



SIM800 GSM / GPRS / SMS Add-on V2.0

Rev 1.2, Feb 2017



Contents

1、 Overview.....	3
2、 Feature	3
3、 Specification	3
4、 Hardware.....	3
5、 Pinmap.....	3
6、 Indicators.....	5
7、 Keys	5
8、 Extension	5
9、 Note	6
10、 RPI SIM800 operation instruction.....	6
11、 Schematic	8

1、 Overview

Raspberry Pi SIM800 GSM/GPRS Add-on V2.0 is customized for Raspberry Pi interface based on SIM800 quad-band GSM/GPRS/BT module. AT commands, can be sent via the serial port on Raspberry Pi, thus functions such as dialing and answering calls, sending and receiving messages and surfing on line can be realized. Moreover, the module supports powering-on and resetting via software.

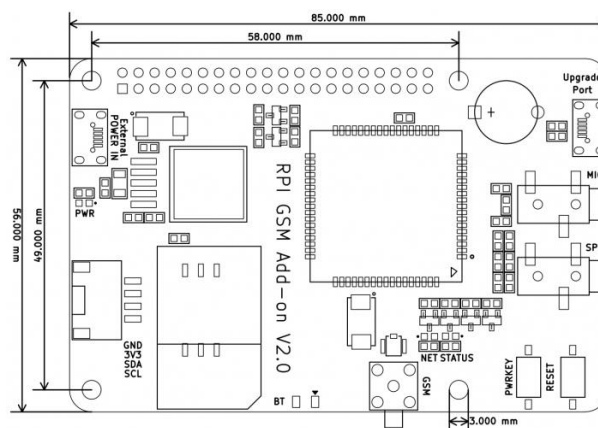
2、 Feature

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- GPRS multislots class 12
- Operation temperature: -40°C to +85 °C
- Stackable

3、 Specification

- PCB size: 85mm X 56mm X 1.6mm
- Operation Level: Digital 3.3V DC
- Interface: UART
- Baud rate: 9600(default)

4、 Hardware



5、 Pinmap

Raspberry PI Pin N.O.	Pin name	Description
1	3.3V	
2	5V	
3	GPIO02	SDA
4	5V	
5	GPIO03	SCL
6	GND	
7	GPIO04	
8	GPIO14	GSM_DIN
9	GND	
10	GPIO15	GSM_DOUT
11	GPIO17	SIM800-PWRKEY
12	GPIO18	SIM800-RST
13	GPIO27	
14	GND	
15	GPIO22	
16	GPIO23	
17	3.3V	
18	GPIO24	
19	GPIO10	SPI_MOSI
20	GND	
21	GPIO09	SPI_MISO
22	GPIO25	
23	GPIO11	SPI_SCK
24	GPIO08	SPI_CE0
25	GND	
26	GPIO07	SPI_CE1
27	ID_SD	
28	ID_SC	
29	GPIO05	
30	GND	
31	GPIO06	
32	GPIO12	
33	GPIO13	

34	GND	
35	GPIO19	
36	GPIO16	
37	GPIO26	
38	GPIO20	
39	GND	
40	GPIO21	

6、 Indicators

- PWR

Power: When there is normal supply to the board, the indicator keeps on.

- 8STATUS

SIM800 status: When SIM800 works normally, the indicator keeps on.

- NET

SIM800 network status: used to indicate network status, the working status of the indicator is as below:

Status	SIM800 behavior
OFF	SIM800 does not work
64ms on/800ms off	SIM800 does not find network
64ms on/3000ms off	SIM800 registers the network
64ms on/300ms off	GPRS communicates

7、 Keys

- PWRKEY

SIM800 powering-on key: Keep pressing the key for more than 1 second, SIM800 will be powered on; keep pressing the key for more than 1 second after the board is powered on, SIM800 will be powered off.

- RST

SIM800 resetting key: press the key, SIM800 will be reset.

8、 Extension

- External POWER IN

If SIM800 will drop off while working, please connect an external at least 5V/2A power supply to this micro USB.

