

Termostat

Autor: [Radu Cristina-Andreea](#)

Grupa: 336CC

Introducere

Scopul acestei aplicații este de a realiza un circuit electronic care să calculeze o temperatură dată ca input și să mențină această temperatură prin pornirea și oprirea unui ventilator. Pentru a identifica cazul în care se află temperatura voi folosi semnalizarea printr-un LED RGB.

Descriere generală

Principiul de funcționare a circuitului este următorul:

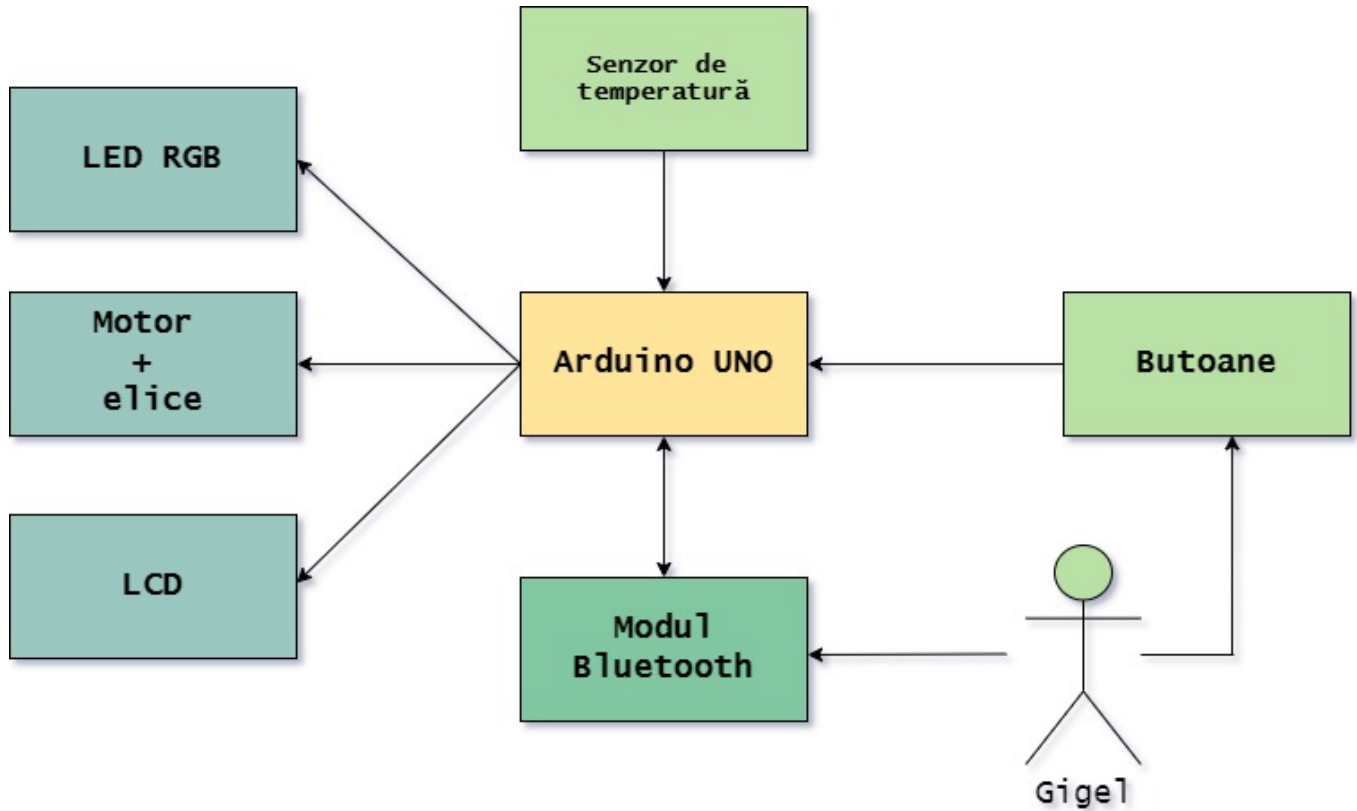
Pentru deblocarea termostatului este necesară conectarea prin Bluetooth de la telefon prin intermediul unei aplicații - BT Terminal. Folosesc această aplicație în loc de Serial Monitor. După conectarea la termostat, se introduce valoarea 1 pentru pornirea circuitului, iar la final, pentru a-l închide se folosește valoarea 0.

