Lab Report

LAB — 05

CSE — 206

Presented By:

- **Name:** Tunazzinur Rahman Kabbo
- **Intake:** 44
- **Sec:** 07
- **ID:** 19202103268

CSE — 206

Presented To:

❖ **Iffat Tamanna**
Lecturer, BUBT
Department of Computer Science & Engineering
Email: iffat@bubt.edu.bd

Lab-05

Name of the experiment: Design, construct and test combination logic circuits like parity generator.

Explanation:

A parity bit, or check bit, is a bit added to a string of binary code. Parity bits are a simple form of error detecting code. Parity bits are generally applied to the smallest units of a communication protocol, typically 8-bit octets, although they can also be applied sparcately to an entire message string of bits. The parity bit ensures that the total number of 1-bits in the string is even or odd. There are two variants of parity bits: even parity bit, odd parity bit.

Even parity: Here all the total number of bits in the message is made even.

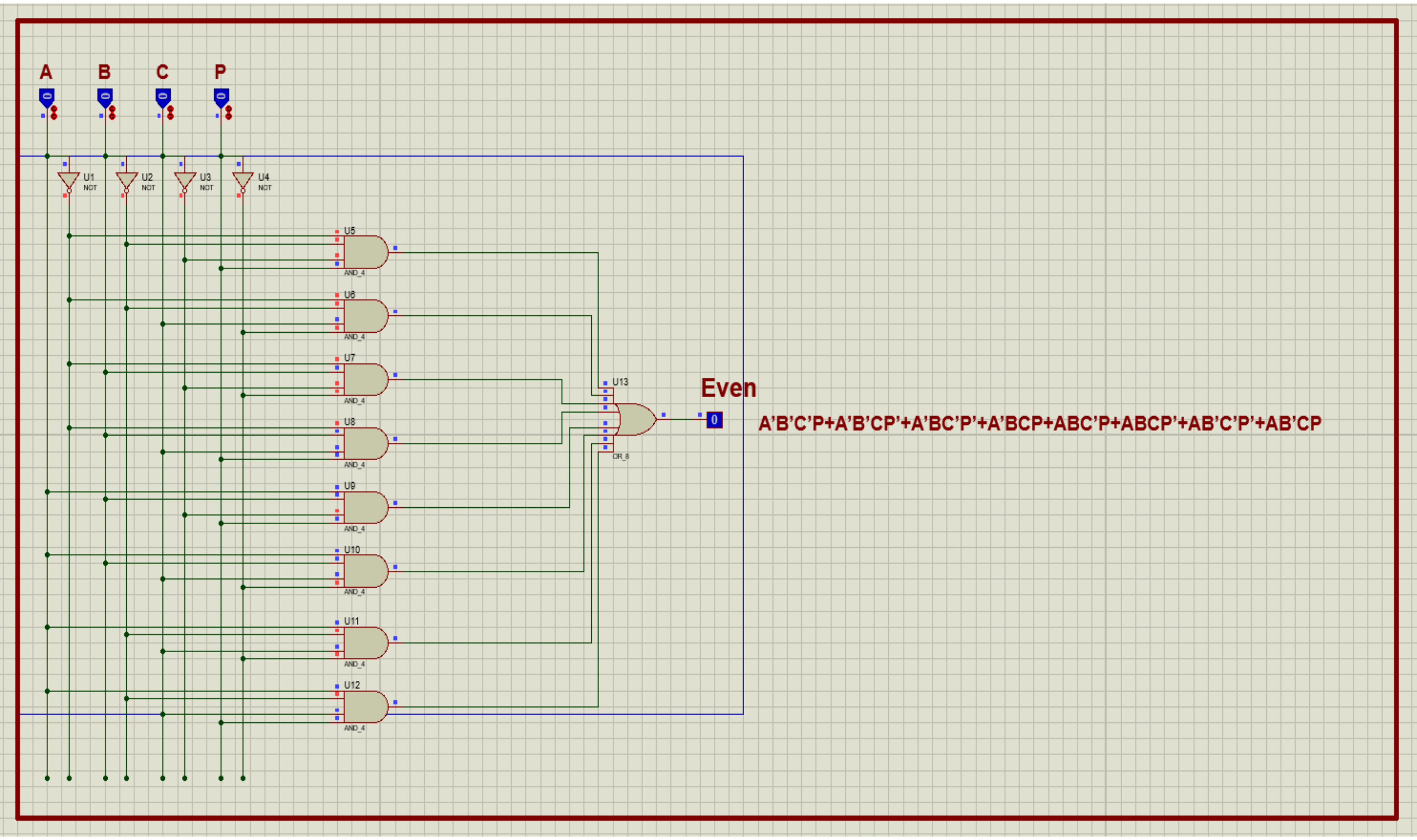
Truth table:

