

ErgoDesk Assistant - Mantu Ionut Gabriel

Introducere

ErgoDesk Assistant este un dispozitiv ergonomic de birou conceput pentru a monitoriza in timp real doi indicatori asociati unei pozitii necorespunzatoare in timpul lucrului la birou: distanta prea mica fata de monitor si inclinarea unui suport de referinta. Sistemul ofera feedback local printr-un afisaj LCD, un LED RGB si un buzzer, astfel incat utilizatorul sa poata corecta rapid pozitia detectata ca problematica.

Ideea proiectului a pornit de la o problema reala intalnita frecvent atat de studenti, cat si de persoanele care lucreaza mult timp la calculator: apropierea excesiva de monitor si mentinerea unei pozitii incomode pentru perioade lungi. Proiectul isi propune sa ofere un mecanism simplu de avertizare si constientizare a acestor obiceiuri.

Dispozitivul poate fi util atat pentru utilizatorul final, cat si pentru mine, deoarece combina senzori, procesare locala si feedback vizual si sonor intr-un produs embedded cu aplicabilitate practica.

Descriere generala

Sistemul are la baza o placa compatibila Arduino Uno, care colecteaza datele de la senzori si genereaza feedback local in functie de starea detectata.

Componentele principale conectate la microcontroler sunt:

- **HC-SR04** - masoara distanta fata de utilizator. Daca utilizatorul ramane prea aproape de ecran pentru mai mult de cateva secunde, sistemul trece in stare de avertizare.
- **MPU6050** - masoara inclinarea si unghiul unui suport de referinta. O inclinare care depaseste pragul ales este interpretata ca indiciu al unei pozitii necorespunzatoare.
- **Fotorezistor** - masoara lumina ambientala si permite adaptarea comportamentului sistemului in functie de conditiile de iluminare, de exemplu reducerea intensitatii feedback-ului seara.
- **LCD 1602 cu interfata I2C** - afiseaza valorile citite de la senzori, starea curenta a sistemului si mesajele de avertizare.
- **LED RGB** - ofera feedback vizual rapid:
 - verde = pozitie normala
 - galben = apropiere de limita
 - rosu = avertizare
- **Buzzer pasiv** - ofera feedback sonor atunci cand starea problematica persista.
- **Buton** - utilizat pentru calibrare, reset sau schimbare mod.

Functionarea sistemului este bazata pe citirea periodica a senzorilor si pe compararea valorilor obtinute cu praguri predefinite sau calibrate de utilizator.

