

راه اندازی یک پودمان پر کاربرد

در این قسمت می‌خواهیم یک ماژول پر کاربرد را با هم راه‌اندازی کنیم که در بسیاری از پروژه‌های رباتیک هم کاربرد دارد. این حسگر خوش‌قیافه که بیشتر برای سنجش فاصله بین یک جسم تا خودش کاربرد دارد، برای پروژه‌هایی مثل متر الکترونیکی، یا عمق‌سنج یا ربات با قابلیت تشخیص مانع به کار می‌رود. مرحله‌های راه‌اندازی این ماژول را با هم بررسی می‌کنیم!

حسگر فاصله‌سنج اولتراسونیک srf05

حسگر اولتراسونیک یا حسگر فاصله‌سنج فراصوت، برای محاسبه فاصله اجسام به کار می‌رود. این حسگر با فرستادن یک تپ (پالس) صوتی کوتاه و دریافت آن پس از انعکاس از سطح جسم، زمان رفت و برگشت امواج صوتی در هوا را محاسبه می‌کند!



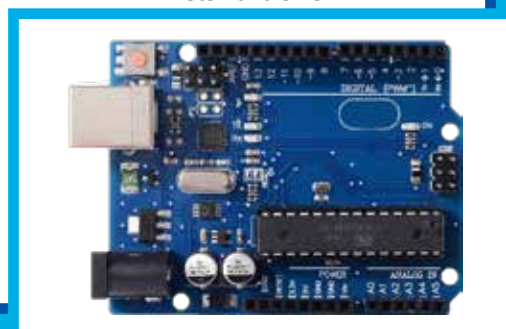
این حسگر یک گیرنده و یک فرستنده دارد که با فرستادن موج صوتی توسط حسگر فاصله‌سنج اولتراسونیک و محاسبه زمان بازگشت آن پس از برخورد با مانع، می‌توان فاصله از مانع را به دست آورد. لازم به ذکر است، مسافت طی شده توسط موج فراصوتی از ارسال تا برگشت، دو برابر فاصله حسگر از مانع است؛ در نتیجه، برای به دست آوردن فاصله، باید مسافت طی شده توسط موج، در بازه ارسال تا دریافت موج را بر دو تقسیم کرد. این حسگر می‌تواند جسم‌هایی در فاصله ۲ تا ۳۰۰ سانتی‌متر را با دقت قابل قبول تشخیص دهد.

قطعات مورد نیاز

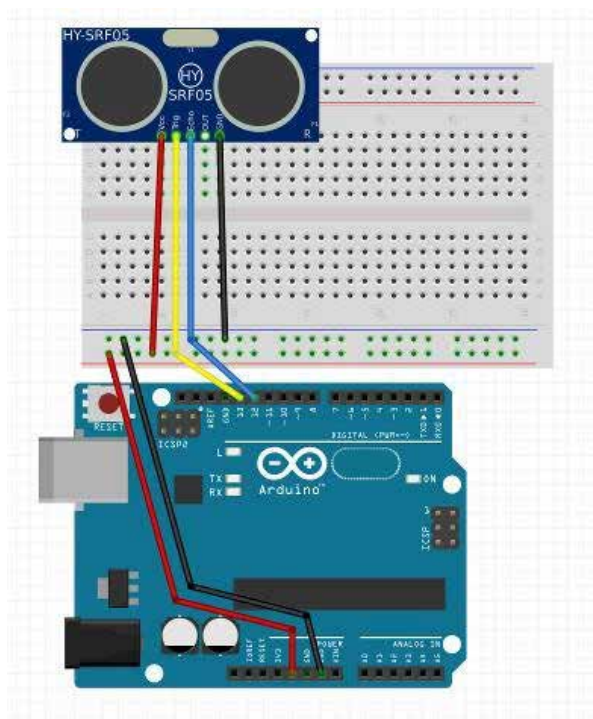
اولتراسونیک



برد آردوینو ماژول



اتصالات مدار



برنامه نویسی

```
int TRIG_PIN=13; int ECHO_PIN=12; long duration; int distance;

void setup
pinMode(TRIG_PIN, OUTPUT); pinMode(ECHO_PIN, INPUT);
Serial.begin(9600);

void loop
  digitalWrite(TRIG_PIN, LOW); delayMicroseconds(2); digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10); digitalWrite(TRIG_PIN, LOW);
  duration= pulseIn(ECHO_PIN, HIGH); int distance= duration*0.0342/;

  Serial.print(«distance to nearest object:»);
  Serial.println(distance);
  Serial.println(« cm»);

  delay(1000);
```

شرح برنامه

```
int TRIG_PIN=13; int ECHO_PIN=12; long duration; int distance;
```

پینی را که با آن موج فراصوت را می فرستیم، «TRIG_PIN 13» نام گذاری می کنیم.
پینی را که توسط آن موج فراصوت را دریافت می کنیم، با نام «ECHO_PIN 12» مشخص می کنیم.

