

MINISTERUL EDUCAȚIEI NAȚIONALE



**UNIVERSITATEA TEHNICĂ**  
DIN CLUJ-NAPOCA

---

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**  
**DEPARTAMENTUL AUTOMATICA**

# ILUMINAT AMBIENTAL

AUTOR: Sava Diana-Alexandra

## **1. Cerința temei:**

Cerința temei este: „Privitul îndelungat la un televizor sau monitor poate să cauzeze oboseala ochilor, de aceea este bine să folosim un iluminat ambiental. Cu cât crește intensitatea luminoasă a televizorului/ecranului, cu atât iluminatul ambiental va fi mai luminos, și invers.”

## **2. Soluții alese:**

Scopul aplicației este adaptarea dinamică a iluminatului ambiental în funcție de intensitatea luminoasă a unui ecran, pentru a reduce oboseala ochilor cauzată de expunerea prelungită la lumină puternică sau insuficientă.

### **▪ Structura logică a aplicației:**

- Sursă de lumină principală (ecran) — emite un nivel variabil de lumină, reprezentat printr-un parametru numeric ce determină intensitatea luminoasă.
- Senzor de lumină — măsoară intensitatea luminii emise de ecran, oferind o valoare continuă ce reflectă nivelul real de iluminare ambientală generată de ecran.
- Sursă de lumină ambientală — reglată automat pentru a ajusta nivelul de iluminare în funcție de datele primite de la senzor, astfel încât să fie menținut un raport optim între luminozitatea ecranului și iluminatul ambiental.



■ **Algoritmul propus:**

- Sistemul operează într-o buclă continuă, care monitorizează și ajustează constant iluminatul ambiental.
- Intensitatea luminoasă a ecranului este setată printr-un input de la utilizator, reprezentând nivelul dorit.
- Valoarea intensității ecranului este transpusă într-un semnal de control (PWM) care reglează sursa de lumină principală.
- Senzorul preia în timp real nivelul de lumină emisă de ecran și transmite această informație sistemului de control.

Avantajul acestei abordări este că iluminatul ambiental se ajustează automat, fără ca utilizatorul să mai intervină.

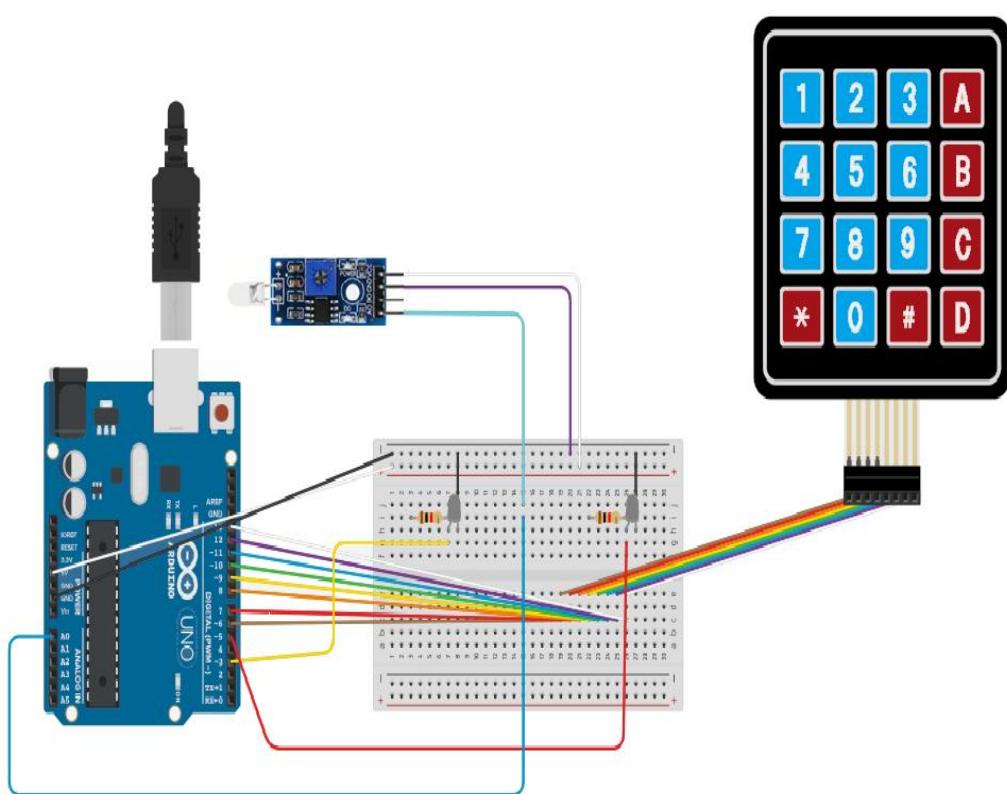
### 3. Rezultate obținute:

