

Fancy Door Lock

Theodor Protopopescu, 332CB

Introducere

Acesta este un sistem de deblocare a unei incuietori pe baza unui PIN tastat pe un keypad, cu afisaj pe un LCD. Parola va putea fi introdusa si modificata din keypad; usa se va putea deschide si din telefon, care va avea o conexiune Bluetooth cu sistemul.

Descriere generală

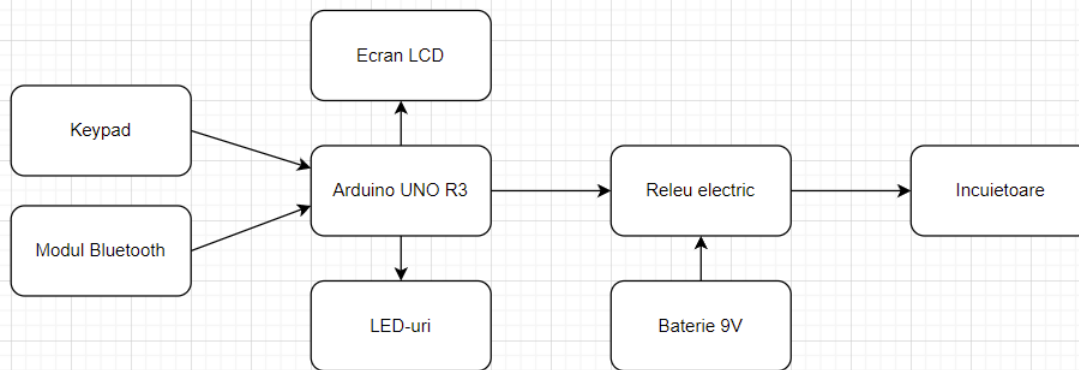
Deschiderea usii se realizeaza prin introducerea parolei, urmata de caracterul "#".

Daca parola introdusa este corecta, se va afisa pe ecranul LCD un mesaj corespunzator, iar incuietoarea se va deschide pentru 3 secunde. In cazul in care parola este gresita, se va afisa un mesaj corespunzator timp de 3 secunde, dupa care parola va putea fi reintrodusa.

La introducerea codului "####", se va activa optiunea de resetare a parolei. Se va solicita vechea parola a incuietorii inainte de a putea fi setata o noua parola.

In plus, deblocarea usii se va putea face si din telefon, prin trimiterea parolei prin aplicatie, dupa conectarea la modulul Bluetooth.

Schema bloc

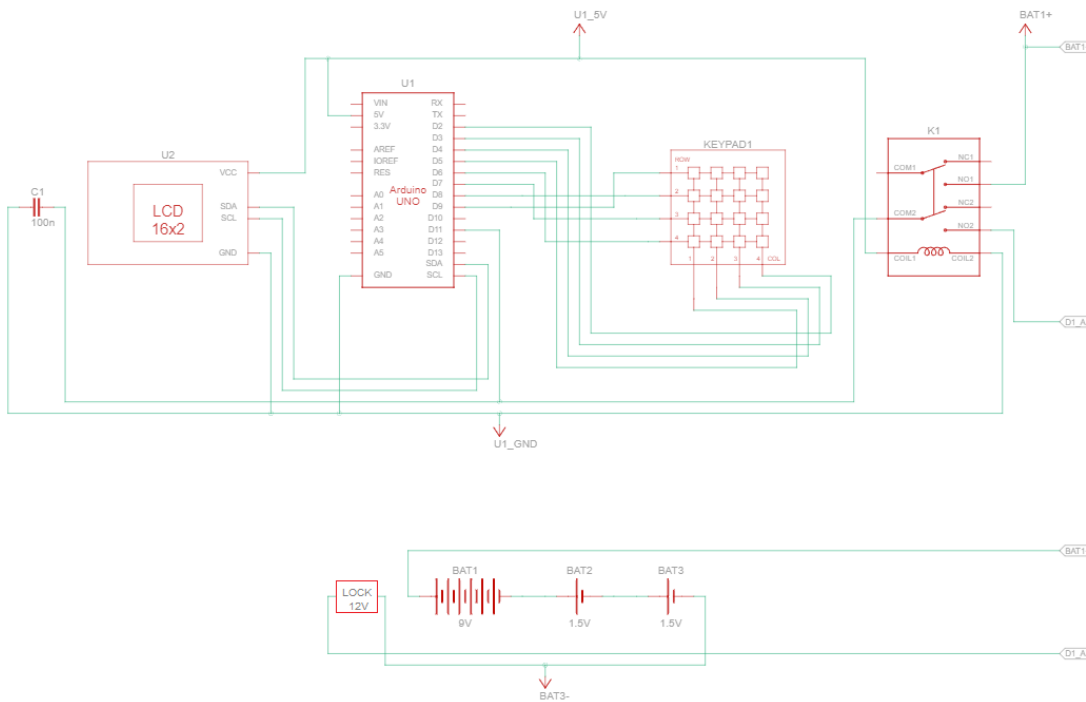


Hardware Design

Lista de piese

- Arduino UNO R3
- Breadboard
- Keypad 4×4
- Ecran LCD
- O baterie 9V si doua baterii de 1,5V
- Releu electric 5V
- Incuietoare electrica 12V
- Modul Bluetooth HC-02
- Fire
- Condensator 10uF

Schema electrica



Software Design

Mediu de dezvoltare

Proiectul a fost programat exclusiv in ArduinoIDE.

