

# Home Monitoring

Nume: Răducanu George-Cristian

Grupă: 331CA

## Introducere

Facilitățile dispozitivului de Home Monitoring:

- măsurarea temperaturii din mediul ambiental
- stocarea datelor pentru statistici ulterioare
- detectarea intrușilor folosind un senzor de proximitate, dotat cu alarmă la alegere

Acest proiect răspunde nevoii de a avea un dispozitiv offline, cu autonomie mare ce poate realiza statistici simple de temperatura cu alarmă integrată. De asemenea dispozitivul Home Monitoring poate fi utilizat și pentru detectarea intrușilor. El poate fi utilizat atât la interior cât și la exterior, iar în cazul plasării în exterior este recomandată amplasarea într-o zonă aflată la umbra. Acest lucru este necesar întrucât măsurarea corectă a temperaturii în exterior se realizează la 1m deasupra solului în condiții de umbră.

## Descriere generală

Dispozitivul este controlat în mod unic printr-o telecomandă în infraroșu. Senzorul de proximitate va monitoriza zona de interes și în cazul detectiei de intruși declanșează o alarmă. Monitorizarea temperaturii se face la intervale periodice de timp și cu ajutorul telecomenzii se pot selecta și afișa statistici legate de temperatura înregistrată.



Dispozitivul are capacitate de stocare pe card SD limitată la valoarea maximă de 128GB (limitarea fiind determinată de adaptorul I2C al cardului). Măsurarea temperaturii este realizată cu ajutorul unui termistor, în detrimentul unui senzor de temperatură întrucât se dovedește a fi mult mai fiabil, ieftin și simplu de folosit decât un senzor de tipul DHT11.

## Hardware Design

Lista pieselor necesare:

- Arduino Uno R3
- Senzor ultrasonic
- Card SD + modul slot Card SD, SPI
- Buzzer piezo-electric
- Display LCD 1602 + adaptor pt I2C
- Thermistor
- Senzor infrarosu + telecomanda
- Rezistente
- LED-uri rosii
- baterie 9V
- breadboard

Schema de proiectare a dispozitivului în ThinkerCAD este afișată mai jos.

În lipsa unui modul SPI pentru cardul SD în softul de proiectare, am reprezentat cu un breadboard mic și chenar.

În cele ce urmează voi detalia conexiunile realizate cu Arduino Uno R3 pentru fiecare componentă și modul.



O detaliere a tuturor legaturilor realizate este descrisa in continuare.

Pentru displayul 1602 cu I2C:

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

Pentru buzzer:

- GND - GND
- (+) - 10

Pentru thermistor am utilizat pinul analogic A0.

Pentru modulul cu card SD ce foloseste SPI trebuie urmatoarele conexiuni:

- VCC-firul rosu
- GND-firul negru
- MISO-port 12 (fir mov)
- MOSI-port 11 (fir portocaliu)
- CS-port 10 (fir roz)
- SCK - port 13 (fir gri)

Pentru senzorul de disantant HC-SR04 am considerat utile urmatoarele conexiuni:

- VCC - 5V
- GND - GND
- EchoPin - 8

- TrigPin - 9

Pentru senzorul IR am realizat urmatoarele conexiuni:

- GND - GND
- VCC - 5V
- OUT - pin 6

Pentru componentele ce necesita protocoalele de comunicatie I2C si SPI sunt utilizate porturile specifice ale Arduino Uno R3.

Pentru celelalte componente se pot utiliza oricare dintre porturile Arduino cu mentiunea ca trebuie facuta conectarea corespunzator in functie de tip: analogic/digital.

## Software Design

Pentru dezvoltarea software a proiectului am utilizat ArduinoIDE 2.0.

Bibliotecile software third-party utilizate sunt:

- SPI - wrappere pentru comunicatia folosind SPI
- SD - biblioteca ce ofera functii de baza in file-systemul unui card SD
- LiquidCrystal\_I2C - driver pentru display-ul 1602 impreuna cu interfata I2C
- DistanceSensor - biblioteca ce ajuta in interactiunea cu senzorul de proximitate SR04
- IRremote - functii pentru ajutor in lucru cu senzorul si telecomanda IR

Pentru inregistrarea temperaturii, datorita utilizarii unui thermistor si nu a unui senzor digital a fost necesara implementarea unei functii separate.

Pentru aceasta trebuie utilizata ecuatia Steinhart-Hart, a carei modelare este descrisa mai jos:

$$\frac{1}{T} = A + B \ln(R) + C (\ln(R))^3$$



Determinarea temperaturii se realizeaza folosind varianta simplificata:  $\frac{1}{T} = \frac{1}{T_0} + B \ln\left(\ln\left(\frac{R}{R_0}\right)\right)$

În dezvoltarea proiectului au fost de asemenea utilizate și timere pentru o funcționare fără busy-waiting.

## Rezultate Obținute

Rezultatele obținute sunt prezentate în linkul de youtube de mai jos

[Demo](#)

## Concluzii

Dezvoltarea de dispozitive și device-uri se dovedește a fi mult mai provocatoare față de dezvoltarea clasică software.

O mare provocare o constă autonomia dispozitivelor. În linkul de mai jos se regăsește o estimare a autonomiei folosind datasheet-uri și măsurători.

### [Estimare consum și autonomie](#)

S-a considerat ca setările userului sunt realizate rar (4 min pe zi). Marii consumatori sunt display-ul 1602, placa Arduino Uno și modulul de card SD (depinzând în funcție de încărcare și viteză de scriere).

Acest lucru este datorat complexității îmbinării hardware-ului cu software-ul. De asemenea o provocare mare este necesitatea asigurării unei autonomii, în cazul produselor alimentate din baterii.

## Download

Download Github:

<https://github.com/GeorgeRaducanu/ProiectPM.git>

## Jurnal

- 20 Aprilie - achiziție piese
- 22-26 Aprilie - confirmare finală proiect
- 06-10 Mai - început asamblare și descoperit că termistorul este o soluție mai fiabilă decât DHT11
- 13-15 Mai - finalizat hardware și început software-ul
- 15-22 Mai - finalizat software + testare
- 22-27 Mai - încercare de înfrumusețare a prototipului

## Bibliografie/Resurse

Ghid Plusivo SR04 & IR remote

[https://en.wikipedia.org/wiki/Steinhart%E2%80%93Hart\\_equation](https://en.wikipedia.org/wiki/Steinhart%E2%80%93Hart_equation)

<https://www.northstarsensors.com/calculating-temperature-from-resistance>

<https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>

<https://www.arduino.cc/reference/en/libraries/hcsr04-ultrasonic-sensor/>

[https://projecthub.arduino.cc/Marcazzan\\_M/how-easy-is-it-to-use-a-thermistor-81ae74](https://projecthub.arduino.cc/Marcazzan_M/how-easy-is-it-to-use-a-thermistor-81ae74)

Export to PDF

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/rrusu/george.raducanu1811>



Last update: **2024/05/26 11:08**