

ROBOFUN.RO
LECȚIA XVI

CURS GRATUIT

ARDUINO ȘI ROBOTICĂ

**Comanda motoarelor de
curent continuu folosind
Arduino și driver-ul de
motoare shield L298**

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs
CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

Comanda motoarelor de curent continuu folosind Arduino si driver-ul de motoare shield L298

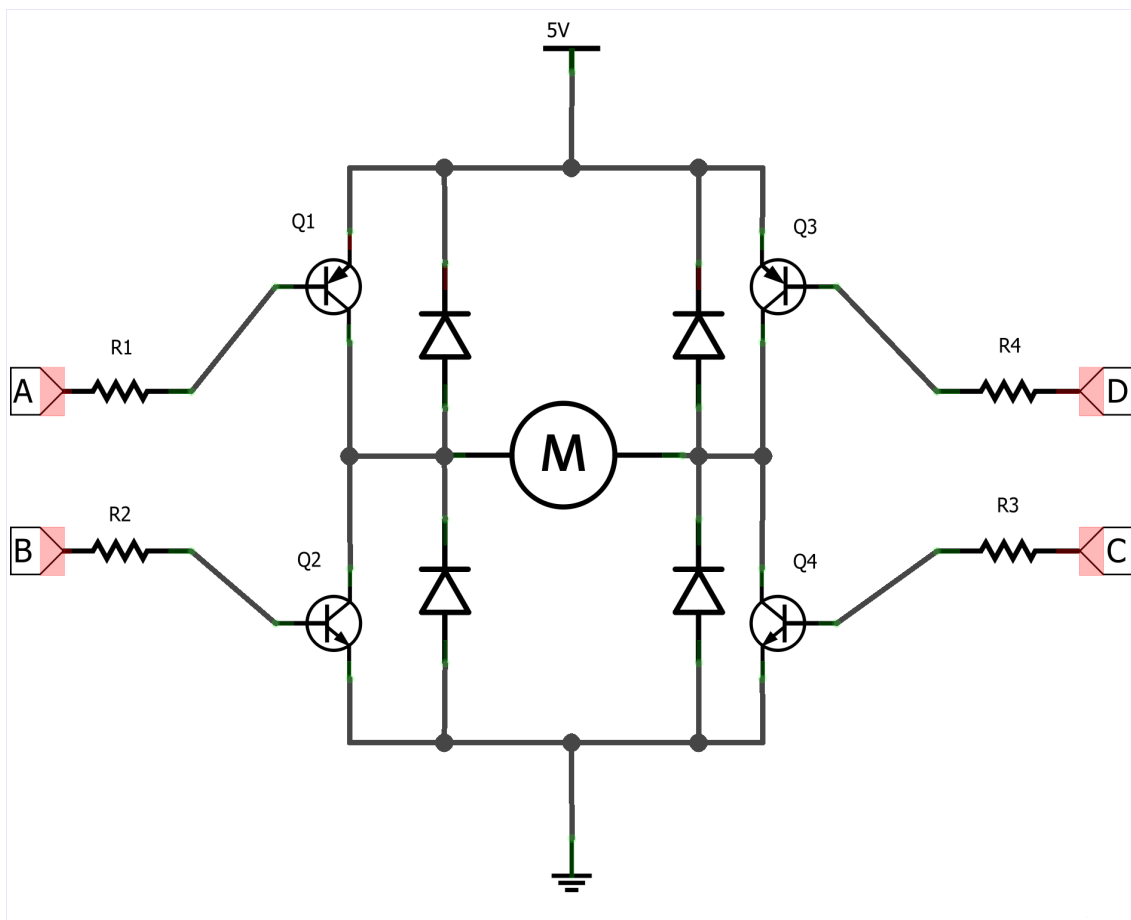
Driver-ul de motoare L298 se utilizeaza pentru controlul motoarelor de curent continuu. Arduino este capabil sa scoata pe porturile lui o putere foarte mica, total insuficienta pentru a invarti un motor. Daca vom conecta un motor electric direct la un port Arduino, cel mai probabil vom obtine arderea procesorului din placa Arduino.

Ca sa nu se intample acest lucru, avem nevoie de un amplificator de putere, care sa ia putere din sursa de alimentare (baterie, de exemplu), si sa o transmita motoarelor asa cum ii spune Arduino. Acest amplificator poarta numele generic de "driver de motoare". Exista o multitudine de drivere de motoare, diferenta majora intre ele fiind cat de multa putere pot conduce (cat de puternice pot fi motoarele pe care le pot controla).

