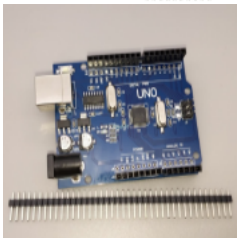
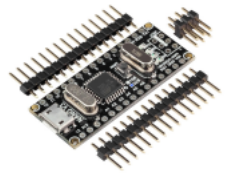
# Acoptex.com








36

members

# EASY Basics: Project 053d SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2)

of Acoptex.com in **UNO**

*Basics: Project 053d*

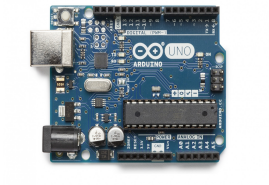
*Project name: SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2)*

*Tags: Arduino, Arduino Uno, SIM808, GPS GPRS GSM module, GPS GPRS GSM Bluetooth module, EVB-V3.2, AT commands, Firmware update, firmware updating, sim808, sim808 firmware update tool, sim808 firmware flash tool, sim800, SIMCOM, updating firmware gsm modem, power on and off without pressing the POWKEY / start button*

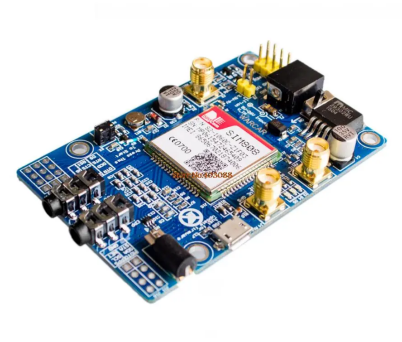
*Attachments: ATsetupsketch; makingcallsketch, answeringcallsketch, sendingsmssketch, readingsmssketch, turnOnOFFbySoftware sketch*

In this project, you needed these parts (***Dear visitors. You can support our project buy clicking on the links of parts and buying them or donate us to keep this website alive. Thank you***):

1.Arduino Uno R3 (you can also use the other version of Arduino)



2. SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2) 1pc



3.Arduino IDE ( you can download it from here )

4.Jumper cables F-M, M-M

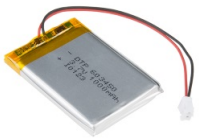
5. IPX GSM antenna with SMA male connector

6. GPS antenna with SMA male connector


7. IPX Bluetooth antenna with SMA male connector

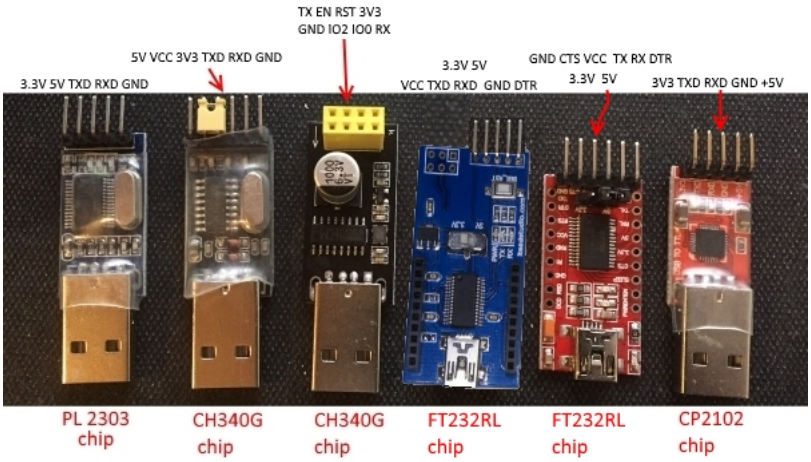
8. To power module: 5V-26V DC 2A power adapter or Litium Ion battery 3.7VDC

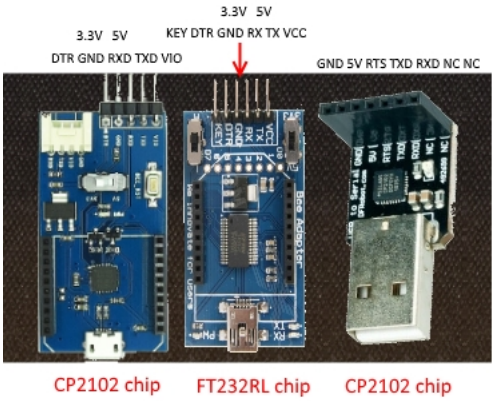
or



9. SIM card

10. USB to TTL/Serial adaptor/converter 1 pc

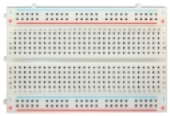





11. USB to Micro-USB Cable 1 pc (optional, for firmware update)

12.Resistor 2 pcs (10 KOhm 1 pc, 4.7 KOhm 1 pc)