A	B	C	P	Result
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

K-Map:

AB \ CP	00	01	11	10
00	0	1	0	1
01	1	0	1	0
11	0	1	0	1
10	1	0	1	0

$$F = A'B'C'P + A'B'CP' + A'BC'P' + A'BCP + ABC'P + ABCP' + AB'C'P' + AB'CP$$


A
B
C
P
U1
NOT
U2
NOT
U3
NOT
U4
NOT
U5
AND_4
U6
AND_4
U7
AND_4
U8
AND_4
U9
AND_4
U10
AND_4
U11
AND_4
U12
AND_4
U13
OR_8
Even
0
A’B’C’P+A’B’CP’+A’BC’P’+A’BCP+ABC’P+ABCP’+AB’C’P’+AB’CP

Odd parity: Here the total number of bits in the message is made odd.

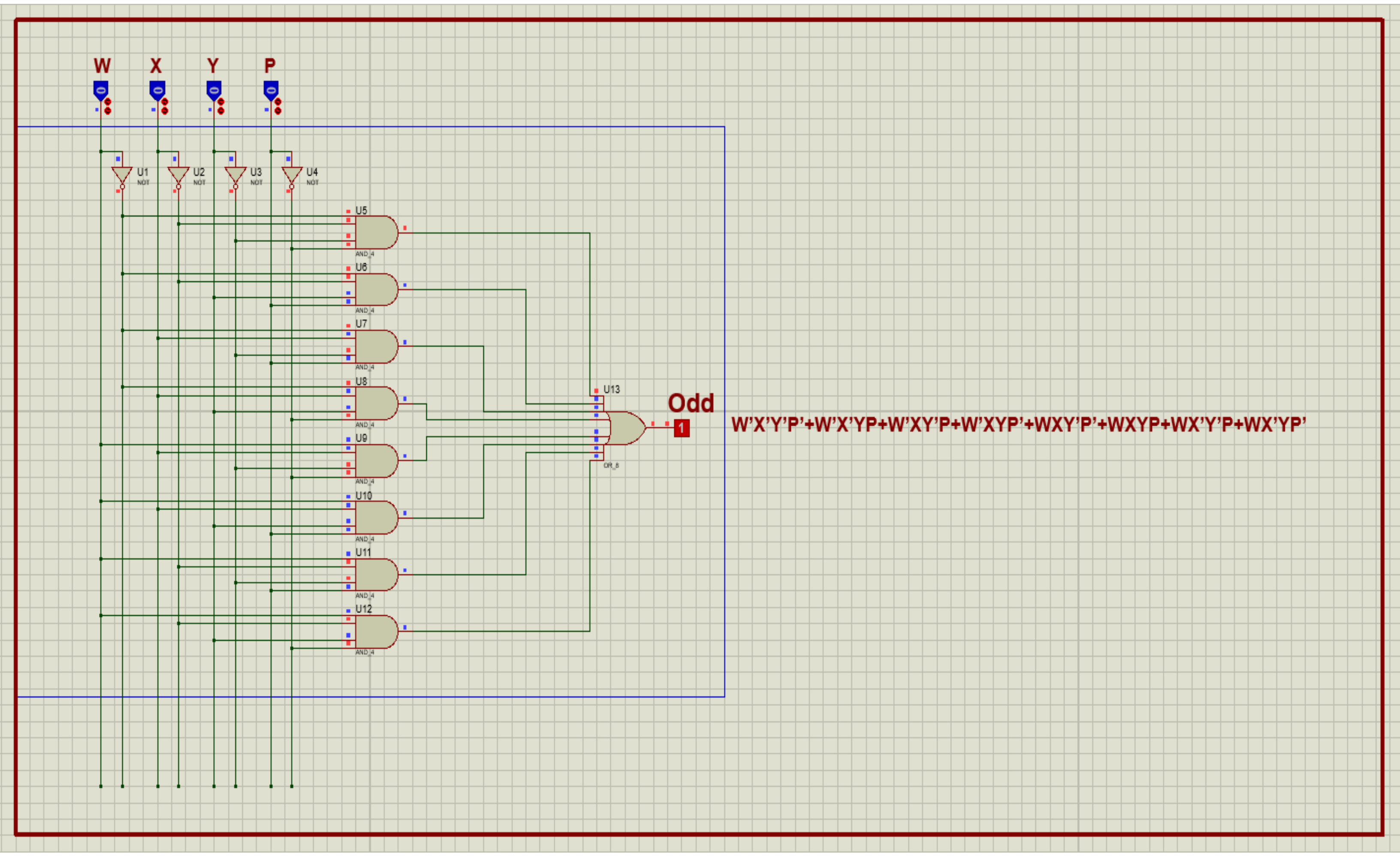
Truth table:

W	X	Y	P	Result
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

K-Map:

WX \ YP	00	01	11	10
00	1	0	1	0
01	0	1	0	1
11	1	0	1	0
10	0	1	0	1

$$F = W'X'Y'P' + W'X'YP + W'XY'P + W'XYP' + WXY'P' + WXYP + WX'Y'P + WX'YP'$$


W
X
Y
P
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
Odd
W’X’Y’P’+W’X’YP+W’XY’P+W’XYP’+WXY’P’+WXYP+WX’Y’P+WX’YP’

Name of the experiment: Design of code converters like BCD to Excess-3

Description: Excess-3 binary code is an unweighted self-complementary BCD code. Self-complementary property means that the 1's complement of an excess-3 number is the excess-3 code of the 9's complement of the corresponding decimal number. This property is used since a decimal number can be nines' complement as easily as a binary number can be one's complemented; just by inverting all bits.

The process of converting BCD to excess-3 is quite simple from othe conventions. The Excess-3 code can be calculated by adding 3 to each four digit BCD code.

Truth table:

Decimal	BCD				Excess-3			
	W	X	Y	P	A	B	C	D
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0
10	1	0	1	0	X	X	X	X
11	1	0	1	1	X	X	X	X
12	1	1	0	0	X	X	X	X
13	1	1	0	1	X	X	X	X
14	1	1	1	0	X	X	X	X
15	1	1	1	1	X	X	X	X

Now, we will use the K-map method to design the logical circuit for the conversion of BCD to Excess-3 code.

