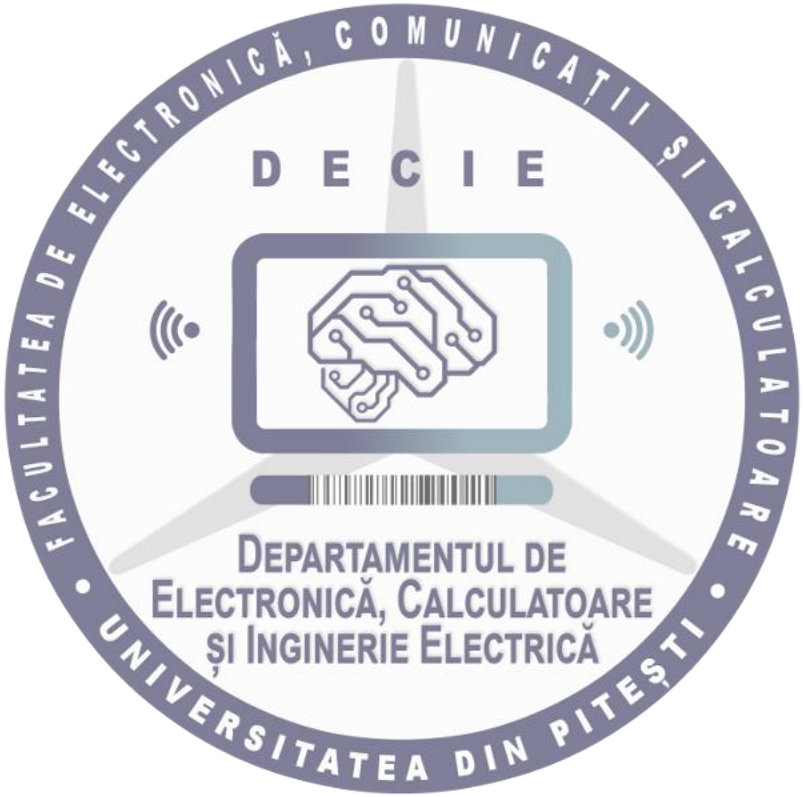
Curs 6. Senzori

prezentat de Adrian Mordachescu



Tipuri de senzori

senzori de camp electromagnetic

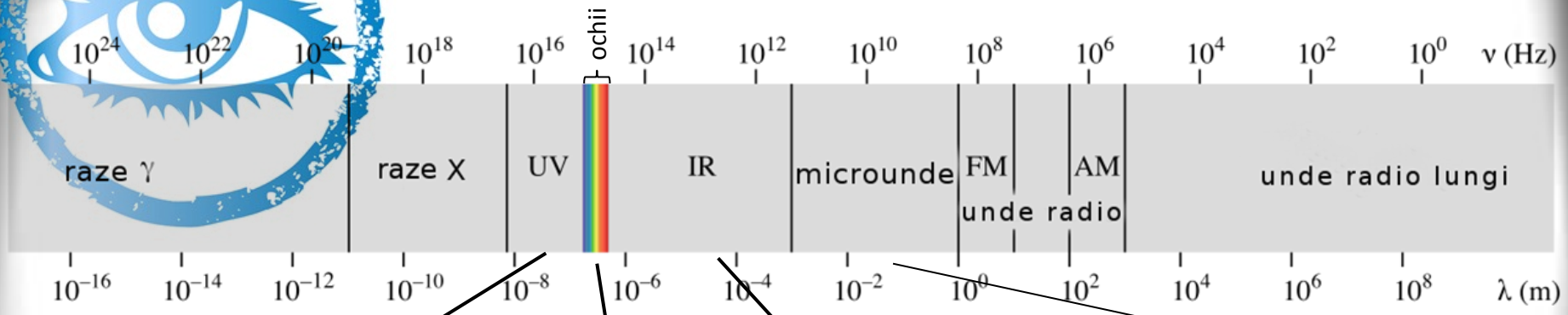
senzori acustici, de inclinare, de acceleratie

senzori gaz

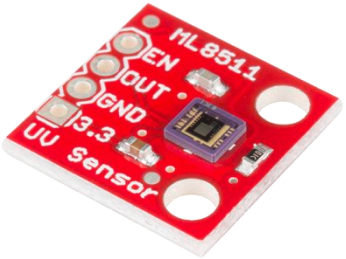
senzori chimici

senzori tactili, de apasare, vibratii mecanice, de temperatura, umiditate

Senzori de camp electromagnetic



Sparkfun ML-8511



67 RON

TCS34725

54 RON

52 RON

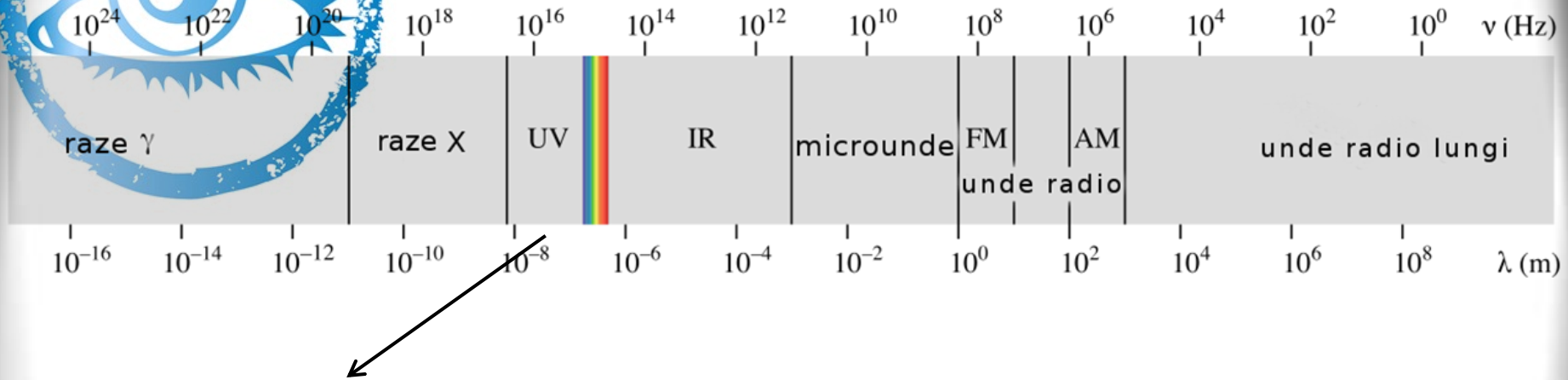
QTR-8 Pololu

SKU: SEN0192

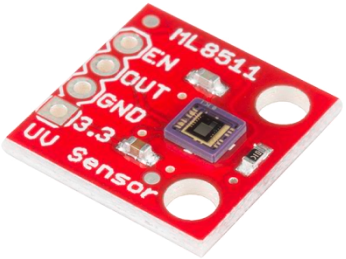
80 RON

Sursa preturi: https://www.robofun.ro

Senzori UV



Sparkfun ML-8511



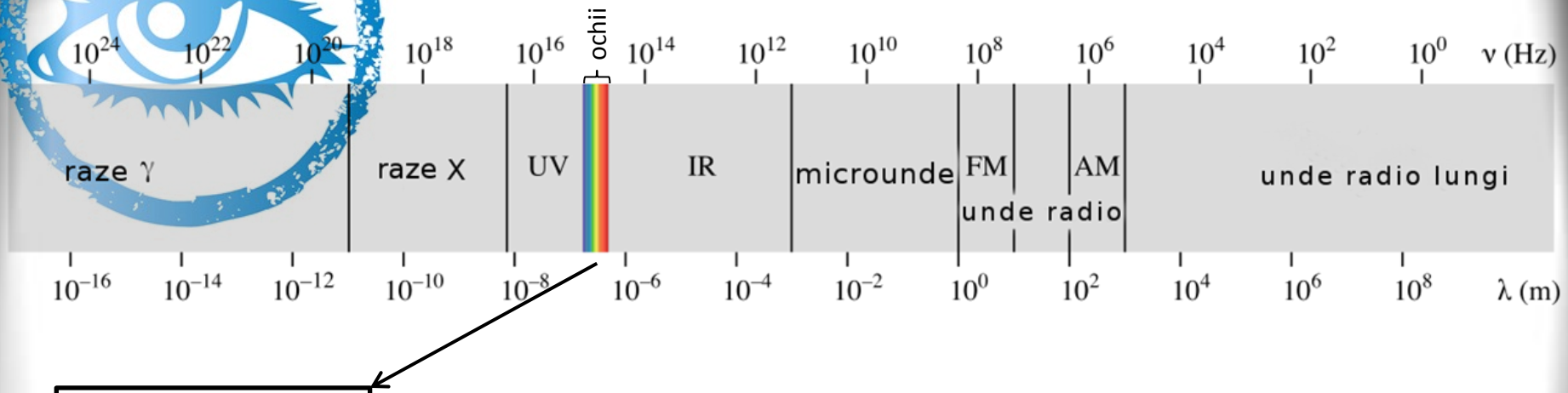
67 RON

Sparkfun ML-8511

Aplicatii:

- Detectia indicelui UV
- Avertizarea utilizatorului de pericolul arsurilor solare

Senzori RGB



TCS34725

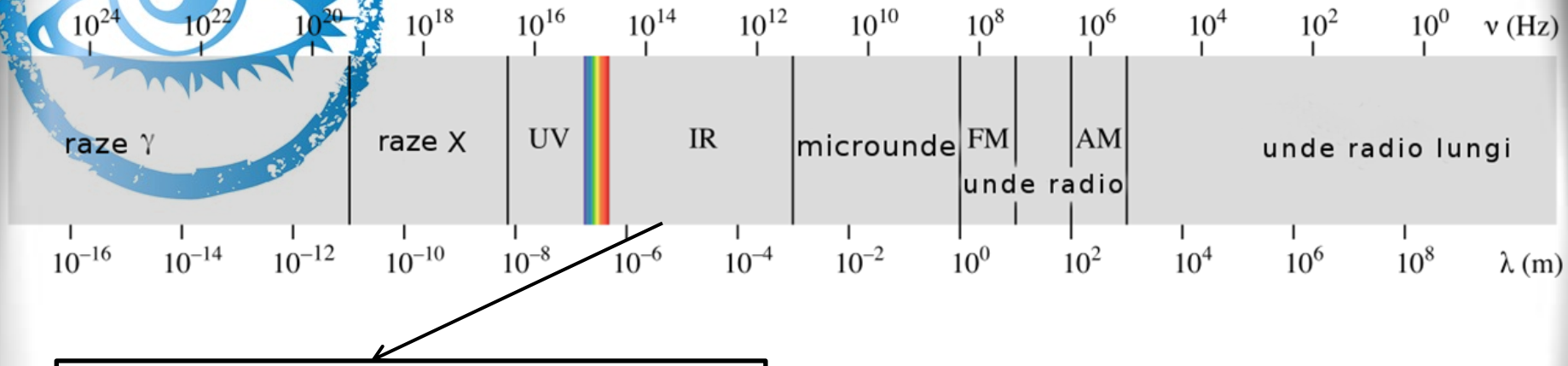
54 RON

TCS34725

Aplicatii:

- Masurarea temperaturii de culoare
- Controlul calitatii LED-urilor
- Transmiterea de informatii unui robot folosindu-ne de culori

Senzori IR

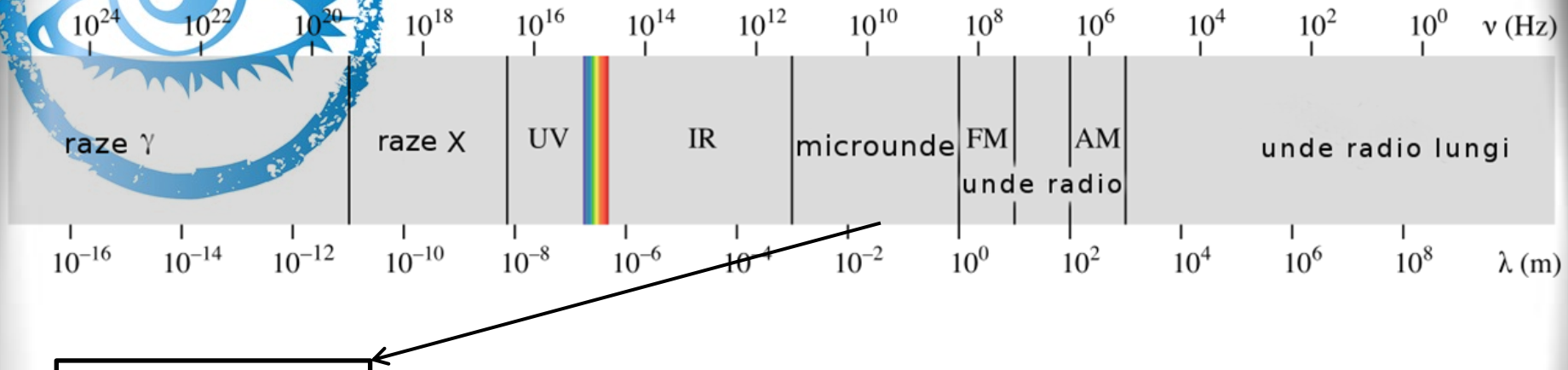


QTR-8 Pololu 52 RON

Aplicatii:

- Urmarirea liniei
- Masurarea reflectivitatii
- Senzor de proximitate
- Harta termica

