# Liana-Maria PĂNĂTĂU (67010) - Line Follower

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

## Introducere

Ce face?

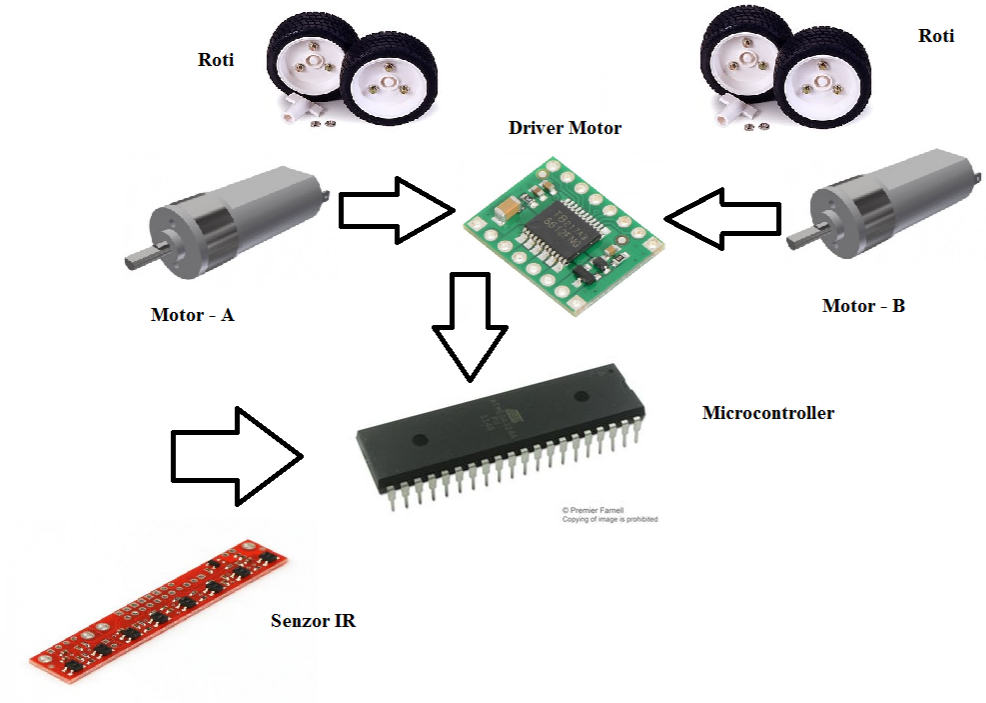
Mini robotelul meu va fi capabil sa urmareasca o traiectorie reprezentata de o linie. Micul robotel va fi compus din doua roti, un driver pentru coordonare roti, un senzor de culoare ce va detecta iesirea de pe linia neagra si unitatea principala de control microcontrollerul AtMega324A.

Care este scopul lui?

Scopul robotelului este de a reusi sa se ghideze si sa reuseasca sa parcurga un drum marcat. Ideea de a realiza un line follower a venit in urma dicutiei cu asistentul de la laborator. In primul rand, m-a atras faptul ca puteam realiza in sfarsit, pe cont propriu, un robot asemanator cu cele vazute in anul I si de care am ramas fascinata! In al doilea rand, mi-am dat seama ca pornind de la un simplu line follower se pot realiza multe alte dispozitive pe aceeasi idee. So...Who knows? Poate aceasta este doar un inceput! :)

## Descriere generală

Schema bloc:




# Hardware Design

Lista de piese:

1. Driver Motoare
2. Motorase
3. Suport Prinderi Motorase
5. Bara Senzori IR
6. Roti
7. Ball Caster pentru sustinere Robot
8. Rezistente
9. ATmega
10.Diode 11. Placuta de baza
12. Condensator
13. Baterie
14. Fire mama-mama
15. Fire tata-tata
16. Pini

Schema electrica:

