Digital Logic Design (EL-1005)

LABORATORY MANUAL

Spring-2025

LAB 05

Adder and Subtractor and Multiplier

______________________________ ________ __

STUDENT NAME ROLL NO SEC

____

INSTRUCTOR SIGNATURE& DATE

MARKS AWARDED: /10

____

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES (NUCES), KARACHI

Lab Session 05: Adder and Subtractor and Multiplier

OBJECTIVES:

After completing this lab, you would be able to know

- Distinguish between Half Adder and Full Adder, their functions and logic diagrams
- Define some useful terminologies like CARRY, SUM, Difference and Borrow
- To study the basic operation and design of multiplier Circuits
- To learn how to implement 2x2 bit multiplier circuit

APPARATUS:

- Logic Trainer, Logisim

COMPONENTS:

ICs 74LS02, 74LS00, ICs 74LS02, 74LS00, 74LS08, 74LS32, 74LS04, Logic Works

Introduction:

Theory:

In electronics, an *adder* or *summer* is a digital circuit that performs addition of numbers.

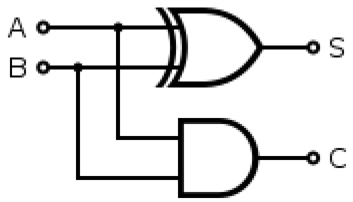
For single bit adders, there are two general types:

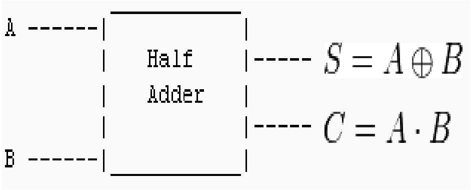
✔ Half Adder
✔ Full Adder

1. Half Adder

A **half adder** is a logic circuit which performs addition of two binary one-bit inputs and has two binary outputs as a result. The outputs are designated as **Sum (S)** and **Carry (C)**.

Circuit Diagram





Truth Table:

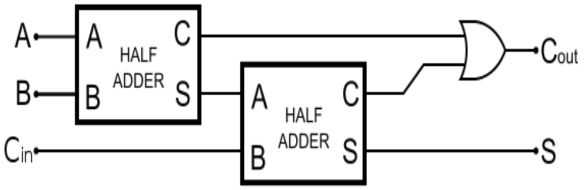
A	B	Sum	Carry
0	0	0	0

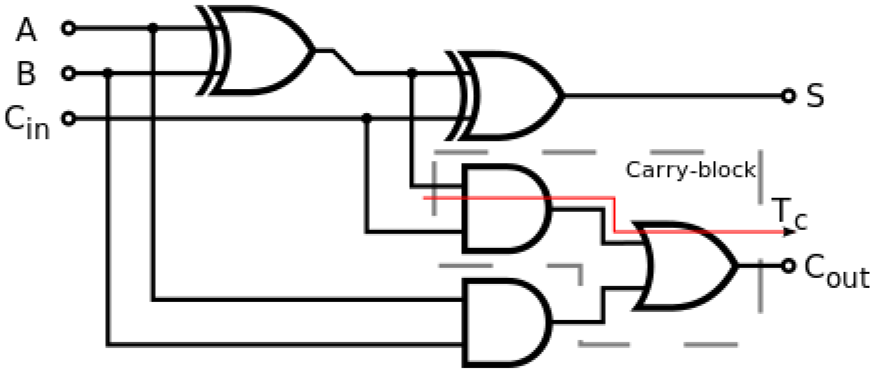
0	1	1	0
1	0	1	0
1	1	0	1

2. Full Adder:

The downfall of half adders is that while they can generate a carry out output, they cannot deal with a carry in signal. This means that they can only ever be stand-alone units, and catted to add multiple bit numbers.

A full adder solves this problem by adding three numbers together - the two addends as in the half adder, and a carry in input. The outputs of the full adder are designated as Sum (S) and Carry out (C_{out}). A block diagram of Full Adder implementation is as follows:





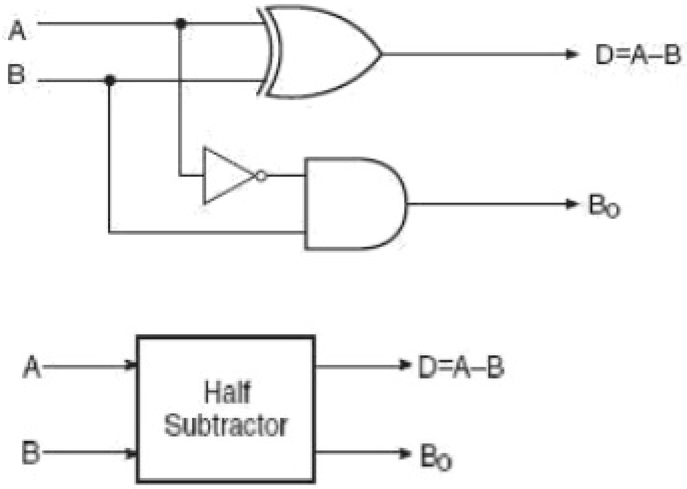
Truth Table:

Inputs			Outputs	
A	B	Cin	S	Coutt

3. Half Subtractor

A half subtractor circuit performs the subtraction of two binary inputs and has two binary outputs as a result. The outputs of the half subtractor are designated as Difference (D) and Borrow (B). The difference and borrow are the binary difference and borrow and has either ‘0’ or ‘1’ logic.