13.Breadboard half size or small size 1 pc



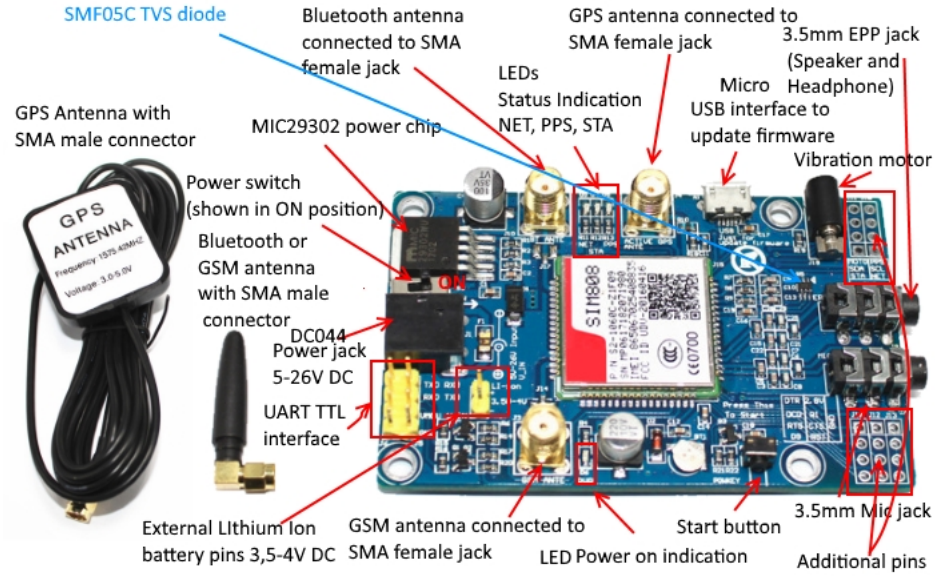
*General*

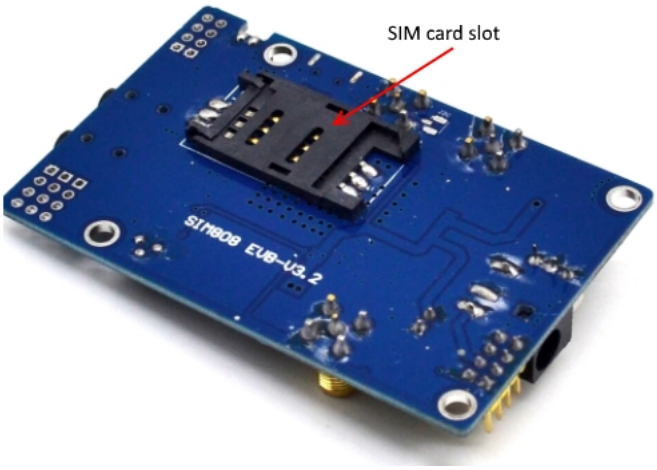
We will learn how to connect SIM808 GPS module to Arduino board and use it. Same time we will upgrade the firmware of the module to the latest version.

*Understanding the USB to TTL converter*

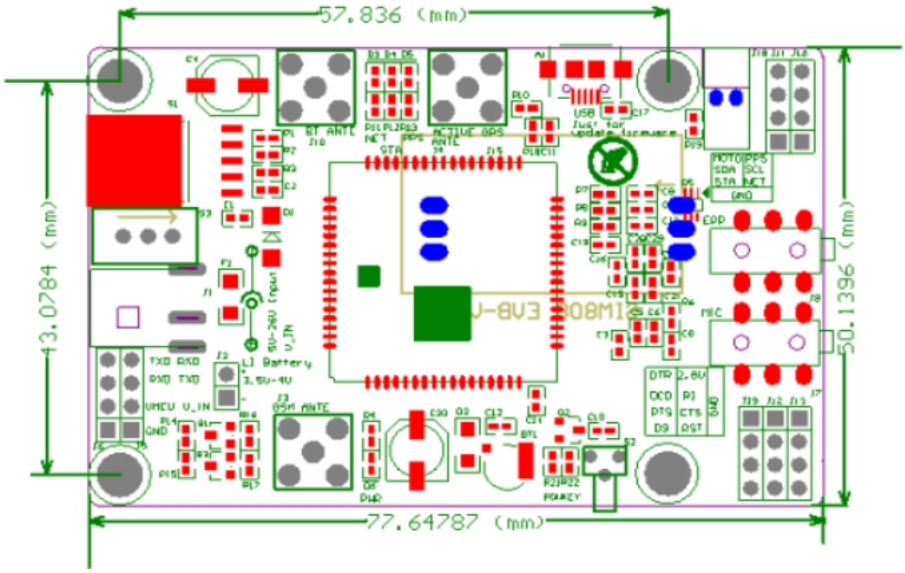
You can read more about them here.

*Understanding the SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2)*






SIM808 module is a complete Quad-Band GSM/GPRS module which combines GPS technology for satellite navigation. The compact design which integrated GPRS and GPS in a SMT package will significantly save both time and costs for customers to develop GPS enabled applications. Featuring an industry-standard interface and GPS function, it allows variable assets to be tracked seamlessly at any location and anytime with signal coverage.




