



Universitatea POLITEHNICA din București
Facultatea de Inginerie Industrială și Robotică



ROBOTICĂ 1 – LABORATOR 6

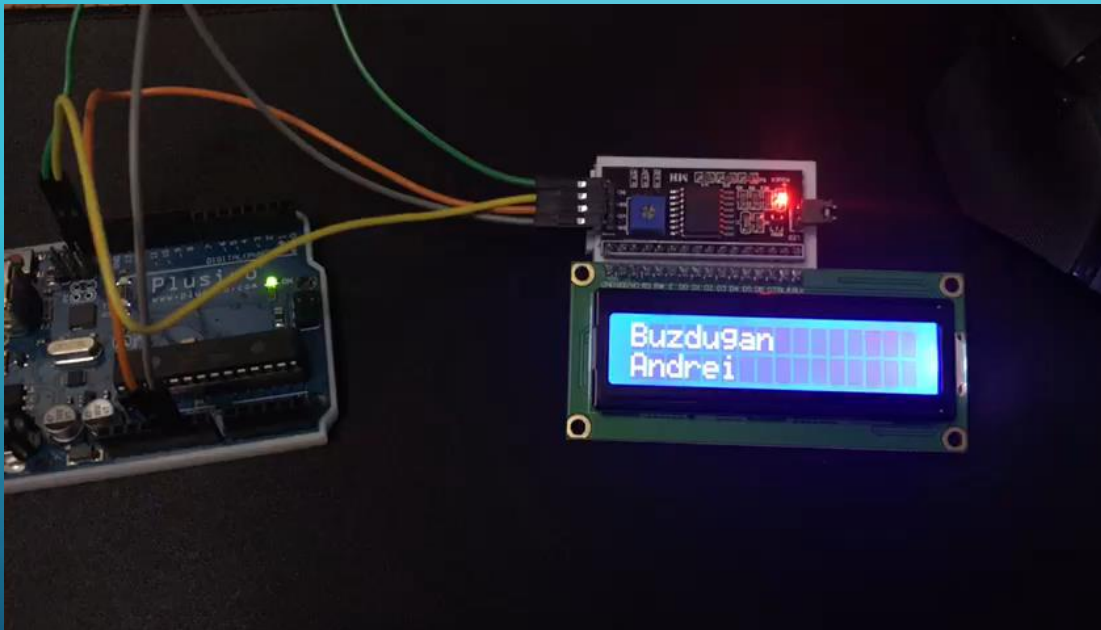
Student: BUZDUGAN Andrei

Coordonator: Ileana DUGĂEȘESCU

APLICAȚIA 1

Asamblarea și programarea unui circuit utilizând LCD. Afișarea unui mesaj pe ecran.

Aici am conectat un LCD 16 x 2 la un I2C, drept urmare din cei 16 pini de pe LCD, am ramas numai cu 4 pini de pe I2C, pe care i-am conectat la placuta arduino uno cu GND, 5V, SDA si SCL. SDA si SCL se regasesc pe placuta arduino la capatul sirului de pini mama de la digital, ultimul fiind SCL, iar penultimul SDA.



CODUL

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

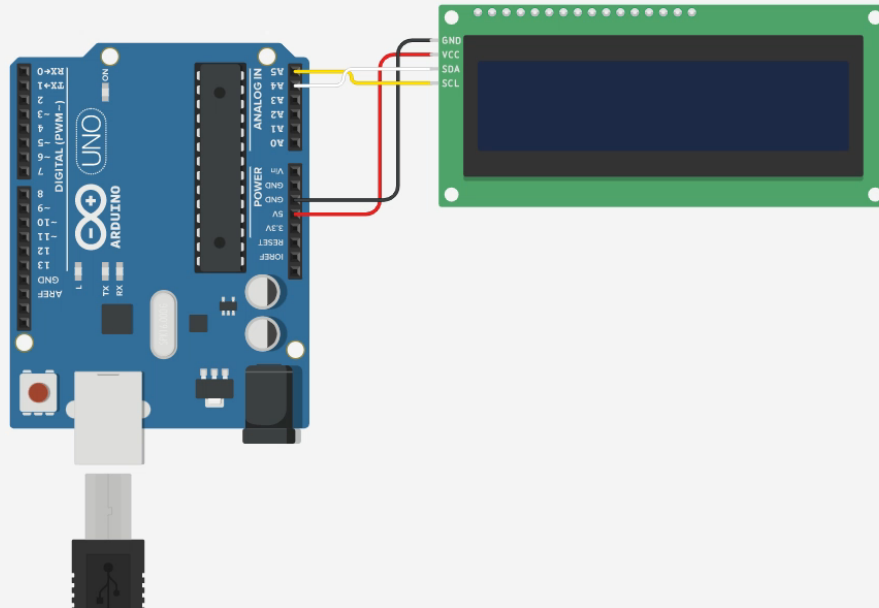
LiquidCrystal_I2C lcd(0x27,16, 2);

