

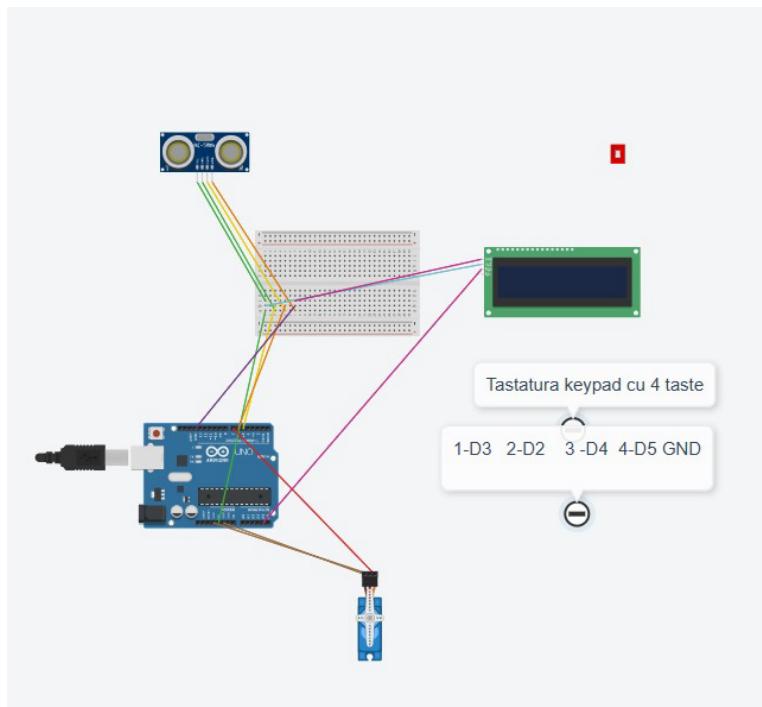
# Automat bancar

Proiectare cu microprocesoare

Calculatoare, grupa 30236

Proiect realizat de Buzila Maria Alexandra si Bacisor Miruna Roxana

Acest proiect simulează un ATM (Automated Teller Machine) folosind o placă Arduino Uno și diverse componente electronice. Scopul principal este de a crea un sistem interactiv care să permită utilizatorului să introducă un card, să valideze un PIN și să retragă numerar, monede în acest caz (simulat prin mișcarea unui servomotor). Proiectul utilizează un senzor ultrasonic pentru detectarea introducerii și scoaterii cardului, un LCD pentru afișarea mesajelor și un keypad pentru introducerea PIN-ului și a numărului de retragere.



Această implementare poate fi utilizată pentru a înțelege concepte precum interacțiunea cu senzori, afișajul pe LCD, utilizarea servomotoarelor și programarea secvențială. Proiectul oferă o experiență realistă, imitând funcționalitatea unui ATM de bază.



### **Componente utilizate**

- Placa Arduino Uno R3
- Breadboard
- LCD cu protocol I2C
- Ultrasonic Sensor
- Servomotor
- O baterie 9V
- Un switch legat la baterie
- Keypad cu 4 butoane

**Legenda legarea componentelor:**

LCD:

Vcc - 5V

GND - GND

SDA - A4

SCL - A5

KEYPAD 4 taste:

Tasta 1: Pin 2

Tasta 2: Pin 3

Tasta 3: Pin 5

Tasta 4: Pin 4

SENZOR:

Vcc - 5V

Trig - D5

Echo - D6

GND - GND

MOTORAS:

galben - D7

rosu - 3.3V

maro - GND

BB:

1: Vcc

3: D5

5: D6

7:GND

## **Tutorial de functionare:**

### **Pornirea sistemului**

- La pornire, pe ecranul LCD apare mesajul "**Bine ai venit!**" pentru câteva secunde, după care este afișat mesajul "**Introduceți cardul!**".
- Sistemul așteaptă introducerea cardului (simulată de un obiect plasat în fața senzorului ultrasonic).

### **Introducerea cardului**

- Când cardul este detectat de senzorul ultrasonic (distanță < 5 cm), mesajul de pe LCD se schimbă în "**Introduceti PIN**".

### **Introducerea PIN-ului**

- Utilizatorul introduce codul PIN (format din 4 cifre) folosind keypad-ul - "1234"
- Pentru fiecare cifra apăsată pe Keypad se afișează pe lcd o steluță pentru a simula și securitatea unui ATM real.