- Upgrade Port

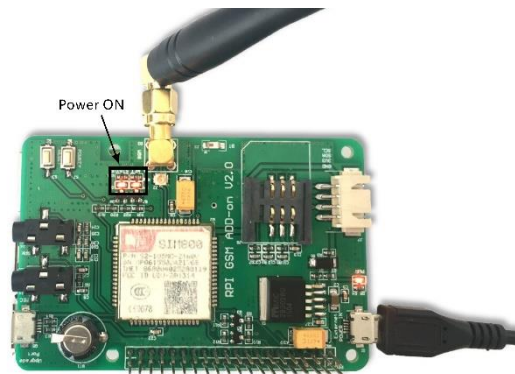
This is only for upgrading firmware of SIM800.

9、 Note

The module does not support hot swapping.

10、 RPI SIM800 operation instruction

Note: Firstly, you should key down the PWRKEY until the SIM800 is powered on.



Download the ArducamSIM800 library and put it in your /home/pi folder.

GitHub link: <https://github.com/supprot/ArducamSIM800.git>

Hardware:

If you are using a Raspberry Pi Model 1 A+/B+, Model 2/3 B or Zero then this add-on board will connect directly to the 40-pin GPIO.

If you are using a Model 1 A/B with a 28-pin GPIO connector, then you will have to connect the board to the Pi with ribbon cable or Dupont connectors as the composite video and audio connectors are in the way. Only the follow seven pins need to be connected:

Pin	Function
1	3.3V
2	5V
6	GND

8	TXD
10	RXD
11	SIM800 Power
12	SIM800 Reset

Software

This library has been tested with Python 3.4.2 running on a Raspberry Pi Model B (Rev.2) with Raspbian Jessie Lite (2016-05-27).

It depends on PySerial and Ben Croston's RPi. GPIO which can be installed (if not already) as follows:

Support for Pi /Pi2/ Pi Model B

```
sudo apt-get update
```

```
sudo apt-get install python3-rpi.gpio
```

```
sudo pip3 install pyserial
```

The file 'sms.py' is both the library and a working example if read/executed:

```
sudo python3 sms.py
```

Support for Pi3

There is now a device tree file called pi3-miniuart-bt which makes the Raspberry Pi 3 disable the Bluetooth and map pl011 UART on pins 14 and 15 as before.

- Step 1 - Install Raspbian Jessie onto a SD card and boot the Pi when connected to a network

Login via terminal or desktop and shell

Configure the system with:

```
sudo raspi-config
```

Expand filesystem and enable serial on advanced page, exit and reboot.

Update the system with:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

- Step 2: Device Tree settings as below:

Add device tree to /boot/config.txt to disable the Raspberry Pi 3 bluetooth.

```
sudo nano /boot/config.txt
```

Add at the end of the file

```
dtoverlay=pi3-miniuart-bt
```

Exit the editor saving your changes and then:

```
sudo reboot
```

Enabling the Serial Console Raspbian Jessie after 18th March 2016 release

To enable the serial console, you need to edit the /boot/cmdline.txt file

```
sudo nano /boot/cmdline.txt
```

Change the file to the following:

```
dwc_otg.lpm_enable=0 console=tty1 console=serial0,115200 root=/dev/mmcblk0p2  
rootfstype=ext4 elevator=deadline fsck.repair=yes rootwait
```

Exit and save your changes

```
sudo apt-get update
```

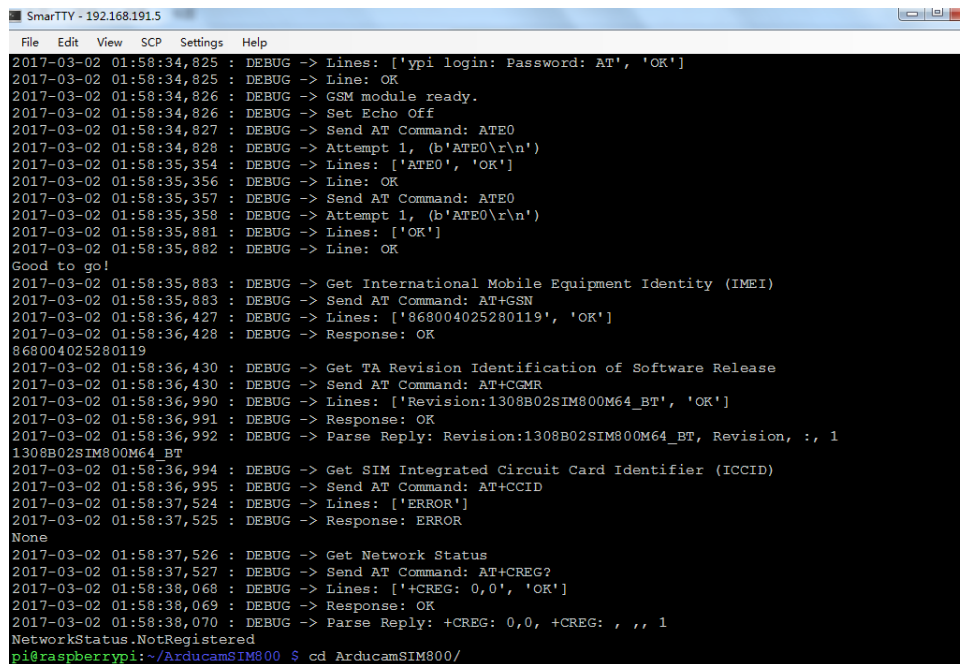
```
sudo apt-get install python3-rpi.gpio
```

```
sudo pip3 install pyserial
```