Circuit Diagram:



Truth Table:

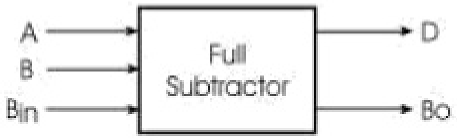
A	**B**	**Difference**	**Borrow**
0	**0**	**0**	**0**
0	**1**	**1**	**1**

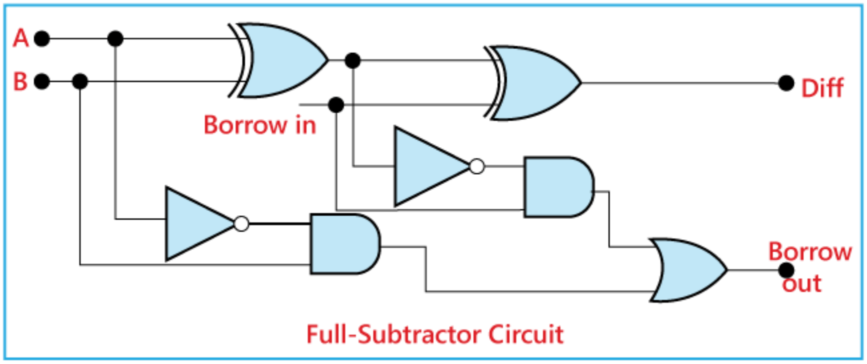
1	0	1	0
1	1	0	0

4. Full Subtractor

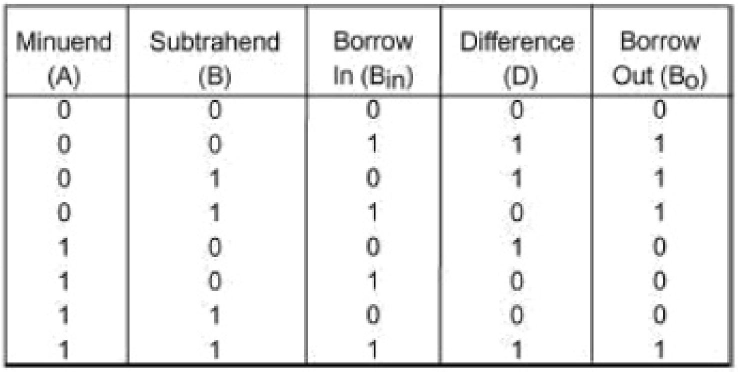
Full subtractor is a logic circuit that performs binary subtraction of two 2-bit numbers. It generates two output namely “Difference” and “Borrow”.

CIRCUIT DIAGRAM





TRUTH TABLE



Minuend (A)	Subtrahend (B)	Borrow In (B_{in})	Difference (D)	Borrow Out (B_O)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

BOOLEAN EXPRESSIONS

D = A'B'Bin + AB'Bin'+ A'BBin' + ABBin = $\mathbf{A \oplus B \oplus B_{in}}$

Bout = A'Bin + A'B + BBin

Multiplier:

A multiplier is a combinational logic circuit that we use to multiply binary digits. Just like the adder and the subtractor, a multiplier is an arithmetic combinational logic circuit. It is also known as a binary multiplier or a digital multiplier.

Multiplication in binary Number system is similar to its decimal counterpart. Two numbers A and B can be multiplied by partial products: for each digit in B, the product of that digit in A is calculated and written on a new line, shifted leftward so that its rightmost digit lines up with the digit in B that was used. The sum of all these partial products gives the final result. Since there are only two digits in binary, there are only two possible outcomes of each partial multiplication:

· If the digit in B is 0, the partial product is also 0.

· If the digit in B is 1, the partial product is equal to A.

For example, the binary numbers **1011** and **1010** are multiplied as follows:

· The AND gates produce the partial products.

· For a 2-bit by 2-bit multiplier, we can just use two half adders to sum the partial products. In general, though, we'll need full adders.

· Here C3 -C0 are the product, not carries.

How does binary multiplication work and how to design a 2-bit multiplier?