Piese folosite: placă compatibilă cu Arduino, breadbord, fire, două LED-uri albe (un LED alb ține loc de ecran, iar al doilea LED alb simulează lumina ambientală), două rezistențe de  $220\ \Omega$  (folosite pentru LED-uri, asigurând un curent constant și protejându-le de suprasarcină), senzor de lumină, tastatură matricială 4x4:

1. Se așteaptă input de la tastatură. Fiecare apăsare de tastă din gama 0–9 corespunde unui nivel prestabilit de intensitate pentru LED-ul 1 alb. Cu cât cifra crește, cu atât și luminozitatea LED-ului crește.
2. La fiecare modificare, se convertește valoarea numerică într-o valoare PWM (de la 0 la 255), ce controlează lumina emisă de „ecran”.
3. Senzorul de lumină preia în mod continuu nivelul de lumină ambientală (emisă de LED-ul 1 alb, care simulează ecranul).
4. Valorile preluate sunt transformate proporțional într-o intensitate a LED-ului 2 alb. Astfel, cu cât ecranul devine mai luminos, cu atât crește și lumina ambientală.
5. Se actualizează valoarea PWM a LED-ului 2 alb.

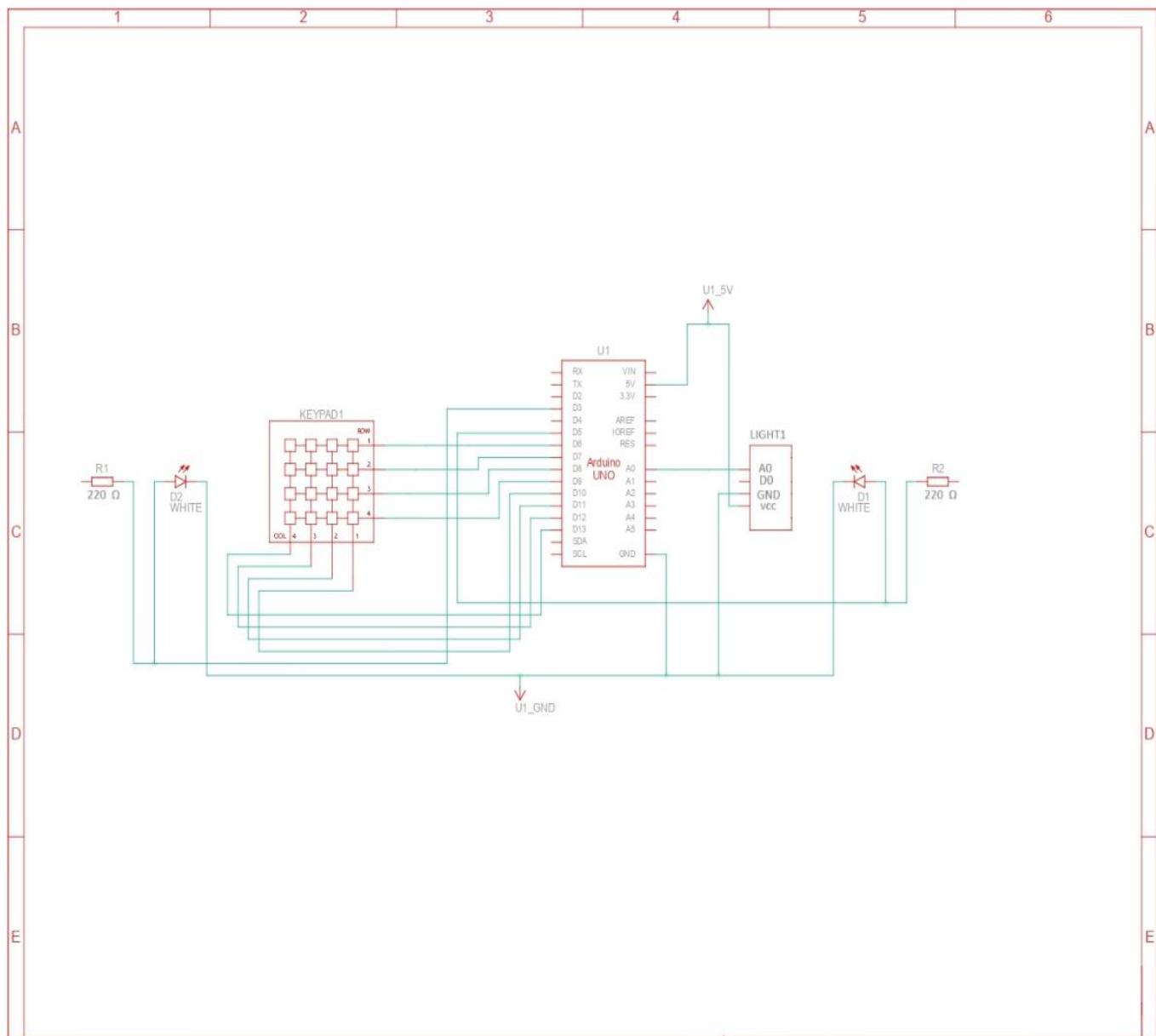
### Conexiuni:

