

ColorTrack - Sistem embedded de detecție și scanare a culorilor

Sistem embedded pentru identificarea culorilor, control vizual prin LED RGB și scanare automată cu motor pas cu pas

Autor: Greere Stefan

Grupa: 334CA

Introducere

ColorTrack este un sistem embedded bazat pe microcontrollerul **ATmega328P**, proiectat pentru detecția și căutarea automată a culorilor. Proiectul pornește de la un scenariu practic de control al calității, în care culoarea unui obiect, a unei etichete sau a unei mostre trebuie verificată rapid și consecvent.

Sistemul poate funcționa în două moduri. Primul mod permite detectarea punctuală a culorii unui obiect plasat în fața senzorului. Rezultatul este afișat pe LCD, transmis prin interfața serială și semnalizat printr-un LED RGB. În acest mod, LED-ul RGB nu indică doar una dintre culorile principale, ci folosește PWM software pentru a aproxima nuanța detectată de senzor.

Al doilea mod permite căutarea automată a unei culori țintă. Utilizatorul selectează culoarea dorită prin interfața serială, iar sistemul rotește o platformă cu ajutorul unui motor pas cu pas. La fiecare pas, senzorul verifică culoarea detectată. Când culoarea țintă este găsită, motorul se oprește, rezultatul este afișat pe LCD, LED-ul RGB se aprinde corespunzător, iar platforma este rotită înapoi în poziția inițială.

Scopul proiectului este realizarea unui sistem capabil să:

- detecteze culoarea dominantă a unui obiect folosind senzorul TCS230;
- afișeze starea sistemului și rezultatele pe un LCD 1602 cu interfață I2C;
- aproximeze nuanța detectată prin LED RGB controlat cu PWM software;
- permită alegerea modului de funcționare prin USART;
- controleze un motor pas cu pas prin driver ULN2003;
- caute automat o culoare țintă pe o platformă rotativă;
- folosească întreruperea externă INT0 pentru pornirea detecției sau scanării;
- păstreze modulul microSD conectat pentru o posibilă extensie de jurnalizare.

Descriere generală

ColorTrack funcționează ca un sistem de inspecție și scanare a culorilor. Utilizatorul interacționează cu sistemul prin interfața serială, alegând unul dintre cele două moduri de funcționare.

Modul 1 - Detectie simplă

În acest mod, utilizatorul trimite comanda `1` prin USART. Sistemul intră în modul de detecție simplă și aprinde LED-ul RGB în alb, indicând faptul că așteaptă o comandă de start. Detecția propriu-zisă se face doar după apăsarea butonului. După citirea senzorului TCS230, microcontrollerul determină culoarea dominantă, afișează rezultatul pe LCD și controlează LED-ul RGB prin PWM software pentru a aproxima nuanța detectată.

Modul 2 - Căutare culoare

În acest mod, utilizatorul trimite comanda `2` prin USART, apoi alege culoarea țintă folosind una dintre comenzile `R`, `G` sau `B`. După apăsarea butonului, motorul pas cu pas rotește platforma în pași succesivi. La fiecare interval de pași, senzorul TCS230 citește culoarea curentă. Dacă este detectată culoarea țintă, motorul se oprește, rezultatul este afișat pe LCD, LED-ul RGB indică vizual culoarea găsită, iar motorul revine în poziția inițială.

Modulele principale ale sistemului sunt:

- microcontrollerul ATmega328P - coordonează întregul sistem;
- senzorul TCS230 - detectează culoarea obiectului;
- LCD-ul 1602 I2C - afișează starea sistemului și rezultatul;
- LED-ul RGB - oferă feedback vizual și aproximare de nuanță prin PWM software;
- butonul de start - declanșează detecția sau scanarea, folosind întreruperea externă INT0;
- driverul ULN2003 - controlează motorul pas cu pas;
- motorul pas cu pas 28BYJ-48 - rotește platforma de scanare;
- interfața USART - permite alegerea modului de funcționare și a culorii țintă;
- modulul microSD - conectat hardware prin SPI, păstrat pentru o extensie ulterioară de jurnalizare.

