

# Smart Lock

## 1. Partea teoretica

În scenariul actual al lumii, securitatea este o preocupare majoră pentru toți, iar problema de securitate se confruntă cu fiecare persoană.[2] Sistemele tradiționale de închidere a usii care utilizează blocarea mecanică și mecanismul cu cheie sunt înlocuite de noi tehnici avansate de sistem de blocare. Aceste tehnici sunt o integrare a dispozitivelor mecanice și electronice și sunt extrem de inteligente. Una dintre caracteristicile proeminente ale acestor sisteme inovatoare de blocare este simplitatea și eficiența ridicată.[1]

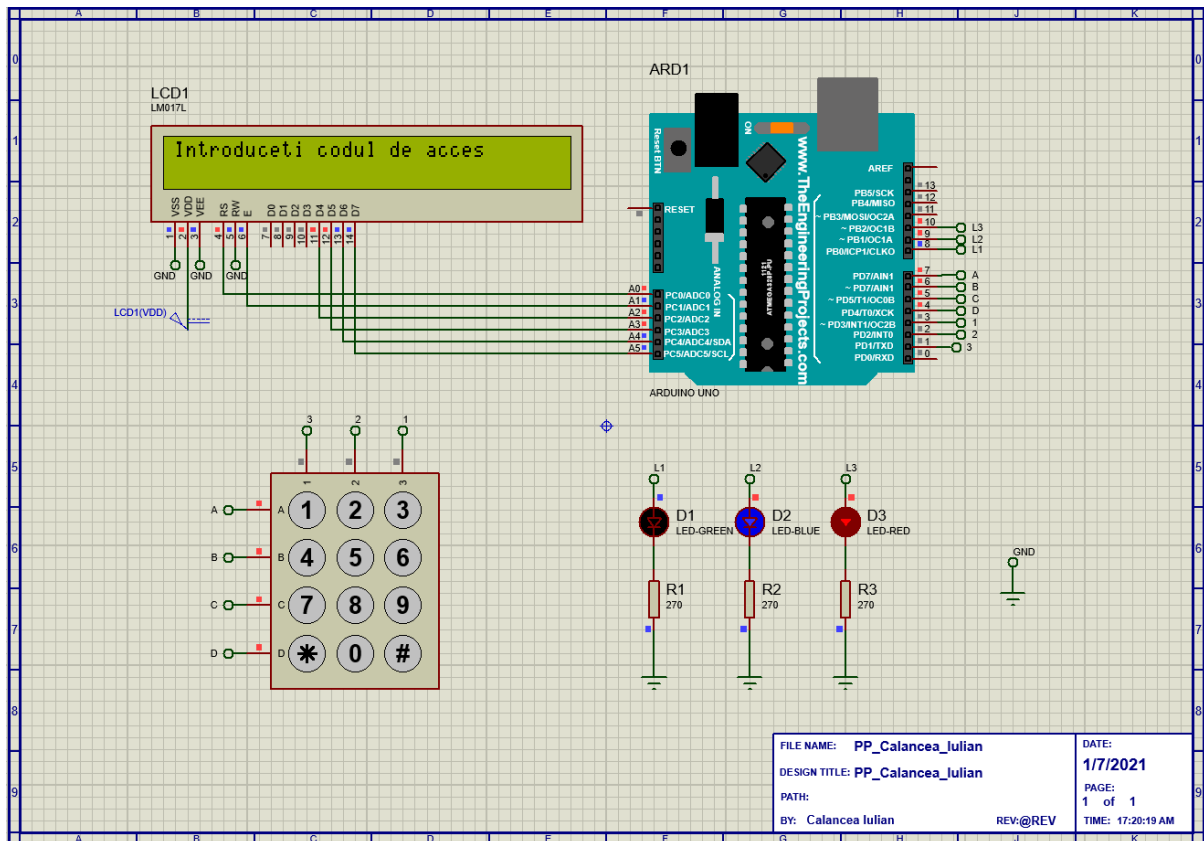
Un astfel de sistem de blocare automată constă dintr-un ansamblu de control electronic, care controlează sarcina de ieșire printr-o parolă. Această sarcină de ieșire poate fi un motor sau orice altă sarcină mecanică / electrică.[1]

