

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.

Cronometru cu Arduino

În acest tutorial vei descoperi cum se poate construi un cronometru simplu care poate măsura perioada de timp cuprinsă între 2 evenimente. Cronometrul nostru va utiliza un senzor de distanță Sharp ca declansator.

Dacă un obiect trece prin fața senzorului suficient de aproape, atunci cronometrul este declansat. Dacă același obiect trece din nou prin fața senzorului, atunci cronometrul este oprit.

Timpul cronometrat este afișat în Monitorul Serial. Cu acest cronometru poți măsura timpi de ordinul minutelor, secundelor sau chiar milisecundelor.

Ca idee, poți folosi cronometrul pentru a măsura timpii roboților urmăritori de linie.

Vei avea nevoie de următoarele componente:

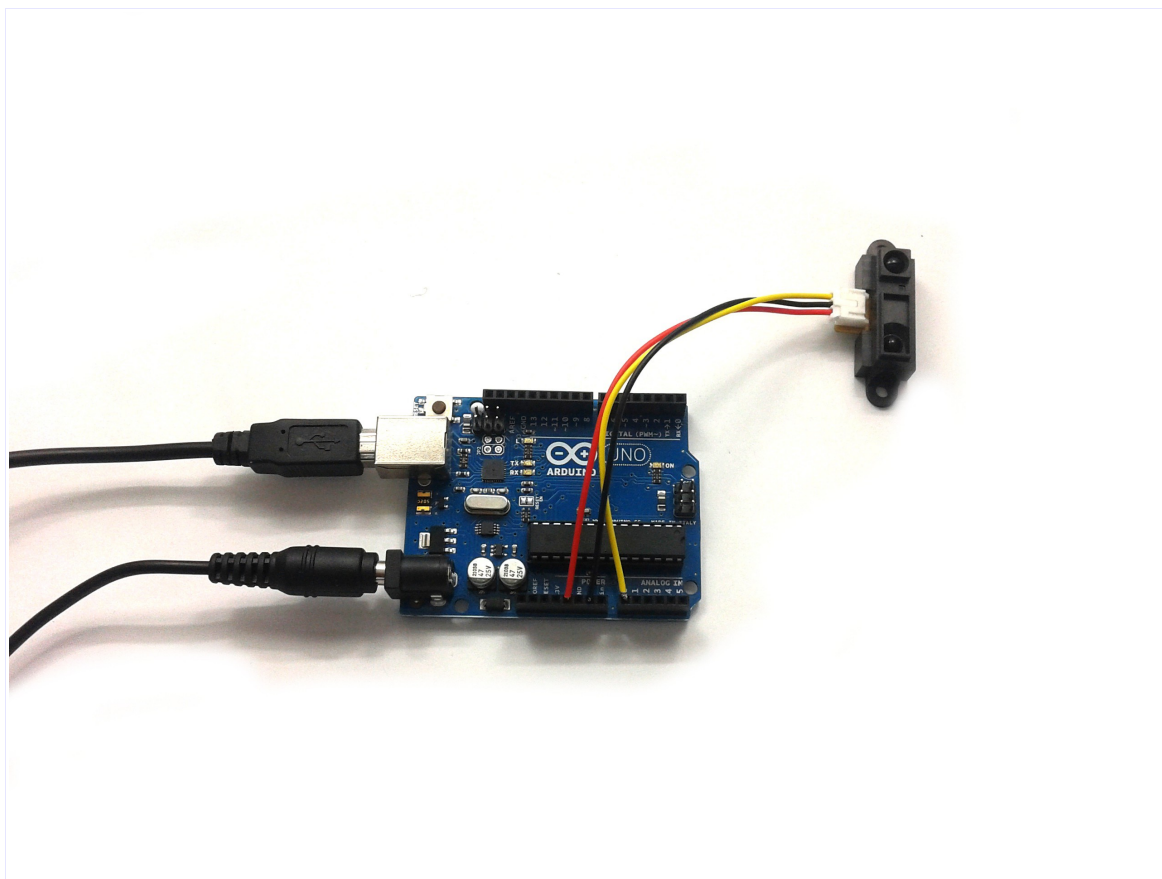
- O placă Arduino - <http://www.robofun.ro/arduino>
- Un senzor de distanță Sharp
http://www.robofun.ro/senzori/infrarosu/senzor_sharp_%20GP2D120XJ00F
- Un cablu senzor 3 fire - http://www.robofun.ro/cablu_senzor_3_fire_12_7_cm

Cum se conectează senzorul ?

