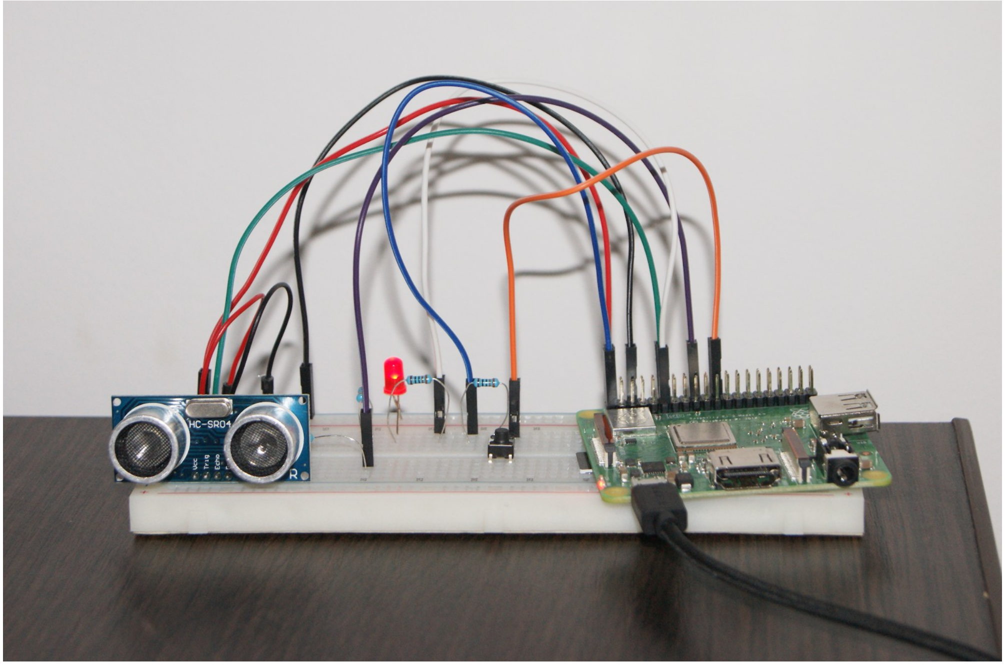
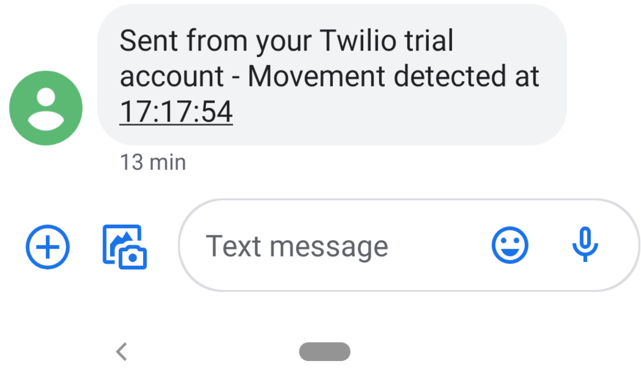
Nume: Leparda Bogdan

Adresa email: lepardabogdan@gmail.com

Poza:

Nume proiect: Senzor de miscare cu trimitere SMS





Sent from your Twilio trial account - Movement detected at 17:17:54

13 min

Acest proiect poate fi o versiune “ieftina” a unui senzor de miscare pentru a oferi un plus de siguranta locuintei voastre. Mai precis, acesta a fost proiectat ca un senzor de deschidere a usii.

Fiind amplasat corect, senzorul ultrasonic poate detecta cand usa se deschide, ledul se va aprinde si pe telefonul dumneavoastra va fi trimis un SMS cu ora exacta la care s-a detectat miscarea. Astfel, veti fi instiintat cand veti primi o “vizita neasteptata” cat timp dumneavoastra nu sunteti acasa. Acest proiect poate lucra cel mai bine impreuna cu o camera de luat vederi pe care sa o puteti accesa de la distanta cand sunteti instiintat de o eventuala miscare in locuinta dumneavoastra. In acest fel, nimic nu va va putea scapa!
Pentru siguranta placutei am adaugat si un buton pentru inchiderea sistemului.

