

[smart_lock.pdf](#)

Smart Lock

Nume: Fasui Violeta Alexandra

Grupa: 334CA

Introducere

Tema proiectului propus este o incuietoare inteligenta. Acesta va simula modul de functionare al unei alarme, tinand cont de starea incuietorii. Astfel vor exista doua situatii:

- Daca incuietoarea este inchisa si se simte miscare, atunci alarma va semnala acest lucru, folosind un buzzer.
- Daca incuietoarea este deschisa, nu se tine cont de datele obtinute de la senzorul de miscare.

Pentru a putea inchide si deschide incuietoarea, voi utiliza o parola ce presupune introducerea unui cod prin keypad. In permanenta, incuietoarea este conectata la un telefon ce ii poate modifica starea.

Descriere generala

In scopul realizarii acestei incuietori voi utiliza un senzor de miscare, un servomotor, un keypad, un buzzer si modelul bluetooth. Incuietoarea va avea doua stari determinate de codul introdus prin keypad:

- inchisa: fie atunci cand se introduce #, fie atunci cand se introduce o parola gresita;
- deschisa: atunci cand parola introdusa este corecta (1234);

Atunci cand incuietoarea este inchisa, vom monitoriza datele obtinute de senzorul pir si vom semnala prezenta unei persoane prin buzzer. De pe telefon vom putea inchide sau inchide incuietoarea.

