



Embedded Systems Programming
(4CS016)

Report Writing

Mini-Projects - Score Board

Student Id : 2059789
Student Name : Prasiddha Regmi
Group : <L4CG9>
Lecturer Name : Mr. Sumanta Silwal
Module Leader : Dr. Sarah Slater
Submitted on : 27-08-2021
Word Count without code : 3154
Word Count Overall : 3683

Table of Contents

| | |
|--|----|
| Acknowledgment: | 1 |
| Abstract: | 2 |
| Introduction:..... | 3 |
| Some Notes:..... | 3 |
| Why I choose this circuit..... | 4 |
| How I have completed this project..... | 4 |
| Tools and component Use:..... | 5 |
| About these tools and components: | 6 |
| Tinkercad:..... | 6 |
| Arduino Uno R3:..... | 7 |
| 7-Segment Decoder: | 8 |
| Cathode 7 Segment Display: | 9 |
| Pushbutton..... | 10 |
| 330 Ω Resistor/10 k Ω Resistor/220 Ω Resistor | 11 |
| NPN Transistor (BJT) | 12 |
| LCD 16 x 2..... | 13 |
| Circuit Diagram & Frizing:..... | 14 |
| Program:..... | 15 |
| First Arduino Code | 15 |
| Second Arduino Code..... | 16 |
| Working Principal: | 22 |
| Code Explanation of 1st Arduino Board:..... | 23 |
| Code Explanation of 2nd Arduino Code:..... | 24 |
| Testing:..... | 25 |
| Conclusion:..... | 27 |
| Reference:..... | 28 |

Acknowledgment:

I would like to thank Herald College and Mr. Sumanta Silwal for letting me work on such a wonderful project where I could develop my skills, and push myself even harder to learn and innovate new things. I would like to express my gratitude towards Mr. Sumanta Silwal for guiding me all the way through this project. He has been the pillar to support me throughout this project.

Abstract:

The goal of my project is to interface a scoreboard with a PushButton to simultaneously keep statistics and run a scoreboard for wrestling, volleyball, and basketball while controlling an external scoreboard similar to those found in every sporting arena. This project also provides a “Welcome message and creator name by scrolling in display in LCD” using a separate Arduino board. We can increase or decrease and reset by pressing the push button. At the End of the game, we can display the winner team at the display by inserting any message in the code.

Introduction:

This project is a scoring game circuit using 7 segment display that can be used for all occasions when needed. The circuit is based on Arduino Uno R3, 7-Segment Decoder, Cathode 7 Segment Display, Pushbutton, 330/1000 Ω Resistor, 220 Ohms Resistors, NPN Transistor (BJT) & Bread Board. There are 3 pushbuttons in the circuit. When S1 is pressed it increases the score of player 1, S2 is pressed then decreases the score of player 1. Similarly, S3 and S4 also increase or decrease the score of player 2. Where 7-Segment Decoder will decode the count to display it on the seven-segment LED display. And S5 Button helps in resetting our Score Board. It is about the working of the circuit. The circuit is designed on the basics of counters and by decoding the binary value using a 7-segment decoder to pass the voltage into the 7-Segment display and show the score of the game. I have also joined one other small project to display Welcome message and Creator Name using a separate Arduino board that scrolls from Right to Left. Also, we can display any other message by inserting easily into the code.

Some Notes:

- Please wait for some time to simulate this circuit.
- Press S5 to reset the Score Board.
- Now press S1 or S2 to increase or decrease the Score of Player 1. Similarly, S3 and S4 also increase or decrease the score of player 2
- Press one button at each time on a scoreboard.
- The circuit can be powered from a 5V Supply using Arduino Board

Why I choose this circuit.

I choose this circuit because nowadays most people play games but they can write the score on the whiteboard. It does not look like good practice. I have seen in many games where people write the scores on a whiteboard and erase the score and again write. I find it strange to see people erasing the score and writing the score each time whenever the score goes up when it can be easily done using this digital scoreboard. This scoreboard not only makes the job easier but also makes it more practicable. I have made this scoreboard in tinker-cad after that I will again make this same type of circuit in real life to make it easier to show score in the game.

How I have completed this project

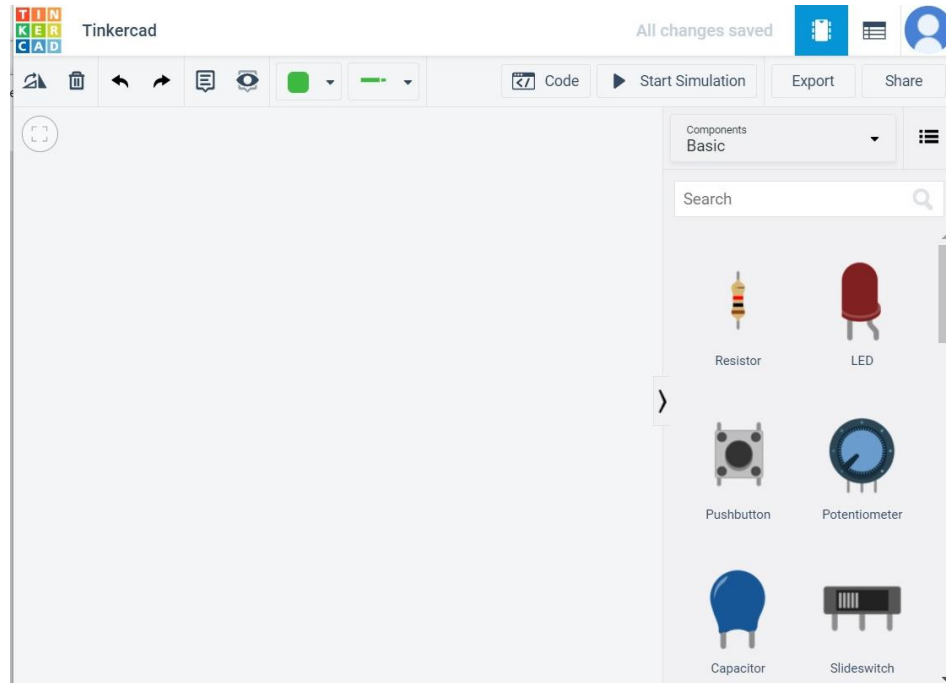
I have completed this project by myself but I cleared some of my doubts with the module teacher and I also took help from Friends, YouTube, and google. This project was confusing. At first, I use only one 7 segment display for both players. When I ran and increased the score of the circuit it did not show one digit score. Again, I have joined 2 two 7-segment displays to show two digits score. This was one of the most challenging aspects for me. By completing these projects I have learned many new ideas and gained knowledge. I also want to make a name scrolling board using many led but tinkercad doesn't support more led and it hangs when running. Then I make a Welcome message scrolling using a 16*2 LCD.

