

Gheorghica Gabriel



Titlu proiect: Masurarea distantei cu senzor HC-SR04.

E-mail: gabriel.gheorghica@yahoo.ro

Acest proiect este compus dintr-un modul de măsurare al distanței cu ajutorul unui senzor cu ultrasunete. Un astfel de modul poate fi folosit ca parte integrată a unor proiecte mai mari ce au nevoie de măsurarea distanței(), detector de mișcare, etc.

Proiectul afișează în consolă distanța până la obiectul din față (range: 25mm - 100mm) cu o acuratețe de câțiva milimetri. În cazul în care distanța este mai mică de 50 mm atunci se va aprinde culoarea verde a ledului, în cazul în care distanța este mai mică decât 25 mm atunci se va aprinde culoarea roșie a ledului, în cazul în care distanța este mai mare decât 50 mm atunci se va aprinde culoarea albastră a ledului, iar în cazul în care distanța este mai mare de 100m nu se mai aprinde ledul.

Componentele folosite la acest proiect sunt:

Hardware:

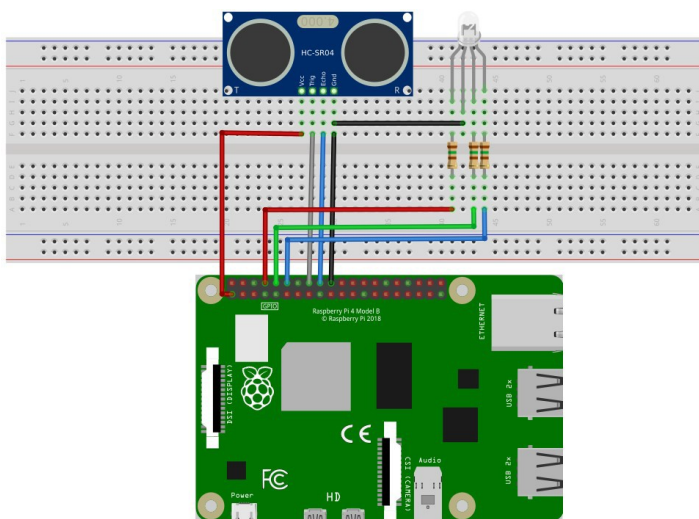
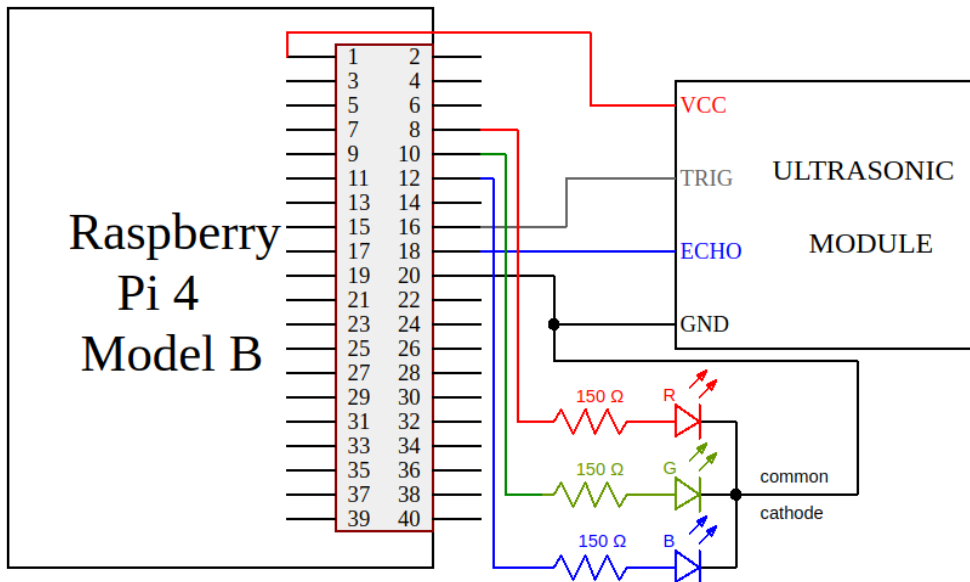
- 1 x senzor ultrasunete HC-SR04+;
- 1 x Raspberry pi 4
- 1 x Breadboard
- 3 x Rezistența 150 ohm
- 1 x RGB Led
- 4 x fire mama-tata
- 4 x fire tata-tata

Software:

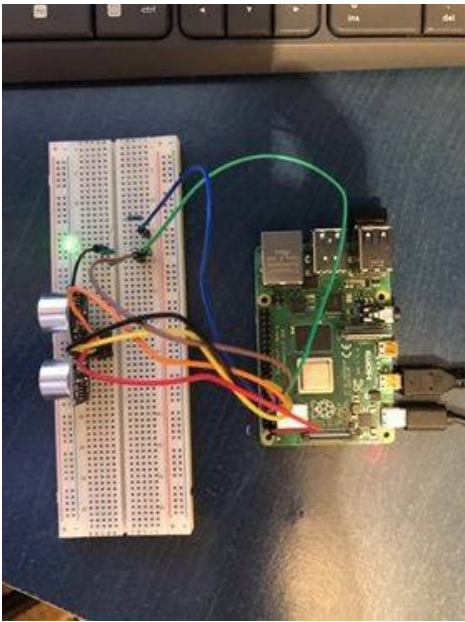
Python3

GPIO library

Schema proiectului:



Circuitul fizic:



Din punct de vedere software proiectul este alcătuit din 3 componente:
Modulul de setare și inițializare ale porturilor GPIO:

```
import time  
import RPi.GPIO as GPIO  
GPIO.setmode(GPIO.BOARD)
```

```
trig = 16  
echo = 18  
redled = 8  
greenled = 10  
blueled = 12
```

```
GPIO.setup(trig, GPIO.OUT)  
GPIO.setup(echo, GPIO.IN)  
GPIO.setup(redled, GPIO.OUT)  
GPIO.setup(greenled, GPIO.OUT)  
GPIO.setup(blueled, GPIO.OUT)
```

În continuare, vom defini o funcție pentru a obține distanța, ca în lecția precedentă,

iar la final vom întoarce această distanță:

```
def calculate_dist():
    GPIO.output(trig, GPIO.HIGH)
    time.sleep(0.00001)
    GPIO.output(trig, GPIO.LOW)
    start = time.time()
    stop = time.time()
    while GPIO.input(echo) == 0:
        start = time.time()
    while GPIO.input(echo) == 1:
        stop = time.time()

    duration = stop - start
    distance = 34300/2 * duration
    return distance

try:
    while True:

        if calculate_dist() < 50:
            GPIO.output(greenled, GPIO.HIGH)
            time.sleep(0.3)
            GPIO.output(greenled, GPIO.LOW)
            time.sleep(0.2)

        if calculate_dist() < 25:
            GPIO.output(redled, GPIO.HIGH)
            time.sleep(0.035)
            GPIO.output(redeled, GPIO.LOW)
            time.sleep(0.025)

        if calculate_dist() > 50:
            GPIO.output(blueled, GPIO.HIGH)
```

```
time.sleep(0.3)
GPIO.output(blueled, GPIO.LOW)
time.sleep(0.2)

if calculate_dist() > 100:
    GPIO.output(blueled, GPIO.LOW)
    GPIO.output(redeled, GPIO.LOW)
    GPIO.output(greenled, GPIO.LOW)

except KeyboardInterrupt:
    pass

#clean all the used ports
GPIO.cleanup()
```