Experiment Report:
Study of Adder and Subtractor Circuits

Abhijnan Saraswat Gogoi
Roll no.: 2411072
PHY241 Course

29th, October 2025

Abstract

This experiment aims to design and implement basic arithmetic circuits — half adder, full adder, half subtractor, and full subtractor — using fundamental logic gates. Each circuit was constructed on a breadboard and tested to verify its truth table and logical functionality. The half adder and half subtractor performed single-bit addition and subtraction, while the full adder and full subtractor extended these operations by incorporating carry and borrow inputs from previous stages. The observed outputs matched theoretical predictions, confirming the correct logical behavior of the circuits. The experiment also demonstrated the modular and hierarchical nature of combinational logic design, where complex digital systems can be realized by interconnecting simpler logic units. This principle underlies the construction of larger computational elements such as arithmetic logic units (ALUs) in digital processors.

1 Objectives

- To construct half and full adder circuit and verify its working.
- To construct half and full subtractor circuit and verify its working.

2 Theory

2.1 Half Adder

A Half Adder is a logic circuit used to perform the arithmetic addition of two single-bit binary inputs. It is the simplest form of an adder circuit and serves as the building block for more complex arithmetic units. The inputs, denoted as A and B, represent the two bits to be added. The circuit produces two outputs: Sum (S), which gives the least significant bit of the result, and Carry (C), which represents the overflow or carry generated during the addition.

The logical expressions for the outputs are given by:

$$S = A \oplus B \tag{1}$$

$$C = A \cdot B \tag{2}$$

where $\oplus$ denotes the XOR (exclusive OR) operation, and $\cdot$ denotes the logical AND operation. The XOR gate produces a high output only when the number of high inputs is odd, while the AND gate produces a high output only when both inputs are high.

Table 1: Truth Table of Half Adder

A	B	Q (Sum)	C (Carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Thus, the Half Adder successfully performs binary addition for two bits but cannot handle carry input from a previous stage, which limits its application to single-bit additions only. The circuit diagram for a half-adder circuit is given in figure 1a.

2.2 Half Subtractor

A Half Subtractor is a logic circuit designed to perform the arithmetic subtraction of two binary bits. The circuit takes two inputs: A and B, and generates two outputs: the Difference (Q) and the Borrow (B). The Borrow

output indicates whether a '1' has been borrowed from a higher bit position during the subtraction.

The logical relationships between inputs and outputs are given by:

$$D = A \oplus B \tag{3}$$
$$B = \overline{A} \cdot B \tag{4}$$

where $\overline{A}$ denotes the logical NOT of A. The XOR gate generates the difference bit, while the combination of NOT and AND gates determines when a borrow is needed.

Table 2: Truth Table of Half Subtractor

A	**B**	**Q (Difference)**	**B (Borrow)**
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

The Half Subtractor, like the Half Adder, is limited to single-bit operations, as it cannot account for a borrow from a previous stage. The circuit diagram of a half-subtractor circuit is given in figure 1b.

2.3 Full Adder

A Full Adder extends the concept of the Half Adder by including an additional input to account for the carry generated from a previous addition stage. It performs the addition of three binary inputs — two significant bits (A and B) and an input carry (C_{N-1}) — and produces two outputs: the Sum (S) and the Carry (C_N).

The Boolean expressions for these outputs are:

$$S = (A \oplus B) \oplus C_N \tag{5}$$
$$C_N = (A \cdot B) + (B \cdot C_{N-1}) + (A \cdot C_{N-1}) \tag{6}$$

The Full Adder can be realized by combining two Half Adders and an OR gate. The first Half Adder adds A and B, generating a partial sum and carry, and the second Half Adder adds the partial sum with C_{in}, while the OR gate combines the two carry outputs to produce C_{out}. The circuit diagram for the full-adder circuit is given in figure 1c.

Table 3: Truth Table of Full Adder

C_{N-1}	**A**	**B**	**Q (Sum)**	C_N **(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

The Full Adder is an essential building block in digital arithmetic, as multiple Full Adders can be cascaded to construct Ripple Carry Adders capable of adding multi-bit binary numbers.

2.4 Full Subtractor

A Full Subtractor is a digital circuit that performs subtraction on three input bits: the bits from the binary numbers (A), (B), and the borrow from the previous stage (B_{N-1}). It produces two outputs: the Difference (D) and the Borrow (B_N).

