راه اندازی ماژول SIM800 قسمت 14 – ارسال پیامک با آردوینو

تاریخ انتشار۹ آبان, ۱۴۰۰ توسط محمد جواد رشیدیانفر

سلام خدمت همه شما مایکروالکامی ها. در مطالب قبلی از سری آموزش های ماژول SIM800 به بررسی و نحوه ارسال و دریافت پیامک با دستورات ماژول SIM800 پرداخته شد. در این قسمت به نحوه استفاده از آنها و با استفاده از برنامه نویسی و ارسال و دریافت پیامک (SMS) با آردوینو پرداخته خواهد شد. پس با من تا انتهای مطلب همراه باشید. همچنین شما میتوینند سایر مطالب من رو از این لینک مطالعه و بررسی کنید.

بخش پیامک ماژول SIM800

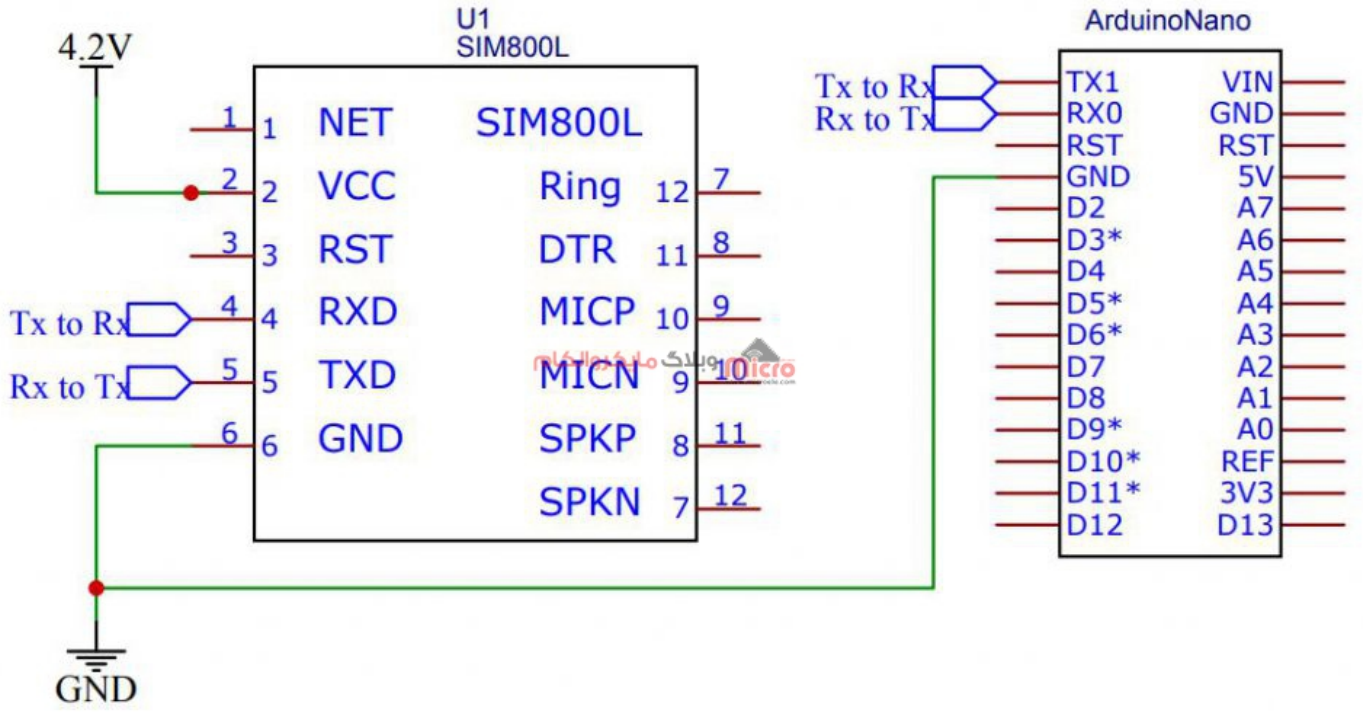
در مطلب قبلی بطور مفصل در خصوص دستورات مورد نیاز (AT Command) ها برای ارسال و دریافت پیامک بحث شد. پس حتما این مطلب را مطالعه بفرمایید. در این مطلب حالت پیامک را در مد کاری Text تنظیم کرده و در اولین اجرا پس از روشن شدن آردوینو، یک پیامک با متن دلخوه به شماره مد نظر ارسال خواهد شد.

