**BÁO CÁO NỘI DUNG DATA LINK**

**NHÓM THỰC HIỆN: 04**

1. **Danh sách thành viên**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | MSSV | Họ và tên | Tỉ lệ công việc được giao hoàn thành |
| 1 | 19120268 | Ngô Đặng Gia Lâm | 100% |
| 2 | 19120338 | Trần Hoàng Quân | 100% |
| 3 | 19120383 | Huỳnh Tấn Thọ | 100% |
| 4 | 19120407 | Lâm Hải Triều | 100% |
| 5 | 19120469 | Sử Nhật Đăng | 100% |

1. **Nội dung**
2. **Parity 1 chiều**

**PHÂN LOẠI:**

* EVEN PARITY: số bit 1 phải là một số chẵn
* ODD PARITY: số bit 1 phải là một số lẻ

Số bit parity: 1 bit

Chiều dài của dữ liệu cần gởi đi: d bit 🡪 nên dữ liệu gửi đi sẽ có (d+1) bit

1. Bên gửi:

Thêm 1 bit parity vào dữ liệu cần gửi đi:

* Mô hình chẵn (even parity): số bit 1 trong d+1 là một số chẵn
* Mô hình lẽ (odd parity): số bit 1 trong d+1 là một số lẻ

1. Bên nhận:

* Nhận D’ có (d+1) bits
* Đếm số bit 1 trong (d+1) bits = x
* Mô hình chẵn: nếu x lẻ 🡪 error
* Mô hình lẻ: nếu x lẻ 🡪 error

**VÍ DỤ 1:**

Dữ liệu nhận: 1001 0011 1010 1100 1

Parity chẵn: sai

Parity lẻ: đúng

🡪 Dữ liệu thật: 1001 0011 1010 1100

**VÍ DỤ 2:**

Dữ liệu nhận: 1011 1011 0101 1010 010

Parity chẵn: đúng

Parity lẻ: sai

🡪 Dữ liệu thật: 1011 1011 0101 1010 01

**VÍ DỤ 3:**

Dữ liệu nhận: 0011 1010 1100 0101

Parity chẵn: đúng

Parity lẻ: sai

🡪 Dữ liệu thật: 0011 1010 1100 010

**VÍ DỤ 4:** Theo mô hình chẵn (even parity):

Dữ liệu muốn gửi: 1011 1001

Dữ liệu được gửi đi sẽ là : 1011 1001 1

Tuy nhiên, khả năng bên nhận sẽ nhận sai dữ liệu vẫn có nhưng rất thấp, giả sử sai ở bit thứ 5, khiến bên nhận nhận được chuỗi bit sau: 1011 0001 1

Nhận thấy có 5 bit có giá trị là 1🡪 Dữ liệu nhận được là sai. Nhưng không thể biết được vị trí bit bị lỗi.

Đây là nhược điểm thứ nhất của phương pháp parity 1 chiều.

**VÍ DỤ 5:** Theo mô hình lẻ (odd parity):

Dữ liệu muốn gửi: 1000 1111 0101

Dữ liệu đúng phải được gửi đi sẽ là: 1000 1111 0101 0

Giả sử có 2 bit bị lỗi trong quá trình truyền dữ liệu, khiến cho chuỗi bit bên nhận bị sai ở hai vị trí (vị trí thứ 2 và , nhận được chuỗi: 1100 1111 0101 1

Nhận thấy có 9 bit có giá trị là 1 🡪 Dữ liệu nhận được sẽ được đánh giá là đúng

Phương pháp này không phát hiện được lỗi khi số bit lỗi trong dữ liệu là một số chẵn.

Và đây cũng là nhược điểm thứ hai của phương pháp parity 1 chiều.

1. **Parity 2 chiều**

* Dữ liệu gửi đi được biểu diễn thành dạng ma trận
* Số bit parity: bit.

Đặc điểm: phát hiện và sửa được 1 bit lỗi.

Bên gửi:

* Biểu diễn dữ liệu cần gửi đi thành dạng ma trận
* Tính giá trị parity của từng dòng, từng cột.