Fluxul general de funcționare este următorul:

1. sistemul este alimentat și inițializează modulele hardware;
2. LCD-ul afișează meniul principal;
3. utilizatorul selectează modul de funcționare prin USART;
4. în modul 1, utilizatorul apasă butonul, iar sistemul face o detecție simplă;
5. în modul 2, utilizatorul alege culoarea țintă, apoi apasă butonul pentru pornirea scanării;
6. motorul rotește platforma și sistemul citește periodic senzorul;
7. dacă este găsită culoarea dorită, motorul se oprește;
8. LED-ul RGB și LCD-ul afișează rezultatul;
9. motorul revine în poziția inițială.



Hardware Design

Stadiul actual al implementării hardware

În stadiul actual, au fost conectate și testate principalele componente hardware ale proiectului. LCD-ul 1602 cu interfață I2C afișează mesaje de stare, senzorul TCS230 este folosit pentru citirea componentelor de culoare, LED-ul RGB oferă feedback vizual, iar motorul pas cu pas este controlat prin driverul ULN2003.

Sistemul permite selectarea modului de funcționare prin USART. Butonul conectat pe pinul INT0 este folosit pentru declanșarea detecției sau a scanării. În modul de scanare, motorul rotește platforma până la găsirea culorii țintă sau până la parcurgerea limitei de rotație. Pentru protejarea machetei, după scanare motorul revine în poziția inițială prin rotirea inversă cu același număr de pași.

Modulul microSD este păstrat conectat la pinii SPI ai microcontrollerului, însă funcția de scriere pe card nu este inclusă în varianta curentă stabilă a proiectului. Acesta rămâne o extensie posibilă pentru jurnalizarea detecțiilor.

Componente folosite

Componentă	Cantitate	Scop
ATmega328P-XMINI	1	Placa de dezvoltare folosită ca unitate centrală a proiectului.
Senzor de culoare TCS230	1	Detectează culoarea obiectului prin semnal digital cu frecvență dependentă de lumină.
LCD 1602 I2C	1	Afișează meniul, modul curent, culoarea detectată și starea scanării.
LED RGB catod comun	1	Oferă feedback vizual; în modul 1 este controlat prin PWM software.
Rezistențe 220Ω	3	Limitează curentul prin canalele LED-ului RGB.
Buton	1	Declanșează detecția sau scanarea folosind întreruperea externă INT0.
Driver ULN2003	1	Permite controlul motorului pas cu pas din pini digitali ai microcontrollerului.
Motor pas cu pas 28BYJ-48	1	Rotește platforma de scanare în modul de căutare culoare.
Modul microSD	1	Modul conectat prin SPI, păstrat pentru extensie de jurnalizare.
Card microSD	1	Suport de stocare planificat pentru salvarea viitoare a detecțiilor.
Breadboard / fire jumper	-	Realizează conexiunile dintre placa de dezvoltare și module.
Sursă de alimentare 5V	1	Alimentează placa și modulele conectate.

Conectarea pinilor

Componentă	Pin componentă	Pin ATmega328P-XMINI	Pin microcontroller	Rol
LCD 1602 I2C	SDA	A4	PC4	Linie de date I2C/TWI
LCD 1602 I2C	SCL	A5	PC5	Linie de ceas I2C/TWI
LCD 1602 I2C	VCC	5V	-	Alimentare
LCD 1602 I2C	GND	GND	-	Masă comună

TCS230	S0	5V	-	Scalare frecvență fixată hardware
TCS230	S1	GND	-	Scalare frecvență fixată hardware
TCS230	S2	A0	PC0	Selectare filtru culoare
TCS230	S3	A1	PC1	Selectare filtru culoare
TCS230	OUT	D8	PB0	Semnal digital de ieșire de la senzor
TCS230	VCC	5V	-	Alimentare
TCS230	GND	GND	-	Masă comună
LED RGB	R	D6	PD6	Control canal roșu prin rezistență 220Ω
LED RGB	G	D5	PD5	Control canal verde prin rezistență 220Ω
LED RGB	B	D3	PD3	Control canal albastru prin rezistență 220Ω
LED RGB	Catod comun	GND	-	Masă comună
Buton start	pin 1	D2	PD2 / INT0	Declanșare prin întrerupere externă
Buton start	pin 2	GND	-	Apăsare = nivel logic LOW
ULN2003	IN1	A2	PC2	Control fază motor
ULN2003	IN2	A3	PC3	Control fază motor
ULN2003	IN3	D9	PB1	Control fază motor
ULN2003	IN4	D4	PD4	Control fază motor
ULN2003	VCC	5V	-	Alimentare driver/motor
ULN2003	GND	GND	-	Masă comună
USART	RX	D0	PD0	Recepție comenzi din terminal
USART	TX	D1	PD1	Transmitere mesaje către terminal
microSD	CS	D10	PB2	Selectare dispozitiv SPI
microSD	MOSI	D11	PB3	Linie SPI Master Out Slave In
microSD	MISO	D12	PB4	Linie SPI Master In Slave Out
microSD	SCK	D13	PB5	Ceas SPI
microSD	VCC	5V	-	Alimentare modul microSD
microSD	GND	GND	-	Masă comună

