

به نام خدا

گزارش پروژه آزمایشگاه سخت افزار

گروه ۶
آترين آريا ۹۶۱۰۹۹۰۶
يوسف شکیبا ۹۶۱۰۹۷۴۷

پايهز ۱۴۰۰

فهرست مطالب

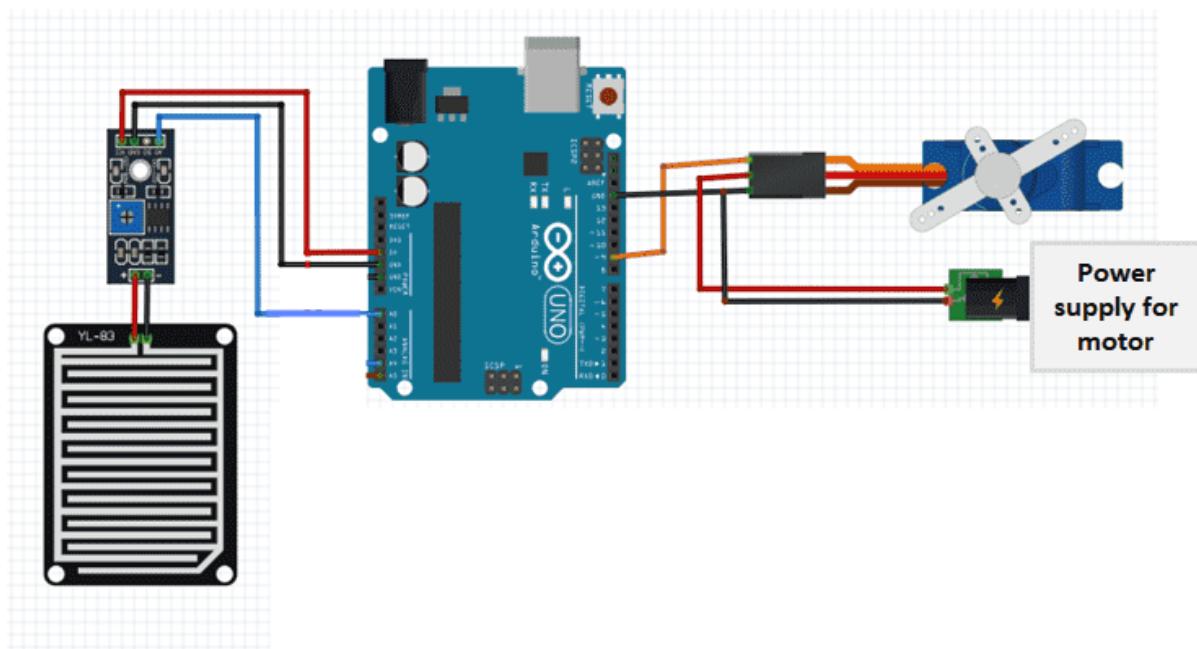
۱	مقدمه	۱
۲	ماژول سروو موتور MG90S	۱.۱
۲	مشخصات فنی	۱.۱.۱
۲	شیوه اتصال	۲.۱.۱
۳	ماژول سنسور باران MH-RD	۲.۱
۳	مشخصات فنی	۱.۲.۱
۳	شیوه اتصال	۲.۲.۱
۴	کد برف پاک کن	۲
۴	تنظیمات اولیه	۱.۲
۴	تابع setup	۲.۲
۵	تابع wipe	۳.۲
۵	تابع loop	۴.۲
۶	وصل کردن قطعات	۳
۷	آزمایش عملکرد	۴

چکیده

فصل ۱

مقدمه

در این پروژه قصد داریم به طراحی و پیاده سازی برف پاک کن هوشمند بپردازیم. این برف پاک کن با استفاده از یک سنسور تشخیص باران در صورت باران به سرو و موتور سیگنال چرخش برای پاک کردن باران می دهد. معماری این سیستم در بلوک دیاگرام زیر آورده شده است.



