



Universitatea POLITEHNICA din București  
Facultatea de Inginerie Industrială și Robotică

---



# ROBOTICĂ 1 – LABORATOR 7

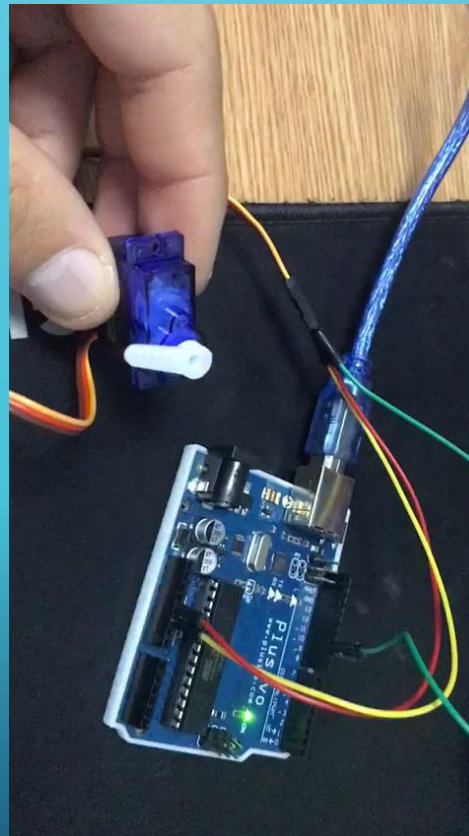
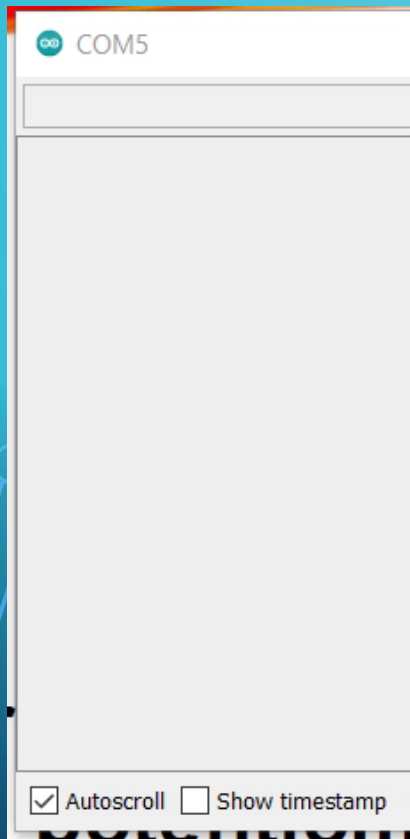
**Student:** BUZDUGAN Andrei

**Coordonator:** Ileana DUGĂEȘESCU

# APLICAȚIA 1

## Asamblarea și programarea unui circuit utilizând un servomotor. Afișarea valorilor obținute de la servomotor pe ecran.

Aici am conectat un servomotor la placuta arduino, cu 5V si GND la pinii aferenti de pe placa si pinul care transmite date, l-am conectat la un pin digital care are simbolul „~”. Iar prin cod, am afișat valorilor obținute de la servomotor pe ecran.



