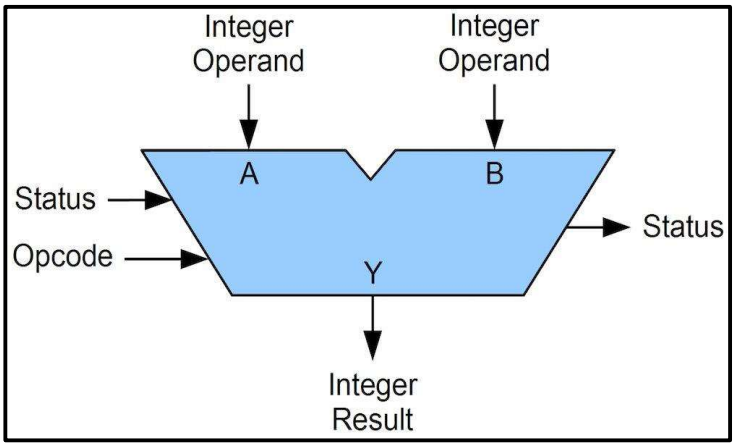
Introduction

In this project, we will build the heart of a simple 4-bit CPU, the ALU (Arithmetic Logic Unit).

Believe it or not, computers existed before microcontrollers and CPUs were around. They used to be built using discrete parts including simple ICs and transistors. CPUs are arguably the center of modern electronics, whether it be a mobile device or a control circuit for a factory. But how do CPUs work? What goes on inside?

A **CPU** consists of three main sections: **memory for variables (registers)**, **control circuitry (microcode)**, and the **ALU**. The **ALU (Arithmetic Logic Unit)** is the part of a CPU that does calculations and condition testing.
For example, if you wish to add two binary numbers, it is the ALU that is responsible for producing the result. If your program needs to execute some code if two values are equal it is the ALU that performs the comparison between the values and then sets flags if the condition is met or not.



Modern CPUs consist of millions of transistors (even billions now!) and cannot possibly be duplicated at home. But a simple CPU (say, a Z80, for example) has only 8500 transistors. Computers in the past (such as many of the IBM mainframe computers) were built with discrete 4000 and 7400 series chips.

This means that we *can* build a CPU at home!!! So why to wait?

This project will be a discrete 4-bit ALU that will be constructed with 7400 series chips.

Theoretical Analysis

Because this project is rather complex you will need the following:

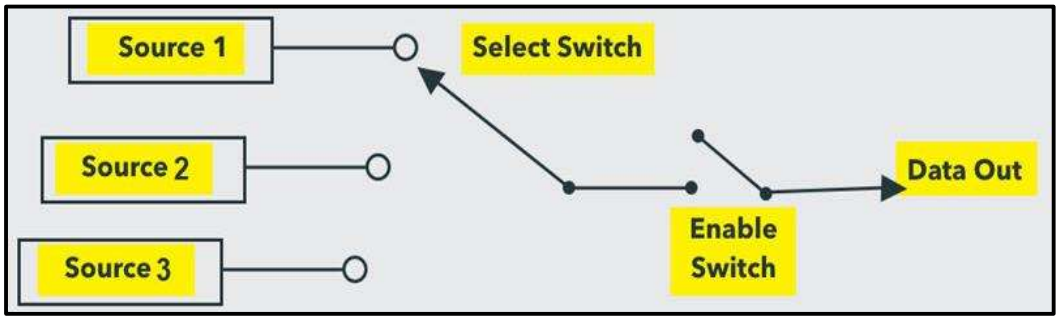
- Basic understanding of Combinational Circuits
- Basic understanding of Logic gates and their construction from universal gates