Biblioteci

Am folosit 3 biblioteci auxiliare, pentru utilizarea LCD-ului cu modul I2C si pentru utilizarea Keypad-ului.

- #include <LiquidCrystal_I2C.h>
- #include <Keypad.h>
- #include <Wire.h>

Organizarea codului

- Definirea bibliotecilor, valorilor initiale pentru parola si codul de resetare, a pinilor si variabilelor auxiliare folosite.

```

#include <LiquidCrystal_I2C.h>
#include <Keypad.h>
#include <Wire.h>
#define LENGTH 5
#define LOCK 11

char data[LENGTH];
char password[LENGTH] = "1234";
char resetCode[LENGTH] = "####";
byte k = 0;
byte z = 1;
const byte column = 4;
const byte rows = 4;
char confirmButton;
int openCommand;
char keys[rows][column]= {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[rows] = {9,8,7,6};
byte colPins[column] = {5,4,3,2};
Keypad myKeypad=Keypad(makeKeymap(keys), rowPins, colPins, rows, column);
LiquidCrystal_I2C lcd(0x27, 16, 2);

```

- Functia setup pentru pornirea LCD-ului si a serialei care va comunica cu modulul Bluetooth.

```

void setup() {
  pinMode(LOCK, OUTPUT);
  digitalWrite(LOCK, LOW);

  lcd.init();
  lcd.backlight();
  Serial.begin(9600);
}

```

- Functia loop, care contine cazurile principale ale programului. Mai intai, determina daca parola va fi introdusa de la tastatura sau prin modulul Bluetooth. In functie de aceasta decizie, va efectua citirea parolei introduse prin modul corespunzator. In cazul in care parola introdusa este codul de resetare, se solicita introducerea parolei vechi, iar apoi setarea parolei noi. In cazul in care se introduce o parola corecta, incuietoarea se deschide.

```

void loop() {
  lcd.setCursor(0,0);
  lcd.print("Enter pass:");

  int t = 1;
  // Se verifica daca se transmite ceva pe seriala
  if (Serial.available() > 0) {
    t = 0;

```

```
}

// Se citeste parola introdusa de la tastatura, sau pe seriala
while(k < 4) {
  if (!t) {
    data[k] = Serial.read();
  } else {
    data[k] = myKeypad.waitForKey();
  }
  if(data[k]) {
    lcd.setCursor(k,1);
    lcd.print(data[k]);
    delay(500);
    k++;
  }
}

z = 1;
while(z) {
  // Se asteapta tasta '#' pentru confirmarea parolei
  if (t) {
    confirmButton = myKeypad.waitForKey();
  } else {
    confirmButton = Serial.read();
  }
  if(confirmButton == '#') {
    z = 0;
    // Daca parola este corecta
    if(strncmp(data, password, 4) == 0) {
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Correct pass");
      digitalWrite(LOCK, HIGH);
      delay(3000);
      digitalWrite(LOCK, LOW);

      // Daca s-a introdus codul de resetare
    } else if(strncmp(data, resetCode, 4) == 0){
      clearData();
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Enter old pass:");
      while (k < 4) {
        data[k] = myKeypad.waitForKey();
        if(data[k]) {
          lcd.setCursor(k,1);
          lcd.print(data[k]);
          delay(500);
          k++;
        }
      }
    }
  }
}
```

```
// Daca parola veche introdusa este corecta
if (strcmp(data, password, 4) == 0) {
    clearPassword();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Enter new pass:");
    while(k < 4) {
        password[k] = myKeypad.waitForKey();
        if(password[k]) {
            lcd.setCursor(k,1);
            lcd.print(password[k]);
            delay(500);
            k++;
        }
    }

// Daca parola veche introdusa este gresita
} else {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Wrong old pass:");
    delay(1000);
}

// Daca parola initiala introdusa este gresita
} else {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Incorrect pass");
    delay(3000);
}

// Daca se introduce din greseala alta tasta in afara de #
} else {
    lcd.clear();
    lcd.print("Press #");
    delay(1000);
}
clearData();
}

lcd.clear();
}
```

- Functii auxiliare, clearData (elibereaza vectorul in care se salveaza parola introdusa) si clearPassword (elibereaza vectorul in care este salvata parola sistemului, pentru a permite resetarea acesteia).

```
void clearData() {
    while (k != 0) {
```

```
    data[k] = 0;
    k = k - 1;
}
return;
}

void clearPassword() {
    while (k != 0) {
        password[k] = 0;
        k = k - 1;
    }
    return;
}
```

Rezultate Obținute

În urma proiectului, am obținut o macheta cu tastatura și afișajul LCD, cu o mică ușă ce se va putea deschide la deblocarea încuietorii, care funcționează conform ciclului descris mai sus.

Concluzii

A fost un proiect interesant, aplicat, în care am simțit că am pus la folosință conceptele și tehnicile dobândite în cadrul laboratoarelor, dar am avut parte și de experiențe noi, utilizând anumite module unice.

Download

[332cb_protopopescu_theodor_fancy_door_lock.zip](#)

Bibliografie/Resurse

1. <https://www.arduino.cc/reference/en/libraries/keypad/>
2. <https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>
3. <https://howtomechatronics.com/tutorials/arduino/arduino-and-hc-05-bluetooth-module-tutorial/>

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/cristip/fancydoorlock>



Last update: **2022/06/01 19:48**