

Alexandru LINCAN (78260) - Smart Line Follower

Autorul poate fi contactat la adresa: **Login pentru adresa**

Introducere

- Proiectul consta in realizarea unui **Line Follower** rapid care este capabil sa ocoleasca obstacolele de pe traseu.
- Un Line Follower este un robot care urmareste o linie neagra pe un fundal alb sau orice alta combinatie de culori care sa contrasteze. Ideea a venit de la numeroasele concursuri de roboti Line Follower care sunt foarte impresionante. Am vrut sa cresc complexitatea proiectului adaugand si functionalitatea de a ocoli obstacole.
- Acest tip de roboti au multe aplicatii practice cum ar fi roboti care sa transporte marfa in cadrul fabricilor sau depozitelor pe un traseu determinat de dinainte.

Descriere generală



Microcontroller-ul primește date de la senzorii de linie pentru a corecta direcția de mers astfel încât să poată urmări traseul. Senzorul de distanță este folosit pentru a detecta din timp obstacolele și a iniția procedura de ocolire. Pentru a controla cele două motoare, este nevoie de un driver de motoare. În algoritmul de urmărire a liniei folosesc un PID controller despre care voi vorbi mai mult în secțiunea Software.

Hardware Design

• Componente de baza:

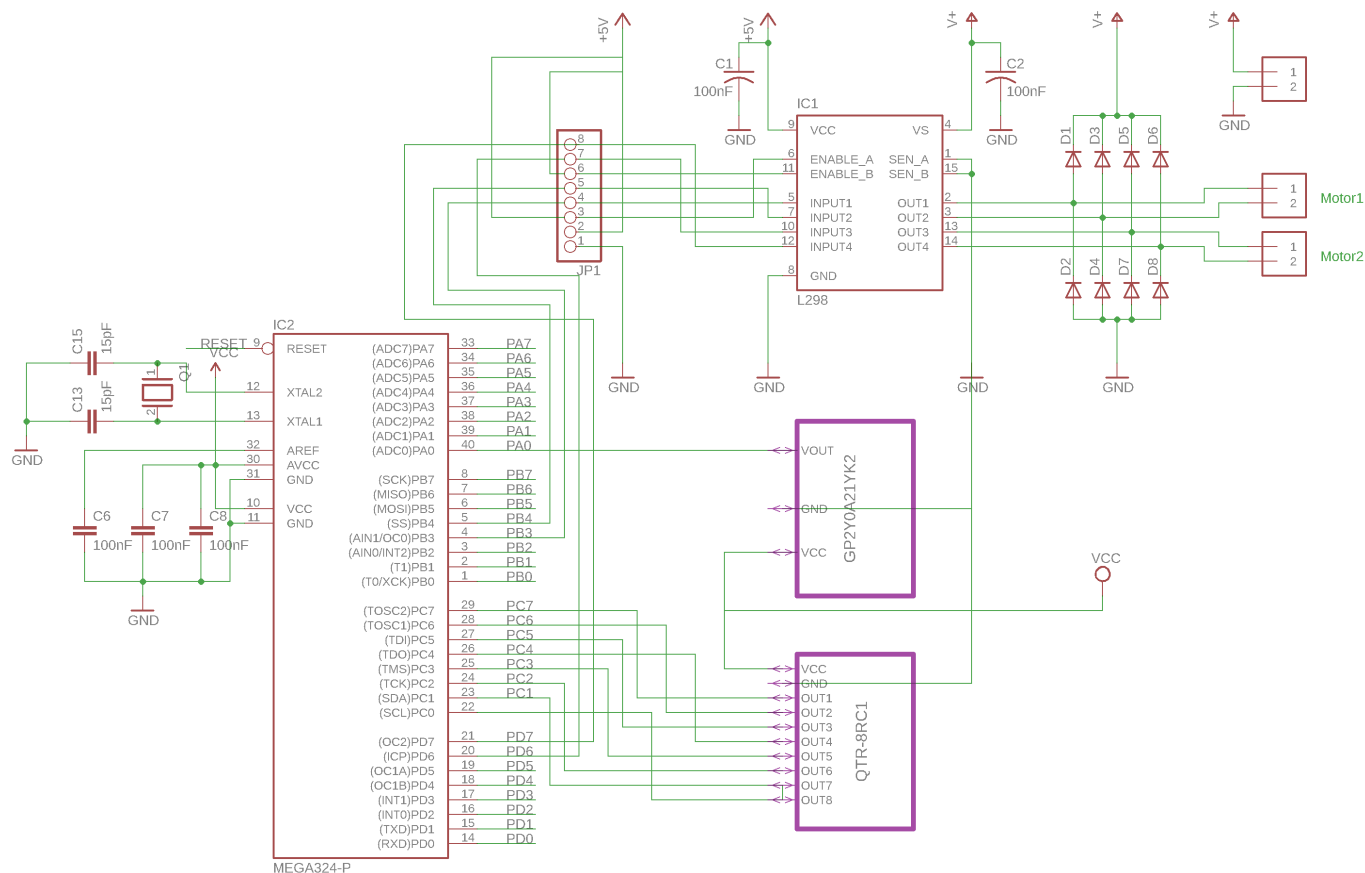
Componenta	Cantitate
Placuta PM	1
ATMEGA324A-PU	1
USB-B	1
Quartz 16MHz	1
Dioda Zener	2
LED 3mm Rosu	2
K1X10 WSL040	1