**Functions:**

- Send and receive GPRS data (TCP/IP, HTTP, etc.)
- Receive GPS data and A-GPS data
- Send and receive SMS messages
- Make and receive phone calls

**General features:**

- Quad-band 850/900/1800/1900MHz
- GPRS multi-slot class 12/10
- GPRS mobile station class B
- Compliant to GSM phase 2/2+ (Class 4 (2 W @ 850/900MHz; Class 1 (1 W @ 1800/1900MHz))
- Bluetooth: compliant with 3.0+EDR
- FM: 76~109MHz worldwide bands with 50KHz tuning step
- Dimensions: 24.0*24.0*2.6mm
- Weight: 3.30g
- Control via AT commands (3GPP TS 27.007, 27.005 and SIMCOM enhanced AT Commands)
- Supply voltage range 3.4 ~ 4.4V
- Low power consumption
- Operation temperature:-40℃ ~85℃

**Specifications for GPRS Data:**

- GPRS class 12: max. 85.6 kbps (downlink/uplink)
- PBCCH support
- Coding schemes CS 1, 2, 3, 4
- PPP-stack
- CSD up to 14.4 kbps
- USSD

**Specifications for SMS via GSM/GPRS:**

- Point to point MO and MT
- SMS cell broadcast
- Text and PDU mode

**Software features:**

- 0710 MUX protocol
- Embedded TCP/UDP protocol
- FTP/HTTP
- MMS
- POP3/SMTP
- DTMF
- Jamming Detection
- Audio Record
- SSL
- Bluetooth 3.0 (optional)
- TTS CN(optional)
- Embedded AT (optional)

**Specification for GPS:**

- Receiver type (22 tracking /66 acquisition -channel; GPS L1 C/A code)
- Sensitivity (Tracking: -165 dBm; Cold starts : -147 dBm)
- Time-To-First-Fix (Cold starts: 30s (typ.); Hot starts: 1s (typ.); Warm starts: 28s (typ.))
- Accuracy (Horizontal position : <2.5m CEP)

**Interfaces:**

- 68 SMT pads including:
- Analog audio interface
- PCM interface (optional)
- SPI interface (optional)
- RTC backup
- Serial interface
- USB interface
- Interface to external SIM 3V/1.8V
- Keypad interface
- GPIO
- ADC
- GSM Antenna pad
- GPS Antenna pad
- Bluetooth Antenna pad

**Compatibility:**

- AT cellular command interface

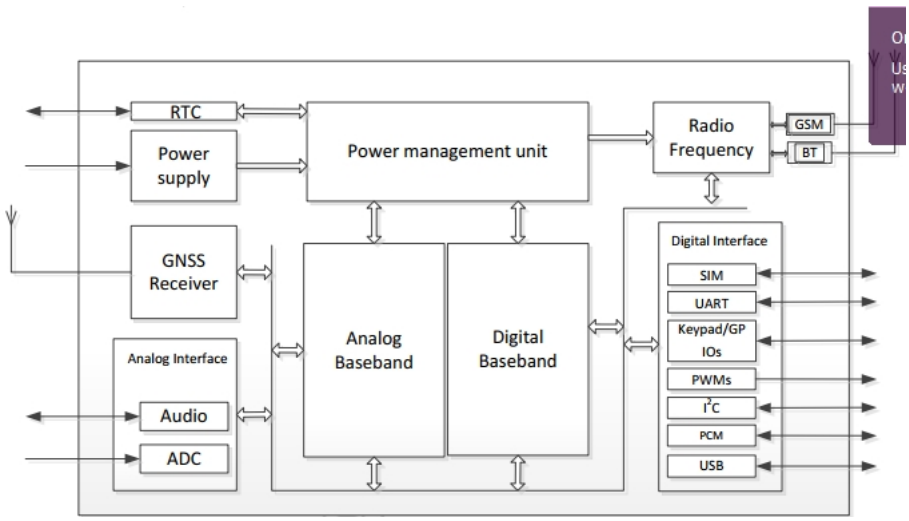
**Certifications:**

- CE
- A-TICK
- TA
- CCC

**SIM808 module:**

- Supports 2G 3G 4G SIM Card
- Can be used with Raspberry Pi and Arduino
- has SMA antenna interface: there is a GSM antenna interface, a GPS antenna interface onboard and a BT antenna interface.
- has three power input interface: DC044 interface and V_IN and a lithium battery interface. Note that the range of DC044 and the V_IN pin voltage input is 5 - 26V, when use the 5V as the power, be sure that the power supply can provide 2A current. The range of voltage of Lithium battery input power is 3.5 - 4.2V.
- has USB interface, which is used to update the firmware of SIM808 module.

**SIM808 functional diagram:**




**Operating Modes**

1.Normal operation

- GSM/GPRS SLEEP. Module will automatically go into sleep mode if the conditions of sleep mode are enabling and there is no on air and no hardware interrupt (such as GPIO interrupt or data on serial port). In this case, the current consumption of module will reduce to the minimal level. In sleep mode, the module can still receive paging message and SMS.

- GSM IDLE. Software is active. Module registered to the GSM network, and the module is ready to communicate.
- GSM TALK. Connection between two subscribers is in progress. In this case, the power consumption depends on network settings such as DTX off/on, FR/EFR/HR, hopping sequences, antenna.
- GPRS STANDBY. Module is ready for GPRS data transfer, but no data is currently sent or received. In this case, power consumption depends on network settings and GPRS configuration.
- GPRS DATA. There is GPRS data transfer (PPP or TCP or UDP) in progress. In this case, power consumption is related with network settings (e.g. power control level); uplink/downlink data rates and GPRS configuration (e.g. used multi-slot settings).
- Charge. The mode support charge function (Default is closed).

2. Power off

Normal power off by sending the AT command “AT+CPOWD=1” or using the PWRKEY. The power management unit shuts down the power supply for the baseband part of the module, and only the power supply for the RTC is remained. Software is not active. The serial port is not accessible. Power supply (connected to VBAT) remains applied.