Binary multiplication works just like normal multiplication. There are four main rules that are quite simple to understand:

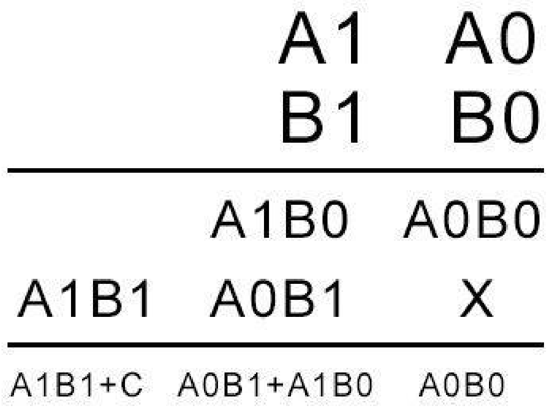
0 x 0 = 0

0 x 1 = 0

1 x 0 = 0

1 x 1 = 1

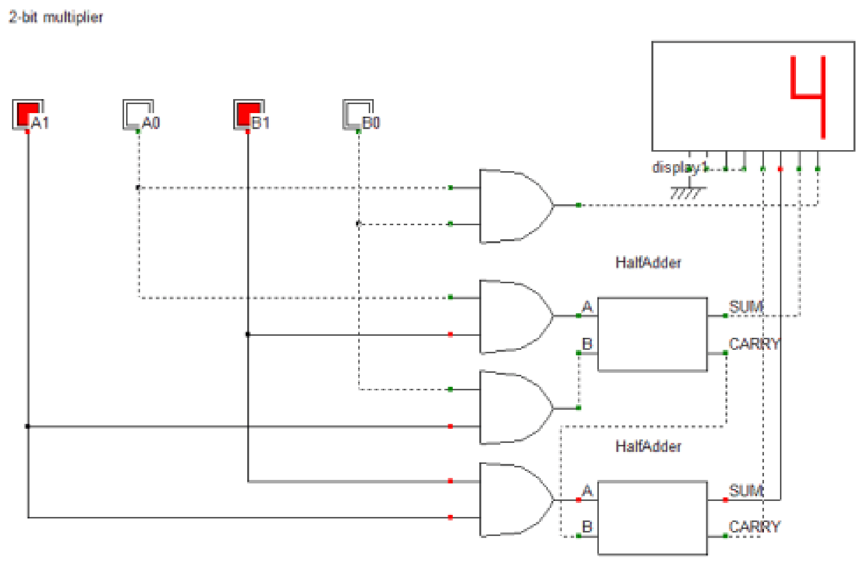
Suppose you have two binary digits A1A0 and B1B0, here's how that multiplication would take place.



	A1	A0
	B1	B0
	A1B0	A0B0
A1B1	A0B1	X
A1B1+C	A0B1+A1B0	A0B0

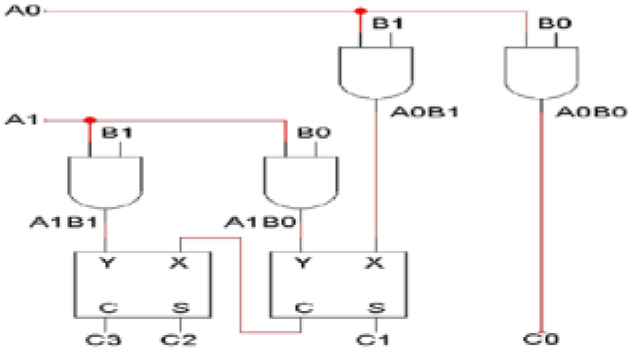
As the number of bits increases, we keep shifting each successive partial product to the left by 1 bit. In the end, we add the digits while keeping in mind the carry that might generate.

Based on the above equation, we can see that we need four AND gates and two half adders to design the combinational circuit for the multiplier. The AND gates will perform the multiplication, and the half adders will add the partial product terms. Hence the circuit obtained is as follows.



A 2x2 Binary Multiplier

$$\begin{array}{ccccc}
 & & & B_1 & B_0 \\
 & & x & A_1 & A_0 \\
\hline
 & & & A_0B_1 & A_0B_0 \\
+ & & A_1B_1 & A_1B_0 & \\
\hline
 & C_3 & C_2 & C_1 & C_0
\end{array}$$



LAB TASKS

Exercise # 1

Use K-Map to find expression for Carry and Sum for Full Adder. Use Truth Table in manual (also implement).

Exercise # 2

Create a 2-bit adder circuit and implement it on breadboard along with truth table.

Exercise # 3

Create a 2-bit subtracter circuit and implement it on breadboard along with truth table.

Exercise # 4

Create a 2-bit binary multiplier along with its truth table and multiply the number 11 and 11.

Exercise # 5

Design 3*3 Bit Binary Multiplier Circuit in Logisim/Logic Works

Exercise # 6

Design 4*4 Bit Binary Multiplier Circuit in Logic Works.

Exercise #7

Design a 4-bit Adder/Subtractor circuit (With the values being your roll number i.e. 23k-8241 the 8241 + 23 and 8241 – 23.). Show the inputs and outputs as well in binary and decimal form.