Name: Shubham Chemate

Roll Number: 31118

Subject: CNS Lab Assignment 03 Writeup and performance screenshots



Subject: Computer Networks & Security



(PICT)	Assignment: 03
PICT, PUNE	
PICI, FORE	
-	Title: Error detection
	- 11
- thus	Isoblem Statement:
-	Write a program for error detection of correction for
	7/8 bits ASCII codes using Hamming codes @
	CRC
	Requirement
to bah	DE: Eclipse (version: 2020 release)
	compiler: que (version: 6.3,0)
	Hardware: Intel 15-8265 U CPU@ 1.60 GH2
Laura	8GB RAM 4 5/2 GB SSP.
	All and the same parties of the same and the
	pescription
	In + hababa was small had all and a
	· Pata Link Layer:
_ sah as _	- In OSI model, data link layer is the 4th layer
	From top 4 2nd from bottom.
	- Pata link layer gets the raw data from physical
	layer try to make physical link reliable.
	- If attempts to defect of recover the data that
	It gets from physical layer.
- Atal	- pata link layer protocol defines the format of
113 <u>41222 122 3</u>	the packet exchanged across nodes, errole defection, retransmission, flow control & random acess.
- DAM	TETUTATION FOR CONTROL TO THE TETUTATION OF THE
	services provided by data link layer:
	i>framing & Link Access:
1615	is data link layer takes parkets from network
	layer & encapsulate them as frames



PICT, PUNE	
	11) It sends each frame bit-by-bit on the
	hardware.
	11/2 At receivers end, it assembles but into
ten for	frames.
Ala Company	PAVOREN PRINT EDIA 250 STE
	2) Addressing
	y provides layer-2 hardware addressing
	mechanism
	11) Hardware address is unique & provided at
	the time of manyfadwing
	a) C) with a state of a state of a state of
	3) Synchronization: marhines should synchronized prohile sending message on link
	Duje sending mersage on inve
	4) Error Control: Froms are detected of attempt is
	made to recover actual bits.
AND MAIN	provides error reporting mechanism to sender.
	From top & 204 From portion
danger	5) Flow control machines on some link may have
910	different speed 1 capacity pata link layer
tot el	ensures flow control.
	A A A A A A A A A A A A A A A A A A A
	6) Multi-Access: To award reduce collisions data
2001-010-010	link layer provides mechanism such as CSMA/IP
	to equip capability of accessing a shared
	media among multiple systems.
	Types of Errors!
X 00.30	There are three types of errors
	LIBERT TO THE PROPERTY OF THE PARTY OF THE P



PICT BLINE

	Dragram	Description. > Only one -bit of given data-unit is corrapted.
2> Burst Error		odata transmission Two more bits in data unit are corrupted
	corrupted counted bit	From first corrupted bit to last. 11) generally happen in serial data transmission.

Methods of every detection & correction:

error detection: Involves checking wheather any error has occurred @ not.

error correction. Involves ascertaining the exact number of bits that has been corrupted of location of corrupted bits.

15 ros detection techniques:



	.1	
	Name	description
- WW. 19-10-	17 parity check.	Adone by adding extra bit called painty bit is there are two types even parity &
		is there are two types even parity &
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Dilay Lagravisor	oda parity
N 200 30	Manual Dogo !	18) sonder courts no of ones f make parity estues even odd by adding
- takens	Pare Copera Ce	parity either even odd by adding
NAME OF THE PARTY	Attant arv	extra 1.
		10) on seceiver end is party is some
a later	a Significant of	as sender then message is anepted
2.0	o day ptob	
		1) Data is divided into fixed saed frames
tan regersor	2> checksum	@ segments.
base as		is sorder: adds the segment using is
	TO SI TO	complement arithernatic to get sum
	A phoops at	They it complements sum to get
		checksum 4 sends it along data frames
		111) Receiver adds incoming segment along
		with the checksym using is complements
	Ne (13342)	Then complements it.
		1 >> If result is sero, data is accepted
25,3411	an saataasia la	and a conformation of the
3	> Cyclic ,	> It involves kinary dwision of data bits.
	Redundancy	by predetermined duism which is
		generaled wing polynomials
		> At sender side after division, remainder
		is appended at the end of data segment
		of Ab receiver end, incoming data unit
		is divided by divisor. If there is o
		remainder dota segment gets auepted



