

Senzorul de parcare

Mi-am propus ca acest senzor de parcare sa aibe urmatoarea functionalitate: daca se afla un obstacol la o distanta mai mica de 50 de centimetri de senzor, sa se aprinda un led rosu si sa emita zgomot care sa atentioneze ca exista un obstacol prea aproape; daca senzorul detecteaza un obstacol la o distanta cuprinsa intre 50 de centimetri si 150 de centimetri, atunci se va aprinde un led galben; pentru un obstacol afat la o distanta mai mare de 150 de centimetri dar mai mica de 4 metri se va aprinde un led verde.

Pentru realizarea acestui mic proiect, este nevoie sa intelegem cum functioneaza senzorul ultrasonic de distanta HC-SR04. Acest senzor poate masura distante intre 2cm si 400cm cu precizie care poate ajunge la 3mm. Acesta foloseste ultrasunete pentru a detecta distanta intre el si cel mai apropiat obiect solid. Este format din 2 componente principale:



- un transmitator ultrasonic – transmite ultrasunete
- un receptor ultrasonic – asculta ultrasunetele; daca receptioneaza ultrasunete, produce un impuls la iesire a carui latime poate fi folosita pentru a determina distanta parcursa de ultrasunete.

Senzorul HC-SR04 are 4 pini: VCC – se conecteaza la 5V, Trig – intrarea, Echo – raspunsul rezultat la iesire, GND – masa. Lungimea impulsului receptionat este proportionala cu timpul necesar pentru detectarea semnalului transmis. Trebuie transmis un impuls de 5V pentru cel putin 10 microsecunde pe pinul Trigger. Atunci HC-SR04 raspunde prin transmiterea a 8 impulsuri de 40KHz, care apoi calatoresc prin aer. Intre timp, pinul Echo merge spre valoarea de 5V pentru a forma inceputul semnalului de ecou. Daca impulsul nu este reflectat inapoi, adica daca nu exista niciun obstacol recunoscut de catre senzor, atunci dupa 38 de milisecunde semnalul Echo va reveni la nivelul 0. Daca impulsul este reflectat inapoi atunci pinul Echo ajunge la nivelul 0 cand semnalul este receptionat si produce un impuls a carui latime variaza intre 150 de microsecunde si 25 de milisecunde, in functie de timpul necesar pentru primirea semnalului.

Calculul distantei se face folosind urmatoarea formula: $d = (\Delta t / 2) * c$, unde:

d =distanța la care se afla obstacolul

Δt =intarzierea semnalului

c=viteza sunetului

În primul rând declarăm niște variabile pentru a lucra mai ușor cu pinii pe care îi folosim și pentru a stoca distanța măsurată undeva. Apoi urmează partea de inițializare a pinilor, din funcția `setup()`.

```
int triggerPin=9;
int echoPin=10;
int redLed=3;
int yellowLed=5;
int greenLed=6;
float distance;
int speakerPin=8;

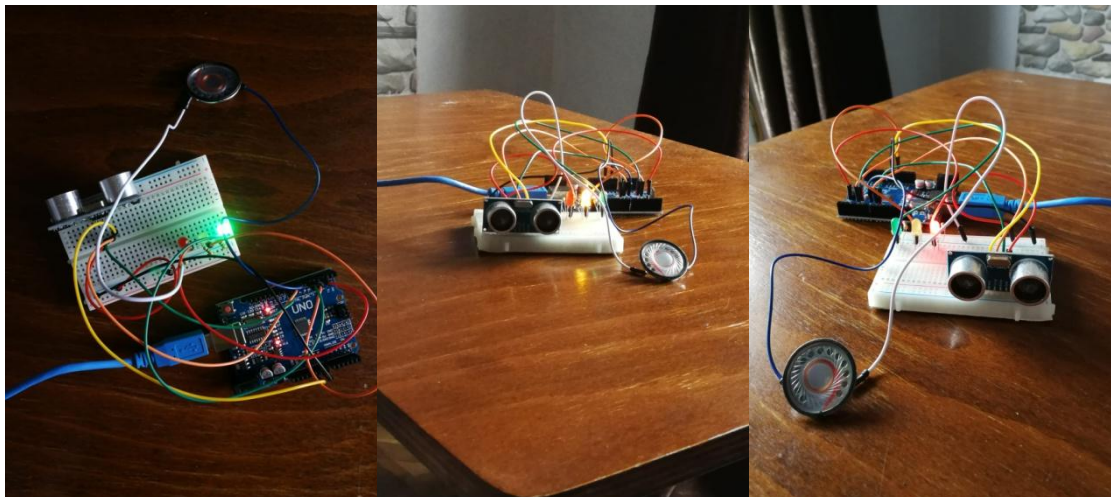
void setup() {
  pinMode(triggerPin,OUTPUT);
  pinMode(echoPin,INPUT);
  pinMode(redLed,OUTPUT);
  pinMode(yellowLed,OUTPUT);
  pinMode(greenLed,OUTPUT);
  pinMode(speakerPin,OUTPUT);
}
```

În funcția `loop()`, transmitem semnalul către pinul Trigger, după cum am spus mai înainte, de 5V, pentru cel puțin 10 microsecunde. Apoi calculăm distanța folosind funcția `pulseIn()` care citește un impuls, fie HIGH, fie LOW și returnează lungimea impulsului în microsecunde. Știind distanța, cu ajutorul unor if-uri comandăm ledurile, și pornim speaker-ul folosind funcția `tone()`.

