Lecture 7

Dr. Krishan Arora
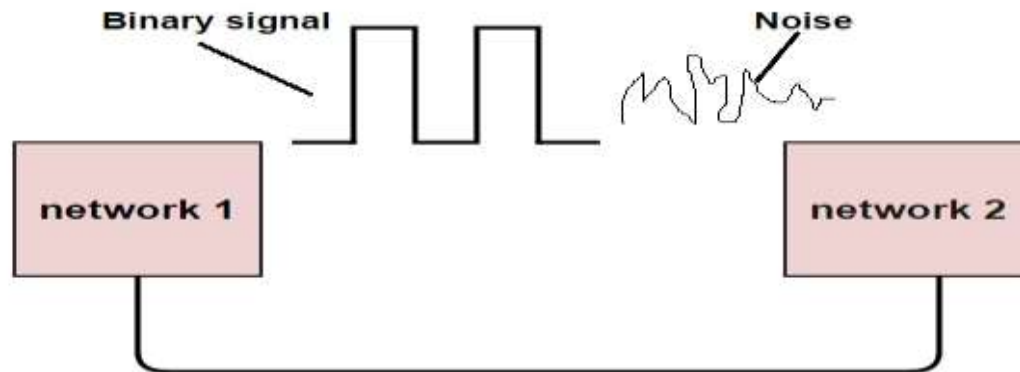**Asstt. Prof. and Head**

Error Detection Code

Error in digital system is the corruption of data due to external noise
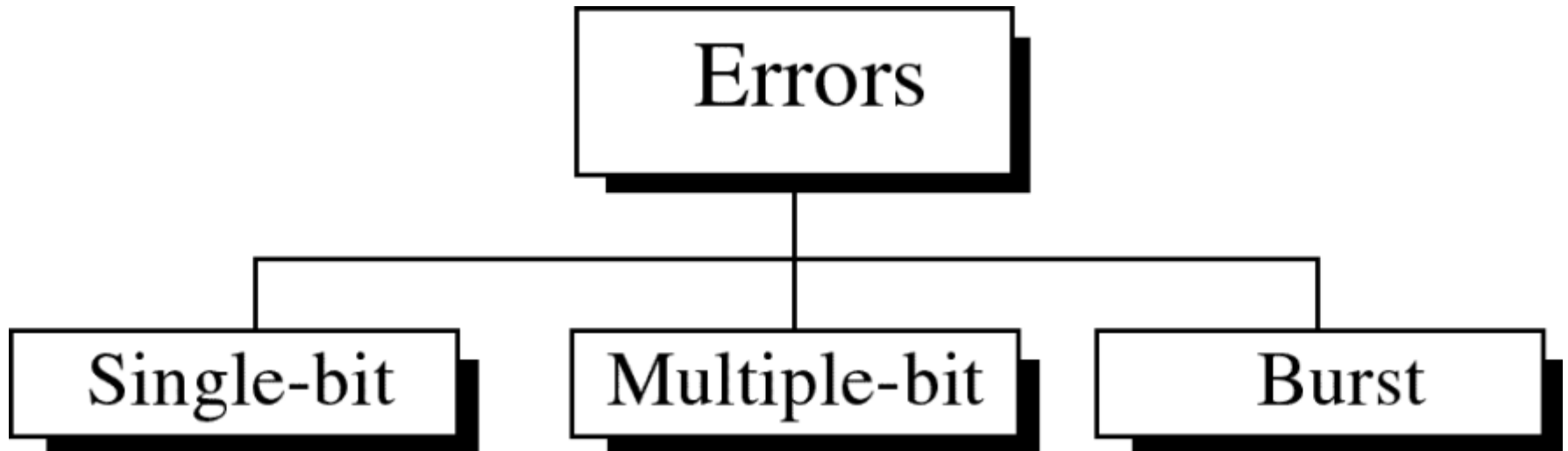
Failure may be temporary or permanent

If 1 is changed to zero or 0 is changed to 1, it is called "Bit error"

errors are transferred from one communication system to another, along with the data.
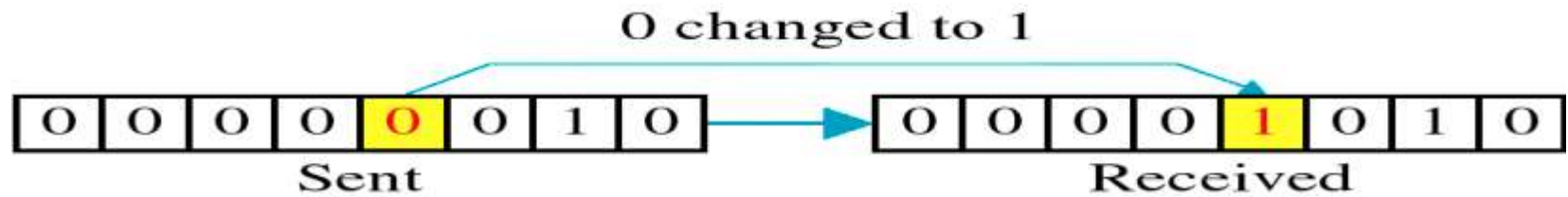
If these errors are not detected and corrected, data will be lost