Odată pornit circuitul, sunt activate celelalte componente ale circuitului. În continuare, se setează temperatura folosind 2 butoane (unul pentru a crește, iar celălalt pentru a micșora temperatura).

Culorile LED-ului RGB:

- Verde indică faptul că temperatura de la senzor diferă cu cel mult 1 grad față de temperatura de la input.
- Roșu indică faptul că temperatura de la senzor este mai mare decât cea primită ca input.
- Albastru indică faptul că temperatura de la senzor este mai mică decât cea primită ca input.

Schema bloc



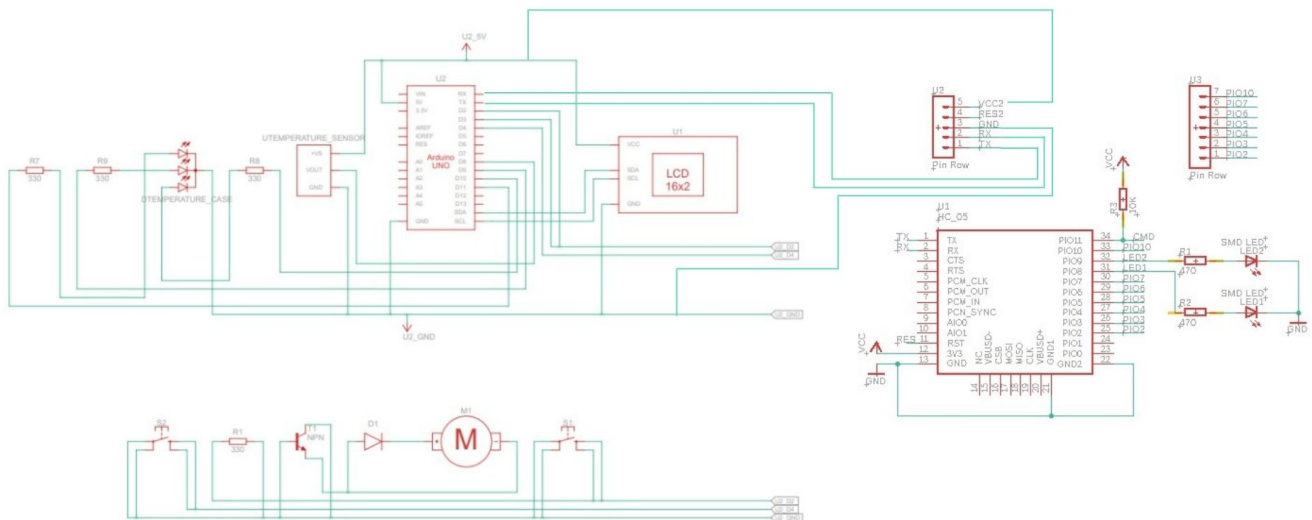
Hardware Design

Piese necesare:

- Arduino Uno
- LCD cu Interfata I2C
- Senzor de temperatură DHT11
- Motor DC
- Tranzistor NPN
- Diodă
- Elice
- Fire Tată-Tată
- Fire Mamă-Tată
- Breadboard
- LED RGB
- Butoane - 2
- Modul Bluetooth HC-05
- Rezistențe 330 Ω

Schemă electrică

Schemă electrică



Software Design

Mediul de dezvoltare

- Draw.io - schema bloc
- Arduino IDE - program
- Tinkercad + Eagle - schema electrică
- Aplicație BT Terminal - conectare prin Bluetooth