Tools and component Use:

| Name | Quantity | Component |
|--|----------|---------------------------|
| U1, U4 | 2 | Arduino Uno R3 |
| U2, U3 | 2 | 7-Segment Decoder |
| Digit1, Digit2, Digit3, Digit4 | 4 | Cathode 7 Segment Display |
| S1, S2, S3, S4, S5 | 5 | Pushbutton |
| R1, R2, R3, R4, R5, R7, R8, R9, R10, R11, R12, R13, R14, R15 | 14 | 330 Ω Resistor |
| R16, R17, R18, R19, R20, R21, R22, R23, R24 | 9 | 10 k Ω Resistor |
| T1, T2, T3, T4 | 4 | NPN Transistor (BJT) |
| R6 | 1 | 220 Ω Resistor |
| U5 | 1 | LCD 16 x 2 |

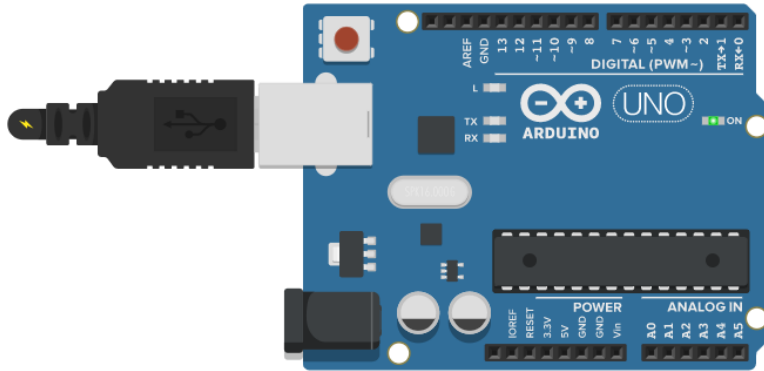
About these tools and components:

Tinkercad:



Tinkercad is a free and easy-to-use application for 3D Design, electronics circuits, and coding. It may not be the best design tool for creating complex designs, but the program can provide support, structure. It is a Virtual app that also helps to simulate different types of circuits.

Arduino Uno R3:



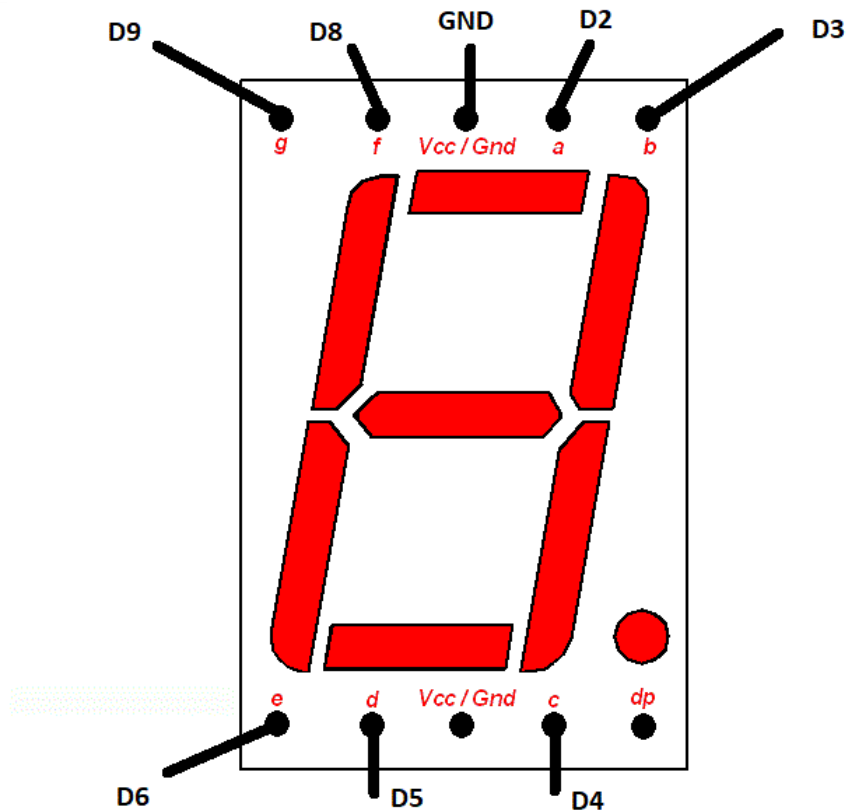
Arduino Uno R3 is a microcontroller board based on the ATmega328 AVR microcontroller. It has 20 digital input and output pins, of which 6 can be used as PWM outputs and 6 can be used as analog inputs. We can load the program using the computer to do more work using code. We can also use 5.0/3.5 V power using this board. So it has a big hand in this circuit. Mainly it is used for any projects.

7-Segment Decoder:



A BCD to 7-segment decoder is the CD4511. It accepts a binary number as an input and uses its outputs to show it on a 7-segment display. This seven-segment BCD decoder has 4 input lines (Input 1, Input 2, Input 3, and Input 4) and 7 output lines (a, b, c, d, e, f, and g). This output is provided to seven LED displays Segment, the displayed decimal number depends on the input.