Senzori microunde



SKU: SEN0192

80 RON

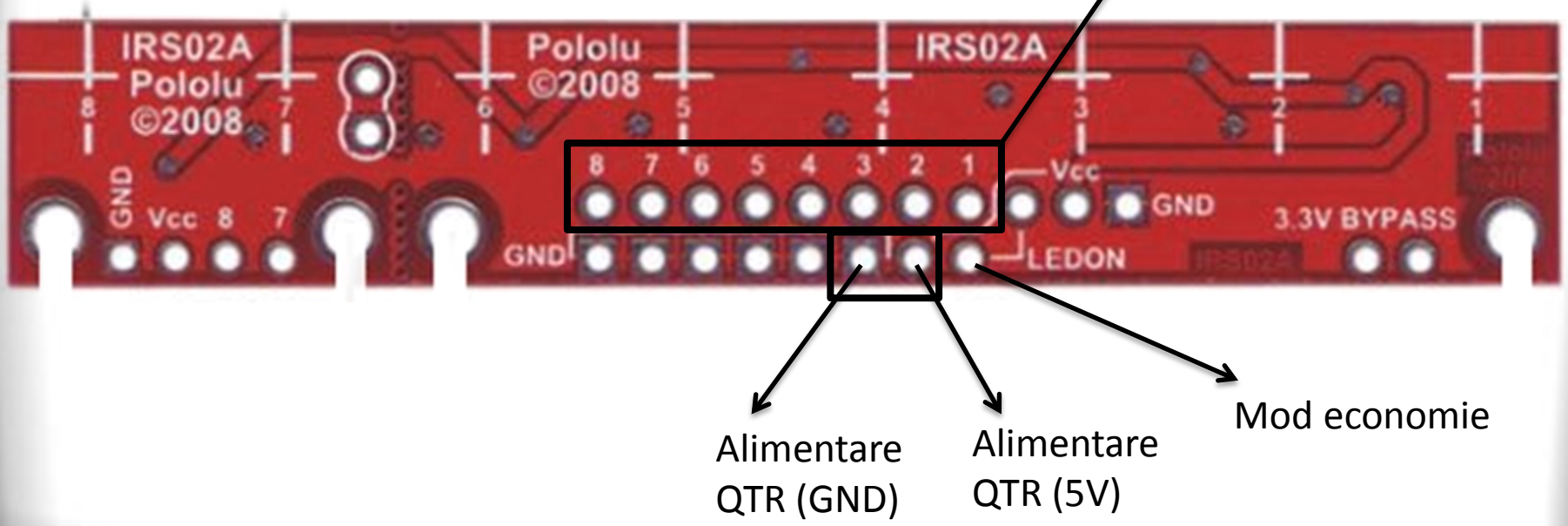
Aplicatii: SKU: SEN0192

- Senzor de parcare
- Masurarea vitezei
- Senzori de miscare usa

Senzori IR

QTR-8RC IR Reflectance Sensor – vedere de sus

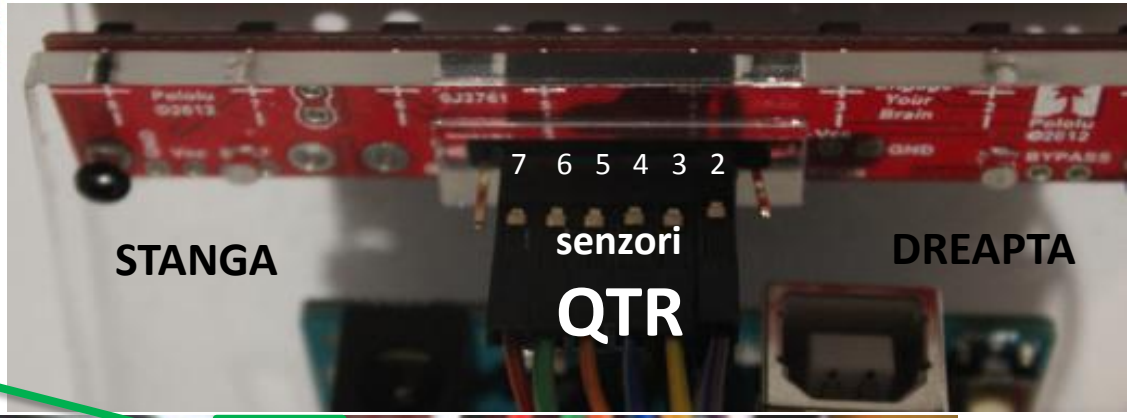
ASAMBLARE QTR



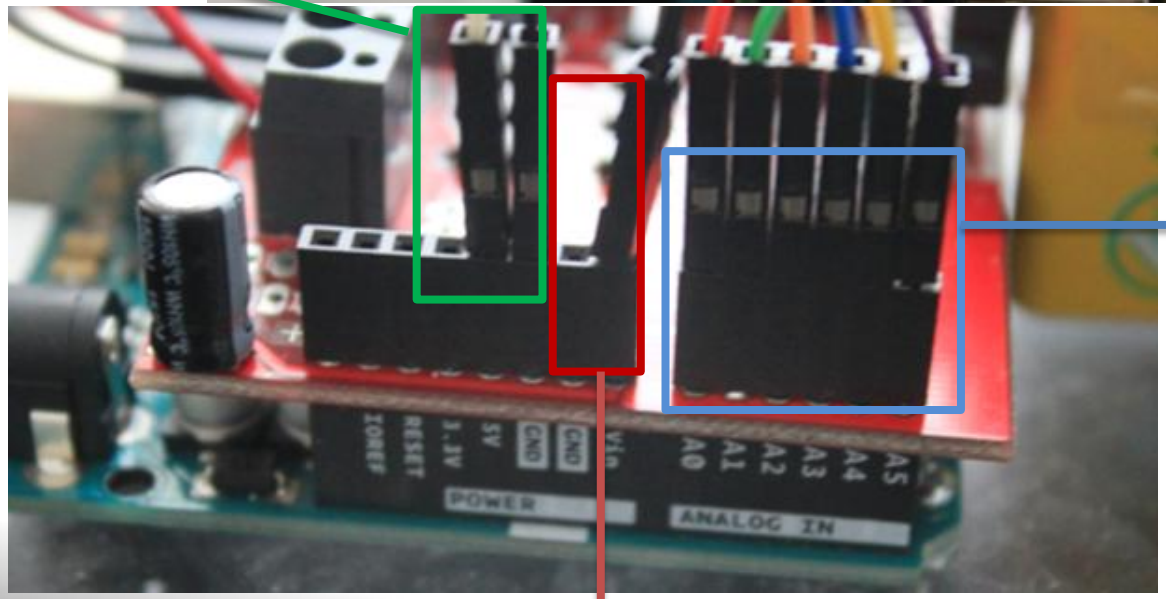
Senzori IR

QTR-8RC IR Reflectance Sensor – vedere de sus

ASAMBLARE



Alimentare QTR



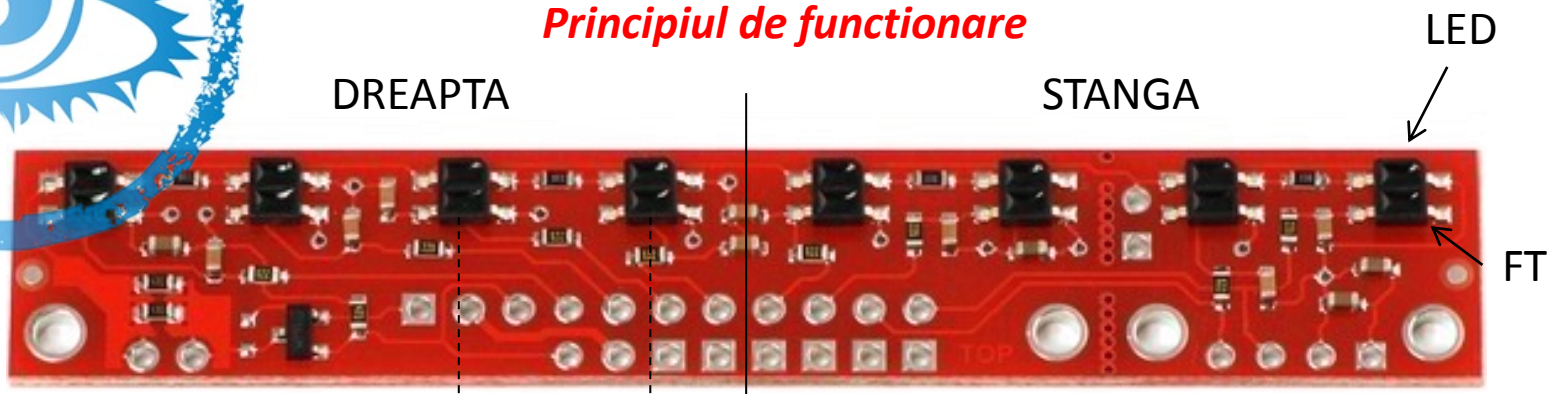
Senzori QTR

Alimentarea Arduino din baterie externa

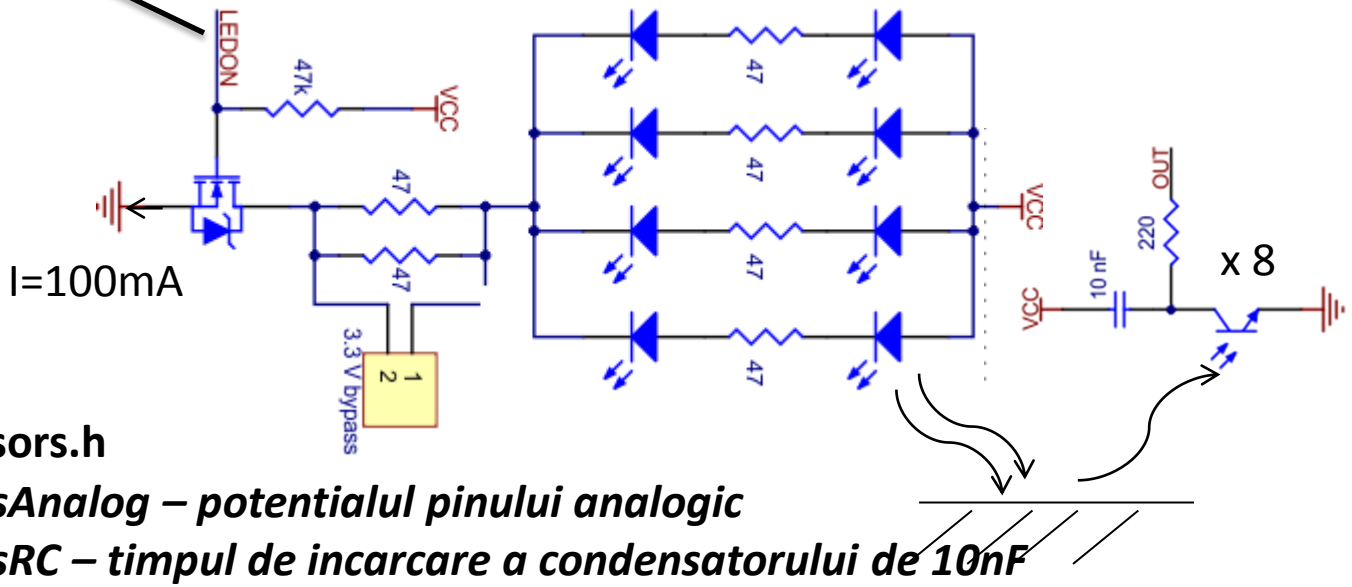
Senzori IR

QTR-8RC IR Reflectance Sensor – vedere de jos

Principiul de functionare



Mod economie



Librarie: QTRSensors.h

Clase: ***QTRSensorsAnalog – potentialul pinului analogic***

QTRSensorsRC – timpul de incarcare a condensatorului de 10nF (trecerea pinului digital din 1 in 0)

Senzori IR

QTR-8RC IR Reflectance Sensor

Pentru instalarea librariei:
a. Downloadati ultima versiune a librariei:
https://github.com/pololu/qtr-sensors-arduino/releases
b. Redenumiti folderul
"qtr-sensors-arduino-xxxx" in "QTRSensors"
c. Mutati folderul "QTRSensors" in folderul "libraries"
din locatia dosarului cu schite (Fisier -> Preferinte)

Librarie: QTRSensors.h

Clase:

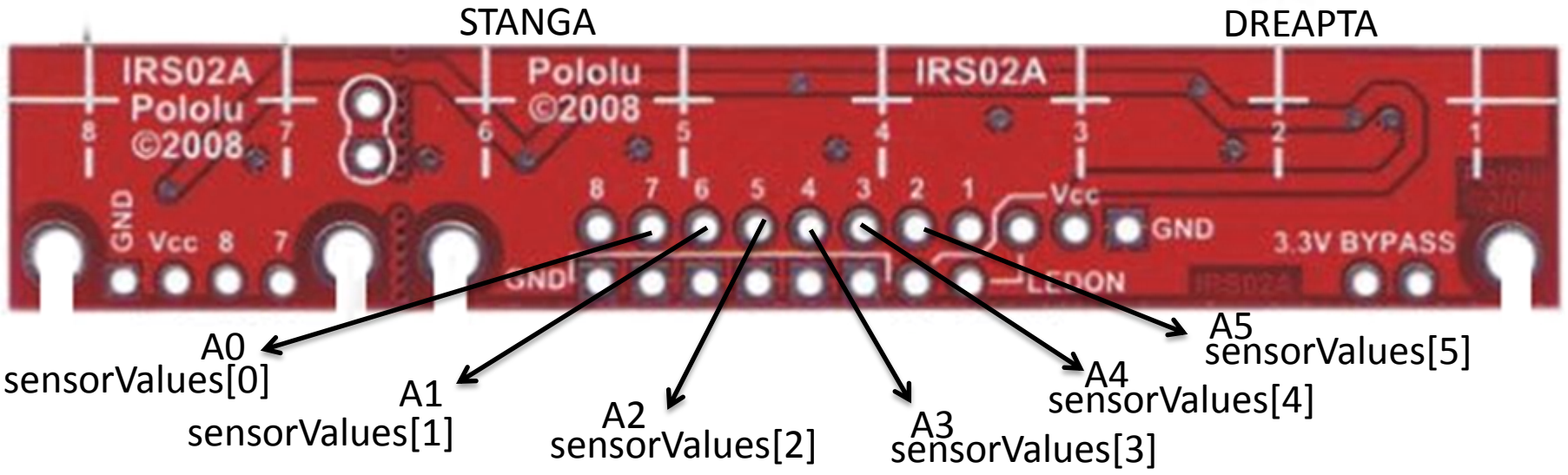
QTRSensorsAnalog *– potentialul pinului analogic (pini analogici 0-5)*
QTRSensorsRC *– timpul de incarcare a condensatorului de 10nF*
(trecerea pinului digital din 1 in 0: pini digitali 0-13 + pini analogici)

Metode:

- *void **read**(sensorValues) -> direct iesirea senzorilor: valori mari la reflexie scazuta*
- *int **readLine**(sensor Values) -> datale normalizate de la senzori si estimarea pozitiei liniei*
- *void **calibrate**() -> calibrarea inaintea de utilizarea metodei readLine*

Senzori IR

QTR-8RC IR Reflectance Sensor - Clasa ***QTRSensorsAnalog***



Clasa ***QTRSensorsAnalog***

```
#include <QTRSensors.h>
#define NUM_SENSORS             6
#define NUM_SAMPLES_PER_SENSOR  4
#define EMITTER_PIN             QTR_NO_EMITTER_PIN        // LED-urile vor fi mereu pornite

QTRSensorsAnalog ir((unsigned char[]) {0, 1, 2, 3, 4, 5},  NUM_SENSORS, NUM_SAMPLES_PER_SENSOR, EMITTER_PIN);
unsigned int sensorValues[NUM_SENSORS];
----
void loop() {
        ir.read(sensorValues);
        // valorile de la senzori se vor memora in tabloul sensorValues
}
```

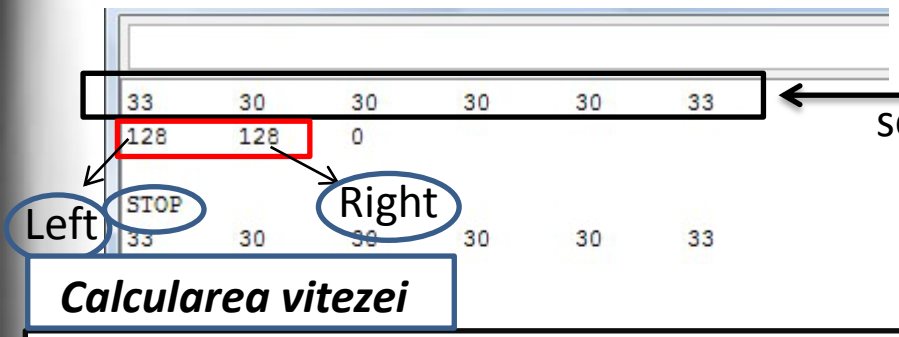
Pinii analogici — *Numarul de senzori folositi* — *Numarul de esantioane mediate pentru o singura citire* — *PIN LED*

Senzori IR

QTR-8RC IR Reflectance Sensor - ***QTRSensorsAnalog***

```
for (unsigned char i = 0; i < NUM_SENSORS; i++)  {
        Serial.print(sensorValues[i]);
        Serial.print('\t');
}
Serial.println();
```

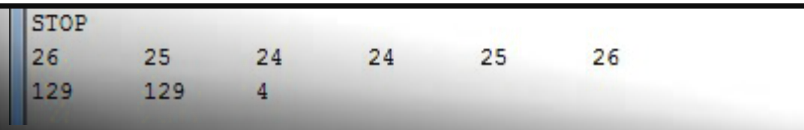
Depanarea

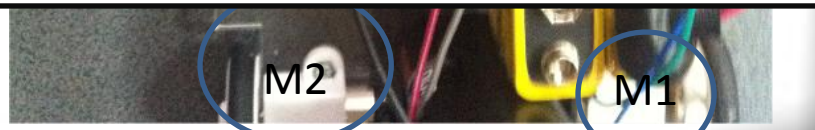


senzori

Calcularea vitezei

Cea mai simpla comanda de corectie este cea in care viteza unui motor este proportionala cu reflexia detectata de senzorii de pe partea lui.
Viteza maxima se obtine atunci cand toti senzorii de pe o parte detecteaza albul absolut.





Senzori IR

QTR-8RC IR Reflectance Sensor - ***QTRSensorsAnalog***

```
STOP
789     731     753     773     750     805
46      44      252

STOP
789     731     753     773     751     805
46      44      253

STOP
789     731     753     773     751     805
46      44      256
```

Calcularea vitezei

Viteza motorului scade atunci cand exista mai putina reflexie de la senzorii de pe partea lui, precum in poza alaturata in care senzorii sunt in gol.

Senzori IR

QTR-8RC IR Reflectance Sensor - ***QTRSensorsAnalog***

```
1020    1020    1020    1020    1020    1020
22      22      36

STOP
1019    1019    1020    1020    1019    1019
22      22      37

STOP
1020    1020    1020    1021    1020    1020
```

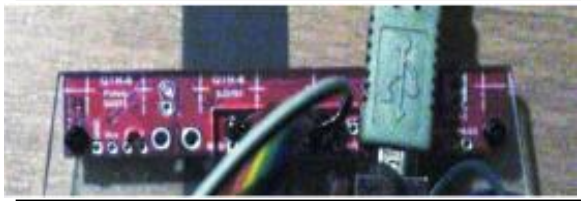
Calcularea vitezei

Viteza minima se obtine atunci cand toti senzorii de pe o parte detecteaza negrul absolut.