❖ Combinational Circuits

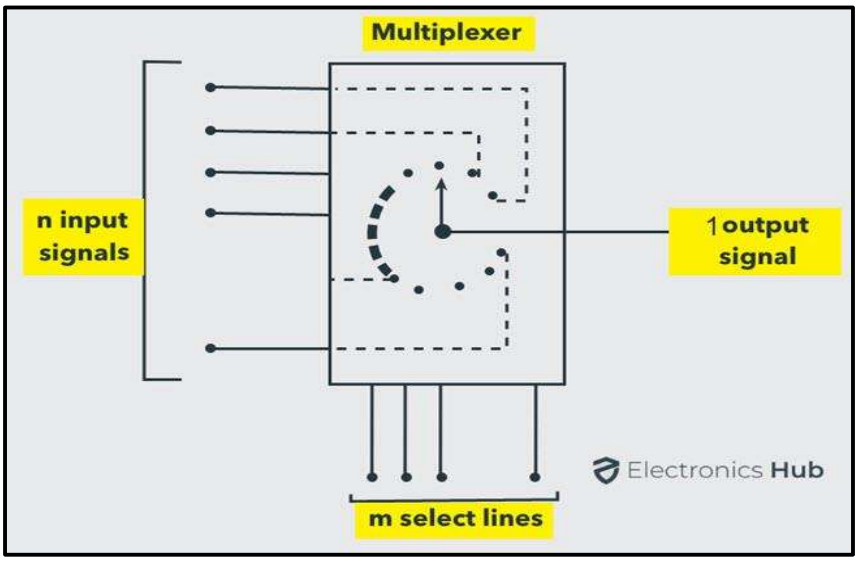
1. Multiplexer (IC NO – 74151)

The multiplexer or MUX is a digital switch, also called as data selector. Circuit with more than one input line, one output line and more than one selects line. It accepts the binary information from several input lines or sources and depending on the set of select lines, a particular input line is routed onto a single output line.

The basic idea of multiplexing is shown in figure below in which data from several sources are routed to the single output line when the enable switch is ON. Therefore, multiplexers are also called as 'many to one' combinational circuits.

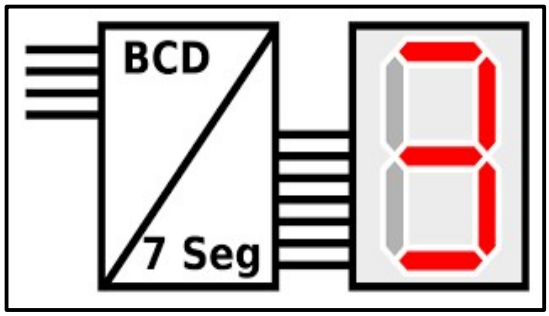


The below figure shows the block diagram of a multiplexer consisting of n input lines, m selection lines and one output line. If there are m selection lines, then the number of possible input lines is 2^m.



2. Decoder Driver (IC NO – 7447)

The **Decoder** is an essential component in **BCD to seven segment decoder** A decoder is nothing, but a combinational logic circuit mainly used for **converting a BCD to an equivalent decimal number.**

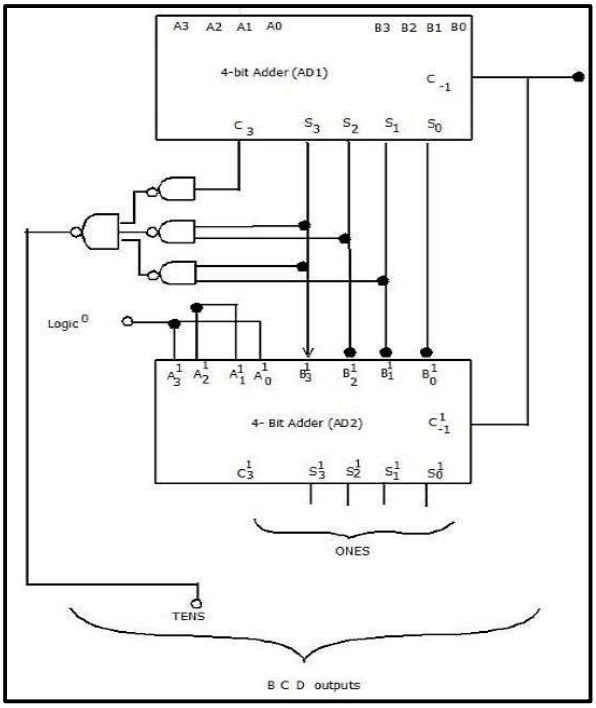


The circuit design, as well as operation, mainly depends on the concepts of **Boolean algebra** as well as logic gates.

