

Smart Home Watch

Autor: Rusu Andrei-Alexandru

Grupa: 333CA

Introducere

Smart Home Watch este un ceas smart de interior pentru casă, construit pe o placă compatibilă Arduino UNO. Proiectul afișează ora, data, temperatura și umiditatea din cameră pe un ecran LCD1602 cu interfață I2C și include o funcție de alarmă semnalizată prin buzzer și LED RGB.

Scopul proiectului este realizarea unui dispozitiv compact și util pentru monitorizarea rapidă a condițiilor din locuință. Ideea de la care am pornit a fost combinarea unui ceas digital cu un mic sistem de monitorizare ambientală, astfel încât utilizatorul să poată vedea într-un singur loc informațiile importante din cameră.

Proiectul este util deoarece poate fi folosit ca ceas de birou/noptieră, indicator de temperatură și umiditate și alarmă. Pentru mine, proiectul este util și ca exercițiu practic de integrare hardware-software: comunicație I2C, citirea senzorilor, afișarea datelor, controlul unor ieșiri digitale, folosirea PWM-ului și lucrul direct cu registrele microcontrollerului.

Descriere generală

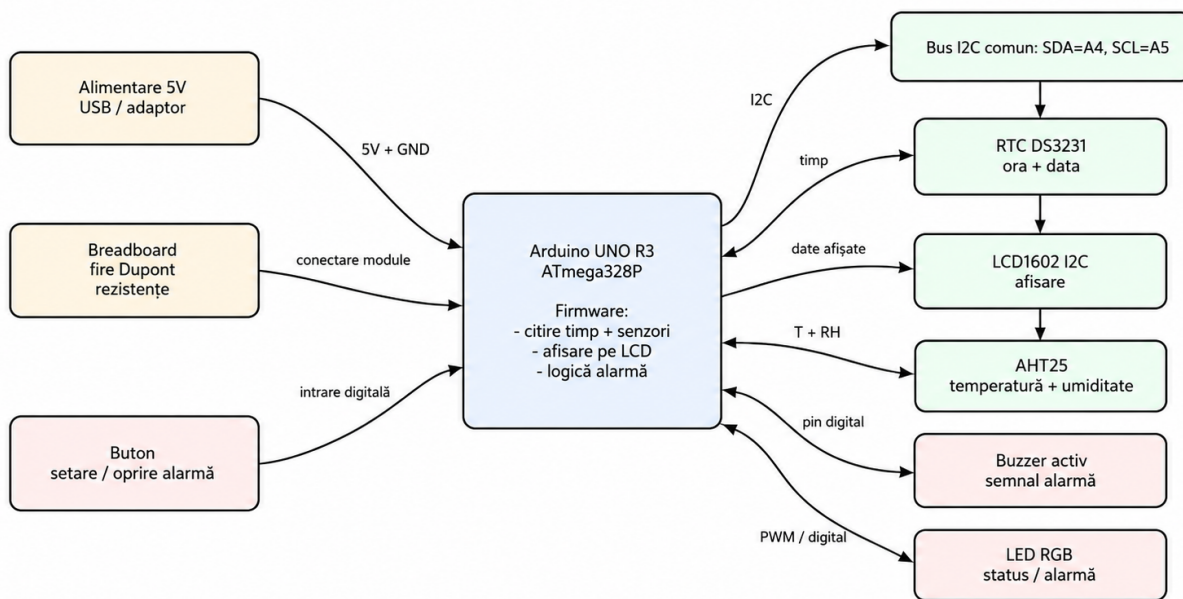
Sistemul este organizat în jurul plăcii compatibile Arduino UNO R3, care controlează toate modulele hardware. Modulul RTC DS3231 oferă ora și data exactă, senzorul AHT25 măsoară temperatura și umiditatea, iar ecranul LCD1602 afișează informațiile pentru utilizator. Pentru funcția de alarmă, Arduino controlează un buzzer activ și un LED RGB.

