**16-bit hamming code**

The following condition is to be satisfied:

2r >= r + m + 1

Where ‘ r ’ is the number of redundant bits (or) **parity** bits

where ‘m’ is the number of **data** bits

Parity bits are P(20 ) to P(2r-1)

And the remaining bits will be data bits D3, D5, D7, Di, …D(r+m) where i ≠ 2k

For 16 bit hamming code,

24 >= 4 +11+1

r = 4 , m = 11

Parity bits are P(20 ) to P(24-1) i.e., P1 , P2, P4,P8

Packet:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | P8 | D7 | D6 | D5 | P4 | D3 | P2 | P1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | P8 | D7 | D6 | D5 | P4 | D3 | P2 | P1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

r = 4 -> parity bits are 20,21,…..,2r-1

20 -> P1  22 -> P4

21 -> bit no 2 -> P2 23 ->P8

**ENCODING METHOD:**

|  |
| --- |
| Even Parity => Total number of 1’s bits to be even. |
| Odd Parity => Total number of 1’s bits to be odd. |

For example: data of 11 bits = 1010 0111 101

Parity bits are obtained form

P1 = XOR (D3, D5, D7, D9, D11, D13, D15)

P1 = XOR (1, 0, 1, 1, 0, 1, 1) = 1

P2 = XOR (D3, D6, D7, D10, D11, D14, D15)

P2 = XOR (1, 1, 1, 1, 0, 0, 1) = 1

P4= XOR (D5, D6, D7, D12, D13, D14, D15)

P4 = XOR (0, 1, 1, 0, 1, 0, 1) = 0

P8 = XOR (D9, D10, D11 D12, D13, D14, D15)

P8 = XOR (1, 1, 0, 0, 1, 0, 1) = 0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | P8 | D7 | D6 | D5 | P4 | D3 | P2 | P1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | **0** | 1 | 1 | 0 | **0** | 1 | **1** | **1** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | P8 | D7 | D6 | D5 | P4 | D3 | P2 | P1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Sender -> Encoded Data -> Receiver

ERROR DETECTION AT RECEIVER END

Actual data sent =1010 0111 0101

For example with noise data received = 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | P8 | D7 | D6 | D5 | P4 | D3 | P2 | P1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |

D14 is modified, we can check at receiver end using parity bits.

**EVEN PARITY**

P1=1 P1 = XOR (D3, D5, D7, D9, D11, D13, D15)

P1 = XOR (1, 0, 1, 1, 0, 1, 1) = 1

P2=1 P2 = XOR (D3, D6, D7, D10, D11, D14, D15)

P2 = XOR (1, 1, 1, 1, 0, 1, 1) = 0 **->** **ERROR DETECTED**

P4=0 P4= XOR (D5, D6, D7, D12, D13, D14, D15)

P4 = XOR (0, 1, 1, 0, 1, 01, 1) = 1  **->** **ERROR DETECTED**

P8=0 P8 = XOR (D9, D10, D11 D12, D13, D14, D15)

P8 = XOR (1, 1, 0, 0, 1, 1, 1) = 1 **->** **ERROR DETECTED**

ERROR CORRECTION AT RECEIVER END:

* Even parity (this is to be known for receiver to identify and correct errors)
* When there is contradiction , make parity bits = 1, else = 0.
* No of ‘1’ bits = Even for Even parity.

**Made of XOR** **NO of bits**

|  |  |
| --- | --- |
| **Make P8=1**  P8 -> D9, D10, D11, D12, D13, D14, D15 | 5 **->**  ODD **->**  Contradiction |
| **Make P4=1**  P4 -> D5, D6, D7, D12, D13, D14, D15 | 5 **->**  ODD **->**  Contradiction |
| **Make P2=1**  P2 -> D3, D6, D7, D10, D11, D14, D15 | 7 **->**  ODD **->**  Contradiction |
| **Make P1 =0**  P1 -> D3, D5, D7, D9, D11, D13, D15 | 6 **->**  EVEN **->**  No Contradiction |

|  |  |  |  |
| --- | --- | --- | --- |
| P8 | P4 | P2 | P1 |
| 1 | 1 | 1 | 0 |

14 bits has error

D14 = 1 **-> error bit**

D14 =0

Hence, noise bit is **REMOVED/CORRECTED**

**Generation of Parity Bits**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data bit number** | **Binary representation of bit number** | | | |
| D3 | 0 | 0 | 1 | 1 |
| D5 | 0 | 1 | 0 | 1 |
| D6 | 0 | 1 | 1 | 0 |
| D7 | 0 | 1 | 1 | 1 |
| D9 | 1 | 0 | 0 | 1 |
| D10 | 1 | 0 | 1 | 0 |
| D11 | 1 | 0 | 1 | 1 |
| D12 | 1 | 1 | 0 | 0 |
| D13 | 1 | 1 | 0 | 1 |
| D14 | 1 | 1 | 1 | 0 |
| D15 | 1 | 1 | 1 | 1 |

P1 = XOR (D3, D5, D7, D9, D11, D13, D15)

P2 = XOR (D3, D6, D7, D10, D11, D14, D15)

P8 = XOR (D9, D10, D11 D12, D13, D14, D15)

P4 = XOR (D5, D6, D7, D12, D13, D14, D15)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | P8 | D7 | D6 | D5 | P4 | D3 | P2 | P1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |