Experiments Name: Binary Adder, Subtractor and BCD Adder.

Objective:
- Understand the concept of binary addition and subtraction.
- Learn about Half and Full binary Adders.
- Perform binary addition and subtraction using IC 7483.
- Understand the concept of BCD addition and implement a BCD adder using IC 7483.

Apparatus:
- 2x IC 7483 4-bit binary Adder.
- 1x IC 7486 Quadruple 2-input XOR gates.
- 1x IC 7408 Quadruple 2-input AND gates.
- 1x IC 7432 Quadruple 2-input OR gates.
- Trainer Board.
- Wires.

Theory:
Digital computers perform a variety of information-processing tasks. Among the functions encountered are the various arithmetic operations. The most basic arithmetic operation is the addition of two binary digits. Adder manages these additions' operations. There are two kinds of Adder, Half Adder and Full Adder. Combining two Half Adders, one Full Adder built.

Half Adder:
Half Adder is a combinational logic circuit that adds two single-bit binary numbers and produces two output bits: the sum and the carry. It is called a "half" adder because it can only add two bits and cannot handle any carry input from previous stages.
The truth table for a half-adder is as follows:

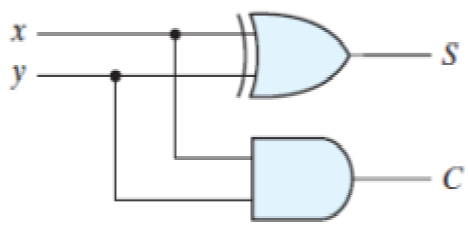
Input A	Input B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Therefore, the logical expression for the sum and carry outputs are:

Sum = A XOR B
Carry = A AND B

The half-adder circuit can be implemented using two logic gates: an XOR gate for the sum output and an AND gate for the carry output. The two input bits are connected to both gates, and the outputs of the two gates are the sum and carry bits, respectively.

Replace X = A
Y = B
S = Sum
C = Carry



Full Adder:
A full adder is a combinational logic circuit that adds three single-bit binary numbers (i.e., three bits) and produces two output bits: the sum and the carry. Unlike a half adder, a full adder can handle an input carry from previous stages.
The truth table for a full adder is as follows:

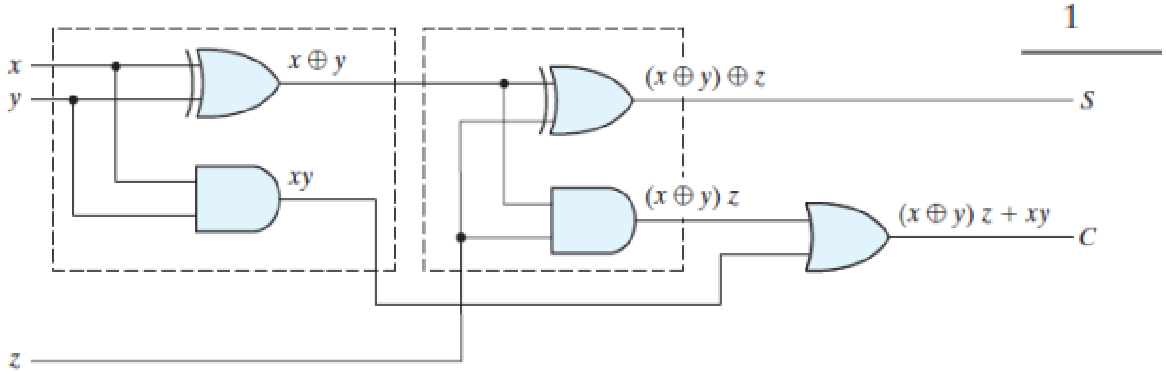
Input A	Input B	Input Carry	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Therefore, the logical expression for the sum and carry outputs are:
Sum = A XOR B XOR Carry_in
Carry_out = (A AND B) OR (Carry_in AND (A XOR B))

The full adder circuit can be implemented using three logic gates: two XOR gates for the sum output and one OR gate and one AND gate for the carry output. The three input bits are connected to the XOR gates, and the output of the XOR gates are then connected to the OR and AND gates to produce the carry and sum bits.

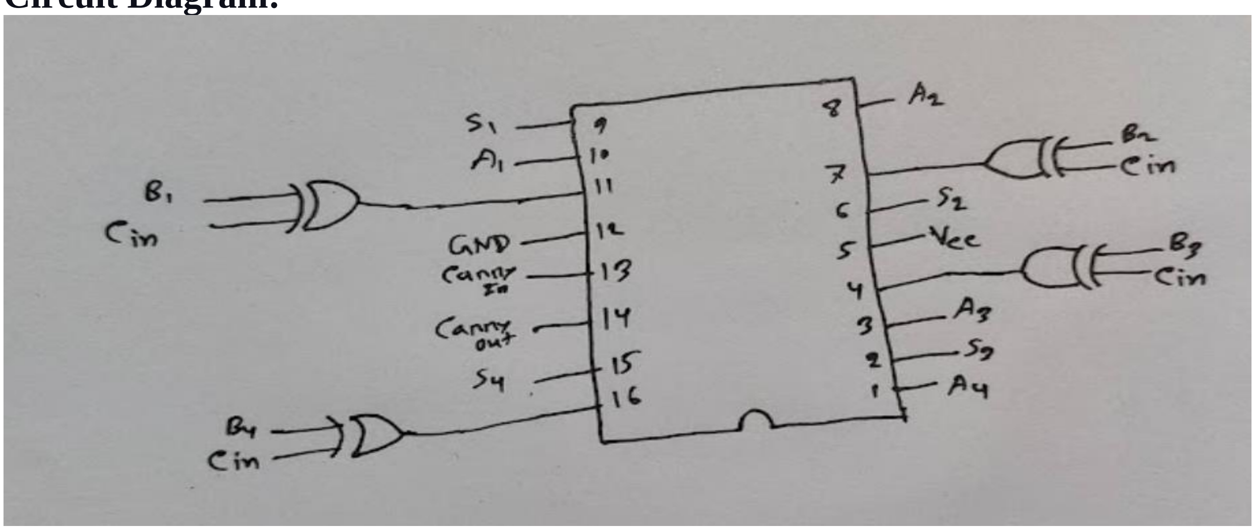
1

1



Replace
X = A
Y = B
Z = Carry_in
S = Sum
C = Carry_out

Circuit Diagram:



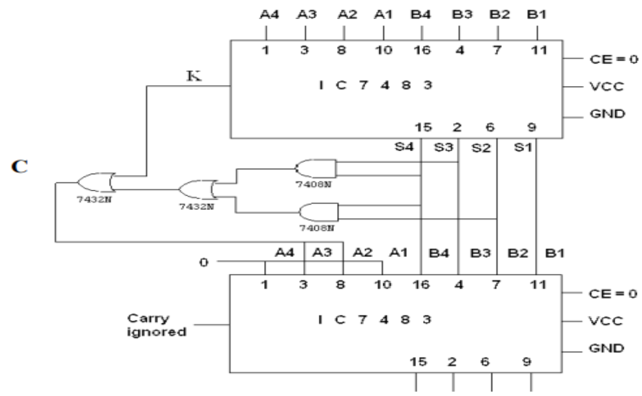


Figure D.2.1: BCD Adder Circuit

Experimental Procedure:

Binary Adder Subtractor:

1. First, we select four switches to represent the bits of input A and another four more binary switches to represent the bits of input B.
2. Then we select another switch for the mode chosen (Carry in).
3. Then we construct the circuits according to the pin details in the circuit diagram section.
4. Then we choose four LEDs to view the output S and another LED for the output carry.
5. Then we test the circuit with the data in Table F.1.1.

BCD Adder:

01. First, we complete Table F.2.1 and F.2.2 for the BCD sum.
02. Then, we construct the circuit shown in Figure D.2.1.
03. We use the first adder output as the input of the second adder.
04. We built the circuit according to the pin details shown in Circuit D.2.1.
05. Then we test the circuit with the data in Table F.2.2.

Simulation:

Attached.

Experimental Data Table:

Department of Electrical & Computer Engineering, NSU

Digital Logic Design Lab

F. Data Sheet:

Group: 05	Section: 08	Instructor's Signature: .... Date: 03/04/2023

F.1 Experimental data (4-bit Binary Adder-Subtractor):

Operation	M	A	B	C4	S4 S3 S2 S1
7 + 5	0	0111	0101	0	1100
4 + 6	0	0100	0110	0	1010
9 + 11	0	1001	1011	1	0100
15 + 15	0	1111	1111	1	1110
7 − 5	1	0111	0101	1	0010
4 − 6	1	0100	0110	0	1110
11 − 2	1	1011	0010	1	1001
15 − 15	1	1111	1111	1	0000

Table F.1.1

F.2 Exper...

