

# Envoyer et recevoir des SMS

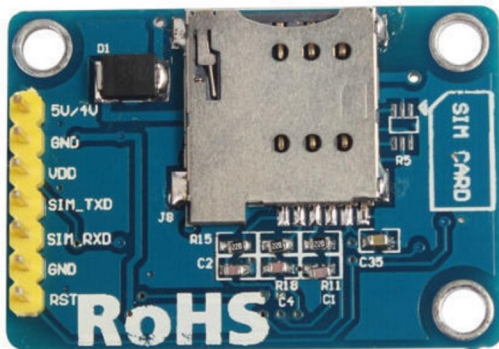
Le module SIM800L permet d'envoyer ou de recevoir des SMS.  
On le trouve sur Ebay pour 8 euros environ.

Il est prêt à l'usage, il suffit d'insérer dans le module une carte nano sim. Les tests ont été réalisés à l'aide d'une carte SIM du célèbre opérateur qui offre des abonnements à 2 € par mois !

La gestion s'effectue par des commandes AT. Il n'a donc pas besoin d'une bibliothèque spéciale, on peut utiliser la bibliothèque standard SoftwareSerial pour commander le module.



## Brochage



7 broches sont disponibles :

**5V**                    **ALIMENTATION EXTERNE**  
**GND**                **ALIMENTATION EXTERNE**

|         |                         |
|---------|-------------------------|
| VDD     | 5V Arduino              |
| SIM_TXD | broche digitale Arduino |
| SIM_RXD | broche digitale Arduino |
| GND     | GND Arduino             |
| RST     | non utilisée            |

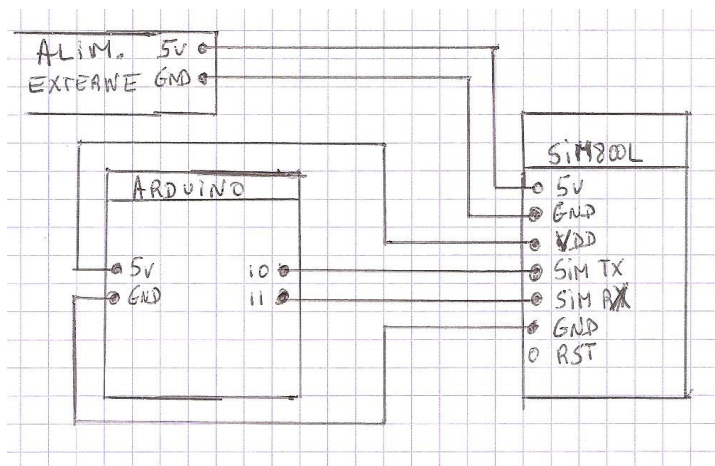
Les deux premières broches 5V et GND doivent être connectées à une source d'alimentation externe pouvant délivrer une intensité de 1 ampère. En effet le module requiert par moment un courant important que ne peut fournir l'Arduino.

Sur certains blogs, on trouve des exemples où ces deux broches ne sont pas utilisées. Je n'ai pas réussi à faire fonctionner le module sans une alimentation externe qui peut être par exemple un module chargeur 5V de téléphone.

## Mise en place de la carte SIM

Sur la figure ci-dessus, orientation la carte Sim est précisée par un schéma. Les contacts doivent être du côté du circuit imprimé.

## Câblage



## Démarrage

Lors de la mise sous tension,  
observer les LEDS

NET (D6) et RING (D2)

La LED NET indique la connexion  
ou non au réseau.

La LED RING indique si le  
composant est sous tension.

Cela commence automatiquement  
par une séquence de tentatives de  
connexion au réseau :

LED RING Allumée et LED NET  
clignote à une fréquence de 1  
seconde. La LED NET clignote de  
cette façon 6 à 7 fois. Si la  
connexion échoue, la séquence  
reprend.