## 2. Circuit electric (Proteus)

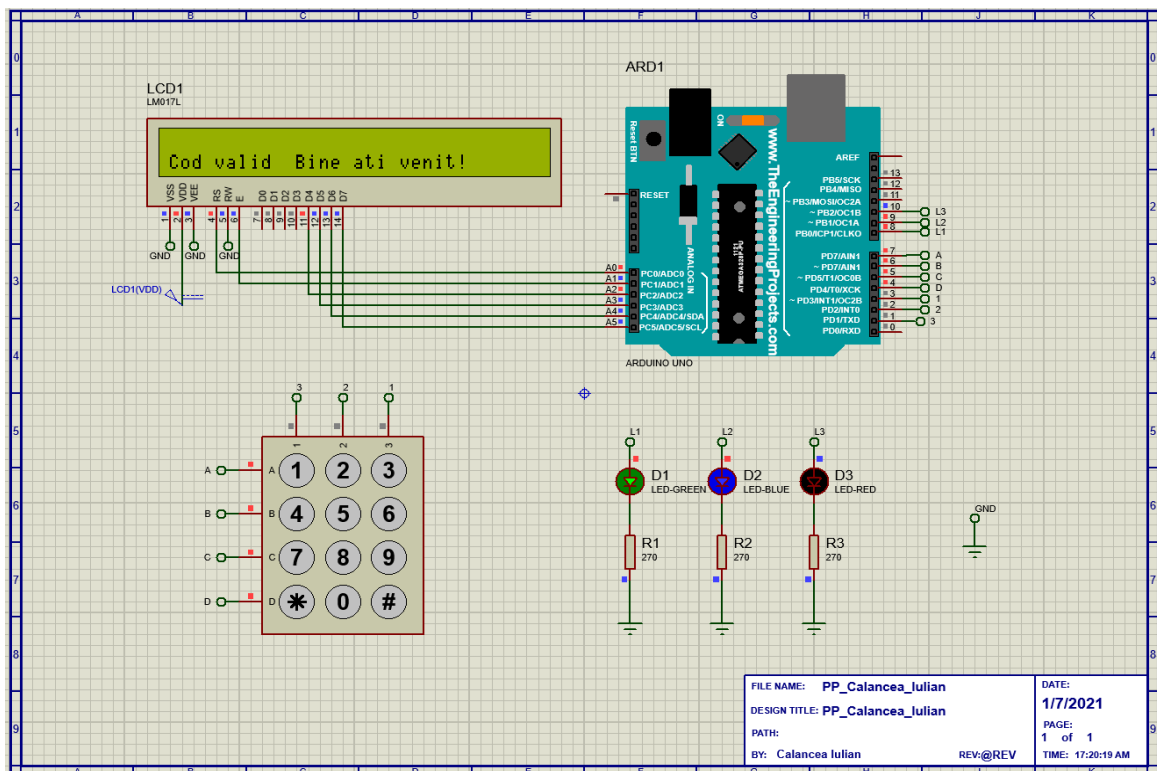
Circuitul realizat reprezintă o încuietore cu cod pentru o ușă. Acesta este compus din: o placă Arduino UNO, un ecran LCD de tip LM017L, 3 leduri: unul roșu, unul albastru și unul verde, un keypad și o sursă de tensiune pentru alimentarea ecranului LCD.

Funcționarea circuitului este una simplă: la început, pe ecranul LCD apare un mesaj de bun venit, apoi apare un mesaj prin care se cere introducerea codului.

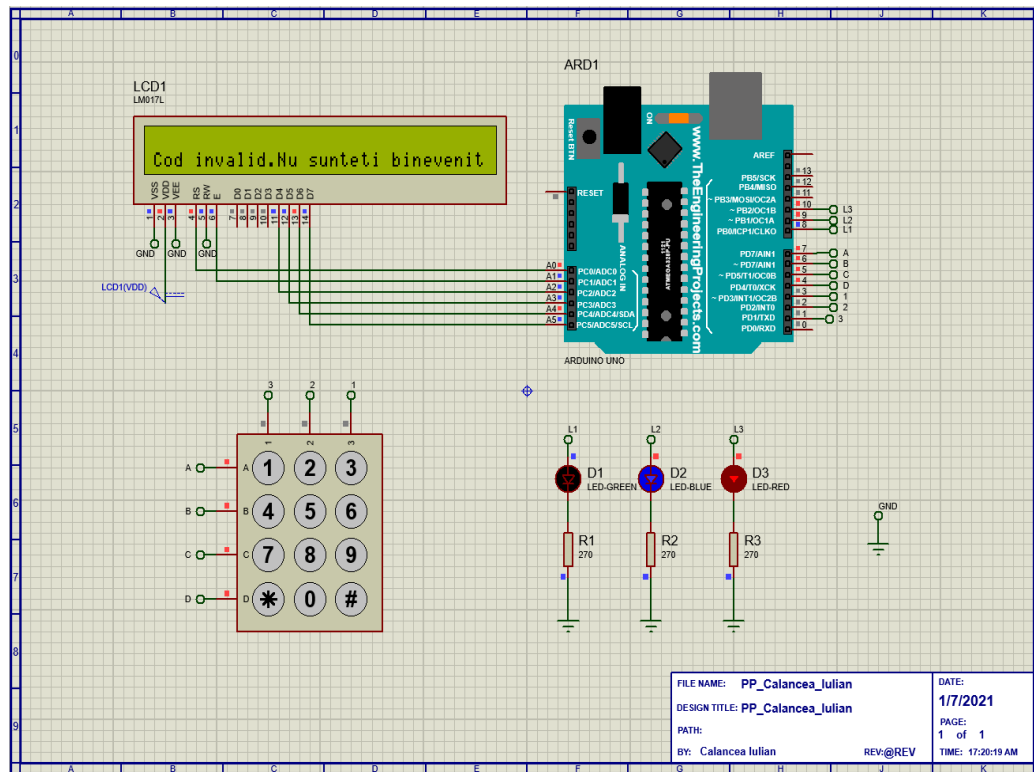
Încuietorea fiind inițial închisă, ledul roșu este aprins. Ledul verde se va aprinde în momentul în care se introduce codul corect. Ledul albastru va sta mereu aprins, arătând dacă încuietorea este funcțională. La introducerea succesivă a 3 coduri gresite, utilizatorul este informat că poliția se află pe drum spre locul în care se află încuietorea.