Cathode 7 Segment Display:

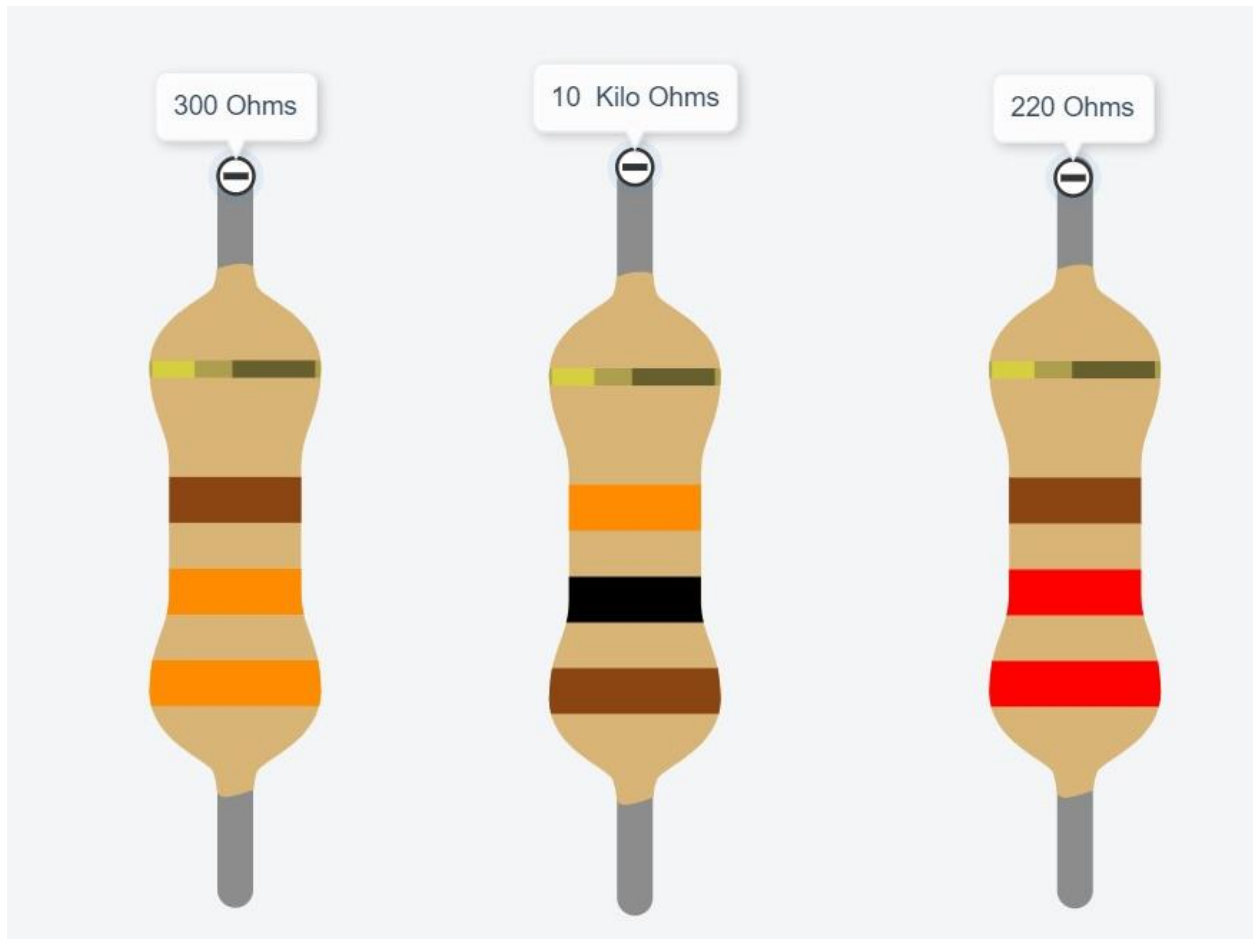


There are two types of 7-segment display and they are common cathode & common anode. The difference between the two displays is that the common cathode connects all the 7-segment cathodes directly, while the common anode connects all the 7-segment anodes. In the common cathode seven segments, all the cathode points are connected together and grounded while the anode points are connected to power using the appropriate segment. In my projects, I use the common cathode where all the common cathodes are connected together and power is connected through a 7-Segment decoder.

Pushbutton

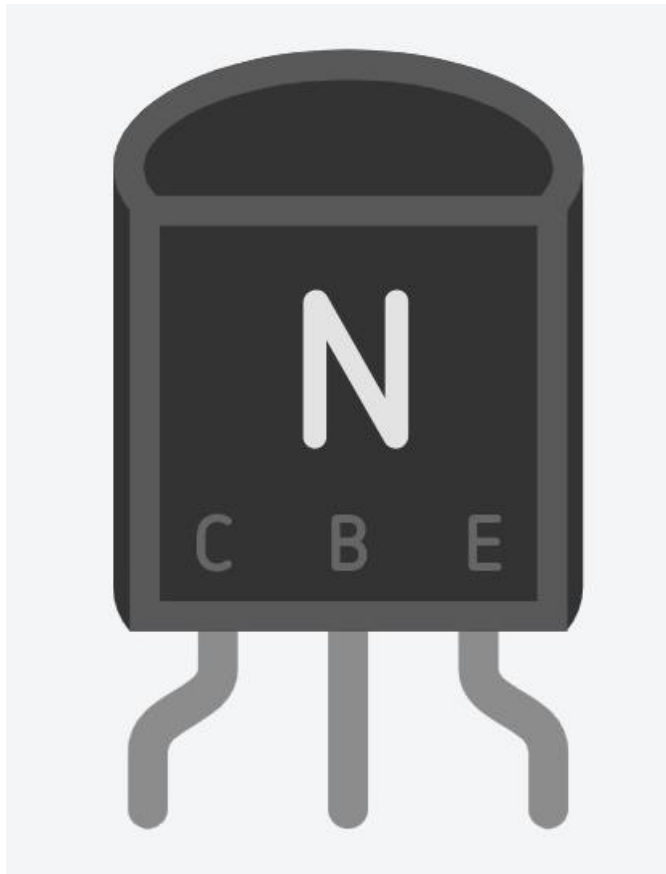


A Pushbutton is a simple switching device that controls a machine or any type of circuit. Pushbuttons are made from a hard material, usually plastic or metal. The surface is mainly flat or shaped to fit a finger or a human hand, making it easier to push. Pushbutton is used to pass the current when the peoples press the switch. There is a spring inside of the button and when peoples press the switch then it became a pressed state where the current will pass. Again the people pressed the switch then it became an unpressed state where the current stop to pass from that button and the machine or device not works. It is the most used component in any circuit or project. There are four pins in the pushbutton where two pins on the left side are already connected every time. Also, The two pins on the right side are connected. When the current pass then there will connect all 4 pins. We need to know that "push" the button means pressing, pushing, pressing, folding, knocking, and punching the button.