Bên nhận:

* Biểu diễn dữ liệu nhận thành ma trận
* 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:** Gửi dữ liệu.

Dùng parity chẵn, N = 8, M = 8.

Dữ liệu cần gửi:

11001100 01010101 01010101 01010101 10101010 11001100 11110000 10110010

Biểu diễn dưới dạng ma trận:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Cột  Dòng | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 3 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 4 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 5 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 6 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 7 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 8 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 9 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |

Các parity bit ở mỗi dòng và mỗi cột được tính dựa theo bảng sau đây:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên dòng / Tên cột | Data | Số lượng bit 1 | Chẵn hay lẻ? | Parity bit |
| Dòng 1 | 11001100 | 4 | Chẵn | 0 |
| Dòng 2 | 01010101 | 4 | Chẵn | 0 |
| Dòng 3 | 01010101 | 4 | Chẵn | 0 |
| Dòng 4 | 01010101 | 4 | Chẵn | 0 |
| Dòng 5 | 10101010 | 4 | Chẵn | 0 |
| Dòng 6 | 11001100 | 4 | Chẵn | 0 |
| Dòng 7 | 11110000 | 4 | Chẵn | 0 |
| Dòng 8 | 10110010 | 4 | Chẵn | 0 |
| Cột 1 | 10001111 | 5 | Lẻ | 1 |
| Cột 2 | 11110110 | 6 | Chẵn | 0 |
| Cột 3 | 00001011 | 3 | Lẻ | 1 |
| Cột 4 | 01110011 | 5 | Lẻ | 1 |
| Cột 5 | 10001100 | 3 | Lẻ | 1 |
| Cột 6 | 11110100 | 5 | Lẻ | 1 |
| Cột 7 | 00001001 | 2 | Chẵn | 0 |
| Cột 8 | 01110000 | 3 | Lẻ | 1 |

Ở bit cuối tại vị trí (9, 9), ta thấy cả cột 9 và dòng 9 đều có số lượng bit chẵn nên ô (9, 9) là 0.

Khi đó, dữ liệu gửi đi sẽ có N = 9, M = 9 và nội dung là 110011000 010101010 010101010 010101010 101010100 110011000 111100000 101100100 101111010.

**Ví dụ 2:** Nhận dữ liệu không lỗi.

Dùng parity chẵn, N = 5, M = 4

Dữ liệu nhận: 11000 01001 00011 10001 00101 00110

Ta biểu diễn dữ liệu nhận dưới dạng ma trận:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cột  Dòng | 1 | 2 | 3 | 4 | 5 |
| 1 | 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 1 |
| 3 | 0 | 0 | 0 | 1 | 1 |
| 4 | 1 | 0 | 0 | 0 | 1 |
| 5 | 0 | 0 | 1 | 0 | 1 |
| 6 | 0 | 0 | 1 | 1 | 0 |

Với từng dòng và từng cột, ta kiểm tra bit parity check như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên dòng/cột | Data | Số lượng bit 1 | Parity check | Đúng/Sai |
| Dòng 1 | 1100 | 2 | 0 | Đúng |
| Dòng 2 | 0100 | 1 | 1 | Đúng |
| Dòng 3 | 0001 | 1 | 1 | Đúng |
| Dòng 4 | 1000 | 1 | 1 | Đúng |
| Dòng 5 | 0010 | 1 | 1 | Đúng |
| Cột 1 | 10010 | 2 | 0 | Đúng |
| Cột 2 | 11000 | 2 | 0 | Đúng |
| Cột 3 | 00001 | 1 | 1 | Đúng |
| Cột 4 | 00100 | 1 | 1 | Đúng |

Tại ô (6, 5), cả dòng 6 và cột 5 đều có số lượng bit chẵn nên ô (6, 5) bit parity check là 0. Kiểm tra thấy đúng.

Vậy dữ liệu nhận được không có lỗi, dữ liệu thực nhận là 1100 0100 0001 1000 0010.