The file 'sms.py' is both the library and a working example if read/executed:

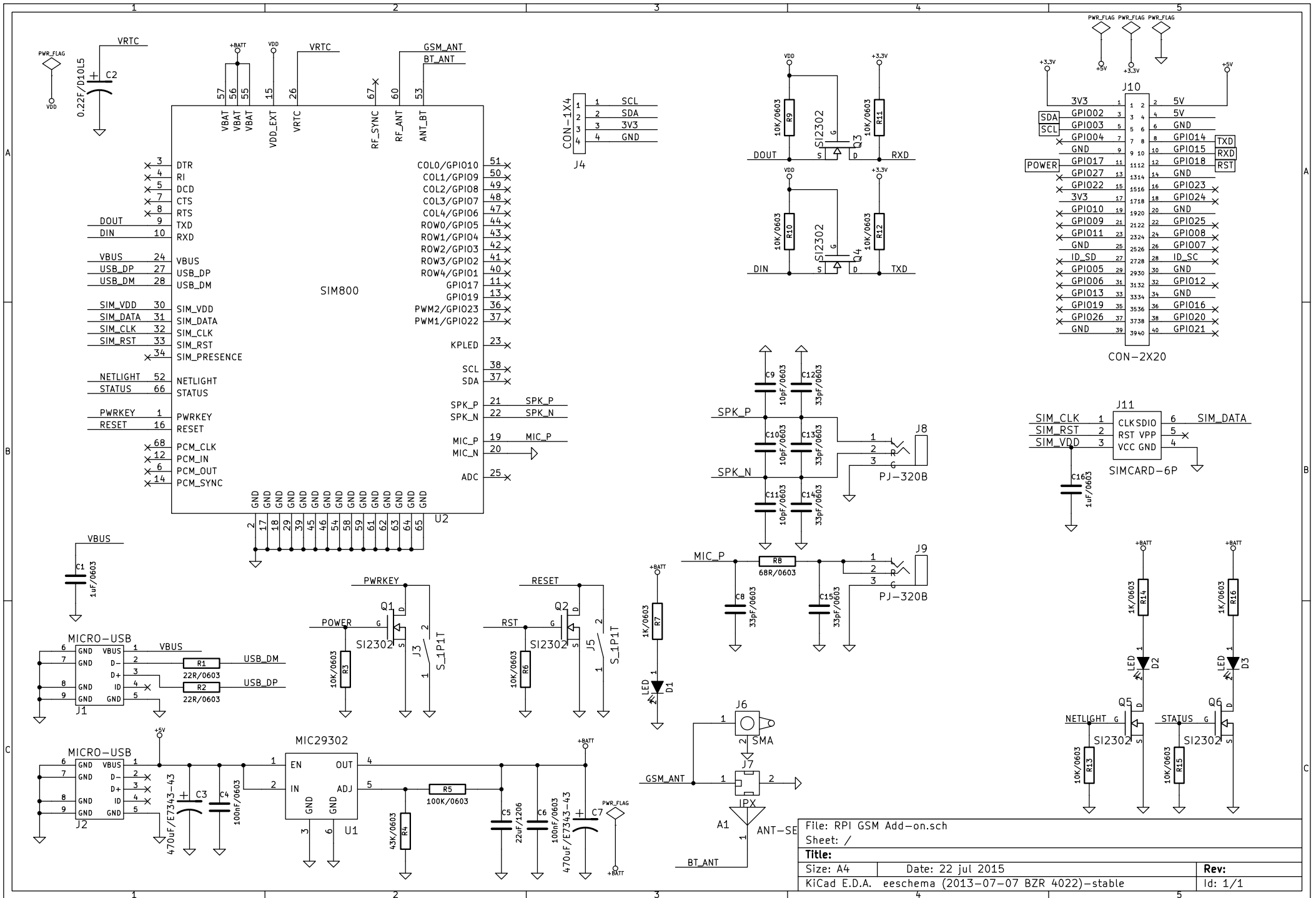
```
sudo python3 sms.py
```

Attach the demo running image:



```
SmarTTY - 192.168.191.5  
File Edit View SCP Settings Help  
2017-03-02 01:58:34,825 : DEBUG -> Lines: ['ypi login: Password: AT', 'OK']  
2017-03-02 01:58:34,825 : DEBUG -> Line: OK  
2017-03-02 01:58:34,826 : DEBUG -> GSM module ready.  
2017-03-02 01:58:34,826 : DEBUG -> Set Echo Off  
2017-03-02 01:58:34,827 : DEBUG -> Send AT Command: ATE0  
2017-03-02 01:58:34,828 : DEBUG -> Attempt 1, (b'ATE0\r\n')  
2017-03-02 01:58:35,354 : DEBUG -> Lines: ['ATE0', 'OK']  
2017-03-02 01:58:35,356 : DEBUG -> Line: OK  
2017-03-02 01:58:35,357 : DEBUG -> Send AT Command: ATE0  
2017-03-02 01:58:35,358 : DEBUG -> Attempt 1, (b'ATE0\r\n')  
2017-03-02 01:58:35,881 : DEBUG -> Lines: ['OK']  
2017-03-02 01:58:35,882 : DEBUG -> Line: OK  
Good to go!  
2017-03-02 01:58:35,883 : DEBUG -> Get International Mobile Equipment Identity (IMEI)  
2017-03-02 01:58:35,883 : DEBUG -> Send AT Command: AT+GSN  
2017-03-02 01:58:36,427 : DEBUG -> Lines: ['868004025280119', 'OK']  
2017-03-02 01:58:36,428 : DEBUG -> Response: OK  
868004025280119  
