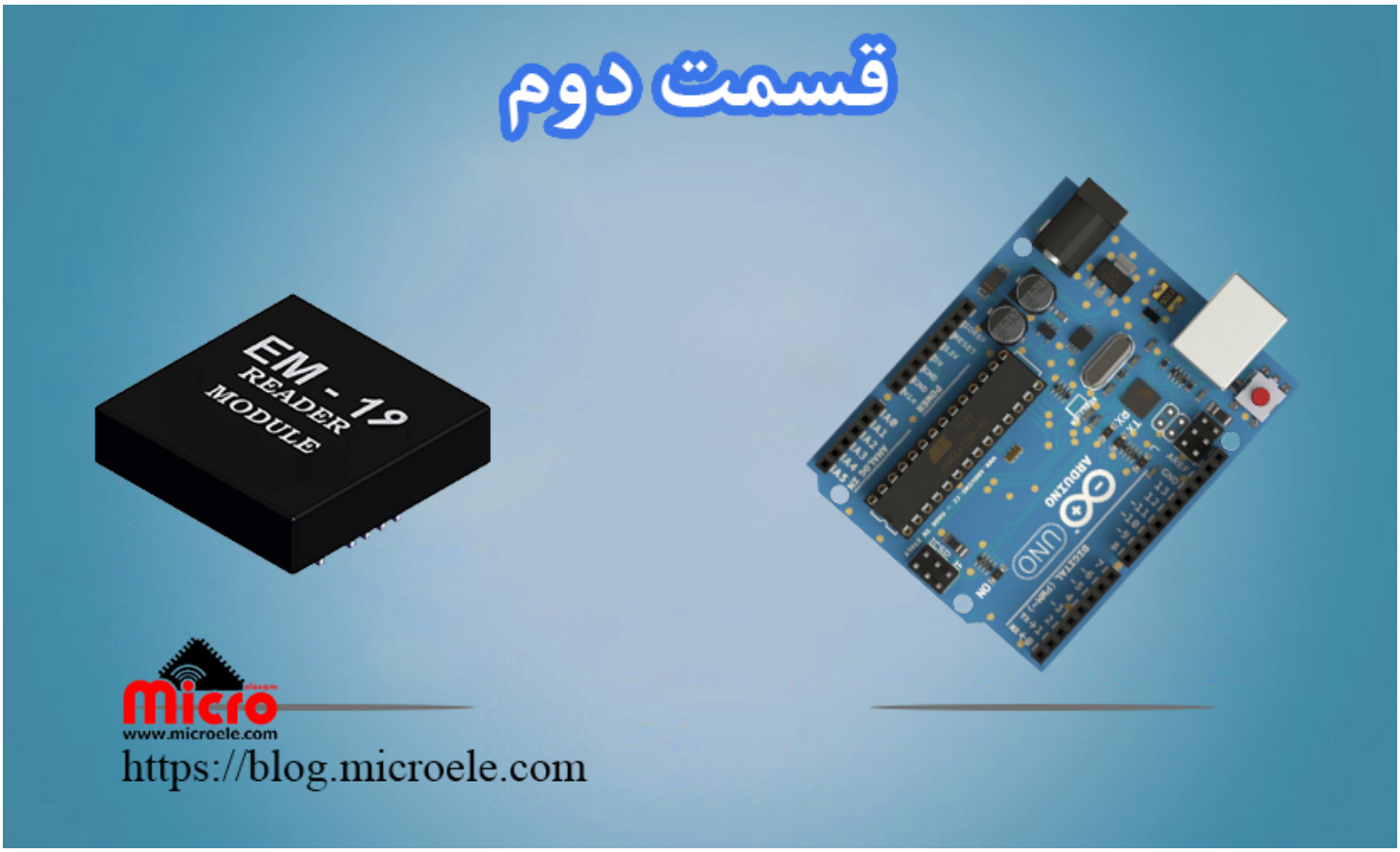
سیستم کنترل تردد مبتنی بر RFID و ماژول EM-19 - قسمت دوم



