ROBOFUN.RO

LECȚIA II

CURS GRATUIT

ARDUINO ȘI ROBOTICĂ

Introducere în Arduino

Led, Buton, Senzor Lumina, Mini Difuzor, Senzor Temperatura, Senzor Umiditate

Led Brick, intensitate variabila

In prima lectie am vazut cum putem aprinde si stinge un led. In acest exemplu vom folosi acelasi led, dar il vom aprinde variabil. Chiar daca Arduino nu poate scoate tensiune variabila pe porturile digitale (scoate ori 5V ori 0V), exista o posibilitate de a genera un semnal de putere variabila pe unul dintre porturile sale. Acest lucru este posibil prin generarea unui semnal dreptunghiular, care se plimba periodic intre 0V si 5V, foarte rapid. In functie de cat timp sta in 5V si cat timp sta in 0V, puterea semnalului variaza. Numele acestui gen de semnal este "PWM". Vom detalia intr-o lectie viitoare acest tip de semnal, deocamdata este suficient sa stim ca exista si ca ii putem controla puterea prin variatia raportului intre timpul cat sta in 1 si cat sta in 0.

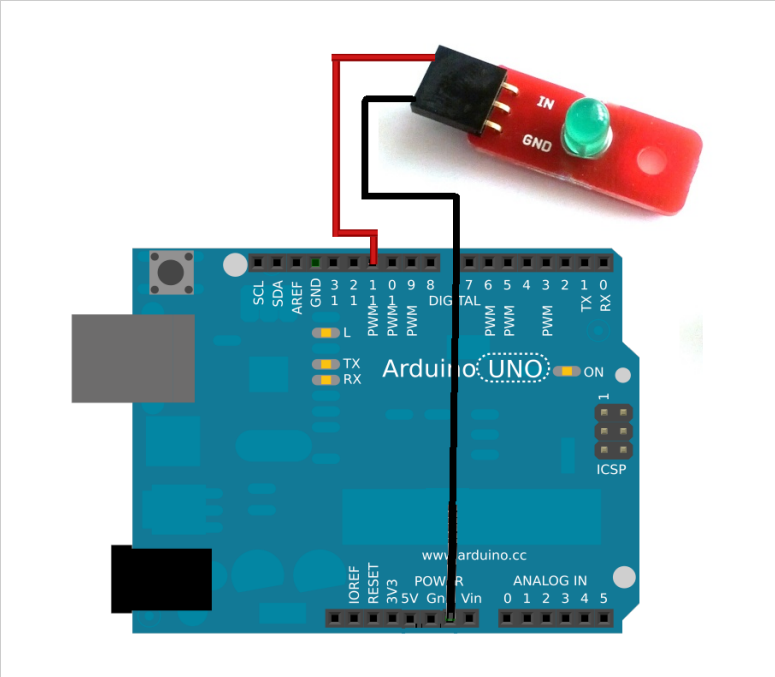
Este interesant de remarcat faptul ca doar 6 din cei 13 pini ai Arduino UNO sunt capabili sa genereze semnal PWM (pinii 3, 5, 6, 9, 10 si 11).

Pentru cazul Arduino Mega, avem 15 pini PWM (de la 2 la 13 si de la 44 la 46).

```
void setup() {
      pinMode(11, OUTPUT);
}
void loop() {
      for (int i = 0; i < 255; i++){
            analogWrite(11, i);
            delay(50);
      }
      for (int i = 255; i > 0; i--){
            analogWrite(11, i);
            delay(50);
      }
}
```

Rutina *setup,* care se executa exact o singura data cand Arduino este alimentat, declara pinul digital 11 (cel la care am conectat led-ul) ca fiind un pin *de iesire.*

In rutina *loop* este interesanta instructiunea *analogWrite*, care defineste puterea semnalului PWM de iesire. Ca parametri, instructiunea *analogWrite* primeste pinul (11, in cazul nostru), si puterea semnalului (variabila, de la 0 la 255). Aceasta instructiune este apelata intr-un ciclu *for*, care modifica valoarea variabilei *i* intre 0 si 255. Efectul va fi ca led-ul se va aprinde gradat pana la maxim, iar apoi se va stinge treptat.



Arduino GND	GND Led
Arduino Digital 11	IN Led

De ce nu merge ?

