

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs
CC BY-NC-ND



Codul sursa din acest document este licentiat

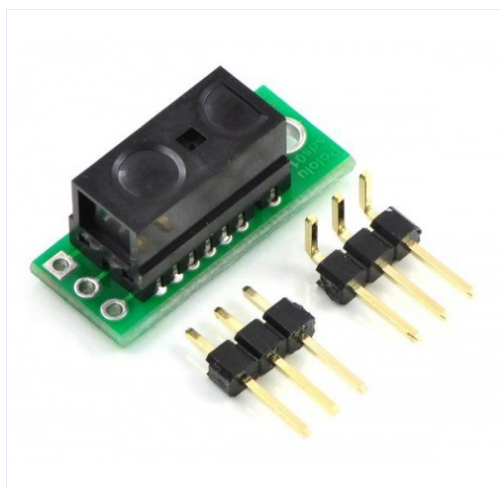
Public-Domain

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.

Raspberry PI si senzorii de distanta Sharp digitali de 5 si 10 cm

Senzorii de distanta Sharp de 5 si 10 cm detecteaza toate obiectele care se afla in raza lor de actiune (ceea ce inseamna obiectele mai apropiate de 5, respectiv 10 cm de senzor). Senzorii au dimensiuni mici, timp de raspuns rapid si consum de energie mic. Iesirea este de tip digitala, adica orice obiect aflat in raza lui de actiune modifica starea iesirii la un 0 logic. Cat timp nici un obiect nu se afla in raza de actiune, iesirea este 1 logic. Conexiunea cu placa Raspberry se realizeaza prin 3 fire cu capete mama-mama.

In acest tutorial vei descoperi cum se conecteaza cei 2 senzori la placa Raspberry PI si cum se citesc datele furnizate intr-un program Python.



Pentru acest tutorial, vei avea nevoie de urmatoarele componente:

- O placa Raspberry PI
<http://www.robofun.ro/raspberry-pi-si-componente/RASPBERRY-PI-B>
- Un alimentator Raspberry PI
<http://www.robofun.ro/raspberry-pi-si-componente/alimentator-raspberry-pi>
- Un senzor Sharp de 5cm (sau un senzor de 10 cm, cel de mai jos)
http://www.robofun.ro/senzori/infrarosu/senzor_distanta_sharp_GP2Y0D805Z0F_5cm
- Un senzor Sharp de 10cm
http://www.robofun.ro/senzori/infrarosu/senzor_distanta_sharp_GP2Y0D810Z0F_10cm
- Fire de conexiune mama-mama

http://www.robofun.ro/fire-conexiune-mama-mama-140mm?keyword=fire&category_id=0

- Breadboard Mini

http://www.robofun.ro/breadboard_mini?keyword=breadboard&category_id=0

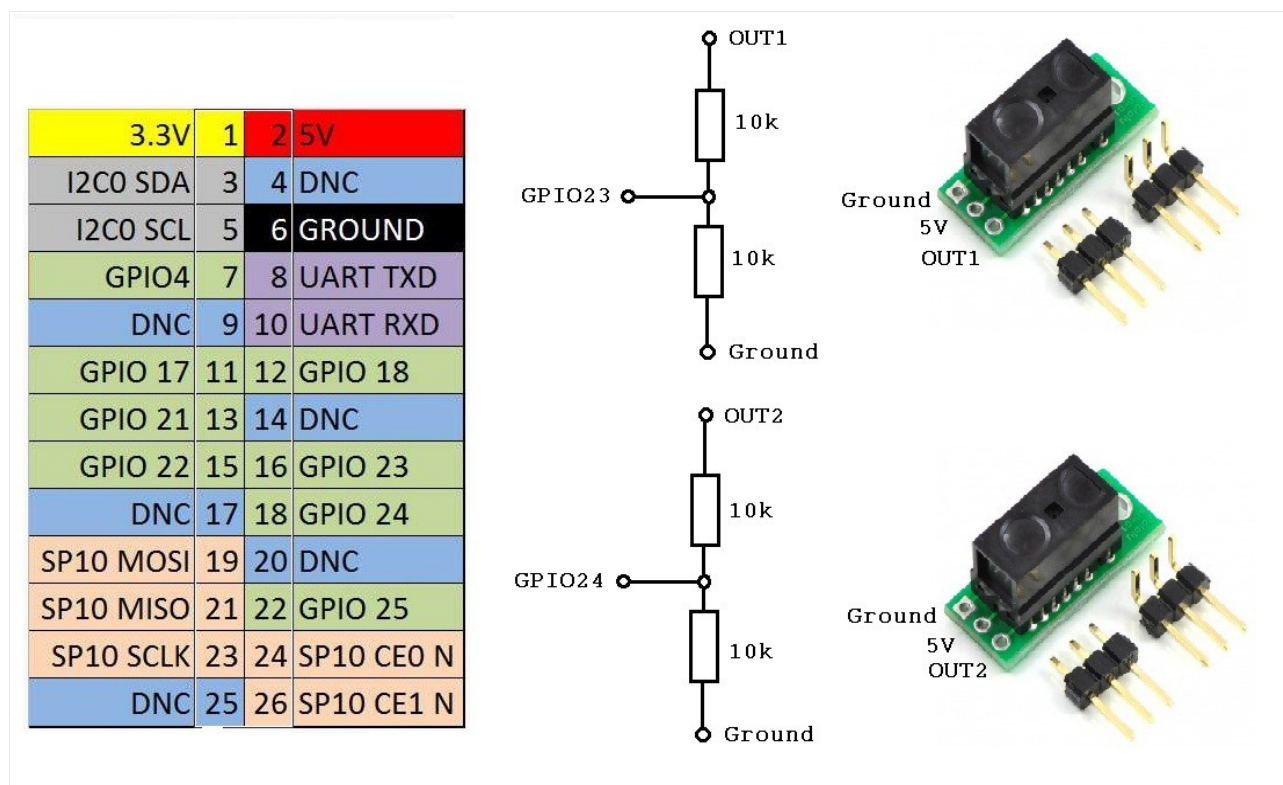
- 4 rezistoare de 10k.