### CODUL

```
#include <Servo.h>
Servo servomotor;
int pin_servomotor = 10;
int pozitie = 0;

void setup() {
  servomotor.attach(pin_servomotor);
  servomotor.write(pozitie);
  Serial.begin(9600);
  delay(10);
}

void loop() {
  pozitie = analogRead(pozitie);
  Serial.print("\n");
  Serial.print(pozitie);

  servomotor.write(0);
  delay(1000);

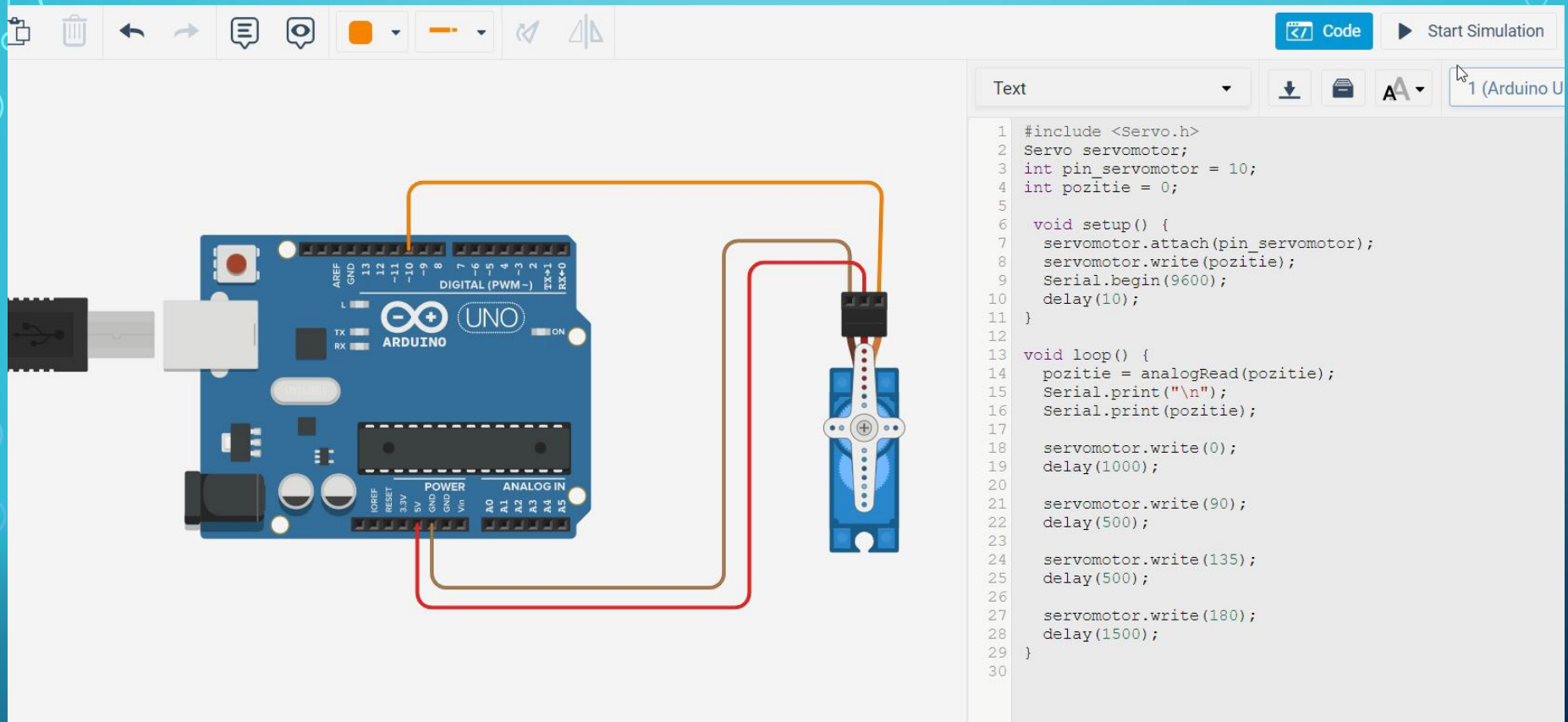
  servomotor.write(90);
  delay(500);

  servomotor.write(135);
  delay(500);

  servomotor.write(180);
  delay(1500);
}
```

# APLICAȚIA 1

Asamblarea și programarea unui circuit utilizând un servomotor. Afișarea valorilor obținute de la servomotor pe ecran în TINKERCAD.



# APLICAȚIA 2

**Asamblarea și programarea unui circuit utilizând un servomotor. Afișarea valorilor citite de la servomotor pe ecran.**

Aici am conectat un LCD cu I2C la placuta Arduino si am afișat valorilor citite de la servomotor pe ecranul LCD-ului.



## CODUL

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

Servo servomotor;
LiquidCrystal_I2C lcd(0x27,16, 2);

void setup(){
  servomotor.attach(10);
  lcd.setBacklight(1);
  lcd.init();
  lcd.backlight();
  lcd.clear();
}

void loop(){
  for(int i = 0; i < 180; i++){
    lcd.println(i);
    servomotor.write(i);
    delay(25);
  }