–sigur ai conectat corect firele ?

–sigur programul s-a incarcat pe Arduino (ai vazut mesajul "Done Uploading" ?)

–daca scoti firul de conectare din pinul 13 (marcat pe led cu "IN") si il muti in pinul VCC, se aprinde led-ul ? (daca nu, atunci led-ul probabil este defect)

Breadboard

Un breadboard este un dispozitiv care permite conectarea extrem de simpla a componentelor electronice, fara lipituri. Pentru a conecta dispozitivele se folosesc fire tata-tata (cu pini la ambele capete), care se introduc in gaurile din breaboard.

Gaurile existente in breaboard sunt conectate intre ele (de obicei pe linie), astfel incat firele introduse pe aceeasi linie vor fi conectate intre ele. In imaginea de mai jos am marcat cu linie neagra pinii conectati intre ei (eu am marcat doar pentru primele 3 coloane, dar toate liniile breadboard-ului sunt conectate intre ele). Un exemplu frecvent de utilizare a breadboard-ului este acela in care dorim sa conectam simultan mai multe dispozitive brick la Arduino (care are doar un singur pin de 5V, si doar 3 pini de GND). In aceasta situatie, vom conecta folosind fire tata-tata pinul de 5V Arduino la una dintre liniile breaboard-ului, la alta linie din breadboard vom conecta unul dintre pinii GND, si in acest mod vom avea disponibile inca patru pini care sunt conectati la 5V (cei care sunt pe aceeasi lini cu cel

conectat la 5V), si patru pini conectati la GND (cei care sunt pe aceeasi linie cu GND). Pentru toate proiectele care urmeaza am considerat ca folosirea unui breadboard se subintelege peste tot pe unde apar doar fire legate impreuna in schema.

Exista multe tipuri de breadboard, mai mari sau mai mici. Unul dintre cele mai mici breadboard-uri este cel de aici - http://www.robofun.ro/breadboard/breadboard_mini , care este suficient pentru situatia in care vrei sa alimentezi mai multe dispozitive folosind acelasi Arduino. Un breadboard ceva mai mare (necesar pentru atunci cand vrei sa mai adaugi si alte componente pe breadboard, in afara de componente brick) este acesta – http://www.robofun.ro/breadboard/breadboard-82x52x10 . Evident, daca si acesta este prea mic pentru ce ai nevoie, poti oricand inlantui doua sau mai multe breadboard-uri intre ele, cu fire.

Debug Serial

Asa cum spuneam mai devreme, o data ce ai urcat programul pe Arduino, acesta ruleaza pe procesorul Arduino, si nu pe PC. La fel de bine poti deconecta complet Arduino de la calculator si sa il alimentezi cu o baterie, programul va continua sa ruleze. Sunt situatii (si nu putine!) cand rezultatele rularii programului sunt cu totul altele decat iti doresti tu, si atunci iti doresti sa ai acces in interiorul Arduino ca sa poti vedea ce se intampla acolo. Din fericire, exista si o alta solutie, ceva mai simpla. Cablul USB de conectare la calculator, pe langa alimentarea Arduino, poate transmite si date catre PC sau de la PC catre Arduino. Acest lucru este extrem de util pentru a vizualiza pe PC valorile din programul care ruleaza pe Arduino.

De exemplu, sa spunem ca avem un senzor de lumina conectat la Arduino si vrem sa aprindem un led atunci cand nivelul de iluminare scade sub o anumita valoare. Am scris programul, l-am urcat pe Arduino, dar cand testam, lucrurile nu functioneaza corect. Ne-am dori sa vedem ce valoare citeste senzorul de lumina, ca sa vedem daca pragul setat de noi in program este corect. Vom face acest lucru trimitand prin cablul USB valoarea citita de senzorul de lumina si vizualizand aceasta valoare pe PC.

```
void setup() {
      Serial.begin(9600);
}
void loop() {
      int lumina = analogRead(0);
      Serial.println(lumina);
      delay(10);
}
```

Pentru a rula codul sursa de mai sus nu este necesar sa conectezi un senzor de lumina. Poti sa urci pur si simplu progamul de mai sus pe Arduino si apoi sa deschizi Serial Monitor (din meniul "Tools", alegi optiunea "Serial Monitor"). Vei vedea o serie de valori aleatoare afisate pe ecran (citiri ale portului analogic 0 al Arduino, la care nu este conectat nimic).