شکل ۱ : بلوک دیاگرام سیستم

در این سیستم به یک سنسور تشخیص باران، یک آردینو اونو و یک سرو و موتور نیاز است. لیست دقیق قطعات به صورت دقیق در جدول زیر آمده است.

نوع	نام	مقدار
Microcontroller	R3 Uno Arduino	۱
Motor Servo	MG90S	۱
Sensor Rain	MH-RD	۱
Wire	Wires Jumper	۶۰

جدول ۱.۱ : قطعات مورد نیاز

در ادامه به توضیح سنسور و موتور استفاده شده و نحوه ای اتصال آن ها به آردوینو می پردازیم.

۱.۱ مژول سرو و موتور MG90S

این مژول با گرفتن سیگنال ورودی از آردوینو در جهت تعیین شده با سرعت مشخص شده پرهی متصل شده را می چرخاند.

۱.۱.۱ مشخصات فنی

این مژول ۱۳ گرم وزن دارد و ابعاد آن به میلیمتر در شکل زیر آمده است.



شکل ۲: مژول MG90S

پره به اندازه ۹۰ درجه در هر جهت می چرخد. این مژول قابلیت چرخاندن ۶۰ درجه در ۰.۱ ثانیه را دارد. این مژول با ۴.۸ ولت کار می کند.

۲.۱ شیوه اتصال

این مژول علاوه بر پین های VCC و GND یک پین سیگنال ورودی دارد. این پین به دلیل PWM بودن به پین دیجیتال شماره ۸ آردوینو وصل می شود و سیگنال ها را دریافت می کند.

۲.۱ مژول سنسور باران MH-RD

همانطور که از نام این سنسور مشخص است، شما می توانید توسط این سنسور شروع بارش باران یا وجود آب را در مکانی حس کنید و با توجه به آن، عملیات یا واکنش های مورد نظر را در محیط اعمال کنید. خروجی آنالوگ این سنسور با توجه به میزان آب واکنش نشان میدهد. به این شکل که در حالت کاملا خشک ولتاژ ۵ ولت را در پین خروجی دارد و در حالتی غیر از این ولتاژ پایین تر را نشان خواهد داد.

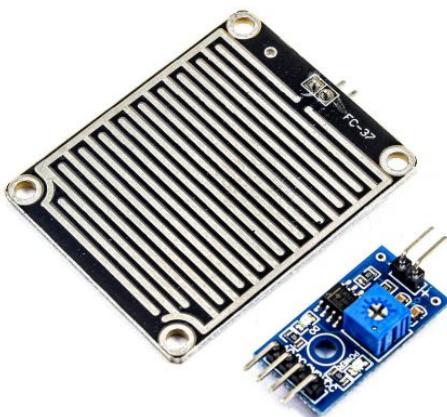


Photo by CafeRobot

شکل ۳: سنسور باران MH-RD

۱.۲.۱ مشخصات فنی

ولتاژ کاری آن ۵ ولت و ولتاژ خروجی آن از صفر تا پنج ولت به صورت آنالوگ است. دارای نشانگر LED برای نمایش رطوبت بیش از حد مجاز و ابعاد برد تشخیص قطرات باران آن $54\text{mm} \times 40\text{mm}$ می باشد.

۲.۲.۱ شیوه‌ی اتصال

ماژول دارای ۴ پین است: Vcc و Gnd و A₀ و D₀. Vcc و Gnd به Supply می‌باشند. آردوینو متصل می‌شود. از آنجا که اطلاعات پیوسته بدرد ما می‌خورد تنها از پین A₀ که آنالوگ است استفاده می‌کنیم و این پین را به که یکی ورودی‌های آنالوگ آردوینو متصل می‌کنیم.

فصل ۲

کد برف پاک کن

در این قسمت کدهای مربوط به برف پاک کن همراه با توضیحات قرار گرفته است.