WX\YP	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	X	X	X	X
10	1	1	X	X

$A = W + XP + XY$

WX\YP	00	01	11	10
00	0	1	1	1
01	1	0	0	0
11	X	X	X	X
10	0	1	X	X

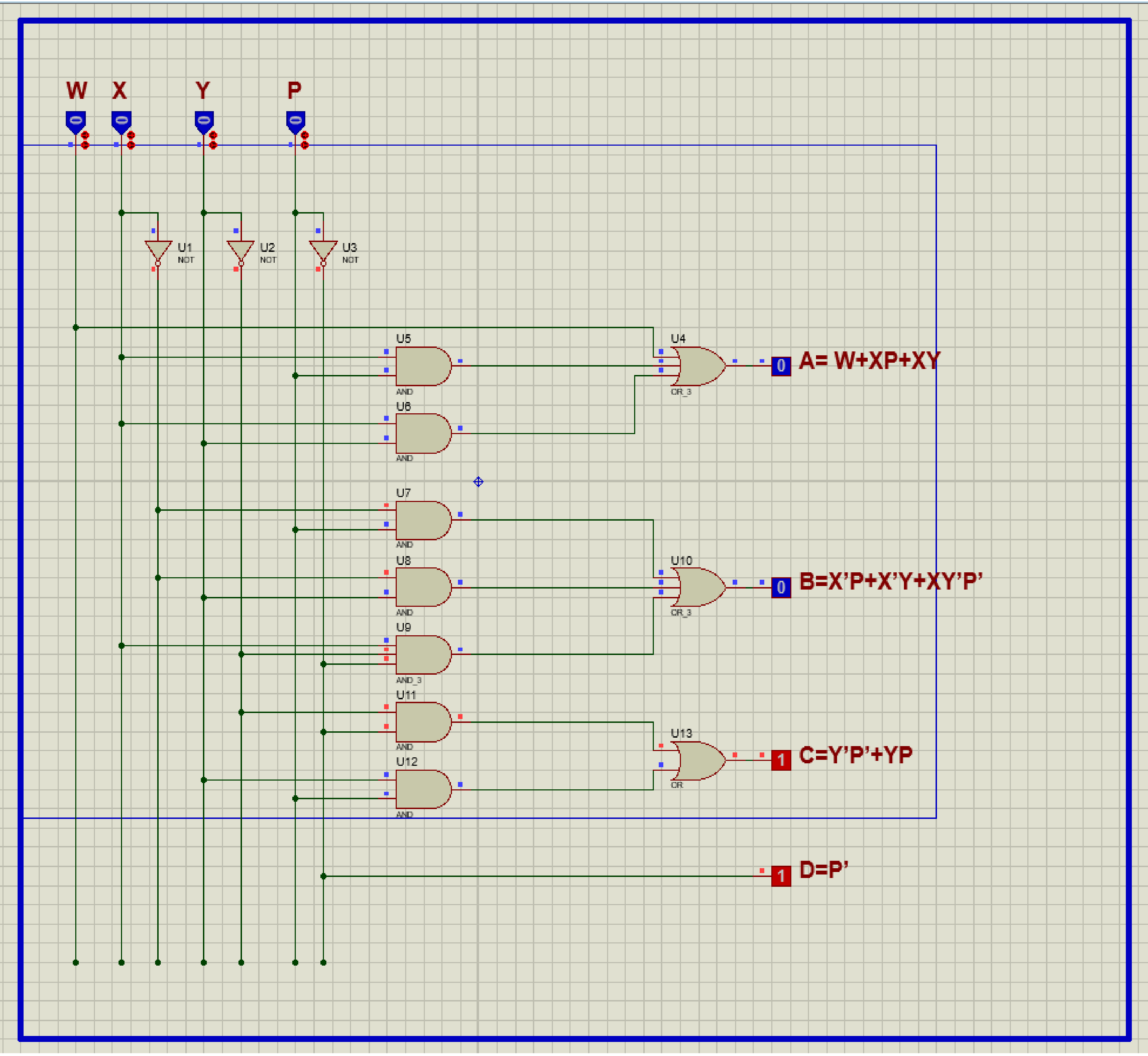
$B = X'P + X'Y + XY'P'$

WX\YP	00	01	11	10
00	1	0	1	0
01	1	0	1	0
11	X	X	X	X
10	1	0	X	X

$C = Y'P' + YP$

	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	X	X	X	X
10	1	0	X	X

$D = P'$


W
X
Y
P
U1 NOT
U2 NOT
U3 NOT
U5 AND
U6 AND
U4 OR_3
A= W+XP+XY
U7 AND
U8 AND
U9 AND_3
U10 OR_3
B=X'P+X'Y+XY'P'
U11 AND
U12 AND
U13 OR
C=Y'P'+YP
D=P'
0
0
1
1

Conclusion:

(i) We have learnt about parity bits.

(ii) We have learnt how to check error of a message using even or odd parity.

(iii) We learnt how to design a parity bit checker circuit.

(iv) We have also learnt how to convert BCD code to Excess-3 code.

(v) We have also learnt how to implement code converter using basic gates.

CSE — 206

THE END

CSE — 206