void setup() {
  lcd.setBacklight(1);
  lcd.init();
  lcd.backlight();
  lcd.clear();
}

void loop() {
  lcd.setCursor(0, 0);
  lcd.print("Buzdugan");
  lcd.setCursor(0, 1);
  lcd.print("Andrei");
}
```

APLICAȚIA 1

Asamblarea și programarea unui circuit utilizând LCD. Afișarea unui mesaj pe ecran în TINKERCAD.



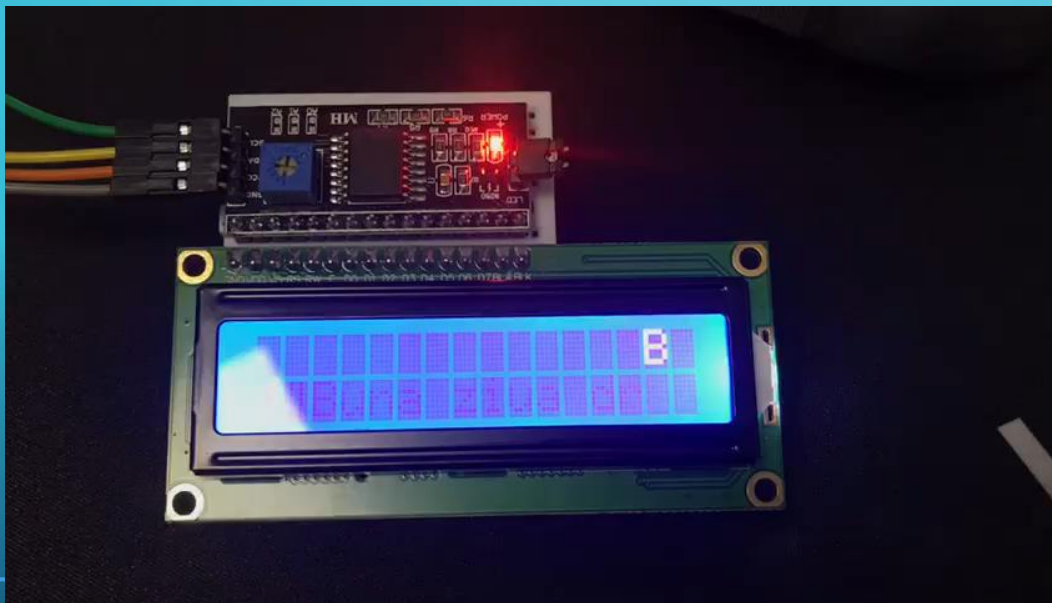
Text