PICT, PUNE		
	Error Correction Techniques	
	Technique.	Description
	Backward Error	is peceived request the sender to
	correction	repronsmit frame if he detects
	Maria Santa Maria	erros.
		1) It is simple technique which
		is efficient when retransmussion
	The state of the s	Is not expensive.
	20	
	>> forward from	13 Receives executes error correcting
	Defection Correction	code if he detects error.
-	D Page Intoh	my This saves bandwidth requirement
		for retransmussion
		ns If there are too-many errors
	-	retransmission is needed.
		112 - 1212 - 121
bulsy	Error correction co	ocles:
	>Hamming (od	es.

	Con a dome I a l
- Johnson	Error correction codes:
	i>Hamming Codes.
	e> Aingry convolution code
	3> Reed. Solomon lode
	1> low. density parity-check code.
4	有种的,然后,我们就是一种的人,我们就是一个人的人,我们就是一个人的人的人,不是一个人的人的人的人,不是一个人的人的人的人的人的人的人,不是一个人的人的人的人

here. 2 = redundant bits m = message length 3) Party Bits. Party bits are appended to make party of 1's either even @ odd. For our analysis let's tensider oven party from now. Mote that: even party: no of bists should be even. For odd 1's extra 1 should be added odd party: no of 1's should be added odd for even 1's extra 1 should be added odd party: no of 1's should be added odd for even 1's extra 1 should be added odd party: no of 1's should be added odd for even 1's extra 1 should be added odd for ev	(PICT)	
here, 2= redundant bits m= message length 3> Party Bits Party bits are appended to make party of 1's either even & odd. For our analysis let's consider oven party from now. Mote that, even party: no of kibits should be even. For odd 1's extra 1 should be added odd party: no of 1's should be added odd party: no of 1's should be added *Algorithm: The calculate the value of 2 = redundant bits by formula 2° > dfrtt i d = length of data word. of effective length of codeword will be n = red should be pipz, as follows: p; = consider all the bits positions whose binary representation includes 1 in least pos significant eg. (1, 2, 5, 7, 19,) p = consider all the bits positions whose binary representation includes 1 in second i position from USB.	PICT, PUNE	
** redundant bits m= message length 3) Party Bits Party bits are appended to make party of 1's either even & odd. For our analysis let's consider oven party from now. Mole that, even party: no of libits should be even. For odd 1's extra 1 should be added odd parity: no of 1's should be added odd parity: no of 1's should be added **Algorithm: 'y calculate the value of **e-redundant bits by formula 2° > dtrtt i d = length of data world. 2) Effective length of codewood will be n= 2+d) where let party bits be 1', 1', 1', 1', 1', 1', 1', 1', 1', 1',		
m= message length 3) Pearly Bits. Party bits are appended to make party of 1's either even 60 odd. For our analysis let's consider oven party from now. Mote that: even parity: no of kibits should be even for odd is extra 1 should be added odd parity: no of 1's should be added odd parity: no of 1's should be added Algorithm: 7) Calculate the value of 2 = zedundent bits by formula 2²> difett i d = length of data word. 2) Effective length of codeword will be n = 2+d 3) where let parity bits be 1°, 1°, 1°, 1°, 1°, 1° (alculate pi 1°, 2 as follows: p1 = consider all the bits positions whose binary representation includes 1 in least pas. Significant eg. (1,3,5,7,9,) p. = consider all the bits positions whose binary representation includes 1 in second position from USB.		
Party Bits. Party bits are appended to make party of 1's either even @ odd. For our analysis let's consider oven party from now. Mole that: even party: no. of libits should be even. For odd 1's extra 1 should be added odd party: no of 1's should be added. Algorithm: The Calculate the value of # = redundant bits by formula 22 ditt! I d = length of data word. Peffective length of codecoold will be not represented in the bits positions whose binary representation includes 1 in least passition whose binary representation includes 1 in least position whose binary representation includes 1 in second position from usp. Preserved and the bits positions whose binary representation from usp.		
Party bits are appended to make party of 1's either even @ odd. For our analysis let's consider oven party from now. Mote that: even party: no of kibits should be even. For odd is extra 1 should be added odd party: no of I's should be added even is extra 1 should be added to a even is extra 1 should be added to Algorithm: 'Algorithm: 'Y calculate the value of e = redundent bits by formula 2° > dtrt1 if d = length of data word. 2) Effective length of codecoold will be not even to even bits be 1', 1', 1', 1', 1', 1', 1', 1', 1', 1',		
either even @ odd. For our analysis let's consider oven party from now. Mote that: even party: no. of libits should be even. For odd I's extra 1 should be added odd party: no of I's should be added odd party: no of I's should be added Algorithm: The factority of the should be added 2 Algorithm: The factority of data world. 2 Effective length of codeworld will be n = 2 edundent bits by formula 22 ditett d d = length of codeworld will be n = 2 et as follows: p = consider all the bits positions whose binary representation includes I in least positions escent position from use. P2 = As we are considering for even party set q		
Mote that: Even parity: no. of kbits should be even. For odd 1's extra 1 should be added. odd parity: no of 1's should be added. odd parity: no of 1's should be added. Palgorithm: Pa		