330 Ω Resistor/10 k Ω Resistor/220 Ω Resistor

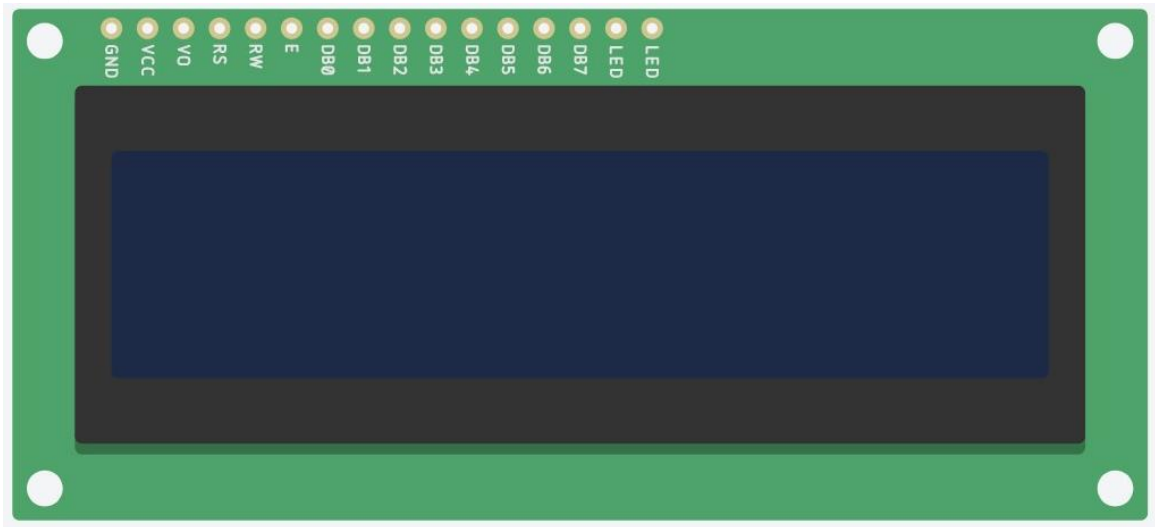
Resistors are electrical components that limit or regulate the flow of current in electronic circuits. The main purpose of resistors is to reduce the flow of current and voltage in any particular part of the circuit. It is made of copper wire wound on a ceramic rod, and the outside of the resistor is coated with insulating paint. There are different types of the resistor and they have a different value. Mainly it is used for all types of the circuit to control the flow of current. I have used resistors in my projects having different values 300 Ohms, 10 Kilo Ohms, and 220 Ohms. The color of the resistor will differ based on its resistance.

NPN Transistor (BJT)



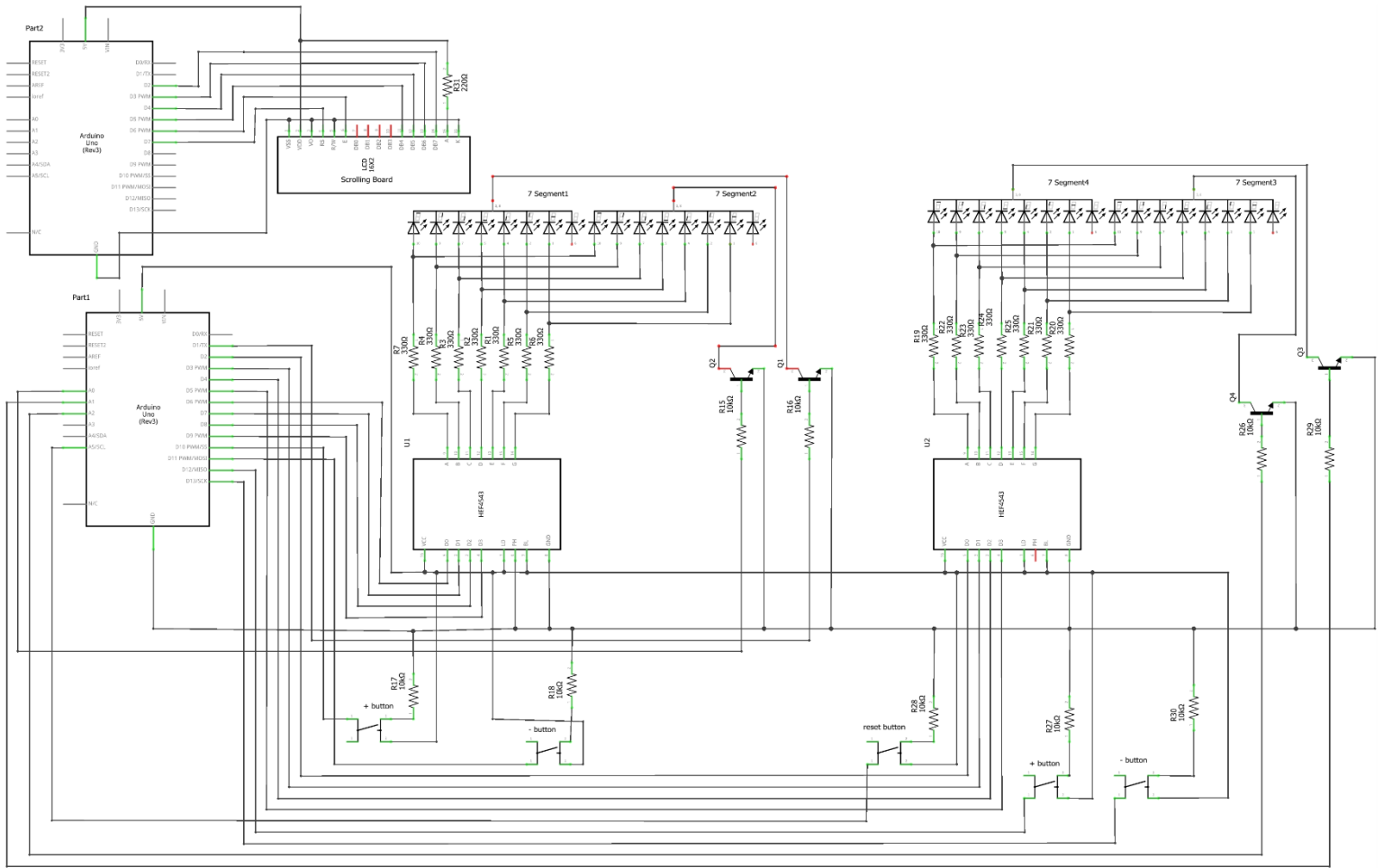
The NPN transistors are the most widely used bipolar junction transistors. It is constructed by sandwiching a P-type semiconductor between two N-type semiconductors. The NPN transistor has three terminals: collector, emitter, and base. NPN transistors behave like two PN junction diodes connected back to back. These back-to-back PN junction diodes are called collector-base junction and base-emitter. In my projects the base is connected with floating pins and collector is connected with common cathode of seven segment Led. And, the emitter is connected with ground. It is mainly used in switching applications.