Senzori IR

QTR-8RC IR Reflectance Sensor - ***QTRSensorsAnalog***

$S_{0(left)}$ Speed Left	$S_{1(left)}$ Speed Right	$S_{2(left)}$ cnt	$S_{3(right)}$	$S_{4(right)}$	$S_{5(right)}$
42 99	39 97	619 0	709	43	44



48 56	534 126	671 0	47	37	43

718 14	629 126	41 0	38	37	42
717 12	653 126	42 0	39	37	44

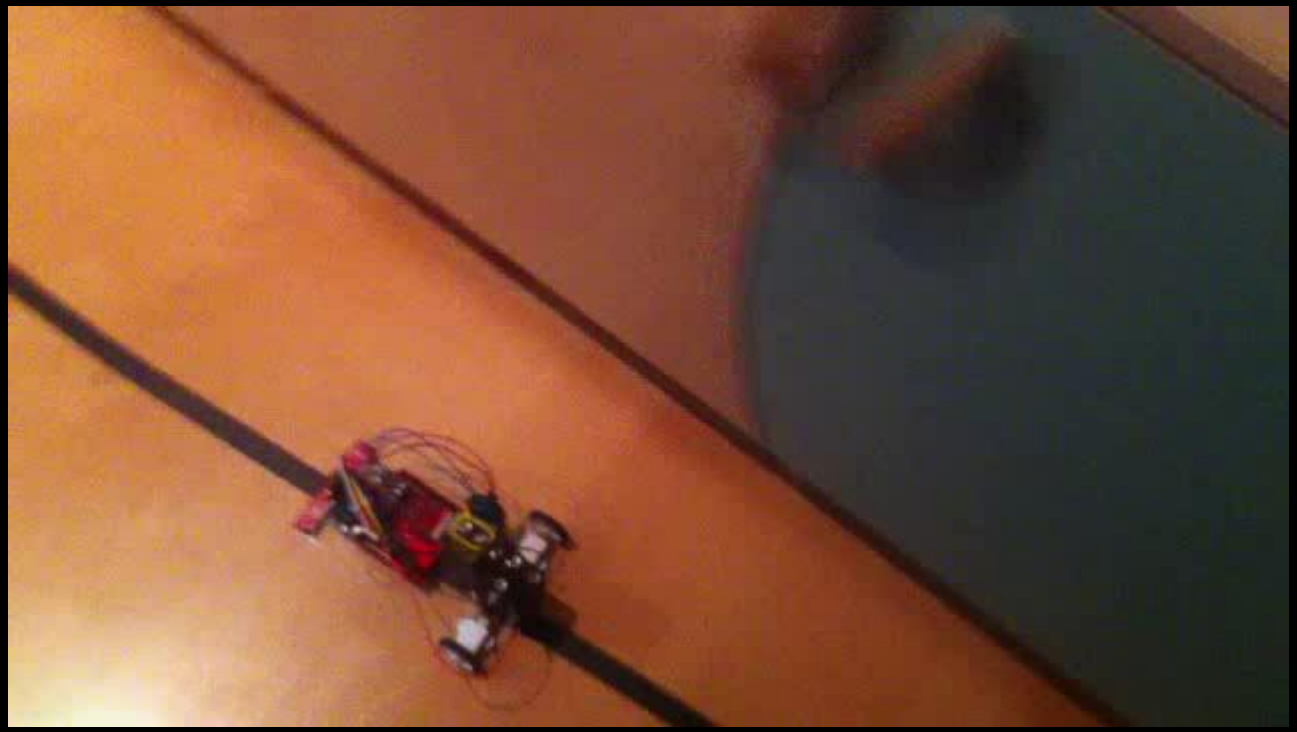
Calcularea vitezei

O corectie mai importanta se poate obtine daca dam o pondere mai mare negrului detectat de senzorii din capete.

41 126 STOP	35 126	36 11	38	37	41

QTR-8RC IR Reflectance Sensor – ***QTRSensorsAnalog***
(video: https://youtu.be/jJ-eyE_z81U)

```
int VITEZA_MAX = 135;
int CORECTIE_MAX = 100;
int CORECTIE_MED = 80;
int CORECTIE_MIN = 50;
```

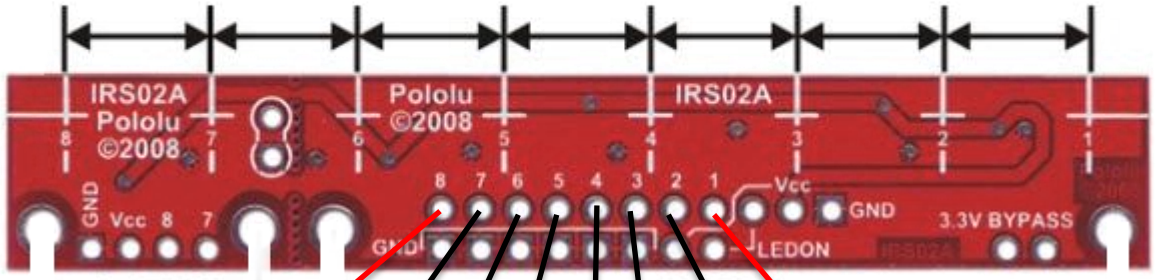


Senzori IR

QTR-8RC IR Reflectance Sensor – Clasa ***QTRSensorsRC***

pozitia centrului benzii negre

poz: 0 1000 2000 3000 4000 5000



STANGA

DREAPTA

Puteti cupla si acest senzor la un pin digital

A0 A1 A2 A3 A4 A5

Puteti cupla si acest senzor la un pin digital

Clasa ***QTRSensorsRC***

```
#include <QTRSensors.h>

QTRSensorsRC  ir((unsigned char[]) {14, 15, 16, 17, 18, 19}, 6);
unsigned int sensorValues[NUM_SENSORS];

void loop() {
  ---
  int poz = ir.readLine(sensorValues);
  ---
}
```

Numarul de senzori folositi

Pinii digitali 14-19 sunt de fapt pinii analogici 0-5

PENTRU ACEASTA METODA ESTE NECESARA CALIBRAREA!!!

Senzori IR

QTR-8RC IR Reflectance Sensor – Clasa ***QTRSensorsRC***

Calibrarea

```
void setup() {
 ----
 int i;
 for (i = 0; i < 250; i++)  // calibrarea dureaza in acest caz 5 secunde
                    {
                    ir.calibrate();
                    delay(20);
                    }
---
}
```

Senzori	S0		S1		S2		S3		S4		S5	
Calibrare	max	min	max	min	max	min	max	min	max	min	max	min
Dinamica	4000	192	4000	120	4000	116	4000	116	4000	156	4000	196
Statica	192	192	116	152	116	116	384	388	4000	4000	1444	1492

Senzori IR

QTR-8RC IR Reflectance Sensor – Clasa ***QTRSensorsRC***

Corectia pozitiei

speed_Left=speed_MED + speedCorrection
speed_Right=speed_MED – speedCorrection
speedCorrection > 0 -> deplasare spre dreapta
speedCorrection < 0 -> deplasare spre stanga

	$S_{0(left)}$	$S_{1(left)}$	$S_{2(left)}$	$S_{3(right)}$	$S_{4(right)}$	$S_{5(right)}$
	Poz	speedCorrection				
	speed_Left	speed_Right				
	0 3074 97	9 17 83	39	1000	80	8
	1000 116 4	132 -71 146	0	0	0	0
	2 0 0	0 -75 150	0	0	1	1
	1 5000 150	0 75 0	0	0	0	0

Senzori IR

QTR-8RC IR Reflectance Sensor – Clasa ***QTRSensorsRC***

LINII DE COD

Initializare

```
int VITEZA_MED = 75;
double KP=0.030;
double KD=0.030;
```

Calcul viteze

```
void loop() {
 ----
  int poz = ir.readLine(sensorValues);
 // eroarea e data de diferenta dintre poz si centrul senzorilor
  int error = poz - 2500;

  // calcularea coeficientului de corectie speedCorrection:
  int speedCorrection = round(KP * error + KD * (error - lastError));
  lastError = error;

  speed_Left = VITEZA_MED + speedCorrection;
  speed_Right = VITEZA_MED - speedCorrection;
 ---
}
```

Senzori IR

QTR-8RC IR Reflectance Sensor – Clasa ***QTRSensorsRC***

LINII DE COD

```
if(abs(speed_Left)<100 && abs(speed_Right)<100) {
    if (speed_Left>0) speed_Left=(int) round(speed_Left+(100-speed_Left)*0.6);
                      else speed_Left=(int) round(speed_Left-(speed_Left-100)*0.6);
    if (speed_Right>0) speed_Right=(int) round(speed_Right+(100-speed_Right)*0.6);
                      else speed_Right=(int) round(speed_Right-(speed_Right-100)*0.6);
}
if(speed_Left>255) speed_Left=255;
if(speed_Right>255) speed_Right=255;
if(speed_Left<-255) speed_Left=-255;
if(speed_Right<-255) speed_Right=-255;
```