K2X10	1
Tasta	2
Rezistentă 100R	3
Rezistentă 470R	3
Rezistentă 1K5	1
Rezistentă 10K	1
Condensator 100nF	3
Condensator 15pF	2
Soclu Microcontroller	1
Jumper	1
7805CT	1
CONNECT JACK 2089	1
Condensator 100nF	1
Condensator electrolitic 16V 10uF	1

• **Componente suplimentare:**

Componenta	Cantitate
Motor electric micro metal 30:1 HP	2
Reflectance Sensor Array QTR-8RC	1
Senzor de distanta Sharp GP2Y0A21YK	1
Acumulator LiPo	1
Driver de motoare L298N	1
Roti aderente	1
Placa din balsa	1

• **Schema electrica:**

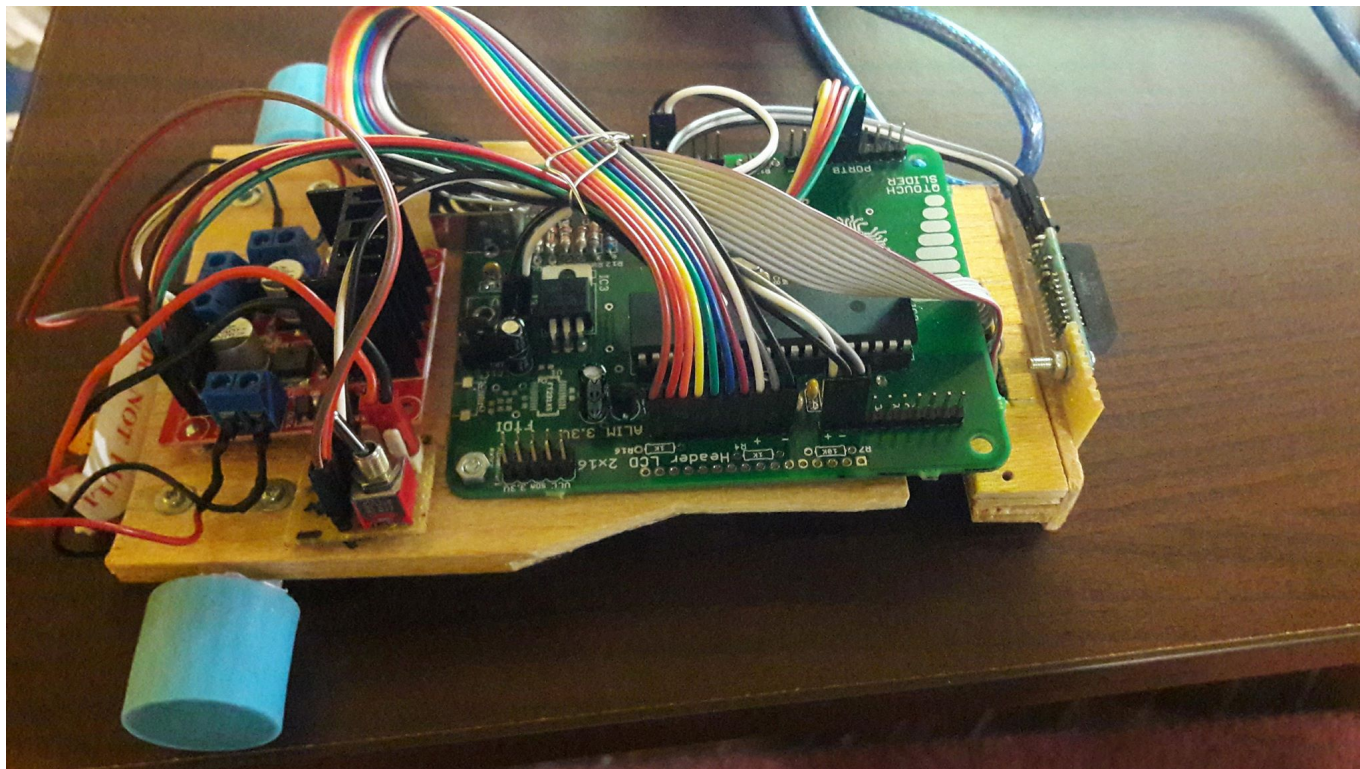


Software Design

- **Mediu dezvoltare:** Windows 10, Programmer's Notepad, WinAVR(compilare), HIDBootFlash pentru incarcarea codului pe placuta.
- **Control motoare:** Pentru controlul motoarelor am folosit pwm pe timer 1 si registrii OCR1A si OCR1B.
- **Senzor linie:** Pentru a citi senzorii digitali de linie am folosit timer 2 si m-am inspirat din biblioteca pololu pentru orangutan din bibliografie.
- **Senzor distanta:** La citirea senzorului de distanta am folosit ADC.
- **PID:** Pentru a realiza un algoritm de urmarire a liniei cat mai bun, am folosit modelul matematic PID care se bazeaza pe cele 3 componente:
 - Proportional - masoara cat de departe se afla robotul de linie in functie de citirile de la senzorii de linie
 - Integrator - masoara eroarea acumulata in timp, adica suma diferentei dintre centrul senzorului de linie si centrul liniei de la fiecare moment de timp
 - Derivator - este rata de deplasare a robotului de la stânga la dreapta sau de la dreapta la stânga