Cum se conecteaza senzorii ?

Nivelele logice sau semnalele generate de catre iesirile celor 2 senzori au tensiuni cuprinse intre 0 si 5V. Mai exact, atunci cand senzorul detecteaza un obiect, acesta va genera o tensiune scazuta de aproximativ 0.6V, iar atunci cand obiectul nu mai este prezent in raza senzorului, iesirea va genera 4.4V.

Placa Raspberry PI interpreteaza semnalele astfel. Orice semnal cu o tensiune cuprinsa intre 2.5 si 3.3V este considerat un 1 logic, iar orice semnal sub 2.5V este considerat un 0 logic. Din cauza ca senzorii tai vor genera mai mult decat poate suporta placa Raspberry PI, adica peste 3.3V, aceasta se poate arde asa ca sunt necesari 2 divizori de tensiune rezistivi. Divizorul rezistiv este un circuit simplu care translateaza nivelele logice de tensiuni mari la nivele logice de tensiuni mici.

Senzorii se conecteaza la placa Raspberry PI urmand schema de mai jos.



Dupa ce ai realizat conexiunile, vei obtine ceva asemanator ca in urmatoarea imagine.

<http://www.robofun.ro/forum>

Programul Python.

Înainte de a trece la scrierea programului Python, trebuie să te asiguri că sistemul de operare al plăcii are cele mai noi update-uri.

1. Update-ul sistemului de operare Raspbian se face prin comenzile:

```
sudo apt-get update && sudo apt-get upgrade
```

2. Mai întâi vei crea un folder în care vei păstra toate fișierele:

```
mkdir senzoriSharp
```

3. Programul Python utilizează un pachet important care accesează portul GPIO al plăcii Raspberry. Instalarea pachetului Rpi.GPIO-0.5.3a se realizează prin următoarea comandă:

```
sudo wget  
http://pypi.python.org/packages/source/R/RPi.GPIO/RPi.GPIO-  
0.5.3a.tar.gz
```

4. Dezarchivează pachetul prin comandă:

```
sudo tar xzf Rpi.GPIO-0.5.3a.tar.gz
```

5. Schimbă locația prin comandă:

```
cd Rpi.GPIO-0.5.3a
```

6. Execută instalarea pachetului:

```
sudo python setup.py install
```

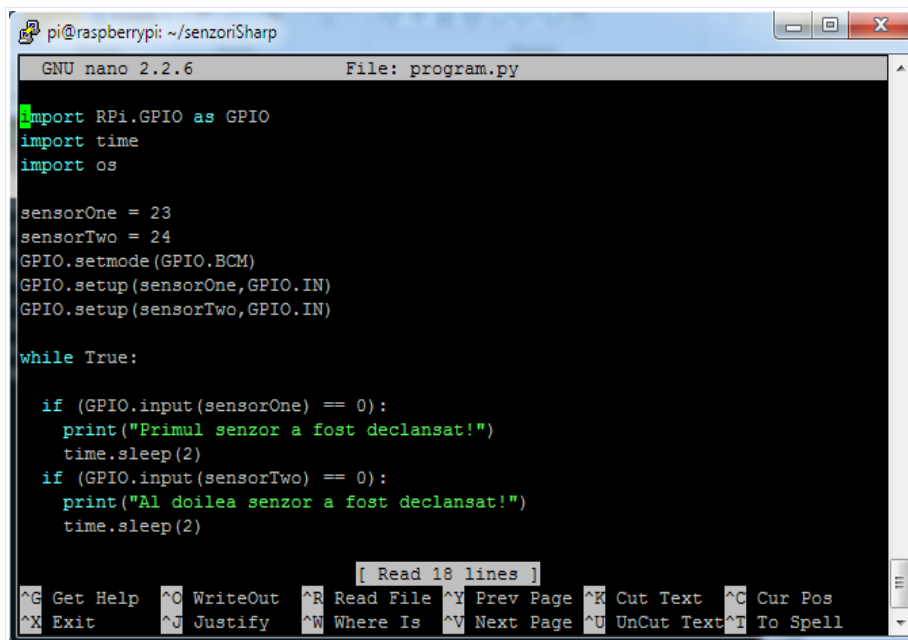
7. Pachetul a fost instalat, acum este momentul să copiezi codul sursă de mai jos în modul următor:

```
sudo nano program.py
```