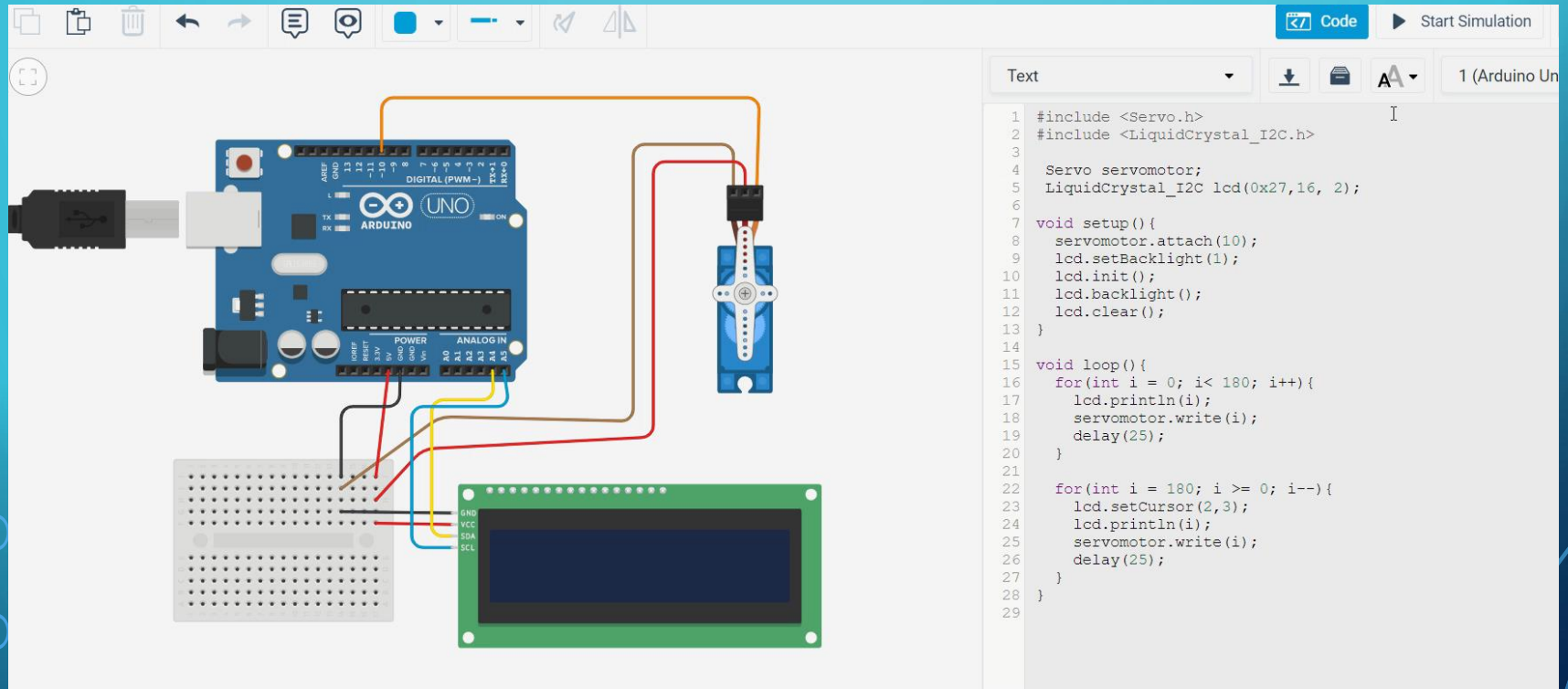
  for(int i = 180; i >= 0; i--){
    lcd.setCursor(2,3);
    lcd.println(i);
    servomotor.write(i);
    delay(25);
  }
}
```

# APLICAȚIA 2

Asamblarea și programarea unui circuit utilizând un servomotor. Afișarea valorilor citite de la servomotor pe ecran în TINKERCAD.

## Metoda 1: NU MERGE TINKERCAD-UL !

OBS: Acest lucru l-am observat și la laboratorul 6, abia am reușit să folosesc I2C-ul.



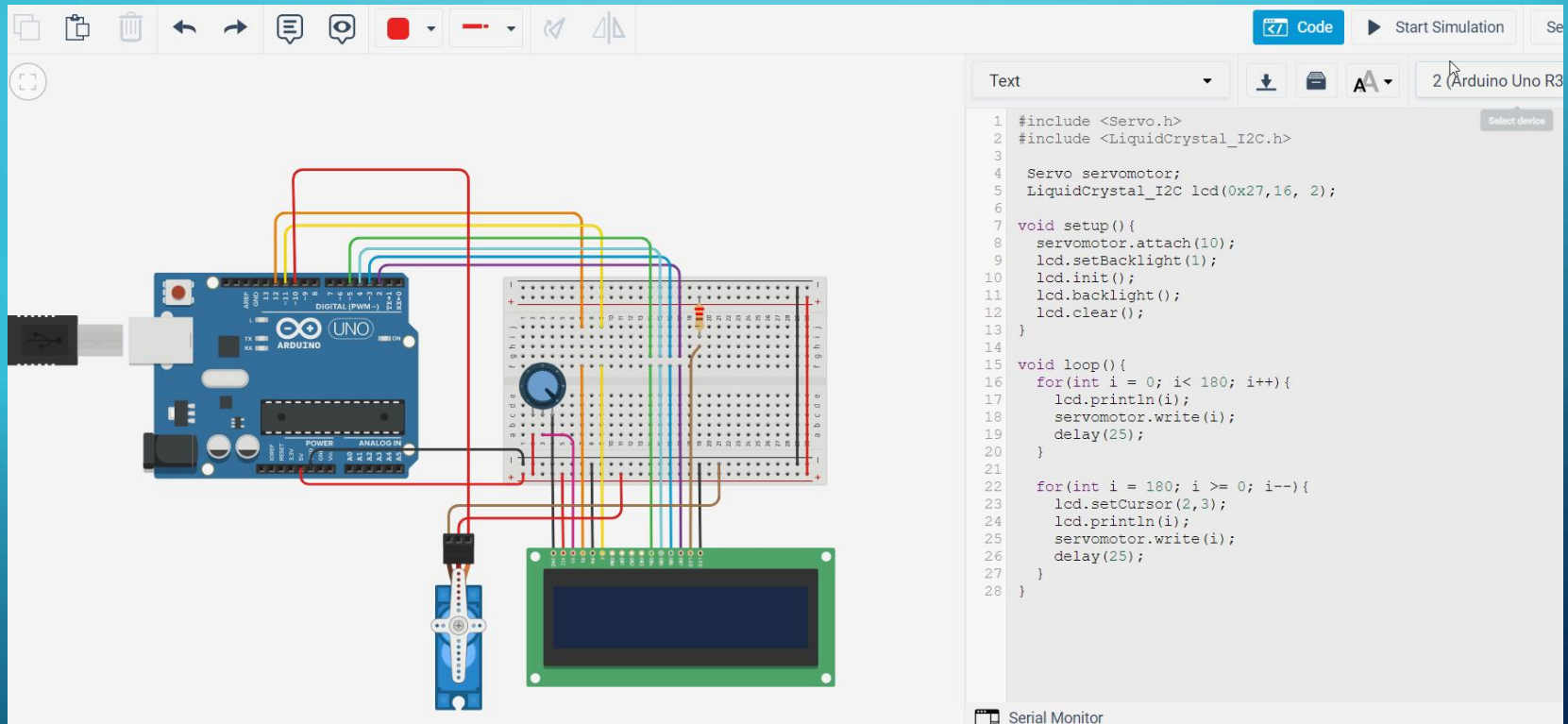


