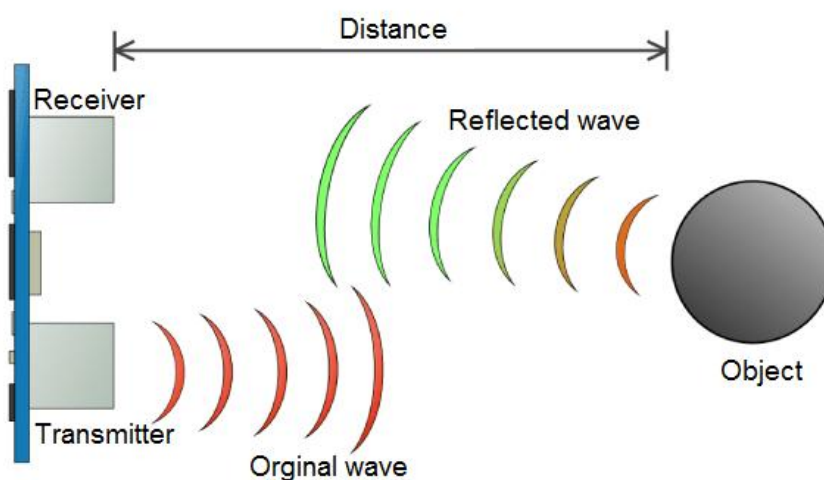


## ۱. کلیات

در این پروژه به نحوه راه اندازی حسگر فراصوت (Ultrasonic) و اندازه گیری فاصله با استفاده از ماژول HC-SR04 پرداخته می شود. این ماژول توانایی اندازه گیری فاصله در بازه ۲ الی ۴۰۰ سانتی متر را دارد. میزان صحت اندازه گیری فاصله با این ماژول در حدود ۳ میلی متر است. همچنین این ماژول توانایی تعیین فاصله با موانعی که زاویه قرارگیری آن ها با خط محوری مرکزی فرستنده امواج فراصوت کمتر از ۱۵ درجه باشد را دارا می باشد. پیاده سازی پروژه به صورتی خواهد بود که فاصله حسگر تا مانع بر حسب سانتی متر اندازه گیری شده و پس از انتقال مقادیر فاصله به کامپیوتر از طریق پورت سریال، امکان قرائت این مقادیر از طریق سربرگ Tools و گزینه Serial Monitor در نرم افزار Arduino IDE میسر می گردد. همچنین، در صورتی که فاصله بین مانع تا حسگر از ۳۰ و ۱۰ سانتی متر کمتر شود، به ترتیب صدای اختار فاصله نزدیک و خیلی نزدیک توسط بلندگو به صدا درآید. هنگامی که یک لایه از مولکول های هوا به جلو رانده می شود، این لایه به نوبه خود لایه دیگری را به جلو می راند و خود به حال اول بر می گردد. لایه جدید نیز لایه دیگری را به جلو می راند و به همین ترتیب این عمل بارها و بارها تکرار می گردد تا انرژی به پایان برسد. این جابجایی مولکول ها اگر بیش از ۱۶ مرتبه در ثانیه تکرار گردد (بسامد بزرگتر از ۱۶) صدا به وجود می آید. گوش انسان توانایی دریافت امواج صوتی در محدوده بسامدی ۲۰ هرتز الی ۲۰ کیلو هرتز را دارد. بسامد امواج فراصوت خارج از محدوده شنوایی انسان قرار دارد و از ۲۰ کیلو هرتز شروع می شود و تا ۱۰ مگاهرتز امتداد دارد. حسگرهای پیزوالکتریک (Piezoelectric) رایج ترین نوع ابزار به منظور ایجاد و دریافت امواج فراصوت می باشند. ماژول های فراصوت اندازه گیر فاصله معمولاً دارای یک فرستنده و یک گیرنده پیزوالکتریک هستند. امواج فرستاده شده از فرستنده پیزوالکتریک پس از برخورد با مانع به سمت ماژول بر می گردند و توسط گیرنده پیزوالکتریک دریافت می شوند (شکل ۱). از این طریق و با در نظر گرفتن زمان بازگشت موج و کیفیت امواج بازتابی می توان به اطلاعاتی راجع به عمق، نوع و سرعت مانع دست یافت. ماژول فراصوت HC-SR04 دارای چهار پایه به نام های Trig، VCC، Echo و GND می باشد. پایه Trig جهت ارسال امواج و پایه Echo جهت دریافت امواج فراصوت استفاده می شود. ولتاژ مناسب جهت راه اندازی این حسگر برابر ۵ ولت بوده و از طریق پایه VCC تغذیه صورت می گیرد (شکل ۲). در هر بار تحریک ماژول توسط آردوینو، فرستنده پیزوالکتریک ۸ موج مربعی با بسامد ۴۰ کیلوهرتز را ارسال می نماید. پس از قطع ارسال امواج از فرستنده پیزوالکتریک، زمان سنج (Timer) داخلی پردازنده برد آردوینو شروع به شمارش می کند تا زمانی که امواج برگشتی از مانع به صورت کامل توسط گیرنده پیزوالکتریک دریافت شود. زمان شمارش شده معادل مدت زمان رفت و برگشت امواج فراصوت در هوا از ماژول تا مانع می باشد. از آنجا که میزان سرعت صدا در هوا برابر با ۳۴۳ متر بر ثانیه (۰/۰۳۴۳ سانتی متر بر میکرو ثانیه) است، مقدار فاصله بین ماژول تا مانع با استفاده از رابطه زیر تعیین می شود:

$$\text{Distance (cm)} = [\text{Duration time (microseconds)} \times 0.0343 \text{ (cm/microseconds)}] / 2$$

که در رابطه فوق، Duration time زمان مابین ارسال و دریافت امواج فراصوت می باشد.