3. Minimum functionality mode

AT command “AT+CFUN” can be used to set the module to a minimum functionality mode without removing the power supply. In this mode, the RF part of the module will not work or the SIM card will not be accessible, or both RF part and SIM card will be closed, and the serial port is still accessible. The power consumption in this mode is lower than normal mode.

You can find the SIM808 datasheet, programs to upload new firmware and other documents here and here.

**Handling GPS Data**

Once the GPS receiver has transmitted the NMEA sentences to the Arduino, and they have been properly read and stored in nice and "simple format", then what follows?

There are several options to convert the "simple format" data into more general format (GPX, KMZ, etc.). For instance:

1. GPS Visualizer is a free online utility "that creates maps and profiles from geographic data". You can input a CSV or tabbed file, a spreadsheet, or drag and drop the data. The appearance of the page is a bit odd but the content is good.
2. GPS Prune is intended to view, edit and convert GPS data. It allows to load text files as well as NMEA files, among quite a number of other options.
3. GPS Babel seems to be the most known GPS data converter. It reads text files with NMEA sentences.

**U-center GNSS evaluation software for Windows**

The u-center GNSS evaluation software for automotive, mobile terminal and infrastructure applications provides a powerful tool for evaluation, performance analysis and configuration of u-blox GNSS receivers. Its unique flexibility makes the u-center GNSS evaluation software an invaluable tool for evaluation, analysis and configuration of u-blox GNSS receivers. u-blox GNSS receivers can be configured using the u-center evaluation software.

- Highly interactive and easy to use
- Full support of all u-blox GNSS receivers
- Extensive configuration and control features
- Real-time display from a GNSS receiver via RS232 and USB interface

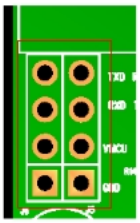
See more details here.

*Signals and connections of the USB to TTL converter*

You can read more about them here.

*Signals and connections of the SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2)*

**The Pins distribution of UART TTL interface is:**



| | |
|---|---|
| TXD | RXD |
| RXD | TXD |
| VMCU | V_IN |
| GND | GND |

**RX** (or **RXD**) - receive data input pin. Connected to Arduino board or USB to TTL converter TX pin.

**TX** (or **TXD**) - transmit data output pin. Connected to Arduino board or USB to TTL converter RX pin.

**V_IN** - power supply 5-26VDC

**LI-ion +** - positive pin for Li-Ion battery 3.5-4V DC

**LI-ion -** - negative pin for Li-Ion battery 3.5-4V DC

**POWKEY** - start button. press it to start the module (how to turn on by software - check the note in **Tips and Hacks below**)

**VMCU** - The pin of VMCU is used to control the high level of TTL UART, so as to realize to match between 1.25V/3.3V /5V systems. For example, if you want to use the 51 MCU to control this board, the pin of VMCU should be connected the DC5V. And if use the STM32 MCU, the pin of VMCU should be connected the DC3.3V.

**GND** - ground. Connected to Arduino board GND pin.

**MIC -** microphone jack input. An external microphone connected to it.

**EPP** - speaker or headphone jack output. Speaker or headphone connected to it.

**PPS** - Pulse per second. This is an output pin on some GPS modules. Generally, when this pin toggles, once a second, you can synchronize your system clock to the GPS clock.

**MOTO** - Vibration motor

**SDA** - I2C serial bus data I/O pin

**SCL** - I2C serial bus clock output pin

**NET** - Network status.

**STA** - Power on status

**DTR** - Data terminal ready input pin. Used for Serial communication.

**2.8V** - power supply 2.8VDC

**RI** - Ring indicator output pin.

**CTS** - Clear to send output pin.

**RST** - Reset input(Active low)

**D9** - GPIO pin.

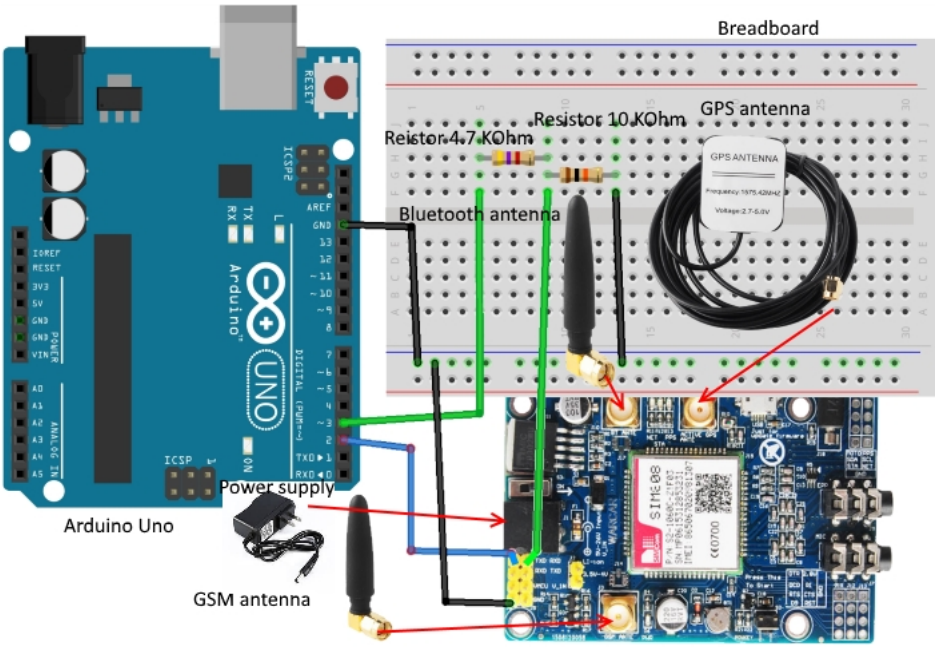
**RTS** - Request to send input pin.