Corectie viteze prea mici sau prea mari

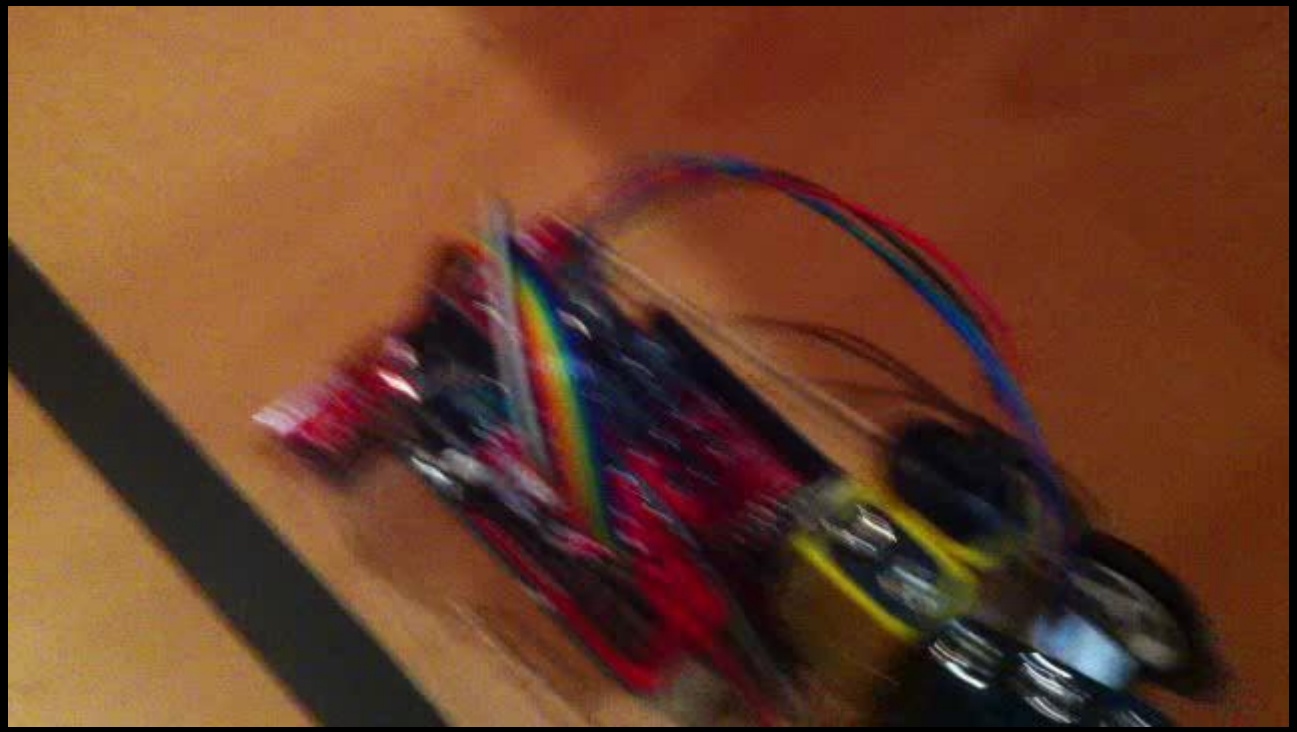
```
If ((poz==0) || (poz==5000)) {
   cnt++;
   if(cnt>40){
              speed_Left=0;
              speed_Right=0;
   }
} else {
   cnt=0;
}
```

Oprire motoare

QTR-8RC IR Reflectance Sensor – **QTRSensorsRC**

(video 1: https://youtu.be/hjvEpLhFzZI)

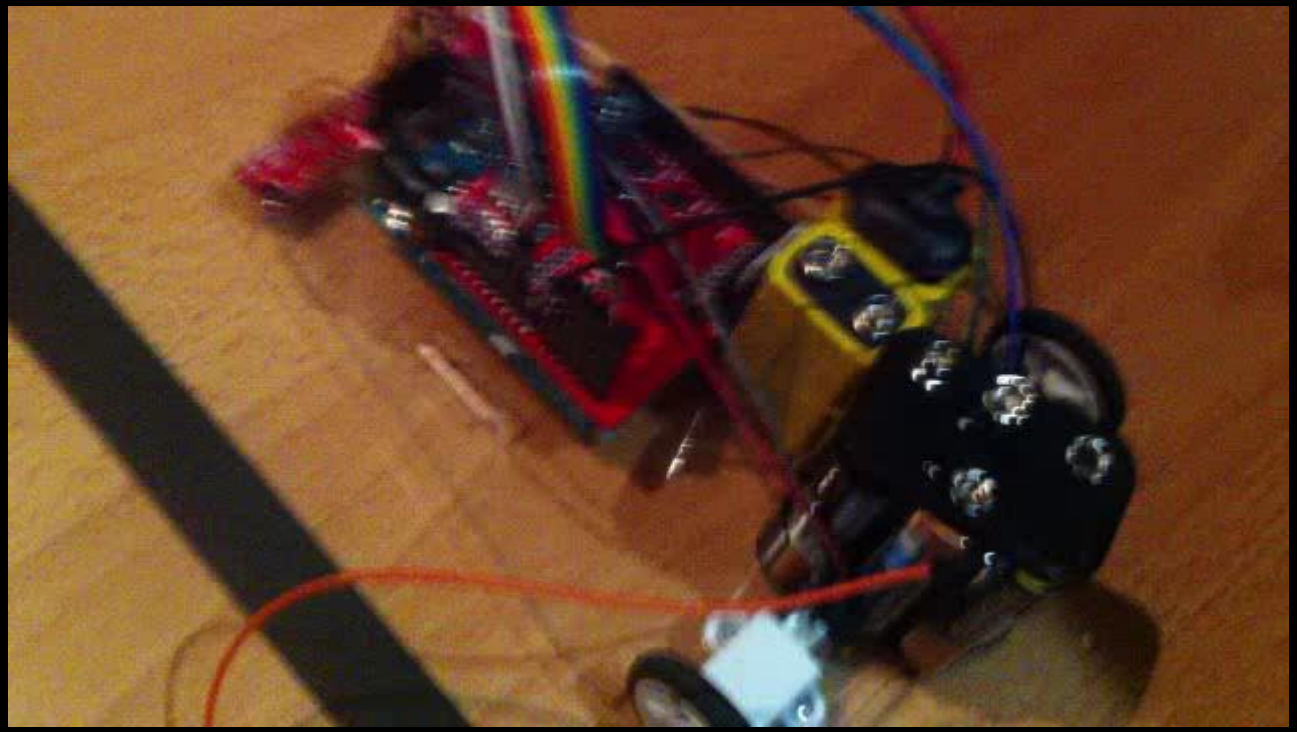
VITEZA_MED = 75
KP = 0,010
KD = 0,100



QTR-8RC IR Reflectance Sensor – **QTRSensorsRC**

(video 2: https://youtu.be/593F5iLoOWw)

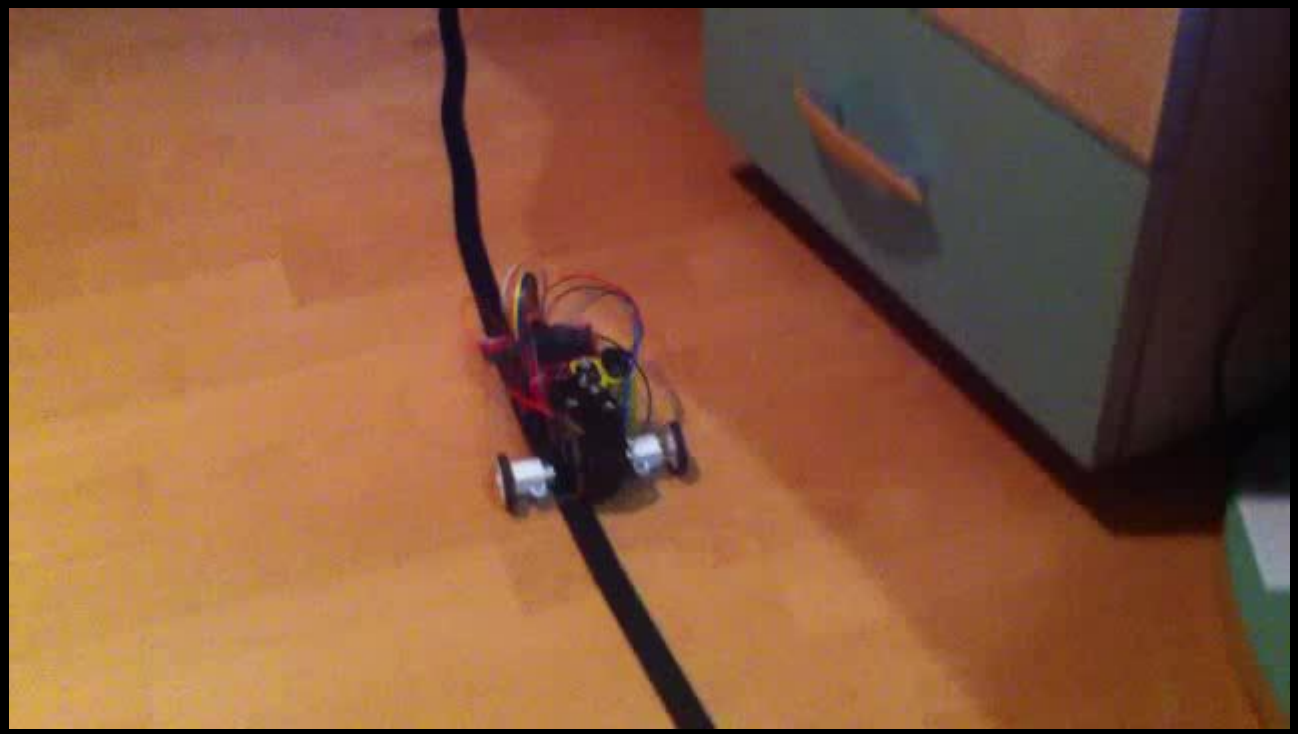
VITEZA_MED = 75
KP = 0,020
KD = 0,100



QTR-8RC IR Reflectance Sensor – **QTRSensorsRC**

(video 3: https://youtu.be/8IBdoOS-5ak)

VITEZA_MED = 75
KP = 0,030
KD = 0,100



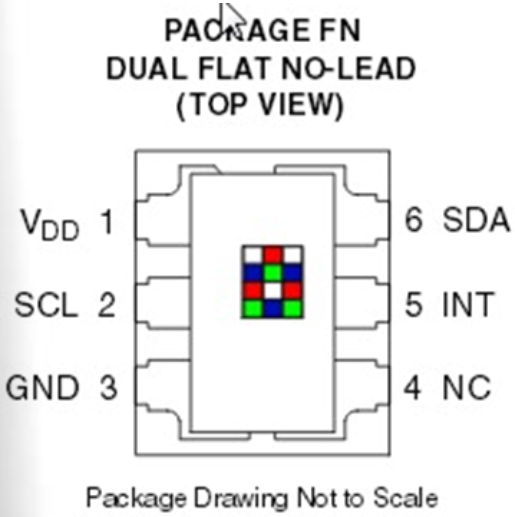
QTR-8RC IR Reflectance Sensor – **QTRSensorsRC**
(video 4: https://youtu.be/RRE8yvW3s9s)