LCD 16 x 2



The term LCD stands for liquid crystal display. It is an electronic display module with a wide range of applications, such as various circuits and equipment such as mobile phones, calculators, and computers. These displays are mainly used for multi-segment and seven-segment LEDs. The main advantage of using this module is economy; simple programming, animation, and no restrictions on displaying custom characters or even special animations. In my projects there is used for showing the some message by scrolling.

Circuit Diagram & Frizing:



Program:

First Arduino Code

```
//Name: Prasiddha Regmi  
//Student_ID: 2059789  
//Date: 2021/08/22
```

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(7,6,5,4,3,2);
```

```
char inp1[]= {"WELLCOME TO MY MINI-PROJECTS"};  
char inp2[]= {"CREATED-BY:PRASIDDHA REGMI "};
```

```
void setup()  
{  
  lcd.begin(28,2);  
}  
void loop()  
{  
  static int num1 = 0;  
  static int num2 = 0;  
  if (num1 >=28){  
    if (num1== 29){  
      lcd.clear();  
      num1 = 0;  
    }  
    else  
    {  
      lcd.scrollDisplayLeft();  
      delay(300);  
    }  
  }  
  else{  
    lcd.setCursor(num1,0);  
    lcd.print(inp1[num1]);  
    num1 =num1 + 1;  
    lcd.setCursor(num2,1);  
    lcd.print(inp2[num2]);  
    num2=num2+1;  
  }  
}
```

Second Arduino Code

```
//Name: Prasiddha Regmi
```

```
//Student_ID: 2059789
```

```
//Date: 2021/08/22
```

```
int input[]{2,3,4,5};
int input2[]{6,7,8,9};
int dig2[]{A1,A2};
int dig[]{1,A0};
int puls[]{10,11,12,13,A5};
int y=0,x=0;
int y1=0,y2=0;
int x1=0,x2=0;

void setup(){
  for(int i=0;i<4;i++){
    pinMode(input[i],OUTPUT);
    pinMode(input2[i],OUTPUT);
  }
  for(int i=0;i<2;i++){
    pinMode(dig[i],OUTPUT);
    pinMode(dig2[i],OUTPUT);
  }
  for(int i=0;i<5;i++){
    pinMode(puls[i],INPUT);
  }
}

void loop(){
  if(digitalRead(puls[0])==HIGH){
    y++;
    while(digitalRead(puls[0])==HIGH){}
    y1=y/10;
    y2=y%10;
    number2(y1,0);
    delay(10);
    number2(y2,1);
    delay(10);
  }
  else if(digitalRead(puls[1])==HIGH){
    y--;
    if(y<0)
      y=0;
    while(digitalRead(puls[1])==HIGH){}
    y1=y/10;
```

```
y2=y%10;
number2(y1,0);
delay(10);
number2(y2,1);
delay(10);
}

else if(digitalRead(puls[2])==HIGH){
    x++;
    while(digitalRead(puls[2])==HIGH){}
    x1=x/10;
    x2=x%10;
    number(x1,0);
    delay(10);
    number(x2,1);
    delay(10);
}
else if(digitalRead(puls[3])==HIGH){
    x--;
    if(x<0)
        x=0;
    while(digitalRead(puls[3])==HIGH){}
    x1=x/10;
    x2=x%10;
    number(x1,0);
    delay(10);
    number(x2,1);
    delay(10);
}
else if(digitalRead(puls[4])==HIGH){
    while(digitalRead(puls[4])==HIGH){}
    y=0;
    x=0;
    y1=y/10;
    y2=y%10;
    number2(y1,0);
    delay(20);
    number2(y2,1);
    delay(20);
    x1=x/10;
    x2=x%10;
    number(x1,0);
    delay(10);
    number(x2,1);
    delay(10);
}
```

```
else{
    number2(y1,0);
    delay(10);
    number2(y2,1);
    delay(10);
    number(x1,0);
    delay(50);
    number(x2,1);
    delay(10);
}

}

void number2(int n, int k){
    if(k==0){
        digitalWrite(dig[0],HIGH);
        digitalWrite(dig[1],LOW);
    }
    if(k==1){
        digitalWrite(dig[0],LOW);
        digitalWrite(dig[1],HIGH);
    }
    if(n==0){
        for(int i=0;i<4;i++)
            digitalWrite(input2[i],LOW);
    }
    if(n==1){
        digitalWrite(input2[0],HIGH);
        for(int i=1;i<4;i++)
            digitalWrite(input2[i],LOW);
    }
    if(n==2){
        digitalWrite(input2[0],LOW);
        digitalWrite(input2[1],HIGH);
        digitalWrite(input2[2],LOW);
        digitalWrite(input2[3],LOW);
    }
    if(n==3){
        digitalWrite(input2[0],HIGH);
        digitalWrite(input2[1],HIGH);
        digitalWrite(input2[2],LOW);
        digitalWrite(input2[3],LOW);
    }
    if(n==4){
        digitalWrite(input2[0],LOW);
        digitalWrite(input2[1],LOW);
    }
}
```

