Proiect CEL

Dodi Mircea Ovidiu, 321AC

Drept inspiratie pentru proiect, am considerat o situatie reala – protejarea unei case impotriva oaspetilor nedoriti.

Pe baza unui Raspberry Pi 3B, 2 senzori infrarosu si unei camera web am creat un mic dispozitiv ce face poze presupusului intrus care declanseaza unul dintre cei doi senzori.

Schema bloc a ansamblului:

A diagram of a computer

Description automatically generated

Pentru a extinde functionalitatea sa, la detectia prezentei umane, Pi-ul conectat la un difuzor (prin jack-ul de 3.5 mm inclus, HDMI sau Bluetooth) scoate un sunet – o alarma, la un delay de cateva secunde dupa declansarea senzorului – suficient timp pentru a prinde in flagrant presupusul oaspete nepoftit, timp in care este pozat,

Atat timp cat senzorii si camera sunt conectati la Pi, ii putem plasa unde dorim – langa usi, in holuri, langa ferestre sau oriunde credem ca cineva ar putea trece.

Schema componentelor:

A diagram of a circuit board

Description automatically generated

Pentru ce voiam sa fac nu era nevoie de un circuit foarte complicat, Pi-ul usurandu-mi mult treaba.

Cei 2 senzori (HW 201), conform datasheet-ului au tensiunea de alimentare recomandata intre 3.3V si 5V – sunt conectati la cate un pin de 5V. Terminalele de OUT sunt conectate la pinii 20 si 21 si, la declansare provoaca scaderea tensiunii pe cei doi pini (echivalent in cod cu GPIO.LOW) .

Cele doua led-uri sunt inseriate cu cate doua rezistente de 330 ohmi pentru a lucra cu o tensiune potrivita. Nu am gasit pe detaliile placutei ce curent sau tensiune furnizeaza pinii GPIO si nici nu dispuneam de un multimetru, asa ca am determinat empiric rezistenta necesara, cu un led cazand victima experientei.

Led-ul rosu este folosit ca indicator pentru a vedea daca Pi-ul citeste raspunsul de la senzori, si cel verde daca script-ul de Python a inceput sa ruleze. Odata ce acest raspuns este primit, webcamul face 3 poze si un sunet este declansat.

In primele faze ale testarii circuitului, raspunsul pozitiv de la senzori era echivalent cu GPIO.HIGH, dupa cum gaseam firesc. In schimb, la un test, senzorii citeau fals pozitiv incontinuu, ei nefiind declansati (dupa cum aratau led-urile rosii de pe ei). Drept ultima instanta, am schimbat conditia din cod, la GPIO.LOW, si surpinzator, dupa asta mergea asa cum ar trebui.

Codul de Python rulat pe Pi este listat mai jos:

*import* RPi.GPIO *as* GPIO #library pentru lucratul cu pinii

*import* time

*import* os # pentru accesarea driverelor placii de sunet

*import* subprocess #pentru lansarea de subprocese

*import* sys

GPIO.setmode(GPIO.BCM) *# setam modul de numerotare al pinilor: BCM (Broadcom SOC channel)*

*# numim pinii*

PIR\_PIN\_1 = 21

PIR\_PIN\_2 = 19

RED\_LED\_PIN = 6

GREEN\_LED\_PIN = 5

*# setam pinii ca intrari sau iesiri*

GPIO.setup(PIR\_PIN\_1, GPIO.IN)

GPIO.setup(PIR\_PIN\_2, GPIO.IN)

GPIO.setup(RED\_LED\_PIN, GPIO.OUT)

GPIO.setup(GREEN\_LED\_PIN, GPIO.OUT)

*# testul propriu-zis*

*try*:

    print("started detection test")

    GPIO.output(GREEN\_LED\_PIN, GPIO.HIGH)

*# LED-ul verde semnaleaza ca script-ul a pornit*

*# si ca PI-ul asculta pe terminalele 19 si 21 semnalul de la senzori*

*while* True:

*# variabile pentru a stoca raspunsul senzorilor*

        response\_1 = GPIO.input(PIR\_PIN\_1)

        response\_2 = GPIO.input(PIR\_PIN\_2)

*# raspunsul pozitiv al senzorilor este LOW si nu HIGH*

*if* response\_1 == GPIO.LOW:

            print("presence detected by sensor 1")

*if* response\_2 == GPIO.LOW:

            print("presence detected by sensor 2")

*if* GPIO.input(PIR\_PIN\_1) or GPIO.input(PIR\_PIN\_2):

            GPIO.output(RED\_LED\_PIN, GPIO.HIGH) *# aprindem LED-ul rosu daca am detectat o intrare*

            subprocess.run(["fswebcam", "suspect\_1.jpg"], *check* = True)

*# se fac pozele si se salveaza ca suspect\_x.jpg*

            subprocess.run(["fswebcam", "suspect\_2.jpg"], *check* = True)

            subprocess.run(["fswebcam", "suspect\_3.jpg"], *check* = True)

            subprocess.run(["mpg321", "alarm.mp3"])

            GPIO.output(GREEN\_LED\_PIN, GPIO.LOW)

*# stingem LED-ul verde daca am detectat o intrare*

            sys.exit() *# si iesim din executie*

        GPIO.output(RED\_LED\_PIN, GPIO.LOW)

*# stingem LED-ul rosu dupa fiecare ciclu de ascultat senzorii*

        time.sleep(0.1)

*except* KeyboardInterrupt:

    print("stopped detection test")

    GPIO.output(GREEN\_LED\_PIN, GPIO.LOW) *# oprim LED-ul verde*

    GPIO.cleanup()

De la ideea in care proprietarul primeste un prompt sau alarma pe telefon ca a fost detectata prezenta in casa am incercat, pe o tableta ce ruleaza Android Termux (distributie de Linux pe Android ce permite utilizarea unui CLI ca pe PC), conectata prin SSH la Raspberry Pi, sa declansez un sunet pe post de alarma. Am reusit, insa doar prin comanda ca atare apelata in SSH – script-ul de Python rulat pe Pi pentru detectia de la senzori nu imi permitea sa o fac, asa ca am abandonat idea – ar fi trebuit sa configurez un server web pe Pi.

Bill of materials:

* Raspberry Pi 3B – 163 lei <https://www.optimusdigital.ro/en/raspberry-pi-boards/9125-raspberry-pi-3-model-b.html>
* Breadboard – 10 lei <https://www.optimusdigital.ro/ro/prototipare-breadboard-uri/8-breadboard-830-points.html?search_query=breadboard&results=142>
* 4 rezistente 330 ohmi – 0.004 centi
* 2 led-uri - 1 leu <https://www.optimusdigital.ro/ro/optoelectronice-led-uri/697-led-verde-de-3-mm-cu-lentile-difuze.html?search_query=led&results=785>
* Camera web USB – 25 de lei <https://www.emag.ro/camera-web-cu-microfon-interlook-full-hd-usb-jack-negru-x10-480p-black/pd/D7KX78MBM/>?

(Tableta si monitorul la care conectam Pi-ul se presupun a fi ale clientului)

Total: 217 lei