

**Gazi University**

**Faculty of Engineering**  
**Microprocessors Laboratory**

**Boolean Algebra Calculator**



Name	Student ID
Ömer Mert Yurdakul	181112041
Oğuz Yalçın	181110075
Muhammed Murat Gökteş	171180029
Ahmet Furkan Karataş	181180046

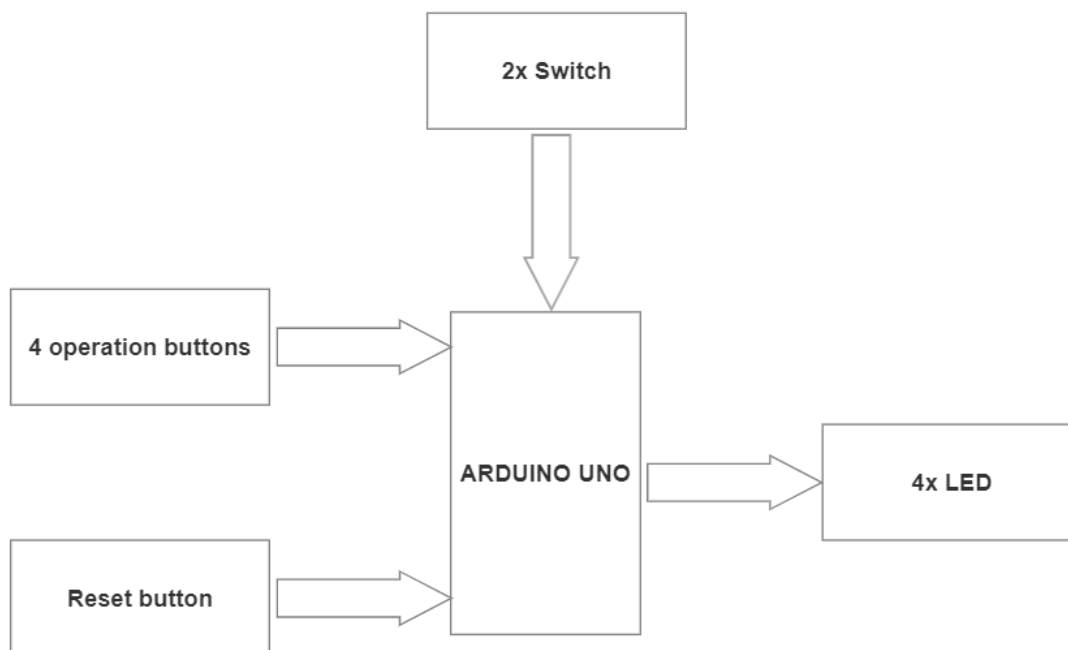
## Table Of Contents

1. Description	6. Flowchart
2. Block Diagram	7. Pseudocode
3. Hardware	8. Software
4. Components list	9. Implementation
5. Circuit Schematic	

### 1. Description

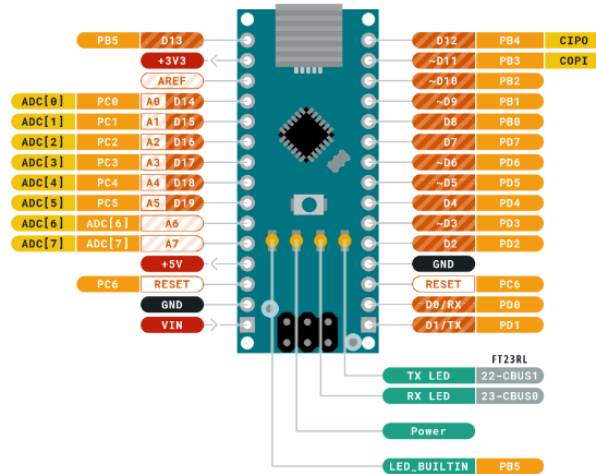
The subject of our project is 'boolean algebra calculator'. That is, to make a calculator that can perform four operations, such as addition, subtraction, multiplication, division. We decided to implement this project using arduino. In this design that works with user input, the user must first decide which two numbers should be subjected to one of the four operations. Here we have used two dip switches that have 4 ports for each one. One of these dip switches corresponds to the first number, while the other corresponds to the second number. The user can set the desired number on the binary base with the keys on the ports. After that, there are 5 buttons in the project we have realized. 4 of these buttons represent addition, subtraction, multiplication and division. The users will decide which operation will be performed of the 2 binary numbers they have selected with the help of these buttons. At the other stage, we used 4 leds to show the result after performing the operations selected by the user. The user can understand what the result is by looking at the status of these leds at the output. Finally, when the users wants to perform a new operation, They can use the reset button, which is the fifth button, to reset the operations and the result, and then follow the steps he has done from the beginning for different situations and get new results.