Pinii A4 și A5 au fost aleși pentru LCD deoarece aceștia corespund interfeței hardware I2C/TWI a microcontrollerului. Pinii D10-D13 au fost rezervați pentru modulul microSD, deoarece aceștia corespund interfeței SPI hardware.

Butonul a fost conectat la D2 deoarece acest pin poate fi folosit ca întrerupere externă INT0. Astfel, apăsarea butonului poate declanșa rapid o acțiune, fără a fi necesară verificarea continuă a stării butonului în bucla principală.

Motorul pas cu pas este controlat prin patru pini digitali conectați la intrările IN1-IN4 ale driverului ULN2003. Driverul este necesar deoarece pinii microcontrollerului nu pot furniza direct curentul necesar pentru alimentarea bobinelor motorului.

Pinii S0 și S1 ai senzorului TCS230 au fost fixați hardware la 5V, respectiv GND. Această configurație setează scalarea frecvenței de ieșire la aproximativ 20%, valoare potrivită pentru măsurarea cu microcontrollerul. Pinii S2 și S3 rămân controlați software pentru selectarea filtrului de culoare.

Schema electrică

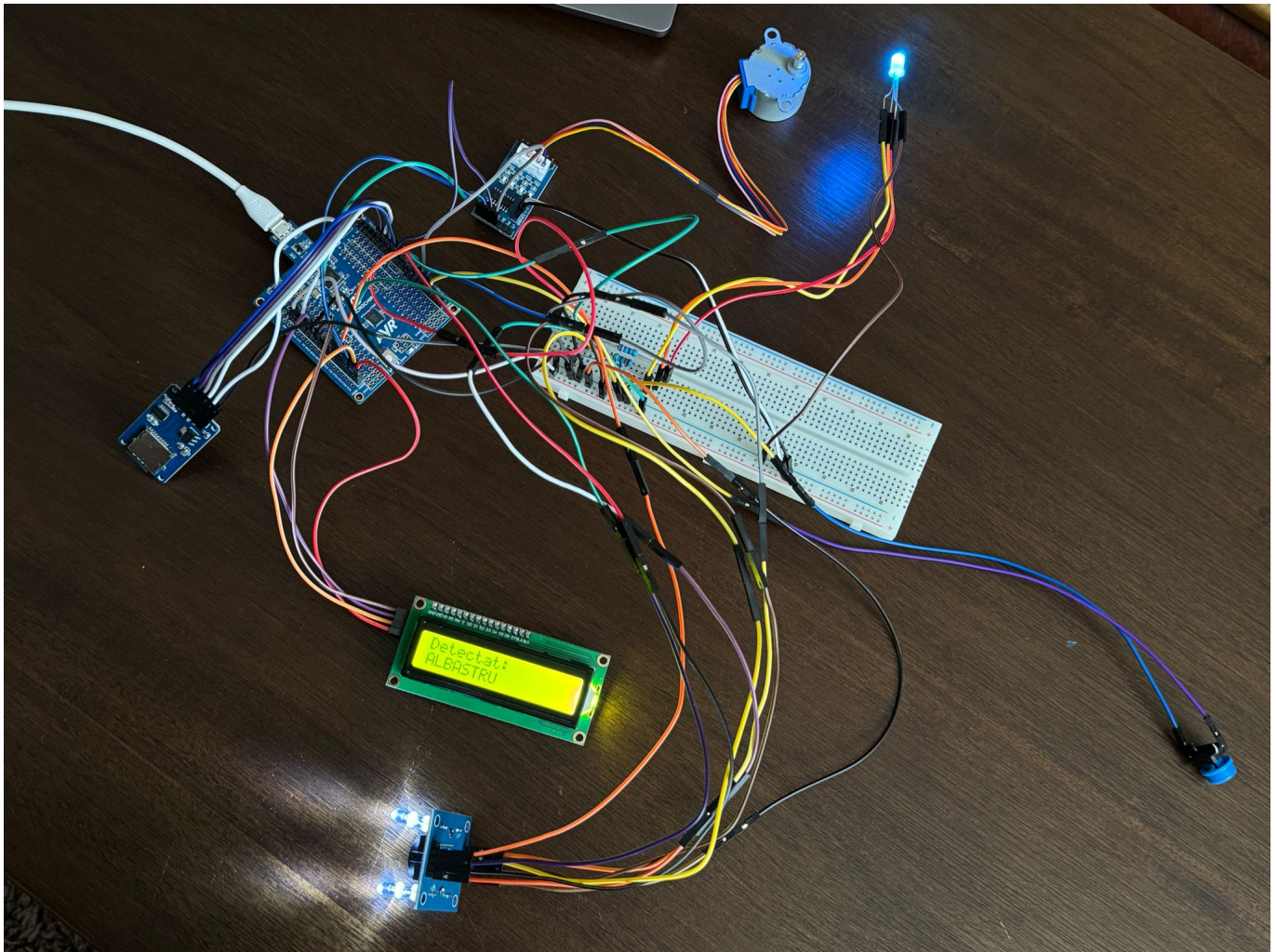
Schema electrică a fost realizată în EasyEDA. Aceasta prezintă conexiunile dintre placa ATmega328P-XMINI și modulele externe folosite: senzorul TCS230, LCD-ul I2C, LED-ul RGB, butonul de start, driverul ULN2003 cu motor pas cu pas și modulul microSD.



