

# Clock Set Data Time

**Autor:** Asmaa Alaghbari

**Grupa:** 336CB

## Introducere



Odată cu dezvoltarea tehnologiei, gadget-urile și aparatele electronice devin din ce în ce mai sofisticate, iar multe dintre acestea au integrat un ceas sau o alarmă. În acest proiect, vom explora cum putem să creăm propriul ceas digital cu funcție de setare a orei și datei, dar și cu alarmă integrată. Pentru aceasta, vom utiliza un circuit integrat RTC (Real Time Clock), microcontrolerul Arduino și alte componente electronice necesare. Scopul acestui proiect este să oferim o soluție practică și utilă pentru oricine dorește să își creeze propriul ceas cu alarmă, sau să înțeleagă mai bine cum funcționează un astfel de dispozitiv.

## Descriere generală

Scopul acestui proiect este de a construi un ceas digital cu alarmă utilizând un circuit integrat RTC (Real Time Clock) și un microcontroler Arduino. Dispozitivul va afișa ora și data curente, iar utilizatorii vor putea seta ora și data în mod manual. De asemenea, va exista și o funcție de alarmă, care va activa un buzzer la momentul setat.

Proiectul implică utilizarea unui circuit integrat RTC, care este proiectat pentru a menține timpul real, chiar și în cazul întreruperilor de alimentare. RTC-ul va fi conectat la microcontrolerul Arduino, care va fi programat să afișeze ora și data curente pe un afișaj LCD. Utilizatorii vor putea seta manual ora și data folosind un buton și o roată de potențiomtru.

Pentru funcția de alarmă, se va utiliza un buzzer, care va fi activat la momentul setat. Utilizatorii vor putea seta momentul alarmei folosind butonul și potențiomtrul.

Proiectul va fi construit utilizând componente electronice comune, cum ar fi breadboard-ul, firele de conexiune, circuitul integrat RTC, afișajul LCD și buzzerul. De asemenea, va fi necesar să se programeze microcontrolerul Arduino utilizând Arduino IDE.

În general, acest proiect este o soluție practică și utilă pentru oricine dorește să își construiască propriul ceas digital cu alarmă, sau să înțeleagă mai bine cum funcționează un astfel de dispozitiv.