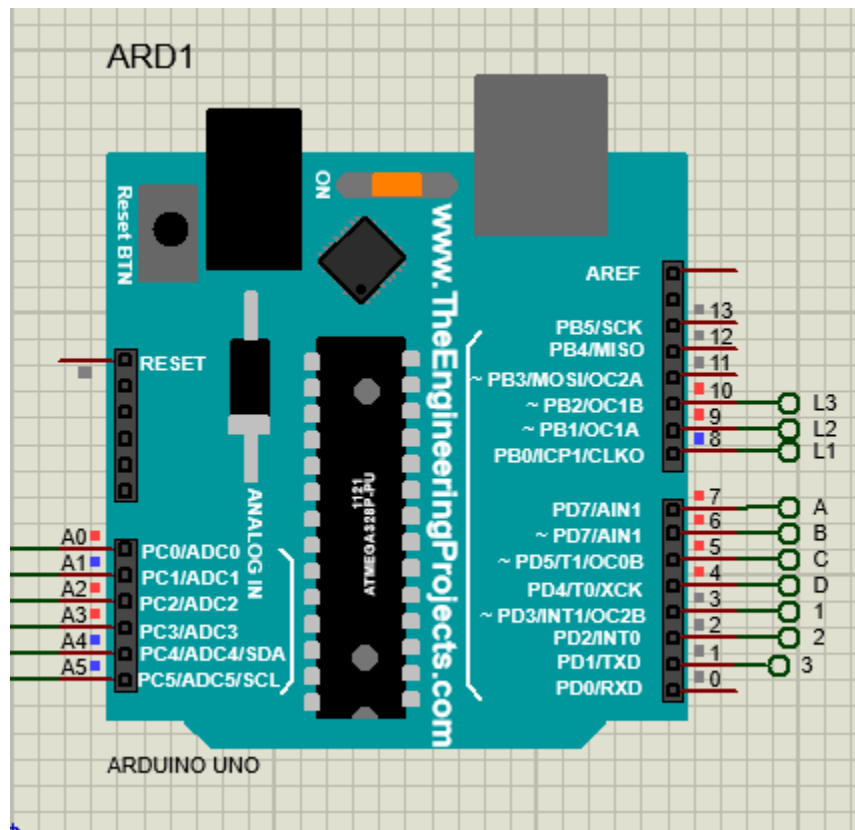
Circuitul principal



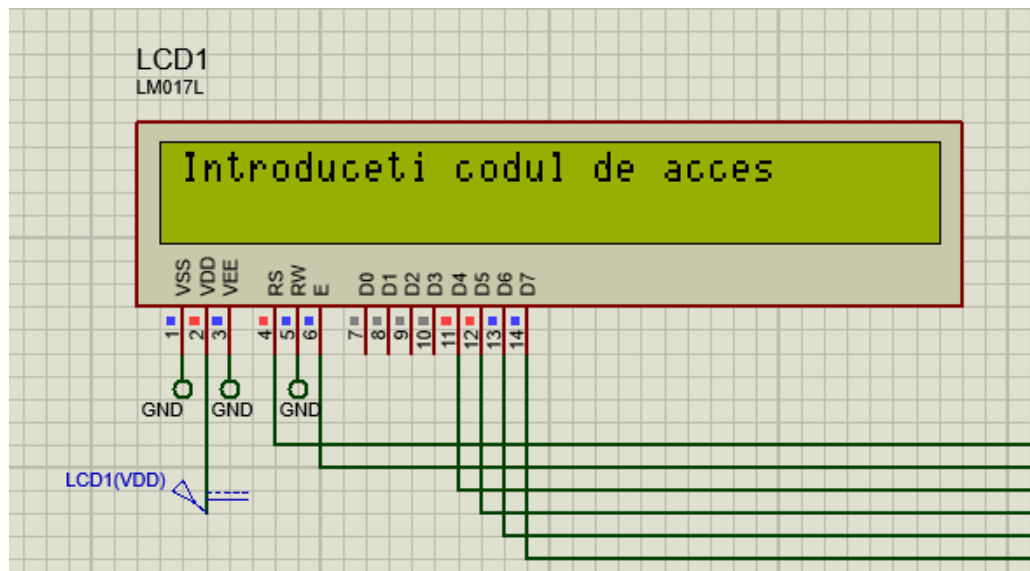
Introducerea corecta a codului



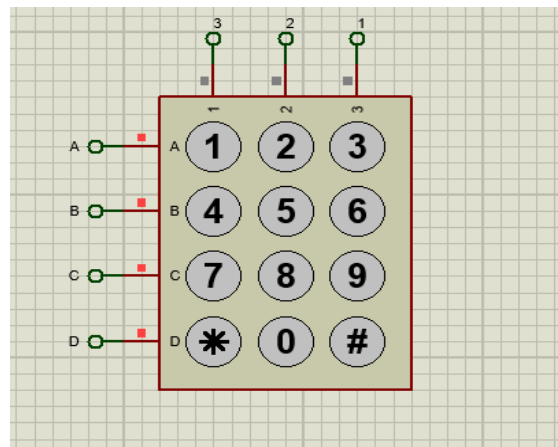
Introducerea incorecta a codului



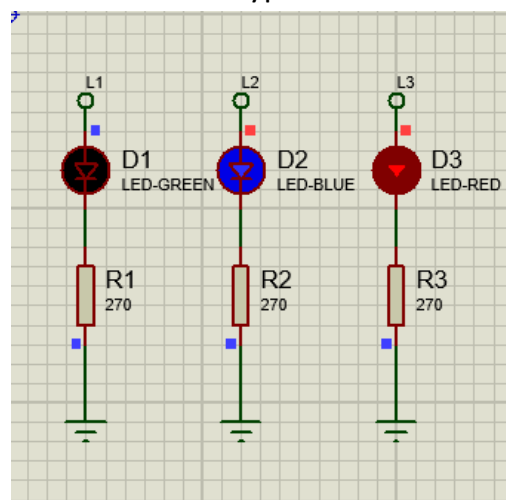
Arduino UNO



Ecran LCD



Keypad



Leduri

### 3. Arduino IDE

```
1 //Librariile necesare
2 #include <Keypad.h>
3 #include <LiquidCrystal.h>
4
5 //Matricea ajutatoare pentru maparea Keypad-ului
6 char tastatura[4][3]={
7   {'*', '0', '#'},
8   {'7', '8', '9'},
9   {'4', '5', '6'},
10  {'1', '2', '3'}};
11
12 byte PiniRanduri[4] = {4,5,6,7}; //Declarare pini pentru randurile Keypad-ului
13 byte PiniColoane[3] = {1,2,3}; //Declarare pini pentru coloanele Keypad-ului
14
15 Keypad keypad = Keypad( makeKeymap(tastatura), PiniRanduri, PiniColoane, 4, 3 ); //Maparea Keypad-ului
16 LiquidCrystal lcd(A0, A1, A2, A3, A4, A5); //Declarare pini pentru LCD
17
18 //Declarare pini leduri
19 int redPin = 10;
20 int bluePin = 9;
21 int greenPin = 8;
22
23 String parola = "1234"; //Declarare parola
24
25 //Variabile necesare programului
26 int pozitie = 0;
27 int greseli = 0;
28 int total = 0;
29
30
31 void setup() {
32
33   pinMode(redPin,OUTPUT);
34   pinMode(bluePin,OUTPUT);
35   pinMode(greenPin,OUTPUT);
36   digitalWrite(bluePin, HIGH);
37
38   lcd.begin(16,2);
39   lcd.setCursor(0,1);
40   lcd.print("Bine ati venit!"); //Afisare mesaj de intampinare
41   delay(1000);
42   lcd.clear();
43   delay(100);
44   setLocked(true); //Functie care inchide si deschide incuietoarea
45   delay(20);
46 }
47
48
49 void loop() {
50
```

```

50
51 lcd.print("Introduceti codul de acces");
52 delay(100);
53 String cod[3];