۱.۲ تنظیمات اولیه

```
#include <Servo.h>

Servo myservo;

int pos = 0;
int sensorValue = 0;
```

شکل ۴: تنظیمات اولیه

کتابخانه Servo.h برای استفاده از عملیات‌های Servo استفاده شده است. همچنین در ابتدا موقعیت Servo هم بر روی صفر تنظیم شده است.

۲.۲ تابع setup

```
void setup() {
    Serial.begin(9600);
    myservo.attach(9);
}
```

شکل ۵: تابع setup

تابع setup که پس از روشن شدن آردوینو صدای زده می‌شود برای مقدارگزاری‌های اولیه استفاده می‌شود. همان طور که می‌بینید سرعت پورت سریال بر اساس بیت بر ثانیه مقدارگذاری می‌شود. علاوه بر آن پین دیجیتال شماره ۹ که یک پین PWM است به سرو و موتور اختصاص داده می‌شود.

۳.۲ wipe تابع

```
void wipe(int Speed)
{
    if(Speed>400){Speed=5000;}
    if(Speed>350&&Speed<=400){Speed=3000;}
    if(Speed<=350){Speed=500;}
    for (pos = 180; pos >= 0; pos--) {
        myservo.write(pos);
        delay(3);
    }
    for (pos = 0; pos <= 180; pos++) {
        myservo.write(pos);
        delay(3);
    }
    delay(Speed);
}
```

شکل ۶: تابع wipe

در این تابع، در ابتدا بسته به ورودی که آن هم نتیجه‌ی سنسور باران است، میزان زمانی که باید بین دو حرکت برف پاک کن فاصله‌ی زمانی ایجاد کند تعیین می‌شود. همانطور که در قسمت کد مشخص است برف پاک کن سه حالت دارد. در دو حلقه‌ی بعدی هم یک حرکت کامل برف پاک کن انجام می‌شود که یک دور رفت و یک دور بازگشت است و در انتها هم آن میزان زمانی که قرار است فاصله‌ی زمانی ایجاد شود صبر می‌کند.

۴.۲ loop تابع

```
void loop() {
    sensorValue = analogRead(A0);
    if(sensorValue>600) {
        myservo.write(180);delay(1000);
    }
    if(sensorValue<=600){
        wipe(sensorValue);
    }
}
```

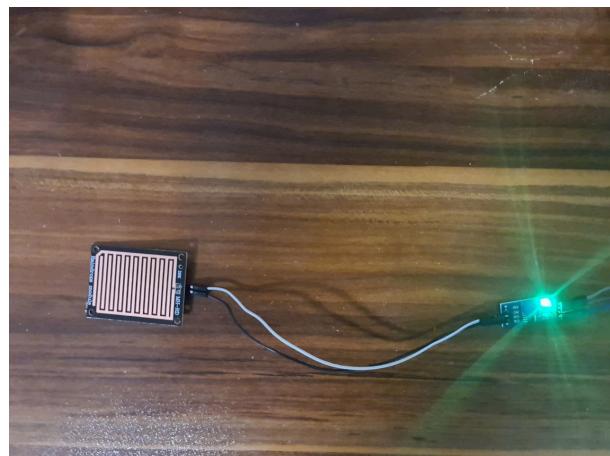
شکل ۷: تابع loop

در ابتدا مقدار ولتاژ سنسور تشخیص باران که به پین A۰ متصل است خوانده می‌شود و به عددی بین ۰ تا ۱۰۲۳ تبدیل می‌شود. در صورتی که این مقدار از یک ترشولیدی کمتر باشد نشان دهنده‌ی وجود باران است و تابع wipe با سرعت مورد نظر صدا زده می‌شود. در غیر این صورت برف پاک کن به زاویه‌ی اولیه منتقل می‌شود. این پروسه در هر سایکل تکرار می‌شود.

فصل ۳

وصل کردن قطعات

در این قسمت نحوه اتصال قطعات در عکس‌ها نشان داده شده است که مشابه توضیحات داده شده در فصل‌های یک و دو است.



