

House Managing Tool

Introducere

Temă

Proiectul reprezintă un tool de managing al gadget-urilor dintr-o casă.

Descriere generală

Principalele funcționalități sunt:

1. Cu ajutorul a două butoane se poate regla intensitatea luminii din cameră (se crește sau se scade intensitatea unui LED).
2. Ușile se deschid automat când o persoană se apropie.
3. Display LCD cu 3 moduri:
 - primul: date despre cameră: temperatură, umiditate, lumină.
 - al doilea: mini-hartă cu fiecare cameră și locul unde se află o persoană în cameră (dau display la un dreptunghi cu dimensiunile camerei scalate și un punct roșu unde e o persoană). La început, când se pornește aplicația, acest senzor o să se rotească pentru a măsura lungimea și lățimea camerei.
 - al treilea: se poate seta o alarma care sună la ora respectivă (tot prin butoane se și selectează ora, ziua când se pune alarma).

Piese

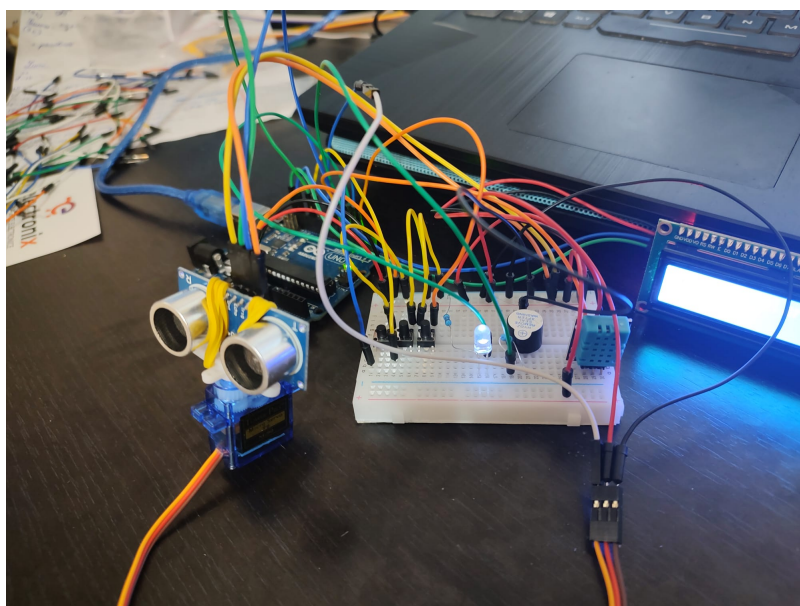
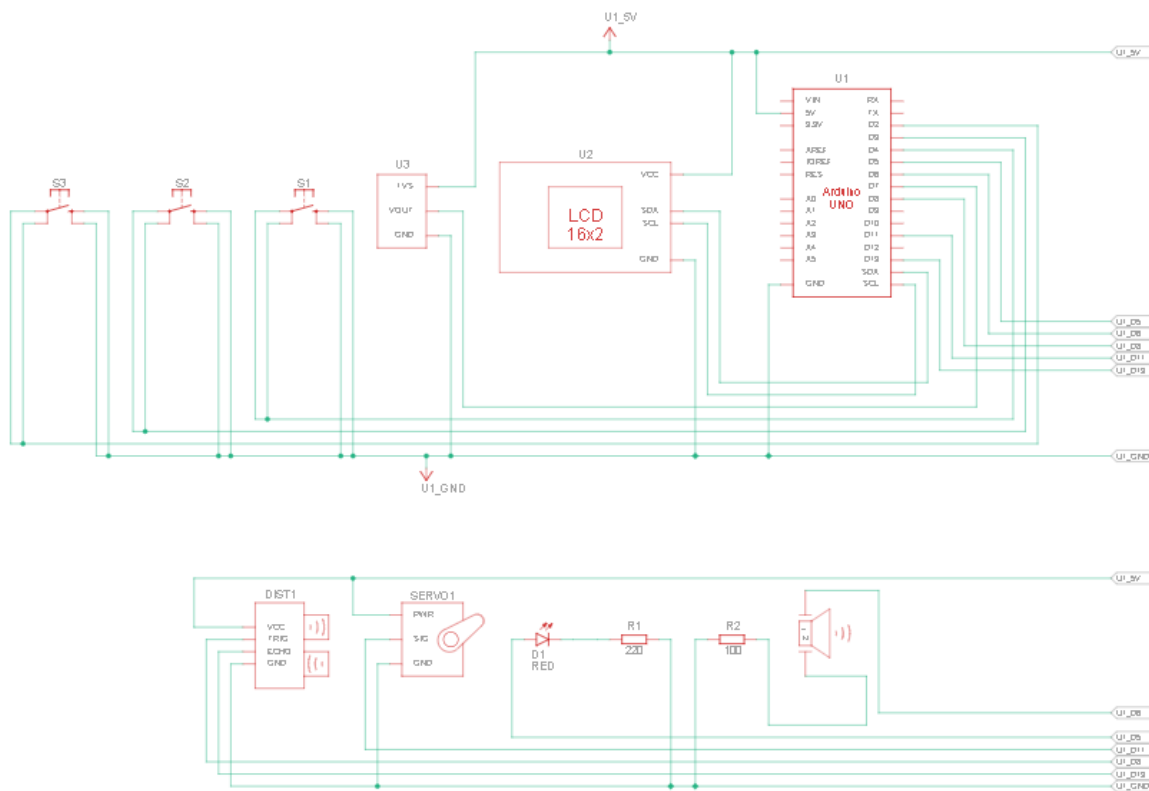
Lista cu piese necesare:

1. ARDUINO UNO R3 ATMEGA328P - <https://cleste.ro/arduino-uno-r3-atmega328p.html>
2. BUTON ROTUND X 2 - <https://cleste.ro/buton-rotund.html#/100-butonrotund-galben>
3. LED RGB 5MM 4PINI - https://cleste.ro/led-rgb-5mm-4pini.html#/94-ledrgb-anod_comun
4. Display LCD - <https://cleste.ro/ecran-lcd-1-8-spi.html>
5. senzor temperatura si umiditate - <https://cleste.ro/senzor-temperatura-si-umiditate-dht11.html>
6. senzor lumina - <https://cleste.ro/fotorezistor-5528-ldr.html>
7. senzor distanta - <https://cleste.ro/senzor-ultrasonic-hc-sr04.html>
8. servo motor - <https://cleste.ro/motor-servo-sg90-9g.html>
9. buzzer - <https://cleste.ro/modul-buzzer.html>

Schema Bloc



Hardware Design



Am folosit uo serie de senzori: DHT11 - pentru temperatura si umiditate, HRC-SR04 si VL53LOX. Am comunicat datele catre PC pentu a fi reprezentate prin interfata saeriala, mentionez ca acest lucru nu apare pe schema electrica.

Software Design

Mediu dezvoltare:

- Arduino IDE
- VScode

Biblioteci:

- Arduino: Wire, LiquidCrystal_I2C, PWMServo, DHT
- Python: serial, time, turtle (pentru desenat)

Functii:

- Python: calculate(d, alfa, s) → calculeaza unde trebuie reprezentat punctul in functie de distanta data de senzor si unghiul curent al motorului; write_read(msg) → scire un mesaj din python catre arduino si asteapta apoi sa primeasca date/raspuns de la arduino; restul sunt pentru reprezentare grafica → draw_rect(L), draw_point(t, x, y).
- Arduino: setAlarm() → seteaza o alarma, face initializarile necesare; increaseLightOrTimer() → functionalitatea/intreruperea butonului 1, in functie de modul curent; decreaseLightOrSetTimer() → functionalitatea butonului 2 in functie de modul curent; ISR(PCINT2_vect) → seteaza modul cu ajutorul butonului 0; base_setup() → face setup-ul initial, necesar tuturor modurilor; mode_0() → primul mod de functionare, display date, control lumina LED, transnitere date catre python pentru reprezentare obiecte; mode_1() → al doilea mod de functionare, reprezentare ora curenta, plus setare alarma.

Concluzii

Aplicatia realizeaza functiile prezentate cu succes, se pot observa informatiile casei, cat si 'harta' camerei. Se pot seta alarmele.

Am observat delay-uri in cadrul afisarilor si ca senzorul cu laser, realizeaza o harta mult mai precisa decat cel ultrasonic, desi range-ul este mai mic.

Download

[housemanagingtool.zip](#)

Bibliografie/Resurse

<https://ocw.cs.pub.ro/courses/pm/prj2023/ncaroi/house-managing-tool>

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2023/ncaroi/house-managing-tool>



Last update: **2023/05/28 16:07**