Instructiunile interesante din programul de mai sus sunt "Serial.begin(9600)", care initializeaza o comunicare seriala intre Arduino si PC cu viteza de 9600 de biti pe secunda si "Serial.println(lumina)", care trimite valoarea variabilei "lumina" catre PC.

Poti utiliza aceasta metoda ori de cate ori vrei sa vezi ce valori au variabilele din programul tau Arduino.

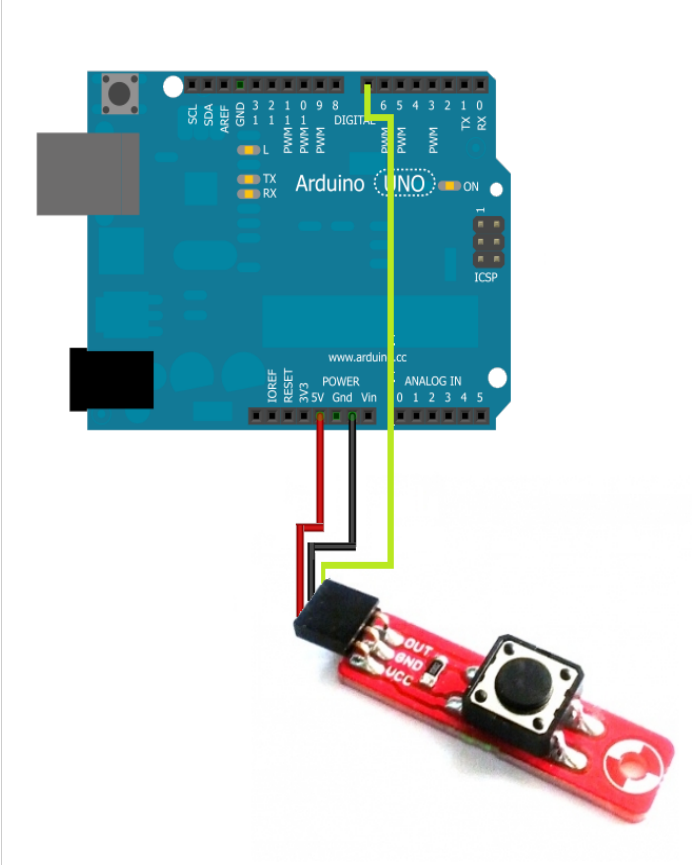
De ce nu merge ?

–daca in loc sa vezi valori numerice in Serial Monitor, primesti o eroare atunci cand alegi optiunea, atunci cel mai probabil portul serial selectat de tine nu este cel pe care este conectat Arduino; mergi in meniul "Tools" -> "Serial Port" si alege o alta optiune. Daca nu ai nici o idee care este optiunea corecta, atunci deconecteaza Arduino de PC si intra din noul in "Tools "-> "Serial Port". Portul care a disparut este acel port pe care era conectat Arduino. Reconecteaza acum Arduino, si selecteaza-l pe acesta.

– daca in loc sa vezi valori numerice in Serial Monitor vezi o serie de caractere ciudate care se schimba continuu, inseamna ca rata de transfer selectata in Serial Monitor nu este aceeasi cu rata de transfer selectata in codul sursa Arduino. Verifica in dreapta jos a ferestrei Serial Monitor ca valoarea selectata sa fie 9600 (aceeasi pe care am selectat-o in functia "setup" din codul Arduino – "Serial.begin(9600);").

Buton Brick

Am vazut in exemplele precedente cum putem folosi porturile digitale Arduino pentru a comanda dispozitive din exterior (led-uri, in exemplele de pana acum). Acum vom vedea cum putem folosi un port digital Arduino pentru a citi informatie din mediu (starea unui buton).



Arduino GND	GND Buton
Arduino 5V	VCC Buton
Arduino Digital 7	OUT Buton

```
void setup() {
      pinMode(7, INPUT);
      Serial.begin(9600);
}
void loop() {
      int stareButon = digitalRead(7);
      Serial.println(stareButon);
      delay(10);
}
```

Primul lucru interesant este faptul ca acum pinul digital 7 (cel la care am conectat un buton) este setat in mod INPUT (spre deosebire de exemplele precedente, unde era de tip OUTPUT). Asta pentru ca urmeaza sa il folosim ca sa citim informatie din mediu.