شکل ۱. نحوه عملکرد فرستنده و گیرنده امواج فراصوت به منظور اندازه گیری فاصله



شکل ۲. ساختمان ماژول فراصوت اندازه گیر فاصله HC-SR04

اهداف زیر از انجام این پروژه مدنظر است:

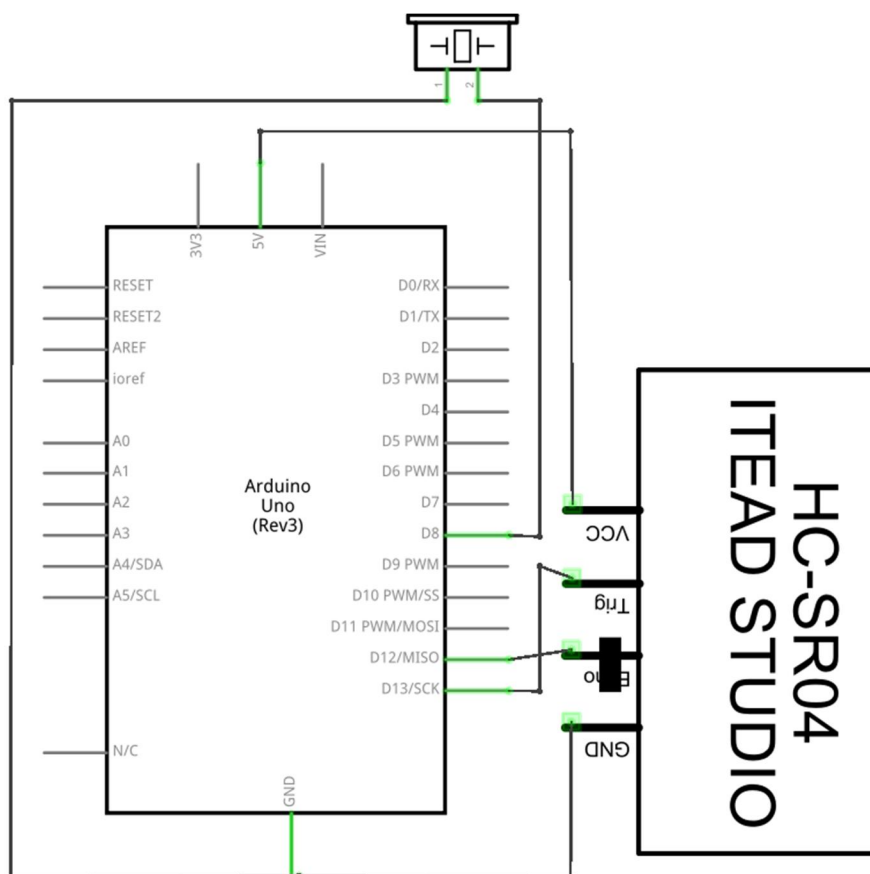
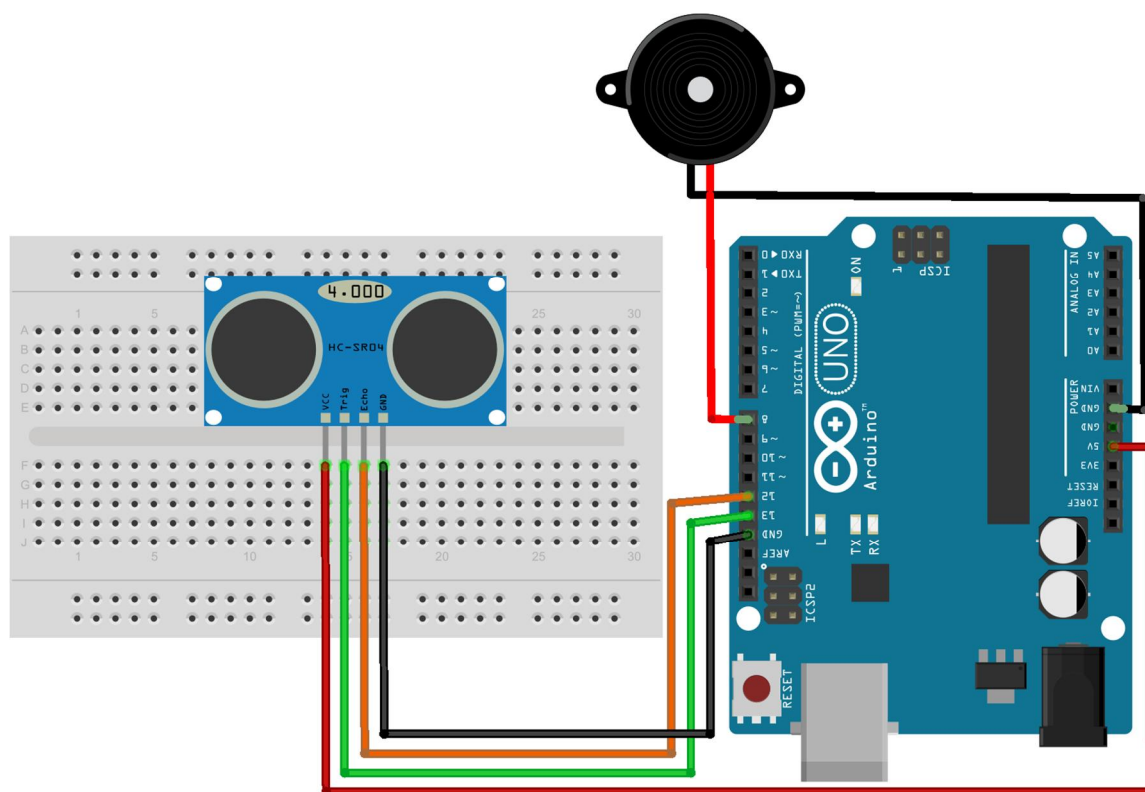
- آشنایی با نحوه ایجاد امواج فراصوت
- آشنایی با نحوه پیکربندی ماژول فراصوت اندازه گیری فاصله
- افزایش مهارت در کار با قطعات الکترونیک و بستن مدار