The Boolean expressions governing its behavior are:

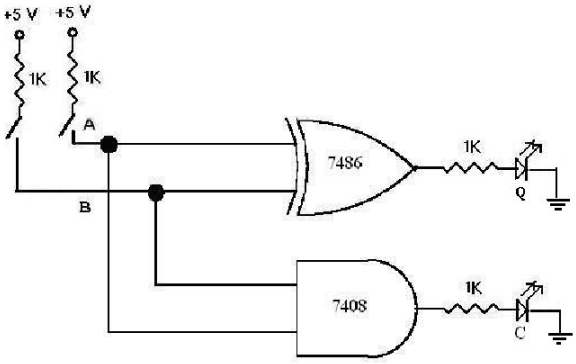
$$D = (A \oplus B) \oplus B_{N-1} \tag{7}$$
$$B_N = (\overline{A} \cdot B) + (B \cdot B_{N-1}) + (\overline{A} \cdot B_{N-1}) \tag{8}$$

Here, the difference output is obtained using the XOR operation, while the borrow output is generated through a combination of AND and OR operations that determine when borrowing is required.

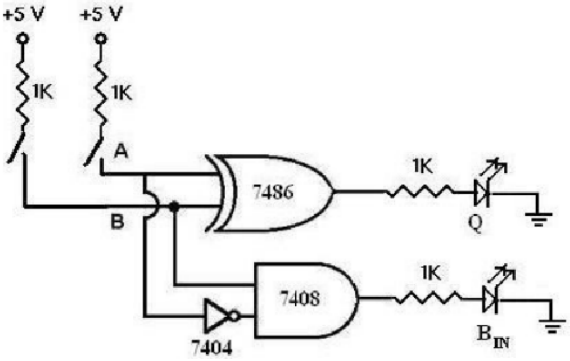
Table 4: Truth Table of Full Subtractor

B_{N-1}	**A**	**B**	**Q (Diff.)**	B_N (Borrow)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

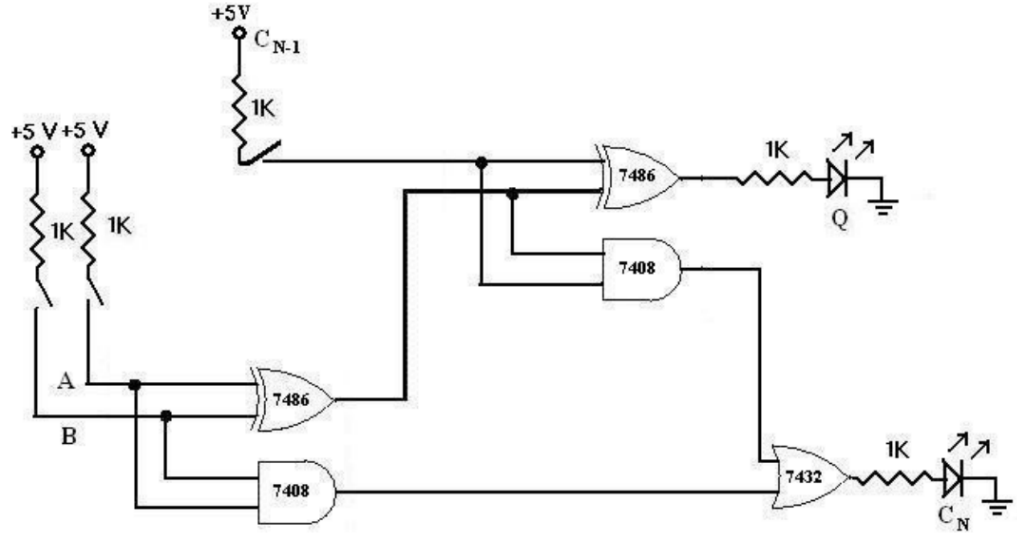
A Full Subtractor can be implemented using two Half Subtractors and an OR gate. When connected in a cascade, multiple Full Subtractors can perform multi-bit binary subtraction. This circuit is crucial in arithmetic logic units (ALUs) and digital processors where subtraction is a frequent operation. The circuit diagram for the full-subtractor circuit is given in figure 1d.