### 2. Block Diagram



### 3. Hardware

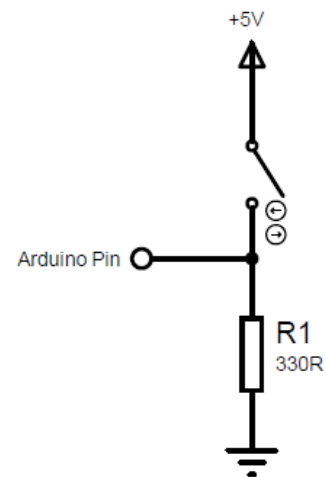
In the hardware part we used Arduino Nano as microcontroller because of the price/performance and large variety of sources and powered with a USB cable. The pinout is given in the below figure.



<https://docs.arduino.cc/hardware/nano>

In order to take two input numbers, we used two 4bit-dip switches each switch represents a one bit of the number. The top part is connected to 5V of the Arduino and the bottom part is connected with pull-down resistors to ground as seen in the figure. The signals are taken from the resistor and connected to the D2-D9 pins. Due to the pull-down connection, the switches create high signal when switched to on and creates low signal switched to off.

There are five push button which are used as operators as summation, subtraction, multiplication, division and clear. The connection of the buttons are similar to the dip switches the only difference is the they are connected to the A0-A4 pins of the Arduino.



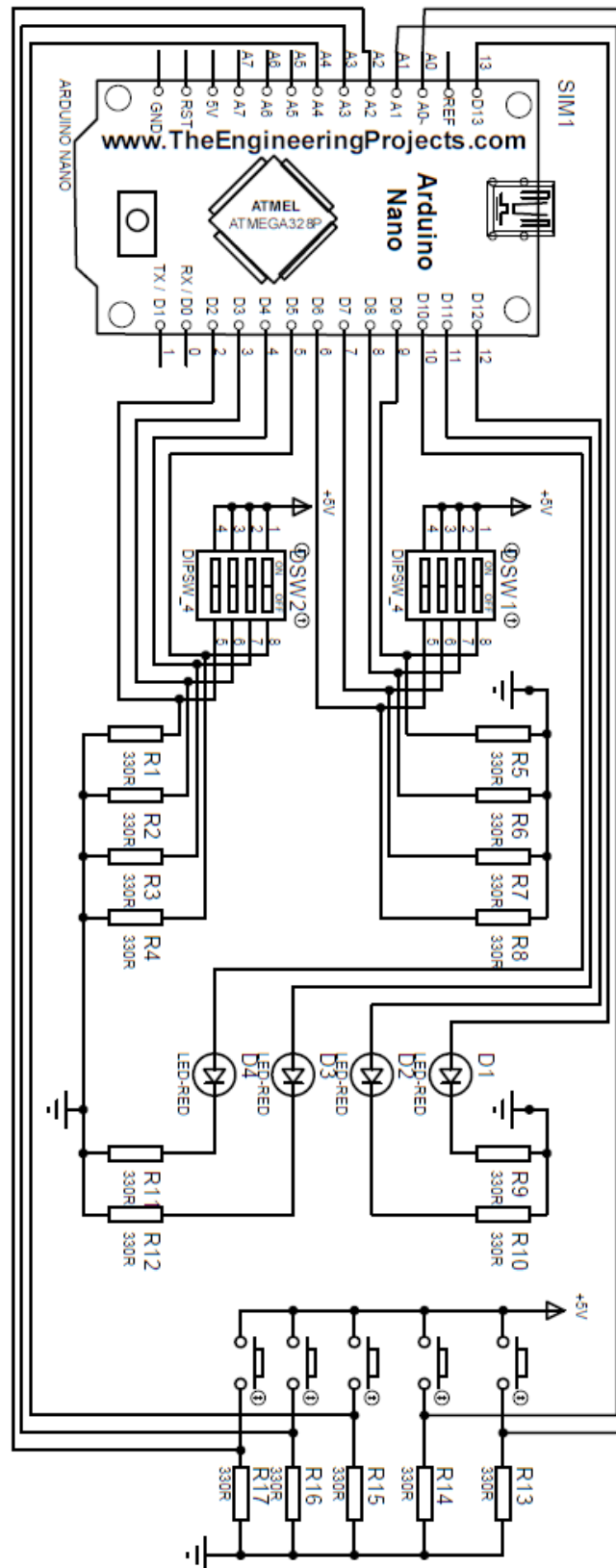
**Pull-Down Resistor**

In order to visualise the output, we used four led. The positive legs are connected to D10-D13 of Arduino and negative legs are connected to ground with 330ohm resistor to protect against overcurrent.

