

Smart Pet Fountain

Introducere

- **Autor:** Diana Novac
- **Grupa:** 331CA

Proiectul implementează o fântână inteligentă pentru animale de companie pentru monitorizarea comportamentului de hidratare al acestora. Sistemul detectează prezența animalului prin senzor ultrasonic, îl identifică individual prin RFID, controlează debitul pompei în funcție de prezență și monitorizează calitatea și nivelul apei în timp real. Frecvența și durata vizitelor sunt stocate în EEPROM și afișate pe un display LCD cu meniu. Ideea a pornit de la nevoia din casele cu mai multe animale de companie de a monitoriza individual cât bea fiecare animal. Identificarea prin RFID e inspirată din sistemul de microcipare deja existent la animale de companie.

Descriere generală

Modulele proiectului

Modulul de detecție și identificare utilizează senzorul de ultrasonic pentru a monitoriza zona frontală a fântânii. La detectarea unui obiect la o distanță sub pragul configurat, se activează cititorul RFID pentru a scana prezența unui tag (în proiectul meu tag-urile vor avea scop demonstrativ, în realitate identificarea ar fi făcută pe baza microcipului animalului). ATmega328p procesează semnalele și generează un semnal PWM către etajul de putere pentru controlul pompei. Senzorul de nivel și cel de turbiditate a apei funcționează prin ADC și oferă date brute despre starea fântânii. Display-ul LCD funcționează atât ca meniu controlat prin butoane (de ex. pentru configurarea pragului de distanță), cât și ca interfață unde utilizatorul poate vedea statisticile pentru fiecare animal.

Schema bloc



Fluxul de funcționare

În starea de repaus, sistemul scanează distanța la intervale regulate, iar pompa rulează cu un debit foarte mic. Când senzorul de distanță confirmă prezența animalului, sistemul scanează tagul RFID și identifică animalul. Debitul pompei crește progresiv, iar în acest timp microcontrolerul măsoară intern durata vizitei și monitorizează calitatea apei. După ce animalul părăsește zona de detecție, pompa își micșorează debitul, iar datele colectate sunt stocate în EEPROM pentru a asigura persistența statisticilor. Dacă senzorul de nivel al apei detectează lipsa acesteia sau turbiditatea depășește un prag de siguranță, sistemul blochează pompa pentru a proteja hardware-ul și afișează un mesaj de eroare pe LCD.

Hardware Design

Lista de componente

- **Microcontroller:** ATmega328p explained mini
- **Senzor de distanță:** Ultrasonic HC-SR04
- **Modul RFID RC522**
- **Senzor nivel apă**
- **Senzor calitatea apei:** TDS (Total Dissolved Solids)
- **Pompă submersibilă 3-6V**
- **Display:** LCD cu modul I2C
- **Butoane de control**
- **Tranzistor MOSFET**
- **Diodă flyback**
- **Rezistențe**
- **Sursă breadboard**

Schema electrică



Pinii folosiți pentru fiecare componentă

- **Ecranul LCD:** Pinii dedicați pentru magistrala I2C pe ATmega328p sunt PC4 pentru SDA și PC5 pentru SCL.
- **Senzorul ultrasonic HC-SR04:** Senzorul necesită un pin de ieșire pentru Trig și un pin de intrare pentru Echo. Am ales pinii PD2 și PD3 ca pini GPIO.
- **Modulul RFID RC522:** Pinii PB2, PB3, PB4, PB5 și PB0. Modulul comunică prin protocolul SPI. Pe

ATMega328p, pinii hardware pentru SPI sunt PB3 (MOSI), PB4 (MISO) și PB5 (SCK). Pinul PB0 e folosit pentru resetarea modulului și poate fi orice pin digital liber, dar am ales PB0 pentru a păstra pinii portului D disponibili pentru butoane și senzorul ultrasonic.

- **Pompa submersibilă:** Pompa e controlată printr-un semnal PWM pentru reglarea debitului. Pinul PB1 e conectat intern la Timer1 al microcontrolerului.
- **Senzorul de nivel al apei și senzorul TDS:** Ambii senzori necesită pini ADC, am ales pinii PC0 și PC1.
- **Butoanele:** Butoanele sunt conectate ca intrări digitale cu rezistențe pull-up externe și am ales pinii PD4, PD5 și PD6.

Software Design

- **Mediu de dezvoltare:** PlatformIO
- **Biblioteci și resurse folosite:** Am făcut configurarea Timer1 pentru generarea semnalului PWM al pompei prin scriere directă în registrele AVR TCCR1A, TCCR1B și OCR1A. Configurarea ADC pentru citirea senzorilor de nivel și TDS se face prin registrele ADMUX și ADCSRA. Am folosit biblioteca *MFRC522.h* pentru comunicația cu cititorul RFID RC522 prin SPI. Biblioteca gestionează inițializarea registrelor interne ale modulului și citirea UID-ului tag-urilor. O altă bibliotecă folosită este *EEPROM.h* pentru salvarea și citirea persistentă a statisticilor per animal.

Mașina de stări

Sistemul este implementat ca o mașină de stări finită cu patru stări: Repaus, Activ, Eroare și Configurare. Starea curentă e reținută într-un enum. Tranzițiile între stări sunt declanșate de flag-uri booleene setate de funcțiile de citire ale senzorilor.

Profiluri animale

Datele fiecărui animal sunt reținute într-o structură *PetProfile* care conține UID-ul tag-ului RFID, numărul de vizite din ziua curentă, durata acestora și timestamp-ul ultimei vizite. Array-ul de aceste structuri e sincronizat cu EEPROM-ul la finalul fiecărei vizite.

Controlul pompei

Debitul e reglat prin modificarea valorii registrului OCR1A care controlează duty cycle-ul semnalului PWM. La detectarea animalului, valoarea e incrementată la fiecare iterație a loop-ului până la nivelul maxim configurat, iar la plecarea animalului procesul e inversat.

Filtrarea ADC

Citirile brute de la senzorul nivel și cel TDS sunt procesate și filtrate pentru a reduce zgomotul.

Funcționalități din laboratoare

- **GPIO** - folosit pentru controlul pinilor de Trig și Echo ai senzorului ultrasonic HC-SR04 și pentru citirea stării butoanelor de meniu
- **Întreruperi** - folosite pentru detectarea apăsărilor de butoane prin PCINT
- **PWM** - duty cycle-ul PWM e modificat pentru a regla debitul pompei
- **SPI** - folosit pentru comunicația cu modulul RFID RC522.
- **I2C** - folosit pentru comunicația cu display-ul LCD 16×2
- **ADC** - folosit pentru citirea semnalelor analogice ale senzorilor de nivel al apei și cel TDS

Jurnal

Bibliografie/Resurse

[Export to PDF](#)

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

Permanent link:
http://ocw.cs.pub.ro/courses/pm/prj2026/farhad_ali.gul/diana_ioana.novac



Last update: **2026/05/20 16:01**