dữ liệu gửi là: **1111 0111 0111**

Dữ liệu thật là: **1011 0111**