یک متغیر با نام duration را برای ذخیره زمان طی شده توسط موج ایجاد می‌کنیم. چون سرعت موج فراصوت بسیار زیاد است، این زمان را برحسب میکروثانیه ذخیره می‌کنیم. به دلیل اینکه میکروثانیه حافظه زیادی اشغال می‌کند، این متغیر را از نوع long می‌سازیم. یک متغیر نیز به اسم distance از نوع int برای ذخیره فاصله می‌سازیم.

```
void setup() { pinMode(TRIG_PIN, OUTPUT); pinMode(ECHO_PIN, INPUT); Serial.begin(9600);
```

در قسمت (void setup) پینی را که موج را می‌فرستد، به‌عنوان خروجی و پینی را که موج را دریافت می‌کند، به‌عنوان ورودی دریافت می‌کنیم. نرخ انتقال اطلاعات را نیز روی ۹۶۰۰ تنظیم می‌کنیم.

```
void loop
  digitalWrite(TRIG_PIN, LOW); delayMicroseconds(2); digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10); digitalWrite(TRIG_PIN, LOW);
  duration= pulseIn(ECHO_PIN, HIGH); int distance= duration*0.0342/;

  Serial.print(«distance to nearest object:»);
  Serial.println(distance);
  Serial.println(« cm»);

  delay(300);
```

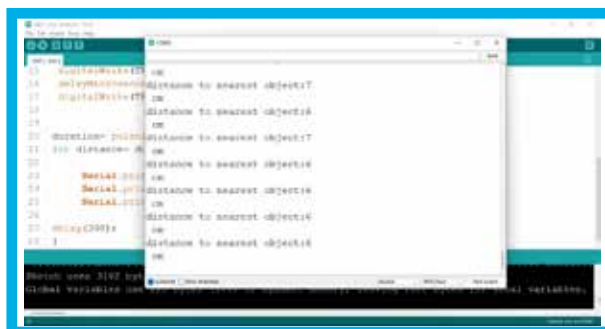
در این قسمت، ابتدا پینی که وظیفه ارسال موج را دارد، به مدت دو میکروثانیه پایین می‌آوریم (low) که مطمئن شویم موج ارسال نمی‌شود. سپس این پین را به مدت ده میکروثانیه زیاد (HIGH) می‌کنیم که موج فرستاده شود.

با استفاده از دستور (pulseIn (ECHO_PIN, HIGH) مقدار زمانی را که طول می‌کشد این موج مسیر رفت و برگشت را طی کند، مشخص می‌کنیم. این تابع دو متغیر می‌گیرد که اولی شماره پین موردنظر و دومی می‌تواند HIGH یا LOW باشد. به‌طور مثال، در این کد بالا آردوینو منتظر می‌ماند که پین ECHO_PIN HIGH شود. خروجی این تابع به‌صورت میکروثانیه است.

موج فراصوت در هر میکروثانیه مقدار ۰/۰۳۴ ثانیه را طی می‌کند. بنابراین، مقدار زمان طی‌شده را که در خط قبل در متغیر duration ذخیره کرده‌ایم، در این عدد ضرب می‌کنیم تا مسافت طی‌شده توسط موج را محاسبه کنیم. در ضمن، چون مقدار مسافت با میزان رفت و برگشت برابر است، برای به‌دست آوردن فاصله، این مقدار را تقسیم بر دو می‌کنیم. این مقدار را در تغییر distance ذخیره می‌کنیم.

در خط‌های بعدی کد نیز مقدار فاصله را با استفاده از دستورات گفته‌شده در زنجیره نمایشگر نمایش می‌دهیم.

خروجی



به این ترتیب، ماژول اولتراسونیک یا همان فاصله‌سنج با هم راه‌اندازی کردیم!