# APLICAȚIA 2

Asamblarea și programarea unui circuit utilizând un servomotor. Afișarea valorilor citite de la servomotor pe ecran în TINKERCAD.

## Metoda 2: TOT NU MERGE TINKERCAD-UL !

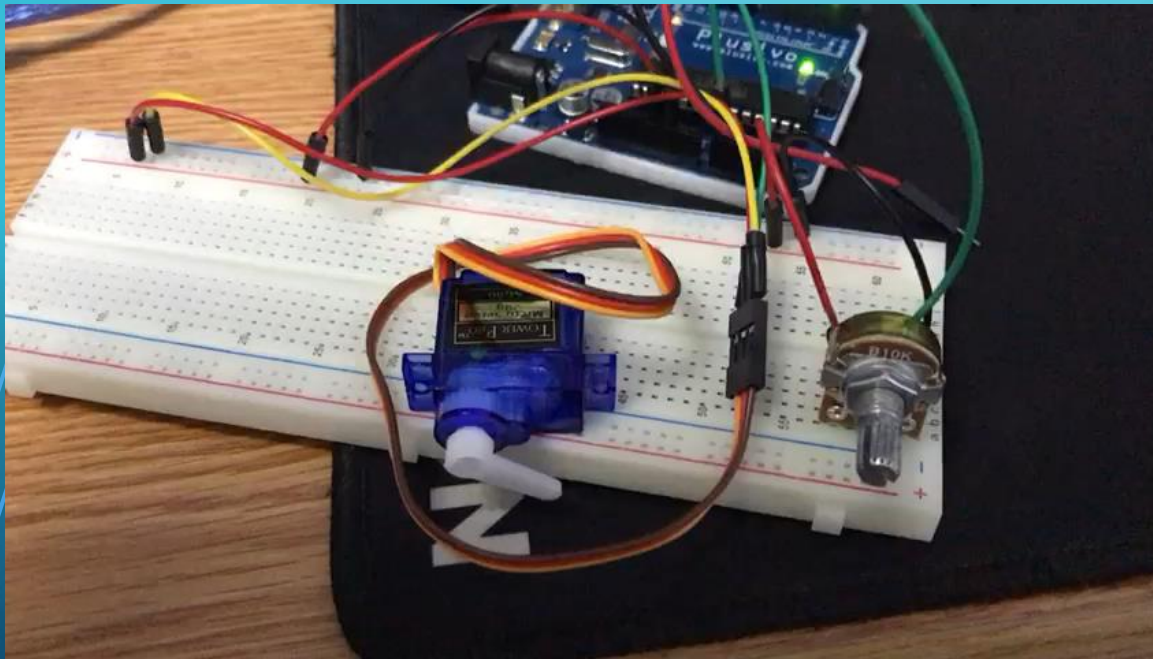
OBS: Am incercat si varianta fara I2C si tot nu merge.



# APLICAȚIA 3

## Asamblarea și programarea unui circuit utilizând un servomotor și potențiometrul.

Aici am luat conexiunile de la aplicația 1 și am mai adăugat un potențiometrul pentru a controla gradele de rotație ale servomotorului cu ajutorul potențiometrului.



### CODUL