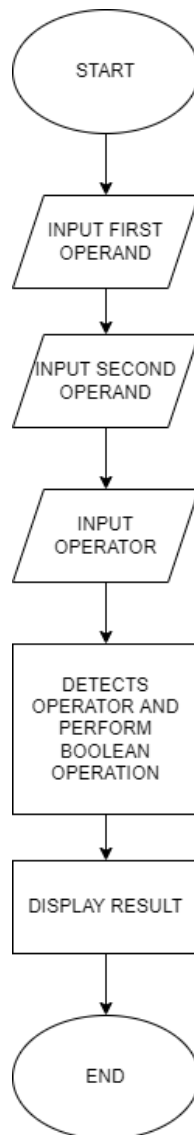
### 4. Components list

Component Name	Amount
Arduino Nano	1
4bit-dip switch	2
Push button	5
Green Led	4
330ohm Resistor	17

## 5. Circuit Schematic



## 6. Flowchart



## 7. Pseudocode

```
BEGIN
NUMBER first_number, second_number, result
CHAR operator
INPUT first_number
INPUT second_number
INPUT operator

IF operator == "+"
    CALCULATE result = first_number + second_number
ELSE IF operator == "-"
    CALCULATE result = first_number - second_number
ELSE IF operator == "*"
    CALCULATE result = first_number * second_number
ELSE IF operator == "/"
    CALCULATE result = first_number / second_number
END IF
    DISPLAY result
END
```

## 8. Software

In this project we firstly used tinkercad website for simulation. We created the circuit then we were thinking about software. We coded with C++ language. We decided to use array, for and switch case. Firstly, we assigned pin numbers. We have two numbers and we have 4 key for each numbers. Then we assigned led pins and button pins. In setup function we coded INPUTS and OUTPUTS. We created 2 functions; GetNumbers and ShowOutputs. GetNumbers function, read the pins and give the numbers to loop function. ShowOutput function, write the result for leds. In loop function we made math processing. Division, Multiplication, Subtraction, Summation process and assign the result.

```
1 int number1Pins[4] = {5, 4, 3, 2}; //4bit first number pins
2 int number2Pins[4] = {9, 8, 7, 6}; //4bit second number pins
3 int resultNumberPins[4] = {13, 12, 11, 10}; //4bit result number pins
4 int operatorPins[5] = {A0, A1, A2, A3, A4}; //sum, sub, mul, div, clear
5
6 uint8_t number1 = 0; //first number
7 uint8_t number2 = 0; //second number
8 uint8_t result = 0; //result number
9
10 void setup() {
11     Serial.begin(9600);
12
13     for(int i = 0; i < 4; i++)
14     {
15         pinMode(number1Pins[i], INPUT); //first number pins set as INPUT
16         pinMode(number2Pins[i], INPUT); //second number pins set as INPUT
17         pinMode(resultNumberPins[i], OUTPUT); //result number pins set as OUTPUT
18     }
19
20     for(int i = 0; i < 5; i++)
21         pinMode(operatorPins[i], INPUT); //operator pins set as INPUT
22
23
24 }
25
26 void loop() {
27
28     for(int i = 0; i < 5; i++)
29     {
30         // Checking if any operator button is pressed
31         if(digitalRead(operatorPins[i])){
32             GetNumbers();
33             switch(i)
34             {
35                 case 0: //Clear
36                     result = 0;
37                     break;
38                 case 1: //Division
39                     result = number1 / number2;
40                     break;
```

```

41     case 2: //Multiplication
42         result = number1 * number2;
43         break;
44     case 3: //Subtraction
45         result = number1 - number2;
46         break;
47     case 4: //Summation
48         result = number1 + number2;
49         break;
50     }
51     ShowOutput(result);
52 }
53 }
54
55 delay(100);
56 }
57
58 void GetNumbers()
59 {
60     for(int i = 0; i < 4; i++)
61     {
62         int temp = digitalRead(number1Pins[i]);
63         bitWrite(number1, i, temp);
64     }
65
66     for(int i = 0; i < 4; i++)
67     {
68         int temp = digitalRead(number2Pins[i]);
69         bitWrite(number2, i, temp);
70     }
71 }
72
73 void ShowOutput(int result)
74 {
75     for(int i = 0; i < 4; i++)
76         digitalWrite(resultNumberPins[i] ,bitRead(result, i));
77
78 }

```

## 9. Implementation