### **Validarea PIN-ului**

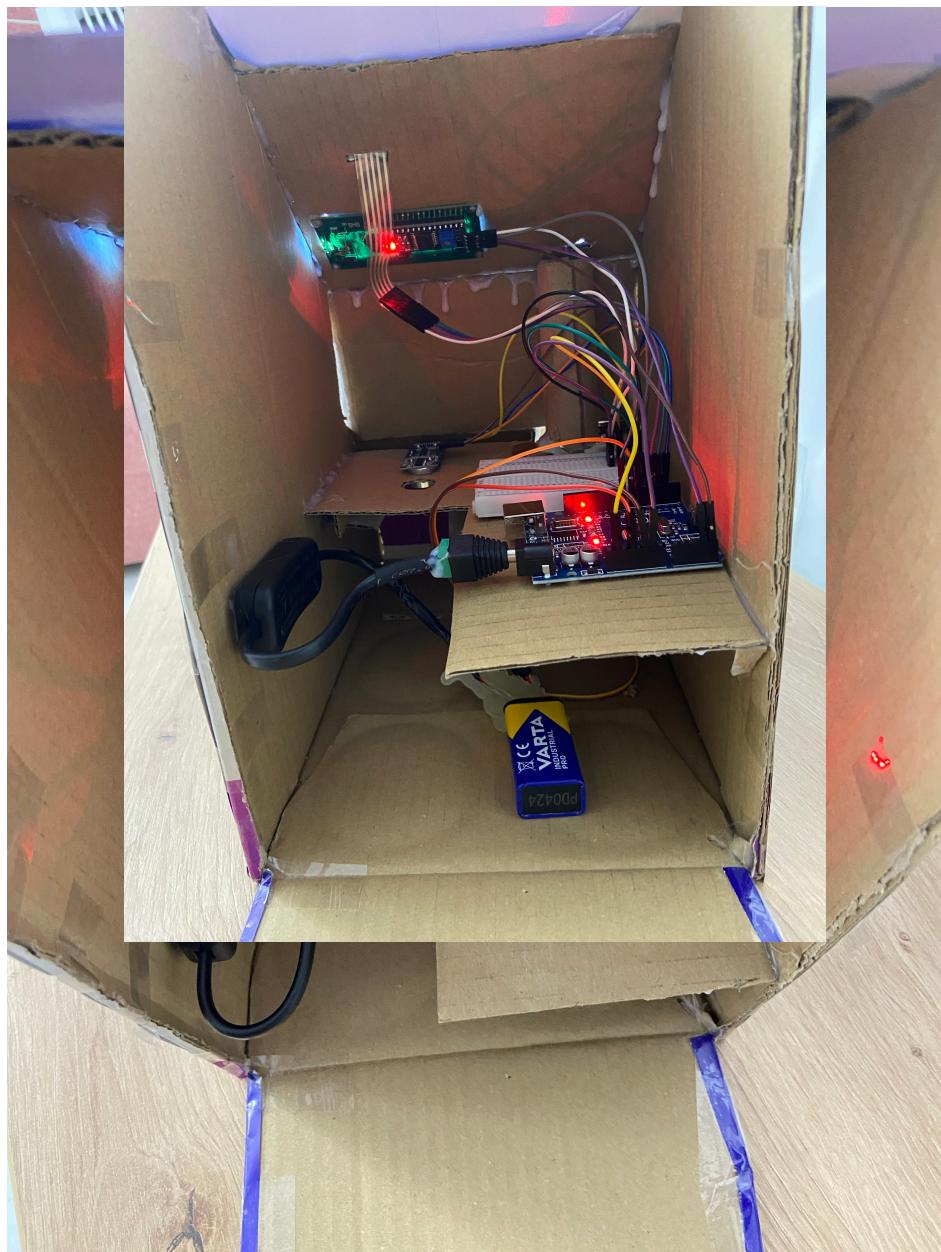
- Dacă PIN-ul introdus este corect (de exemplu, "1234"), LCD-ul afișează mesajul "**Pin correct! Retragere numerar...**".
- Servomotorul este activat pentru a simula retragerea banilor (rotirea servo-ului pentru a elibera monede).
- După finalizarea retragerii, pe LCD apare mesajul "**Retragere cu succes**".

### **PIN incorrect**

- Dacă PIN-ul introdus este greșit, LCD-ul afișează mesajul "**PIN gresit! Încercați din nou.**" și procesul reîncepe, solicitând introducerea cardului.

