

Proiectare cu microprocesoare

Arduino Slot Machine

Andrei-Denis Alexandru, Tamas Dari

Gr. 30231

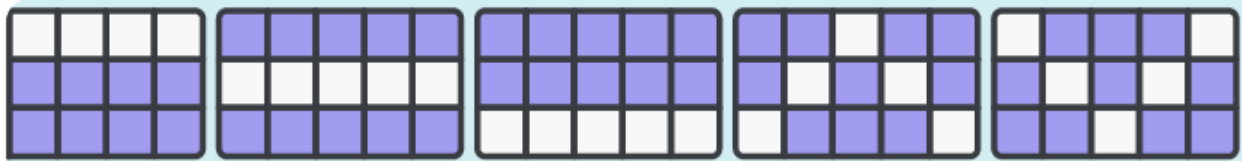
Profesor indrumator: Razvan Itu

An universitar 2022-2023

1. Prezentarea temei

Tema pe care am ales-o pentru proiectul prezent este implementarea cu Arduino a unui dispozitiv similar cu un aparat de jocuri de noroc. Utilizatorul dispune de o suma de bani pentru a plasa un pariu, in speranta primirii uneia sau mai multor linii castigatoare.

Se considera castig momentul in care se primesc cel putin 3 simboluri de acelasi fel pe una din liniile desemnate.



Cu cat sunt mai multe simboluri identice pe linie, cu atat castigul este mai mare, iar mai multe linii castigatoare aduc un castig adunat

In cazul in care se primește o linie castigatoare, utilizatorul are posibilitatea de a incerca sa isi dubleze castigurile ghicind urmatorul simbol generat de dispozitiv (sansa de 50%).

Pentru implementarea proiectului, vom avea nevoie de o placuta Arduino, un display LCD 20x4 cu modul I2C si 5 butoane (implementarea noastra foloseste o matrice keypad 4x4 de butoane).

2. Solutia aleasa

Pentru dezvoltarea proiectului am folosit placutele de dezvoltare Arduino Uno (personala) si Arduino Mega (laborator), varianta Uno fiind suficienta pentru functionarea sa corecta.

Am inceput prin a face un design al simbolurilor afisate de dispozitiv. Tinand cont ca displayul nu poate oferi culori RGB, am hotarat sa cream urmatoarele simboluri: cirese, clopotel, dolar, septar, inima plina (neagra) si inima rosie (goala), sub forma de arrayuri de 8 bytes, urmand ca fiecare simbol sa fie asociat unui caracter (numar de la 0 la 5).

```
byte cirese[8] = { 0b111100, 0b011110, 0b000001, 0b000010, 0b000110, 0b010101, 0b110111, 0b110111, };
byte septar[8] = { 0b000000, 0b111111, 0b100001, 0b000001, 0b000010, 0b000100, 0b000100, 0b000000, };
byte bell[8] = { 0b000000, 0b000100, 0b011110, 0b011110, 0b011110, 0b111111, 0b000100, 0b000000, };
byte dolar[8] = { 0b000000, 0b000100, 0b011111, 0b101000, 0b011110, 0b000101, 0b111110, 0b000100, };
```



UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

Am creat apoi o functie ce genereaza o matrice 3x5 de numere random cu valori intre 0 si 4, pentru a mapa matricea de simboluri.

```
void generateRandomMatrix(int arr[3][5]) {  
    for (int i = 0; i < 3; i++) {  
        for (int j = 0; j < 5; j++) {  
            arr[i][j] = random(0, 5);  
        }  
    }  
    matrixIsInitiated = true;  
}
```

Aceasta functie este apelata cand balansul utilizatorului este mai mare decat valoarea pariului selectat.

```
int arr[3][5];  
if (digitalRead(pinButtonRoll) == LOW) {  
    balance += winning;  
    modDublaj = false;  
    if (balance - bet >= 0) {  
        balance -= bet;  
        lcd.clear();  
        generateRandomMatrix(arr);  
        printRandomMatrix(arr);  
        winning = checkPaylines(arr);  
        lcd.setCursor(0, 0);  
        lcd.print(balance);  
        lcd.setCursor(0, 1);  
        lcd.print(winning);  
        lcd.setCursor(0, 2);  
        lcd.print(bet);  
    }  
    delay(500);  
}
```



UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

Urmatoarea functie creata este cea responsabila cu verificarea liniilor castigatoare, verificand initial liniile orizontale iar apoi cele diagonale. Fiecarui simbol ii este atribuita o valoare (ex: cireasa valoreaza cel mai putin, in timp ce saptele valoreaza cel mai mult), iar in functie de numarul de aparitii, castigul este cumulat si actualizat.

```
float checkPaylines(int arr[3][5]) {  
    float totalWinnings = 0;  
  
    // Check horizontal paylines  
    for (int i = 0; i < 3; i++) {  
        int count = 0;  
        for (int j = 0; j < 5; j++) {  
            if (arr[i][0] == arr[i][j]) {  
                count++;  
            } else {  
                break;  
            }  
        }  
        int value = 0;  
        if (arr[i][0] == 0)  
            value = 1;  
        else if (arr[i][0] == 2)  
            value = 5;  
        else  
            value = 3;  
        if (count == 3) {  
            totalWinnings += 1.25 * value;  
        } else if (count == 4) {  
            totalWinnings += 1.75 * value;  
        } else if (count == 5) {  
            totalWinnings += 2.50 * value;  
        }  
    }  
  
    int countDiag = 0;  
    if (arr[0][0] == arr[1][1] && arr[1][1] == arr[2][2]) {  
        countDiag = 3;  
    }  
    if (arr[0][0] == arr[1][1] && arr[1][1] == arr[2][2]  
        && arr[2][2] == arr[1][3]) {  
        countDiag = 4;  
    }  
    if (arr[0][0] == arr[1][1] && arr[1][1] == arr[2][2]  
        && arr[2][2] == arr[1][3] && arr[1][3] == arr[0][4]) {  
        countDiag = 5;  
    }  
  
    int value = 0;  
    if (arr[0][0] == 0)  
        value = 1;  
    else if (arr[0][0] == 2)  
        value = 5;  
    else  
        value = 3;  
    if (countDiag == 3) {  
        totalWinnings += 1.25 * value;  
    }  
    else if (countDiag == 4) {  
        totalWinnings += 1.75 * value;  
    }  
    else if (countDiag == 5) {  
        totalWinnings += 2.50 * value;  
    }  
}
```



UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

În cazul unui câștig, utilizatorul poate intra în modul de dublare prin apăsarea uneia din cele două butoane (Rosu și Negru). Acestuia îi se prezintă ultimele 3 simboluri generate de dispozitiv, iar acesta trebuie să ghicească culoarea următorului simbol. În cazul în care răspunsul utilizatorului este corect, câștigul acestuia se dublează.

```
if ((modDublaj == false) && (winning > 0) &&
    (digitalRead(pinButtonN) == LOW || digitalRead(pinButtonR) == LOW)){
    modDublaj = true;
    doubleWin();
    delay(500);
}
if ((modDublaj == true) && (digitalRead(pinButtonR) == LOW)){
    int newChoice = 5;
    for(int i = 2; i > 0; i--){
        symbols[i] = symbols[i - 1];
        symbols[0] = random(4, 6);

        if( newChoice == symbols[0] ) {
            winning = winning * 2;
        }
        else{
            winning = 0;
            modDublaj = false;
        }
    }
}
```

Utilizatorul are de asemenea șansa să își schimbe valoarea pariului cu apăsarea unui buton dedicat, acesta fluctuând între 1, 2, 5 și 10.

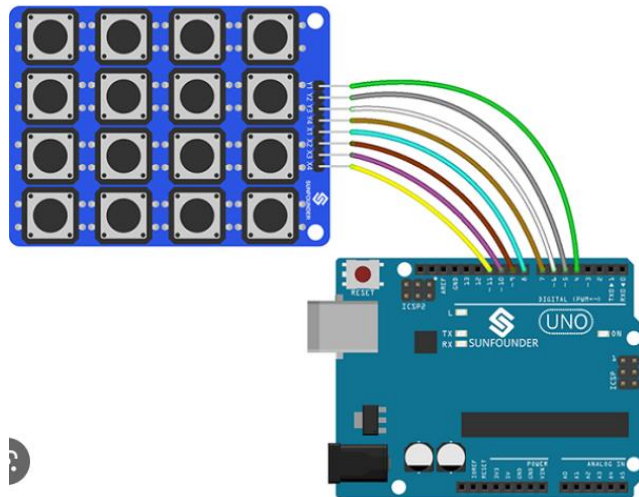
```
if (digitalRead(pinButtonBet) == LOW) {
    increaseBet();
    lcd.clear();
    if (matrixIsInitiated == true){
        printRandomMatrix(arr);
        winning = checkPaylines(arr);
    }
    lcd.setCursor(0,0);
    lcd.print(balance);
    lcd.setCursor(0,1);
    lcd.print(winning);
    lcd.setCursor(0,2);
    lcd.print(bet);
    delay(500);
}
```