Driver-ul din aceasta sectiune este bazat pe integratul L298, fiind un driver de nivel mediu din punct de vedere al puterii conduse. Poate controla motoare care necesita cel mult 2 Amperi (daca nu-ti e clar ce sunt aia "Amperi", vezi tutorialul "Analogie electricitate - curgerea fluidelor").

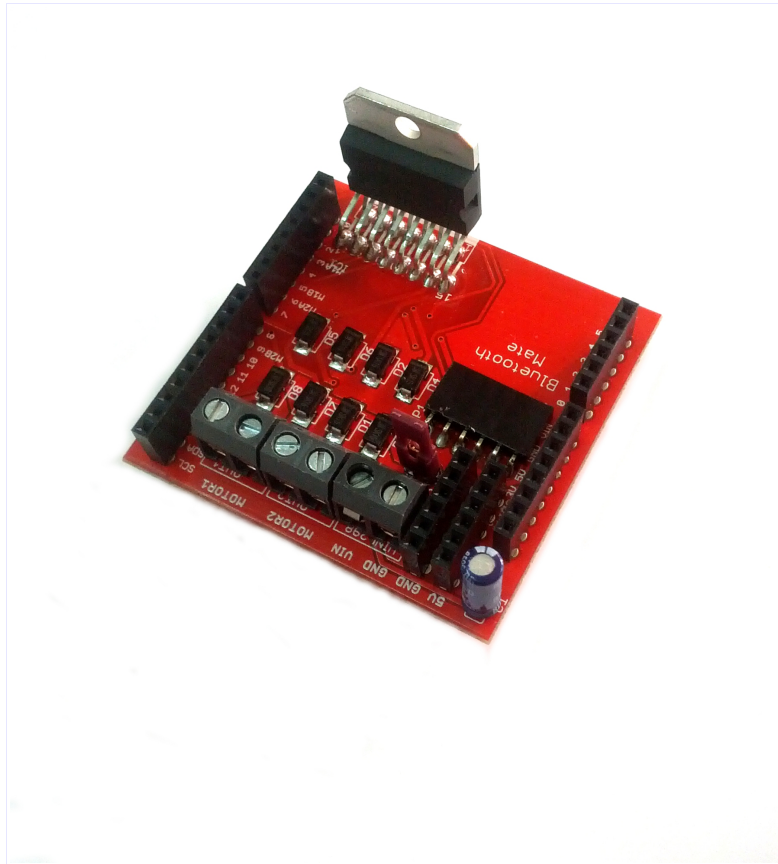
Ca sa intelegi mai bine lucrurile, in cele ce urmeaza vom prezenta simplificat modul de functionare interna al integratului L298. Nu este necesar sa intelegi neaparat cum functioneaza in interior ca sa il poti utiliza, dar ajuta. Integratul L298 este format din foarte multe tranzistoare conectate in cascada. Pentru simplificare, vom considera doar patru tranzistoare, conectate ca mai jos.

In schema de mai jos este o punte H alcatuita din tranzistoare bipolare. Motorul este simbolizat cu M si diodele au rol de protectie. Daca aplici tensiune pe punctele A,B,C,D conform tabelului de adevar, vei obtine cele 4 situatii. Daca nu aplici nici o tensiune pe bazele tranzistoarelor atunci motorul ramane in starea in care era inainte, adica nu se va roti. Punctele A,B,C,D se conecteaza la porturile platformei Arduino.



A	B	C	D	Rezultat obtinut
1	0	1	0	Motorul se roteste in sensul acelor de ceasornic
0	1	0	1	Motorul se roteste in sensul invers acelor de ceasornic
1	0	0	1	Motorul franeaza
0	1	1	0	Motorul franeaza