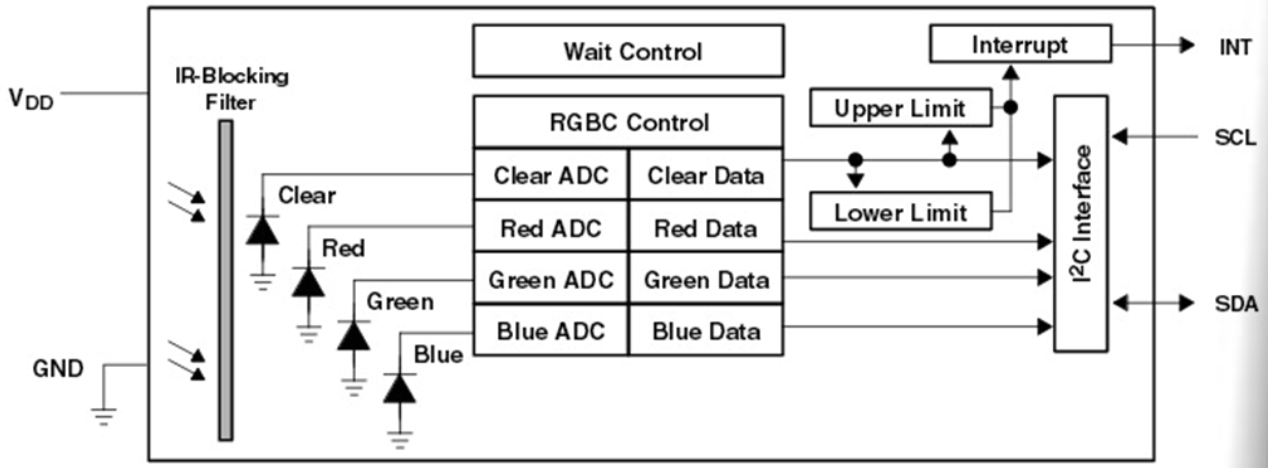
VITEZA_MED = 75
KP = 0,030
KD = 0,030

Senzori RGB

TCS34725 Senzor RGB cu Filtru IR

Principiul de functionare



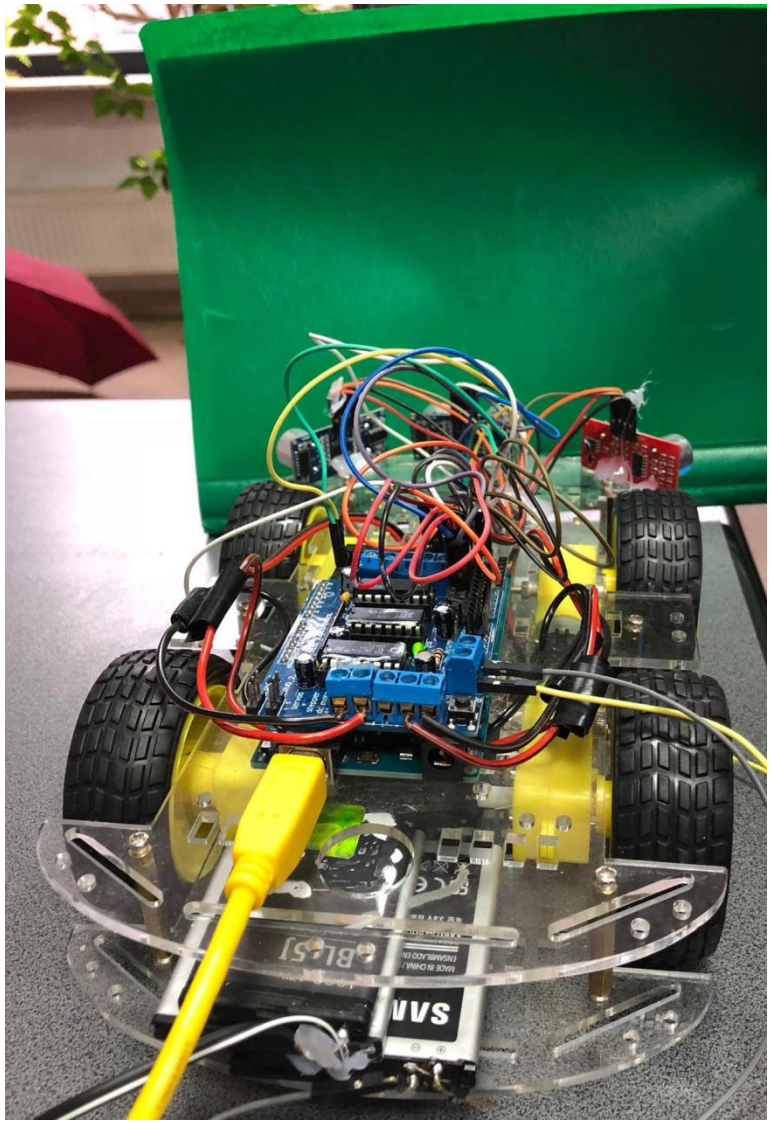


Senzori RGB

TCS34725 Senzor RGB cu Filtru IR

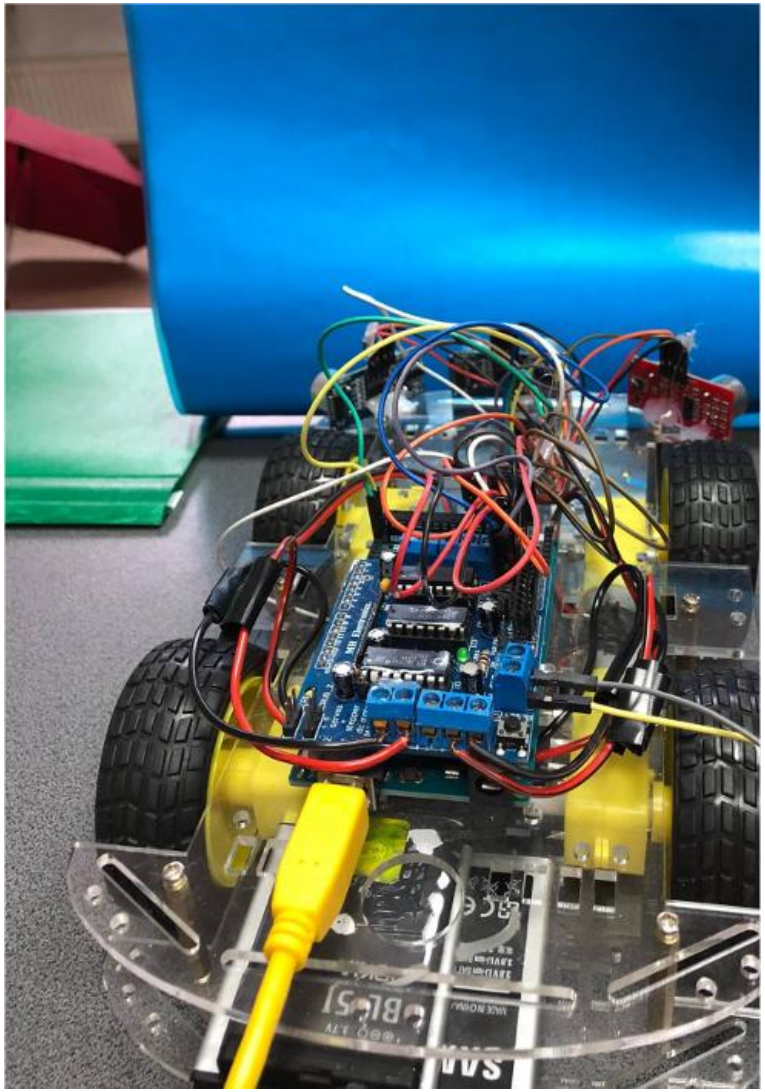
Arduino

```
OPRIT
GRESIT
Rosu:389    Verde:681    Albastru:501
Centru: 11.93    Stanga : 0.00    Dreapta :13.17
inaInte
OPRIT
GRESIT
Rosu:777    Verde:1218    Albastru:941
Centru: 6.97    Stanga : 7.31    Dreapta :9.45
OPRIT
GRESIT
Rosu:570    Verde:828    Albastru:643
Centru: 8.90    Stanga : 0.00    Dreapta :10.55
inaInte
OPRIT
```