```
1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 LiquidCrystal_I2C LCD(0x27,16,2);
5
6
7 String Nume;
8 String Prenume;
9
10
11 void setup()
12 {
13   Wire.begin();
14
15   LCD.init();
16   LCD.backlight();
17
18   Nume = "Buzdugan";
19   Prenume = "Andrei";
20
21
22   LCD.clear();
23   LCD.setCursor(0,0);
24   LCD.print(Nume);
25   LCD.setCursor(0,1);
26   LCD.print(Prenume);
27 }
28
29
30 void loop()
31 {
32 }
```

APLICAȚIA 2

Asamblarea și programarea unui circuit utilizând LCD. Derularea unui mesaj pe ecran.

Aici am realizat derularea unui text pe LCD, cu același montaj de circuit ca la aplicația 1, dar cu un cod diferit.



CODUL

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16, 2);

char array1[]="Buna ziua draga lume!";
char array2[]="La revedere!";

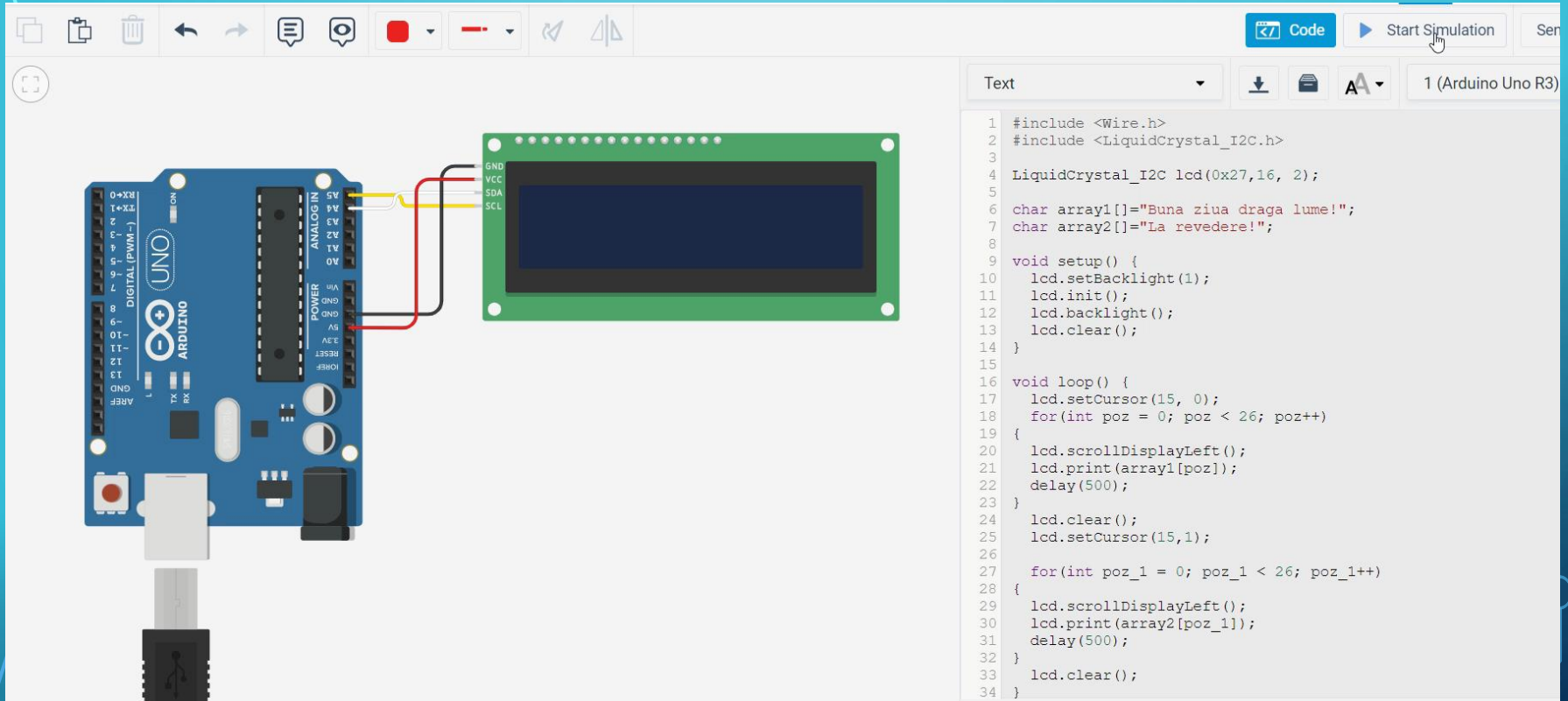
void setup() {
  lcd.setBacklight(1);
  lcd.init();
  lcd.backlight();
  lcd.clear();
}

void loop() {
  lcd.setCursor(15, 0);
  for(int poz = 0; poz < 26; poz++)
  {
    lcd.scrollDisplayLeft();
    lcd.print(array1[poz]);
    delay(500);
  }
  lcd.clear();
  lcd.setCursor(15,1);

  for(int poz_1 = 0; poz_1 < 26; poz_1++)
  {
    lcd.scrollDisplayLeft();
    lcd.print(array2[poz_1]);
    delay(500);
  }
  lcd.clear();
}
```

APLICAȚIA 2

Asamblarea și programarea unui circuit utilizând LCD. Derularea unui mesaj pe ecran în TINKERCAD.