Cele trei butoane sunt folosite pentru interacțiuni: schimbarea ecranului afișat, activarea/dezactivarea alarmei, setarea orei și minutului alarmei și oprirea acesteia. Butonul MODE schimbă ecranul curent sau scade valoarea în modul de setare, butonul NEXT activează/dezactivează alarma sau crește valoarea în modul de setare, iar butonul OK confirmă setările sau oprește alarma.

RTC-ul DS3231, ecranul LCD1602 și senzorul AHT25 comunică prin I2C, deci folosesc aceleași linii SDA și SCL ale plăcii Arduino. Buzzerul este controlat printr-un semnal PWM generat cu Timer1, LED-ul RGB este controlat prin PWM, iar butoanele folosesc pini digitali cu rezistențe interne de pull-up.

Schema bloc

Schema bloc - Smart Home Watch



Ceas smart de interior: ora, data, temperatura, umiditate, alarmă și feedback vizual/sonor

Hardware Design

BOM - Bill of Materials

Componentă	Cantitate	Link	Preț unitar (RON)
Placă dezvoltare UNO R3 compatibilă Arduino, ATmega328P, CH340G	1	Link	30.93
Modul RTC DS3231 compatibil Arduino/Micro:Bit	1	Link	33.88
Ecran LCD1602 cu modul I2C/IIC	1	Link	24.99
Senzor temperatură și umiditate AHT25	1	Link	20.99
Modul buzzer activ compatibil Arduino	1	Link	3.24
Set 180 mini butoane switch	1	Link	30.49
Modul LED RGB 3 culori	1	Link	2.13
Breadboard 400 puncte	1	Link	4.56
Set fire Dupont mamă-tată 40p, 30 cm	1	Link	9.99
Set fire Dupont tată-tată 40p, 10 cm	1	Link	4.99
Preț total componente			166.19 RON

Funcționalitatea componentelor

Componentă	Rol în proiect
------------	----------------

Placă Arduino UNO compatibilă	Controlează toate modulele hardware și rulează logica proiectului
RTC DS3231	Furnizează ora și data curentă
LCD1602 I2C	Afișează ora, data, temperatura, umiditatea și meniul alarmei
AHT25	Măsoară temperatura și umiditatea din cameră
Buzzer activ	Semnalizează sonor alarma și oferă feedback la apăsarea anumitor butoane
LED RGB	Oferă feedback vizual pentru starea alarmei și pulsează roșu când alarma sună
Butoane	Permit navigarea prin ecrane, setarea alarmei și oprirea acesteia
Breadboard și fire Dupont	Permit realizarea conexiunilor între module fără lipire permanentă

Schema electrică

Schema electrică prezintă conexiunile dintre placa Arduino UNO R3 și modulele folosite în proiect. LCD-ul 1602 cu adaptor I2C, modulul RTC DS3231 și senzorul AHT25 sunt conectate pe aceeași magistrală I2C. Acestea folosesc liniile SDA și SCL ale plăcii Arduino, împreună cu alimentarea comună de 5V și GND.

Buzzerul activ este conectat pe pinul D9, corespunzător ieșirii OC1A a Timer1, pentru a putea genera semnal PWM prin configurarea directă a registrelor. LED-ul RGB este conectat pe pini PWM: D5 pentru roșu, D6 pentru verde și D11 pentru albastru. Cele trei butoane sunt conectate la pini digitali și la GND, folosind rezistențele interne de pull-up ale microcontrollerului.