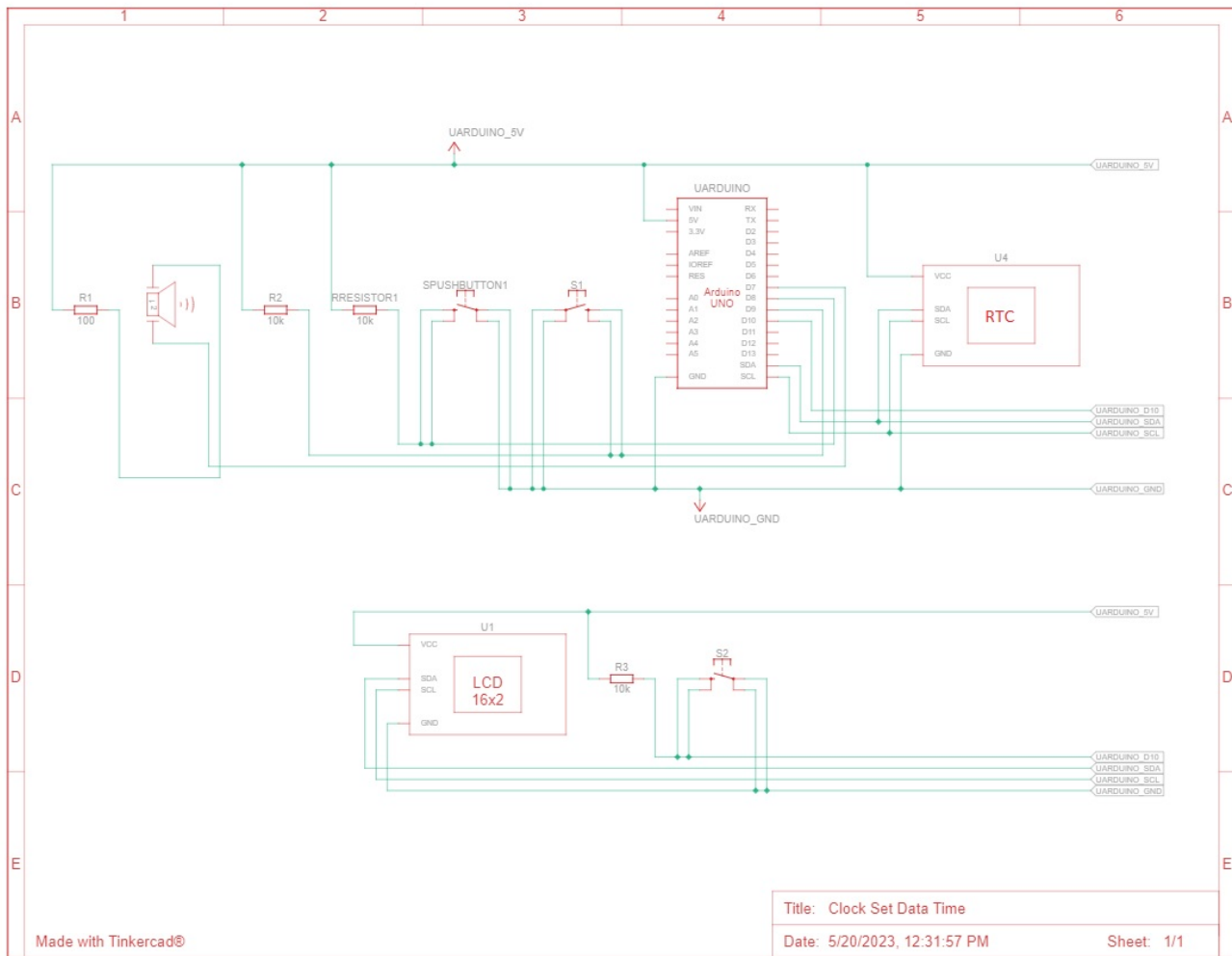
## Hardware Design



Un proiect complet pentru Clock Set Data Time și Alarm ar implica proiectarea și construirea unui circuit integrat care să aibă capacitatea de a afișa ora curentă, data și să genereze o alarmă atunci când este setată la un anumit moment de timp. Acest circuit ar avea nevoie de următoarele componente:

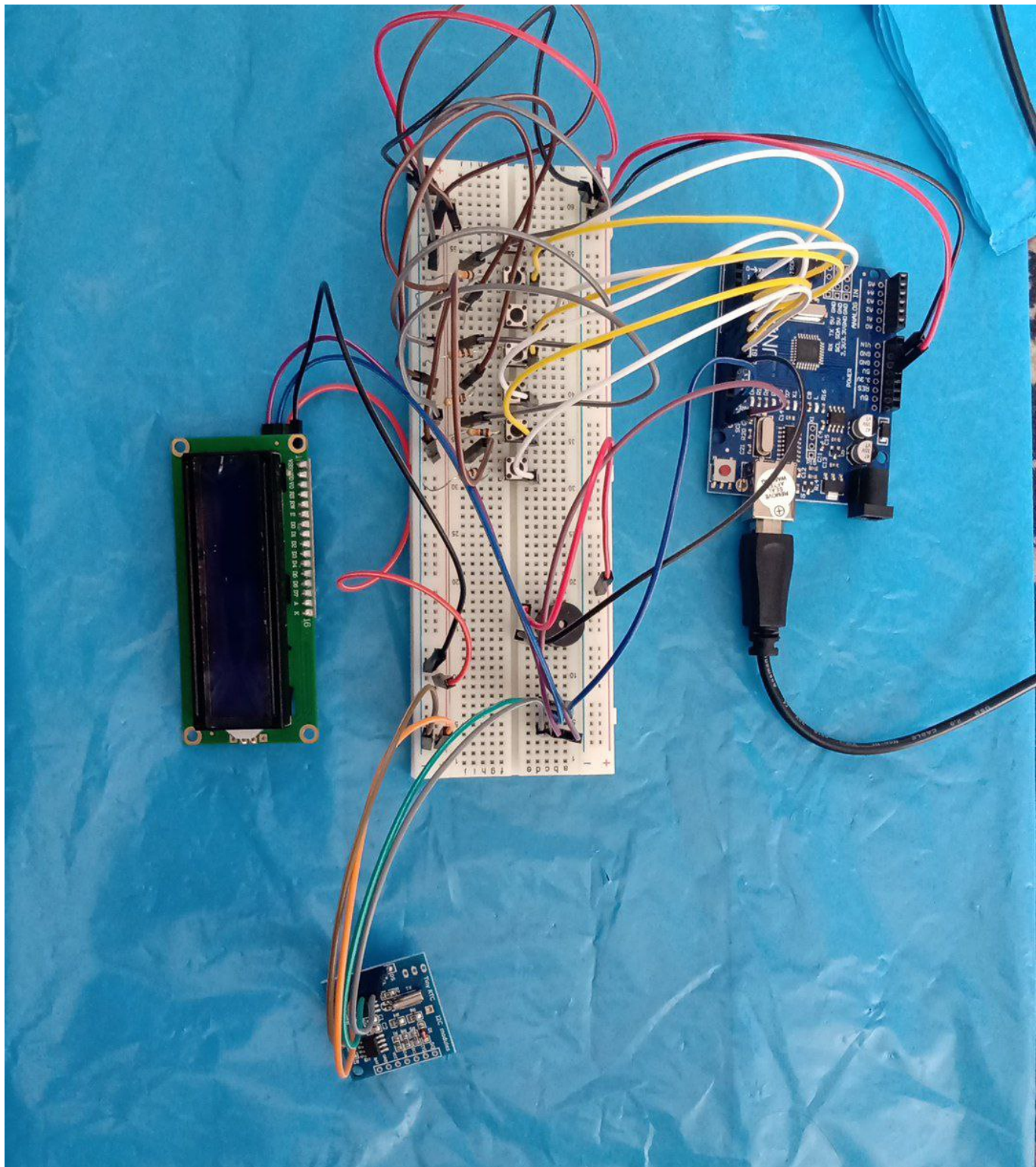
- Microcontroler - Pentru a controla afișajul, setarea timpului și alarmei, și pentru a genera sunetul de alarmă, se utilizează un microcontroler (Arduino Uno)
- Afișaj - Pentru a afișa ora și data, se utilizează un afișaj LCD.
- Buton - Un buton poate fi folosit pentru a seta timpul și data, precum și pentru a activa sau dezactiva alarmele.
- Buzzer - Pentru a genera sunetul de alarmă, se poate utiliza un buzzer.
- RTC (Real Time Clock) - Pentru a menține ora și data exactă, se poate utiliza un RTC, cum ar fi modelul DS1307.

### Schemă electrică:



### Implementarea circuitului:

Circuitul poate fi construit folosind un breadboard și cabluri de legare, împreună cu componentele menționate mai sus. După ce circuitul este construit, microcontrolerul trebuie programat pentru a gestiona RTC-ul și afișajul, precum și pentru a activa buzzer-ul la ora setată pentru alarmă.