Schema bloc



Hardware Design

Bill of Materials

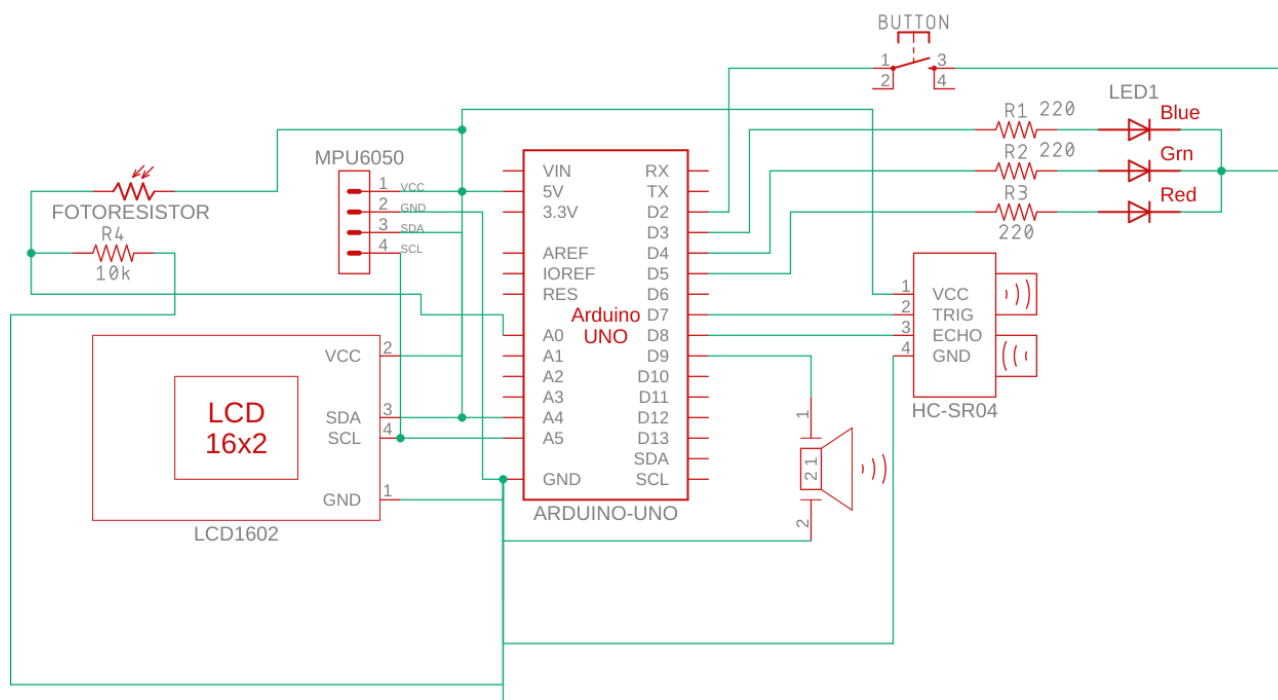
Componenta	Cantitate	Link	Pret unitar (RON)
Placa compatibila Arduino Uno	1	Kit Plusivo Microcontroller Starter	inclus in kit
Senzor ultrasonic HC-SR04	1	Kit Plusivo Microcontroller Starter	inclus in kit
LED RGB	1	Kit Plusivo Microcontroller Starter	inclus in kit
Buzzer pasiv	1	Kit Plusivo Microcontroller Starter	inclus in kit
Fotorezistor	1	Kit Plusivo Microcontroller Starter	inclus in kit
Buton	1	Kit Plusivo Microcontroller Starter	inclus in kit
Breadboard	1	Kit Plusivo Microcontroller Starter	inclus in kit
Fire de legatura	mai multe	Kit Plusivo Microcontroller Starter	inclus in kit
Rezistente 220 ohm	3	Kit Plusivo Microcontroller Starter	inclus in kit
Rezistenta 10k ohm	1	Kit Plusivo Microcontroller Starter	inclus in kit
Modul accelerometru/giroscoap MPU6050	1	eMAG MPU6050	15.46
Afisaj LCD 1602 cu I2C	1	eMAG LCD 1602 I2C	30.25

Pret kit baza: 74.99 RON

Pret componente achizitionate separat: 45.71 RON

Pret total hardware utilizat in proiect: 120.70 RON

Schema electrica



Conexiuni principale

• LCD 1602 I2C

- VCC → 5V
- GND → GND
- SDA → A4
- SCL → A5

• MPU6050

- VCC → 5V
- GND → GND
- SDA → A4
- SCL → A5

• HC-SR04

- TRIG → D7
- ECHO → D8
- VCC → 5V
- GND → GND

• Buzzer pasiv

- + → D9
- - → GND

• LED RGB

- conectat la pinii D3, D5 si D6 prin rezistente de 220 ohm
- pinul comun va fi conectat in functie de tipul LED-ului

• Fotorezistor

- conectat intr-un divizor de tensiune
- valoarea este citita pe pinul A0

- **Buton**

- conectat la D2 si GND
- citit folosind INPUT_PULLUP

Demonstrație video a funcționării hardware-ului:

Software Design

- **Repository GitHub:** <https://github.com/ionutmantu8/ErgoDeskAssistant>

- **Mediu de dezvoltare:** **Arduino IDE**, folosit pentru scrierea, compilarea si incarcarea firmware-ului pe placa **Arduino Uno**.

- **Tehnologii si surse folosite:**

1. **C/C++**
2. ``Arduino.h``
3. ``avr/io.h``
4. ``avr/interrupt.h``
5. ``util/twi.h``
6. ``math.h``
7. ``string.h``
8. Nu au fost folosite biblioteci externe dedicate pentru **LCD I2C** sau **MPU6050**; accesul la aceste componente a fost implementat manual, la nivel de registre.