În schemă, modulele externe sunt reprezentate ca blocuri cu pini, deoarece acestea sunt module prefabricate. Scopul schemei este evidențierea conexiunilor dintre placa de dezvoltare și componentele folosite.

Imagini ale montajului și teste hardware

În imagine se observă montajul integrat al proiectului: placa ATmega328P-XMINI, LCD-ul 1602 I2C, senzorul TCS230, LED-ul RGB, driverul ULN2003 cu motor pas cu pas, butonul de start și modulul microSD conectat fizic. LCD-ul este alimentat și afișează mesaje, iar LED-ul RGB este aprins, ceea ce demonstrează funcționarea componentelor de afișare/feedback vizual.



Software Design

Stadiul actual al implementării software

În stadiul actual, aplicația este funcțională și integrează principalele module hardware ale proiectului: senzorul TCS230, LCD-ul 1602 I2C, LED-ul RGB, butonul de start, motorul pas cu pas, USART-ul și modulul microSD.

Proiectul permite utilizarea proiectului în două moduri de funcționare:

- **Modul 1 - detecție simplă:** utilizatorul selectează modul prin USART, apoi apasă butonul. Sistemul citește culoarea cu senzorul TCS230, afișează rezultatul pe LCD, aprinde LED-ul RGB folosind PWM software și salvează culoarea detectată pe cardul microSD.
- **Modul 2 - căutare culoare:** utilizatorul selectează prin USART culoarea dorită, iar motorul pas cu pas rotește platforma la stânga și la dreapta până când culoarea țintă este detectată. După scanare, motorul revine în poziția inițială.

Biblioteci și resurse software folosite

Pentru dezvoltare am folosit:

- **PlatformIO** - pentru organizarea proiectului, compilare și upload pe placă;
- **avr-gcc** - compilatorul folosit pentru microcontrollerul ATmega328P;
- **avr-libc** - pentru accesul la registrele AVR, întârzieri, întreruperi și tipuri de date standard;
- **Petit FatFs** - pentru comunicarea cu sistemul de fișiere FAT de pe cardul microSD.

Structura proiectului

Proiectul a fost organizat într-un număr redus de fișiere, pentru a păstra codul ușor de urmărit:

- **main.c** - inițializează modulele și rulează bucla principală;
- **app.c** - conține logica celor două moduri de funcționare;
- **hardware.c** - conține funcțiile pentru LCD, USART, PWM software, buton, motor și senzor;
- **sd_log.c** - conține funcțiile pentru scrierea pe cardul microSD;
- **pff.c, sd.c, spi.c** - conțin partea de comunicare cu cardul SD și biblioteca Petit FatFs;
- fișierele **.h** sunt puse separat în folderul ``include``.

Interacțiunea dintre module este următoarea: utilizatorul alege modul prin USART, butonul declanșează acțiunea curentă prin întreruperea externă INT0, senzorul TCS230 este citit cu ajutorul Timer1, rezultatul este afișat pe LCD prin I2C, LED-ul RGB este controlat prin PWM software, iar motorul este comandat prin driverul ULN2003.

