Department of Computer Science and Engineering
BRAC University
CSE 260: Digital Logic Design

Experiment # 3

Parity Generator and Checker

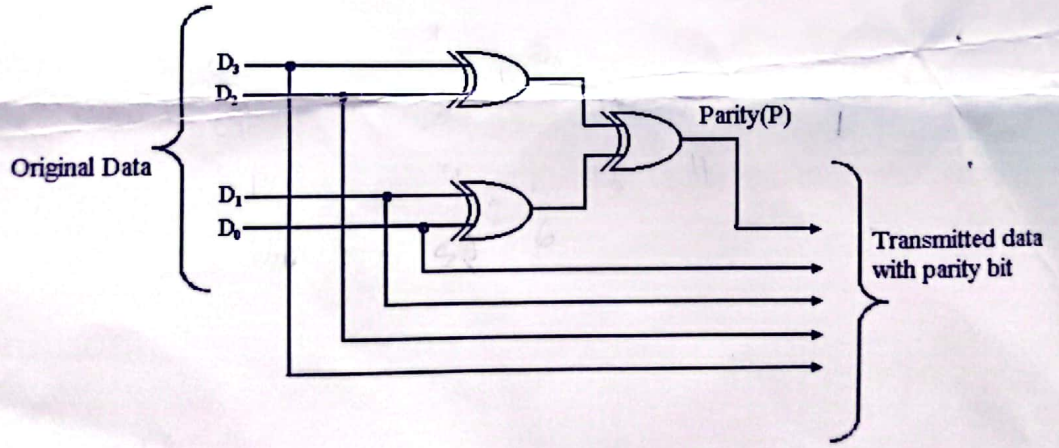
Objective:

- To design and implement an Even parity Generator and Even parity checker using XOR gates) (IC-7486).

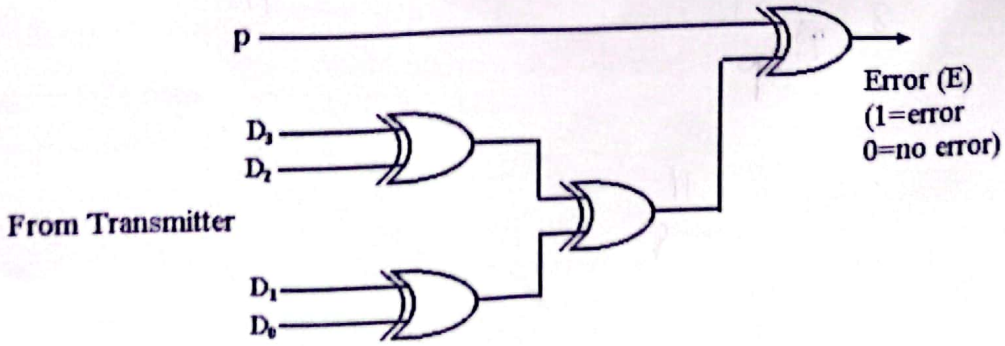
Required Components and Equipments

1. AT-700 Portable Analog/Digital Laboratory
2. 7400×3

Diagram of Circuit:
Even Parity generator



Even Parity Checker



Procedure:

- Construct the Circuit of Figure 1, on the breadboard of AT-700.
- Remember each IC's pin 14 connected to "+5V" position of DC Power Supply of AT-700, and pin 7 connected to "GND" position.
- Connect the inputs to Data switches and outputs to any position of LED Display.
- Determine the parity generator's output for each of the following sets of input data, $D_3D_2D_1D_0$; (a) 0111; (b) 1001; (c) 0000; (d) 0100
- Determine the parity checker's output for each of the following sets of data from the transmitter

P	D_3	D_2	D_1	D_0	Error (E)
0	1	0	1	0	
1	1	1	1	0	
1	1	1	1	1	
1	0	0	0	0	

Report:

The report should cover the followings

1. Name of the Experiment
2. Objective
3. Required Components and Equipments
4. Experimental Setup (You must draw the diagrams)
5. Results in Tabulated form.
6. Discussions (Explanation of the results)

Name of the Experiment:

Parity Generator and Checker

Objective:

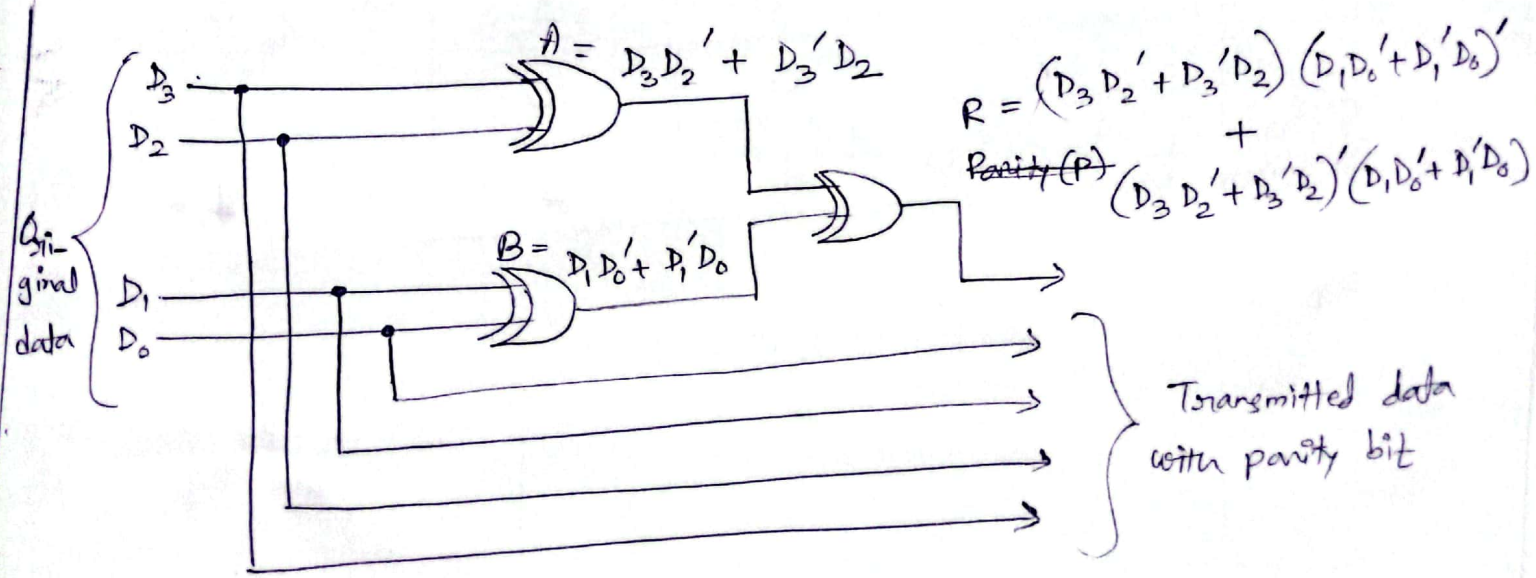
- To design and implement an Even parity Generator and Even parity ~~the~~ checker using XOR gates (IC-7486).

Required Components and Equipments:

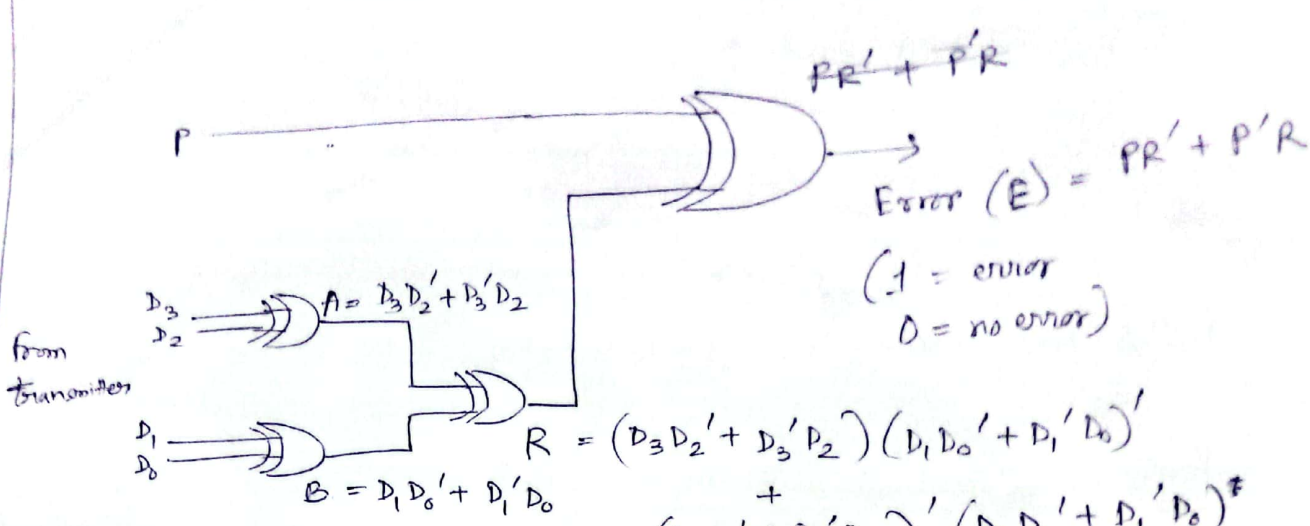
1. AT-700 Portable Analog/Digital Laboratory
2. 7400 × 3

Experimental Setup:

Even parity Generator



Even Parity Checker



Results in Tabulated form:

Even parity ~~Ge~~ Generator's output:

D_3	D_2	D_1	D_0	Output = P	A	B	R
0	1	1	1	1	1	0	1
1	0	0	1	0	1	1	0
0	0	0	0	0	0	0	0
0	1	0	0	1	1	0	1

Even parity Checker's output

P	D_3	D_2	D_1	D_0	A	B	R	Error (E)
0	1	0	1	0	1	1	0	0
1	1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	0	1
1	0	0	0	0	0	0	0	1

Discussions:

for the Even parity Generator, ~~D_3, D_2 D~~ the inputs are D_3, D_2, D_1 and D_0. The output is R. If the inputs are an odd number of 1, the output (R) comes out to be 1 in order to make it even. Otherwise, 0 is produced as the output (R)

The XOR gate in the even parity checker takes two inputs: ~~R and the a different input~~ R and an external input, P. It compares the two inputs and checks if R and P are equivalent. If they are ~~the same~~ equivalent, 0 (no error) is produced as the output, otherwise, 1 (error) is ~~produced~~ the output.

This is similar to how we check if ~~a sender's output is equivalent to a receiver's input~~ a sender's ~~de~~ data is equivalent to the data that is received by a receiver.