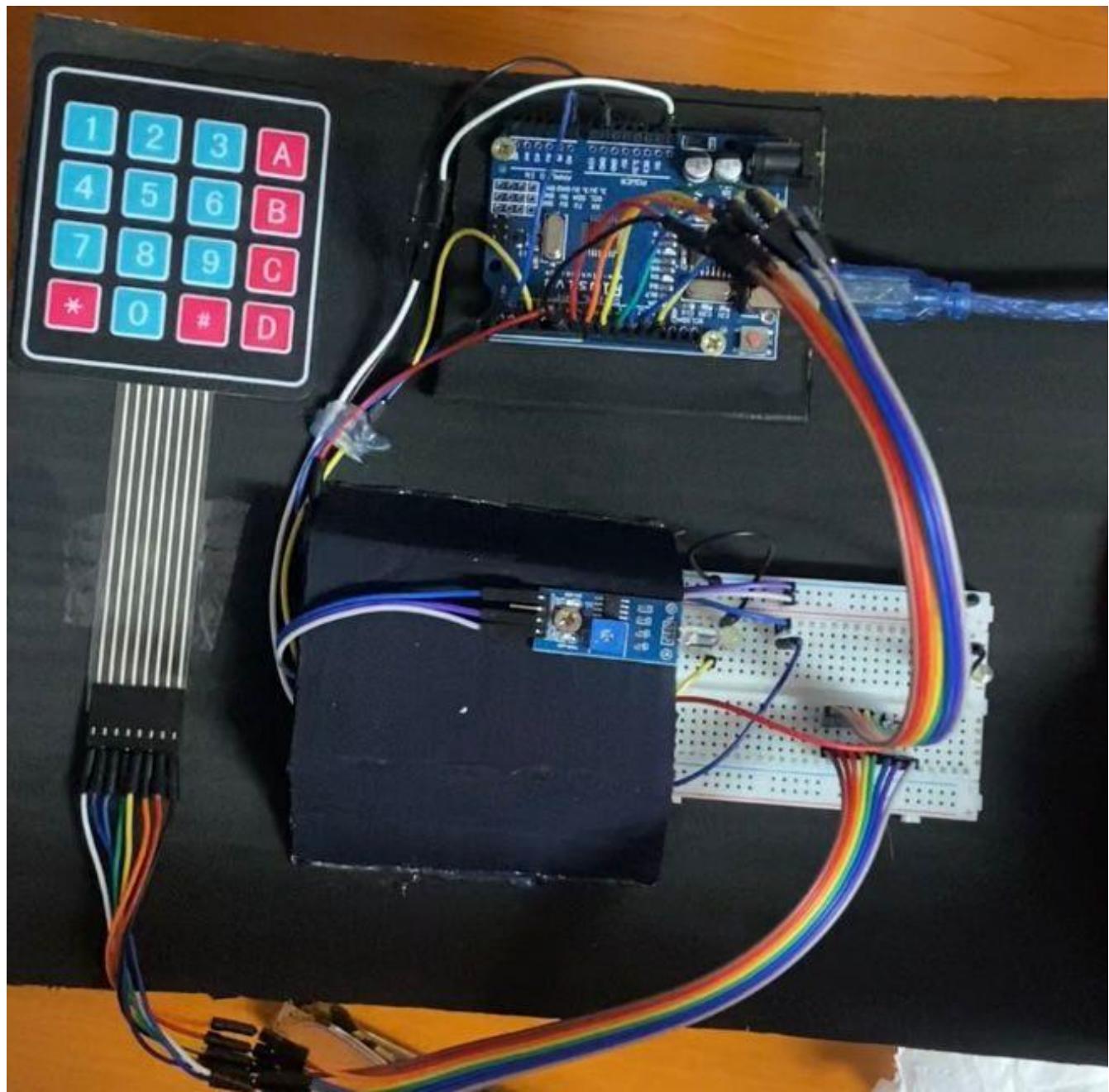
- LED 1 alb – pin digital 3
- LED 2 alb – pin digital 5
- senzor de lumină – pin analogic A0
- tastatură 4x4 – pini digitali 6-13