Funcționalități din laborator folosite

În cadrul proiectului au fost folosite mai multe concepte studiate la laborator:

- **GPIO** - pentru controlul LED-ului RGB, al pinilor senzorului TCS230 și al driverului ULN2003;
- **întrerupere externă INT0** - pentru butonul de start;
- **Timer0** - pentru măsurarea timpului și debounce la buton;
- **Timer1** - pentru măsurarea semnalului generat de senzorul TCS230;
- **Timer2** - pentru implementarea PWM software pentru LED-ul RGB;
- **I2C/TWI** - pentru afișarea mesajelor pe LCD 1602;
- **USART** - pentru selectarea modului de funcționare și alegerea culorii țintă;
- **SPI** - pentru comunicarea cu modulul microSD;

Elementul de noutate al proiectului

Elementul de noutate este combinarea detecției de culoare cu o platformă rotativă controlată automat. Proiectul nu doar detectează static o culoare, ci poate căuta o culoare țintă aleasă de utilizator, rotind motorul la stânga și la dreapta până la găsirea acesteia.

Un alt element interesant este folosirea PWM-ului software pentru LED-ul RGB. În modul de detecție simplă, LED-ul nu afișează doar roșu, verde sau albastru, ci încearcă să aproximeze nuanța detectată de senzor.

Calibrarea senzorului

Senzorul TCS230 nu oferă direct valori RGB între 0 și 255, ci generează un semnal digital cu frecvență dependentă de intensitatea luminii. În implementare, microcontrollerul măsoară durata impulsurilor pentru filtrele roșu, verde și albastru.

Calibrarea a fost realizată experimental, prin testarea mai multor obiecte colorate și observarea valorilor brute citite de senzor. Pe baza acestor valori au fost ajustate pragurile pentru culorile principale și pentru câteva nuanțe suplimentare, cum ar fi portocaliu, roz, galben și mov.

În cod, o valoare mai mică înseamnă o frecvență mai mare, deci o componentă de culoare mai intensă. Din acest motiv, comparațiile sunt făcute invers față de un sistem RGB clasic.

Optimizări realizate

Au fost realizate mai multe optimizări pentru a păstra proiectul stabil pe ATmega328P:

- pinii S0 și S1 ai senzorului TCS230 au fost fixați hardware, pentru a elibera pini digitali;
- motorul nu se rotește 360°, ci scanează limitat la stânga și la dreapta, pentru a proteja macheta și firele;
- după fiecare scanare, motorul revine în poziția inițială;
- LED-ul RGB este controlat prin PWM software, fără a schimba conexiunile hardware;
- codul a fost reorganizat astfel încât să nu fie nici prea monolitic, nici împărțit în prea multe fișiere;
- pentru cardul microSD a fost folosită Petit FatFs, deoarece are consum redus de memorie;
- logarea pe microSD a fost păstrată simplă, salvând culoarea detectată, nu toate valorile brute, pentru a reduce memoria folosită.

Validare software

Funcționalitățile au fost validate incremental. Mai întâi au fost testate individual LCD-ul, LED-ul RGB, butonul, senzorul TCS230 și motorul pas cu pas. După aceea, componentele au fost integrate în aplicația finală.

Validarea finală a constat în:

- selectarea modului de funcționare prin USART;
- declanșarea acțiunii prin buton;
- detectarea culorilor în modul 1;
- afișarea rezultatului pe LCD;

- controlul LED-ului RGB prin PWM software;
- scanarea stânga-dreapta cu motorul în modul 2;
- oprirea motorului la detectarea culorii țintă;
- revenirea motorului în poziția inițială;
- salvarea culorilor detectate pe cardul microSD în modul 1.

Fragmente relevante de cod

Pentru a evidenta modul in care au fost folosite functionalitatile din laborator, mai jos sunt prezentate cateva fragmente reprezentative din codul proiectului.

Selectarea modului prin USART

Utilizatorul interactioneaza cu sistemul prin terminalul serial. In functie de caracterul primit, aplicatia schimba modul de functionare sau asteapta culoarea tinta.

```
if (c == '1') {
    current_mode = MODE_SIMPLE_DETECTION;
    rgb_white();
    lcd_show_mode_1_waiting();

    usart_print("Mod 1 selectat: detectie simpla.\r\n");
} else if (c == '2') {
    current_mode = MODE_SCAN_COLOR;
    waiting_for_target = 1;
    rgb_white();

    usart_print("Mod 2 selectat: cautare culoare.\r\n");
}
```

Acest fragment arata folosirea USART pentru controlul aplicatiei. Modul de lucru nu este ales din cod, ci de utilizator, in timpul rularii.