تاریخ انتشار۴ اسفند, ۱۳۹۹ توسط محمد جواد رشیدیانفر

سلام به همه شما مایکروالکامی های عزیز. امروز قراره با هم دیگه بخش دوم آموزش رو شروع کنیم. پیشنهاد میکنم اگر قسمت اول این آموزش رو مطالعه نکردید حتما اون رو مطالعه کنید. چون در قسمت اول کلیات کار و توضحیات کامل رو قرار دادم. پس قسمت اول آموزش ساخت سیستم کنترل تردد مبتنی بر RFID و ماژول EM-19 رو مطالعه کنید و اگر سوال و یا موردی بود اون رو از طریق دیدگاه در همون بخش، قرار بدید.

در قسمت اول آموزش ماژول EM-19 رو معرفی و مورد بررسی قرار دادیم و اون رو راه اندازی کردیم. در بخش بعدی

اطلاعات تگ و کار های RFID خودمون رو بدست آوردیم و اون ها روی روی سریال مانیتور آردوینو نمایش دادیم. در این قسمت قصد داریم اطلاعات خوانده شده رو روی حافظه میکروکنترلر ذخیره کنیم. با این کار قصد داریم با استفاده از کارت های تعریف شده در حافظه یک کار خاص مثلا فعال شدن قفل برقی را انجام بدیم. به این صورت که اگر کارت RFID تعریف شده در برنامه، شناسایی شد یک پایه خروجی از آردوینو فعال بشه که این پایه رو میتونیم به سایر وسایل خروجی با یک سری تمهیدات لازم متصل کنیم. و در صورتی که کارت تعریف نشده نبود بازر فعال بشه و شروع به آلارم زدن کند.

شما میتونید با یکم خلاقیت به جای اتصال بازر به مدار خودتون و صدای بوق بازر، از آموزش آژیر پلیس که توسط حسین سلطانی آماده شده و روی سایت قرار گرفته استفاده کنید. شما میتونید با استفاده از این آموزش آلارم صوتی خودتون رو پیشرفته تر اش کنید. حتی شما میتونید قبل از اتصال خروجی به بلندگو از یک آمپلی فایر استفاده کنید یا خودتون برای اون طراحی کنید. که پیشنهاد میکنم از آمپلی فایر های موجود استفاده کنید. برای خرید آمپلی فایر هم میتونید به فروشگاه مایکروالکام مراجعه کنید. بزودی برای این مورد هم در مطالب بعدی یک آموزشی تهیه خواهم که راحت تر بتونید از اون استفاده کینند.

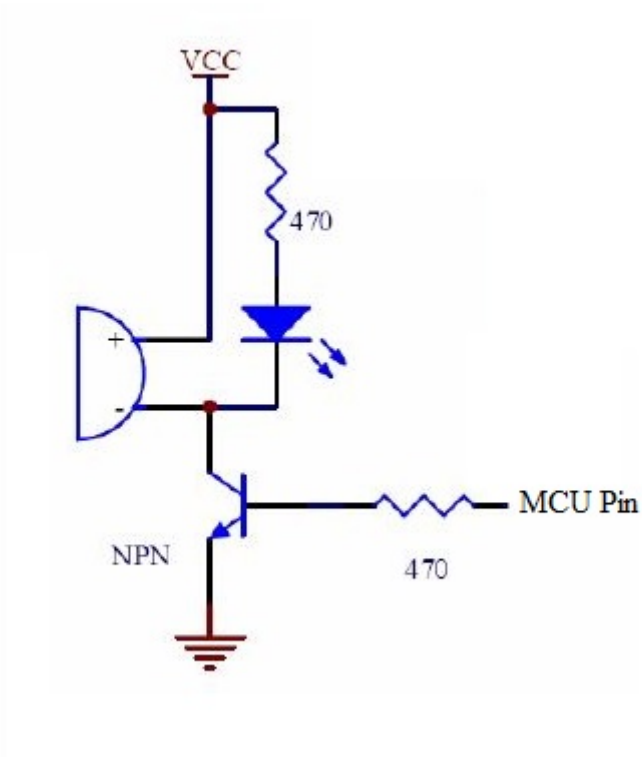
خب بریم سراغ آموزش قسمت دوم، در بخش اول این آموزش ابتدا شماتیک و قطعات مورد نیاز (در قسمت اول بهشون اشاره شده) رو بررسی خواهیم کرد. و در بخش بعدی برنامه نویسی و در انتها نتیجه گیری و یک دمو از نحوه کارکرد عملی این قسمت رو با هم خواهیم دید.