Rutina loop citeste starea butonului (care poate fi 0 sau 1 – apasat sau destins) si afiseaza aceasta stare in consola seriala. Poti vedea aceasta informatie deschizand Serial Monitor in Arduino IDE.

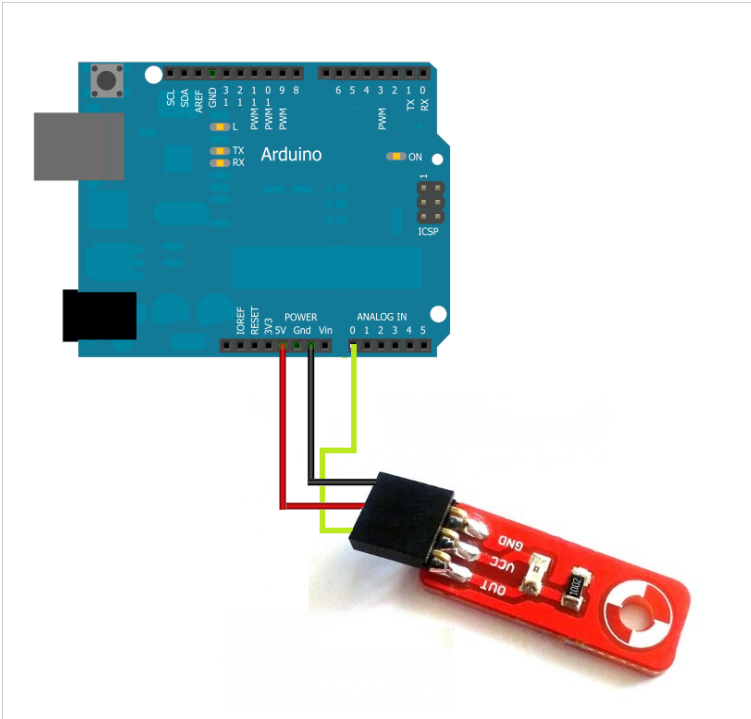
De ce nu merge ?

–sigur ai conectat corect firele ?

–sigur programul s-a incarcat pe Arduino (ai vazut mesajul "Done Uploading") ?

Senzor Lumina Brick

Dupa ce am comandat led-uri si am citit valori digitale din mediu, senzorul de lumina este primul exemplu de citire a valorilor analogice din mediu. Un senzor de lumina da o valoare numerica intre 0 si 1023, valoare proportionala cu nivelul de iluminare din mediul ambiant.



Arduino GND	GND Senzor
Arduino 5V	VCC Senzor
Arduino Analog 0	OUT Senzor

```
void setup() {
      Serial.begin(9600);
}
void loop() {
      int nivelIluminare = analogRead(0);
      Serial.println(nivelIluminare);
      delay(10);
}
```

In rutina *setup* singurul lucru pe care il facem este sa initializam o comunicare seriala cu PC-ul, pe care o vom folosi ca sa transmitem si sa vizualizam pe PC valorile citite de senzorul de lumina.

Rutina *loop* citeste valoarea data de senzorul de lumina (conectat la portul serial 0) si afiseaza aceasta valoare in consola seriala. Poti vedea aceasta informatie deschizand Serial Monitor in Arduino IDE.

Pentru a testa ca lucrurile functioneaza corect, pune degetul peste senzorul de lumina. Vei observa ca valoarea pe care o vezi in Serial Monitor scade.