## ۲. قطعات و لوازم مورد نیاز

- برد مورد (۱ عدد)
- سیم Jumper (۶ عدد)
- ماژول فراصوت HC-SR04 (۱ عدد)
- بلندگو ۸ اهم (۱ عدد)
- برد آردوینو Uno (۱ عدد)

## ۳. پیاده سازی مدار و برنامه نویسی

مدار پروژه اندازه گیری فاصله با ماژول فراصوت بر اساس آنچه در شکل ۳ آمده است قابل پیاده سازی است. پس از بستن مدار، برنامه مدنظر بر اساس آنچه در شکل ۴ آمده است جهت پروگرام کردن برد آردوینو استفاده می شود. توجه شود که حتماً فایل کتابخانه ای آردوینو مربوط به نت های موسیقی (pitches.h) در پوشه ای که برنامه آردوینو در آن قرار دارد بایستی موجود باشد تا برنامه Arduino IDE قادر به کامپایل کردن برنامه مربوط به راه اندازی بلندگو گردد.



شکل ۳. نحوه پیاده سازی مدار پروژه اندازه گیری فاصله با ماژول فراصوت

```
#include "pitches.h"
int TrigPin = 13;
int EchoPin = 12;
int melody_1[] = {NOTE_C4, NOTE_C5, NOTE_A3, NOTE_A4, NOTE_AS3, NOTE_AS4, 0, 0};
int noteDurations_1[] = {1000/4, 1000/8, 1000/8, 1000/4, 1000/4, 1000/4, 1000/4, 1000/4};
int melody_2[]={NOTE_G6, NOTE_E7, NOTE_G7, NOTE_A7, 0, NOTE_F7, NOTE_G7, 0, NOTE_E7, 0, NOTE_C7};
float noteDurations_2[]={1000/8, 1000/8, 1000/8, 1000/12, 1000/8, 1000/8, 1000/8, 1000/12, 1000/8, 1000/8, 1000/8};
unsigned long duration, distance;

void setup() {
  pinMode(TrigPin, OUTPUT);
  pinMode(EchoPin, INPUT);
  Serial.begin(9600);
}

void loop() {
  digitalWrite(TrigPin,LOW); //Stop triggering signal
  delayMicroseconds(2);
  digitalWrite(TrigPin,HIGH); //Start triggering signal
  delayMicroseconds(5);
  digitalWrite(TrigPin, LOW); //Stop triggering signal
  duration = pulseIn(EchoPin, HIGH); //Duration between send and receive of ultrasonic wave (micoroseconds)
  distance = (duration*0.0343)/2; // Distance between ultrasonic module and obstacle (cm)
  Serial.print("Distance to obstacle (cm)= ");
  Serial.println(distance);
  if (distance < 10){
    SoundSignal_2(); //Active Alarm2 music
    delay(100);
  }
  else if(distance < 30){
    SoundSignal_1(); //Active Alarm1 music
    delay(100);
  }
}

//Alarm1 music
void SoundSignal_1(){
  for (int i = 0; i < 8; i++) {
    tone(8, melody_1[i], noteDurations_1[i]);
    delay(noteDurations_1[i] * 1.30);
    noTone(8);
  }
}

//Alarm2 music
void SoundSignal_2(){
  for (int i = 0; i < 11; i++) {
    tone(8, melody_2[i], noteDurations_2[i]);
    delay(noteDurations_2[i] * 1.30);
    noTone(8);
  }
}
```

شکل ۴. برنامه پروژه اندازه گیری فاصله با ماژول فراصوت