Ví dụ 3: Nhận dữ liệu và sửa lỗi.

Dùng parity chẵn, N = 5, M = 4.

Dữ liệu nhận: 11000 01001 00001 10001 00101 00110.

Ta biểu diễn dữ liệu nhận dưới dạng ma trận:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cột  Dòng | 1 | 2 | 3 | 4 | 5 |
| 1 | 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 1 |
| 3 | 0 | 0 | 0 | 0 | 1 |
| 4 | 1 | 0 | 0 | 0 | 1 |
| 5 | 0 | 0 | 1 | 0 | 1 |
| 6 | 0 | 0 | 1 | 1 | 0 |

Với từng dòng và từng cột, ta kiểm tra bit parity check như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên dòng/cột | Data | Số lượng bit 1 | Parity check | Đúng sai |
| Dòng 1 | 1100 | 2 | 0 | Đúng |
| Dòng 2 | 0100 | 1 | 1 | Đúng |
| Dòng 3 | 0000 | 0 | 1 | Sai |
| Dòng 4 | 1000 | 1 | 1 | Đúng |
| Dòng 5 | 0010 | 1 | 1 | Đúng |
| Cột 1 | 10010 | 2 | 0 | Đúng |
| Cột 2 | 11000 | 2 | 0 | Đúng |
| Cột 3 | 00001 | 1 | 1 | Đúng |
| Cột 4 | 00000 | 0 | 1 | Sai |

Ta thấy ở dòng 3 và cột 4 xảy ra lỗi, vậy ta đảo bit ở vị trí giao là bit ở vị trí (3, 4) thành bit 1.

Kiểm lại thấy không có lỗi, vậy ta đã sửa lỗi thành công.

1. **Checksum**

Bên gửi:

* d bit trong dữ liệu gửi đi được xem như N số, mỗi số có chiều dài k bit: x1, x2,…, xn
* Tính tổng X = x1 + x2 + … + xn
* Tính **bù 1** của X => Giá trị checksum

Bên nhận:

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

**VÍ DỤ 1:**

Có một đoạn nhị phân sau: 00010110011001010110101111010011

Hãy tính mã checksum có độ dài 4 của đoạn mã nhị phân trên.

**Giải**: Chia đoạn mã trên thành các đoạn con có độ dài là 4:

0001 0110 0110 0101 0110 1011 1101 0011

Tính tổng X:

|  |
| --- |
| 0001  0110  -------  0111  0110  -------  1101  0101  -------  0010  1  --------  0011  0110  --------  1001  1011  -------  0100  1  -------  0101  1101  -------  0010  1  ------  0011  0011  ------  0110 |

X = 0110, bù 1 của X = 1001 chính là mã checksum cần tìm.

**VÍ DỤ 2:**

Nhận được một đoạn thông tin dưới dạng nhị phân và mã checksum của nó:

1101 1111 0001 1101 0101 1011 1010 0001 0100

Checksum: 0010.

Hãy kiểm tra xem mã trên có còn nguyên vẹn trong quá trình gửi hay không.

**Giải:**

Ta tính tổng cả dãy và checksum:

|  |
| --- |
| 1101  1111  ------  1100  1  ------  1101  0001  ------  1110  1101  ------  1011  1  ------  1100  0101  ------  0001  1  -------  0010  1011  ------  1101  1010  ------  0111  1  ------  1000  0001  -------  1001  0100  -------  1101  0010  ------  1111 |

Vì: Tổng + checksum = 1111 => đoạn mã trên bảo toàn trong quá trình gửi nhận.

**VÍ DỤ 3:**

Nhận được một đoạn thông tin dưới dạng nhị phân và mã checksum của nó:

1101 1111 0001 1101 0101 1011 1010 1011 0101

Checksum: 0010.

Hãy kiểm tra xem mã trên có còn nguyên vẹn trong quá trình gửi hay không.