Types of Errors

Single-bit error

0 changed to 1

| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | → | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

Sent   Received

Burst error

Sent

| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |

Burst error

| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |

Received

Two errors

| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | → | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

Sent   Received
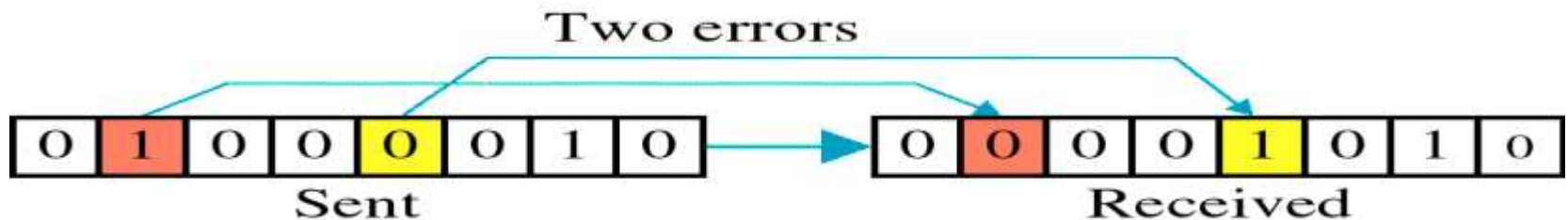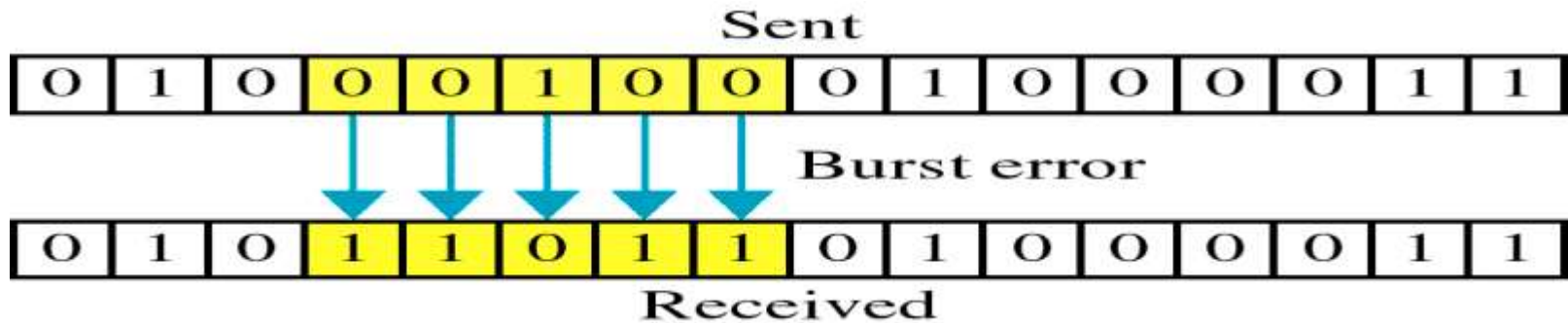
Parity Bit

Parity bit (P) is an extra bit included in the message to make the total number of 1 either even or odd

If ODD parity chosen P is such that total number of bit is odd

If EVEN parity chosen P is such that total number of bit is even

Parity bit generated during transmission end and send along with message

At destination parity of received message is check

If parity at received end is not same as transmission end implies at least one bit changed (bit error occurs)

3 bit data			Message with even parity		Message with odd parity	
A	B	C	Message	Parity	Message	Parity
0	0	0	000	0	000	1
0	0	1	001	1	001	0
0	1	0	010	1	010	0
0	1	1	011	0	011	1
1	0	0	100	1	100	0
1	0	1	101	0	101	1
1	1	0	110	0	110	1
1	1	1	111	1	111	0

Hamming code

- Hamming code is used to detect and correct the error in the transmitted data. So, it is an error detection and correction code. It was originally invented by *Richard W. Hamming* in the year 1950. Hamming codes detect 1-bit and 2-bit errors.

- While transmitting the message, it is encoded with the redundant bits. The redundant bits are the extra bits that are placed at certain locations of the data bits to detect the error. At the receiver end, the code is decoded to detect errors and the original message is received.

- **Selecting the number of redundant bits**

The hamming code uses the number of redundant bits depending on the number of information bits in the message.

Let n be the number of information or data bits, then the number of redundant bits P is determined from the following formula,

$$2^P \geq n + P + 1$$

For example, if 4-bit information is to be transmitted, then n=4. The number of redundant bits is determined by the trial and error method.

Let P=2, we get,

$$2^2 \geq 4 + 2 + 1$$

The above equation implies 4 not greater than or equal to 7. So let's choose another value of P=3.

$$2^3 \geq 4 + 3 + 1$$

Now, the equation satisfies the condition. So number of redundant bits, P=3.