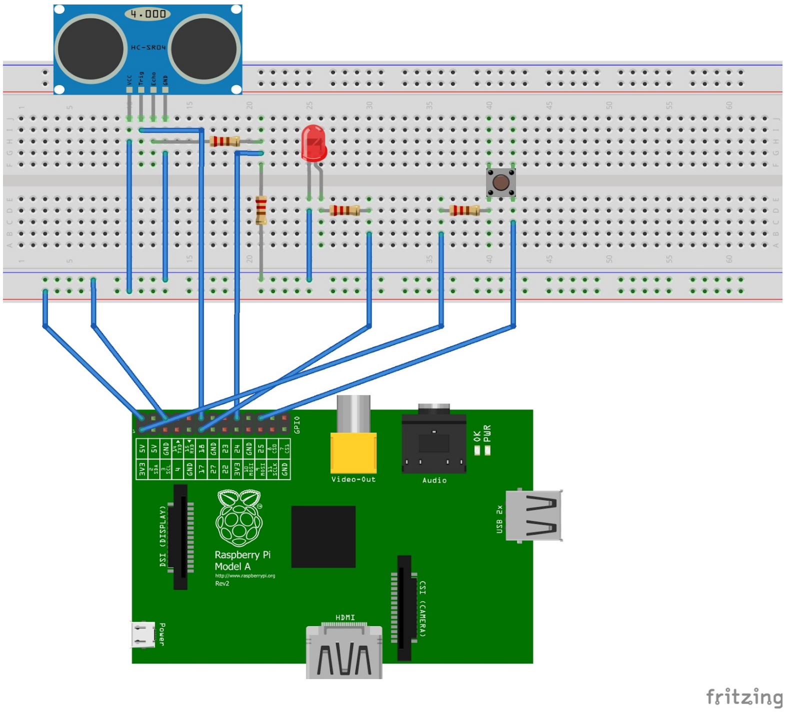
Componente:

- Raspberry Pi 3a+ cu Raspbian OS + alimentator
- Senzor ultrasonic HC-SR04
- Led
- Buton
- 3 rezistori de 100-300 Ohmi
- Breadboard si fire de legatura

Aplicatii:

- Raspbian OS
- Twilio + cont gratuit pe platforma

Schema:



Cod:

distance.py

```
import RPi.GPIO as GPIO
import time
from datetime import datetime
from twilio.rest import Client

account_sid = "AC98d93f113aea957db6e1d22b54ba0de2"
auth_token = "94bac4aa23ad9d50e5ff8c0c58b9445e"

GPIO.setmode(GPIO.BCM)

GPIO_TRIGGER = 18
GPIO_ECHO = 24
GPIO_LED = 17

GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
GPIO.setup(GPIO_ECHO, GPIO.IN)
GPIO.setup(GPIO_LED, GPIO.OUT)

def distance():
    GPIO.output(GPIO_TRIGGER, 1)
    time.sleep(0.0001)
    GPIO.output(GPIO_TRIGGER, 0)

    StartTime = time.time()
    StopTime = time.time()

    while GPIO.input(GPIO_ECHO) == 0:
        StartTime = time.time()

    while GPIO.input(GPIO_ECHO) == 1:
        StopTime = time.time()

    TimeElapsed = StopTime - StartTime

    distance = (TimeElapsed * 34300) / 2
    return distance

def set_dist_ref():
    dist_ref = 0
    time.sleep(0.5)
    dist_ref += distance()
    time.sleep(0.5)
    dist_ref += distance()
    time.sleep(0.5)
    dist_ref += distance()
    dist_ref = dist_ref / 3
    print("Reference distance = %.1f cm" %dist_ref)
    return dist_ref

if __name__ == '__main__':
    try:
        client = Client(account_sid, auth_token)
        print("Please wait, the sensor is configuring...")
        time_to_pass = 0
        movement_detected = False
        times_detected = 0

        dist_ref = set_dist_ref()
        while True:
            time.sleep(0.2)
            dist = distance()
            print("Measured distance = %.1f cm" %dist)
            if(movement_detected == False):
                if (dist_ref - dist > 10 or dist_ref - dist < -10):
                    times_detected += 1
                else:
                    times_detected = 0
                if (times_detected >= 5):
                    now = datetime.now()
                    current_time = now.strftime("%H:%M:%S")
                    print("Movement detected at " + current_time)

                    message = client.api.account.messages.create(
                            to = "+40752657444",
                            from_ = "+12062782879",
                            body = "Movement detected at " + current_time)

                    movement_detected = True
                    time_to_pass = 60
                    times_detected = 0
            if(time_to_pass != 0):
                time_to_pass -= 0.2
            if(time_to_pass <= 0 and movement_detected == True):
                movement_detected = False
                dist_ref = set_dist_ref()

if(movement_detected == True):
                GPIO.output(GPIO_LED, 1)
            else:
                GPIO.output(GPIO_LED, 0)

    except KeyboardInterrupt:
        print("Measurement stopped")
        GPIO.cleanup()
```

shutdown.py

```
from gpiozero import Button
import time
import os

stopButton = Button(25)

while True:
    if stopButton.is_pressed:
        time.sleep(1)
        if stopButton.is_pressed:
            os.system("shutdown now -h")
    time.sleep(1)
```

/etc/rc.local

```
sudo python /home/pi/distance.py &
sudo python /home/pi/shutdown.py &

exit 0
```