**DCD** - Data carrier detect output pin.

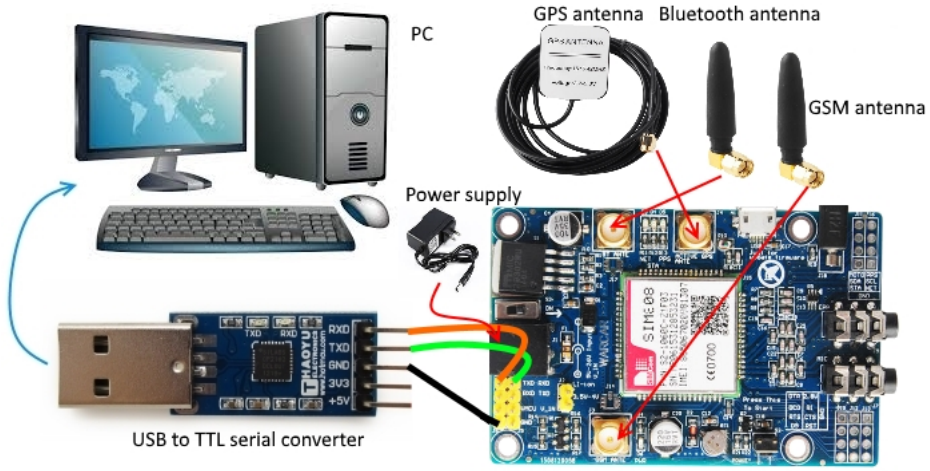
The power supply range of SIM808 is from 3.4V to 4.4V. Recommended voltage is 4.0V.The transmitting burst will cause voltage drop and the power supply must be able to provide sufficient current up to 2A.

*Wiring*

**1. Using Arduino board**

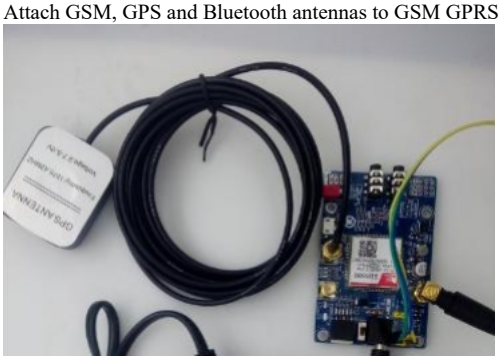
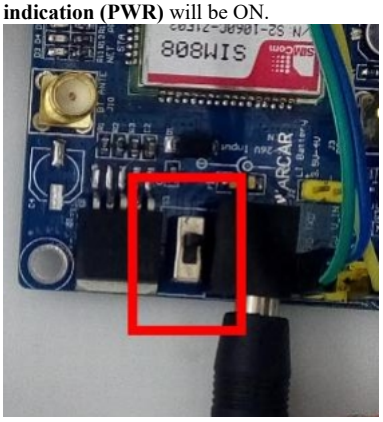


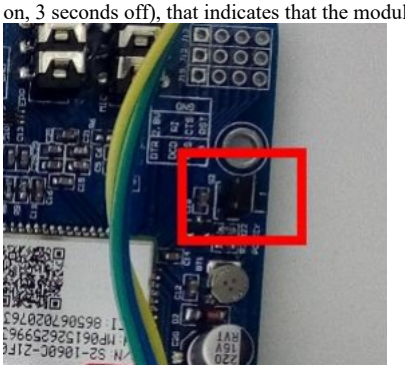
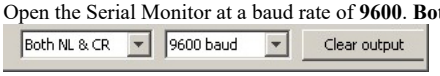