The screenshot shows the Tinkercad software interface. On the left, an Arduino Uno R3 is connected to an I2C LCD display. The connections are as follows:

- Red wire: 5V (Arduino) to VCC (LCD)
- Black wire: GND (Arduino) to GND (LCD)
- Yellow wire: A5 (Arduino) to SCL (LCD)
- Blue wire: A4 (Arduino) to SDA (LCD)

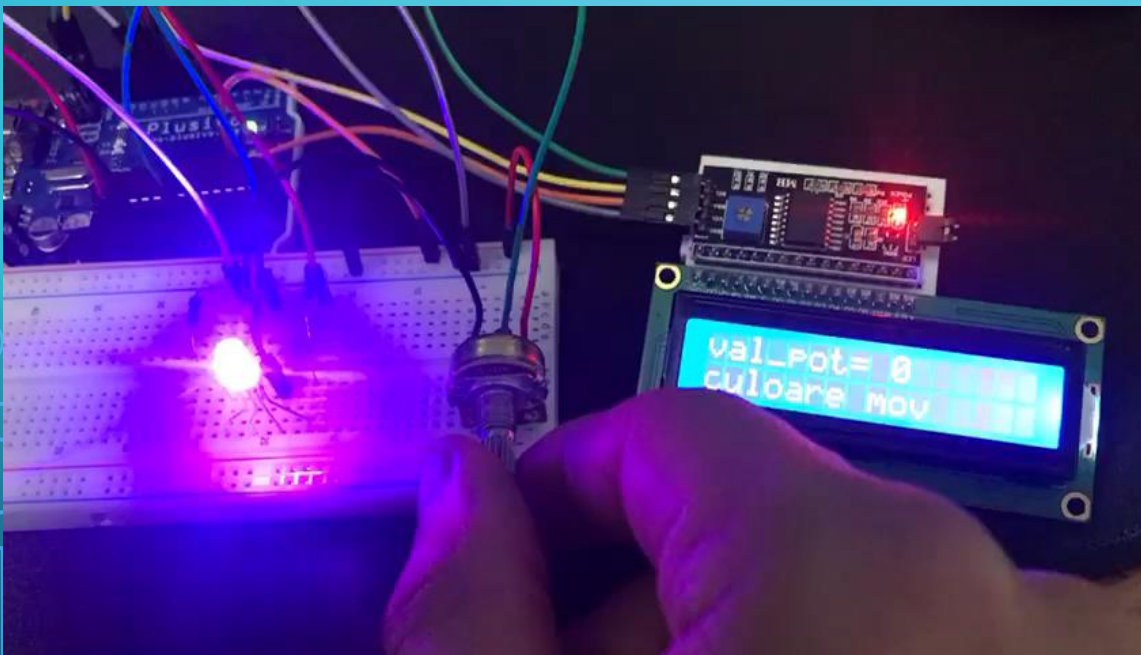
On the right, the code editor shows the following C++ code:

```
1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 LiquidCrystal_I2C lcd(0x27,16, 2);
5
6 char array1[]="Buna ziua draga lume!";
7 char array2[]="La revedere!";
8
9 void setup() {
10   lcd.setBacklight(1);
11   lcd.init();
12   lcd.backlight();
13   lcd.clear();
14 }
15
16 void loop() {
17   lcd.setCursor(15, 0);
18   for(int poz = 0; poz < 26; poz++)
19   {
20     lcd.scrollDisplayLeft();
21     lcd.print(array1[poz]);
22     delay(500);
23   }
24   lcd.clear();
25   lcd.setCursor(15,1);
26
27   for(int poz_1 = 0; poz_1 < 26; poz_1++)
28   {
29     lcd.scrollDisplayLeft();
30     lcd.print(array2[poz_1]);
31     delay(500);
32   }
33   lcd.clear();
34 }
```


APLICAȚIA 3

Asamblarea și programarea unui circuit cu led RGB, potențiometru și LCD.

Aici am m-am adăugat un led de tip RGB cu 3 rezistoare de 220 de Ohm pe care l-am legat la pinii Digitali de pe placa plus un potentiometru cu care comut culorile de pe led-ul RGB, și le citesc împreună cu valoarea lui pe un LCD.