Pinout

Componentă	Pin componentă	Pin Arduino / conexiune	Rol
LCD1602 I2C	VCC	5V	Alimentare display
LCD1602 I2C	GND	GND	Masă
LCD1602 I2C	SDA	SDA / A4	Date I2C
LCD1602 I2C	SCL	SCL / A5	Clock I2C
RTC DS3231	VCC	5V	Alimentare modul RTC
RTC DS3231	GND	GND	Masă
RTC DS3231	SDA	SDA / A4	Date I2C
RTC DS3231	SCL	SCL / A5	Clock I2C
AHT25	VCC / VIN	5V	Alimentare senzor
AHT25	GND	GND	Masă
AHT25	SDA	SDA / A4	Date I2C
AHT25	SCL	SCL / A5	Clock I2C
Buzzer activ	S / I/O	D9 / OC1A	Semnal PWM pentru alarmă sonoră
Buzzer activ	+	5V	Alimentare buzzer
Buzzer activ	-	GND	Masă
LED RGB	R / RED	D5 / OC0B	Canal roșu, PWM

LED RGB	G / GREEN	D6 / OC0A	Canal verde, PWM
LED RGB	B / BLUE	D11 / OC2A	Canal albastru, PWM
LED RGB	GND / -	GND	Masă
Buton MODE / MINUS	pin 1	D2 / INT0	Schimbare mod / scădere valoare
Buton MODE / MINUS	pin 2	GND	Intrare cu pull-up intern
Buton NEXT / PLUS	pin 1	D3 / INT1	Navigare / creștere valoare
Buton NEXT / PLUS	pin 2	GND	Intrare cu pull-up intern
Buton OK / STOP	pin 1	D4	Confirmare / oprire alarmă
Buton OK / STOP	pin 2	GND	Intrare cu pull-up intern

Software Design

Mediu de dezvoltare: PlatformIO, folosind framework-ul Arduino pentru placa Arduino UNO compatibilă cu ATmega328P.

Biblioteci utilizate:

- **Wire:** biblioteka standard pentru comunicația I2C.
- **LiquidCrystal_I2C:** folosită pentru controlul ecranului LCD1602 prin adaptorul I2C.
- **RTCLib:** folosită pentru comunicarea cu modulul RTC DS3231 și citirea orei/datei.
- **Adafruit_AHTX0:** folosită pentru citirea temperaturii și umidității de la senzorul AHT25.

Funcționalitate software

Programul funcționează ca un sistem cu mai multe stări, fiecare stare reprezentând un ecran sau o funcție a ceasului:

- **Ecran ora/data:** afișează ora curentă și data citite din RTC DS3231.
- **Ecran temperatură/umiditate:** afișează valorile citite de la senzorul AHT25.
- **Ecran alarmă:** afișează ora alarmei și starea acesteia: ON/OFF.
- **Setare oră alarmă:** permite modificarea orei alarmei.
- **Setare minute alarmă:** permite modificarea minutelor alarmei.
- **Alarmă activă:** buzzerul sună intermitent, LED-ul RGB pulsează roșu, iar alarma poate fi oprită prin apăsarea oricărui buton.

Cele trei butoane au roluri diferite în funcție de starea curentă:

- **MODE / MINUS:** schimbă ecranul afișat, iar în modul de setare scade ora sau minutul.
- **NEXT / PLUS:** activează/dezactivează alarma, iar în modul de setare crește ora sau minutul.
- **OK / STOP:** intră în setarea alarmei, confirmă valorile și oprește alarma atunci când aceasta sună.

Pentru a evita declanșarea accidentală a alarmei în timpul modificării acesteia, programul ignoră verificarea alarmei cât timp utilizatorul se află în ecranele de setare a orei sau minutului. De asemenea, când alarma este salvată, declanșarea este blocată pentru minutul curent, astfel încât alarma să nu pornească instant dacă a fost setată exact la ora și minutul curent.