قطعات مورد نیاز

- آردوینو
- برد بورد
- ماژول EM-19
- تگ و یا کارت RFID فرکانس 125KHz
- Buzzer
- LED
- سیم

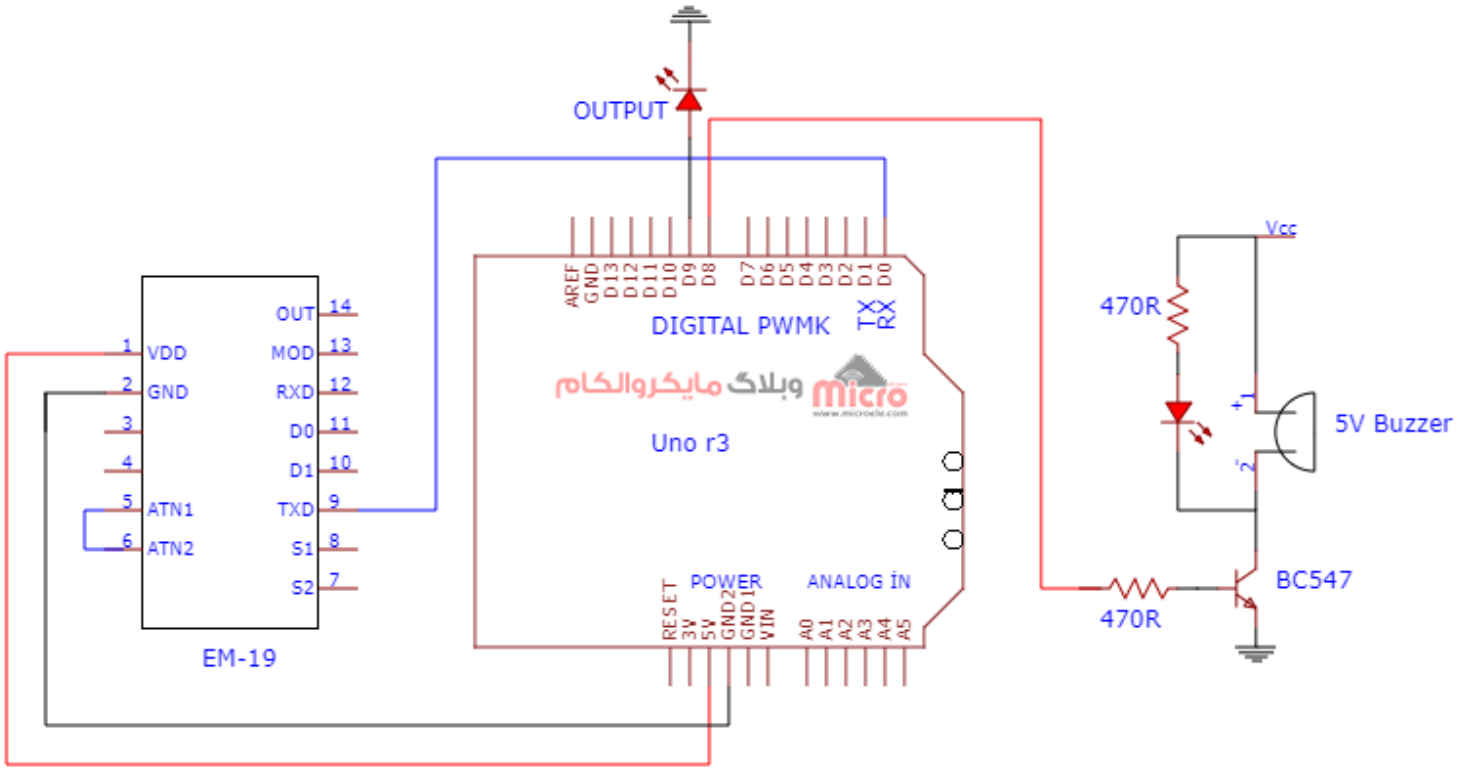
شماتیک

برای راه اندازی بازر، اصولی نیست که اون رو مستقیم به پایه میکروکنترلر یا آردوینو وصل کنیم بخاطر مسائل جریان کشی و محدود بودن جریان پین های میکروکنترلر و آردوینو. پس برای اینکه به میکرو آسیب نرسه و اون رو اصولی درایو کنیم از مدار زیر استفاده میکنیم.



مدار مناسب برای راه اندازی و درایو کردن بازر

برای ترانزیستور مورد استفاده در مدار بالا میتونید از مدل BC547 یا 2N2222 استفاده کنید. همینطور اگر LED رو قرار ندادید هم مشکلی نیست. طبق توضیحات ارائه شده در قسمت قبلی آموزش، برای استفاده از ماژول در حالت آنتن داخلی باید پایه های 5 و 6 ماژول EM19 رو بهم وصل کنید. در تصویر زیر مدار کامل را مشاهده میکنید:



شماتیک

همینطور که در تصویر بالا هم قابل مشاهده هست، از پایه دیجیتال شماره 8 آردوینو برای اتصال به بازر استفاده شده است. و از پایه دیجیتال شماره 9 بعنوان نمایش فعال شدن خروجی استفاده شده و به اون یک LED سبز رنگ وصل شده. که میتونیم مثلا مدار قفل برقی رو بهش وصل کنیم.