Biblioteci

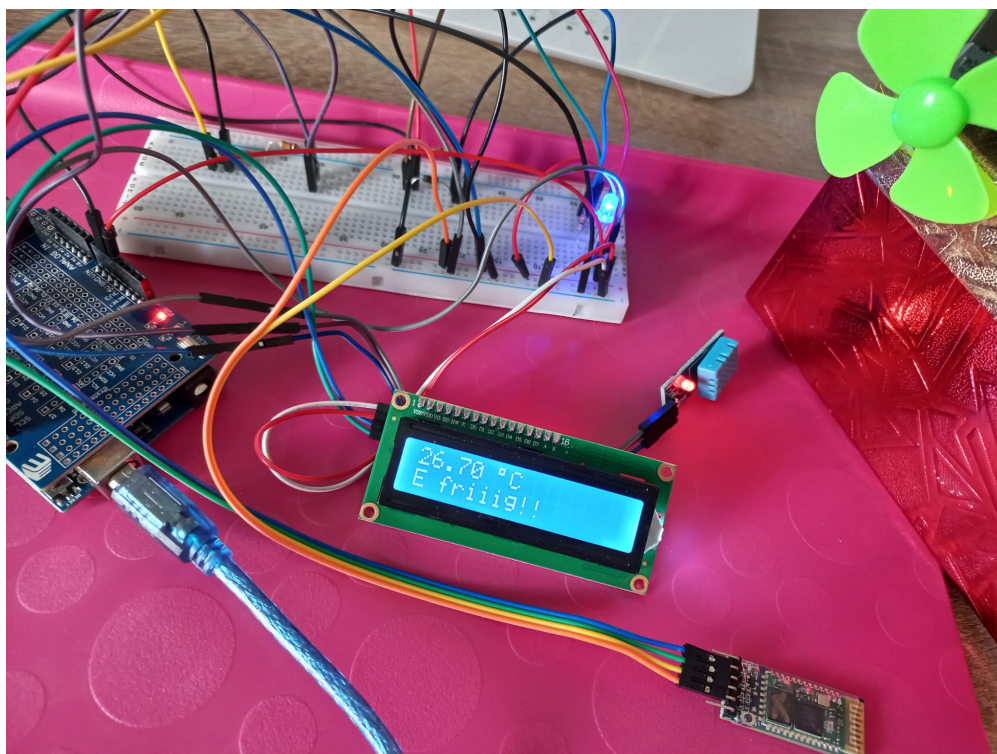
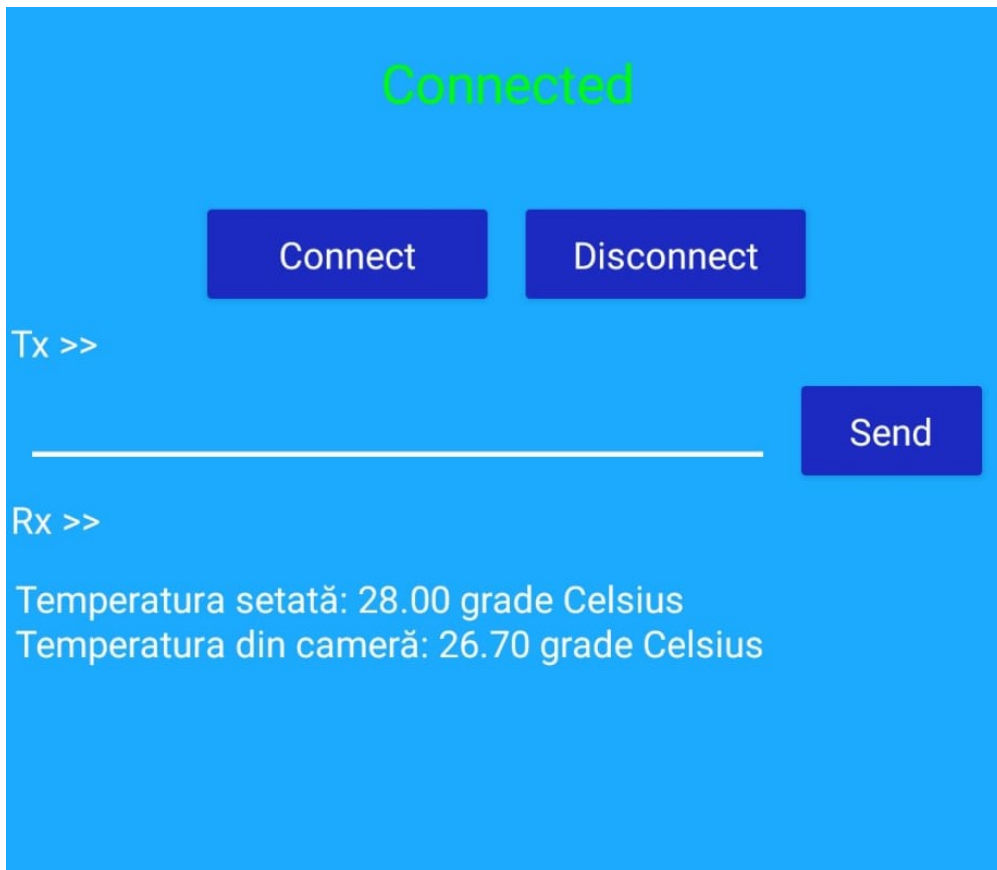
- LiquidCrystal_I2C.h pentru LCD I2C
- DHT.h pentru senzorul de temperatură