54 while(pozitie<4){ //Se efectueaza operatii atata timp cat nu am ajuns la finalul codului
55
56 char apasat = keypad.getKey(); // functia keypad.getKey() returneaza true in momentul in care este apasata o tasta
57 if(apasat)
58 {
59 lcd.setCursor(pozitie,1); //pregatim cursorul pentru introducerea caracterelor
60 lcd.print('*');
61 delay(350);
62
63 if(apasat == '*' || apasat == '#') // Daca se apasa tasta * sau # se sterge codul apasat pana in acel moment si se revine la prima pozitie
64 {
65 pozitie = 0;
66 setLocked(true);
67 }
68
69 else if(apasat == parola[pozitie]) //Daca tasta apasata corespunde cu caracterul de pe aceeaasi pozitie din sirul parola atunci se continua cu introducerea codului
70 {
71 cod[pozitie]=apasat;
72 pozitie++;
73 }
74
75 else if(apasat != parola[pozitie]) //Daca tasta apasata nu corespunde cu caracterul de pe aceeaasi pozitie atunci se incrementeaza variabila greseli si se continua
76 {
77 greseli++;
78 pozitie++;
79 }
80 if(pozitie == 4) //Testam daca am introdus intregul cod de 4 cifre
81 {
82 if(greseli > 0) //Testa daca avem cel putin o gresala in codul introdus
83 {
84 total++; //Incrementam variabila total care numara numarul de coduri gresite introduse
85 greseli = 0;
86 pozitie = 0;
87 lcd.clear();
88 lcd.setCursor(0,1);
89 lcd.print("Cod invalid.Nu sunteti binevenit"); //Afisarea mesajului pentru introducerea unui cod invalid
90 setLocked(true); //Apelam functia setLocked(true) care o sa mentina mecanismul inchis
91 delay(1000);
92 lcd.clear();
93 lcd.print("Introduceti codul de acces");
94 }
95
96 else if(greseli == 0) //Testam daca nu avem nici o gresala in codul introdus
97 {
98 pozitie = 0;
99 greseli = 0;
100 lcd.clear();
101 lcd.setCursor(0,1);
102 lcd.print("Cod valid");