A **Seven segment LED display circuit** can be built with eight LEDs. The **common terminals are anode otherwise cathode.** A general anode seven segment display includes 8 pins where **7-pins are input pins** that are marked with **from a to g** & **8th-pin is connected to supply.**

3. BCD Adder (Formed by 4-bit binary/parallel adders) (IC NO – 7483)

In this, if 4- bit sum output is not a valid BCD digit, or if carry C3 is generated, then decimal 6 (0 1 1 0) is to be added to the sum to get the correct result. Following fig shows 4-bit BCD Adder. BCD adder can be cascaded to add numbers several digit 1009 by connecting the carry out of a stage to the carry into the next stage.

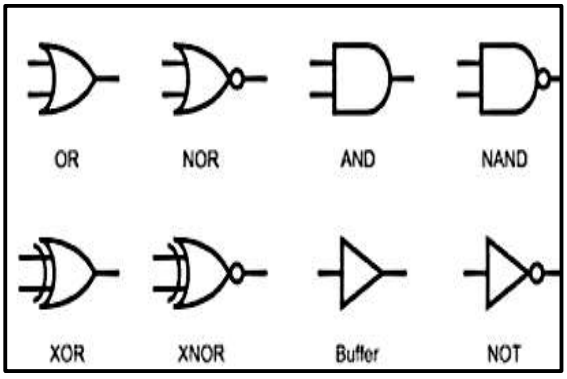


❖ Logic gates

A logic gate is a device that acts as a building block for digital circuits. They perform basic logical functions that are fundamental to digital circuits. Most electronic devices we use today will have some form of logic gates in them.
For example, logic gates can be used in technologies such as smartphones, tablets or within memory devices.

In a circuit, logic gates will make decisions based on a combination of digital signals coming from its inputs. Most logic gates have two inputs and one output. At any given moment, every terminal is in one of the two binary conditions, *false* or *true*. False represents 0, and true represents 1. Depending on the type of logic gate being used and the combination of inputs, the binary output will differ.

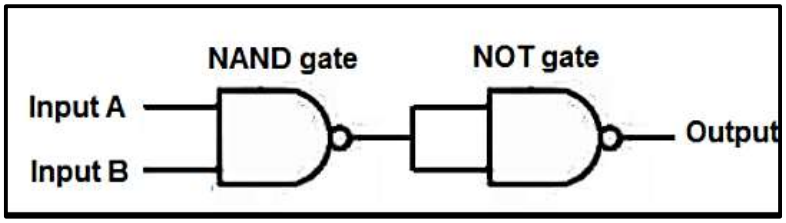
There are **7 basic logic gates**:
AND (7408), OR (7432), XOR (7486), NOT (7404), NAND (7400), NOR (7402), and XNOR (74266).



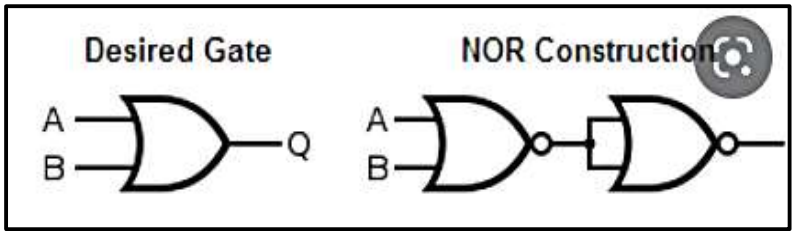
For more info. regarding the operation of each logic gate separately, you can check out the link below:

https://whatis.techtarget.com/definition/logic-gate-AND-OR-XOR-NOT-NAND-NOR-and-XNOR

1) Basic Construction of AND & OR gate from universal gates (NAND and NOR respectively)



We need two NAND gates to create an AND gate. The first NAND gate returns HIGH if either input LOW is or both inputs are LOW. Then the second NAND gate is configured as a NOT gate to invert the output of the first NAND gate.