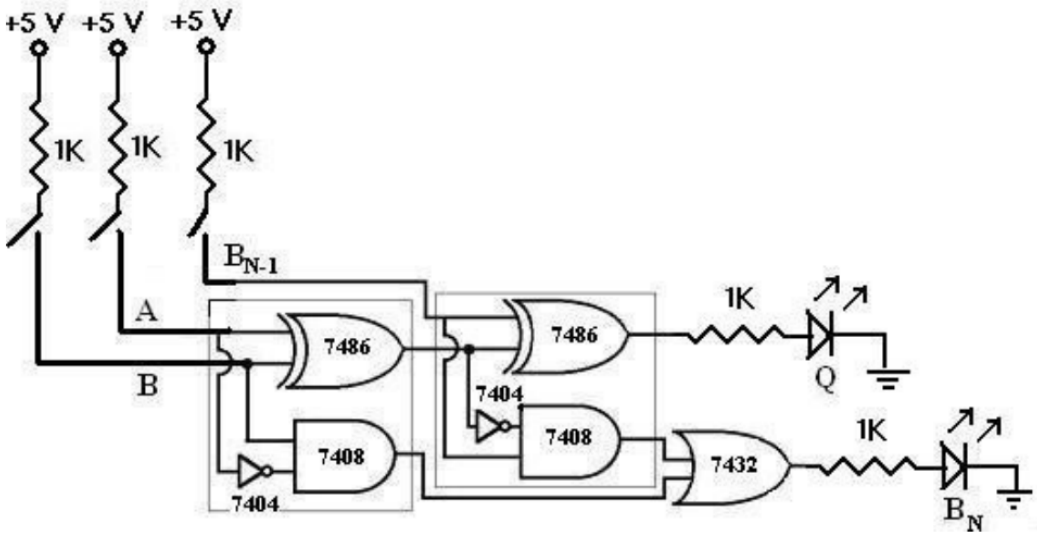
(a) Half adder circuit



(b) Half subtractor circuit



(c) Full adder circuit



(d) Full subtractor circuit

Figure 1: Circuit diagrams

3 Materials Required

(i) Resistors (1kΩ, 5 Nos), (ii) ICs (XOR-7486, AND-7408, OR-7432, NOT-7404), (iii) D.C. Power supply (5V), (iv) LEDs (2 Nos), (v) Connecting wires, (vi) Breadboard.

4 Observations

The observations made using the half-adder, half-subtractor, full-adder and full-subtractor circuits are as follows:

Table 5: Half Adder Truth Table

A	B	Q	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Table 6: Full Adder Truth Table

C_{N-1}	A	B	Q	C_N
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

Table 7: Half Subtractor Truth Table

A	B	Q	B_N
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Table 8: Full Subtractor Truth Table

B_{N-1}	A	B	Q	B_N
0	0	0	0	0
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

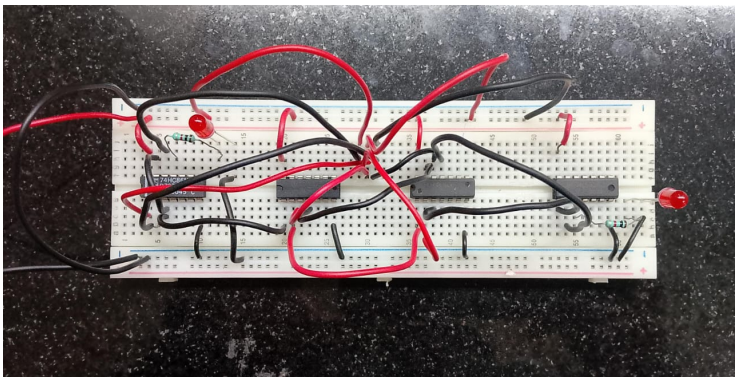


Figure 2: Full subtractor circuit implementation
ICs from left to right: 7486, 7404, 7408, 7432 with their notch towards the left.

5 Results

The observed truth tables of all four circuits (see tables 5, 6, 7 and 8) matched with the theoretical predictions, verifying the correctness of their logical behavior. The half-adder and half-subtractor circuits successfully implemented basic binary addition and subtraction of single-bit numbers. The full-adder and full-subtractor circuits successfully extended these operations by including carry and borrow inputs respectively.

6 Discussion

The experimental implementation of the half-adder, full-adder, half-subtractor, and full-subtractor circuits validated the theoretical principles of digital arithmetic operations using basic logic gates. Each circuit functioned according to its expected Boolean expressions and corresponding truth tables.

During the experiment, the logical outputs were clearly indicated by the LEDs, confirming proper gate-level operation. However, minor fluctuations in LED brightness were occasionally observed, which can be attributed to variations in contact resistance on the breadboard or slight voltage drops across connections. These fluctuations did not affect the logical validity of the results, indicating that the circuit design remained functionally stable.

The experiment highlights the modular nature of combinational logic design, where complex arithmetic operations can be realized by interconnecting simpler units such as full adders and full subtractors. The construction of full adders and full subtractors from their half counterparts demonstrates the hierarchical design methodology commonly employed in digital systems. This principle of modularity forms the basis of larger computational units such as arithmetic logic units (ALUs) in processors.

7 Conclusion

The half and full adder and subtractor circuits were successfully constructed and verified. The experimental results conformed to the theoretical truth tables, confirming the correct operation of these fundamental arithmetic logic circuits.

8 Precautions

1. Ensure correct pin connections of each IC according to its datasheet.
2. Apply the supply voltage (5V) only after verifying circuit connections.
3. Common ground connection must be maintained between all ICs and power source.
4. Avoid loose connections on the breadboard for stable LED outputs.