**Giải:** Ta tính tổng cả dãy và check sum

|  |
| --- |
| 1101  1111  ------  1100  1  ------  1101  0001  ------  1110  1101  ------  1011  1  ------  1100  0101  ------  0001  1  -------  0010  1011  ------  1101  1010  ------  0111  1  ------  1000  1011  -------  0011  1  ------  0100  0101  -------  1001  0010  ------  1011 |

Vì: => đoạn mã trên đã bị thay đổi trong quá trình gửi nhận.

1. **Hamming Code kiểu 1**

Mỗi hamming code:

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

Đặc điểm:

* Sửa lỗi 1 bit
* Nhận dạng được 2 bit lỗi
* Sửa lỗi nhanh hơn Parity code 2 chiều

Bên 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 => tạo 1 Hamming Code
* Đặt các bit dữ liệu thật vào các vị trí không phải là lũy thừa của 2 trong Hamming Code

+ Lưu ý: vị trí được đánh số từ 1 đến M

* Tính check bits
* Tính giá trị của các bit parity

**Ví dụ: chiều thuận – tạo hamming code từ dữ liệu cần gửi**

**VÍ DỤ 1**:

**Đề bài**: Hãy tạo Hamming Code từ dữ kiện sau:

* Dữ liệu cần gửi là: 10101101010
* M = 15
* Sử dụng parity chẵn

**Giải:** Từ dữ liệu cần gửi là 10101101010, ta có bảng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | 1 |  | 0 | 1 | 0 |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

* Ta lại có, ở các vị trí không phải là lũy thừa của 2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

* Xét vị trí bit hamming code 20, ta lấy các bit có vị trí bit thứ 1 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị trí 3, 5, 7, 9, 11, 13, 15 tương ứng là 1 0 0 1 0 0 0, có tổng cộng 2 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 20 là 0

Hiện tại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 |  | 1 |  | 0 | 1 | 0 |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

* Xét vị trí bit hamming code 21, ta lấy các bit có vị trí bit thứ 2 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 3, 6, 7, 10, 11, 14, 15 tương ứng là 1 1 0 1 0 1 0, có tổng cộng 4 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 21 là 0

Hiện tại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 |  | 0 | 1 | 0 |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

* Xét vị trí bit hamming code 22, ta lấy các bit có vị trí bit thứ 3 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 5, 6, 7, 12, 13, 14, 15 tương ứng là 0 1 0 1 0 1 0, có tổng cộng 3 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 22 là 1

Hiện tại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

* Xét vị trí bit hamming code 23, ta lấy các bit có vị trí bit thứ 4 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 9, 10, 11, 12, 13, 14, 15 tương ứng là 1 1 0 1 0 1 0, có tổng cộng 4 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 23 là 0

Hiện tại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy: dữ liệu gửi sẽ là: 001101001101010

**Ví dụ: chiều nghịch – suy ngược lại dữ liệu đúng cần gửi từ hamming code nhận được**

**VÍ DỤ 2: Việc gửi dữ liệu không bị lỗi, nên bit nhận được không bị lỗi**

**Đề bài:** Hãy kiểm tra dữ liệu nhận được là đúng hay là sai. Nếu đúng, hãy đưa ra dữ liệu thật. Nếu sai, hãy sửa (nếu được). Nếu sửa được, hãy đưa ra dữ liệu thật. Dữ kiện:

* Dữ liệu nhận được 001101001101010
* M = 15
* Dùng parity chẵn

**Giải:** Từ dữ liệu nhận được là 001101001101010, ta có bảng:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Ta lại có, ở các vị trí không phải là lũy thừa của 2:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

* Xét vị trí bit hamming code 20, ta lấy các bit có vị trí bit thứ 1 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị trí 3, 5, 7, 9, 11, 13, 15 tương ứng là 1 0 0 1 0 0 0, có tổng cộng 2 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 20 phải là 0

Ta kiểm tra lại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE |  |  |  |  |  |  |  |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy hamming code ở vị trí 20 chính xác