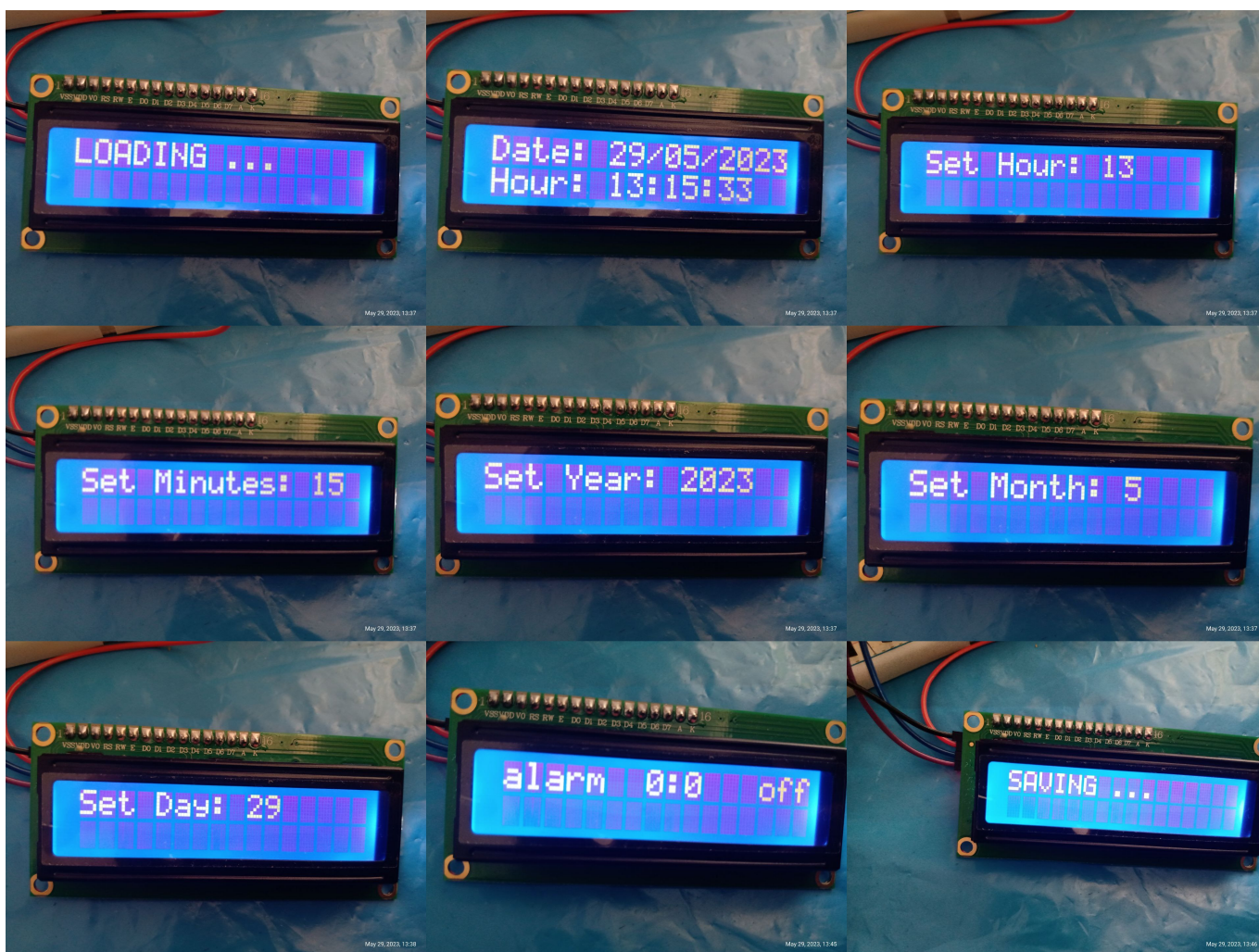


## Software Design

Codul furnizat este o implementare a unui ceas cu alarmă utilizând un Arduino și un modul RTC (Real-Time Clock). Acesta folosește o serie de biblioteci precum **Wire.h**, **RTCLib.h**, **TimeLib.h** și **LiquidCrystal\_I2C.h** pentru a permite comunicarea cu modulul RTC și afișarea datelor pe un afișaj LCD. Codul este structurat în jurul unei bucle principale **loop()**, care se execută în mod continuu. În această buclă, se verifică valoarea variabilei **menu** pentru a determina starea curentă a programului și se apelează funcțiile corespunzătoare pentru a afișa și gestiona diferitele aspecte ale ceasului.

De exemplu, funcția **DisplayDateTime()** afișează data și ora curentă pe afișajul LCD. Utilizatorul poate apăsa butonul "SET" pentru a trece în modul de setare a orei, minutei, anului, lunii sau zilei, iar funcțiile corespunzătoare precum **DisplaySetHour()** sau **DisplaySetMinute()** permit modificarea valorilor corespunzătoare. Există, de asemenea, funcția **DisplaySetAlarm()** care permite setarea alarmei. Utilizatorul poate ajusta ora și minutul alarmei utilizând butoanele specifice, iar starea alarmei (pornită sau oprită) poate fi modificată prin apăsarea butonului "ALARM ON/OFF". În funcția **Saving()**, valorile actualizate ale datei și orei sunt salvate în modulul RTC, iar utilizatorul primește un mesaj de confirmare pe afișaj. Codul conține, de asemenea, utilizarea funcției **tone()** pentru a genera un sunet de alarmă dacă ora curentă se potrivește cu ora și minutul setate pentru alarmă.