برنامه نویسی

در این قسمت به تشریح گام به گام و نحوه برنامه نویسی در محیط برنامه نویسی ARDUINO IDE خواهیم پرداخت.

در ابتدا کد کارت های RFID که میخوایم مجاز باشند برای فعال کردن خروجی رو در یک متغییر از نوع const (یک مقدار ثابت) و بصورت String ذخیره خواهیم کرد. برای مطلع شدن از نحوه خواندن و دریافت کد کارت های RFID حتما قسمت اول این آموزش رو مطالعه کنید.

```
const String user1 = "00003B5D0C6A";
const String user2 = "00003FE1FA24";
const String user3 = "2C00FAEE9EA6";
```

در تابهSetup() تنظیمات اولیه را اعم از تنظیم باودریت و تعریف خروجی ها انجام دادیم. و در حلقه اصلی برنامه یا loop() شروع به کد نویسی اصلی برنامه خواهیم کرد.

از تکه کد زیر برای بررسی کد خوانده شده توسط ماژول RFID مدل EM-19 استفاده میکنیم. نحوه عملکرد این تکه کد به این صورت هست که دیتای دریافتی را با کارت های تعریف شده در ابتدای برنامه تطبیق میده و اگر مقدار دریافتی با مقدار تعریف شده در متغیر در ابتدای برنامه برابر بود، خروجی رو فعال خواهد کرد.