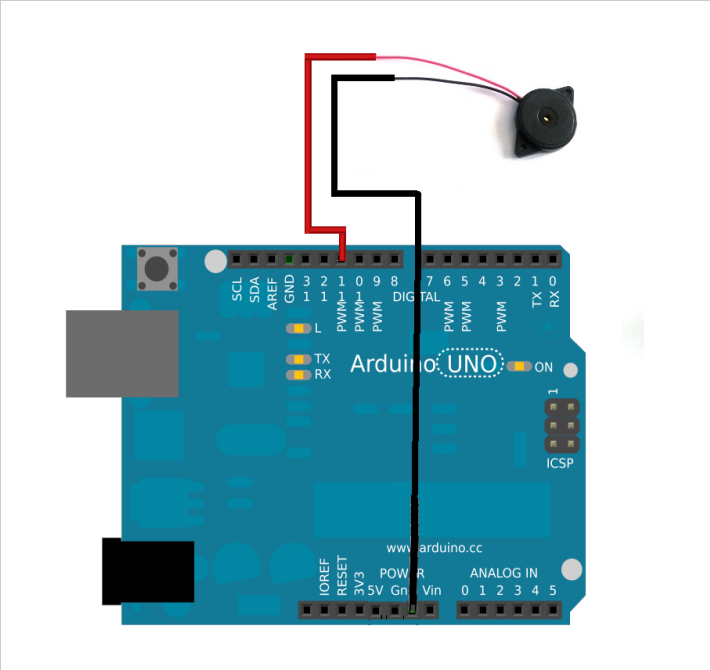
De ce nu merge ?

–sigur ai conectat corect firele ?

–sigur programul s-a incarcat pe Arduino (ai vazut mesajul "Done Uploading") ?

Mini Difuzor

Mini difuzorul este cea mai simpla modalitate de a crea sunete cu Arduino. Utilizarea acestuia este extrem de simpla. Conectezi firul negru la pinul GND al placii Arduino, iar firul rosu la un pin PWM digital al placii Arduino. Placa Arduino UNO are 6 astfel de pini (3, 5, 6, 9, 10 si 11), iar placa Arduino Mega are 15 astfel de pini (de la pinul 2 pana la pinul 13 si de la pinul 44 la pinul 46). Un semnal de tip PWM este un semnal care se misca foarte rapid intre 5V si 0V, astfel incat membrana difuzorului este si ea miscata la fel de rapid, generand sunete. Frecventa cu care semnalul se misca intre 5V si 0V determina frecventa sunetului.



Arduino GND	Fir Negru Difuzor
Arduino Digital 11	Fir Rosu Difuzor

```
void setup() {
      pinMode(11, OUTPUT);
}

void loop() {
      for (int i = 1500; i < 4000; i++) {
            tone(11, i);
            delay(10);
       }
       for (int i = 4000; i > 1500; i--) {
            tone(11, i);
            delay(10);
       }
}
```

Partea interesanta din codul de mai sus este instructiunea *tone*, care primeste ca parametri pinul la care este conectat difuzorul (in cazul nostru pinul 11) si frecventa sunetului (in cazul nostru, variabila *i*). Variabila *i* se modifica intre 1500 de Hertzi si 4000 de Hertzi. Efectul obtinut este cel de sirena. Ca sa opresti sunetul complet, instructiunea este *noTone(<pin>)*; In cazul de mai sus, *noTone(11)* opreste complet sunetul.

Senzor Temperatura Brick

Senzorul de temperatura brick este un alt exemplu de senzor care ofera valori analogice care

depind de temperatura din mediul ambiant. Din valorile citite de la senzori se obtine valoarea temperaturii in mediul ambiant in grade Celsius, aplicand o formula matematica simpla, formula prezentata in codul sursa de mai jos.

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.println(citesteTempInCelsius(10));
}