- **Organizarea codului:**

1. ``ErgoDeskRegisters.ino`` - initializarea aplicatiei si bucla principala
2. ``config.h`` - constante, praguri, adrese I2C, timpi
3. ``app_types.h`` - structuri pentru starea aplicatiei, senzori si alerte
4. ``gpio.cpp`` - configurarea si accesul la pinii digitali
5. ``interrupts.cpp`` - intreruperea externa pentru buton
6. ``pwm.cpp`` - configurarea PWM hardware pentru LED-uri
7. ``adc_regs.cpp`` - citirea LDR-ului prin ADC
8. ``twi_regs.cpp`` - comunicatia I2C/TWI la nivel de registri
9. ``lcd_regs.cpp`` - controlul LCD-ului prin backpack I2C
10. ``mpu_regs.cpp`` - initializarea si citirea MPU6050
11. ``logic.cpp`` - logica principala a aplicatiei

- **Functionalitatea firmware-ului:**

1. citeste **MPU6050** pentru determinarea inclinarii/posturii
2. citeste **HC-SR04** pentru determinarea distantei fata de ecran
3. citeste **fotorezistorul** pentru estimarea luminii ambientale
4. calculeaza abaterea fata de postura calibrata

5. detecteaza:
 1. alerta de postura
 2. alerta de distanta
 3. nivel critic pentru fiecare
 4. caz combinat cand ambele sunt active
6. ofera feedback prin:
 1. **LCD**
 2. **LED-uri**
 3. **buzzer**

- **Algoritmi si mecanisme implementate:**

1. **calibrarea posturii** prin medierea mai multor mostre de la accelerometru
2. **debounce software** pentru buton
3. detectie **short press / long press**
4. activarea alertelor doar daca situatia persista un anumit timp
5. stabilizarea severitatii pentru LED-uri, pentru a evita schimbarile bruste
6. reglarea luminozitatii LED-urilor in functie de valoarea citita de la LDR

- **Concepte de laborator evidentiate in implementare:**

1. **GPIO** - configurarea butonului si a senzorului ultrasonic
2. **Intrepereri** - detectarea schimbarii starii butonului
3. **PWM / Timere** - controlul intensitatii LED-urilor
4. **ADC** - citirea fotorezistorului
5. **I2C / TWI** - comunicarea cu LCD-ul si cu MPU6050

Rezultate Obținute

- A fost realizata o aplicatie firmware functionala pentru monitorizarea posturii si a distantei fata de ecran. - Sistemul citeste corect datele de la:

1. **MPU6050** pentru inclinare
2. **HC-SR04** pentru distanta
3. **LDR** pentru nivelul de lumina ambientala

- A fost implementata **calibrarea posturii**, astfel incat sistemul sa foloseasca o pozitie de referinta adaptata utilizatorului. - Au fost implementate si testate mecanisme de alertare:

1. **vizuala** prin LCD si LED-uri
2. **sonora** prin buzzer

- Intensitatea LED-urilor este adaptata in functie de lumina ambientala citita de la fotorezistor. - Butonul permite:

1. comutarea intre modul de afisare a alertelor si modul de afisare a valorilor
2. recalibrarea posturii prin apasare lunga

- Codul a fost organizat modular, pe fisiere separate, pentru a separa clar logica aplicatiei de accesul la periferice.

Concluzii

Proiectul demonstreaza integrarea mai multor periferice hardware intr-o aplicatie embedded unitara si functionala. - Au fost aplicate practic concepte importante din laborator, precum **GPIO, intreruperi, PWM, ADC si I2C/TWI**. - Utilizarea accesului la registre pentru perifericele principale a permis o intelegere mai clara a modului in care microcontroller-ul controleaza direct componentele externe. - Implementarea modulara face codul mai usor de urmarit, testat si extins. - Sistemul obtinut poate fi dezvoltat in continuare prin adaugarea unor functionalitati suplimentare, de exemplu jurnalizarea datelor, conectivitate wireless sau statistici privind postura utilizatorului.

Jurnal

- 16.04.2026 - Alegerea temei proiectului si stabilirea directiei generale.
- 30.04.2026 - Achizitionarea componentelor care imi lipsesc.
- 05.05.2026 - Crearea paginii de documentatie pe wiki si completarea documentatiei pentru prima etapa.
- 12.05 - Hardware-ul proiectului a fost terminat
- 19.05 - Software-ul proiectului a fost terminat
- 23.05 - Update-ul final al documentatiei

Bibliografie/Resurse

- Laboratoarele de PM puse la dispozitie în cadrul disciplinei
- [ATmega328P Datasheet](#)
- [MPU6050 Register Map and Descriptions](#)
- [PCF8574 / PCF8574A Datasheet](#)
- [HC-SR04 Datasheet](#)

[Export to PDF](#)

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

Permanent link:
<https://ocw.cs.pub.ro/courses/pm/prj2026/victor.stoica0203/ergodeskassistant>



Last update: **2026/05/23 15:41**