```
digitalWrite(input2[2],HIGH);
digitalWrite(input2[3],LOW);
}
if(n==5){
digitalWrite(input2[0],HIGH);
digitalWrite(input2[1],LOW);
digitalWrite(input2[2],HIGH);
digitalWrite(input2[3],LOW);
}
if(n==6){
digitalWrite(input2[0],LOW);
digitalWrite(input2[1],HIGH);
digitalWrite(input2[2],HIGH);
digitalWrite(input2[3],LOW);
}
if(n==7){
digitalWrite(input2[0],HIGH);
digitalWrite(input2[1],HIGH);
digitalWrite(input2[2],HIGH);
digitalWrite(input2[3],LOW);
}
if(n==8){
digitalWrite(input2[0],LOW);
digitalWrite(input2[1],LOW);
digitalWrite(input2[2],LOW);
digitalWrite(input2[3],HIGH);
}
if(n==9){
digitalWrite(input2[0],HIGH);
digitalWrite(input2[1],LOW);
digitalWrite(input2[2],LOW);
digitalWrite(input2[3],HIGH);
}
}

void number(int n, int k){
if(k==0){
    digitalWrite(dig2[0],HIGH);
    digitalWrite(dig2[1],LOW);
}
if(k==1){
    digitalWrite(dig2[0],LOW);
    digitalWrite(dig2[1],HIGH);
}
}
if(n==0){
    for(int i=0;i<4;i++)
```

```
    digitalWrite(input[i],LOW);
}
if(n==1){
    digitalWrite(input[0],HIGH);
    for(int i=1;i<4;i++)
        digitalWrite(input[i],LOW);
}
if(n==2){
    digitalWrite(input[0],LOW);
    digitalWrite(input[1],HIGH);
    digitalWrite(input[2],LOW);
    digitalWrite(input[3],LOW);
}
if(n==3){
    digitalWrite(input[0],HIGH);
    digitalWrite(input[1],HIGH);
    digitalWrite(input[2],LOW);
    digitalWrite(input[3],LOW);
}
if(n==4){
    digitalWrite(input[0],LOW);
    digitalWrite(input[1],LOW);
    digitalWrite(input[2],HIGH);
    digitalWrite(input[3],LOW);
}
if(n==5){
    digitalWrite(input[0],HIGH);
    digitalWrite(input[1],LOW);
    digitalWrite(input[2],HIGH);
    digitalWrite(input[3],LOW);
}
if(n==6){
    digitalWrite(input[0],LOW);
    digitalWrite(input[1],HIGH);
    digitalWrite(input[2],HIGH);
    digitalWrite(input[3],LOW);
}
if(n==7){
    digitalWrite(input[0],HIGH);
    digitalWrite(input[1],HIGH);
    digitalWrite(input[2],HIGH);
    digitalWrite(input[3],LOW);
}
if(n==8){
    digitalWrite(input[0],LOW);
    digitalWrite(input[1],LOW);
```