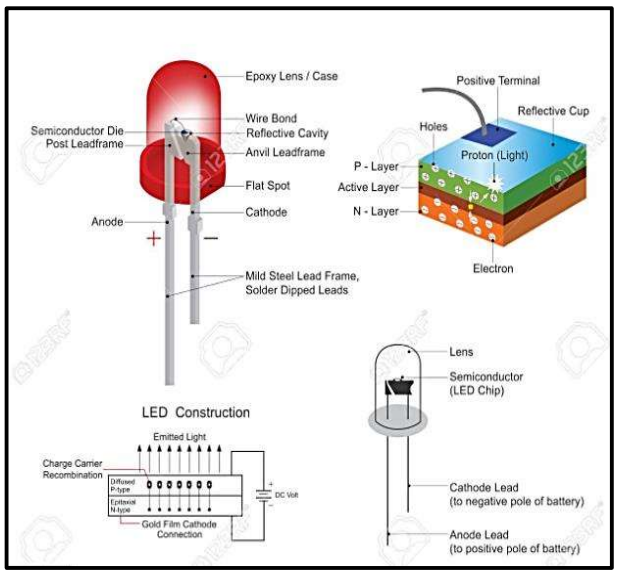


Similarly, now we need two NOR gates to create an OR gate. The first NOR gate

returns LOW if either input is HIGH or both inputs are HIGH. Then the second NOR gate is configured as a NOT gate to invert the output of the first NOR gate.

❖ LED & 7 Segment Display

A light-emitting diode (LED) is a semiconductor device that emits light when an electric current is passed through it. (**Longer Lead – Anode and Shorter Lead – Cathode)**



For 7 Segment Display - Refer Page no. 8 point (2)

Exp. Analysis

❖ Project Pre-requisites

1) Hardware:

Different Logic gates, Mux, Breadboard, 4-bit Binary adder, Decoder driver, 7 segment Led Display wires LED

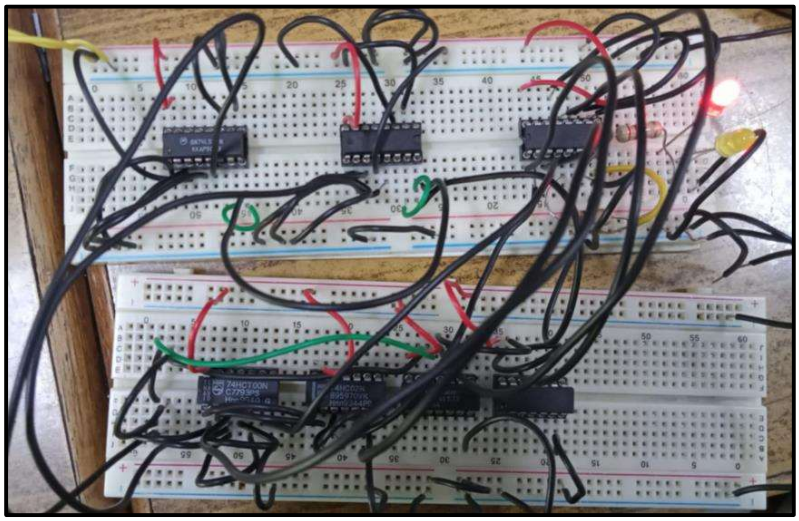
2) Software:

Tinker CAD: Having an estimate about the **I.C. internal structure** and **pin diagram of combinational circuits.**

