Detector de culoare

Introducere

Acest proiect are ca scop detectarea de culoare a unui obiect fara a fi nevoie sa il privesti. Astfel vor fi folosite niste foite colorate pentru a fi indicata culoarea acestora. Proiectul va avea la baza cunostinte din laboratoarele cu urmatoarele concepte:

1. UART
2. I2C
3. Timere
4. Intreruperi

Descriere generală

Acest proiect detecteaza culoarea unui obiect fara a fi nevoie sa il privesti. Când se plaseaza o foaie colorată în fața senzorului, acesta detectează culoarea și o afișează pe ecranul LCD 16×2, pe ecranul PC-ului apare un mesaj sugestiv si se aprinde beculetul cu culoarea detectata. Flow-ul programului va fi urmatorul:

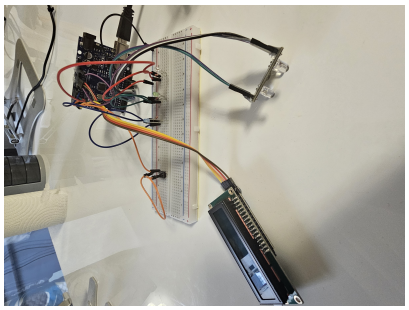
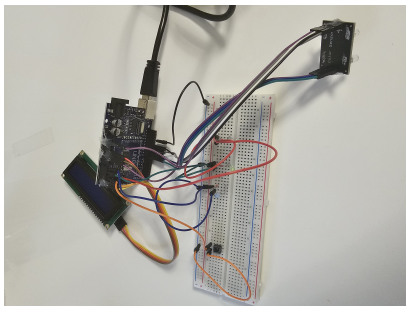
- Se apasa butonul de start
- Utilizatorul plaseaza o foita colorata pentru detectare
- Timer-ul incepe sa functioneze pentru a permite procesarea
- Culoarea este apoi indicata atat printr-un led, cat si printr-un mesaj sugestiv afisat pe ecranul PC-ului si pe ecran I2C

Hardware Design

Piese necesare: Arduino Uno

- LCD I2C 1602
- TCS-230 Senzor de culoare
- Placa de dezvoltare UNO R3 Arduino compatibil ATmega328p

- Rezistente
- BreadBoard
- Fire Tata-Tata, Tata-Mama
- Buton
- Foite colorate
- LED

Software Design

Programul folosește bibliotecile Wire și LiquidCrystal_I2C pentru a comunica cu ecranul LCD prin intermediul protocolului I2C.

Structurile si functiile folosite sunt:

Structuri:

- Enum `State`: folosit pentru a defini două stări posibile ale programului, WAITING și DETECTING. - Obiectul `LiquidCrystal_I2C`: utilizat pentru a controla ecranul LCD.

Functii:

- `setup()`: Funcția de setup care este apelată o singură dată când programul începe. Aici sunt inițializate pinurile și sunt stabilite anumite setări. - `loop()`: Această funcție este inima programului. Ea se execută repetitiv atâta timp cât placa Arduino este alimentată. Conține o structură switch care modifică comportamentul programului în funcție de starea actuală (WAITING sau DETECTING).

Alte funcții utilizate sunt:

- `digitalWrite()`: Aceasta este o funcție Arduino care stabilește starea unui pin digital la HIGH sau LOW. - `digitalRead()`: Aceasta este o altă funcție Arduino care citește starea unui pin digital. - `pulseIn()`: Aceasta este o funcție Arduino care măsoară durata unui puls pe un pin digital. - `millis()`: Aceasta este o funcție Arduino care returnează numărul de milisecunde de când placa a început să ruleze programul curent. - `delay()`: Aceasta este o funcție Arduino care oprește execuția programului pentru un anumit număr de milisecunde. - Funcții de control al LCD-ului (print, setCursor, clear, etc.).

Acest program așteaptă apăsarea unui buton pentru a începe detectarea culorilor. Detectează culoarea unui obiect și apoi afișează culoarea pe un ecran LCD și aprinde un LED corespunzător (roșu,

verde sau albastru). În cazul în care butonul este apăsat din nou în timpul detectării, programul se întoarce în starea de așteptare și oprește toate LED-urile.