```
digitalWrite(input[2],LOW);  
digitalWrite(input[3],HIGH);  
}  
if(n==9){  
digitalWrite(input[0],HIGH);  
digitalWrite(input[1],LOW);  
digitalWrite(input[2],LOW);  
digitalWrite(input[3],HIGH);  
}  
}
```

Working Principal:

I have used two Arduino boards. So, In first Arduino board, it displays a given text in the LCD that scrolls from left to right. This project consists of only just a few lines of code with a single character array. In the encoding technique, I select a character from the array and print it to the first position on the LCD, then move it to the second position, then print the second character to the first position, and then the first character moves to the third position. The second and third characters of location and the second position are printed in the first position. Applying your technique to the character makes the character appear to be moving.

Now in the second Arduino Board, I have used a 7-segment decoder, resistor, 7- segment displays, transistor, pushbutton breadboard, wire, and Arduino board. In this digital scoreboard circuit, I have displayed the score in 7 segment display. In this circuit, I have joined two 7-segment displays by joining with each other of the same pin to shows the two digits score. In transistor the negative value goes to the emitter and the base is joined with the digital pins of the Arduino and the collector is connected with the common cathode of the 7 segment display.

When we press the Plus pushbutton then the high value goes to PIN 10 and according to code it adding 1 to the previous value and this signal goes to the 7-segment decoder and it gives a high value to the specific pin of the 7 segment display then it shows the score. Similarly, when we pressed the – push button of player 1 then the high value goes to the PIN 11 and according to the code it subtracts 1 from the previous value and this signal goes to 7-segment decoder and it gives a specified value to the 7 segment displays then it shows the score And when we press the reset button then the high value goes to analog value A0 and according to the code zero is divided by previous displayed value and that value goes to 7 segment decoder then it gives the high value to the specific pins of the 7 segment decoder and this makes 0. Similarly for player 2 score increase/decrease/reset by the same method. And For display double digits, the working process is that when we press the push button more than 9 times then the number of pressed times modulus 10 then it gives quotient and remainder after that the quotient value goes to right side of 7 segment display and remainder goes to left slide of 7 segment display then is shows the score.

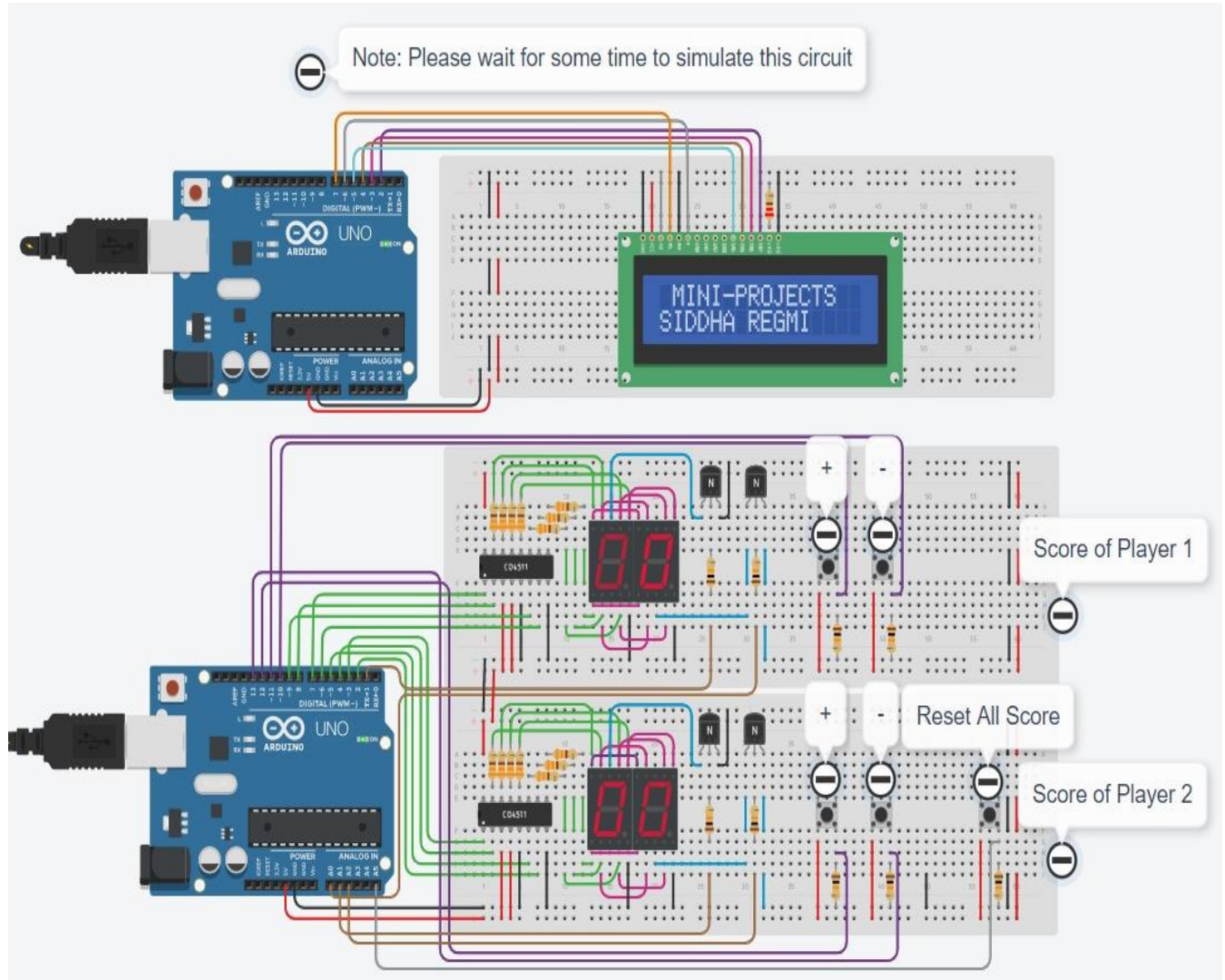
Code Explanation of 1st Arduino Board:

A library "<LiquidCrystal.h>" has been called to allow the Arduino to control LCD. Through calling this library we were able to display the desired texts in the LCD. Then we assigned the connected pins in the library. After this, two variables were assigned to display the desired input in the LCD. Then, in the setup function "lcd.begin(28,2);" was used to define the number of rows and columns the LCD has and initialize the LCD. This function must always be called before we call other functions. Then two integer variables were assigned 0 to count the number of columns the LCD displays and if the count reaches more than 28(which is the total number of characters on each row) the screen of the screen of the LCD will be cleared using the function "lcd.clear();" after which the value of the variable will be made 0 again. And if the number of the integer variable becomes 28 the function "lcd.scrollDisplayLeft();" has been used because the data that is needed to be shown in the LCD is greater than the amount of data the LCD can display, and so this function will display the data by scrolling from the left first. And now to display the data firstly "lcd.setCursor(num1,0);" function is used to set the cursor on the desired place in the LCD. Afterwards, "lcd.print()" function has been used to display the character of the data and to display the another character the num1 variable is increased by one each time to move the cursor next to the previously displayed character. In this way all the data in the row1 is displayed and in the same manner the second row data is also displayed. So, ultimately this is how the desired data are displayed in the LCD.

Code Explanation of 2nd Arduino Board:

Firstly, various lists had been assigned to the pins based on their functions they do in the circuit. Then 3 pairs of two variables (i.e. x, y, x1, y1, x2, y2) were assigned the value 0 which would later be used to count the scores. The variable x and y records the scores of one team each. Then in the “void setup()” function the pins were declared whether they will give output or take input. And now in the main program every time the pushbutton used to increase the score is pressed the value of the x or the y variable will increase by 1 based on which side’s button is pressed. And to avoid increasing the value of x or y variable when the button is kept on being pressed while loop has been used with an empty loop until the button is no longer being pressed and so as long as the button is kept on being pressed the program will be stuck in the while loop. Then the digits stored in the variable x or y will be divided into two parts and each part will be assigned to x1 or y1 and x2 or y2 respectively. x1 or y1 variable will consist of the digit having the value in tens place and x2 or y2 variable will consist of the digit having ones place. Then the value x1 or y1 will be assigned to the function “number(x1, 0)” or “number2(y1,0)” and the value of x2 or y2 will be assigned to the function “number(x2,1)” or “number2(y2,1)” after a slight delay. (Note: the function of the function “number(,)” and “number2(,)” will be explained later on). Each time the subtracting button is pressed the same code used to add the score is but the only difference is each time the button is pressed the value of variable x or y is subtracted by 1 instead of adding 1 each time and rest of the function is same. And if the reset button is pressed the value of x and y variable will be made 0 again. Then the functions “number(0,0)”, “number(0,1)”, “number2(0,0)” and “number2(0,1)” are called with delay between each function. And when none of the button is pressed the functions “number(0,0)”, “number(0,1)”, “number2(0,0)” and “number2(0,1)” are called with delay between each function. In both “number()” and “number2()” function the first variable is to determine the which digit should be shown in the 7 Segment Display (the pins connected with the 7-Segment Decoder will send current based or not send the current based on which number must be displayed). And so with the help of 7-Segment Decoder the necessary digits will be displayed in the LCD .

Testing:

Without pressing any pushbutton

As I have mentioned in the introduction part of this project there are 5 push buttons in the circuit. When the first button S1 is pressed we can see a rise in score of the player 1., pressing S2 will lead to the drop in score. S3 and S4 also have the similar functions as S1 and S2 but for player 2. The 7-Segment Decoder will decode the count to display it on the Seven Segment LED display. S5 helps to reset our score board. I have also joined a separate Arduino board for displaying the welcome message and the Creators name which scrolls from right to left in the display.

After pressing pushbutton

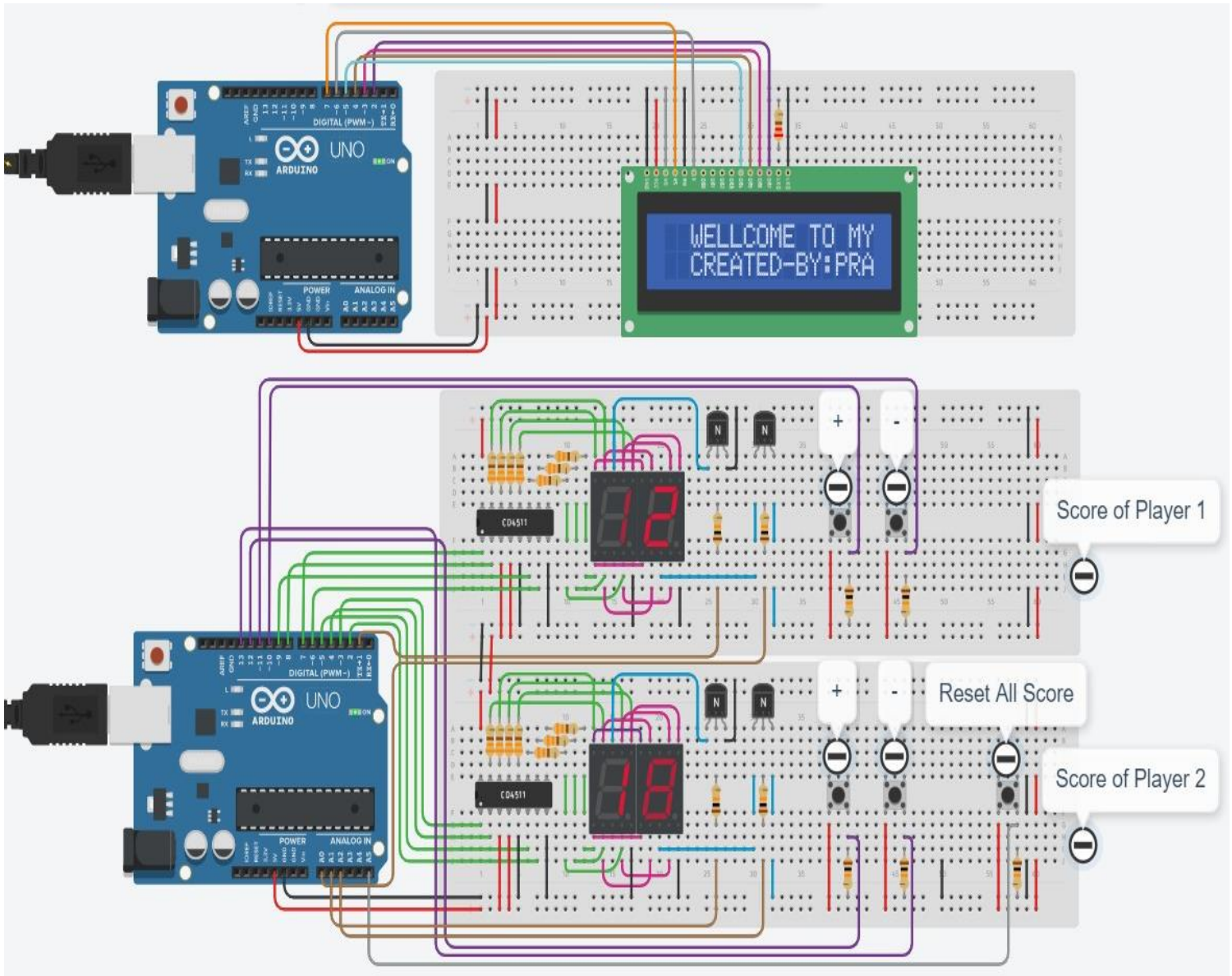


Fig: Circuit of Score-Board

Note: To copy this circuit please go through this link:

<https://www.tinkercad.com/things/bJeWMWaffYt>

Conclusion:

Overall, this project is very practical and useful in many ways. The main aim of this overall project is to make the work of the score tracker easier. This project is more efficient because it is easier to just push button to increase or decrease or reset the score than writing and erasing scores in a board. In real life it is small so, we can easily carry from one place to another place. Mainly this type of project use for school sport because it is cheap and easy to use. Also, it displays the given text in LCD which helps to show the winning team at the last with the help of the computer. This project is made with the help of 7-segment decoder, 7 segment display, 2 Arduino, Breadboard, Resistors, wires, NPN transistors, Push-buttons, LCD, wires and some simple coding. And is some games like Basketball the score doesn't always increase by 1 so this exact project may not be usable for Basketball. But this can be tackled by adding more push buttons to add more variety of scores. Sometime there may be incidents where a person might mistakenly press the button due to which the score may increase and to tackle such problems a push button is present in the circuit with the function to decrease the score. And after the game is over, the scores of both teams can easily be brought back to 0 by simply pressing the pushbutton meant to reset the scores.

Reference:

(Arduino, 2021)

<https://store-usa.arduino.cc/products/arduino-uno-rev3/>

(GeeksforGeeks, 2021)

<https://www.geeksforgeeks.org/bcd-to-7-segment-decoder/>

(Electronics Tutorials, 2020)

<https://www.electronics-tutorials.ws/blog/7-segment-display-tutorial.html>

(Herga, 2020)

<https://www.herga.com/news-media/technical-blog-archive/what-is-a-push-button-switch->

(electronicsnotes, 2018)

https://www.electronics-notes.com/articles/electronic_components/resistors/resistor-types.php

(Circuit Globe, 2021)

<https://circuitglobe.com/npn-transistor.html>

(Circuit, 2021)

<https://www.allaboutcircuits.com/projects/interface-an-lcd-with-an-arduino/>

(Homemade Circuit Projects, 2020)

<https://www.homemade-circuits.com/making-electronic-scoreboard-using-ic/>

(Electro-Tech-Online.com, 2018)

<https://www.electro-tech-online.com/threads/pure-logic-gates-basketball-scoreboard.153123/>