Decimal Value	Binary Sum					BCD Sum				
	K	Z_3	Z_2	Z_1	Z_0	C	S_3	S_2	S_1	S_0
0	0	0	0	0	0					
1	0	0	0	0	1					
2	0	0	0	1	0					
3	0	0	0	1	1					
4	0	0	1	0	0					
5	0	0	1	0	1					
6	0	0	1	1	0					
7	0	0	1	1	1					
8	0	1	0	0	0					
9	0	1	0	0	1	0	1	0	0	1
10	0	1	0	1	0	1	0	0	0	0
11	0	1	0	1	1	1	0	0	0	1
12	0	1	1	0	0	1	0	0	1	0
13	0	1	1	0	1	1	0	0	1	1
14	0	1	1	1	0	1	0	1	0	0
15	0	1	1	1	1	1	0	1	0	1
16	1	0	0	0	0	1	0	1	1	0
17	1	0	0	0	1	1	0	1	1	1
18	1	0	0	1	0	1	1	0	0	0
19	1	0	0	1	1	1	1	0	0	1

Table F.2.1

Operation	A	B	Overflow Carry	Sum
9 + 0	Ø1001	0000	0	1001
9 + 1	1001	0001	1	0000
9 + 2	1001	0010	1	0001
9 + 3	1001	0011	1	0010
9 + 4	1001	0100	1	0011
9 + 5	1001	0101	1	0100
9 + 6	1001	0110	1	0101
9 + 7	1001	0111	1	0110
9 + 8	1001	1000	1	0111
9 + 9	1001	1001	1	1000

Table F.2.2

Prittika Dhar

Results:

After implementation of our circuits, we test it with the data table F.1.1, F.2.1; and we get the exact output as the table.

Questions and Answers (Q/A):

01.…

In this experiment, we use XOR gates for the input of B to convert in the first complements, and then we use the M bit as carry-in to convert the B to the second complements.

Truth Table of XOR Gates:

Input M	Input B	Output
0	0	0
0	1	1
1	0	1
1	1	0

According to the truth table, when the M bit is 0, the inputs of B will not change. And when the M bit is 1, inputs of B will invert. We know that for the subtractor, we need to convert the negative number to second compliments then we can add and get the subtractor result. We use M bit as a mood choice, like 0 for add and 1 for subtractor. When M is 0, the input of B will not change, and the adder will add the number. When M is 1, the input of B will invert, which means B will convert into a first complement. And then, M bit will enter as carry-in so that B will convert into second compliments and the adder will then add the number and show out the subtractor result.

02. Simulation attached.

03. In our BCD adder circuit, Top Adder IC will take input of A and B, add them, and give the output with a carry. And the bottom Adder IC will take the top adder's output as input, and another input will be 0110 or 0000. In this case, OR and AND gates will build logic to decide when to add 0110 and 0000. Here we can see that if the carry output of the top adder IC is 1, it will add 0110 as input for the bottom adder IC. In another case, if S4 and S3 are '1' or S4 and S2 are '1,' it will add 01110 as input for the bottom adder IC. In other cases, it will add 0000 as the input for the bottom adder IC. Then the bottom Adder IC will give the exact output of BCD Sum.

    The principles behind BCD Sum:
    From 0 to 9, Binary Sum and BCD Sum are the same. From 10 to 19, the BCD Sum is the same as the Binary Sum of 0-9; just a carry bit 1 will be added before the BCD Sum. BCD is Binary Coded Decimal. Every bit of a Decimal number will represent a 4-bit Binary number. From 10 to 19, MSB is '1', and LSB is 0-9. That's why BCD Sum is the same as 0-9, just an extra 1 bit in the front.

Discussion:

Through this experiment, we now understand the concept of binary addition and subtraction. We can use Half and Full Binary Adders. We can do addition and subtraction by using this adder. We also learn the concept of BCD Sum. In the first part, our circuit was unable to do a subtractor. Later we identified that the adder IC was damaged. Then we replace the IC and get the accurate result. In the second part, we don't face any problems and compete within the time. In short, we learned about the adder and subtractor circuits.