Error Control

Lecture => 1

Receiver	Sender
Received data (10110010)	Transmitted data 10110010.

If Received data is not same as "Transmitted data" then "chance of error".

Error :— Corrupted data [flipped data bits]

Type of error :—

1> Single bit error
2> Burst error. (Multiple bit error)

1> Single bit error :—

Only One bit in the received data has changed.

Transmitted data = 10110010
Receiver data = 10111011

2> Burst error :—

More than one [two or more] bit in the received data have changed.

Burst length :— length from first corrupted bit to the last corrupted bit [inclusive]

* ~~Math for Machine learning~~ :—

Transmitted data = 10110010.
Received data = 11110100
$\xleftrightarrow{\quad 6 \quad}$

In case of burst error
total number of corrupted data bits is less than equal to Burst length.

Q Consider ASCII character "A" is transmitted by transmitter, but ASCII character "D" is received by receiver. Calculate ~~the~~ burst length?

Soln ⇒ A = 65 , D = 68

0100 0001.
0100 $\underbrace{0100}$ Ans = 3

* Based on Redundant Bits
[Parity bits or extra bits]

1> Error detection
2> Error detection and correction.

1> ~~Error deet~~ **Error detection :—**

→ Can only detect error (s)
→ Not able to correct
→ Retransmission of corrupted data.

Two error detection technique :—
1> Cyclic Redundancy Check (CRC)
2> Checksum.

2> **Error detection and Correction :—**

→ Can detect as well as Correct error (s)
→ forward error correction.

Two error detection and correction technique :—

1> 2D Parity
2> Hamming Code.

* **One-bit parity :** block (n bits)

Data	
(n-1) bit	One bit

Transmitter protocol :—

1) **Even parity :—** if number of one's in the data is even than transmitter set parity bit "zero"

else

set parity bit "one"

2) **Odd parity :—**

→ reverse of Even parity.

Q Suppose " Even parity "
DATA = "1011101"

Ans ⇒ 1011101 $\underset{P}{1}$.

Receiver Protocol :—

1) **Even Parity :—**
if receiver find numbers of one in the received block is even (including parity) then reciever concluded "no error detected"

else
receiver concluded "error detected"

2> odd parity :– (Reverse)

Case-1 :– No any error

Data = "1011101"
Transmitted Data = 10111011
Received Data = 10111011

Receiver Concluded :– No any error detected, accept the data.

Case-2 One-bit error :–

Data = "1011101"

Transmitted Data = 10111011
Received Data = 10101011.

Receiver Concluded : Error detected, reject the data.

Case-3 Two-bit error :–

Data = "1011101"
Transmitted data = 10111011.
Received data = 10011111

Receiver Concluded :– No any error detected, accept the data.

Case -> IV **Three Bit error** :—

Data = "1011101"

Transmitted Data = 10111011.
Received Data = 10100111.

Receiver Concluded :— Error detected, reject the data.

Q. let suppose, even parity is used in one-bit parity error detection technique. If receiver find total 295 one's in the received block (including parity) then what receiver concluded.

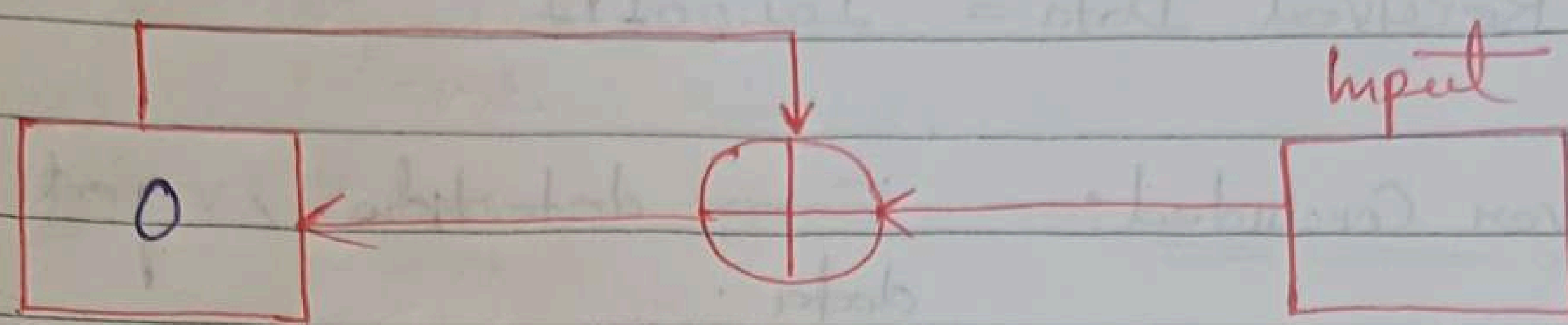
A> No any error detected
B> Error detected
C> Unable to detect error
D> Data insufficient.