* Xét vị trí bit hamming code 21, ta lấy các bit có vị trí bit thứ 2 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 3, 6, 7, 10, 11, 14, 15 tương ứng là 1 1 0 1 0 1 0, có tổng cộng 4 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 21 phải là 0

Ta kiểm tra lại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE | NE |  |  |  |  |  |  |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy: hamming code ở vị trí 21 chính xác

* Xét vị trí bit hamming code 22, ta lấy các bit có vị trí bit thứ 3 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 5, 6, 7, 12, 13, 14, 15 tương ứng là 0 1 0 1 0 1 0, có tổng cộng 3 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 22 phải là 1

Ta kiểm tra lại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE | NE |  | NE |  |  |  |  |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy bit hamming code ở bị trí 22 là chính xác

* Xét vị trí bit hamming code 23, ta lấy các bit có vị trí bit thứ 4 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 1 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 9, 10, 11, 12, 13, 14, 15 tương ứng là 1 1 0 1 0 1 0, có tổng cộng 4 bit 1

+ Vì ta dùng parity chẵn

* Bit hamming code ở vị trí 23 phải là 0

Ta kiểm tra lại

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE | NE |  | NE |  |  |  | NE |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

* Ta có:

+ Ở các bit được dánh dấu NE, bit ở vị trí thứ 2i sẽ có hệ số ki= 0

+ Ngược lại, các bit được đánh dấu E, bit ở vị trí thứ 2i sẽ có hệ số ki= 1

* Vì tất cả các bit hamming code đều được đánh dấu NE
* Bit nhận được không bị lỗi

Vậy: với bit nhận đuợc là: 001101001101010

Dữ liệu thật sẽ là: 10101101010

**VÍ DỤ 3: Việc gửi dữ liệu bị phát sinh lỗi, nên bit nhận được cũng bị lỗi**

**Đề bài:** Hãy kiểm tra dữ liệu nhận được là đúng hay là sai. Nếu đúng, hãy đưa ra dữ liệu thật. Nếu sai, hãy sửa (nếu được). Nếu sửa được, hãy đưa ra dữ liệu thật. Dữ kiện:

* Dữ liệu nhận được 111000011101010
* M = 15
* Dùng parity lẻ

**Giải:** Từ dữ liệu nhận được là 111000011101010, ta có bảng:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

* Ta lại có, ở các vị trí không phải là lũy thừa của 2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 0 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

* Xét vị trí bit hamming code 20, ta lấy các bit có vị trí bit thứ 1 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 0 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị trí 3, 5, 7, 9, 11, 13, 15 tương ứng là 1 0 0 1 0 0 0, có tổng cộng 2 bit 1

+ Vì ta dùng parity lẻ

* Bit hamming code ở vị trí 20 phải là 1

Ta kiểm tra lại bảng:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE |  |  |  |  |  |  |  |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy hamming code ở vị trí 20 chính xác

* Xét vị trí bit hamming code 21, ta lấy các bit có vị trí bit thứ 2 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 0 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị trí 3, 6, 7, 10, 11, 14, 15 tương ứng là 1 0 0 1 0 1 0, có tổng cộng 3 bit 1

+ Vì ta dùng parity lẻ

* Bit hamming code ở vị trí 21 phải là 0

Ta kiểm tra lại bảng:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE | E |  |  |  |  |  |  |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy: bit hamming code ở vị tri 21 không chính xác

* Xét vị trí bit hamming code 22, ta lấy các bit có vị trí bit thứ 3 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 0 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

* Ta có các bit ở vị trí 5, 6, 7, 12, 13, 14, 15 tương ứng là 0 0 0 1 0 1 0, có tổng cộng 2 bit 1
* Vì ta dùng parity lẻ
* Bit hamming code ở vị trí 22 phải là 1

Ta kiểm tra lại:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE | E |  | E |  |  |  |  |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy: bit hamming code ở vị tri 22 không chính xác