Senzorul de distanță Sharp se conectează la placa Arduino urmând tabelul de mai jos:

Arduino GND	Senzor Sharp fir Negru
Arduino 5V	Senzor Sharp fir Rosu
Arduino A0	Senzor Sharp fir Galben

După ce ai conectat senzorul, vei obține o imaginea ca cea de mai jos.

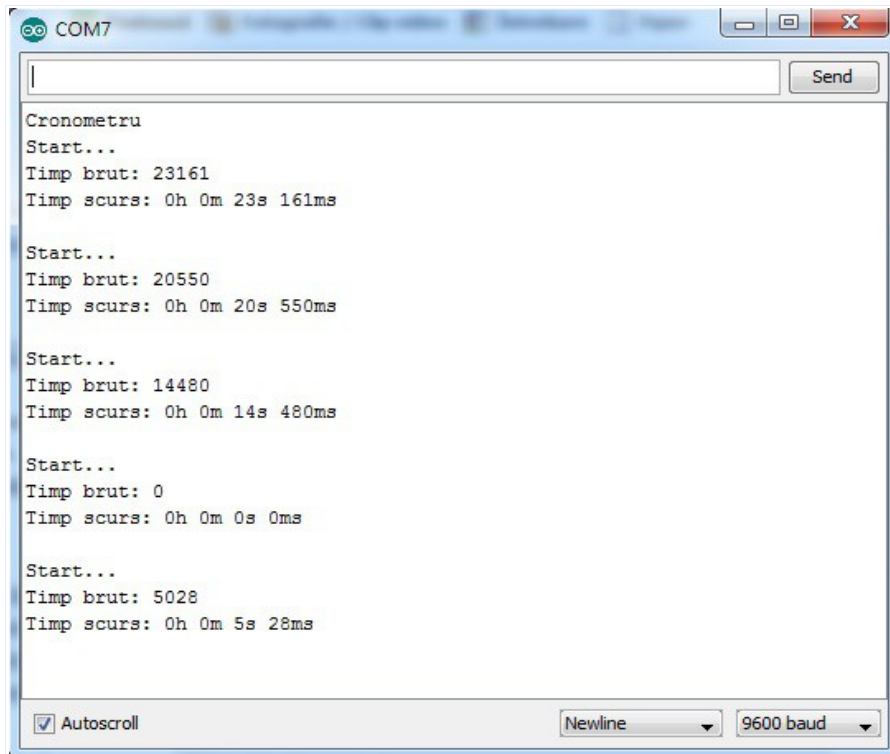


Codul sursa.

Acum este momentul sa incarci sketch-ul in placa Arduino. Copiaza codul sursa de mai jos (copy/paste) si deschide Monitorul Serial.

Modul de functionare al cronometrului este simplu. Placa Arduino citeste in mod constant valoarea senzorului de distanta. Daca valoarea s-a incadrat in pragul presetat din sketch, atunci cronometrul s-a declansat. Cu alte cuvinte, daca senzorul a detectat (sa presupunem) un robot, atunci placa Arduino incepe cronometrarea. Daca acelasi robot trece din nou prin fata senzorului, atunci placa Arduino opreste cronometrarea si o afiseaza pe Monitorul Serial.

Iata un exemplu:



```
unsigned long start, finished, elapsed;  
int IRpin = 0;  
boolean stareSenzor = false;
```

```
void setup()  
{  
  Serial.begin(9600);  
  Serial.println("Cronometru");  
}  
void displayResult()  
{  
  float h, m, s, ms;  
  unsigned long over;  
  elapsed = finished - start;  
  h = int(elapsed / 3600000);  
  over = elapsed % 3600000;  
  m = int(over / 60000);  
  over = over % 60000;  
  s = int(over / 1000);  
  ms = over % 1000;  
  Serial.print("Timp brut: ");  
  Serial.println(elapsed);  
  Serial.print("Timp scurs: ");  
  Serial.print(h, 0);  
  Serial.print("h ");  
  Serial.print(m, 0);
```

```

    Serial.print("m ");
    Serial.print(s, 0);
    Serial.print("s ");
    Serial.print(ms, 0);
    Serial.println("ms");
    Serial.println();
}
void loop()
{
    int pragSenzor = 300;
    if (analogRead(IRpin) > pragSenzor && stareSenzor == false)
    {
        delay(500);
        stareSenzor = true;
        start = millis();
        Serial.println("Start...");
    }
    if (analogRead(IRpin) > pragSenzor && stareSenzor == true) {
        stareSenzor = false;
        finished = millis();
        delay(1000);
        displayResult();
    }
}

```