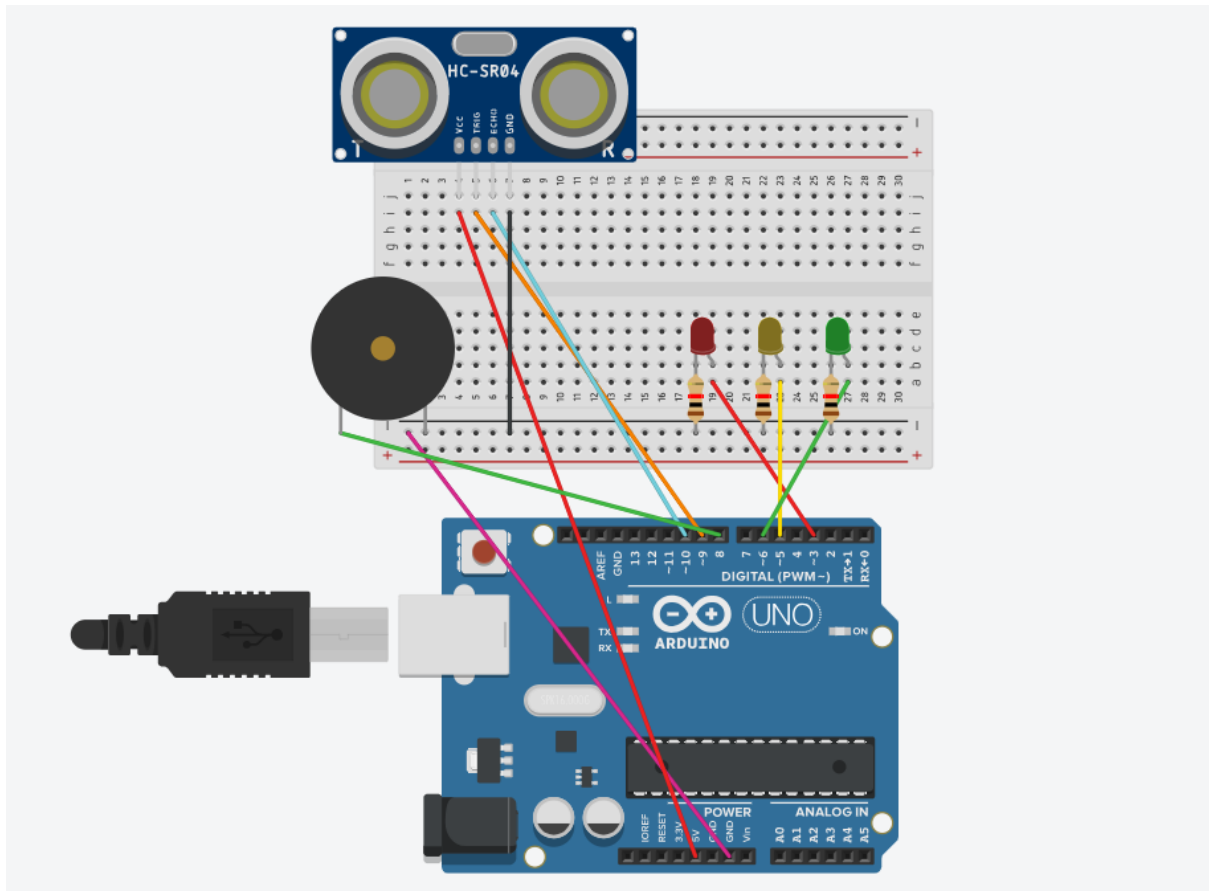
```
digitalWrite(triggerPin,LOW);
delayMicroseconds(2);
digitalWrite(triggerPin,HIGH);
delayMicroseconds(10);
digitalWrite(triggerPin,HIGH);

distance = ( pulseIn(echoPin,HIGH) / 2 ) * 0.0343;
```

Schematicul este foarte ușor, pe breadboard vom pune senzorul HC-SR04, 3 leduri și 3 rezistențe, câte o rezistență pentru fiecare led, ca să ne asigurăm că nu avem un curent prea mare prin leduri deoarece acesta poate duce la arderea ledurilor. Pinii senzorului îi conectăm astfel cu placa: pinul de GND îl ducem în unul din pinii GND, pinul Echo în pinul 10, pinul Trig în pinul 9, iar pinul de VCC îl ducem în pinul de 5V. La cele trei leduri trebuie să avem grijă cum conectăm plusul la plus și minusul la minus, în caz contrar ledurile nu se vor aprinde. Plusul de la speaker este conectat la pinul 8, plusul de la ledul roșu la pinul 3, pe cel de la ledul galben la pinul 5, iar pe cel verde la pinul 6 al placutei, deoarece vrem ca acestea să se aprindă comandat în funcție de distanță. Iar minusul de la fiecare led și minusul de la speaker este conectat la o rezistență, care apoi este legată la minusul de pe breadboard, pe care l-am adus de pe un pin de GND al placutei Arduino.



Schematicul arata in felul urmatoar:



Bibliografie: https://www.youtube.com/watch?v=6F1B_N6LuKw