A logo with orange and red letters

Description automatically generated with medium confidence

**Chất lượng Xây dựng Thương hiệu**

**Gửi Nhận SMS & Gọi với Module GSM SIM800L & Arduino UNO R3**

Mô-đun SIM800L GSM/GPRS là một modem GSM thu nhỏ có thể được sử dụng trong nhiều dự án IoT. Bạn có thể sử dụng mô-đun này để thực hiện hầu như mọi thứ mà một chiếc điện thoại di động thông thường có thể làm, chẳng hạn như gửi tin nhắn SMS, thực hiện cuộc gọi điện thoại, kết nối Internet qua GPRS và nhiều hơn nữa.

Trên hết, mô-đun này hỗ trợ mạng GSM/GPRS bốn băng tần, nghĩa là nó có thể hoạt động ở hầu hết mọi nơi trên thế giới.

**Tổng quan về phần cứng**

* Phần trung tâm của mô-đun này là chip di động GSM SIM800L của Simcom.
* Điện áp hoạt động của chip nằm trong khoảng từ 3,4V đến 4,4V
* Giao tiếp với bộ vi điều khiển qua UART
* Mô-đun hỗ trợ tốc độ truyền dữ liệu từ 1200 bps đến 115200 bps và có tính năng tự động phát hiện tốc độ truyền dữ liệu.
* Mô-đun này cần một ăng-ten ngoài để kết nối với mạng
* Bo mạch cũng có đầu nối U.FL nếu bạn muốn giữ ăng-ten ở khoảng cách xa bo mạch.

**Đặc trưng**

* Hỗ trợ Quad-band: GSM850, EGSM900, DCS1800 và PCS1900
* Kết nối với bất kỳ mạng GSM toàn cầu nào với bất kỳ SIM 2G nào
* Thực hiện và nhận cuộc gọi thoại bằng loa ngoài 8Ω và micrô electret
* Gửi và nhận tin nhắn SMS
* Gửi và nhận dữ liệu GPRS (TCP/IP, HTTP, v.v.)
* Quét và nhận chương trình phát thanh FM

**SIM800L GSM/GPRS Module Pinout**

A close-up of a computer chip

Description automatically generated

NET chân cắm là nơi bạn có thể hàn ăng-ten xoắn ốc đi kèm với mô-đun.

VCC là chân nguồn. Hãy nhớ rằng chip SIM800L có dải điện áp hoạt động từ 3,4 V đến 4,4 V, do đó, việc kết nối mô-đun này với đầu ra 5V của Arduino có thể gây hư hỏng cho mô-đun. Nó thậm chí không chạy ở mức 3,3 V! Thay vào đó, hãy sử dụng pin Li-Po hoặc bộ chuyển đổi buck DC-DC.

RST (Reset) là chân reset. Nếu bạn đặt mô-đun vào một vị trí hoàn toàn không tốt, hãy kéo chân này xuống mức THẤP trong 100ms để thực hiện reset cứng.

RxD (Receiver) Pin được sử dụng để gửi lệnh đến mô-đun. Ghim này là tốc độ truyền tự động nên tốc độ truyền tốc độ mà bạn gửi lệnh “AT” sau khi thiết lập lại là tốc độ truyền tốc độ được sử dụng.

TxD (Transmitter) chân truyền dữ liệu từ mô-đun đến vi điều khiển.

GND là chân nối đất.

RING pin là Ring Indicator. Về cơ bản đây là pin ‘interrupt-out’ từ mô-đun. Theo mặc định, nó ở mức CAO và có thể được cấu hình để ở mức THẤP khi nhận được cuộc gọi hoặc tin nhắn SMS.

DTR Pin điều khiển chế độ ngủ. Kéo nó CAO sẽ khiến mô-đun vào chế độ ngủ, vô hiệu hóa giao tiếp nối tiếp. Kéo nó THẤP trong khoảng 50 ms sẽ đánh thức mô-đun.

MIC± là đầu vào micrô vi sai. Bạn có thể kết nối micrô electret ngoài trực tiếp vào hai chân này.

SPK± là giao diện loa khác biệt. Bạn có thể kết nối loa trực tiếp với hai chân này.

**Sơ đồ mạch của mô-đun SIM800L GSM/GPRS**

Sơ đồ mạch của SIM800L sử dụng IC SIM800L xử lý tất cả phần cứng phức tạp và bạn chỉ cần kết nối một vài dây để bắt đầu chạy. Sơ đồ mạch này sẽ rất hữu ích cho triển khai mạch này trong dự án PCB của mình.

A diagram of a cell phone

Description automatically generated

Gía đỡ thẻ sim được kết nối trực tiếp với mô-đun SIM800L; nguồn điện cho thẻ sim chỉ được điều chỉnh bởi mô-đun SIM800. Và bạn cũng có thể thấy có một tụ điện tantalum 100uF lớn được kết nối với mô-đun, điều này là cần thiết vì mô-đun đột nhiên rút ra các xung dòng điện mà yêu cầu nguồn điện đột ngột được đáp ứng bởi tụ điện tách. Ngoài ra, chúng ta có một tụ điện tantalum và một tụ điện tách cùng với đèn LED chỉ báo.