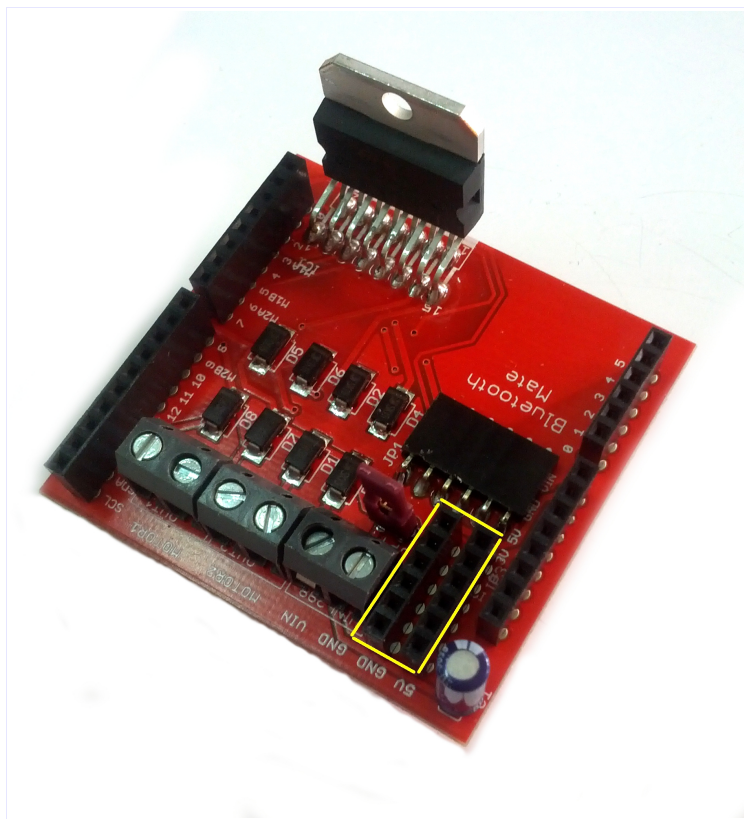
Noua varianta de shield L298



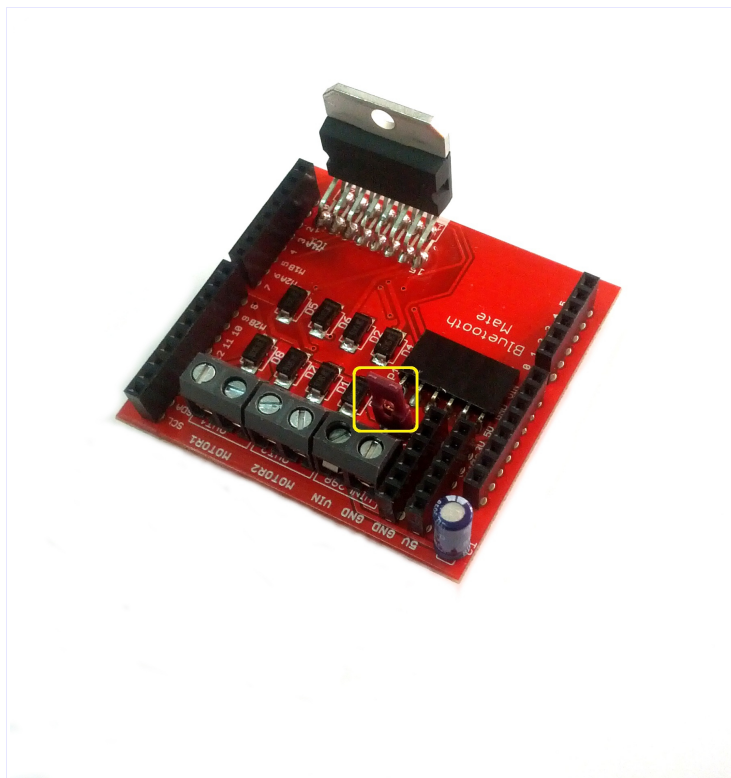
Descrierea shield-ului

Pe shield sunt prezenti o serie de conectori:

1. 2 conectori cu surub marcati cu MOTOR1 si MOTOR2 care permit conectarea celor 2 motoare de curent continuu.
2. 1 conector cu surub marcat cu VIN si GND care permite conectarea sursei de alimentare externa pentru motoare. Totodata permite alimentarea placii Arduino (detaliat la punctul 5).
3. Pinii de comanda ai driverului sunt conectati prin shield la pinii 3, 5, 6 si 9 de pe platforma Arduino.
4. 2 barete de 6 pini, fiecare, marcate cu GND si 5V care permit alimentarea cu usurinta a senzorilor care accepta 5V. Baretele sunt evidentiata in imaginea de mai jos.