Funcții

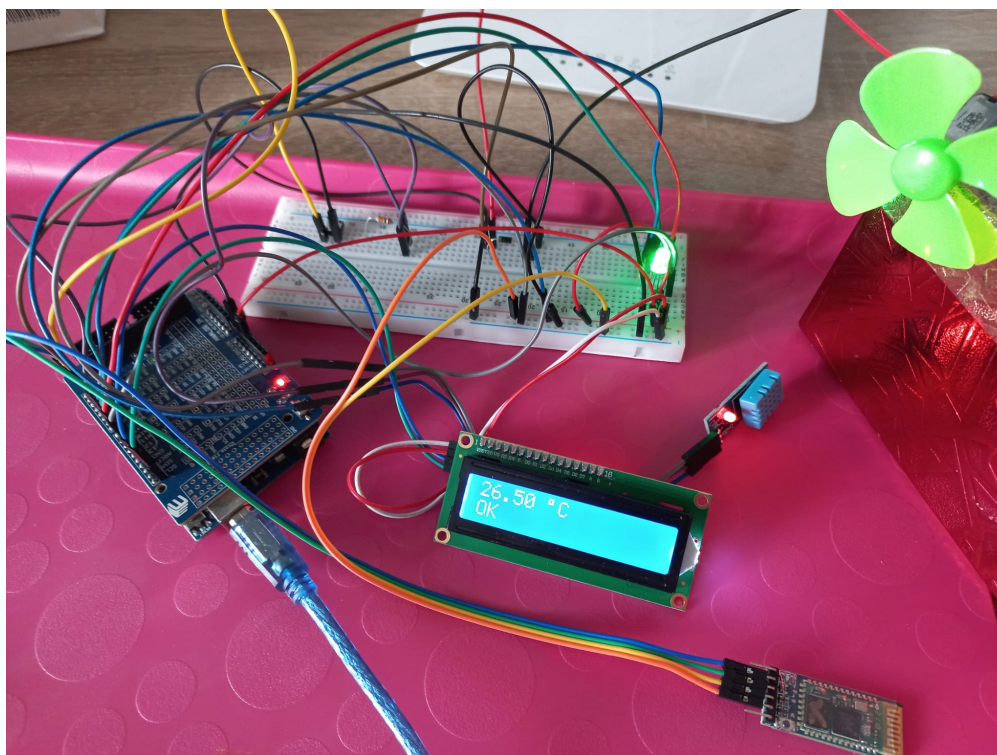
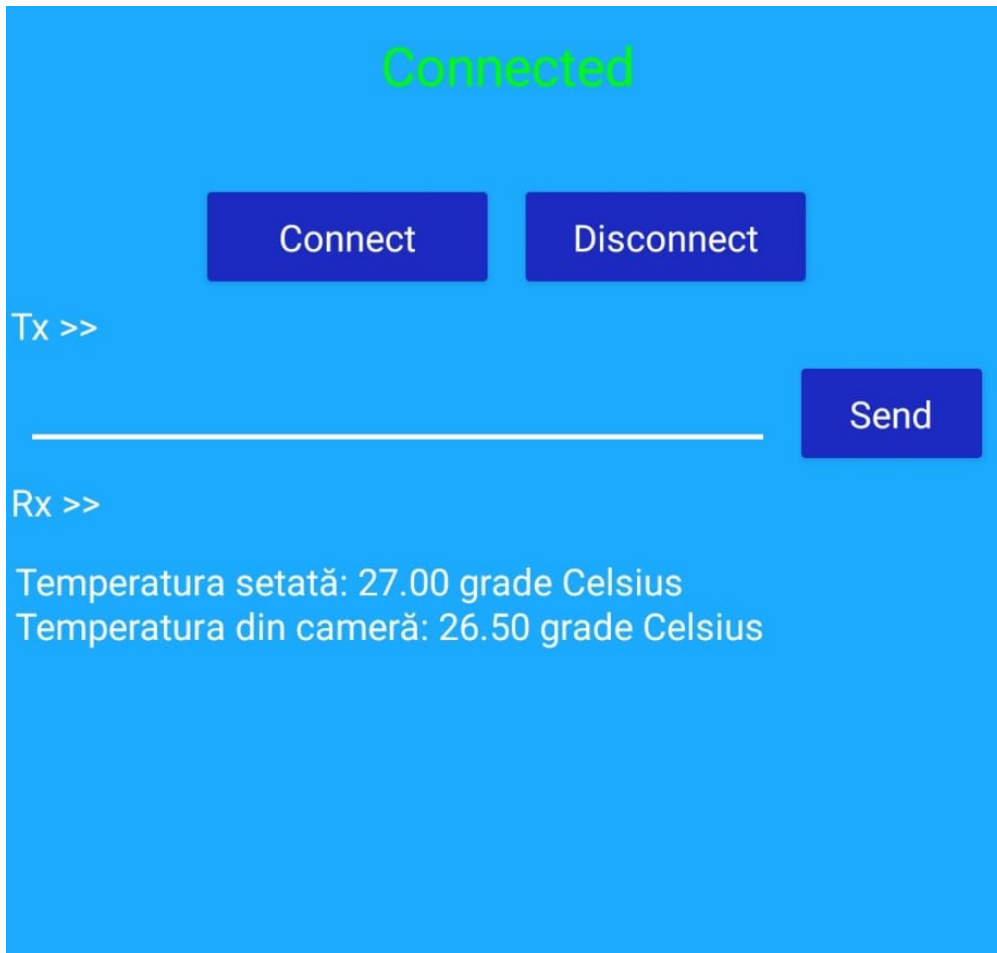
- dht() pentru inițializarea modului DHT11; aici am folosit funcția readTemperature() pentru a extrage temperatura înregistrată de senzor
- lcd() pentru inițializarea ecranului LCD; funcțiile folosite aici sunt print() și setCursor() pentru a afișa informații pe ecran
- setup() - inițializarea modulelor și a pinilor
- loop() - flow-ul circuitului