```
#include <Servo.h>
```

```
Servo myservo;
```

```
int potpin = A0;
```

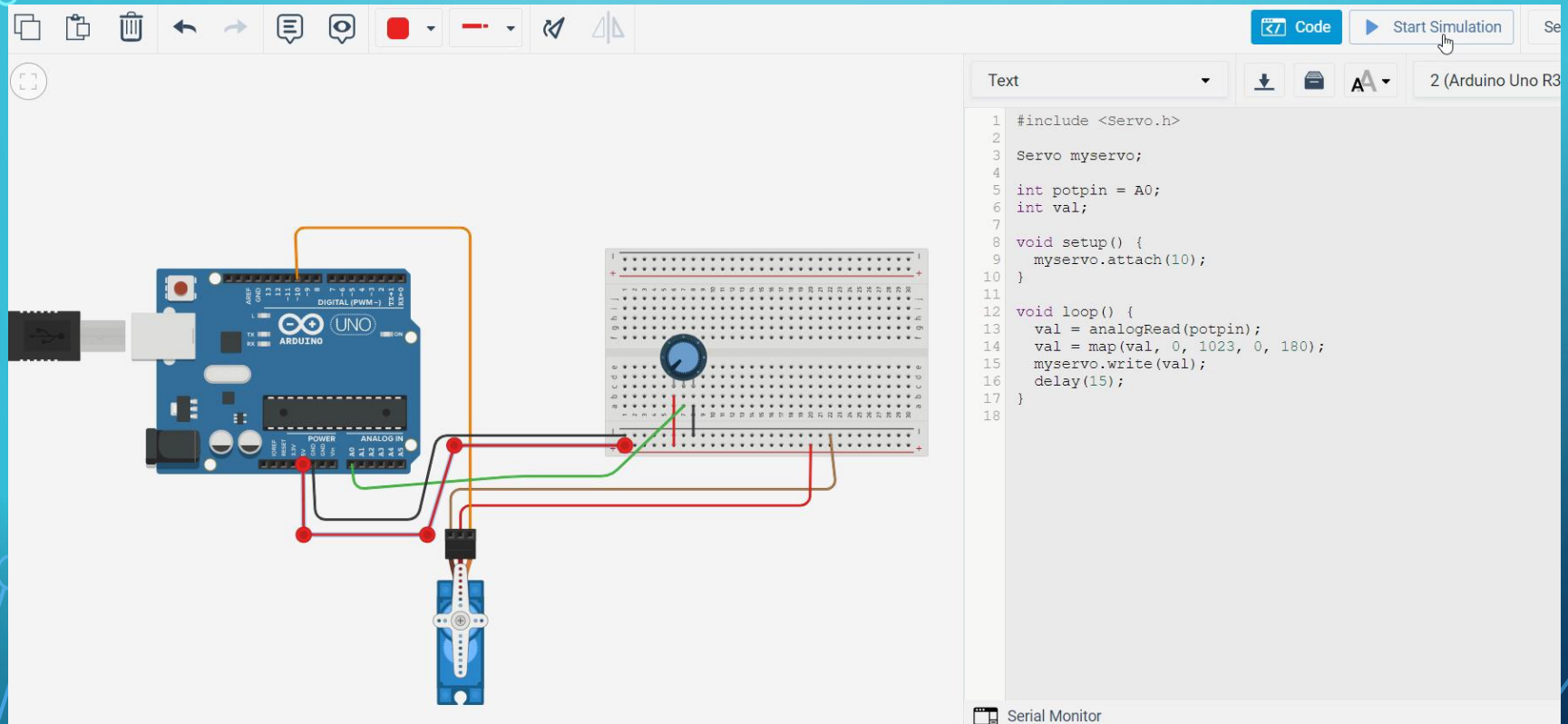
```
int val;
```