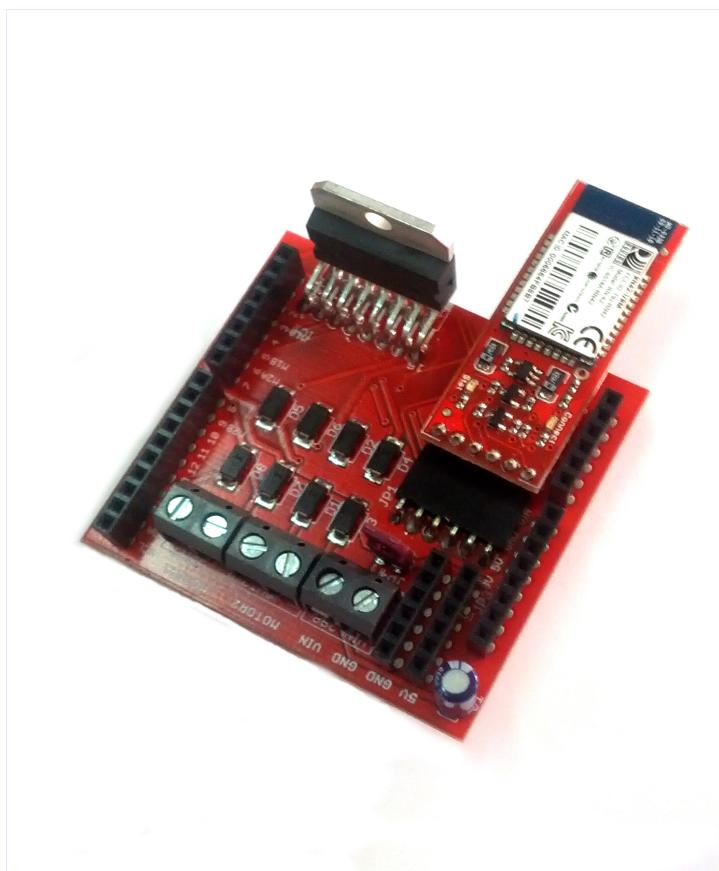


5. 1 jumper marcat cu culoarea MOV care conecteaza pinul VIN cu surub la pinul VIN al placii Arduino. Jumper-ul este evidentiat in urmatoarea imagine.



Atentie: Daca jumper-ul este conectat atunci placa Arduino se alimentata direct din driver si se utilizeaza numai o singura sursa de alimentare (conectata la conectorul de la punctul 2). Daca jumper-ul este deconectat atunci placa Arduino nu se mai alimenteaza direct din driver. Daca vrei sa alimentezi separat shield-ul si driver-ul, atunci jumperul va sta intotdeauna scos. Concluzie: jumper-ul se comporta ca un comutator pentru placa Arduino.

6. 1 conector Bluetooth Mate. Aici poti conecta modemul Bluetooth Mate atunci cand vrei sa construiesti un robot [Magician](#) sau un alt gen de robot care se controleaza de la distanta prin Bluetooth. Conectorul este simplu si este de tip plug and play. Conectezi modemul si ai deja la dispozitie o conexiune seriala prin Bluetooth.