103 lcd.setCursor(11,2);
104 lcd.print("Bine ati venit!");
105 setLocked(false); //Apelam functia setLocked(false) care o sa deschida mecanismul
106 lcd.clear(); //Curatam ecranul LCD-ului
107 delay(1000);
108 lcd.print("Introduceti codul de acces");
109 }
110
111 if(total == 3) //Testam daca am introdus 3 coduri gresite de acces
112 {
113 total = 0;
114 lcd.clear();
115 lcd.print("Politia este pe drum");
116 delay(1000);
117 digitalWrite(bluePin, LOW);
118 digitalWrite(redPin, LOW);
119 delay(300);
120 for(int i=0; i<4;i++){ //Folosim instructiunea for pentru a putea aprinde si stinge ledurile albastru si rosu intermitent
121 digitalWrite(redPin, HIGH);
122 delay(300);
123 digitalWrite(bluePin, HIGH);
124 digitalWrite(redPin, LOW);
125 delay(300);
126 digitalWrite(bluePin, LOW);
127 }
128 digitalWrite(bluePin, HIGH);
129 digitalWrite(redPin, HIGH);
130 lcd.clear();
131 lcd.print("Introduceti codul de acces");
132 }
133 }
134 }

```

```
135
136 //Functia setlocked deschide sau inchide mecanismul. Totodata aceasta functie aprinde ledurile in functie de corectitudinea codului introdus
137 void setLocked(int inchis)
138 {
139     if (inchis)
140     {
141         digitalWrite(redPin, HIGH);
142         digitalWrite(greenPin, LOW);
143         delay(1000);
144     }
145     else
146     {
147         digitalWrite(redPin, LOW);
148         digitalWrite(greenPin, HIGH);
149         delay(2000);
150         digitalWrite(redPin, HIGH);
151         digitalWrite(greenPin, LOW);
152     }
153 }
```

## 4. Bibliografie

[1]: <https://www.electronicshub.org/password-based-door-lock-system-using-8051-microcontroller/>

[2]: <https://www.elprocus.com/password-based-electronic-lock-system/>