Rezultate obținute

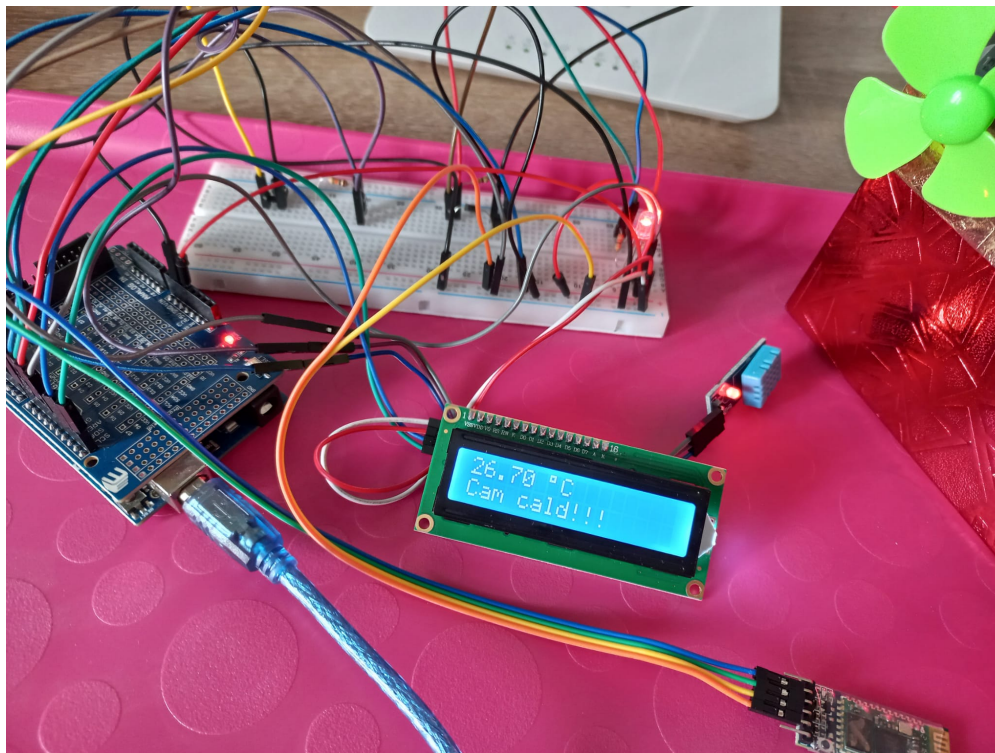
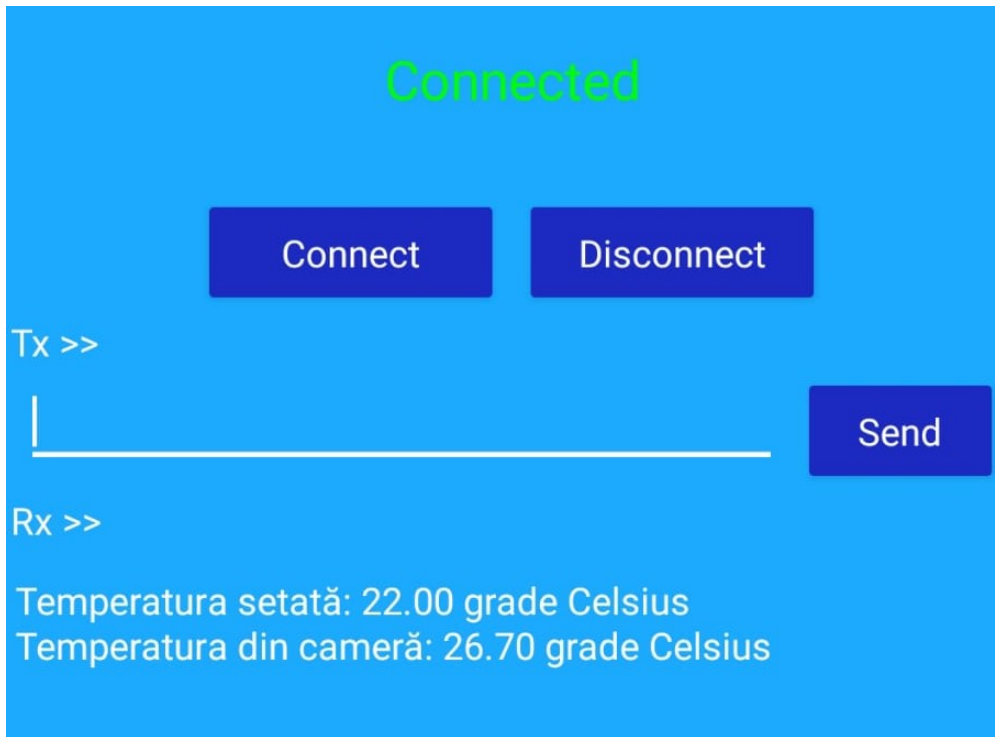
Cazul 1 - Temperatura din cameră este mai mică decât temperatura setată



Cazul 2 - Temperatura din cameră este în intervalul dorit



Cazul 3 - Temperatura din cameră este mai mare decât temperatura setată



Concluzii

Mi-a plăcut să lucrez la acest proiect și mă bucur că am putut să pun în practică tot ce am învățat.

Download

[termostat_336cc.rar](#)

Bibliografie/Resurse

<https://github.com/ErichStyger/mcuoneclipse/blob/master/Eagle/Library/Bluetooth%20HC-05.lbr>

<https://create.arduino.cc/projecthub/gatoninja236/arduino-lcd-thermostat-a02f52>

https://www.tutorialspoint.com/arduino/arduino_dc_motor.htm

<https://cdn-shop.adafruit.com/datasheets/WS2812B.pdf>

<https://components101.com/wireless/hc-05-bluetooth-module>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/sgherman/termostat>



Last update: **2022/06/02 06:59**