ISTANBUL TECHNICAL UNIVERSITY

COMPUTER ENGINEERING DEPARTMENT

BLG 242E
LOGIC CIRCUITS LABORATORY
EXPERIMENT REPORT

EXPERIMENT NO	:	3
EXPERIMENT DATE	:	07/04/2023
LAB SESSION	:	FRIDAY - 14.00
GROUP NO	:	G08

GROUP MEMBERS:

150200081 : KAAN KARATAŞ
150210719 : NACİ TOYGUN GÖRMÜŞ

SPRING 2023

Contents

1 INTRODUCTION

Throughout this experiment, the C.A.D.E.T. was tested while a half adder, full adder and adder subtractor circuits were designed. According to the circuits, truth tables were produced, and the outcomes were compared to the C.A.D.E.T. unit. The experiments' primary goal is to create adders and later use them to design a 4 bit adder subtractor circuit.

2 MATERIALS AND METHODS

Tools used on this experiment:

- C.A.D.E.T
- 7400 series ICs
  - 74xx08 - Quadruple 2-input Positive AND Gates
  - 74xx32 - Quadruple 2-input Positive OR Gates
  - 74xx83 - 4-bit Binary Full Adder
  - 74xx86 - Quadruple 2-input Positive Exclusive Or (XOR) Gates

3 PRELIMINERY

- When performing binary addition for unsigned numbers, each bit position is added up, and any carries from lower positions are propagated to higher positions. The outcome is the two numbers added together. We first represent the two numbers in two's complement form and then add them using binary addition for signed addition. The most important bit, which represents the sign of the outcome, is then discarded if it contains any carry.

- When performing binary subtraction for unsigned numbers, each position of a bit is subtracted, and any borrow from a lower position is propagated to the following higher position. The outcome is the difference between the two numbers. We first represent the two numbers in two's complement form and then subtract them using binary subtraction to perform signed subtraction. After that, we throw away any borrows from the most important bit, which represents the outcome's sign.

- When the result of an addition or subtraction operation exceeds the maximum value that can be represented by the number of bits used for the calculation, the terms "carry" and "borrow" are used. A borrow happens when a subtraction operation needs an extra bit to represent the difference, whereas a carry happens when an addition operation needs an extra bit to represent the sum. These ideas are employed to assess whether a result is accurate or not.

- When the outcome of an arithmetic operation exceeds the highest value that the number of bits employed for the calculation can represent, it is said to have overflowed. Both addition and subtraction operations can have it. When an overflow happens, the calculation should be redone using more bits because the result is invalid.

4 EXPERIMENT

4.1 Part 1

A half adder is implemented using 1 AND and 1 XOR gate with 2 inputs labeled A and B and 2 outputs representing Sum and Carry.

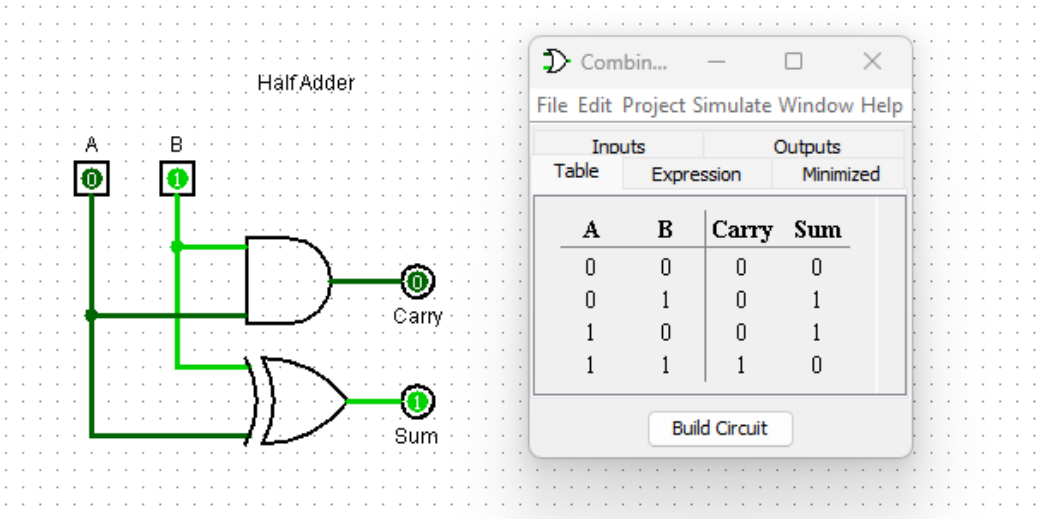

Figure 1: Half Adder Circuit

A	B	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Truth Table of Half Adder

4.2 Part 2

A full adder is implemented by adding an additional C_{in} input.

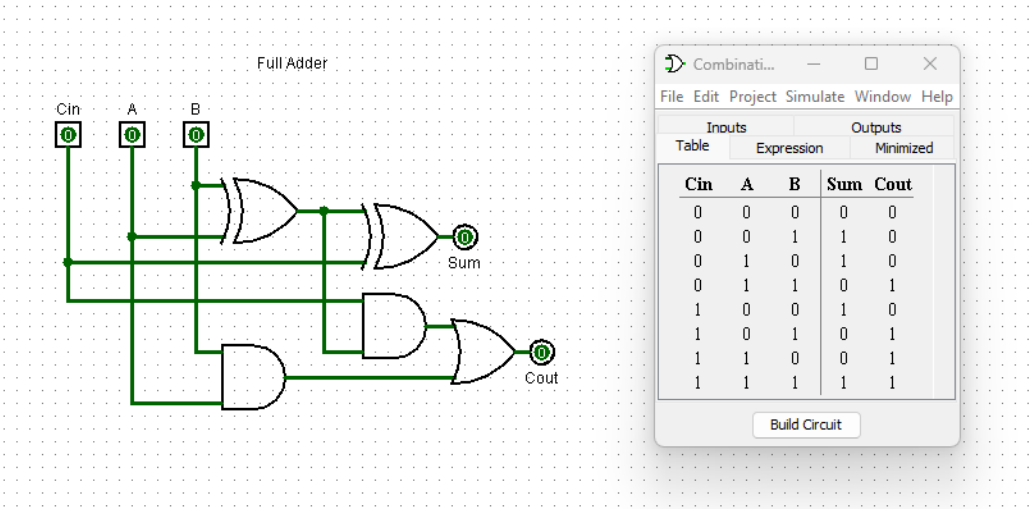

Figure 2: Full Adder Circuit

C_{in}	A	B	Sum	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Truth Table of Full Adder