# Software Design

Pentru implementarea codului aplicatiei am folosit ca mediu de dezvoltare AVR Studio.
Biblioteci folosite: avr/interrupt.h, util/delay.h, stdio.h, stdlib.h
Structura codului este urmatoarea:

```
  Fiecare motor este identificat ca MOTOR_A si MOTOR_B. Am definit, de asemenea, si urmatoarele stari de miscare ale robotelului: FORWARD, FLOAT, BREAK, BACKWARD.
  Avand acesti identificatori, am implementat urmatoarele metode:
     - ADC_init(void) si ADC_get(uint channel) -> metode preluate din solutia laboratorului 5;
     - initPWM care initializeaza Timer0 in modul Fast PWM, cu topul 0xFF, modul non-inverting si prescaler 0;  am ales sa folosesc Timer0 pentru ca 8 biti sunt suficient pentru setarea unui pwm dorit;
     - setMotorDirection(Motor motor, Direction dir) care seteaza pentru un anumit Motor: A, B, o anumita directie: FLOAT, BREAK, FORWARD, BACKWARD.
     - initGPIO(void) initializeaza toate porturile de care am nevoie si anume: PORT B -> pentru directia motoarelor si pwm(PB3, PB4); PORTA -> pentru senzori;
     - setMotorPwm(Motor motor, unsigned int speed) seteaza o anumita viteza pentru un Motor;
In main este implementata logica dispozitivului. Dupa initializarea ADC-ului, a pwm-ului si stabilirea directiei si vitezei initiale, in functie de rezultatele intoarse de senzori, robotelul se misca stanga sau dreapta pentru a se mentine pe mijlocul dungii negre.
```

# Rezultate Obţinute

Rezultatele obtinute in urma realizarii proiectului au fost urmatoarele:
- am reusit sa implementez functionalitatea dorita;
- robotelul se misca pe o dunga neagra si reuseste sa o urmareasca;
- am invatat sa fac debug pe componente hardware;
- am inteles notiunile pe care nu le aveam clar lamurite de la laborator;
- am invatat sa citesc foarte bine un datasheet;

# Concluzii

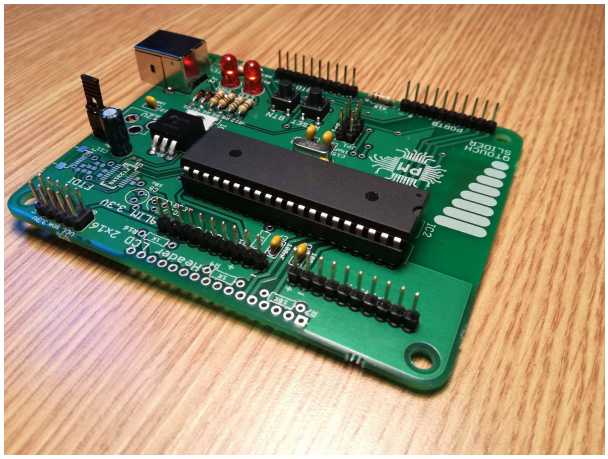
A fost o experienta unica, din care am avut foarte multe de invatat. Mi-a placut faptul ca am aplicat teoria de la laborator intr-o aplicatie practica, nu cu scop didactic. Iar la final, satisfactia ca robotelul se misca a fost geniala!