3. Schema circuitului

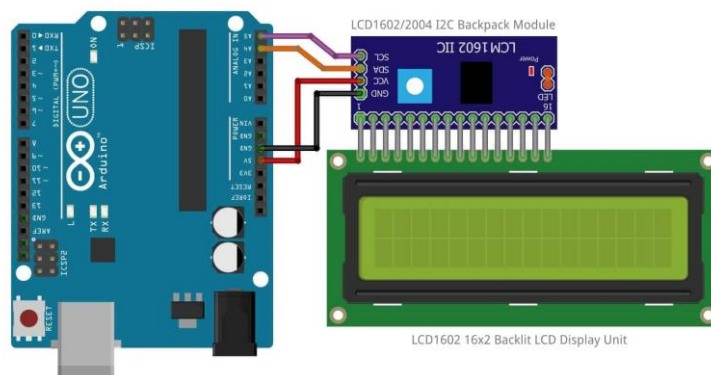
Pentru realizarea proiectului am folosit o matrice de butoane pentru citirea operațiilor utilizatorului. Utilizatorul are posibilitatea să dea următoarele instrucțiuni cu ajutorul butoanelor:

- Roll
- Change Bet
- Double Red
- Double Black
- Add money

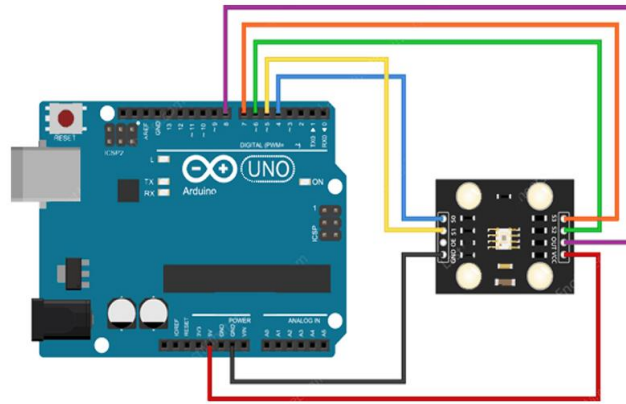
Matricea de butoane este conectat în felul următor, dar noi folosim numai primele cinci butoane:



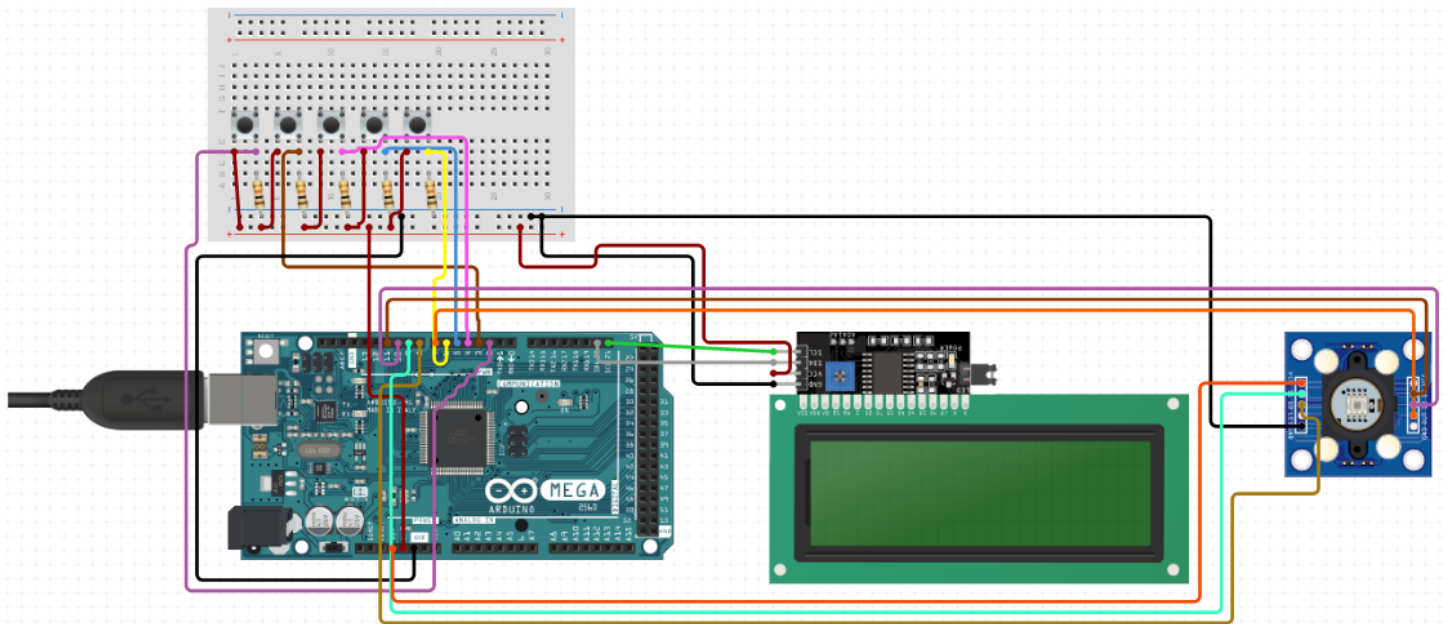
Pentru a comunica cu utilizatorul am folosit un display LCD 20x4 cu modul I2C.



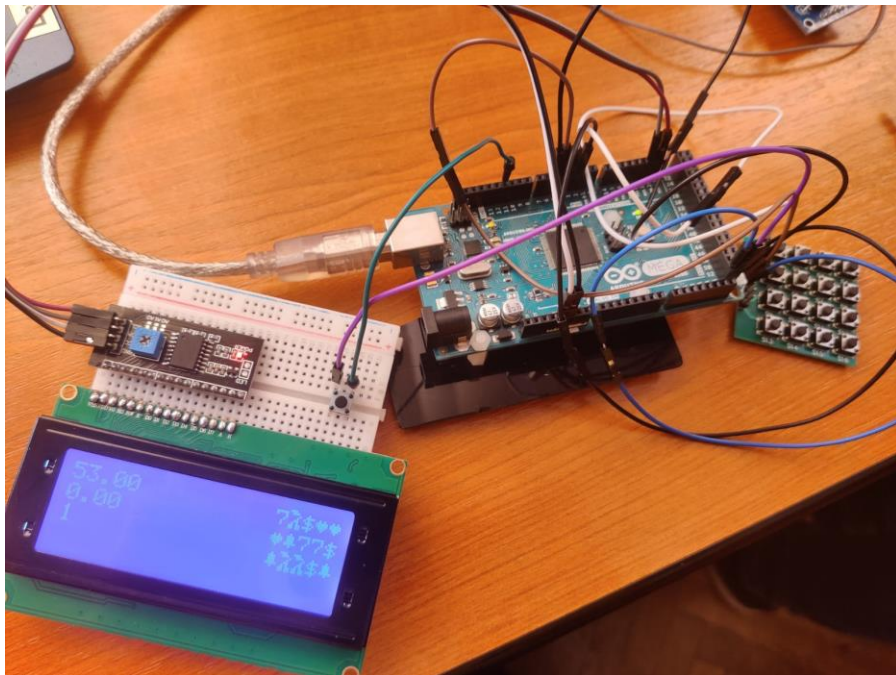
În final pentru citirea bancnotelor am folosit un senzor de culoare, dar folosirea senzorului a cauzat câteva probleme.



Circuitul cu toate componente arată în felul următor:



4. Poze cu proiectul





UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

5. Bibliografie

- <https://rydepier.wordpress.com/2015/10/20/4x4-matrix-keyboard-with-arduino/>
- <https://forum.arduino.cc/t/lcd-2004-20x4-i2c-library/951243>
- <https://circuitdigest.com/microcontroller-projects/arduino-currency-counter-using-ir-and-color-sensor>