4.3 Part 3

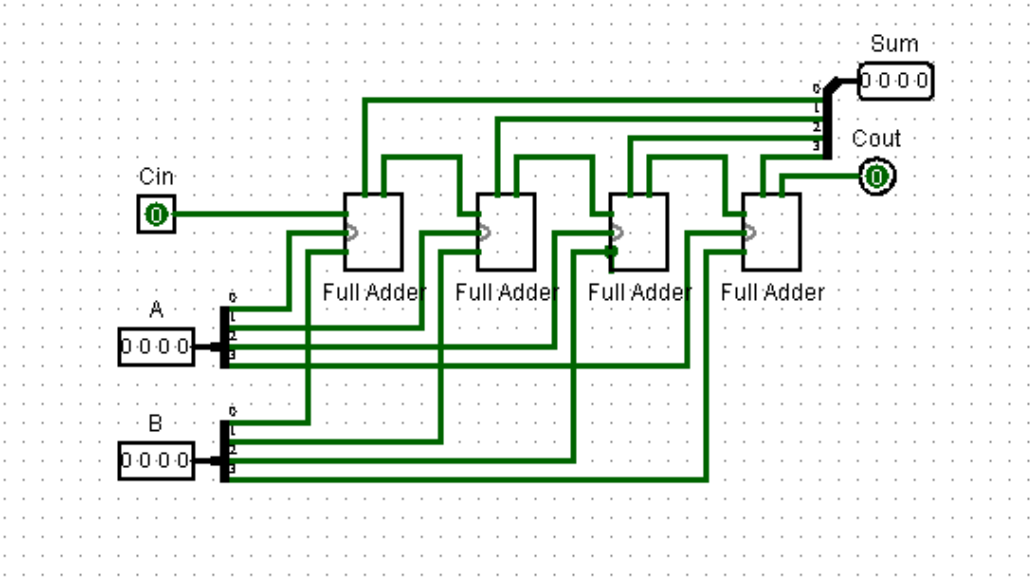

Figure 3: 4-bit Full Adder Circuit

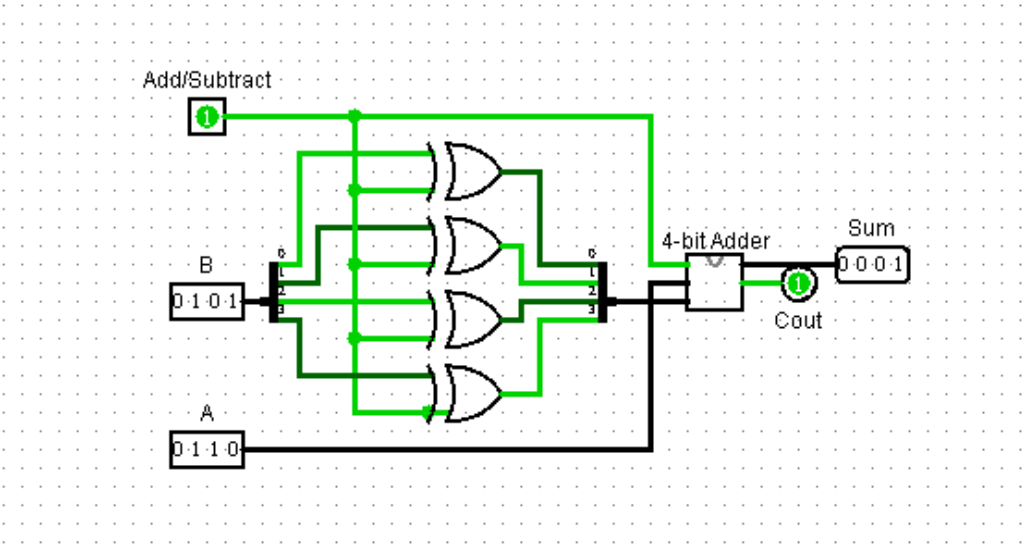

Figure 4: Adder Subtractor Circuit

A	B	Carry	Binary	Decimal
0101	0111	0	1100	12
1101	1001	1	0110	22 (extra bit)
1111	1111	1	1110	30 (extra bit)
0110	1101	1	0011	19 (extra bit)

Unsigned A + B

A	B	Overflow	Binary	Decimal
0101	0111	1	1100	-3 (can not be represented)
1101	1001	1	0110	10 (can not be represented)
1111	1111	0	1110	-2
0110	1101	0	0011	3

Signed A + B

A	B	Barrow	Binary	Decimal
0101	0111	1	1110	13 (can not be represented)
1101	1001	0	0100	4
1111	1111	0	0000	0
0110	1101	1	1001	9 (can not be represented)

Unsigned A - B

A	B	Overflow	Binary	Decimal
0101	0111	0	1110	-2
1101	1001	0	0100	4
1111	1111	0	0000	0
0110	1101	1	1001	9 (can not be represented)

Signed A - B

5 RESULTS

Every circuit design is implemented using Logisim. The intended outputs are obtained when the circuits are configured with all of the provided functionalities. A half adder, full adder and adder subtractor circuits are implemented. The circuits' outcomes were consistent with their truth tables.

6 DISCUSSION

In the experiment, we learnt to implement a half adder, and use the adder to make a full adder by adding additional carry input. After that, by using the full adder we designed a adder subtractor circuit and tested it with different outputs.

7 CONCLUSION

We completed the experiment as it was not a hard one. However, due to an issue that could not get identified by instructors, adder subtractor ciruit's second bit did not work as expected. Despite testing every possibility that could cause the issue including wires, gates, LEDs etc. nothing resolved the problem despite the fact that the connections were certainly correct as stated by many instructors. The most likely and only left untested issue was probably the C.A.D.E.T. device.