Utilizarea noțiunilor din laborator

Proiectul include noțiuni din mai multe laboratoare PM:

- **I2C/TWI:** LCD1602 I2C, RTC DS3231 și AHT25 folosesc aceeași magistrală I2C, conectată la SDA/A4 și SCL/A5.
- **PWM/Timere:** buzzerul este controlat cu Timer1 în Fast PWM Mode 14, cu TOP în ICR1 și prescaler 8. LED-ul RGB este controlat prin PWM folosind registrele timerelor asociate pinilor D5, D6 și D11.
- **Înteruperi:** butoanele MODE și NEXT folosesc întreruperi externe pe INT0 și INT1, configurate prin registre.
- **GPIO:** butoanele sunt configurate prin registrele DDRD și PORTD, folosind rezistențele interne de pull-up.
- **UART:** Serial Monitor este folosit pentru debugging, afișând ora, data, temperatura, umiditatea, starea alarmei și ecranul curent.

Structura codului

Codul este împărțit în mai multe zone logice:

- **inițializarea modulelor:** configurarea comunicației I2C, inițializarea LCD-ului, RTC-ului și senzorului AHT25;
- **configurarea registrelor:** setarea Timer1 pentru buzzer, setarea PWM-ului pentru LED RGB, configurarea butoanelor și a întreruperilor externe;
- **funcțiile de afișare:** actualizarea celor două linii ale LCD-ului în funcție de ecranul curent;
- **logica alarmei:** verificarea orei curente, pornirea alarmei, oprirea alarmei și semnalizarea audio-vizuală;
- **logica butoanelor:** navigare între ecrane, setarea orei/minutului și confirmarea setărilor;
- **debugging UART:** transmiterea periodică a stării sistemului către Serial Monitor.

Detalii de implementare

Pentru buzzer, semnalul sonor este generat pe pinul D9 folosind Timer1 în Fast PWM Mode 14. Registrul ICR1 stabilește valoarea TOP, iar OCR1A stabilește duty cycle-ul. Pentru frecvența de aproximativ 2 kHz și prescaler 8, valoarea ICR1 este calculată folosind formula:

- $f_{PWM} = F_{CPU} / (\text{prescaler} * (1 + ICR1))$

Pentru LED-ul RGB, valorile de intensitate sunt scrise direct în registrele OCR0A, OCR0B și OCR2A. În starea normală, LED-ul indică dacă alarma este activă sau dezactivată, iar în starea de alarmă acesta pulsează roșu.

Pentru butoane, pinii D2, D3 și D4 sunt configurați ca intrări cu pull-up intern. Butoanele MODE și NEXT folosesc întreruperi externe, iar butonul OK este citit prin polling și debounce software. În acest mod, sistemul poate reacționa rapid la comenzile utilizatorului fără a bloca execuția programului.

Rezultate Obținute

Video Proiect

În urma implementării, proiectul Smart Home Watch funcționează ca un ceas de interior care afișează ora, data, temperatura și umiditatea din cameră. Utilizatorul poate naviga între ecrane folosind butoanele, poate activa sau dezactiva alarma și poate seta ora și minutul acesteia.

Când alarma este declanșată, buzzerul emite un semnal sonor intermitent, iar LED-ul RGB pulsează roșu. Alarma poate fi oprită prin apăsarea oricărui buton. Valorile citite de la senzor și starea sistemului sunt transmise și în Serial Monitor pentru verificare și debugging.

Funcționalitățile demonstrate în video sunt:

- afișarea orei și datei pe LCD;
- afișarea temperaturii și umidității;
- navigarea între ecrane cu ajutorul butoanelor;
- setarea alarmei;
- feedback sonor prin buzzer;
- feedback vizual prin LED RGB;
- oprirea alarmei din butoane.

Concluzii

Proiectul Smart Home Watch a demonstrat integrarea mai multor componente hardware într-un sistem embedded funcțional, bazat pe o placă compatibilă Arduino UNO. Dispozitivul combină funcțiile unui ceas digital cu monitorizarea temperaturii și umidității și cu o alarmă configurabilă.

O parte importantă a proiectului a fost folosirea magistralei I2C pentru conectarea mai multor module pe aceleași două fire de comunicație. Acest lucru a simplificat cablarea și a făcut sistemul mai ușor de extins. De asemenea, proiectul a permis utilizarea practică a noțiunilor de PWM, timere, întreruperi externe, GPIO și UART.