8. Copiaza codul sursa de mai jos, in fisierul program.py, salveaza-l cu CTRL X si Y si executa-l cu comanda:

```
sudo python program.py
```

Programul arata astfel:



```
pi@raspberrypi: ~/senzoriSharp
GNU nano 2.2.6 File: program.py

import RPi.GPIO as GPIO
import time
import os

sensorOne = 23
sensorTwo = 24
GPIO.setmode(GPIO.BCM)
GPIO.setup(sensorOne,GPIO.IN)
GPIO.setup(sensorTwo,GPIO.IN)

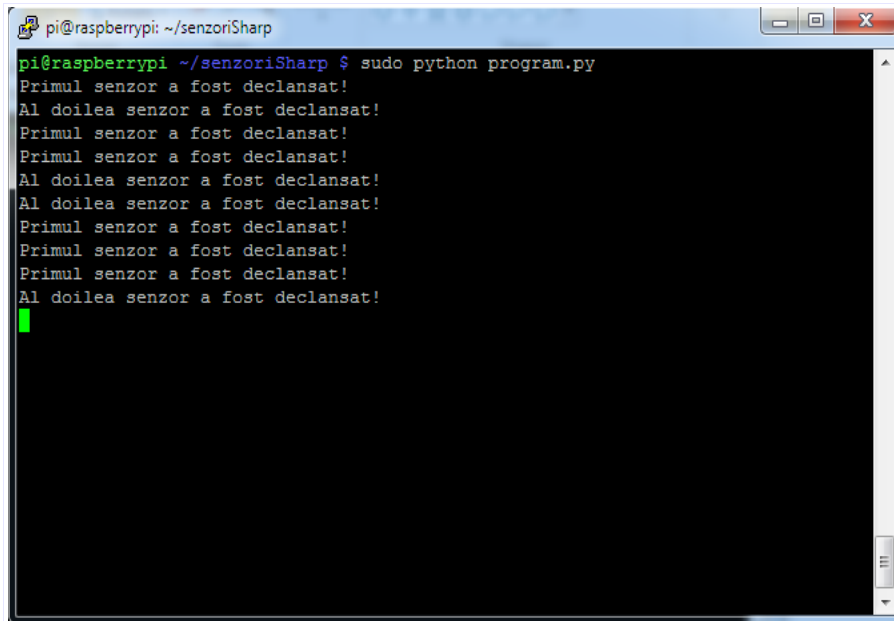
while True:

    if (GPIO.input(sensorOne) == 0):
        print("Primul senzor a fost declansat!")
        time.sleep(2)
    if (GPIO.input(sensorTwo) == 0):
        print("Al doilea senzor a fost declansat!")
        time.sleep(2)
```

[Read 18 lines]

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

Acopera senzorii rand pe rand si vei observa mesajele care apar pe monitor.



```
pi@raspberrypi: ~/senzoriSharp
pi@raspberrypi ~/senzoriSharp $ sudo python program.py
Primul senzor a fost declansat!
Al doilea senzor a fost declansat!
Primul senzor a fost declansat!
Primul senzor a fost declansat!
Al doilea senzor a fost declansat!
Al doilea senzor a fost declansat!
Primul senzor a fost declansat!
Primul senzor a fost declansat!
Primul senzor a fost declansat!
Al doilea senzor a fost declansat!
```

Codul sursa

```
import RPi.GPIO as GPIO
import time
import os

sensorOne = 23
sensorTwo = 24
GPIO.setmode(GPIO.BCM)
GPIO.setup(sensorOne, GPIO.IN)
GPIO.setup(sensorTwo, GPIO.IN)

while True:

    if (GPIO.input(sensorOne) == 0):
        print("Primul senzor a fost declansat!")
        time.sleep(2)
    if (GPIO.input(sensorTwo) == 0):
        print("Al doilea senzor a fost declansat!")
        time.sleep(2)
```