**2. Using USB to TTL converter**



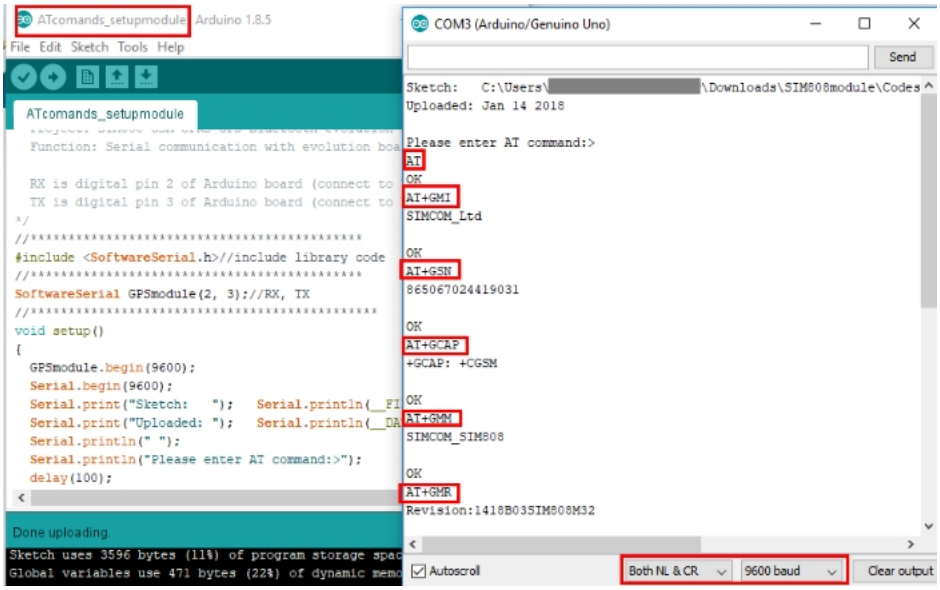

*Step by Step instruction*

1. Configuring the GSM GPRS GPS Bluetooth module with Arduino board.

1. Do wiring.
2. Attach GSM, GPS and Bluetooth antennas to GSM GPRS GPS Bluetooth module.
3. 
4. Insert a valid unlocked SIM card to SIM card slot of GSM GPRS GPS Bluetooth module.
5. You can power up GSM GPRS GPS Bluetooth module with power supply adapter (5-26VDC 2A) DC044 interface, with V_IN pin (5-26VDC 2A) and with Lithium Ion battery 3,5-4VDC.
6. 
7. We used the **5VDC 2A power adapter**. Push the power switch to ON position - towards the SIM808 IC. When the module powered on, the **LED power on indication (PWR)** will be ON.
8. 

9. Press **Start button** for 2 seconds. **Status indication LEDs (NET, STA, PPS)** will be ON. One of them starts to flash (1 second on, 1 second off) - SIM808 begins to work. If the power supply, GSM, Bluetooth, GPS antennas and SIM card connected to the module correctly, the LED will be flash slowly (1 second on, 3 seconds off), that indicates that the module is registered to the network, and you can make a call or do something else.
10. 
11. Open Arduino IDE.
12. Plug your Adruino Uno board into your PC and select the correct board and com port
13. Verify and upload the ATsetupsketch to your Adruino Uno
14. Open the Serial Monitor at a baud rate of **9600**. **Both NL & CR** must be selected too.
15. 

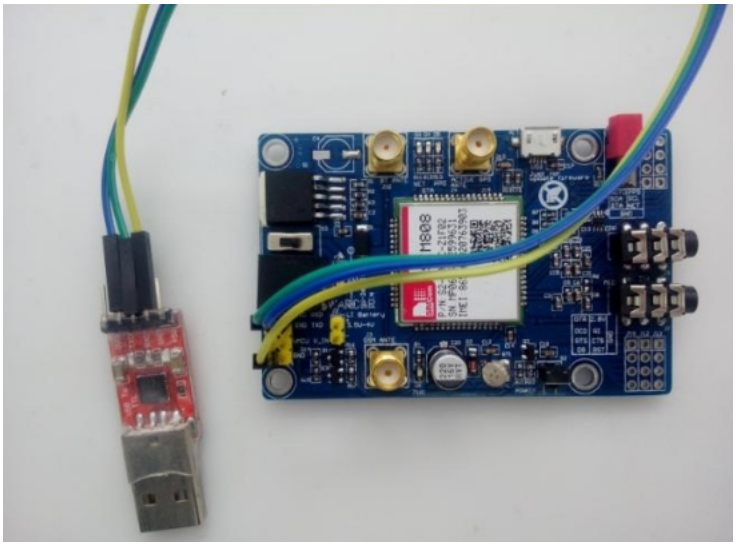
16. First you need to check if AT commands are working - enter "**AT**" and press button Send.This would print **OK** which signifies of working connection and operation of the module.
17. Enter "**AT+GMI**" and press button Send. Returns the Manufacturer Identification of the module: "SIMCOM_Ltd OK"
18. Enter "**AT+GSN**" and press button Send. Returns the TA Serial Number Identification (IMEI) of the module: "865067024419031 OK"
19. Enter "**AT+GMM**" and press button Send. Returns the TA Model Identification of the module: "SIMCOM_SIM808 OK"
20. Enter "**AT+GMR**" and press button Send. Returns the TA Revision Identification of Software Release of the module: "Revision:1418B03SIM808M32 OK"
21. Enter "**ATI**" and press button Send. Returns the product identification information: "SIM808 R14.18 OK"
22. We will show you some at commands for GSM.Check for a list of the available AT commands here.

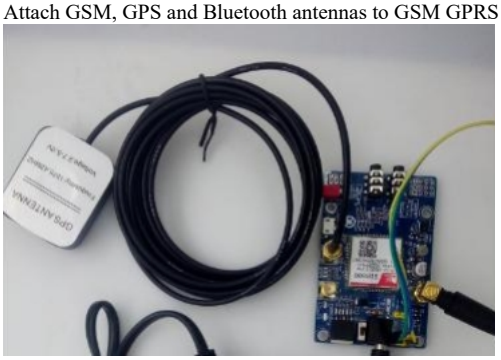
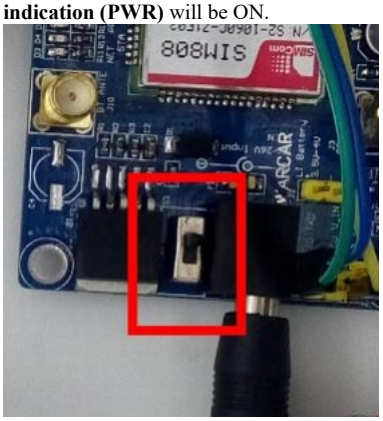
23. 