Mote that: even parity: no. of kbits should be even. For odd 1's extra 1 should be added. odd parity: no of 1's should be added. odd parity: no of 1's should be added. **Algorithm: i) Calculate the value of **e = redundant bits by formula 2° > dtrtt i d = length of data word. ?) Effective length of codeword will be **n = 2+d **y where let parity bits be 1', 1', 1', 1', 1', 1', 1', 1', 1', 1',		
even parity: no. of libits should be even. For odd 1's extra 1 should be added odd parity: no of 1's should be added even 1's extra 1 bits by formula 2°> ditett 1 d = length of data word. 2) Effective length of codeword will be n = 2+d 3) where let parity hits be 1', 1', 1', 1', 1', 1', 1', 1', 1', 1',		
odd party: no of 1's should be added odd party: no of 1's should be added even 1's extra 1 should be added Algorithm: i) Calculate the value of e = redundant bits by formula 2° > dtrtt i d = length of data word. 2) Effective length of codeword will be n = 2+d s) where let party bits be 1', 1', 1', 1', 1', 1', 1', 1', 1', 1',		
odd party: no of 1's should be odd. for even 1's extra 1 should be added Algorithm: i) Calculate the value of e=redundant bits by formula 22 ditett d d e length of data word. 2) Effective length of codeword will be n=red s) where let party bits be p. p. p. p. calculate p. p. as follows: p. = consider all the bits positions whose binary representation includes 1 in least pos. significant eg. (1, 3, 5, 7, 9,) p. = consider all the bits positions whose binary representation includes 1 in second position from us. Presented As we are considering for even party set q	1	ndd is entre 1 should be even. For
even 1's extra 1 should be added Algorithm: i) Calculate the value of i = redundant bits by formula 2° > dfrit i d = length of data word. 2) Effective length of codecoold will be N = retd s) where let parity bits be 1', 1', 1', 1', 1', 1', 1', 1', 1', 1',		
Algorithm: 2 = redundent bits by formula 22 > difett 2 = redundent bits by formula 22 > difett 2 d = length of data word. 2) Effective length of codeword will be 2 = retd 3) where let parity bits be 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,		ences is entra I should be added
is calculate the value of 2 = redundant bits by formula 22 > dfrt1 i d = length of data word. 2) Effective length of codeword will be n = retd s) where let party bits be fire, fr, fr, fr, fr, fr calculate profit as follows: p = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1,3,5,7,9,) p = consider all the bits positions whose binary representation includes 1 in second position from LSB.		
2 = redundent bits by formula 22 > dfrt d = length of data word. 2) Effective length of codewood will be 11 = 2+d 3) where let party bits be fi, fr, fr, fr, fr (alculate pi, fr, as follows: p, = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1, 3, 5, 7, 9,) p. = consider all the bits positions whose binary representation includes 1 in second position from usp. Pr = As we are considering for even parity set q		
2) Effective length of codecoold will be N=2+d s) where let party bits be P, P2, P3, P9. calculate p1, P2, as follows: p1 = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1, 3, 5, 7, 9,) p2 = consider all the bits positions whose binary representation includes 1 in second position from USB. P2 = The are considering for even party set q		
2) Effective length of codeword will be N = 2+d 3) where Let party bits be P, P2, P3, P4. calculate p1, P2, as follows: p1 = consider all the bits positions whose binary representation includes 1 in least poss significant eg. (1, 3, 5, 7, 9,) p2 = consider all the bits positions whose binary representation includes 1 in second position from USB. P2 = 4) As we are considering for even parity set q		& d = length of data word.
s) where let party bits be P, P2, P3, P2 calculate p1, P2, as follows: p, = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1,3,5,7,9,) p. = consider all the bits positions whose binary representation includes 1 in second position from LSB. P2 = 4) As we are considering for even parity, set q		
calculate prips as follows: p, = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1,3,5,7,9,) p. = consider all the bits positions whose binary representation includes 1 in second position from LSB. Pr = As we are considering for even parity, set q		
calculate prips as follows: p, = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1,3,5,7,9,) p. = consider all the bits positions whose binary representation includes 1 in second position from LSB. Pr = As we are considering for even parity, set q		3> where let party bits be P. Pz. Pz, Pz.
p, = consider all the bits positions whose binary representation includes 1 in least pas. significant eg. (1,3,5,7,9,) p. = consider all the bits positions whose binary representation includes 1 in second position from LSB. Presentation for even parity, set q		calculate prips as follows:
representation includes I in least pas. significant eg. (1,3,5,7,9,) p. = consider all the bits positions whose binary representation includes I in second position from LSB. Pr = Pr =		p, = consider all the bits positions whose binary
p. = consider all the bits positions whose binary representation includes 1 in second position from LSB. Pre Pre are considering for even parity, set q		representation includes 1 in least pas.
position from LSB. Pr = Pr are considering for even parity, set q		significant eg. (1,3,5,7,9,)
Pr = Pr = .		p. = eonsider all the bits positions whose.
Pr = Pr = .		binary representation includes 1 in second
4) As we are considering for even parity, set q		position from LSB.
4) As we are considering for even parity, set q		
parity bit to 1 if the total number of is		
pacity bit to 1 if the total number of 1's		4) As we are considering for even parity, set q
		party bit to 1 if the total number of 15