float citesteTempInCelsius(int count) {
  float temperaturaMediata = 0;
  float sumaTemperatura;
  for (int i =0; i<10; i++) {
    int reading = analogRead(0);
    float voltage = reading * 5.0;
    voltage /= 1024.0;
    float temperatureCelsius = (voltage - 0.5) * 100 ;
    sumaTemperatura = sumaTemperatura + temperatureCelsius;
  }
  return sumaTemperatura / (float)count;
}
```

La fel ca in exemplele precedente, in rutina *setup* singurul lucru pe care il facem este sa initializam o comunicare seriala cu PC-ul, pe care o vom folosi ca sa transmitem si sa vizualizam pe PC valorile citite de senzorul de temperatura.

Rutina loop nu face altceva decat sa apeleze rutina "citesteTempInCelsius" care calculeaza temperatura in grade Celsius pe baza valorii citite de la senzor. Pentru a diminua influenta surselor de erori asupra citirilor, temperatura se calculeaza pe baza a zece citiri succesive, care sunt mediate.

Ca sa testezi ca lucrurile functioneaza corect, pune degetul peste senzorul de temperatura. Vei observa ca valoarea pe care o vezi in Serial Monitor creste.

De ce nu merge ?

–sigur ai conectat corect firele ?

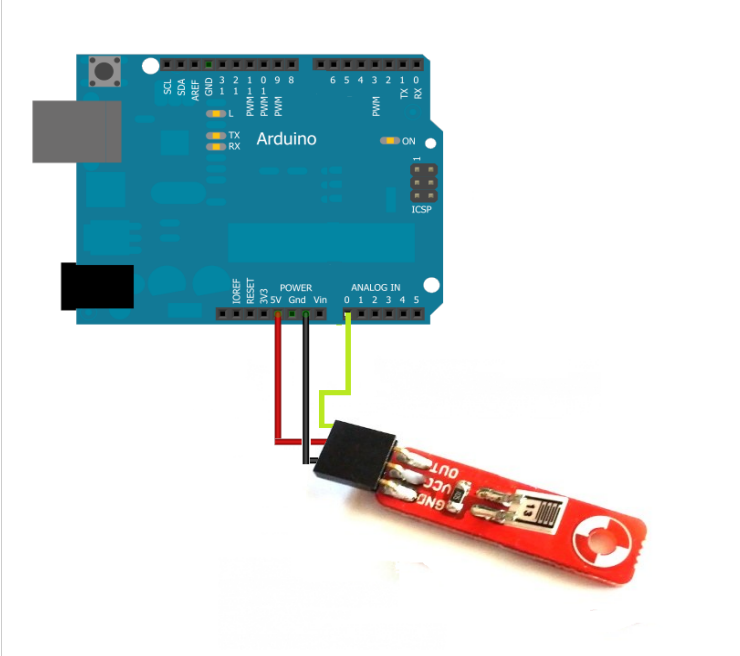
–sigur programul s-a incarcat pe Arduino (ai vazut mesajul "Done Uploading") ?

Senzor Umiditate Brick

Senzorul de umiditate brick este un senzor care ofera o valoare analogica care depind de nivelul de umiditate din mediul ambiat. Din valoarea citita de la senzor se poate deduce valoarea exacta a umiditatii (%), dar deducerea formulei matematice este destul de dificila, asa ca in cele ce urmeaza ne vom multumi sa obtinem un nivel calitativ al nivelului umiditatii (“mai umed”, “mai putin umed”).

Mai exact vom obtine o valoare care variaza in functie de umiditate exact in acelasi mod tot timpul. Daca astazi valoarea citita pe senzor este 453, si ieri a fost tot 453, atunci vom putea spune “astazi umiditatea din aer are acelasi nivel ca si ieri”.

Ca o paranteza, daca ai nevoie de un senzor foarte precis, si care iti ofera direct nivelul umiditatii din aer, in unitati standard, atunci iti sugerez STH15, un senzor etalonat si extrem de precis - http://www.robofun.ro/senzor_temperatura_umiditate_sht15.



Arduino GND	GND Senzor
Arduino 5V	VCC Senzor
Arduino Analog 0	OUT Senzor

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  int nivelUmiditate = analogRead(0);
  Serial.println(nivelUmiditate);
  delay(50);
}
```

Codul sursa de mai sus este aproape identic cu cel de la senzorul de lumina, asa ca nu voi insista

asupra lui.

Ca sa testezi ca lucrurile functioneaza corect, sufla peste senzor, usor. Vei observa ca valoarea pe care o vezi in Serial Monitor se modifica, datorita faptului ca respiratia ta contine vapori de apa. Alta varianta este sa desfaci o sticla de plastic care contine apa pe jumatata, si sa introduci senzorul in prima jumatate a sticlei (NU in apa, in zona in care sticla este goala).

Aceasta a fost lectia 2. In final, as vrea sa te rog sa ne oferi feedback asupra acestei lectii, pentru a ne permite sa le facem mai bune pe urmatoarele.

Este vorba despre un sondaj cu 4 intrebari (oricare este optionala), pe care il poti accesa dand click aici.

Sau ne poti contacta direct prin email la contact@robofun.ro .

Iti multumim,

Echipa Robofun.RO