24.

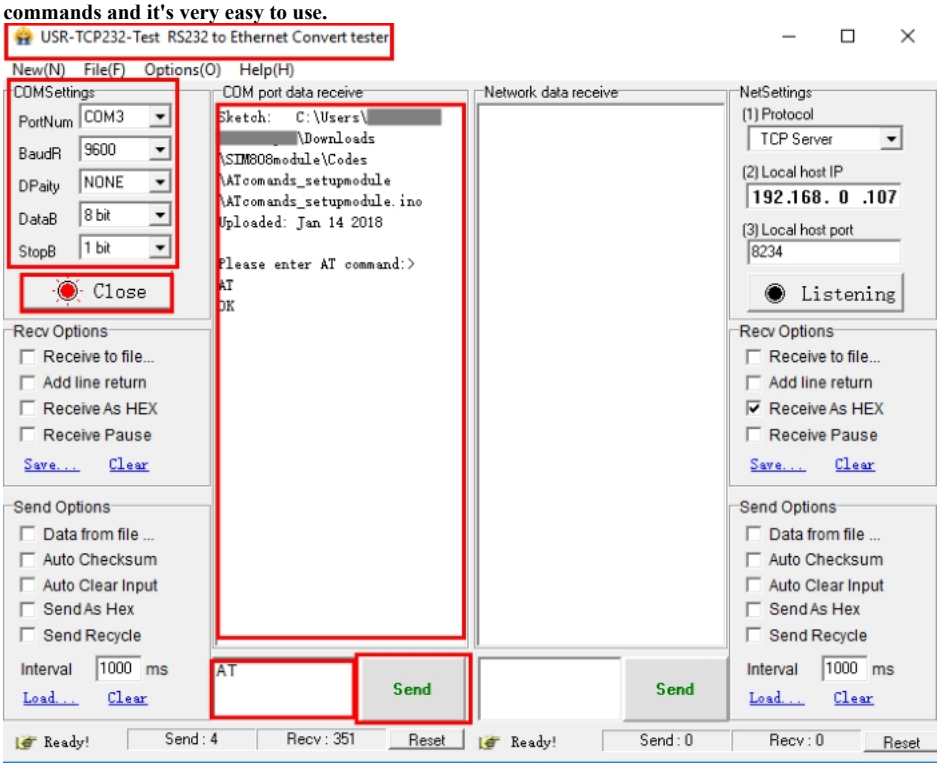
| | |
|---|---|
| AT | Check module status. |
| AT + CPIN = "XXXX" | Enter the PIN of the SIM. Change XXXX by the PIN. |
| AT + CREG? | Check the connection to the network. |
| ATDXXXXXXX; | Make a call Replace XXXXXXXXX with the number you want to call. |
| ATA | Pick up a call. |
| ATH | End the call. |
| AT + CMGF = 1 | Set the text mode to send or receive messages. Returns ">" as inductor. |
| AT + CMGS = "XXXXXXXXX" | No. to which we are going to send the message. |
| AT + CLIP = 1 | We activate the call identification. |
| AT + CNMI = 2,2,0,0,0 | We configure the module to show the SMS through the serial port. |
| AT + CGATT = 1 | We connect to the GPRS network. |
| AT + CSTT = "APN", "user", "password" | We define APN. user and password |
| AT + CIICR | We activate the wireless data profile |
| AT + CIFSR | We get our IP |
| AT + CIPSTART = "TCP", "IP address", "port" | We indicate the type of connection, IP address and port to which we made the connection |
| AT + CIPSEND | We prepare the sending of data. Returns ">" as inductor. |
| AT + CIPCLOSE | We closed the connection. |
| AT + CIPSHUT | Close the PDP context of the GPRS |
| AT + CGPSPWR = 1 | Activate the GPS. |
| AT + CGPSSTATUS? | Check that the GPS has found the network. |
| AT + CGPSINF = 0 | Get the GPS data: Mode, Latitude, Longitude, Altitude, Timetable, Response time, Number of satellites, Speed, Course |
| AT + CGPSOUT = 32 | to obtain GPS data using the NMEA specification: UTC Time, State, Latitude, Longitude, Speed in knots, Defeat angle in degrees, Date, Magnetic variation, Cheksum data |
| AT + CGPSPWR = 0 | Close the GPS. |