## Rezultate Obținute



Utilizatorul poate seta ora, minutul, anul, luna, ziua și alarma folosind butoanele disponibile. Rezultatele obținute prin utilizarea acestui cod includ:

1. Afișarea curentă a datei și a orei pe ecranul LCD.
2. Posibilitatea utilizatorului de a seta ora curentă prin intermediul butoanelor.
3. Posibilitatea utilizatorului de a seta minutul curent prin intermediul butoanelor.
4. Posibilitatea utilizatorului de a seta anul curent prin intermediul butoanelor.
5. Posibilitatea utilizatorului de a seta luna curentă prin intermediul butoanelor.
6. Posibilitatea utilizatorului de a seta ziua curentă prin intermediul butoanelor.
7. Posibilitatea utilizatorului de a seta alarmă și de a o activa/dezactiva prin intermediul butoanelor.

## 8. Salvarea valorilor actualizate ale datei și orei în modul RTC.

Astfel, prin utilizarea acestui cod, utilizatorul poate ajusta și menține data și ora corectă pe ceas, precum și să configureze o alarmă pentru a fi activată la o anumită oră și minut specificate.

## Concluzii

proiectul Clock Set Data Time și Alarma este o modalitate excelentă de a învăța cum să utilizați un circuit integrat RTC și un afișaj LCD pentru a construi un ceas digital cu alarmă. Prin intermediul acestui proiect, puteți învăța cum să utilizați funcțiile de programare ale Arduino și să vă familiarizați cu procesul de conectare a componentelor hardware la o placă de dezvoltare.

În concluzie, codul furnizat permite implementarea unui ceas digital cu funcționalități de afișare a datei și orei, setare a valorilor pentru oră, minut, an, lună, zi și activare/dezactivare a alarmei. Acesta oferă o interfață simplă și intuitivă utilizatorului prin intermediul butoanelor, permițându-i să ajusteze și să mențină corectitudinea datelor și orei afișate. Prin adăugarea funcționalității de alarmă, utilizatorul poate programa un semnal sonor la o oră și minut dorite. Codul reprezintă o bază solidă pentru dezvoltarea unui ceas digital personalizat și poate fi adaptat și extins pentru a îndeplini cerințe specifice.

Proiectul poate fi, de asemenea, util în viața de zi cu zi, fiind un instrument util pentru monitorizarea timpului și setarea alarmelor pentru diferite activități. Prin personalizarea și adaptarea proiectului la nevoile dvs., puteți obține un dispozitiv personalizat și util, care poate fi utilizat în diferite situații.

## Download

Acest fișier [pm\\_clock\\_set\\_data\\_time\\_-\\_files.zip](#) conține toate resursele necesare pentru proiectul "Clock with Alarm" pe platforma Arduino. Fișierul include următoarele elemente:

1. Codul sursa: **PM\_Software.ino**: Este fișierul principal al proiectului care conține codul sursa pentru implementarea unui ceas cu funcție de alarma. Acesta poate fi deschis și editat cu Arduino IDE. 2.

Biblioteci:

1. RTCLib: O bibliotecă pentru a permite comunicarea cu modulul RTC (Real-Time Clock).
2. LiquidCrystal\_I2C: O bibliotecă pentru controlul display-ului LCD (I2C).
3. Alte biblioteci specifice pot fi incluse în fișierul ZIP, dacă sunt folosite în proiect.

3. README.md

4. Imagini: conține imagini relevante pentru proiect, cum ar fi scheme de conexiune sau capturi de ecran ale aplicației.

## Jurnal

- 19.04.2023: Alegere tema proiect

- 02.05.2023: Realizarea paginii proiectului și a descrierii succinte a acestuia.
- 04.05.2023: Adaugare schema bloc
- 05.05.2023: Finalizare Milestone proiect - documentație
- 14.05.2023: Finalizare Milestone - hardware
- 28.05.2023: Finalizare Milestone - Software

## Bibliografie/Resurse

- [Clock Set Data Time](#)
- [PM Lab 1](#)
- [PM Lab 3](#)
- [PM Lab 5](#)
- [Simple Arduino Clock](#)
- [Tinkercad](#)
- [Clock Set Data Time - PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/iotelea/clocksetdatatime>



Last update: **2023/05/30 17:36**