# Download

Cod sursa + Makefile + README
pm-2017-versiuneafinala.zip

# Jurnal

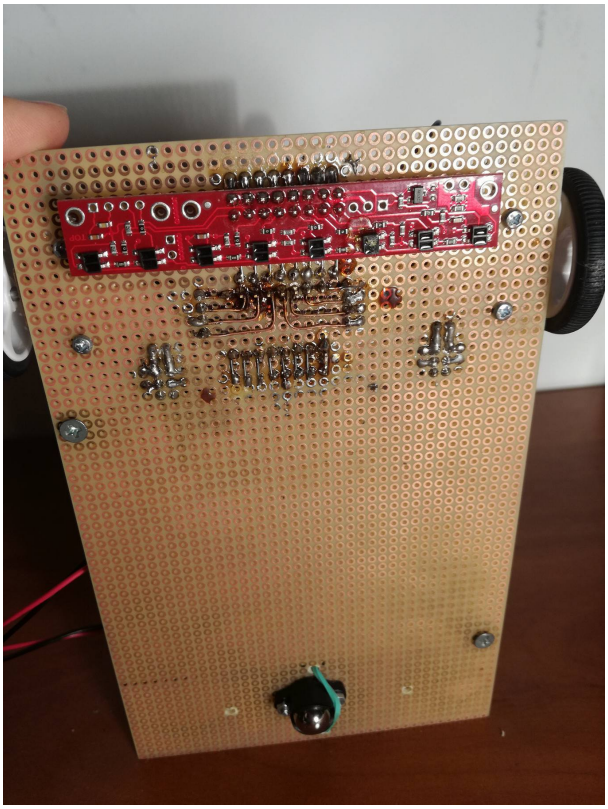
1. Am inceput sa lipesc componentele de baza ale placutei.

Placuta de baza: 

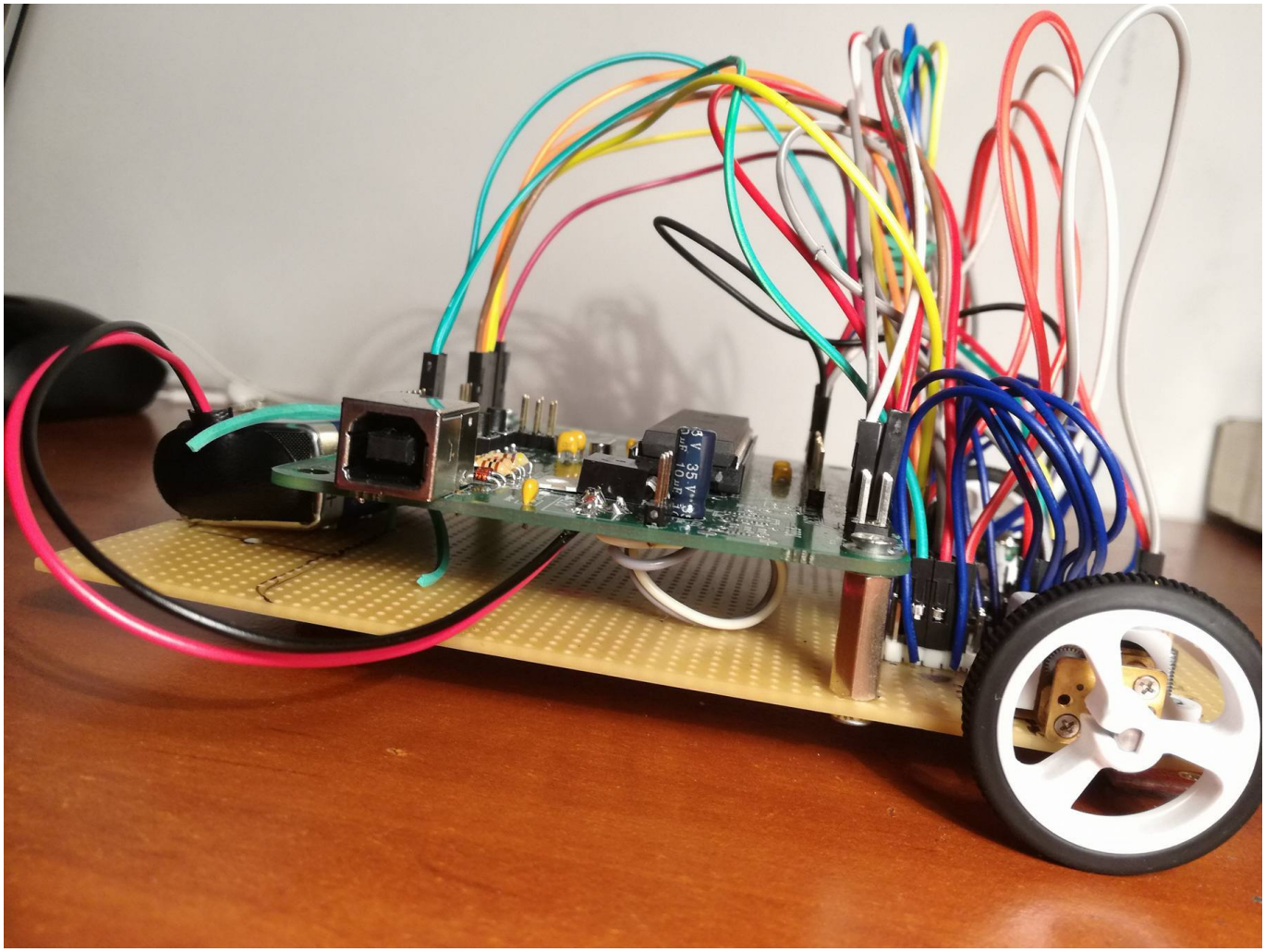
2. Am lipit si celelalte componente si am legat componentele la microcontroller.