```
void setup() {  
  myservo.attach(10);  
}
```

```
void loop() {  
  val = analogRead(potpin);  
  val = map(val, 0, 1023, 0, 180);  
  myservo.write(val);  
  delay(15);  
}
```

# APLICAȚIA 3

Asamblarea și programarea unui circuit utilizând un servomotor și potențiometru.  
în TINKERCAD.

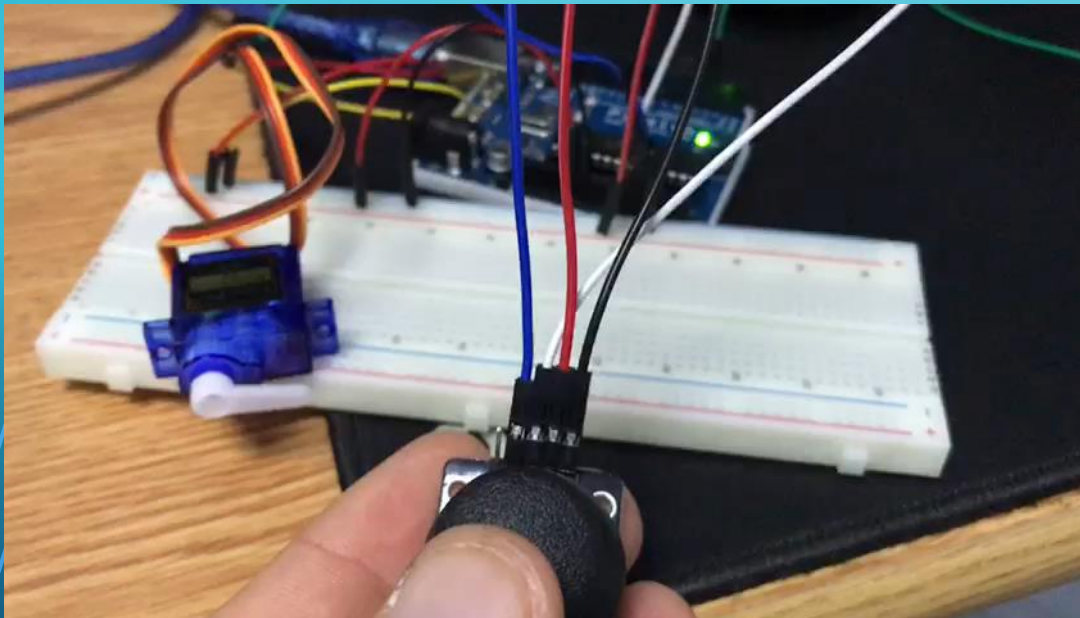




# APLICAȚIA 4

## Realizarea și programarea unui circuit utilizând componente la alegere.

Iar la ultima aplicatie am realizat controlul gradelor de rotatie ale servomotorasului cu ajutorul unui modul Joystick.



### CODUL

```
#include <Servo.h>
Servo servo1;
Servo servo2;
int joyX =A0;
int joyY =A1;
int joyVal;

void setup()
{
  servo1.attach(3);
  servo2.attach(5);
}

void loop()
{
  joyVal = analogRead(joyX);
  joyVal = map (joyVal, 0, 1023, 0, 180);
  servo1.write(joyVal);

  joyVal = analogRead(joyY);
  joyVal = map (joyVal, 0, 1023, 0, 180);
  servo2.write(joyVal);
  delay(15);
}
```