شکل ۸: اتصال سنسور

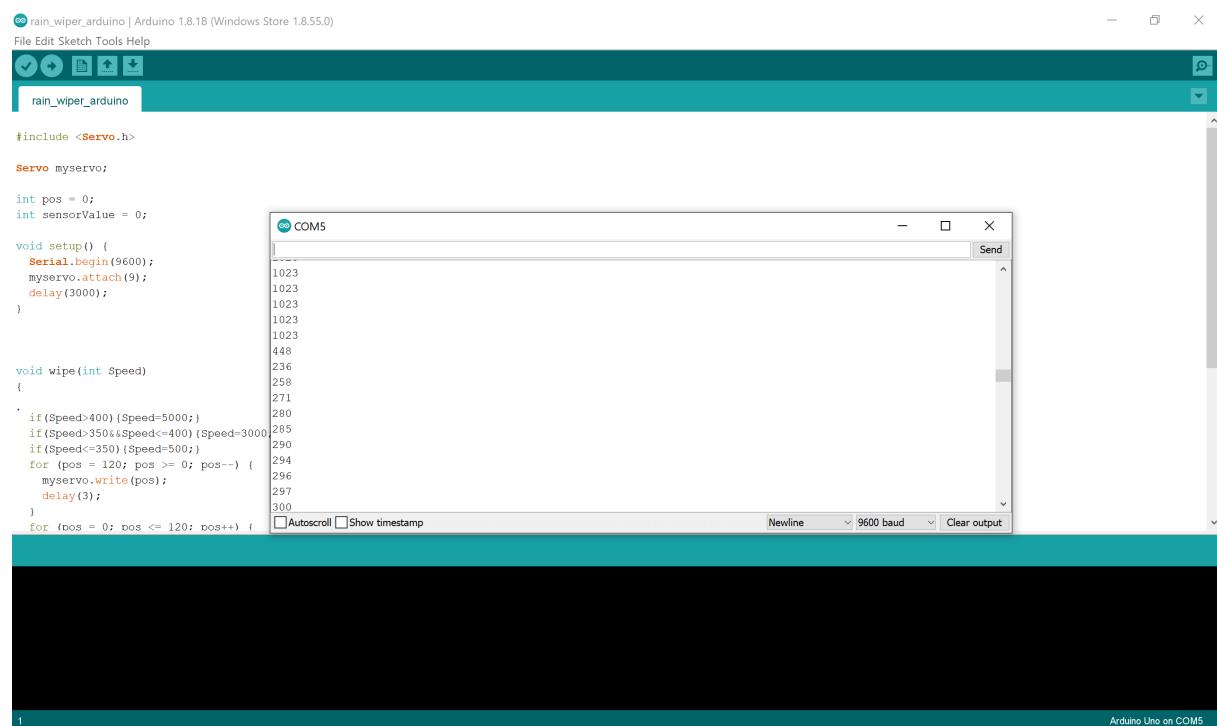


شکل ۹: اتصال سرو

فصل ۴

آزمایش عملکرد

پس از متصل کردن قطعات کد را بر روی آردوینو آپلود کرده و آن را تست می کنیم. میزان آب را با آب پاش در سه حالت کم و متوسط و زیاد تنظیم کردیم. سرعت برف پاک کن مناسب با میزان آب تغییر میکرد. در اشکال زیر می توانید آب روی سنسور و مقدار گذارش شده توسط سنسور را مشاهده کنید.



شکل ۱۰: تغییر از حالت خشک به حالتی که میزان آب زیاد است.

```

rain_wiper_arduino | Arduino 1.8.18 (Windows Store 1.8.55.0)
File Edit Sketch Tools Help
rain_wiper_arduino

#include <Servo.h>

Servo myservo;

int pos = 0;
int sensorValue = 0;

void setup() {
  Serial.begin(9600);
  myservo.attach(9);
  delay(3000);
}

void wipe(int Speed)
{
  if(Speed>400) {Speed=5000;}
  if(Speed>350&&Speed<=400) {Speed=3000}
  if(Speed<350) {Speed=500;}
  for (pos = 120; pos >= 0; pos--) {
    myservo.write(pos);
    delay(3);
  }
  for (pos = 0; pos <= 120; pos++) {