CODUL

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16, 2);

int red=9;
int green=10;
int blue=11;
int poten1 = A0;
int val_pot;

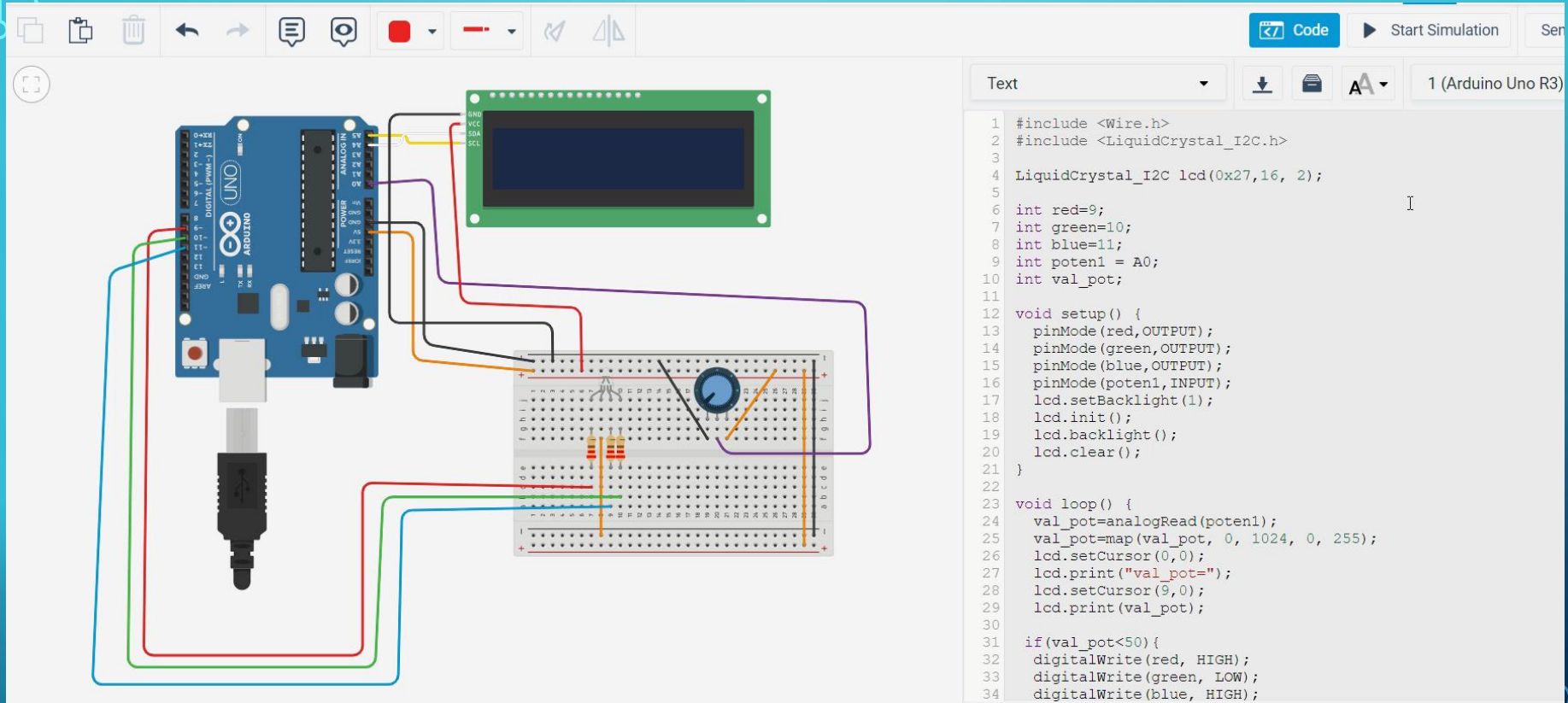
void setup() {
  pinMode(red,OUTPUT);
  pinMode(green,OUTPUT);
  pinMode(blue,OUTPUT);
  pinMode(poten1,INPUT);
  lcd.setBacklight(1);
  lcd.init();
  lcd.backlight();
  lcd.clear();
}

void loop() {
  val_pot=analogRead(poten1);
  val_pot=map(val_pot, 0, 1024, 0, 255);
  lcd.setCursor(0,0);
  lcd.print("val_pot=");
  lcd.setCursor(9,0);
  lcd.print(val_pot);

  if(val_pot<50){
    digitalWrite(red, HIGH);
    digitalWrite(green, LOW);
    digitalWrite(blue, HIGH);
    lcd.setCursor(0,1);
    lcd.print("culoare mov");
    delay(500);}
  else{
    digitalWrite(red, LOW);
    digitalWrite(green, HIGH);
    digitalWrite(blue, HIGH);
    lcd.setCursor(0,1);
    lcd.print("culoare albastru");
    delay(500);}
}
```

APLICAȚIA 3

Asamblarea și programarea unui circuit cu led RGB, potențiomtru și LCD în TINKERCAD.



The screenshot displays the Tinkercad software interface. On the left, a circuit is assembled featuring an Arduino Uno R3 board connected to an LCD screen and a breadboard. The breadboard contains an RGB LED strip and a potentiometer. Wires connect the Arduino's digital pins to the LED's anode, cathode, and ground, and its analog pin to the potentiometer's wiper. The LCD is connected to the Arduino's I2C pins. On the right, the code editor shows the following C++ code:

```
1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 LiquidCrystal_I2C lcd(0x27,16, 2);
5
6 int red=9;
7 int green=10;
8 int blue=11;
9 int poten1 = A0;
10 int val_pot;
11
12 void setup() {
13   pinMode(red,OUTPUT);
14   pinMode(green,OUTPUT);
15   pinMode(blue,OUTPUT);
16   pinMode(poten1,INPUT);
17   lcd.setBacklight(1);
18   lcd.init();
19   lcd.backlight();
20   lcd.clear();
21 }
22
23 void loop() {
24   val_pot=analogRead(poten1);
25   val_pot=map(val_pot, 0, 1024, 0, 255);
26   lcd.setCursor(0,0);
27   lcd.print("val_pot=");
28   lcd.setCursor(9,0);
29   lcd.print(val_pot);
30
31   if(val_pot<50){
32     digitalWrite(red, HIGH);
33     digitalWrite(green, LOW);
34     digitalWrite(blue, HIGH);
```

APLICAȚIA 4

Realizarea și programarea unui circuit utilizând componente la alegere.

Pe baza circuitului asamblat pentru aplicatia 1 am realizat un cod mai complex cu ajutorul internetului, prin care am realizat diferite caractere prin combinatii de biti intr-o matrice.

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
byte inima[8] = {
0b00000,
0b01010,
0b11111,
0b11111,
0b01110,
0b00100,
0b00000,
0b00000
};
```

```
byte clopotel[8] = {
0b00100,
0b01110,
0b01110,
0b01110,
0b01110,
0b11111,
0b00000,
0b00100,
0b00000
};
```

```
byte omulet[8] = {
0b11111,
0b10101,
0b10101,
0b11111,
0b11111,
0b01110,
0b01010,
0b11011,
0b00000
};
```

```
byte verificat[8] = {
0b00000,
0b00001,
0b00011,
0b10110,
0b11100,
0b01000,
0b00000,
0b00000
};
```

```
byte difuzor[8] = {
0b00001,
0b00011,
0b01111,
0b01111,
0b01111,
0b00011,
0b00001,
0b00000
};
```

```
byte sunet[8] = {
0b00001,
0b00011,
0b00101,
0b01001,
0b01001,
0b01011,
0b11011,
0b11000
};
```

```
byte craniu[8] = {
0b00000,
0b01110,
0b10101,
0b11011,
0b01110,
0b01110,
0b00000,
0b00000
};
```

```
byte lacat[8] = {
0b01110,
0b10001,
0b10001,
0b11111,
0b11011,
0b11011,
0b11111,
0b00000
};
```

```
void setup()
{
  lcd.init();
  lcd.backlight();
  lcd.createChar(0, inima);
  lcd.createChar(1,
clopotel);
  lcd.createChar(2,
omulet);
  lcd.createChar(3,
verificat);
  lcd.createChar(4,
difuzor);
  lcd.createChar(5, sunet);
  lcd.createChar(6, craniu);
  lcd.createChar(7, lacat);
```

```
  lcd.clear();
```

```
  lcd.print("Buzdugan
Andrei");
}
```

```
void loop()
{
  lcd.setCursor(0, 1);
  lcd.write(0);

  lcd.setCursor(2, 1);
  lcd.write(1);

  lcd.setCursor(4, 1);
  lcd.write(2);

  lcd.setCursor(6, 1);
  lcd.write(3);

  lcd.setCursor(8, 1);
  lcd.write(4);

  lcd.setCursor(10, 1);
  lcd.write(5);

  lcd.setCursor(12, 1);
  lcd.write(6);

  lcd.setCursor(14, 1);
  lcd.write(7);
}
```



APLICAȚIA 4

Realizarea și programarea unui circuit utilizând componente la alegere în TINKERCAD.