* Xét vị trí bit hamming code 23, ta lấy các bit có vị trí bit thứ 4 bằng 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | = |  |  |  |  | 21 | + | 20 | = | 0 | 0 | 1 | 1 |
| 0 | 5 | = |  |  | 22 | + |  |  | 20 | = | 0 | 1 | 0 | 1 |
| 0 | 6 | = |  |  | 22 | + | 21 |  |  | = | 0 | 1 | 1 | 0 |
| 0 | 7 | = |  |  | 22 | + | 21 | + | 20 | = | 0 | 1 | 1 | 1 |
| 1 | 9 | = | 23 | + |  |  |  |  | 20 | = | 1 | 0 | 0 | 1 |
| 1 | 10 | = | 23 | + |  |  | 21 |  |  | = | 1 | 0 | 1 | 0 |
| 0 | 11 | = | 23 | + |  |  | 21 | + | 20 | = | 1 | 0 | 1 | 1 |
| 1 | 12 | = | 23 | + | 22 |  |  |  |  | = | 1 | 1 | 0 | 0 |
| 0 | 13 | = | 23 | + | 22 | + |  |  | 20 | = | 1 | 1 | 0 | 1 |
| 1 | 14 | = | 23 | + | 22 | + | 21 |  |  | = | 1 | 1 | 1 | 0 |
| 0 | 15 | = | 23 | + | 22 | + | 21 | + | 20 | = | 1 | 1 | 1 | 1 |

+ Ta có các bit ở vị tri 9, 10, 11, 12, 13, 14, 15 tương ứng là 1 1 0 1 0 1 0, có tổng cộng 4 bit 1

+ Vì ta dùng parity lẻ

* Bit hamming code ở vị trí 23 phải là 1

Ta kiểm tra:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NE | E |  | E |  |  |  | NE |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 20 | 21 |  | 22 |  |  |  | 23 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |

Vậy: bit hamming code ở vị tri 23 là chính xác

* Ta có:

+ Ở các bit được dánh dấu NE, bit ở vị trí thứ 2i sẽ có hệ số ki= 0

+ Ngược lại, các bit được đánh dấu E, bit ở vị trí thứ 2i sẽ có hệ số ki= 1

* Do đó:
* Bit thứ 6 bị lỗi

Từ đó ta có:

* Dữ liệu nhận được: 111000011101010
* Dữ liệu nhận đúng: 111001011101010
* Dữ liệu thật: 10101101010

1. **Hamming Code kiểu 2**

Lưu ý: số bit parity được ràng buộc theo công thức:

Với:

p: số bit parity

d: số bit data

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, 64, …)
* Tất cả vị trí còn lại là bi dữ liệu sẽ được mã hóa (vị trí 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 17, …)
* 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 bi được tính hay bỏ qua.

+ Vị trí 1: Lấy 1 bit, bỏ qua 1 bit (1, 3, 5, 7, 9, 11, 13, 15, …)

+ 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, …)

+ Vị trí 16: Lấy 16 bits, bỏ qua 16 bits (16 -> 31, 48 -> 63, 80 -> 95, …)

* 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: chiều thuận – tạo hamming code từ dữ liệu cần gửi**

**VÍ DỤ 1**: Dữ liệu cần gửi là: 1101 0100 1010 0011

Độ dài dữ liệu cần gửi là d = 16.

Do đó, ta chọn số bit Parity là p = 5 (thỏa mãn công thức )

Các vị trí đặt bit Parity là các vị trí có giá trị là lũy thừa của 2, tức ta đặt tại 1, 2, 4, 8, 16. Những vị trí không được chọn còn lại lần lượt chứa dữ liệu cần gửi.

Ta xem bảng dưới đây để hình dung rõ hơn.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  | **1** |  | **1** | **0** | **1** |  | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** |  | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xác định p1:**

- Ta lấy 1 bit và bỏ 1 bit bắt đầu từ p1, nên các vị trí lấy sẽ là 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21.