PICT, PUNE	
	in the positions it checks is odd.
	5) Place these redundant bits at the positions of
	powers of 2 1e. 1,2,4, etc. other positions will
	be occupied by data hits.
11 10	Sure a semand whole approximate and the semantic services are semantic services are semantic services are semantic services and the semantic services are semantic semantic services are semantic sem
	At receivers end:
	17 Calculate the values of fi, B Pr.
	if they are of opposite parity error exist. 2) Calculating the position of corrupted bit.
	>> calculating the position of corrupted bit
	convert the value of (Pa Ry. P) to
	deumal.
Collable	This gives the Possible of curriapted bit.
	example:
S Helphan	Even parity
	consider the message: 110/011
7	here d=7
	using formula 227/d+2+1
31	ne get 2=4.
	zorottot in ing stolerko
uspoid 2	hence our message length will be 2+d=11
	1 1 1 0 1 0 1 0
9,000	P1 P2 d3 P4 dr d6 d7 P8 dg d10 d11
	consider!
	p= d3 d5 d7 d9 d1 = 10100 = 1
	p= d2 ded7 dod1= 10100dd 15 = 1
13 - 2 - 2	and a de de de de la
	$P_{q} = d_{5}d_{6}d_{7} = [0] = 0$ $eve_{1} 1's'$
	$P_8 = dg d_0 d_{11} = 0 0 = 1$
	18-379011-010-4



PIGT	
PICT, PUNE	
	our coderoord to send is:
<u> </u>	1111010110110
	on teceivers end:
	let we received: 110101010
	here 1= 111100 = 0
	erey 1's
	P ₂ = 0 10 = 0
	q = 0 0 = 0 $ q = 0 0 = 0$
	18 = 1010 = 0
	As all parity bits satisfy parity condition,
	the received message is an epted.
	let we received 11100011010
	here p_= 110100 = 1
	$p_2 = 0 0 = 0$
	p= 0001 = 1
	Pg = 1010 = 0
	As bits p, & P2 violates parety condition
	these must be an error in received message
	Error pasition!
	consider desimal representation of P8 Pals P;
	1e. 0/01 which is 5.
	Herrie en bit-5 is corrupted.
	Correct ensg. should be 11/0/01/010.
	O