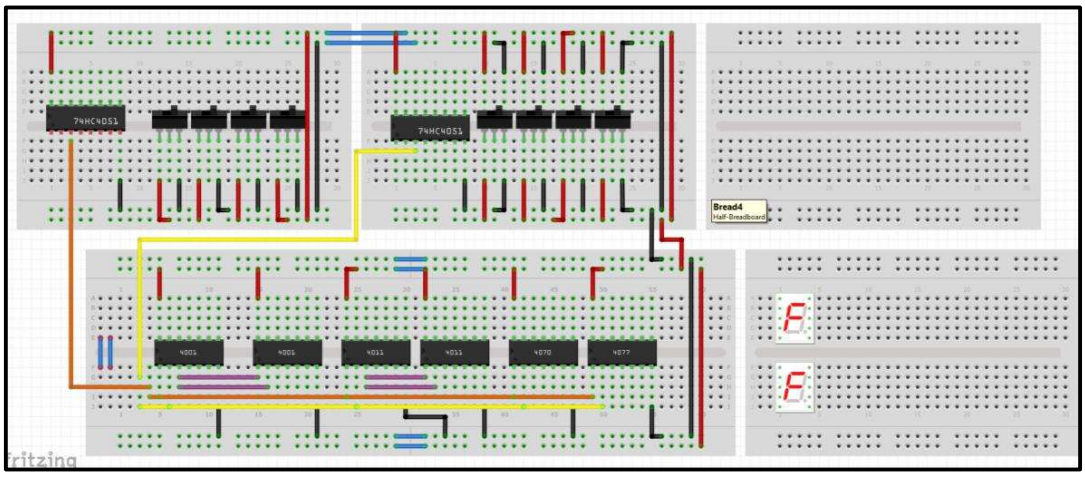
Fritzing: Having an estimate about the **hardware connections on Breadboard.**

Proteus: Having an idea about, **how to design the circuit** in several stages.

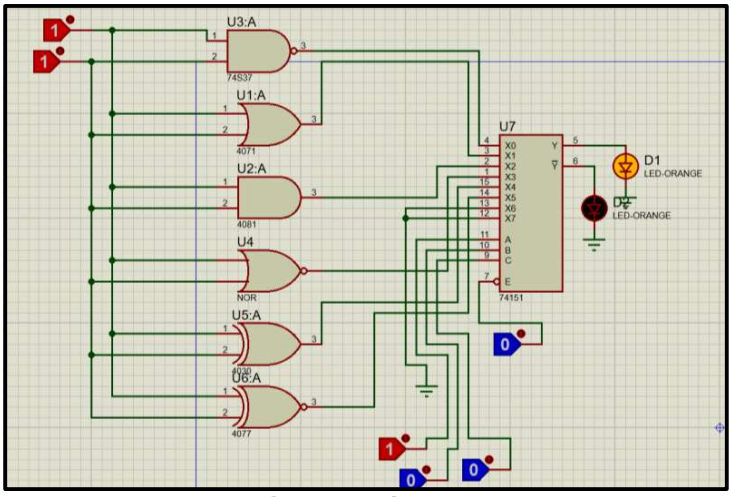
❖ Some Images of initial tries



(Fault: NOR gates input get connected in reversed order and Selection of outputs from mux doesn't takes place)