Placuta de baza(Update):
Link(video): → https://drive.google.com/file/d/0ByKCyRGQF6A0RkkxVmJaTjlSMVU/view?usp=sharing

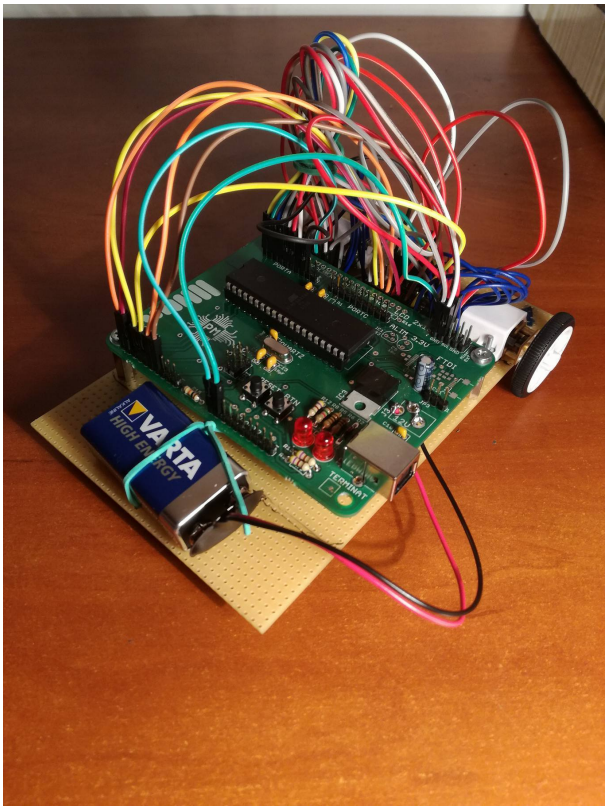
4. Am invatat sa folosesc senzorii :D



3. Am reparat rotile care nu erau lipite tocmai cum trebuie:



Robotelul la final:



# Bibliografie/Resurse

1. Resurse software:

- Datasheet ATmega324
- http://www.electroschematics.com/9941/avr-pwm/
- https://www.pololu.com + pagina producatorului
- http://cs.curs.pub.ro/wiki/pm/lab3;
- http://cs.curs.pub.ro/wiki/pm/lab5

2. Resurse hardware:

- https://www.robofun.ro/mecanice/driver/driver_dc/driver-TB6612FNG
- https://www.robofun.ro/mecanice/motoare/micro_metal/50-1-micro-metal-motor-hpcb-6v-with-extended-motor-shaft
- https://www.robofun.ro/mecanice/sasiu/suport_motor_micro_extins
- https://www.robofun.ro/mecanice/mecanice-encoder/optical_encoder
- https://www.robofun.ro/senzori/infrarosu/bara_senzori_linie_qtr_8a
- https://www.robofun.ro/mecanice/roti_senile/roti_32_7mm
- https://www.robofun.ro/mecanice/roti_senile/ball_caster_9_5mm

* Documentația în format PDF

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

Permanent link:
**https://ocw.cs.pub.ro/courses/pm/prj2017/dnovischi/linefollower2017**

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