PICT, PUNE	
	Conclusion.
	In this assignment, we have learned about
	various methods of error detection & correction
	in data link layor along with their types.
	Also implemented program for Hamming code error correction & detection with the help of
	our lab guide
	0 = 00 1/11 = 11 1/21
	0 = 0.110 H = 1
	0 = 1010 = 7
	TO SECTION SEC
	en unione of the said plans the said to
	24 (4 2) 1
	O TO HE O CONTRACTOR OF THE OWNER
	4 = 0010 (1 = 0 No.)
	O = OROH = A
	The second secon
	DUDING THE SEASON AND AND DESCRIPTION OF THE SEASON OF THE
	GO ASTANDADOS, CAR LOUVIND AND LIESTED TO THE CONTROL OF THE CONTR
	Description of School Section 1997
	Market I and the state of the s

```
**Welcome to Hamming Code Error Detector and Corrector created by Shubham.**

Enter the message: pict

Message Transfer is started...
```

```
---->Tranferring: p
====This is senders end
Choose Parity:
       1. for Odd Parity.
       2. for Even Parity.
Encoding Your Message with Odd Parity...
Number of redundant bits used are: 4
Encoded Message is:
1 1 0 1 0 0 0 0 1 1
p1 p2 d3 p4 d5 d6 d7 p8 d9 d10 d11
Encoded Successfully.
Your Message is Being Transferred to Receiver. Plz Wait..
====This is Medium====
Is any bit corrupted (if yes enter pos else enter -1)
: -1
This is receiver End====
Calculated value of parity bits:
0 0 0
p1 p2 p4 p8
No Errors detected. Succeessfully Accepted.
Received Character: p
```

Screenshot-1: Odd Parity with no error

```
---->Tranferring: i
====This is senders end
Choose Parity:
       1. for Odd Parity.
       2. for Even Parity.
: 1
Encoding Your Message with Odd Parity...
Number of redundant bits used are: 4
Encoded Message is:
0 1 1 0 0 0 1
                         1 0
p1 p2 d3 p4 d5 d6 d7 p8 d9 d10 d11
Encoded Successfully.
Your Message is Being Transferred to Receiver. Plz Wait..
====This is Medium====
Is any bit corrupted (if yes enter pos else enter -1)
: 7
This is receiver End====
Calculated value of parity bits:
1 1 1
p1 p2 p4 p8
Corrupted bit found at pos: 7
Correcting the corrupted bit...
Received Character: i
```

Screenshot-2: Odd Parity with error

```
---->Tranferring: t
====This is senders end
Choose Parity:
       1. for Odd Parity.
       2. for Even Parity.
Encoding Your Message with Even Parity...
Number of redundant bits used are: 4
Encoded Message is:
       0
           1
                   1
                      0
                          1
                              1
   1
              0
p1 p2 d3 p4 d5 d6 d7 p8 d9 d10 d11
Encoded Successfully.
Your Message is Being Transferred to Receiver. Plz Wait..
====This is Medium====
Is any bit corrupted (if yes enter pos else enter -1)
: -1
This is receiver End====
Calculated value of parity bits:
0 0
       0
           0
p1 p2 p4 p8
No Errors detected. Succeessfully Accepted.
Received Character: t
```

Screenshot-3: Even Parity with no error

```
----->Tranferring: c
====This is senders end
Choose Parity:
       1. for Odd Parity.
       2. for Even Parity.
: 2
Encoding Your Message with Even Parity...
Number of redundant bits used are: 4
Encoded Message is:
1 1 1 1 1 0
                      0
                          0
p1 p2 d3 p4 d5 d6 d7 p8 d9 d10 d11
Encoded Successfully.
Your Message is Being Transferred to Receiver. Plz Wait..
====This is Medium====
Is any bit corrupted (if yes enter pos else enter -1)
: 10
This is receiver End====
Calculated value of parity bits:
0 1 0
           1
p1 p2 p4 p8
Corrupted bit found at pos: 10
Correcting the corrupted bit...
Received Character: c
```

Screenshot-4: Even Parity with error

```
****Your Message is Successfully Decoded****
Decoded Message is: pict

Error Detection and Correction Process is completed.

Do you want to check again?
: 0
Thank You!
```

Thank You!