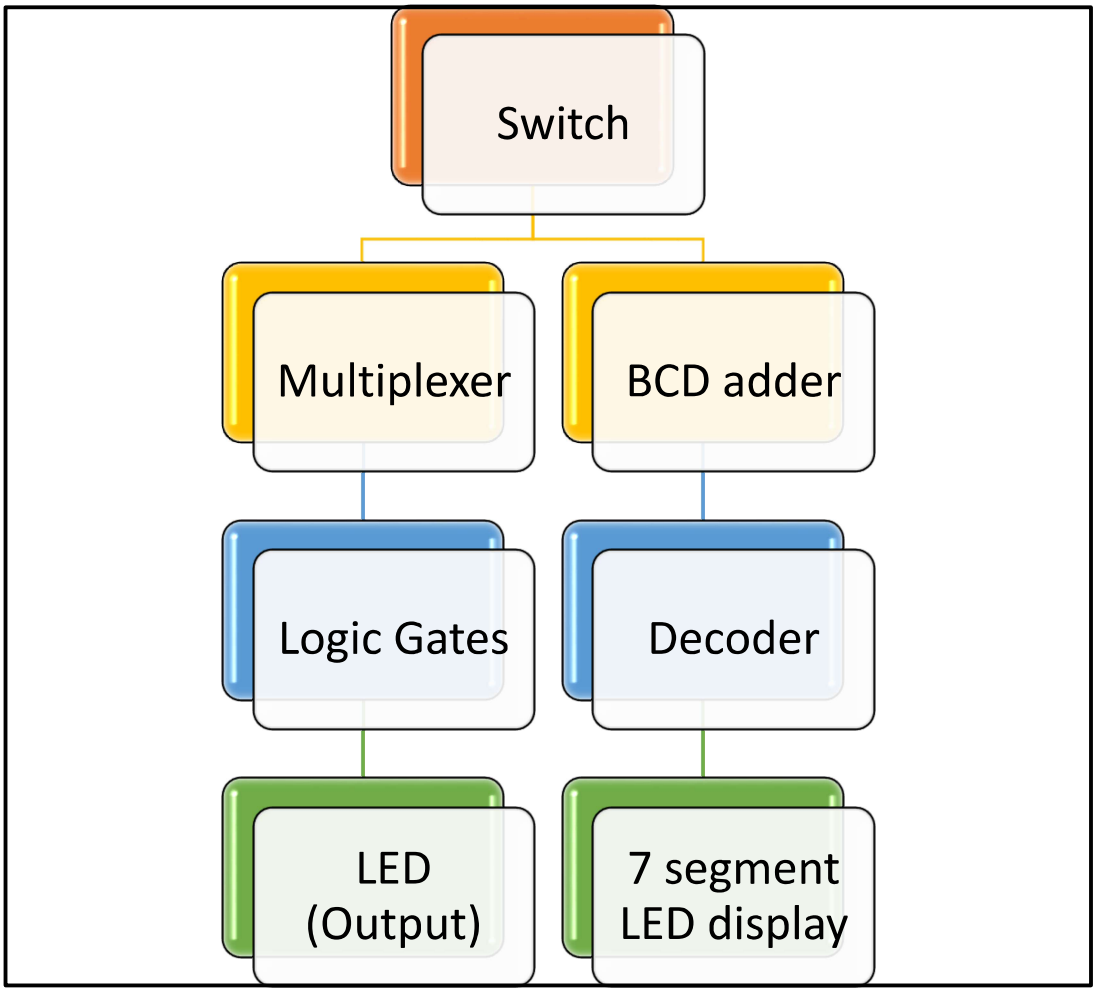
(**Fault:** Wrong use of concepts while using logic gates, **Disadvantage:** Some I.C. are not available)