### **Scoaterea cardului**

- După retragerea banilor, sistemul așteaptă ca utilizatorul să scoată cardul (distanță măsurată de senzor > 5 cm) și afișează mesajul "**Scoateți cardul**"
- Dupa acest pas procesul se reia

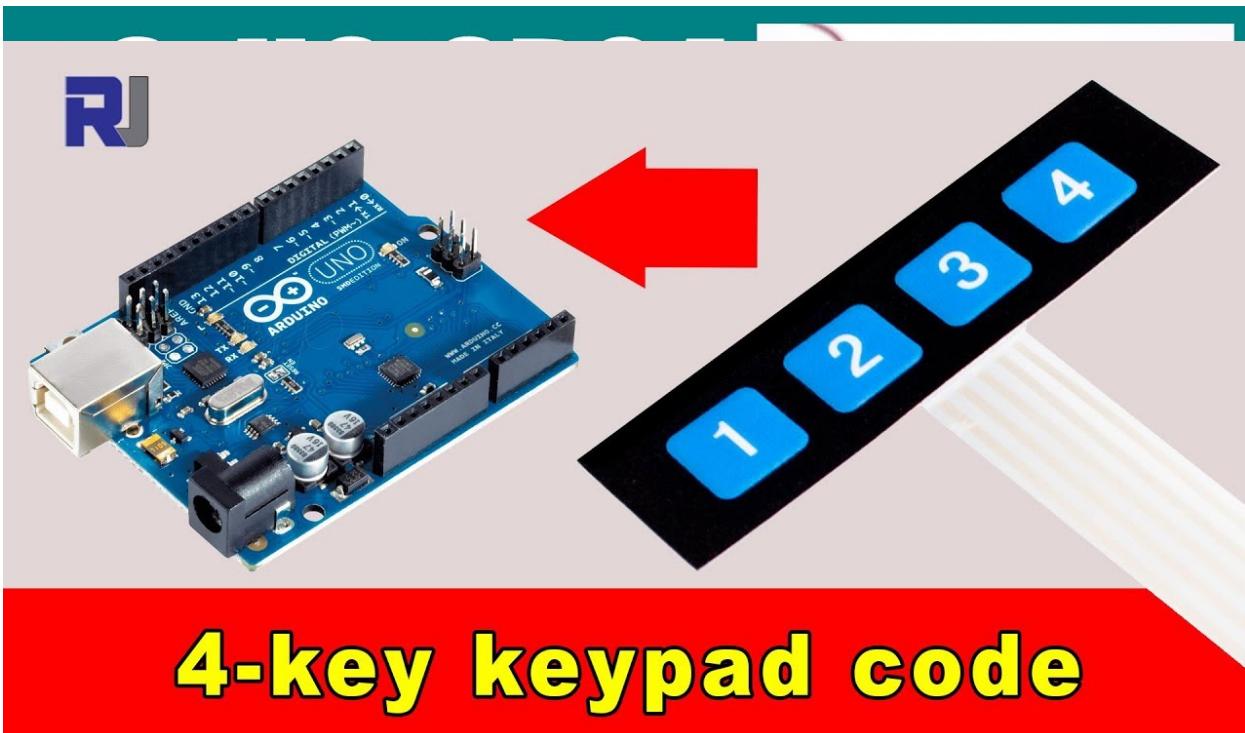


**Demonstratie practica:**



**Bibliografie:**

1.



## 4-key keypad code

2.

3. <https://www.hackster.io/Josdebum/atm-machine-prototype-0b5c60>
4. <https://docs.arduino.cc/learn/electronics/servo-motors/>

### Anexa:

Codul implementat:

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

int trigPin = 8;
int echoPin = 6;
int servoPin = 7;
long duration, dist;
Servo servo;

LiquidCrystal_I2C lcd(0x27, 16, 2); // Adresa LCD-ului (de obicei 0x27)

// Definirea pini pentru keypad
#define key1 2 //Pin 2 pentru tasta 1
```

```

#define key2 3
#define key3 5
#define key4 4

String correctPIN = "1234";
String enteredPIN = "";
String enteredSuma = "";
int suma = 0;

bool prevStateKey1 = true; // Starea anterioara a tastei 1
bool prevStateKey2 = true;
bool prevStateKey3 = true;
bool prevStateKey4 = true;

long average;
long aver[3];

void setup() {
    Serial.begin(9600);

    pinMode(key1, INPUT_PULLUP);
    pinMode(key2, INPUT_PULLUP);
    pinMode(key3, INPUT_PULLUP);
    pinMode(key4, INPUT_PULLUP);

    servo.attach(servoPin); // Atasează servo-ul la pinul corespunzător

    pinMode(trigPin, OUTPUT); //setam pinii pentru senzor ultrasonic
    pinMode(echoPin, INPUT);

    lcd.init();
    lcd.backlight(); // activează iluminarea de fundal pentru lcd
    lcd.setCursor(0,0);
    lcd.print("Bine ai venit!");
    delay(2000);
    lcd.clear();
}

void loop() {
    lcd.clear();
    lcd.print("Introduceti card: ");