2017-03-02 01:58:36,430 : DEBUG -> Get TA Revision Identification of Software Release  
2017-03-02 01:58:36,430 : DEBUG -> Send AT Command: AT+CGMR  
2017-03-02 01:58:36,990 : DEBUG -> Lines: ['Revision:1308B02SIM800M64_BT', 'OK']  
2017-03-02 01:58:36,991 : DEBUG -> Response: OK  
2017-03-02 01:58:36,992 : DEBUG -> Parse Reply: Revision:1308B02SIM800M64_BT, Revision, :, 1  
1308B02SIM800M64_BT  
2017-03-02 01:58:36,994 : DEBUG -> Get SIM Integrated Circuit Card Identifier (ICCID)  
2017-03-02 01:58:36,995 : DEBUG -> Send AT Command: AT+CCID  
2017-03-02 01:58:37,524 : DEBUG -> Lines: ['ERROR']  
2017-03-02 01:58:37,525 : DEBUG -> Response: ERROR  
None  
2017-03-02 01:58:37,526 : DEBUG -> Get Network Status  
2017-03-02 01:58:37,527 : DEBUG -> Send AT Command: AT+CREG?  
2017-03-02 01:58:38,068 : DEBUG -> Lines: ['+CREG: 0,0', 'OK']  
2017-03-02 01:58:38,069 : DEBUG -> Response: OK  
2017-03-02 01:58:38,070 : DEBUG -> Parse Reply: +CREG: 0,0, +CREG: , , , 1  
NetworkStatus.NotRegistered  
pi@raspberrypi:~/ArducamSIM800 $ cd ArducamSIM800/
```

11. Schematic



File: RPI GSM Add-on.sch		Rev:	
Sheet: /		Date: 22 jul 2015	
Title:		Id: 1/1	
Size: A4	KiCad E.D.A. eeschema (2013-07-07 BZR 4022)-stable		