Hardware Design

Modul Bluetooth

Modul PIR - senzor de prezenta

Servomotor

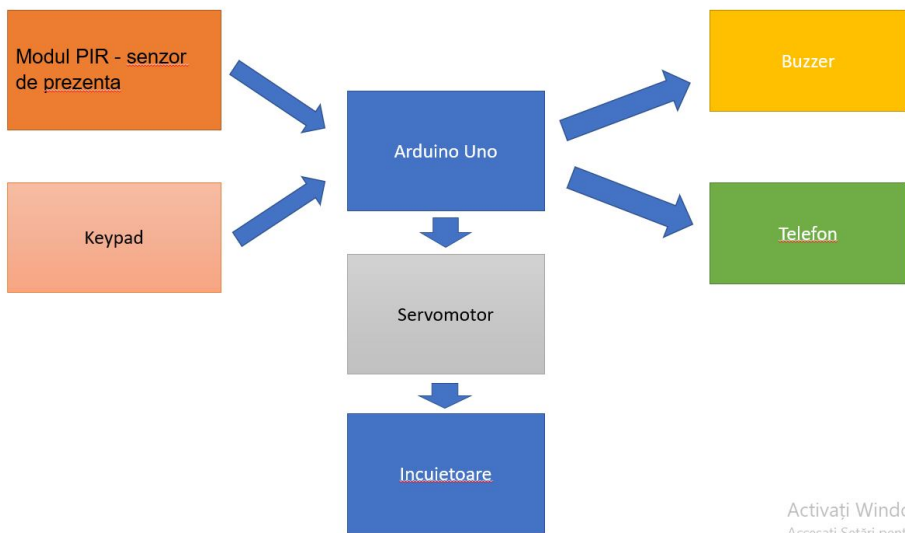
Keypad

Buzzer

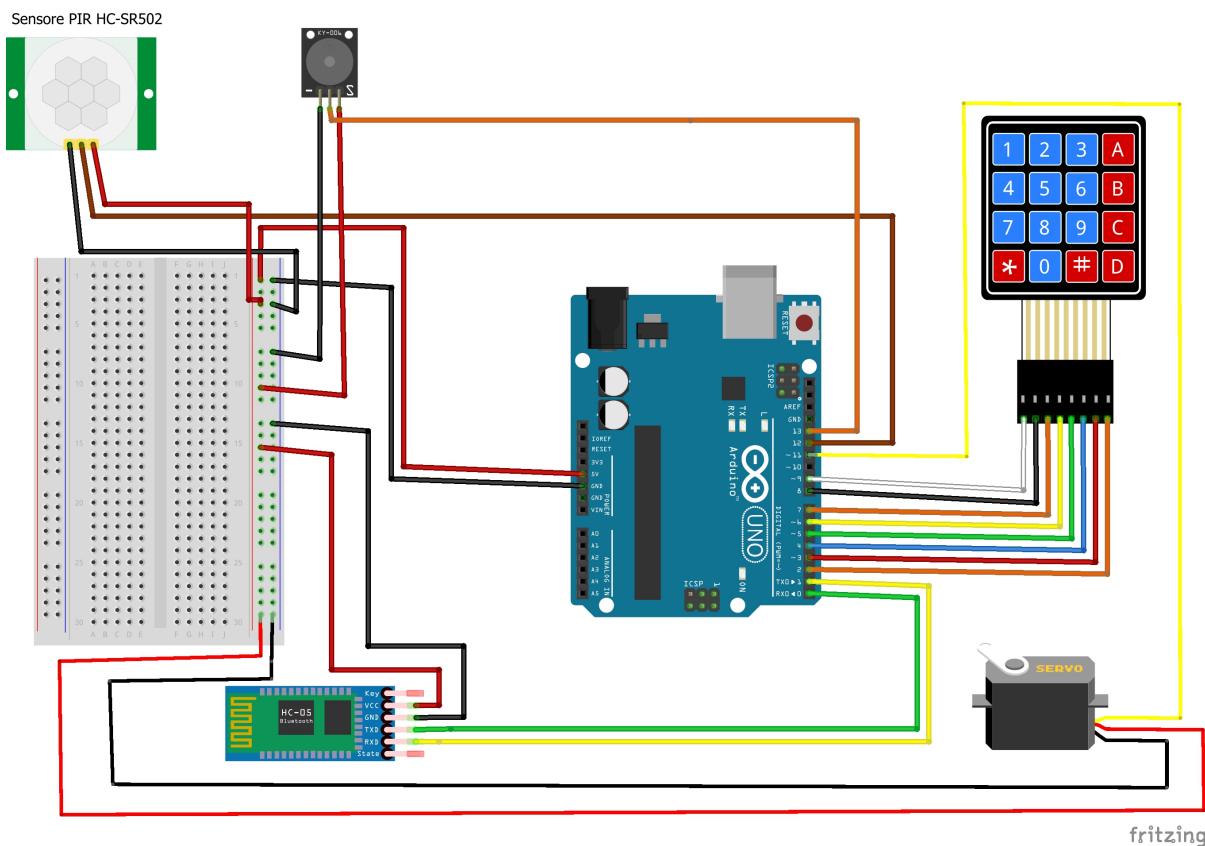
Breadboard

Arduino UNO

Fire



Activati Windows
Accesati Setari pentru a activa \



Software Design

In scopul realizarii proiectului, am utilizat o variabila care monitorizeaza starea incuietorii. Aceasta isi schimba starea astfel:

- este inchisa atunci cand se apasa tasta '#' sau atunci cand parola introdusa este gresita.
- este deschisa atunci cand combinatia de caractere retinuta in variabila password este aceeași cu cea setata initial ca fiind codul de deschidere ("1234").

Pe masura ce tastele sunt apasate, codul este in permanenta retinut intr-un string, iar in momentul in care se introduce caracterul special "*", codul este verificat in functia checkPassword(). Se face compararea a doua stringuri, pentru a putea determina atat starea incuietorii, cat si pe cea a buzzer-ului. Dupa fiecare verificare parola este resetata. Buzzerul verifica atat starea incuietorii, cat si

datele obtinute de senzorul pir. Daca incuietoarea este inchisa iar valoarea senzorului pir este 1, atunci alarma suna. In loop, se verifica constant datele primite de la telefon. Acesta poate trimite caracterele 'c' pentru a inchide incuietoarea, sau 'o' pentru a o deschide.