Pornirea actiunii folosind butonul

Butonul este conectat la intreruperea externa INT0. In bucla principala, actiunea executata depinde de modul selectat anterior.

```
if (button_pressed) {
    button_pressed = 0;

    if (current_mode == MODE_SIMPLE_DETECTION) {
        perform_simple_detection();
    } else if (current_mode == MODE_SCAN_COLOR) {
```

```
        perform_scan_for_color(current_target);
    }
}
```

Astfel, acelasi buton este folosit atat pentru detectia simpla, cat si pentru pornirea scanarii automate.

Detectarea culorii cu TCS230

Senzorul TCS230 este citit prin selectarea pe rand a filtrelor rosu, verde si albastru. Valorile obtinute sunt comparate pentru a determina culoarea dominanta.

```
color_t read_detected_color(uint16_t *red, uint16_t *green, uint16_t *blue)
{
    *red = read_color_value(tcs_select_red);
    *green = read_color_value(tcs_select_green);
    *blue = read_color_value(tcs_select_blue);

    return detect_color(*red, *green, *blue);
}
```

Citirea senzorului se bazeaza pe masurarea semnalului generat pe pinul OUT. In implementare, valoarea mai mica indica o componenta de culoare mai puternica.

Scanarea cu motorul pas cu pas

In modul de cautare, motorul roteste platforma in pasi mici. Dupa fiecare grup de pasi, sistemul citeste din nou culoarea.

```
while (steps_done < SCAN_SIDE_LIMIT) {
    motor_rotate_forward(SCAN_CHUNK_STEPS);
    steps_done += SCAN_CHUNK_STEPS;

    detected = read_detected_color(&red, &green, &blue);

    if (detected == target) {
        found = 1;
        break;
    }
}
```

Pentru a proteja macheta, motorul nu se roteste continuu la 360 de grade, ci cauta limitat la stanga si la dreapta. Dupa scanare, motorul revine in pozitia initiala.

Demo video

Funcționarea proiectului este prezentată în următorul video:

- [Demo video ColorTrack](#)

Concluzii

ColorTrack propune un sistem embedded pentru detecția și scanarea automată a culorilor. Proiectul are o utilitate practică în zona de control al calității la scară redusă, unde culoarea unui obiect poate fi un criteriu de validare, sortare sau identificare.

Un element suplimentar al proiectului este controlul LED-ului RGB prin PWM software, care permite aproximarea vizuală a nuanței detectate în modul de detecție simplă. În varianta finală, proiectul include și o funcție simplă de jurnalizare pe cardul microSD, folosită în modul de detecție simplă. Astfel, sistemul nu doar afișează culoarea detectată, ci poate păstra și un istoric minimal al detecțiilor cu ultima culoare.

Download

Codul sursă al proiectului este disponibil în repository-ul GitHub:

- [Repository GitHub ColorTrack](#)

Repository-ul conține:

- codul sursă al proiectului;
- fișierele header din folderul `include`;
- fișierele sursă din folderul `src`;
- fișierul `platformio.ini`;

Bibliografie/Resurse

Resurse Hardware

- Datasheet ATmega328P;
- Datasheet TCS230 Color Sensor;
- Documentație LCD 1602 cu interfață I2C;
- Documentație driver ULN2003;
- Documentație motor pas cu pas 28BYJ-48;
- Documentație modul microSD;

- Documentație protocol SPI;
- Documentație protocol I2C/TWI;
- Documentație USART pentru microcontrollere AVR.

Resurse Software

- Documentație avr-gcc;
- Documentație avr-libc;
- Documentație PlatformIO;
- Exemple pentru comunicație I2C/TWI pe AVR;
- Exemple pentru comunicație USART pe AVR;
- Exemple pentru controlul motoarelor pas cu pas;
- Exemple pentru implementarea PWM software pe microcontrollere AVR.

[Export to PDF](#)

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

Permanent link: http://ocw.cs.pub.ro/courses/pm/prj2026/vlad.radulescu2901/ioan_stefan.greere 

Last update: **2026/05/24 15:39**