**Si la connexion réussit, la LED NET clignote à une fréquence de 3 secondes et la LED RING reste allumée de façon permanente.**

On peut alors utiliser le programme suivant pour tester quelques commandes AT

## Programme de test du module

```
#include <SoftwareSerial.h>

SoftwareSerial sim800l(10, 11); // RX, TX

void setup()
{
  Serial.begin(9600);
  sim800l.begin(9600);
}

void loop()
{
  if (sim800l.available()) Serial.write(sim800l.read());
  if (Serial.available()) sim800l.write(Serial.read());
}
```

Ce programme permet simplement de dialoguer avec le module via la console Arduino. Exemple :

En rouge les commandes envoyées, en noir les réponses du module, en bleu commentaires

**AT** Vérification si on est connecté au module  
 OK  
**ATI** Statut du modem  
 SIM800 R14.18  
 OK  
**AT+CPIN?** Vérification si la carte SIM est prête  
 +CPIN: READY  
 OK  
**AT+CMGF=1** Gestion des SMS en mode texte (code Ascii de base)  
 OK  
**AT+CREG?** Vérification de la connexion au réseau GSM  
 +CREG: 0,5  
 OK  
**AT+COPS?** Nom du réseau sur lequel on est connecté  
 +COPS: 0,0,"Orange F"  
 OK  
**AT+CMGD=?** Nombre max de SMS et commandes autorisées  
 +CMGD: (1-100),(0-4)  
 OK  
**AT+CMGR=1** Lecture du SMS n° 1  
 +CMGR: "REC READ","+33XXXXXXXXX","", "18/10/06,15:18:04+08"  
 Test  
 OK  
**AT+CMGR=20** Lecture du SMS n° 20 (inexistant)  
 OK  
**AT+CMGR=2** Lecture du SMS n° 2  
 +CMGR: "REC READ","+336XXXXXXXXX","", "18/10/08,12:50:55+08"  
 Hello  
 OK

## Description de quelques commandes

|           |                                     |                                      |
|-----------|-------------------------------------|--------------------------------------|
| ATI       | SIM800 R14.18                       | Statut du modem<br>Statut du réseau. |
|           |                                     | 1 <sup>er</sup> indice : 0 -> GSM    |
| AT+CREG ? | +CREG : 0,5                         | 2 <sup>ème</sup> indice :            |
|           |                                     | 0 -> pas de réseau                   |
|           |                                     | 5 -> réseau ok                       |
| AT+CPIN ? | +CPIN READY                         | Pas de code PIN nécessaire           |
| AT+COPS ? | +COPS : <mode>,<format>,<opérateur> | Affiche l'opérateur sélectionné      |

On trouve facilement sur le web des tutos décrivant les commandes AT pour GSM

# Programme de démo

Ce programme montre notamment comment envoyer un SMS

```
#include <SoftwareSerial.h>

// A MODIFIER

char numero[] = "+336XXXXXXXXX";
char message[] = "XXXXXXXXXXXXXXXXXXXXX";

// FIN DES MODIFS

SoftwareSerial sim800l(10, 11); // RX, TX

void sendSms(char numero[], char message[])
{
    Serial.println("Sending text message...");
    sim800l.print("AT+CMGF=1"); // passe en mode SMS TEXTE
    sim800l.write(0x0D);
    delay(1000);
    sim800l.print("AT+CMGS=\"");
    sim800l.print(numero);
    sim800l.print("\");
    sim800l.write(0x0D);
    while (sim800l.read() != '>'); // Attendre le prompt '>'
    sim800l.print(message);
    delay(500);
    // CTR+Z en langage ASCII, indique la fin du message
    sim800l.write(0x1A);
    sim800l.write(0x0D);
    sim800l.write(0x0A);
    Serial.println("Text send"); // Le message est envoyé.
}

void setup()
{
    Serial.begin(9600);
    sim800l.begin(9600);
}

void traiterCmd(char c)
{
    switch (c)
    {
        case '\r':
        case '\n':
            break;
        case '1':
            sim800l.print("AT\r");
            break;
        case '2':
            sendSms(numero, message);
            break;
        case '3':
            sim800l.print("AT+CMGL=\"ALL\"\r");
            break;
        case '4':
            sim800l.print("AT+CREG?\r");
            break;
        case '5':
            sim800l.print("AT+COPS?\r");
            break;
        case '6':
```

```

        sim8001.print("AT+CMGD=?\r");
        break;
default:
    Serial.println();
    Serial.println("Commandes");
    Serial.println("1 : AT");
    Serial.println("2 : send sms");
    Serial.println("3 : affiche tous les messages recus");
    Serial.println("4 : etat du reseau");
    Serial.println("5 : operateur");
    Serial.println("6 : etat sms");
    break;
    }
}

void loop()
{
    char c;
    if (sim8001.available()) Serial.write(sim8001.read());
    if (Serial.available())
    {
        c = Serial.read();
        traiterCmd(c);
    }
}

```