Rezultate Obținute

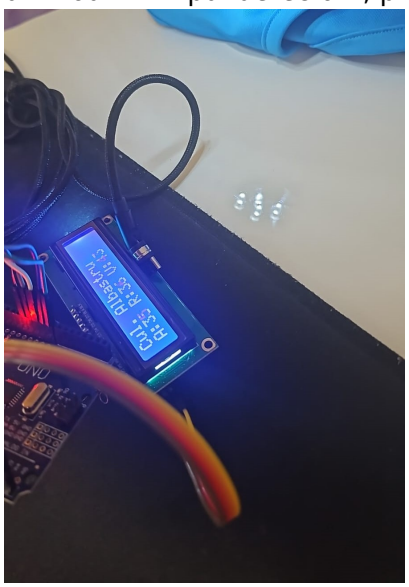
Rezultatele se obțin prin utilizarea unui senzor de culoare care funcționează cu trei fotodiodi filtrate pentru roșu, verde și albastru. Senzorul emite frecvențe diferite pentru culorile detectate.

Pentru a citi aceste frecvențe, programul setează pinii S2 și S3 ai senzorului la diferite valori, în funcție de culoarea pe care dorește să o citească.

Pentru a citi roșu, programul setează S2 și S3 la LOW. Pentru a citi verde, S2 și S3 sunt setate la HIGH. Pentru albastru, S2 este setat la LOW și S3 la HIGH.

Odată ce pinii sunt setați corespunzător, programul utilizează funcția `pulseIn()` pentru a citi frecvența de ieșire de la senzor.

Frecvențele roșu, verde și albastru sunt comparate pentru a determina care este cea mai mică. Dacă frecvența roșie este cea mai mică, se consideră că obiectul este roșu. Similar, dacă frecvența verde sau albastră este cea mai mică, se consideră că obiectul este verde sau albastru. Rezultatul acestei detecții este apoi afișat pe un ecran LCD și transmis prin portul serial pentru a fi citit de un alt dispozitiv. De asemenea, LED-ul corespunzător culorii detectate este aprins. Dacă butonul este apăsat din nou în timpul detectării, programul se întoarce în starea de așteptare și oprește toate LED-urile.



Concluzii

Proiectul exemplifică utilizarea senzorilor de culoare și a plăcilor Arduino pentru a crea un dispozitiv simplu de detectare a culorilor. Rezultatele sunt afișate într-un mod interactiv și instructiv, prin utilizarea unui ecran LCD și a unor LED-uri colorate.

Utilizarea protocolului I2C pentru a comunica cu ecranul LCD este un exemplu de comunicare serială între placa Arduino și alte componente hardware. I2C este un protocol de comunicare popular datorită flexibilității și eficienței sale.

În concluzie, acest cod exemplifică multe dintre principiile și tehnicile utilizate în programarea plăcilor Arduino și a dispozitivelor electronice. Deși aparent simplu, el demonstrează eficient cum se pot utiliza senzorii și alte componente pentru a crea dispozitive interactiv și util în viața de zi cu zi.

Download

arhivapm.zip

Bibliografie/Resurse

Resurse:

1. Documentație Arduino:

1. Website-ul oficial Arduino: https://www.arduino.cc/
2. Ghidul de referință Arduino: https://www.arduino.cc/reference/en/

2. Senzor de culoare TCS230:

1. Fișa tehnică a senzorului TCS230: https://www.mouser.com/datasheet/2/588/TCS230_E11-1225542.pdf
2. Tutorial pentru senzorul de culoare TCS230: https://create.arduino.cc/projecthub/jameeltronics/color-detection-using-tcs230-tcs3200-color-sensor-and-arduino-b1e0cd

3. LCD I2C:

1. Biblioteca LiquidCrystal_I2C pentru Arduino: https://www.arduinolibraries.info/libraries/liquid-crystal-i2-c
2. Tutorial pentru utilizarea unui LCD I2C cu Arduino: https://dronebotworkshop.com/i2c-lcd-display/

Export to PDF

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

Permanent link:
http://ocw.cs.pub.ro/courses/pm/prj2023/drtranca/detector_de_culoare

Last update: **2023/05/27 21:28**