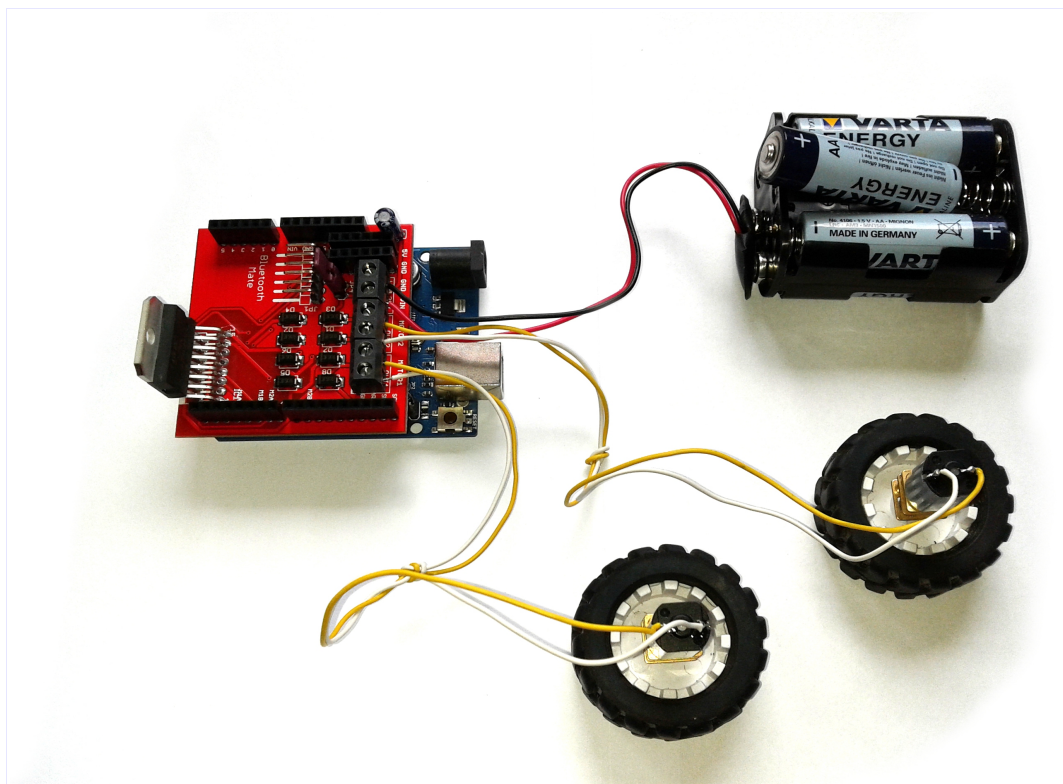


Cum alimentam shield-ul si placa Arduino ?

Alimentarea shield-ului se face prin pinii cu surub VIN si GND. Shield-ul se alimenteaza in functie de necesitatile motoarelor. Daca motoarele accepta o tensiune de cel mult 9V atunci poti utiliza cu usurinta o cutie cu 6 baterii. Tensiunea totala a bateriilor este de 9V dar driver-ul consuma intern intre 2.5 si 4.5V. In acest mod, pe motor, ajunge o tensiune cuprinsa intre 4.5V si 6.5V, suficient cat sa isi indeplineasca functia si sa nu il arzi.

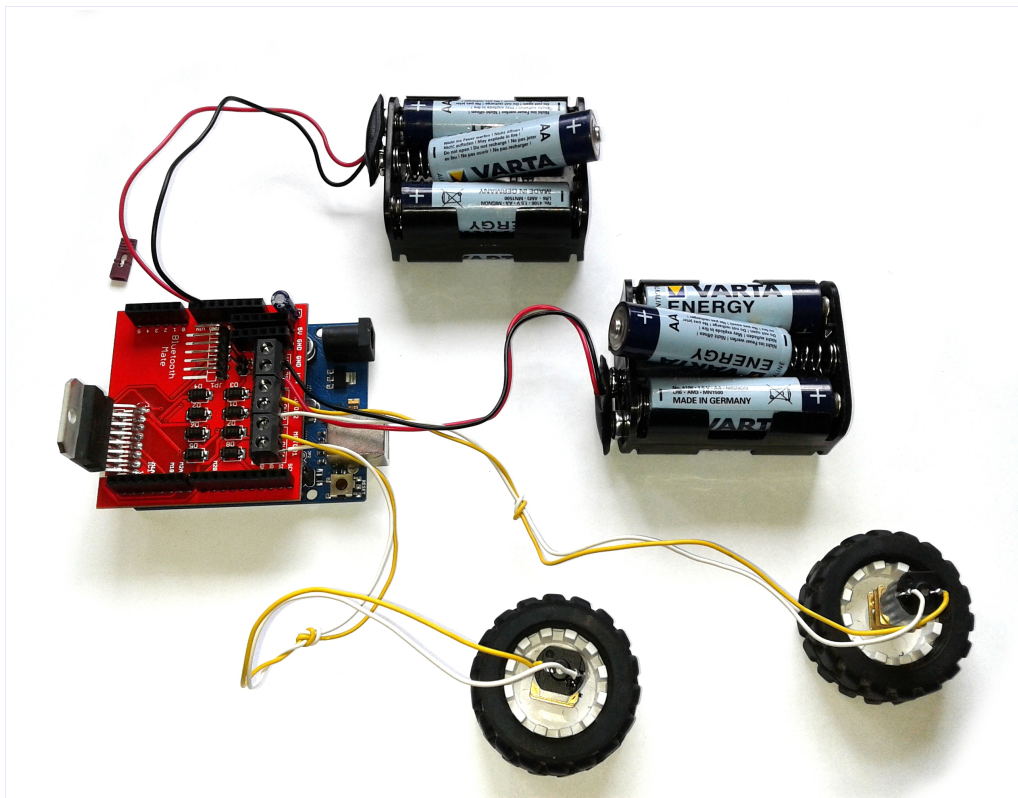
Alimentarea placii Arduino se poate realiza, in primul caz, direct din shield. Poti face lucrul asta utilizand o singura cutie de baterii sau o singura sursa de alimentare (acumulator LiPO). Cutia de baterii sau sursa de alimentare se conecteaza la conectorul cu surub marcat cu VIN si GND iar jumper-ul de culoare MOV va sta si el conectat. Daca scoti jumper-ul placa Arduino se opreste dar shield-ul ramane in continuare alimentat, deci robotul nu este oprit complet.