در ساب روتین دستور شرطی، یک حلقه تکرار for تعریف شده که با استفاده از اون بازر رو فعال میکنیم. به این صورت که به تعداد 3 مرتبه به مقدار 150 میلی ثانیه وصل و 170 ثانیه غیر فعال خواهد بود.

```
if (input_data == user1 || input_data == user2 || input_data == user3)
    {
        digitalWrite(9, HIGH);
        Serial.println("Lock is OPEN");
        for (int i = 0; i < 3; i++)
        {
            digitalWrite(8, HIGH);
            delay(150);
            digitalWrite(8, LOW);
            delay(170);
        }
        digitalWrite(9, LOW);
    }
```

در صورتی که دیتای دریافتی کارتی که توسط ماژول EM-19 خوانده شده با مقادیر تعریف شده برابر نباشد، از تکه کد زیر بلافاصله بعد از دستور if در مرحله قبلی استفاده خواهیم کرد. داخل ساب روتین این دستور نیز ار حلقه for استفاده شده که عملکردش دقیقا مشابه مرحله قبلی هست که توضیح دادم. با این تفاوت که تعداد تکرار این حلقه 5

مرتبه خواهد بود و بازر به مدت 250 میلی ثانیه وصل و 250 میلی ثانیه قطع خواهد بود.

```
else
    {
        for (int i = 0; i < 5; i++)
        {
            digitalWrite(8, HIGH);
            delay(250);
            digitalWrite(8, LOW);
            delay(250);
        }
    }
```

سورس کد کامل:

```
//Directed By Microelecom. Weblog Team.
String input_data = "";
تعریف کارت های مجاز در برنامه//
const String user1 = "00003B5D0C6A";
const String user2 = "00003FE1FA24";
const String user3 = "2C00FAEE9EA6";

void setup()
{
    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    Serial.begin(9600);
    Serial.println("RFID Reader...");
    delay(500);
```

```
    Serial.println("Please Insert Your Card...");
}

void loop()
{
  if (Serial.available())
  {
    input_data = Serial.readString();
    if (input_data == user1 || input_data == user2 || input_data == user3)
    {
        digitalWrite(9, HIGH);
        Serial.println("Lock is OPEN");
        for (int i = 0; i < 3; i++)
        {
            digitalWrite(8, HIGH);
            delay(150);
            digitalWrite(8, LOW);
            delay(170);
        }
        digitalWrite(9, LOW);
    }
    else
    {
        for (int i = 0; i < 5; i++)
        {
            digitalWrite(8, HIGH);
            delay(250);
            digitalWrite(8, LOW);
            delay(250);
        }
    }
  }
```