**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOROARE**  
**DEPARTAMENTUL AUTOMATICA**





## 4. Testări și verificări:

Aplicația este simplă și intuitivă, iar utilizarea sa nu necesită cunoștințe tehnice din partea utilizatorului.

Instrucțiuni de utilizare:

1. Dispozitivul trebuie conectat la sursa de alimentare (spre exemplu prin cablu USB la un calculator/ laptop sau un adaptor de alimentare)
2. Utilizatorul folosește tastatura matricială 4x4 pentru a seta intensitatea luminoasă a „ecranului”:
  - Tasta **0** – LED-ul ecran (LED 1) este oprit.
  - Tastele **1–9** – cresc progresiv intensitatea luminii emise, unde **1** este minimul și **9** este maximul.
3. Imediat ce „ecranul” (LED 1) este pornit, „lumina ambientală” (LED 2) se activează automat.
4. Intensitatea luminii ambientale este ajustată automat, proporțional cu luminozitatea „ecranului”. Cu cât ecranul este mai luminos, cu atât și iluminatul ambiental devine mai intens.

Această funcționalitate este menită să protejeze ochii utilizatorului atunci când stă timp îndelungat în fața unui ecran, oferind un iluminat ambiental adaptiv, care reduce contrastul dintre ecran și mediu.

## Observații:

- Nu este necesară nicio calibrare manuală. Senzorul de lumină și algoritmul automat se ocupă de ajustarea luminii ambientale.
- Sistemul reacționează în timp real la modificările făcute de utilizator.

## ANEXE:

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

```
const byte ROWS = 4;
```

```
const byte COLS = 4;
```

```
char keys[ROWS][COLS] = {
```

```
    {'1','4','7','*'},
```

```
    {'2','5','8','0'},
```

```
    {'3','6','9','#'},
```

```
    {'A','B','C','D'}
```

```
};
```

```
byte rowPins[ROWS] = {9, 8, 7, 6};
```

```
byte colPins[COLS] = {13, 12, 11, 10};
```

```
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
```

```
const int ledAlb1Pin = 3;
```

```
const int ledAlb2Pin = 5;
```

```
const int senzorLuminaPin = A0;
```

```
int intensitateAlb1 = 0; // 0-9, controlat de tastatura
```

```
void setup() {  
    Serial.begin(9600);  
    pinMode(ledAlb1Pin, OUTPUT);  
    pinMode(ledAlb2Pin, OUTPUT);  
    Serial.println("Start control led alb si lumina ambientala");  
}
```

```
void loop() {  
    char key = keypad.getKey();  
  
    if (key != NO_KEY) {  
        Serial.print("Tasta detectata: ");  
        Serial.println(key);  
    }  
}
```

```
if (key >= '0' && key <= '9') {  
    intensitateAlb1 = key - '0';  
}  
else if (key == 'A') {  
    intensitateAlb1 = 9;
```

```
intensitateAlb1 = 0;  
}  
}
```

```
int pwmAlb1 = map(intensitateAlb1, 0, 9, 0, 255);  
analogWrite(ledAlb1Pin, pwmAlb1);
```

```
int valoareLumina = analogRead(senzorLuminaPin);  
Serial.print("Valoare senzor lumina: ");  
Serial.println(valoareLumina);
```

```
int intensitateAlb2;  
  
intensitateAlb2= map(valoareLumina, 0, 1023, 255, 0);  
analogWrite(ledAlb2Pin, intensitateAlb2);
```

```
Serial.print("Intensitate alb1: ");  
Serial.print(pwmAlb1);  
Serial.print(" | Intensitate alb2: ");  
Serial.println(intensitateAlb2);
```

```
delay(100); }
```