قطعات مورد نیاز

- برد راه انداز ماژول SIM800L
- LM2596 Buck DC-DC Convertor
- آداپتور 12V
- آنتن GSM با سوکت IPEX
- مبدل سریال

شماتیک

در ابتدا خروجی ماژول کاهنده ولتاژ را روی 4.2V تنظیم نموده و طبق شماتیک زیر اتصالات مدار را برقرار نمایید. پایه VCC ماژول SIM800 را به مثبت ماژول کاهنده ولتاژ متصل کنید.



شماتیک

سورس کد ارسال و دریافت پیامک با آردوینو

در سورس کد چند تابع وجود دارد که جهت خوانا بودن کد ها و قابلیت توسعه آن در نظر گرفته شده است. در ادامه به بررسی هر بخش از کد ها پرداخته خواهد شد. با توجه به نیاز فعلی این مطلب، نیاز به نوشتن کدی در حلقه لوپ برنامه نمی‌باشد. بعد از روشن کردن آردوینو در بدو راه اندازی برد، در صورت صحت عملکرد و فراهم بودن شرایط، ماژول یک پیامک به شماره تعریف شده در برنامه ارسال خواد کرد.

تابع setup

در این تابع باودریت ارتباط سریال سخت افزای تنظیم و توابع مورد نیاز فراخوانی شده است. نهایتا در همین تابع بعد از روشن شدن برد و طی کردن روال دستورات، SMS مورد نظر ارسال خواهد شد.

```
void setup()
{
  Serial.begin(9600);
  GSM_Init();
  SMS_Init();
  Send_data(messages);
  Send_SMS();
}
```

تابع GSM_Init

در این تابع ابتدا AT Command ها مورد نیاز جهت تنظیم ماژول و صحت از نحوه اتصال و همچنین آماده بودن ماژول، ارسال خواهد شد. مادامی که پاسخ مد نظر از طرف ماژول دریافت نشود کامند متناوبا ارسال خواهد شد. البته شما می‌توانید با تغییر کوچکی در برنامه به تعداد دفعات مشخصی کامند را ارسال نمایید.

```
void GSM_Init()
{
  Serial.println("Finding SIM800 GSM...");
  // Check connection
  boolean at_flag = 1;
  while (at_flag)
  {
    Serial.println("AT");
    delay(5);
    while (Serial.available() > 0)
    {
      if (Serial.find("OK"))
      {
        at_flag = 0;
        Serial.println("UART connection: True");
      }
```

```
  }
  delay(1000);
}

// ECO Disabling Mode
at_flag = 1;
while (at_flag)
{
  Serial.println("ATE0");
  delay(5);
  while (Serial.available() > 0)
  {
    if (Serial.find("OK"))
    {
      at_flag = 0;
      Serial.println("ECO Disabling: Successful");
    }
  }
  delay(1000);
}

// CPIN?
at_flag = 1;
while (at_flag)
{
  Serial.println("AT+CPIN?");
  delay(5);
  while (Serial.available() > 0)
  {
    if (Serial.find("+CPIN: READY"))
    {
      at_flag = 0;
```

```
        Serial.println("Network Connection: Successful");
      }
    }
    delay(1000);
  }
```

تابع SMS_Init

در این تابع شماره موبایل مقصد را وارد می‌کنیم. برای این کار 2 حالت وجود داد. یا مستقیم شماره را وارد کنیم (حالت اول) یا شماره را از داخل یک رشته بخوانیم (حالت دوم). هر دو حالت در کد زیر قابل مشاهده است.

```
void SMS_Init()
{
  Serial.println("AT+CMGF=1");
  delay(400);
  Serial.println("AT+CMGS=\"" + number + "\"");  //حالت دوم
  // Serial.println("AT+CMGS=\"+989xxxxxxxxx\"");  // حالت اول
  delay(400);
}
```