Un alt aspect de luat in considerare este tensiunea de alimentare a placii Arduino. Placa se poate alimenta cu o tensiune cuprinsa intre 6.5 si 12V. Ideal este sa nu ajungi la 12V pentru ca nu este cazul, 9V fiind suficient si pentru motoare dar si pentru Arduino.



A doua varianta de alimentare a placii Arduino este dintr-o alta cutie de baterii. In cazul asta vei avea o cutie de baterii pentru shield si o alta cutie de baterii pentru Arduino. Cutia de baterii a shield-ului o conectezi, clasic, la conectorul cu surub iar cealalta cutie de baterii o conectezi fie la mufa jack a placii Arduino, fie la conectorii fara surub marcati cu GND si VIN.

Atentie: In cazul asta jumper-ul MOV va sta neaparat deconectat.



Avantaje si dezavantaje

Primul caz, cel cu alimentare dintr-o singura cutie de baterii, are avantajul ca ocupa un spatiu mult mai mic, deasemenea si costul robotului este mai redus. Dezavantajul este ca atunci cand bateriile se descarca prea mult, placa Arduino se poate reseta sau senzorii pot genera semnale eronate din cauza ca motoarele necesita un curent prea mare cand pornesc sau cand isi schimba sensul de rotatie.

In al doilea caz, avantajul este ca placa Arduino nu se va mai reseta, ea fiind alimentata separat. Dezavantajul este spatiul ocupat care este mult mai mare si eventual costul cutiei suplimentare cu baterii.

Cum estimam cat timp functioneaza robotul ?

Timpul de functionare al robotului este dat de capacitatea bateriei. In exemplul din acest tutorial s-au folosit 6 baterii alcaline de 1.5V. In total bateriile genereaza 9V si in functie de baterie, capacitatea lor este cuprinsa intre 1200 si 2000mAh (miliamperi ori ora).

Ca sa ne dam seama cat timp functioneaza robotul trebuie sa stim cat consuma acesta. Consumul il putem determina cu un ampermetru conectat in serie intre robot si cutia de baterii. Realizeaza o medie a consumului cand robotul se afla in diverse situatii, de exemplu: cand motoarele se rotesc cu viteza maxima sau cand motoarele se rotesc la jumatate din viteza sau cand motoarele merg in

gol. Sa presupunem ca o medie a consumului este de 200mA. In acest caz impartim capacitatea bateriei la consumul de 200mA.

Rezulta: $2000\text{mAh} / 200\text{mA} = 10 \text{ h}$, reprezinta durata de functionare (exprimata in ore) a robotului cand consumul lui mediu este de 200mA.

Capacitatea unei baterii dreptunghiulare de 9V este de 200mAh. Daca alimentam robotul cu o baterie de 9V atunci putem estima timpul de functionare.

Rezulta: $200\text{mAh} / 200\text{mA} = 1 \text{ h}$, o singura ora de functionare.

Din cele doua situatii, prima este eficienta pentru ca iti ofera de 10 ori mai mult o autonomie de functionare.