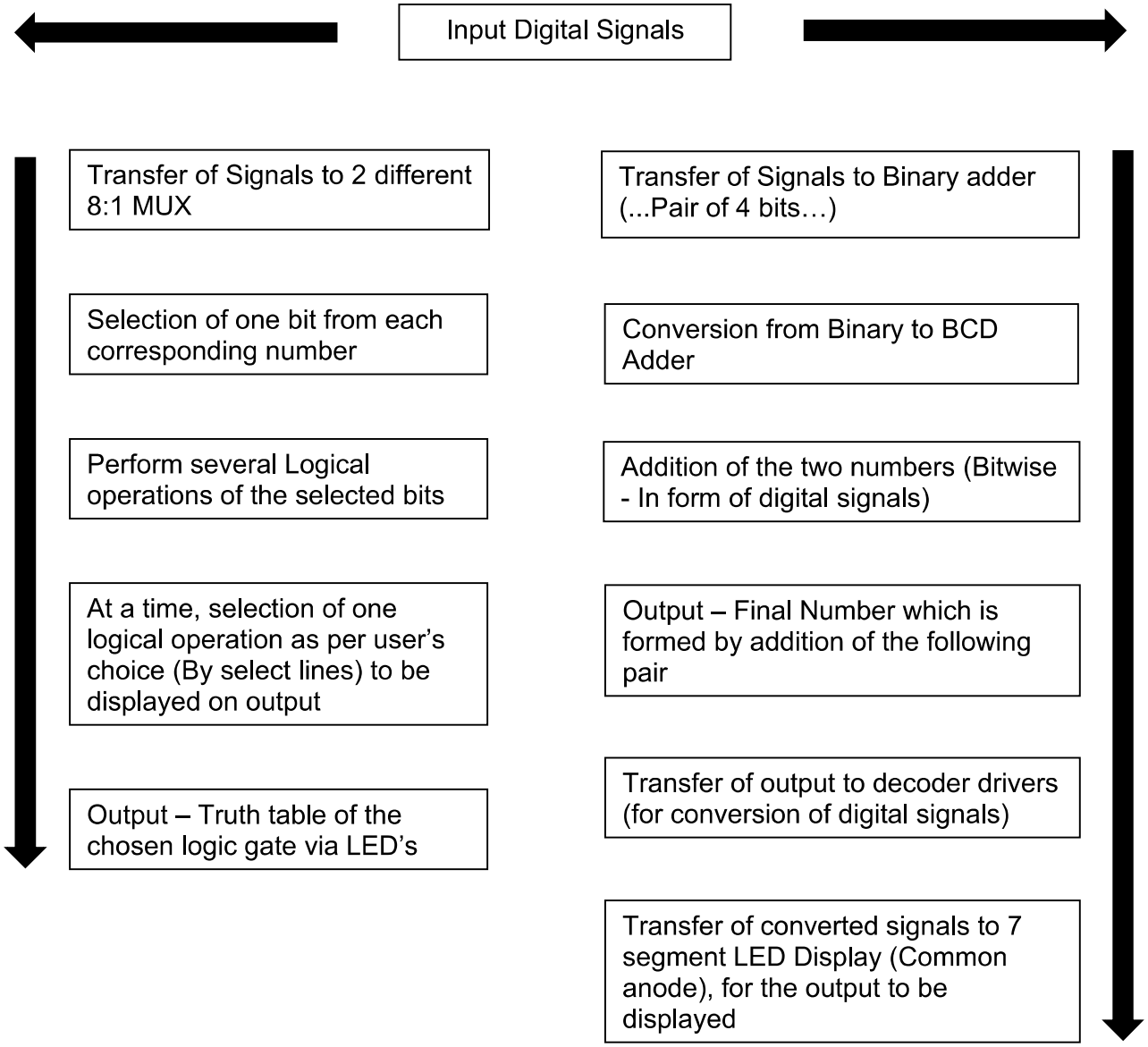
…First Successful attempt…

(**Advantage:** Learn how to design a proper circuit by dividing it in 2-4 subparts, properly understand the working of whole circuit)