تابع Send_data

با استفاده از این تابع، متن پیامک مورد نظر خود را بصورت **انگلیسی** به ورودی تابع داده و نسبت به ارسال آن اقدام خواهیم کرد.

```
void Send_data(String messages)
{
  Serial.print(messages);
  delay(200);
}
```

تابع Send_SMS

در هنگام ارسال پیامک در آخرین مرحله بعد از انجام کلیه موارد، نیاز به ارسال ctrl+z می‌باشد. برای انجام این مورد از این تابع استفاده شده که کد هگز این دستور (26) را در آخرین مرحله ارسال خواهد کرد. در صورتی که همه مراحل صحیح طی شود و مشکلی وجود نداشته باشد پس از چند ثانیه بعد از ارسال این دستور، پیامک به مقصد خواهد رسید.

```
void Send_SMS()
{
  Serial.write(26);
}
```

کد های کامل ارسال پیامک از طریق SIM800

```
String messages = "https://blog.microele.com";
String number = "+989xxxxxxxxx";

void GSM_Init()
{
  Serial.println("Finding SIM800 GSM...");
  // Check connection
  boolean at_flag = 1;
  while (at_flag)
  {
    Serial.println("AT");
    delay(5);
    while (Serial.available() > 0)
    {
      if (Serial.find("OK"))
      {
        at_flag = 0;
```