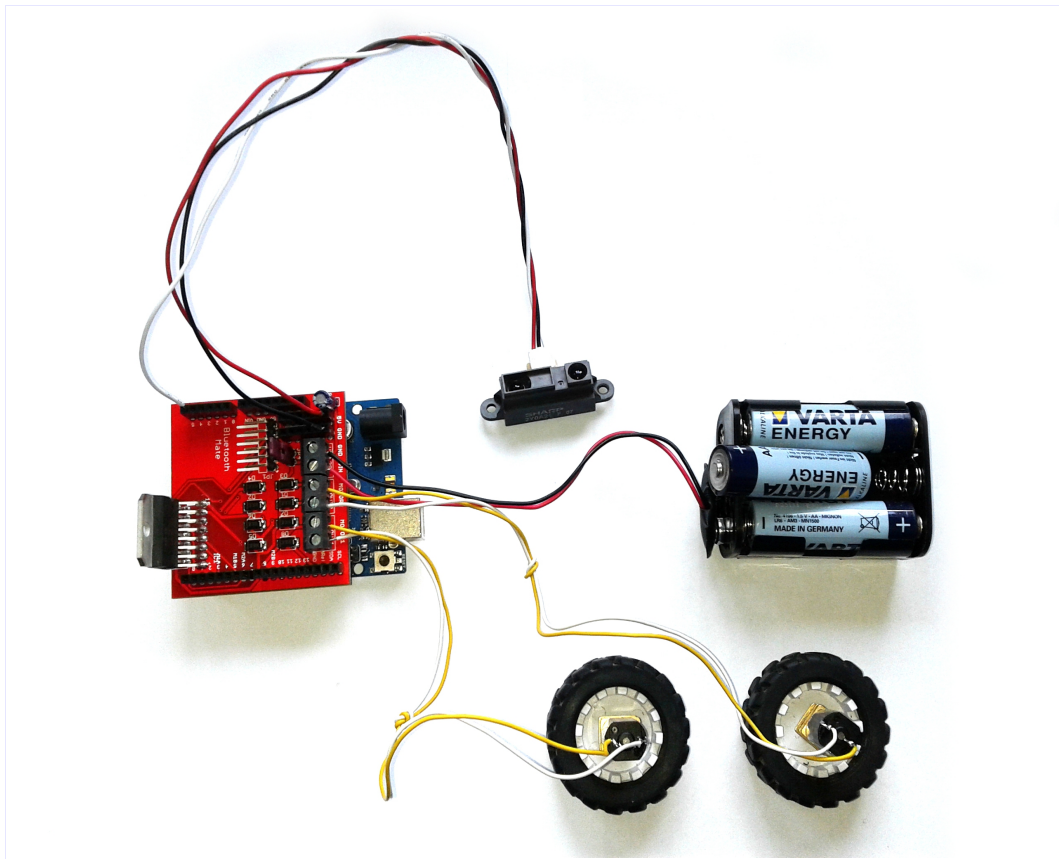
Cum alimentez senzorii direct din shield ?

Shield-ul este echipat cu 2 barete de 6 pini, fiecare, marcate cu 5V si GND. Baretele au rolul de a alimenta o gama larga de senzori care accepta 5V.

Conectarea senzorului la shield se realizeaza astfel:

Firul rosu al senzorului	Pin bareta 5V
Firul negru al senzorului	Pin bareta GND
Firul alb al senzorului	Pin analog sau digital

In imaginea urmatoare s-a alimentat un senzor Sharp din cele doua barete de 6 pini.



Un exemplu de cod sursa.

Daca vrei sa testezi functia de baza a driverului atunci incarca in Arduino codul sursa urmator.

```
int MOTOR2_PIN1 = 3;
int MOTOR2_PIN2 = 5;
int MOTOR1_PIN1 = 6;
int MOTOR1_PIN2 = 9;

void setup() {
    pinMode(MOTOR1_PIN1, OUTPUT);
    pinMode(MOTOR1_PIN2, OUTPUT);
    pinMode(MOTOR2_PIN1, OUTPUT);
    pinMode(MOTOR2_PIN2, OUTPUT);
    Serial.begin(9600);
}

void loop() {
    go(255,-255);
    delay(1000);
}
```

```

    go(-255,-255);
    delay(1000);
    go(-255,255);
    delay(1000);
    go(255,255);
    delay(1000);

}

void go(int speedLeft, int speedRight) {
    if (speedLeft > 0) {
        analogWrite(MOTOR1_PIN1, speedLeft);
        analogWrite(MOTOR1_PIN2, 0);
    }
    else {
        analogWrite(MOTOR1_PIN1, 0);
        analogWrite(MOTOR1_PIN2, -speedLeft);
    }

    if (speedRight > 0) {
        analogWrite(MOTOR2_PIN1, speedRight);
        analogWrite(MOTOR2_PIN2, 0);
    }
    else {
        analogWrite(MOTOR2_PIN1, 0);
        analogWrite(MOTOR2_PIN2, -speedRight);
    }
}

```