❖ Block Diagram



❖ Flow Chart

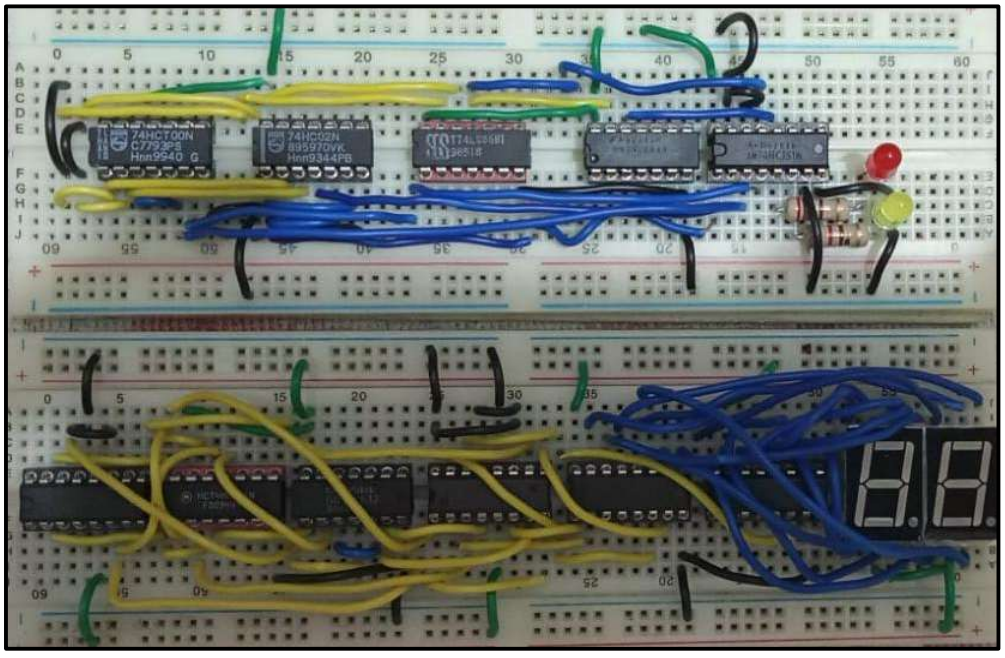

Input Digital Signals
Transfer of Signals to 2 different 8:1 MUX
Selection of one bit from each corresponding number
Perform several Logical operations of the selected bits
At a time, selection of one logical operation as per user's choice (By select lines) to be displayed on output
Output – Truth table of the chosen logic gate via LED's
Transfer of Signals to Binary adder (...Pair of 4 bits…)
Conversion from Binary to BCD Adder
Addition of the two numbers (Bitwise - In form of digital signals)
Output – Final Number which is formed by addition of the following pair
Transfer of output to decoder drivers (for conversion of digital signals)
Transfer of converted signals to 7 segment LED Display (Common anode), for the output to be displayed

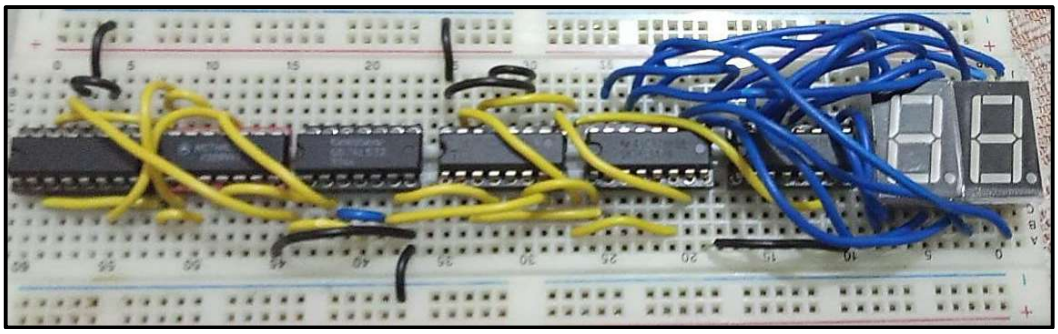
❖ Debugging on Hardware

After having the proper idea about the design and working of circuit on proteus software, now the turn comes to put out the circuit on hardware.

This time, a neat and clean circuit was made by us on 3 different breadboards.

- **Image for the same are provide below:**





(Fault: Not able to get proper outputs at LED display even after 6 different tries…)

(**Final Result:** While connecting 3rd pin of both Decoder driver to LOW, found that 7 segment LED displays are broken, all 7 led segments are not glowing…)

❖ Final ALU Design