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

#define delay_time 100

Servo servol;
String pass = "1234";
int lock = 0;

String password;
const byte Rows = 4;
const byte Cols = 4;

int pir = 12;
int move = 0;
char phone_data;
int buzzer = 13;

char keys[Rows][Cols] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

byte rowPins[Rows] = {9, 8, 7, 6};
byte colPins[Cols] = {5, 4, 3, 2};

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, Rows, Cols);

void setup() {
  Serial.begin(9600);
  servol.attach(11);
  keypad.addEventListener(keypadEvent);
  pinMode(pir, INPUT);
  Serial.println("==== Incuietoare deschisa =====");
  servol.write(180);
  pinMode (buzzer, OUTPUT) ;
}

void loop() {
  keypad.getKey();
  move = digitalRead(pir);
  Serial.println("Senzor PIR:");
  Serial.println(move);
  delay(delay_time);
}
```

```
Serial.println(keypad.getKey());
if(lock == 1 && move == 1)
{
  Serial.println("==== SUNA ALARMA ====");
  for (int i = 0; i <10; i++)
  {
    digitalWrite (buzzer, HIGH) ;
    delay (2) ;
    digitalWrite (buzzer, LOW) ;
    delay (2) ;
  }
}
phone_data = Serial.read();
if(phone_data == 'c')
{
  lock = 1;
  servo1.write(0);
} else if(phone_data == 'o')
{
  lock = 0;
  servo1.write(180);
}
}

void keypadEvent(KeypadEvent ekey) {
  switch(keypad.getState())
  {
    case PRESSED:
      switch (ekey)
      {
        case '*': checkPassword(); break;
        case '#' : password = ""; lock = 1; servo1.write(0);
Serial.println("==== INCUIETOARE INCHISA ===="); break;
        default: password += ekey;
      }
    }
}

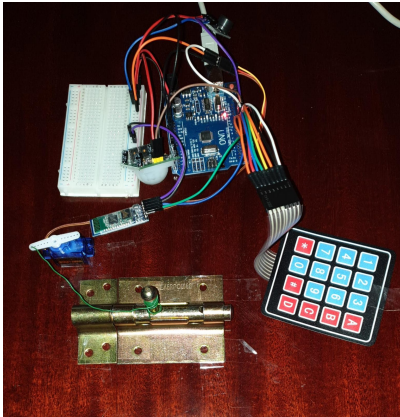
void checkPassword()
{
  Serial.println(password);
  if(password.compareTo(pass))
  {
    password = "";
    servo1.write(0);
    lock = 1;
    Serial.println("!!!! PAROLA INTRODUSA ESTE GRESITA !!!!");
  } else
  {
    servo1.write(180);
    password = "";
  }
}
```

```
lock = 0;
Serial.println("!!!! PAROLA INTRODUSA ESTE CORECTA !!!!");
}
}
```

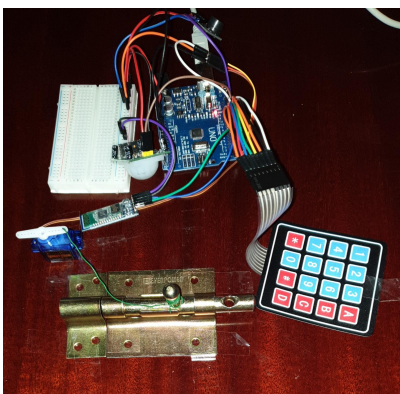
Rezultate Obținute

Proiectul simuleaza functionarea unei alarme. Acest lucru este evidentiat atat in demo, cat si in urmatoarele imagini:

- Incuietoare inchisa



- Incuietoare deschisa



<https://youtu.be/QhZYxrSIBnA>

Concluzii

Proiectul a necesitat atat acumularea cunostintelor prezentate la laborator, cat si creativitatea fiecaruia. Consider ca proiectul a fost unul practic, cu o utilitate reala, ce poate fi privit ca o experienta constructiva.

Jurnal

25 aprilie: am ales tema a proiectului

29 aprilie: am modificat ideea de inceput, am ales piesele si am creat aceasta pagina

10 mai: am comandat piesele suplimentare de care aveam nevoie

14 mai: implementare proiect

26 mai: finalizare proiect

Resurse

[smart_lock.pdf](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/cghenea/smartlock>

Last update: **2021/05/31 21:34**