*

Suppose even parity

DATA = " 1 0 1 1 1 0 1"
$d_1\ d_2\ d_3\ d_4\ d_5\ d_6\ d_7$

Transmitted data = 1 0 1 1 1 0 1 1
P



0	d_1 = 1
1	d_2 = 0
1	d_3 = 1
0	d_4 = 1
1	d_5 = 1
0	d_6 = 0
0	d_7 = 1
1	P = 0
P = 1	

AT Receiver

Input = " 1 0 1 1 1 0 1 1 "
$d_1\ d_2\ d_3\ d_4\ d_5\ d_6\ d_7\ P$

if Result == zero
then Receiver concluded
No any error detected
else
Receiver concluded
" Error detected "

0

0	0	d_1	= 1
0	1	d_2	= 0
0	1	d_3	= 1
1	0	d_4	= 1
0	1	d_5	= 1
1	0	d_6	= 0
1	0	d_7	= 1
0	1	P	= 1

Result = 1 0

* Block Code :—

K-bit input $\longrightarrow$ n bits output

no. of parity bit in each codeword $= (n-k)$

k = input data bits

n = code length

$\longrightarrow$ 2^k codewords of length n

[Codewords = data with parity]

One bit parity (with even parity) and 3 data bits

$d_1 d_2 d_3$	$\longrightarrow$	$d_1 d_2 d_3\ P$
0 0 0		0 0 0 0
0 0 1		0 0 1 1
0 1 0		0 1 0 1
0 1 1		0 1 1 0
1 0 0		1 0 0 1
1 0 1		1 0 1 0
1 1 0		1 1 0 0
1 1 1		1 1 1 1

$\rightarrow$ Block Code

set of ($2^k = 8$) Codeword

linear code

$\downarrow$

Cyclic Code

* **Linear Code :—**

Linear Combination of codewords.

if c_i and c_j are codeword in set
then c_k is also be codeword in that set
where $c_k = c_i \oplus c_j$

$$\begin{array}{r} 0110 \\ 1010 \\ \hline 1100 \end{array}$$

$$\left\{ \begin{array}{l} 0+1=1 \\ 1+0=1 \\ 0+0=0 \\ 1+1=0. \end{array} \right.$$

* **Cyclic code :—**

→ Cyclic Combination of linear codewords

if c is codeword
then bit-wise cyclic left or right shift on
C is also be a codeword.

$$c_1 c_2 c_3 c_4 \xrightarrow{\text{left shift}} c_2 c_3 c_4 c_1$$

$$c_1 c_2 c_3 c_4 \xrightarrow{\text{cyclic Right shift}} c_4 c_1 c_2 c_3$$

Valid Codewords V/s Invalid Codewords

* **Valid Codewords** :—

0000 0011 0101 0110 1001 1010 1100 1111

* **Invalid Codewords** :—

0001, 0010, 0100, 0111, 1000, 1011, 1101, 1110.

→ Transmitter always transmit valid codeword in the channel.

→ If receiver receives a codewords from channel, which is belongs to valid code set then receiver concluded "no error detected".

→ If receiver receives a codeword from channel, which is belong to invalid codeword set then receiver conclude "error detected".

Transmitter → **Receiver**

$\boxed{\text{Valid codeword}}$ → Invalid codeword

1) 3-bit error
2) single bit error

$\boxed{\text{Valid codeword}}$ → valid codeword.

1) No any error
2) Two bit error