- Các giá trị bit lần lượt là: ? 1 1 1 0 0 1 1 0 0 1

- Có 6 bit 1 🡪 chẵn 🡪 p1 = 0

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** |  | **1** |  | **1** | **0** | **1** |  | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** |  | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xác định p2:**

- Ta lấy 2 bit và bỏ 2 bit bắt đầu từ p2, nên các vị trí lấy sẽ là 2, 3, 6, 7, 10, 11, 14, 15, 18, 19.

- Các giá trị bit lần lượt là: ? 1 0 1 1 0 0 1 0 0

- Có 4 bit 1 🡪 chẵn 🡪 p2 = 0

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** |  | **1** | **0** | **1** |  | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** |  | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xác định p4:**

- Ta lấy 4 bit và bỏ 4 bit bắt đầu từ p4, nên các vị trí lấy sẽ là 4, 5, 6, 7, 12, 13, 14, 15, 20, 21.

- Các giá trị bit lần lượt là: ? 1 0 1 0 1 0 1 1 1

- Có 6 bit 1 🡪 chẵn 🡪 p4 = 0

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** |  | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** |  | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xác định p8:**

- Ta lấy 8 bit và bỏ 8 bit bắt đầu từ p8, nên các vị trí lấy sẽ là 8, 9, 10, 11, 12, 13, 14, 15.

- Các giá trị bit lần lượt là: ? 0 1 0 0 1 0 1

- Có 3 bit 1 🡪 lẻ 🡪 p8 = 1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** |  | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xác định p16:**

- Ta lấy 16 bit và bỏ 16 bit bắt đầu từ p16, nên các vị trí lấy sẽ là 16, 17, 18, 19, 20, 21.

- Các giá trị bit lần lượt là: ? 0 0 0 1 1

- Có 2 bit 1 🡪 chẵn 🡪 p16 = 0

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

Vậy dữ liệu gửi là: 0010 1011 0100 1010 0001 1

**Ví dụ: chiều nghịch – giải hamming code từ dữ liệu nhận**

**VÍ DỤ 2: Việc gửi dữ liệu không bị lỗi, nên dữ liệu nhận được là đúng**

- Dữ liệu nhận được là 0010 1011 0100 1010 0001 1

- Dữ liệu nhận có độ dài là 21.

- Số bit Parity trong chuỗi nhận được là p = 5, nằm ở các vị trí 1, 2, 4, 8, 16.

- Dữ liệu thật có độ dài d = 16, nằm ở các bit còn lại của chuỗi.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p1:**

- Ta lấy 1 bit và bỏ 1 bit bắt đầu từ p1, nên các vị trí lấy sẽ là 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21.

- Các giá trị bit lần lượt là: ? 1 1 1 0 0 1 1 0 0 1

- Có 6 bit 1 🡪 chẵn 🡪 p1 = 0 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ |  |  |  |  |  |  |  |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p2:**

- Ta lấy 2 bit và bỏ 2 bit bắt đầu từ p2, nên các vị trí lấy sẽ là 2, 3, 6, 7, 10, 11, 14, 15, 18, 19.

- Các giá trị bit lần lượt là: ? 1 0 1 1 0 0 1 0 0

- Có 4 bit 1 🡪 chẵn 🡪 p2 = 0 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | Đ |  |  |  |  |  |  |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p4:**

- Ta lấy 4 bit và bỏ 4 bit bắt đầu từ p4, nên các vị trí lấy sẽ là 4, 5, 6, 7, 12, 13, 14, 15, 20, 21.

- Các giá trị bit lần lượt là: ? 1 0 1 0 1 0 1 1 1

- Có 6 bit 1 🡪 chẵn 🡪 p4 = 0 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | Đ |  | Đ |  |  |  |  |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p8:**

- Ta lấy 8 bit và bỏ 8 bit bắt đầu từ p8, nên các vị trí lấy sẽ là 8, 9, 10, 11, 12, 13, 14, 15.

- Các giá trị bit lần lượt là: ? 0 1 0 0 1 0 1

- Có 3 bit 1 🡪 lẻ 🡪 p8 = 1 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | Đ |  | Đ |  |  |  | Đ |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p16:**

- Ta lấy 16 bit và bỏ 16 bit bắt đầu từ p16, nên các vị trí lấy sẽ là 16, 17, 18, 19, 20, 21.