    //asteptam introducerea cardului

    dist = 100;
    while (dist > 5) {
        measure();
    }
}

```

```

//facem o medie aritmetica intre 3 masurari
//conditia de siguranta ca a fost introdus cardul
for(int i = 0; i <= 2; i++) {
    measure();
    aver[i] = dist;
    delay(10);
}
dist = (aver[0] + aver[1] + aver[2]) / 3;

if (dist<5){
    //cardul a fost introdus
    lcd.clear();
    lcd.print("Introduceti PIN:");

    // citire PIN
    enteredPIN = "";
    while (enteredPIN.length() < 4){
        if (readKey(key1, prevStateKey1)) {
            enteredPIN += "1";
            lcd.setCursor(0, 1);
            lcd.print("*");
        }
        if (readKey(key2, prevStateKey2)) {
            enteredPIN += "2";
            lcd.setCursor(0, 1);
            lcd.print("*");
        }
        if (readKey(key3, prevStateKey3)) {
            enteredPIN += "3";
            lcd.setCursor(0, 1);
            lcd.print("*");
        }
        if (readKey(key4, prevStateKey4)) {
            enteredPIN += "4";
            lcd.setCursor(0, 1);
            lcd.print("*");
        }
    }

    // verificare PIN
    if(enteredPIN == correctPIN) {
        lcd.clear();
        lcd.print("PIN corect!");
        lcd.setCursor(0, 1);
        lcd.print("Retragere...");
        delay(2000);
    }
}

//citire suma

```

```

enteredSuma = "";

lcd.clear();
lcd.print("Suma dorita: ");

while (enteredSuma.length() < 1) {
    if (readKey(key1, prevStateKey1)) {
        enteredSuma += "1";
        lcd.setCursor(0, 1);
        lcd.print(enteredSuma);
    }
    if (readKey(key2, prevStateKey2)) {
        enteredSuma += "2";
        lcd.setCursor(0, 1);
        lcd.print(enteredSuma);
    }
    if (readKey(key3, prevStateKey3)) {
        enteredSuma += "3";
        lcd.setCursor(0, 1);
        lcd.print(enteredSuma);
    }
    if (readKey(key4, prevStateKey4)) {
        enteredSuma += "4";
        lcd.setCursor(0, 1);
        lcd.print(enteredSuma);
    }
}

suma = enteredSuma.toInt();

//numar miscari servo = suma introdusa
for (int i = 0; i < suma; i++) {
    servo.write(110);
    delay(1000);
    servo.write(0);
    delay(1000);
}

servo.write(110);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Retragere finalizata");
lcd.setCursor(0, 1);
lcd.print("cu succes!");
delay(4000);

//asteapta scoaterea cardului
lcd.clear();
lcd.print("Scoateti card...");
```

```

dist = 100;

}else {
    lcd.clear();
    lcd.print("PIN gresit!");
    lcd.setCursor(0, 1);
    lcd.print("Incercati iar: ");
    delay(3000);
}

}else{
    //daca card nu a fost introdus, adica distanta > 5
    lcd.clear();
    lcd.print("Introduceti card: ");
}

delay(2000);
}

//functia de masurare a distantei
void measure() {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(5);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(15);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    dist = (duration / 2) / 29.1; //obtinere distanta
}

//citire a starii unui buton cu debouncing si eliminarea citirilor false
bool readKey(int keyPin, bool &prevState) {
    bool currentState = digitalRead(keyPin);

    // detecteaza doar trecerea de la HIGH la LOW (buton apasat)
    if (!currentState && prevState) {
        delay(100); //debouncing
        currentState = digitalRead(keyPin);
        if (!currentState) { // verifica daca butonul ramane apasat
            prevState = false;
            return true;
        }
    }

    // actualizeaza starea pentru trecerea de la LOW la HIGH (buton eliberat)
    if (currentState && !prevState) {

```

```
prevState = true;  
}  
  
return false; //nicio apasare  
}
```