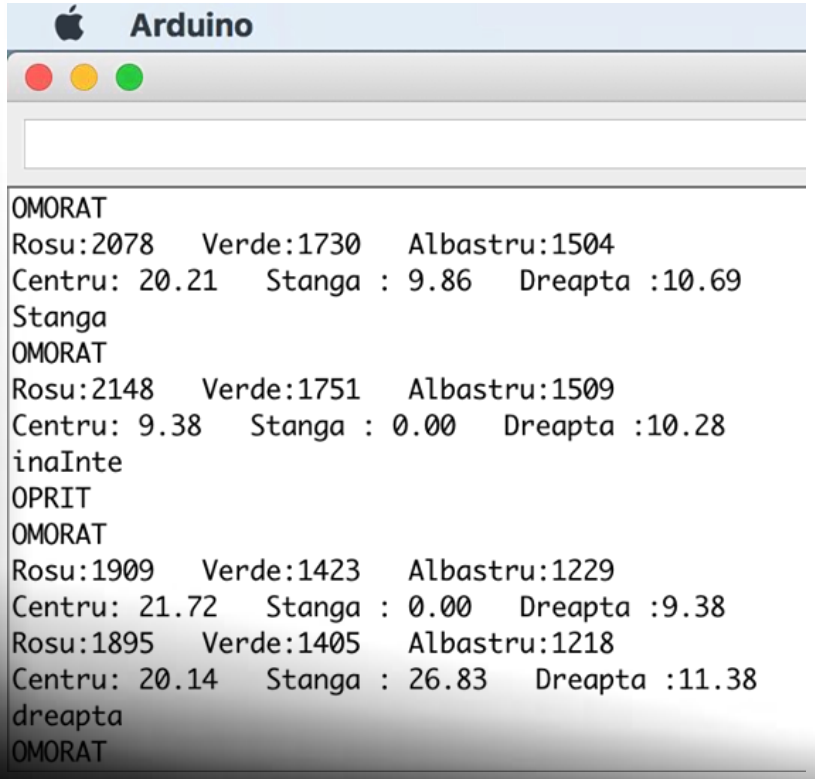
Senzori RGB

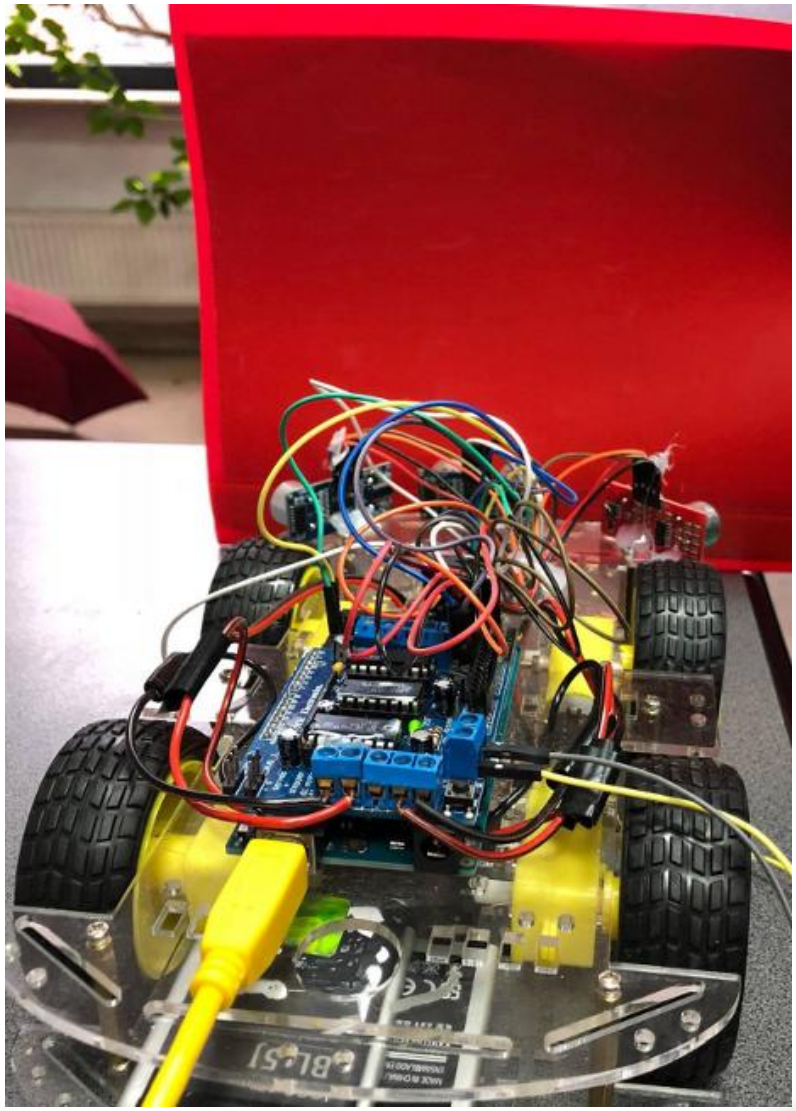
TCS34725 Senzor RGB cu Filtru IR



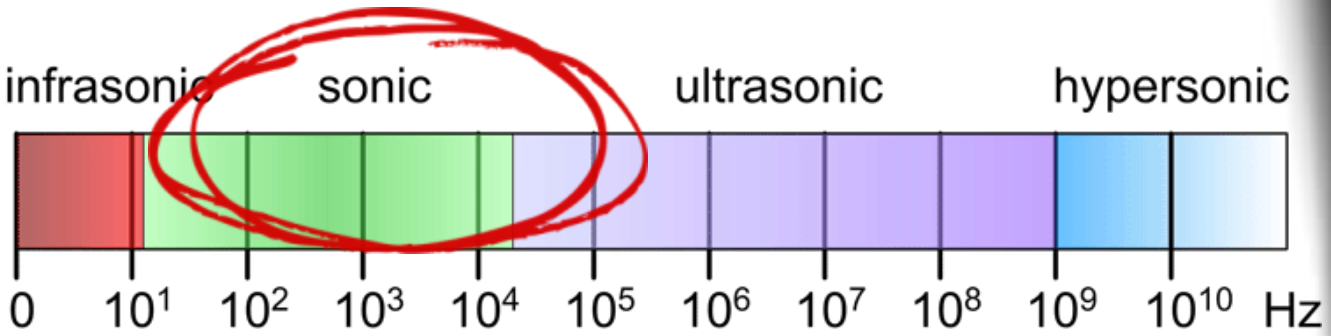
Senzori RGB

TCS34725 Senzor RGB cu Filtru IR





Senzori acustici



Banda de frecvente: 20Hz - 20kHz

7 RON

Microfon Capacitiv

36 RON

Microfon cu Breakout

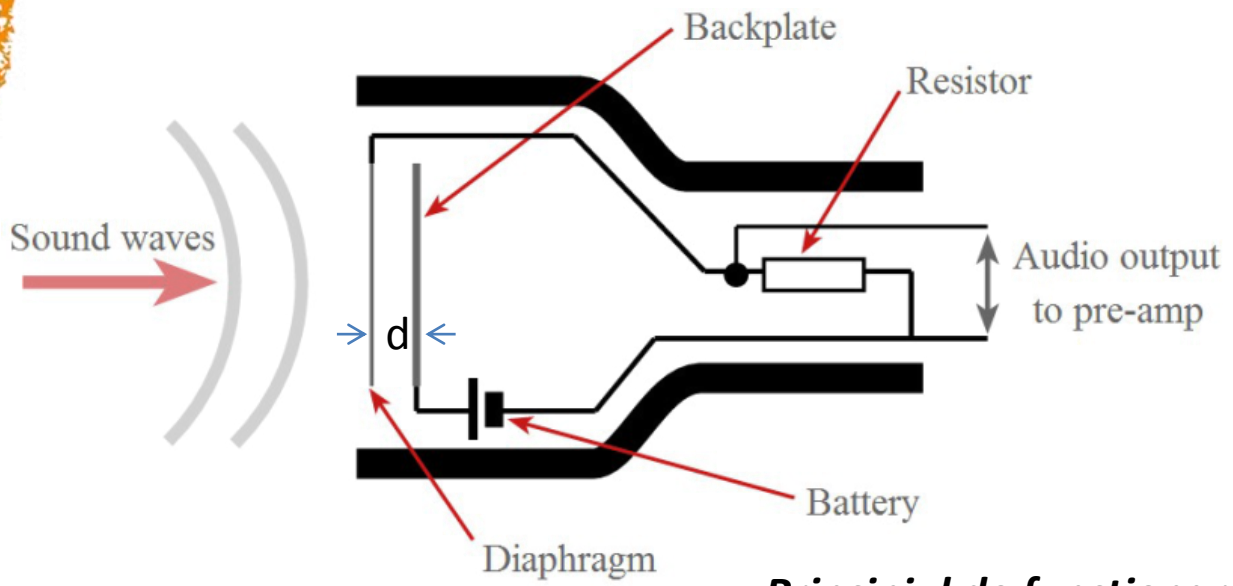
58 RON

Microfon MEMS

Aplicatii: comanda audio

Sursa preturi: https://www.robofun.ro

Senzori acustici



Principiul de functionare

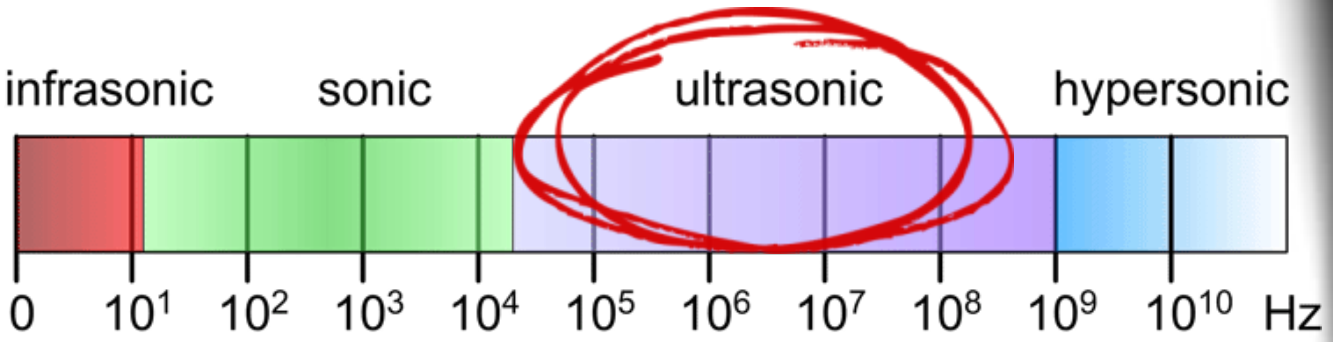
Q – sarcina pe diafragma capacitiva – aprox. const. fata de variatiile rapide ale diafragmei

⟹ scade d ⟹ creste capacitatea ⟹ scade tensiunea pe condensator
⟹ tensiunea pe rezistor devine pozitiva

⟹ creste d ⟹ scade capacitatea ⟹ creste tensiunea pe condensator
⟹ tensiunea pe rezistor devine negativa

Sursa preturi: https://www.robofun.ro

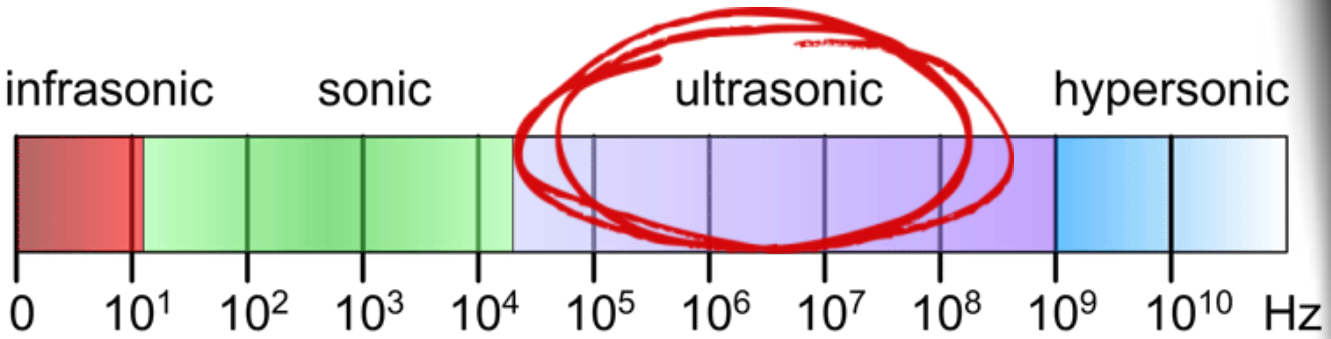
Senzori acustici



animal	frecventa de jos (Hz)	frecventa de sus (Hz)
om: https://www.youtube.com/watch?v=H-iCZElJ8m0	20	20,000
pisica	100	32,000
catel	40	46,000
cal	31	40,000
elefant	16	12,000
vaca	16	40,000
liliac	1,000	150,000
greierele si lacusta	100	50,000
rozatoare	1,000	100,000
balene si delfini	70	150,000
foca / leul de mare	200	55,000