```

Arduino Uno on COM5

شکل ۱۱: میزان آب متوسط

```

rain_wiper_arduino | Arduino 1.8.18 (Windows Store 1.8.55.0)
File Edit Sketch Tools Help
rain_wiper_arduino

#include <Servo.h>

Servo myservo;

int pos = 0;
int sensorValue = 0;

void setup() {
  Serial.begin(9600);
  myservo.attach(9);
  delay(3000);
}

void wipe(int Speed)
{
  if(Speed>400) {Speed=5000;}
  if(Speed>350&&Speed<=400) {Speed=3000}
  if(Speed<350) {Speed=500;}
  for (pos = 120; pos >= 0; pos--) {
    myservo.write(pos);
    delay(3);
  }
  for (pos = 0; pos <= 120; pos++) {

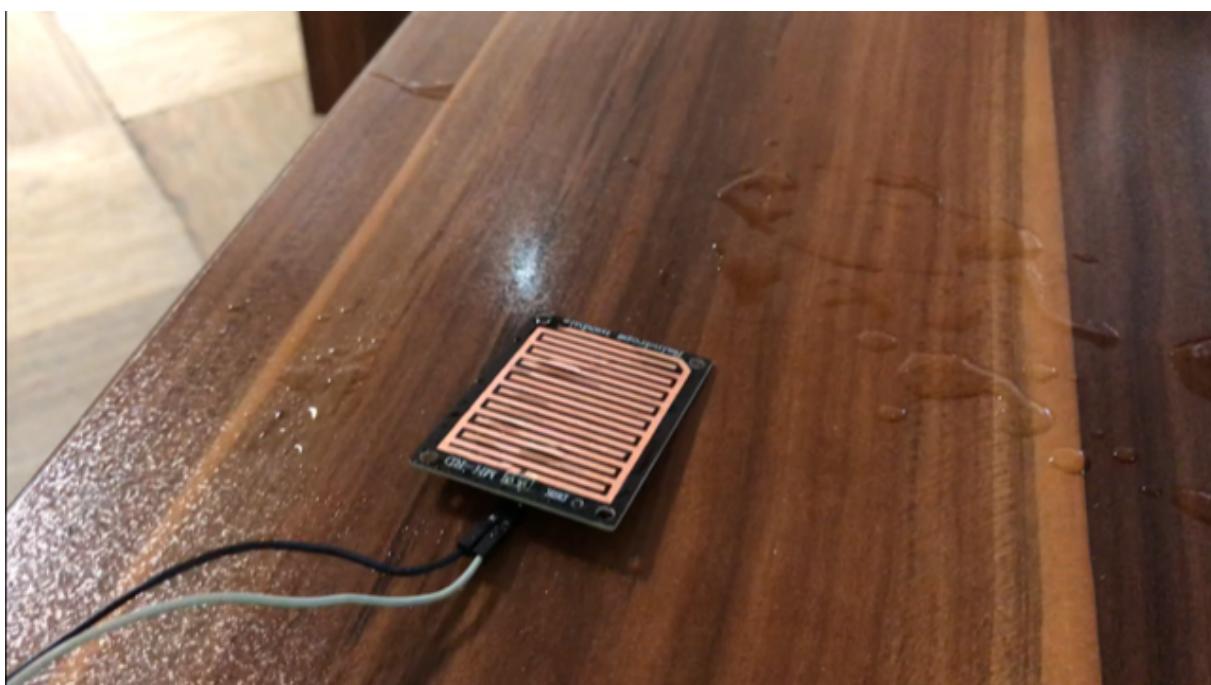
```

Arduino Uno on COM5

شکل ۱۲: تغییر از میزان آب کم به زیاد.



شکل ۱۳: آب کم بر روی سنسور



شکل ۱۴: آب متوسط بر روی سنسور



شکل ۱۵: آب زیاد بر روی سنسور