- Aceste 3 componente adunate dau controlul robotului fiecare avand un impact asupra lui in functie de cele 3 constante: Kp, Ki si respectiv Kd ce trebuiesc setate la niste valori cat mai bune ce se pot obtine din multe teste
- **Evitare obstacole:** In momentul in care senzorul de distanta detecteaza unui obstacol, voi initia rutina de ocolire a acestuia ce presupune deplasarea robotului in afara pistei si inapoi pe aceasta, pe langa obstacol.

Rezultate Obținute

Un robotel care urmareste linia si se opreste cand intalneste obstacole.



Concluzii

- Pentru a nu avea probleme cu inertia in momentul virajului la curbe, am incercat sa atasez componentele de un cadru cat mai mic si cat mai usor (din lemn de balsa). Acest lucru s-a dovedit dificil deoarece placuta de PM are dimensiuni relativ mari si este grea.
- Tunarea constantelor K_p , K_i si K_d de la PID a durat mult timp deoarece a fost realizata prin trial and error.
- Alegerea motoarelor s-a dovedit dificila deoarece acestea trebuie sa fie mici si usoare (pentru a se potrivi pe cadrul de lemn) dar in acelasi timp sa fie si puternice si rapide pentru ca robotul sa termine traseul cat mai rapid.

Download

Arhiva proiect: [proiect_alexandru_lincan.zip](#)

Bibliografie/Resurse

- http://cs.curs.pub.ro/wiki/pm/_media/doc8272.pdf
- <https://github.com/pololu/libpololu-avr>
- https://en.wikipedia.org/wiki/PID_controller
- Documentația în format [PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2018/cpetrisor/celmaismecherid>



Last update: **2021/04/14 15:07**