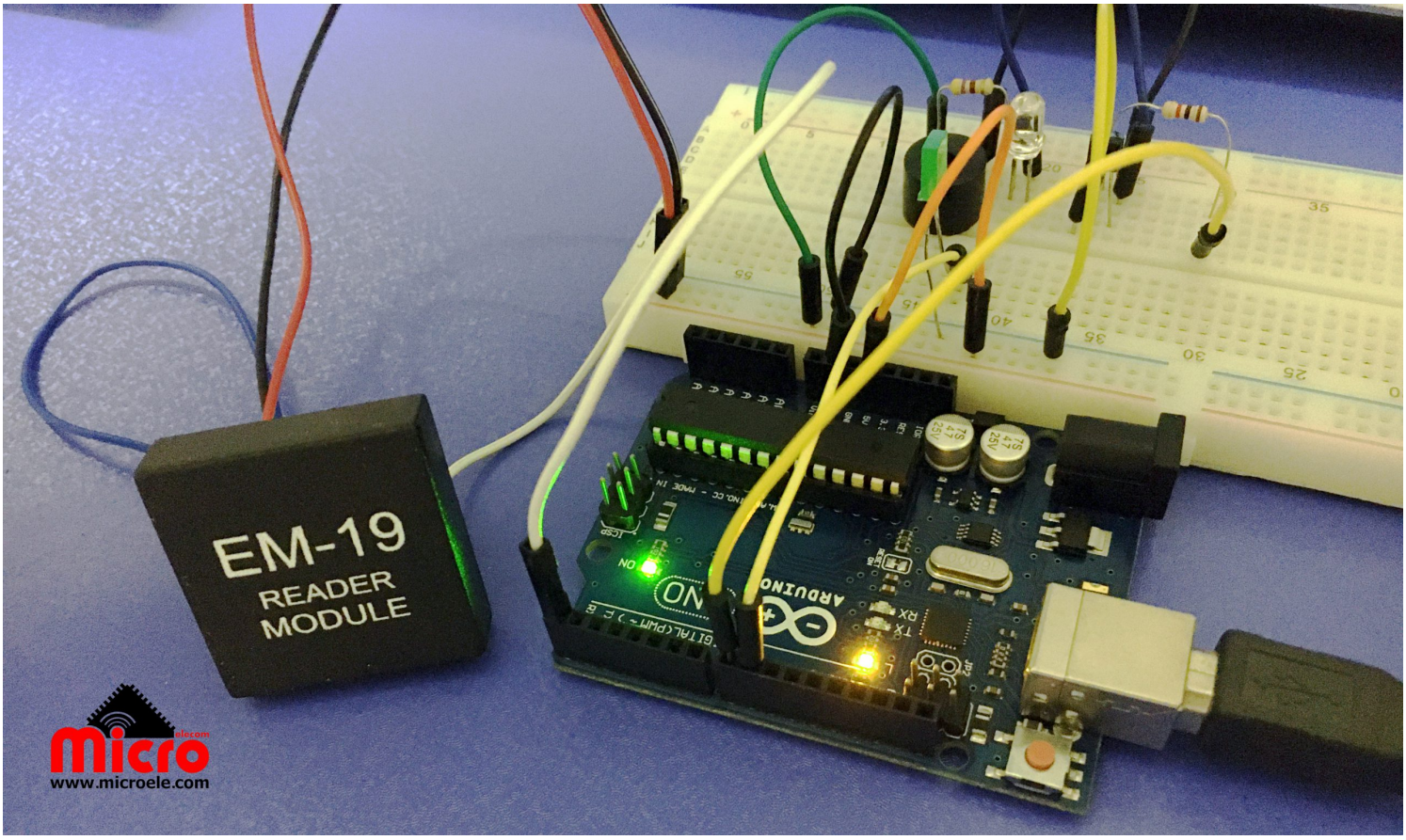
```
}
```

برای دریافت کد کارت ها از تابع Serial.readString() استفاده شده است. چون با استفاده از این تابع دیتا رو بصورت رشته (String) دریافت میکنیم، سرعت عملکرد این تابع مقداری کند هست. که با روش های جایگزین میشه سرعت رو بهبود بخشید و حتی با استفاده از همین تابع سریع دیتا رو دریافت کرد.

برای دانلود سورس کد برنامه میتونید از این لینک اقدام کنید.

نتیجه گیری

در این مطلب با هم دیگه نحوه برنامه نویسی و شرط گذاری برای کارت های خوانده شده رو مورد بررسی قرار دادیم. و نتیجه نهایی رو بصورت یک فیلم در ادامه مطلب میتونید مشاهده کنید.



پروژه ماژول *RFID EM-19 EM19* با آردوینو

https://blog.microele.com/wp-content/uploads/2021/02/EM-19-Part-2-Blog-Video.mp4

امیدوارم از این آموزش کمال بهره را برده باشید. در صورتی که هرگونه نظر یا سوال داشتید درباره این آموزش لطفا اون رو در انتهای همین صفحه در قسمت دیدگاه ها قرار بدید. در کوتاه ترین زمان ممکن به اون ها پاسخ خواهم داد. اگر این مطلب براتون مفید بود، اون رو حتما با دوستانتون به اشتراک بگذارید. همینطور میتونید اون رو پس از اجرای عملی توی اینستاگرام با هشتگ #microelecom به اشتراک بگذارید و پیج مایکروالکام (@microelecom) رو هم منشن کنید.