- Các giá trị bit lần lượt là: ? 0 0 0 1 1

- Có 2 bit 1 🡪 chẵn 🡪 p16 = 0 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | Đ |  | Đ |  |  |  | Đ |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | Đ |  |  |  |  |  |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

Vậy dữ liệu nhận là đúng. Dữ liệu thật là 1101 0100 1010 0011

**VÍ DỤ 3: Việc gửi dữ liệu bị lỗi, nên dữ liệu nhận được là sai**

- Dữ liệu nhận được là 0010 1011 0100 1110 0001 1

- Dữ liệu nhận có độ dài là 21.

- Số bit Parity trong chuỗi nhận được là p = 5, nằm ở các vị trí 1, 2, 4, 8, 16.

- Dữ liệu thật có độ dài d = 16, nằm ở các bit còn lại của chuỗi.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p1:**

- Ta lấy 1 bit và bỏ 1 bit bắt đầu từ p1, nên các vị trí lấy sẽ là 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21.

- Các giá trị bit lần lượt là: ? 1 1 1 0 0 1 1 0 0 1

- Có 6 bit 1 🡪 chẵn 🡪 p1 = 0 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ |  |  |  |  |  |  |  |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p2:**

- Ta lấy 2 bit và bỏ 2 bit bắt đầu từ p2, nên các vị trí lấy sẽ là 2, 3, 6, 7, 10, 11, 14, 15, 18, 19.

- Các giá trị bit lần lượt là: ? 1 0 1 1 0 1 1 0 0

- Có 5 bit 1 🡪 lẻ 🡪 p2 = 1 tuy nhiên p2 trong dữ liệu nhận lại là 0 🡪 SAI

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | S |  |  |  |  |  |  |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p4:**

- Ta lấy 4 bit và bỏ 4 bit bắt đầu từ p4, nên các vị trí lấy sẽ là 4, 5, 6, 7, 12, 13, 14, 15, 20, 21.

- Các giá trị bit lần lượt là: ? 1 0 1 0 1 1 1 1 1

- Có 7 bit 1 🡪 lẻ 🡪 p4 = 1 tuy nhiên p4 trong dữ liệu nhận lại là 0 🡪 SAI

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | S |  | S |  |  |  |  |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p8:**

- Ta lấy 8 bit và bỏ 8 bit bắt đầu từ p8, nên các vị trí lấy sẽ là 8, 9, 10, 11, 12, 13, 14, 15.

- Các giá trị bit lần lượt là: ? 0 1 0 0 1 1 1

- Có 4 bit 1 🡪 chẵn 🡪 p8 = 0 tuy nhiên p8 trong dữ liệu nhận lại là 1 🡪 SAI

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | S |  | S |  |  |  | S |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

**Xét p16:**

- Ta lấy 16 bit và bỏ 16 bit bắt đầu từ p16, nên vị trí lấy sẽ là 16, 17, 18, 19, 20, 21.

- Các giá trị bit lần lượt là: ? 0 0 0 1 1

- Có 2 bit 1 🡪 chẵn 🡪 p16 = 0 🡪 ĐÚNG

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đ | S |  | S |  |  |  | S |  |  |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |
| p1 | p2 |  | p4 |  |  |  | p8 |  |  |
|  |  | d3 |  | d5 | d6 | d7 |  | d9 | d10 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | Đ |  |  |  |  |  |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **1** |
|  |  |  |  |  | p16 |  |  |  |  |  |
| d11 | d12 | d13 | d14 | d15 |  | d17 | d18 | d19 | d20 | d21 |

Vậy p2, p4 và p8 sai. Do đó, vị trí dữ liệu sai là 2 + 4 + 8 = 14

🡪 d14 sửa 1 thành 0. Vậy dữ liệu gửi là 0010 1011 0100 1010 0001 1

Dữ liệu thật là 1101 0100 1010 0011.