- **Choosing the location of redundant bits**

For the above example, the number of data bits n=4, and the number of redundant bits P=3. So the message consists of 7 bits in total that are to be coded. Let the rightmost bit be designated as bit 1, the next successive bit as bit 2 and so on.

The seven bits are bit 7, bit 6, bit 5, bit 4, bit 3, bit 2, bit 1.

In this, the redundant bits are placed at the positions that are numbered corresponding to the power of 2, i.e., 1, 2, 4, 8,… Thus the locations of data bit and redundant bit are

D4, D3,D2,P3,D1,P2.P1.

Assigning the values to redundant bits

- Now it is time to assign bit value to the redundant bits in the formed hamming code group. The assigned bits are called a parity bit.

- Each parity bit will check certain other bits in the total code group. It is one with the bit location table, as shown below.

Bit Location	7	6	5	4	3	2	1
Bit designation	D4	D3	D2	P3	D1	P2	P1
Binary representation	111	110	101	100	011	010	001
Information / Data bits	D4	D3	D2		D1		
Parity bits				P3		P2	P1

- Parity bit P1 covers all data bits in positions whose binary representation has 1 in the least significant position(001, 011, 101, 111, etc.). Thus P1 checks the bit in locations 1, 3, 5, 7, 9, 11, etc..

- Parity bit P2 covers all data bits in positions whose binary representation has 1 in the second least significant position(010, 011, 110, 111, etc.). Thus P2 checks the bit in locations 2, 3, 6, 7, etc.

- Parity bit P3 covers all data bits in positions whose binary representation has 1 in the third least significant position(100, 101, 110, 111, etc.). Thus P3 checks the bit in locations 4, 5, 6, 7, etc.

- Each parity bit checks the corresponding bit locations and assign the bit value as 1 or 0, so as to make the number of 1s as even for even parity and odd for odd parity.

Example problem 1

Encode a binary word 11001 into the even parity hamming code.

Given, number of data bits, n =5.

**To find the number of redundant bits,**

Let us try P=4.

$$2^4 \geq 5 + 4 + 1$$

The equation is satisfied and so 4 redundant bits are selected.

So, total code bit = n+P = 9

The redundant bits are placed at bit positions 1, 2, 4 and 8.

Construct the bit location table.

Bit Location	9	8	7	6	5	4	3	2	1
Bit designation	D5	**P4**	D4	D3	D2	**P3**	D1	**P2**	**P1**
Binary representation	1001	1000	0111	0110	0101	0100	0011	0010	0001
Information bits	1		1	0	0		1		
Parity bits		1				1		0	1

To determine the parity bits

For P1: Bit locations 3, 5, 7 and 9 have three 1s. To have even parity, P1 must be 1.

For P2: Bit locations 3, 6, 7 have two 1s. To have even parity, P2 must be 0.

For P3: Bit locations 5, 6, 7 have one 1s. To have even parity, P3 must be 1.

For P4: Bit locations 8, 9 have one 1s. To have even parity, P2 must be 1.

Thus the encoded 9-bit hamming code is 111001101.

How to detect and correct the error in the hamming code?

- After receiving the encoded message, each parity bit along with its corresponding group of bits are checked for proper parity. While checking, the correct result of individual parity is marked as 0 and the wrong result is marked as 1.

- After checking all the parity bits, a binary word is formed taking the result bits for P1 as LSB. So formed binary word gives the bit location, where there is an error.

Example problem 2

Let us assume the even parity hamming code from the above example (111001101) is transmitted and the received code is (110001101). Now from the received code, let us detect and correct the error.

To detect the error, let us construct the bit location table.

Bit Location	9	8	7	6	5	4	3	2	1
Bit designation	D_5	P_4	D_4	D_3	D_2	P_3	D_1	P_2	P_1
Binary representation	1001	1000	0111	0110	0101	0100	0011	0010	0001
Received code	1	1	0	0	0	1	1	0	1

- *Checking the parity bits*

- For P1 : Check the locations 1, 3, 5, 7, 9. There is three 1s in this group, which is wrong for even parity. Hence the bit value for P1 is 1.

- For P2 : Check the locations 2, 3, 6, 7. There is one 1 in this group, which is wrong for even parity. Hence the bit value for P2 is 1.

- For P3 : Check the locations 3, 5, 6, 7. There is one 1 in this group, which is wrong for even parity. Hence the bit value for P3 is 1.

- For P4 : Check the locations 8, 9. There are two 1s in this group, which is correct for even parity. Hence the bit value for P4 is 0.

- The resultant binary word is 0111. It corresponds to the bit location 7 in the above table. The error is detected in the data bit D4. The error is 0 and it should be changed to 1.

- **Thus the corrected code is 111001101**.

Quick Quiz (Poll 1)

Why do we require hamming codes?

    a) Error correction
    b) Encryption only
    c) Decryption
    d) Bit stuffing