Prin folosirea directă a registrelor pentru Timer1, LED RGB, butoane și întreruperi, proiectul nu se bazează doar pe funcțiile standard Arduino, ci demonstrează și înțelegerea modului în care perifericele microcontrollerului ATmega328P sunt configurate la nivel low-level. Bibliotecile au fost păstrate pentru componentele I2C mai complexe, precum LCD-ul, RTC-ul și senzorul AHT25, deoarece acestea oferă o interfață stabilă și clară pentru dispozitivele respective.

În final, proiectul este funcțional și poate fi folosit ca un ceas de birou/noptieră cu monitorizare ambientală și alarmă.

Download

- [Cod sursă Smart Home Watch](#)

Jurnal

- 27.04.2026 - Alegerea temei proiectului și stabilirea ideii de Smart Home Watch
- 29.04.2026 - Documentarea componentelor necesare și alegerea modulelor I2C
- 06.05.2026 - Crearea paginii OCW pentru proiect
- 08.05.2026 - Realizarea descrierii generale și a listei inițiale de componente
- 11.05.2026 - Comandarea componentelor hardware necesare
- 14.05.2026 - Realizarea schemei bloc, schemei electrice și pinout-ului
- 17.05.2026 - Testarea comunicației I2C cu LCD-ul, RTC-ul și senzorul AHT25
- 19.05.2026 - Implementarea funcției de alarmă, a butoanelor și a feedback-ului prin buzzer și LED RGB
- 21.05.2026 - Integrarea PWM-ului pentru buzzer și LED RGB și configurarea unor periferice direct prin registre
- 23.05.2026 - Testarea finală pe hardware, încărcarea videoclipului demonstrativ și finalizarea paginii OCW

Bibliografie/Resurse

Resurse Hardware

- Placă UNO R3 compatibilă Arduino, ATmega328P, CH340G - Sigmanortec: <https://sigmanortec.ro/Placa-dezvoltare-UNO-R3-Arduino-Compatibil-ATmega328p-CH340G-cu-bara-pini-p170362384>
- Modul RTC DS3231, compatibil Arduino/Micro:Bit, Keyestudio - Bitmi: <https://www.bitmi.ro/electronica/modul-rtc-ds3231-compatibil-arduino-si-micro-bit-keyestudio-11162.html>
- Ecran LCD1602 cu modul I2C/IIC - Bitmi: <https://www.bitmi.ro/electronica/ecran-lcd1602-cu-modul-i2c-iic-10487.html>
- Senzor temperatură și umiditate AHT25 - Bitmi: <https://www.bitmi.ro/electronica/senzor-de-temperatura-si-umiditate-aht25-10684.html>
- Modul buzzer activ compatibil Arduino - Bitmi: <https://www.bitmi.ro/electronica/modul-buzzer-activ-compatibil-arduino-10397.html>
- Set mini butoane switch - Bitmi: <https://www.bitmi.ro/componente-electronice/set-180-mini-butoane-switch-10523.html>
- Modul LED RGB 3 culori - Bitmi: <https://www.bitmi.ro/electronica/modul-led-rgb-3-culori-10401.html>

Resurse Software și Documentație

Last update:
2026/05/24
16:39

pm:prj2026:victor.stoica0203:andrei.rusu1504 <http://ocw.cs.pub.ro/courses/pm/prj2026/victor.stoica0203/andrei.rusu1504>

- Laboratoarele PM
- [Datasheet ATmega328P](#)
- [RTCLib](#)
- [Adafruit AHTX0](#)
- [LiquidCrystal_I2C](#)
- [Arduino Serial](#)
- [Arduino Wire](#)

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2026/victor.stoica0203/andrei.rusu1504>



Last update: **2026/05/24 16:39**