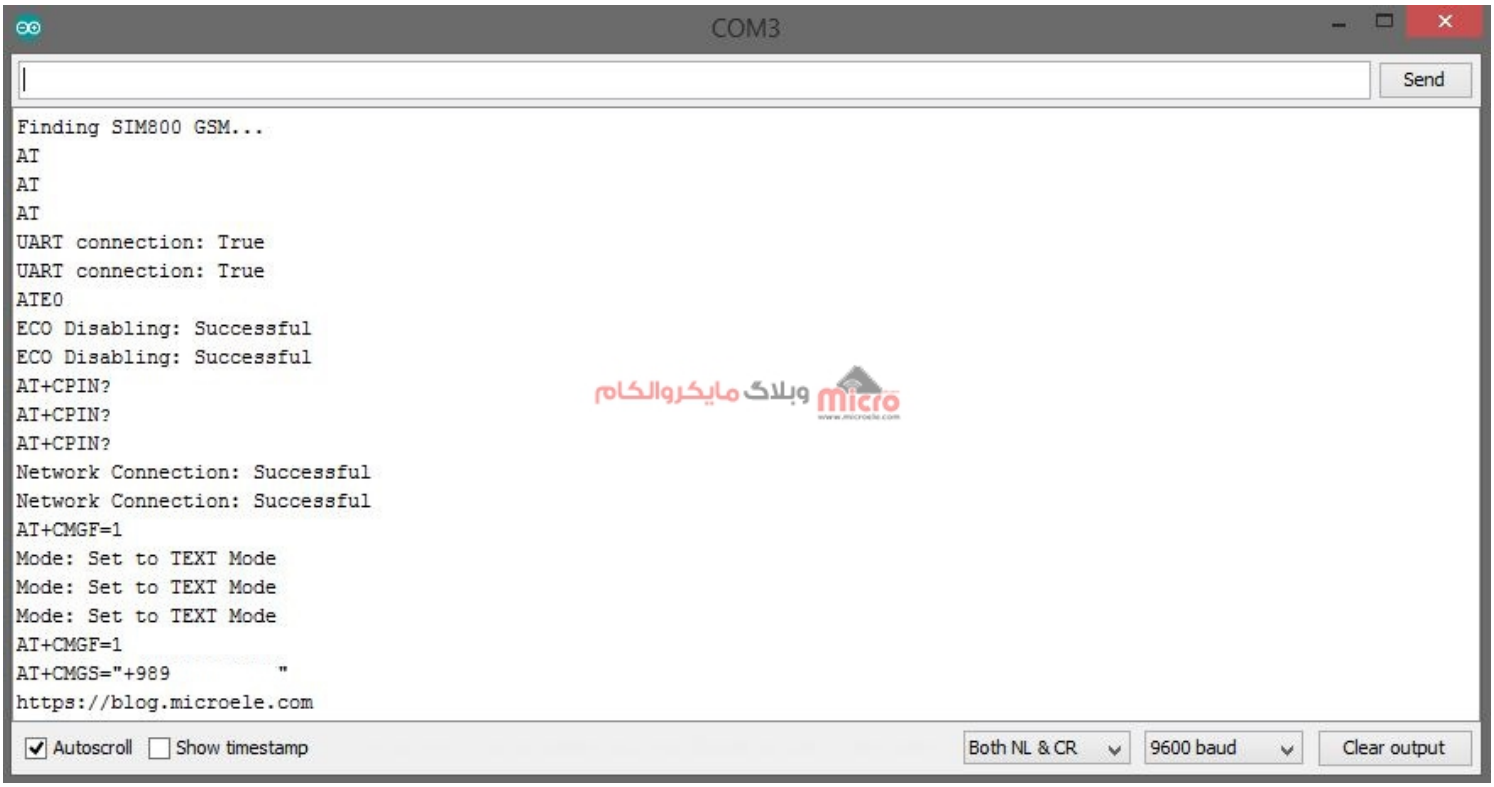
```
      Serial.println("UART connection: True");
    }
  }
  delay(1000);
}

// ECO Disabling Mode
at_flag = 1;
while (at_flag)
{
  Serial.println("ATE0");
  delay(5);
  while (Serial.available() > 0)
  {
    if (Serial.find("OK"))
    {
      at_flag = 0;
      Serial.println("ECO Disabling: Successful");
    }
  }
  delay(1000);
}

// CPIN?
at_flag = 1;
while (at_flag)
{
  Serial.println("AT+CPIN?");
  delay(5);
  while (Serial.available() > 0)
  {
    if (Serial.find("+CPIN: READY"))
```

```
      {
        at_flag = 0;
        Serial.println("Network Connection: Successful");
      }
    }
    delay(1000);
  }

  // Text Mode
  at_flag = 1;
  while (at_flag)
  {
    Serial.println("AT+CMGF=1");
    delay(5);
    while (Serial.available() > 0)
    {
      if (Serial.find("OK"))
      {
        at_flag = 0;
        Serial.println("Mode: Set to TEXT Mode");
      }
    }
    delay(1000);
  }
}

void SMS_Init()
{
  Serial.println("AT+CMGF=1");
  delay(400);
  Serial.println("AT+CMGS=\"" + number + "\"");
  // Serial.println("AT+CMGS=\"+989xxxxxxxxx\"");
```

```
  delay(400);
}

void Send_data(String messages)
{
  Serial.print(messages);
  delay(200);
}

void Send_SMS()
{
  Serial.write(26);
}

void setup()
{
  Serial.begin(9600);
  GSM_Init();
  SMS_Init();
  Send_data(messages);
  Send_SMS();
}

void loop()
{

}
```

مشاهده نتیجه و دریافت پیامک در تلفن همراه

در حین اجرای برنامه با باز کردن سریال مانیتور و تنظیم باودریت روی 9600 (طبق تصویر زیر) فرایند اجرای دستورات قابل مشاهده است.



بررسی فرایند اجرای برنامه در سریال مانیتور

پیامک دریافتی از سمت ماژول

نتیجه گیری

در این مطلب نحوه ارسال پیامک و کامند های مربوطه مورد بررسی قرار گرفت. پیش از ارسال پیامک یا اقدام برای آن، باید از صحت عملکرد ماژول و متصل بودن به شبکه اطمینان حاصل نمود. با استفاده از کامند های مربوط این امکان فراهم شده است.

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 به اشتراک بگذارید. همینطور میتونید این آموزش را پس از اجرای عملی توی اینستاگرام با هشتگ microelecom# به اشتراک بگذارید و پیج مایکروالکام (microelecom@) رو هم منشن کنید.