Sursa: https://www.britannica.com/science/ultrasonics

Senzori acustici



Frecventa de functionare: 40kHz

25 RON

HC SR-04

Unghi de functionare: 15 °
Distanta: 3cm - 4m
Rezolutie: 1cm

179 RON

Maxbotix LV EZ4

210 RON

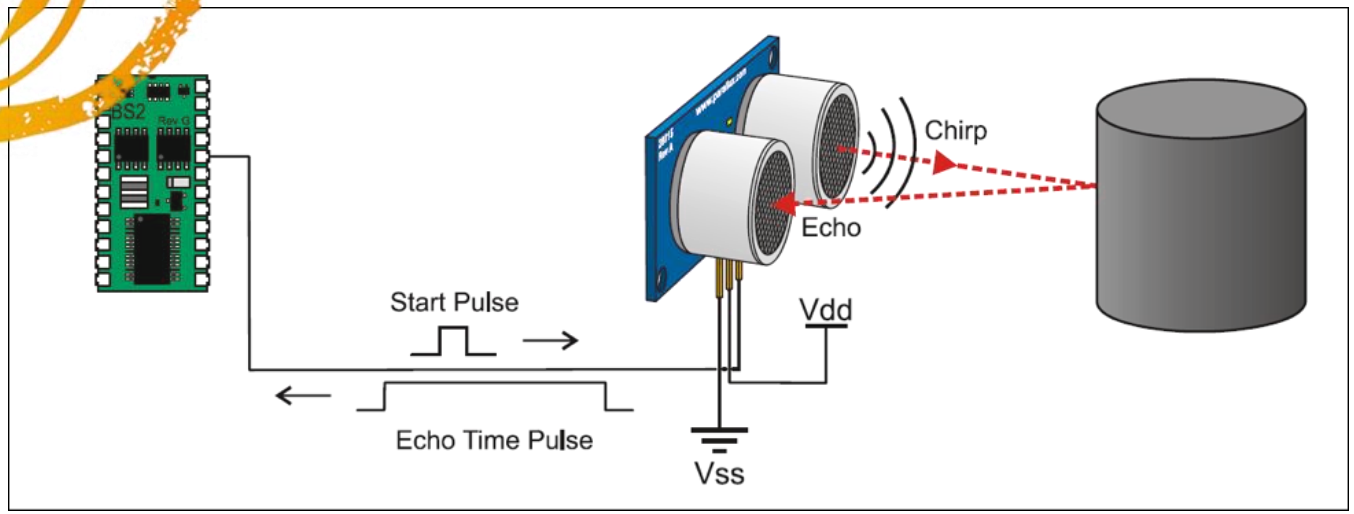
Paralax Ping

Aplicatii: senzori de distanta

Sursa preturi: https://www.robofun.ro

Senzori ultrasonici

Principiul de functionare

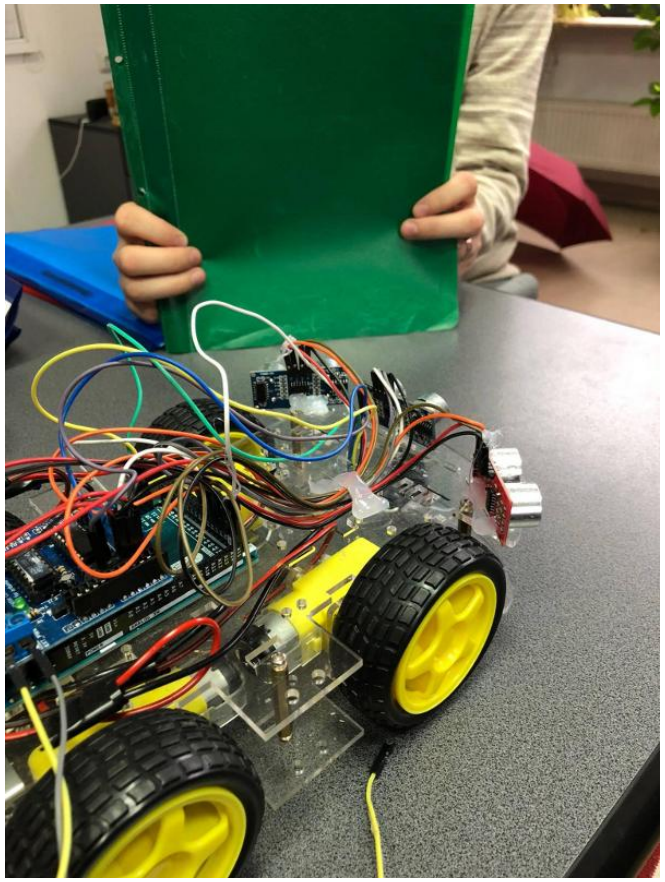


Principiul Sonarului – Putem realiza caracterizarea obiectului ţintă folosind ecoul generat de propagarea sunetului in spaţiul din jurul obiectului.

Sursa preturi: https://www.robofun.ro

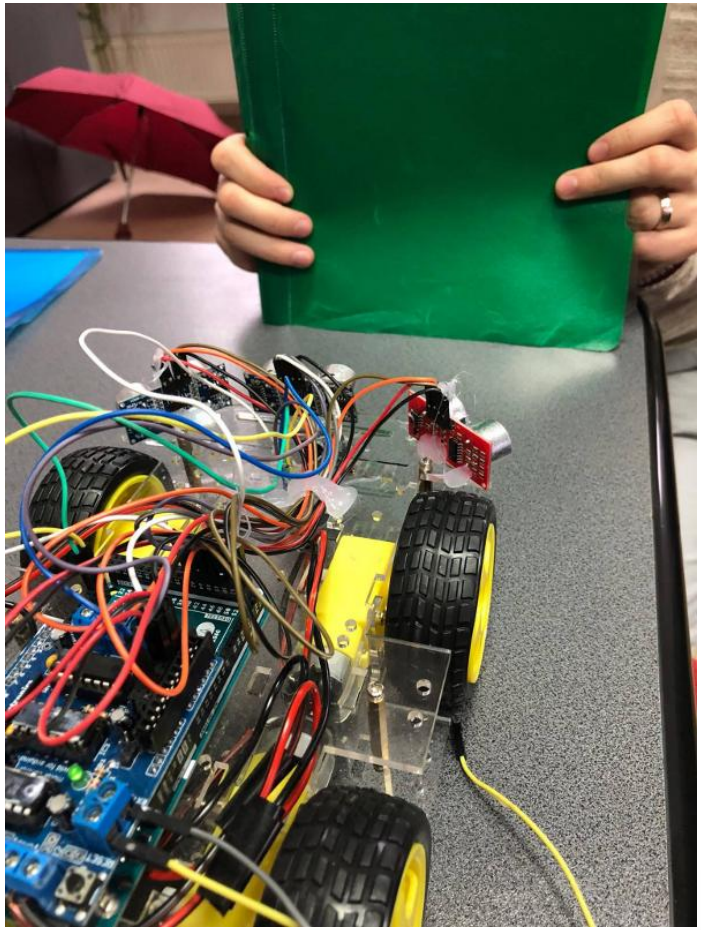
Senzori ultrasonici


```
Centru: 0.00   Stanga : 22.21   Dreapta :0.00
GRESIT
Rosu:542   Verde:540   Albastru:556
Centru: 0.00   Stanga : 21.66   Dreapta :0.00
GRESIT
Rosu:548   Verde:545   Albastru:561
Centru: 0.00   Stanga : 24.62   Dreapta :0.00
GRESIT
Rosu:551   Verde:548   Albastru:564
Centru: 0.00   Stanga : 24.14   Dreapta :0.00
GRESIT
Rosu:554   Verde:551   Albastru:567
Centru: 0.00   Stanga : 22.83   Dreapta :0.00
GRESIT
Rosu:557   Verde:554   Albastru:570
Centru: 0.00   Stanga : 22.90   Dreapta :0.00
```



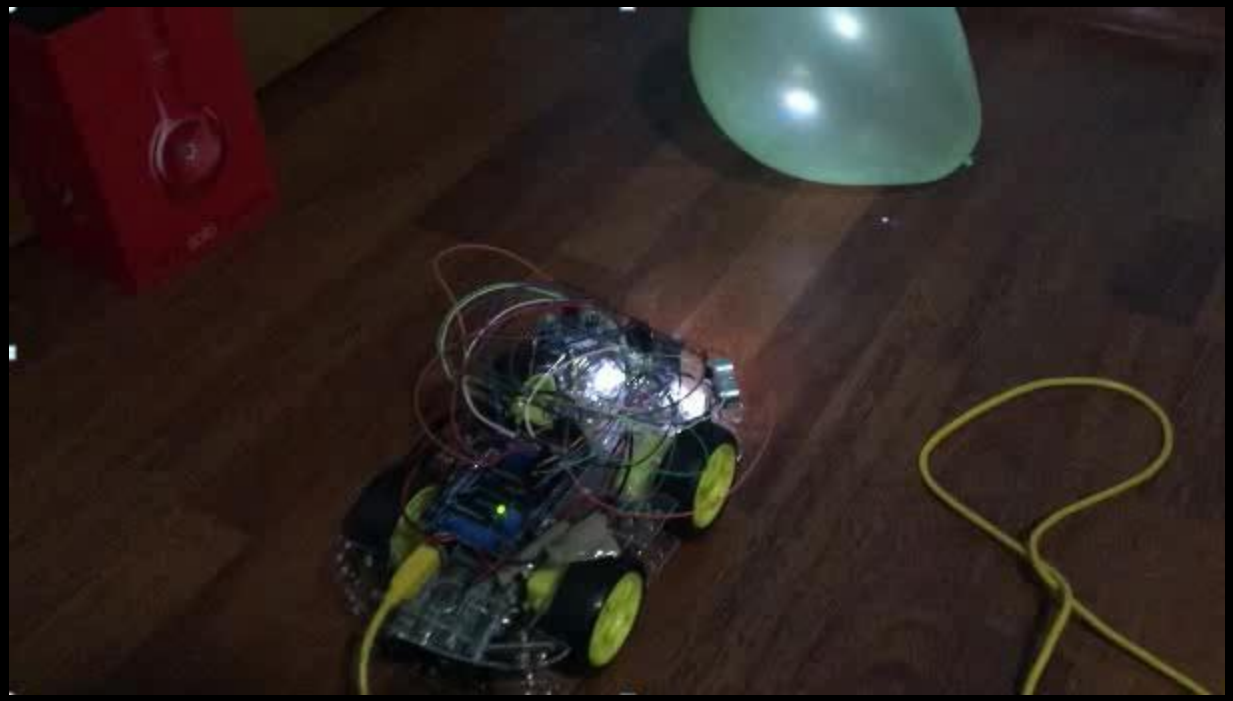
Senzori ultrasonici


```
Rosu:94   Verde:120   Albastru:112
Centru: 21.03   Stanga : 0.00   Dreapta :0.00
INAINTE
GRESIT
Rosu:91   Verde:115   Albastru:107
Centru: 20.62   Stanga : 0.00   Dreapta :26.48
inaInte
GRESIT
Rosu:97   Verde:124   Albastru:115
Centru: 20.14   Stanga : 0.00   Dreapta :0.00
INAINTE
GRESIT
Rosu:97   Verde:124   Albastru:115
Centru: 20.55   Stanga : 0.00   Dreapta :0.00
INAINTE
```



Robotul cu HC SR-04 si TCS34725

(video: https://youtu.be/ZF7SfD8D7Z0)



Robotul cu HC SR-04 si TCS34725

LINII DE COD

Citire distanta

```
digitalWrite(TRIG_PIN2,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN2,LOW);
while(digitalRead(ECHO_PIN2) == 0);
t3=micros();
while(digitalRead(ECHO_PIN2) == 1);
t4=micros();
pulse_width2 = t4 - t3;
if(pulse_width2<=MAX_DIST){


cm1=pulse_width2/58.0;}
else {
  cm1=0;
}
```

Oprire centru

```
void oprire_centru(int distanta_centru){
  if(distanta_centru<20 && distanta_centru>0 ){
    oprire();
    delay(3000);
    detectie_culoare();

  }
}
```

Deplasare inainte

```
void inainte()
{
 dreapta.setSpeed(80);
 stanga.setSpeed(80);
 stanga.run(FORWARD);
 dreapta.run(FORWARD);

}
```

Detectie culoare

```
void culoare(int r,int b,int g){
  if(r>480 && r>b && r>g){
  inainte();

}
else if(r>350 && r<460 ){
  inapoi();
  delay(300);
  viraj_stanga();
  delay(600);

}
}
```

Comparare distanta

```
if(distanta_centru<distanta_dreapta && distanta_centru<distanta_stanga && distanta_centru>0 && distanta_stanga>0 && distanta_dreapta>0)
{
  inainte();
  oprire_centru(distanta_centru);
  }
```

Multumim pentru atentie!

Bafta la concurs!