Thiết lập các kết nối cơ bản, chúng ta sẽ giao tiếp với module SIM800L bằng cách gửi các lệnh AT.

**AT** – Đây là lệnh AT cơ bản nhất. Nó cũng khởi chạy Auto-baud Waster. Nếu nó hoạt động, bạn sẽ thấy ký tự AT lặp lại và sau đó là OK, nó muốn nói với bạn rằng nó OK và nó hiểu chính xác những gì bạn đã gửi! Sau đó, bạn có thể gửi một số lệnh để truy vấn module và nhận thông tin như sau:

**AT + CSQ** – Kiểm tra cường độ tín hiệu ’- Số đầu tiên là cường độ dB, nó phải cao hơn 5. Cao hơn là tốt hơn. Tất nhiên nó phụ thuộc vào ăng-ten và vị trí của module!

**AT + CCID** – lấy số thẻ SIM – kiểm tra xem thẻ SIM có ổn không và bạn có thể xác minh số của thẻ SIM.

**AT+CREG?:** Kiểm tra xem bạn đã đăng ký vào mạng hay chưa. Số thứ hai trong phản hồi đầu ra phải là 1 hoặc 5. 1 cho biết bạn đã đăng ký vào mạng gia đình và 5 cho biết mạng chuyển vùng. Bất kỳ số nào khác ngoài hai số này cho biết bạn chưa đăng ký vào bất kỳ mạng nào.

**ATI**: Trả về tên mô-đun và bản sửa đổi..

**AT+COPS?:** Kiểm tra xem bạn đang kết nối với mạng nào.

**AT+CBC**: Trả về trạng thái pin Li-Po. Số thứ hai là mức pin (trong trường hợp của chúng tôi là 93%) và số thứ ba là điện áp thực tế tính bằng mV (trong trường hợp của chúng tôi là 3,877 V)

A screenshot of a computer program

Description automatically generated

**Kết nối Mô-đun GSM SIM800L với Arduino**

Bây giờ bạn đã quen với mô-đun SIM800L, đã đến lúc kết nối nó với Arduino!

Để bắt đầu, hãy kết nối ăng-ten với mô-đun và lắp thẻ micro SIM vào ổ cắm.

Tiếp theo, kết nối chân Tx của mô-đun với chân kỹ thuật số số 3 của Arduino, vì chúng ta sẽ giao tiếp với mô-đun thông qua phần mềm UART.

Sơ đồ đầu tiên, một pin Li-Po 1200mAh được kết nối.

A circuit board with wires and wires

Description automatically generated

Sơ đồ thứ hai, bộ chuyển đổi buck DC-DC LM2596 được kết nối.

A circuit board with wires and a qr code

Description automatically generated

**Project mẫu sử dụng với module SIM800L và module relay để điều khiển thiết bị từ xa qua tin nhắn SMS. Khi nhận được tin nhắn với nội dung "ON" hoặc "OFF", vi điều khiển sẽ bật hoặc tắt relay tương ứng.**

**Sơ đồ kết nối**

**A circuit board with wires

Description automatically generated**

**CODE ví dụ:**

<https://github.com/haohao113/SIM800L>

#include <Arduino.h>

#include <SoftwareSerial.h>

const int RX\_PIN = 2;

const int TX\_PIN = 3;

const int RELAY\_PIN = 7;          // Chân điều khiển relay

const String SDT = "+1234567890"; // Số điện thoại được phép điều khiển

SoftwareSerial sim800l(RX\_PIN, TX\_PIN); // SIM800L Tx & Rx được kết nối với Arduino #3 & #2

void setup()

{

  pinMode(RELAY\_PIN, OUTPUT);

  digitalWrite(RELAY\_PIN, LOW); // Tắt relay ban đầu

  Serial.begin(9600);

  sim800l.begin(9600);

  sim800l.println("AT"); // Sau khi kiểm tra kết nối thành công, nó sẽ trả về OK

  delay(1000);

  if (sim800l.available())

  {

    Serial.println(sim800l.readString());

  }

  sim800l.println("AT+CMGF=1"); // Cấu hình chế độ TEXT

  delay(1000);

  if (sim800l.available())

  {

    Serial.println(sim800l.readString());

  }

  sim800l.println("AT+CNMI=1,2,0,0,0"); // Thiết lập để nhận tin nhắn SMS tự động

  delay(1000);

  if (sim800l.available())

  {

    Serial.println(sim800l.readString());

  }

}

void loop()

{

  if (sim800l.available())

  {

    String message = sim800l.readString();

    Serial.println(message);

    if (message.indexOf(SDT) != -1) // Kiểm tra số điện thoại gửi tin nhắn

    {

      if (message.indexOf("ON") != -1) // Kiểm tra nội dung tin nhắn

      {

        digitalWrite(RELAY\_PIN, HIGH); // Bật relay

        Serial.println("Relay ON");

      }

      else if (message.indexOf("OFF") != -1)

      {

        digitalWrite(RELAY\_PIN, LOW); // Tắt relay

        Serial.println("Relay OFF");

      }

    }

    else

    {

      Serial.println("Số điện thoại không được phép");

    }

  }

}