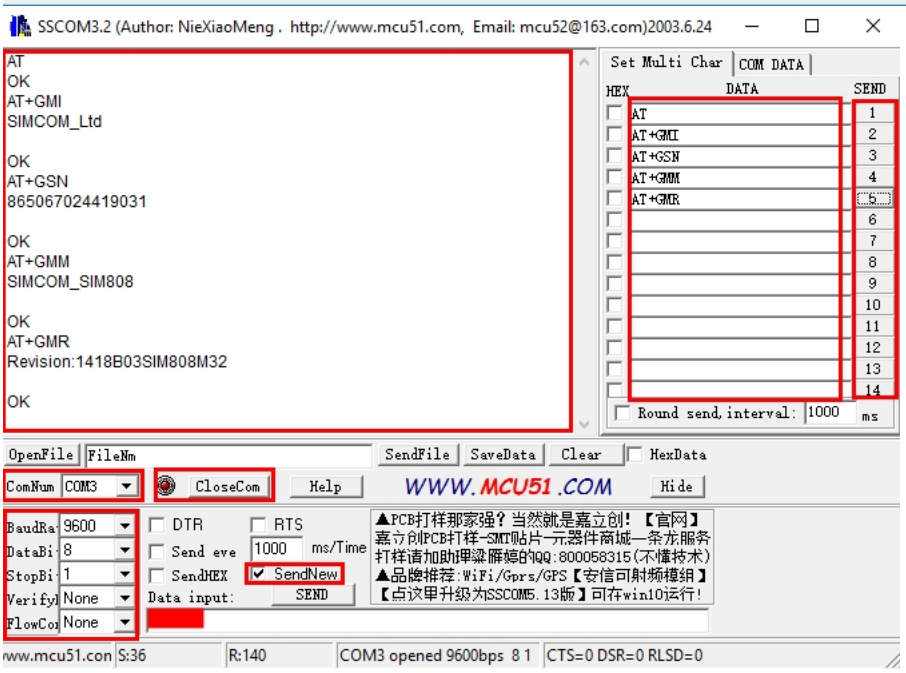
2. Configuring the GSM GPRS GPS Bluetooth module *USB to TTL converter*

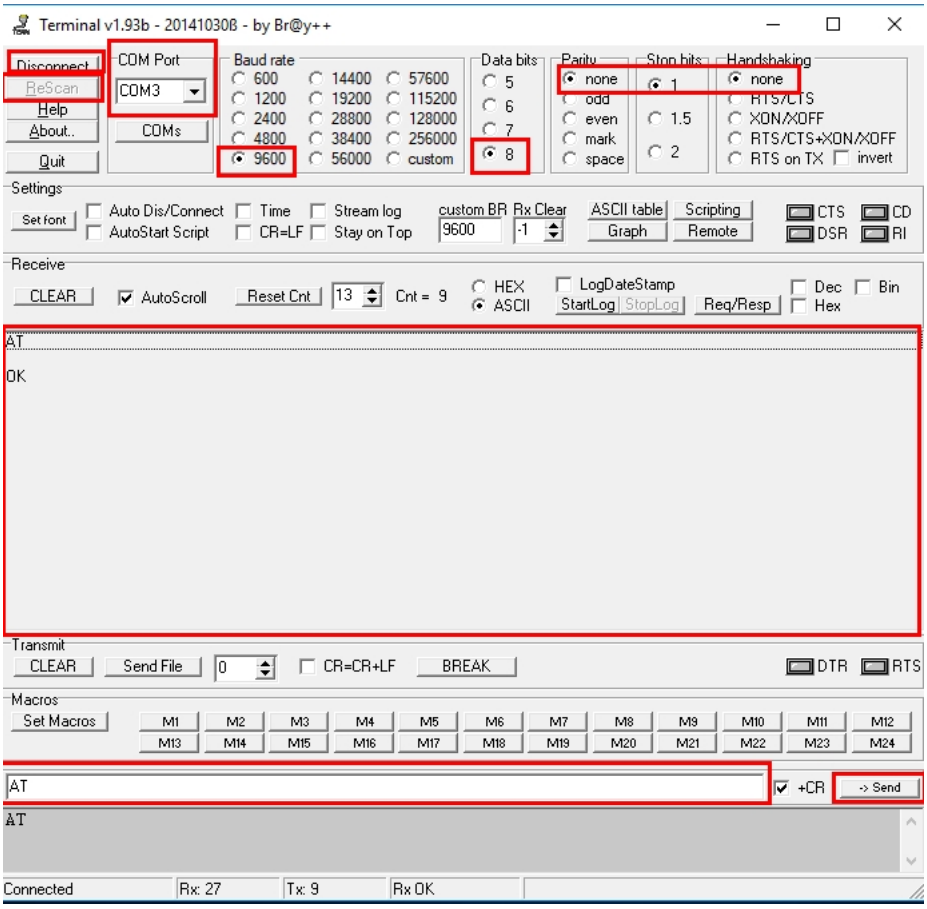


1. Do wiring.
2. Attach GSM, GPS and Bluetooth antennas to GSM GPRS GPS Bluetooth module.
3. 
4. Insert a valid unlocked SIM card to SIM card slot of GSM GPRS GPS Bluetooth module.
5. You can power up GSM GPRS GPS Bluetooth module with power supply adapter (5-26VDC 2A) DC044 interface, with V_IN pin (5-26VDC 2A) and with Lithium Ion battery 3,5-4VDC.
6. 
7. We used the **5VDC 2A power adapter**. Push the power switch to ON position - towards the SIM808 IC. When the module powered on, the **LED power on indication (PWR)** will be ON.
8. 

9. Press **Start button** for 2 seconds. **Status indication LEDs (NET, STA, PPS)** will be ON. One of them starts to flash (1 second on, 1 second off) - SIM808 begins to work. If the power supply, GSM, Bluetooth, GPS antennas and SIM card connected to the module correctly, the LED will be flash slowly (1 second on, 3 seconds off), that indicates that the module is registered to the network, and you can make a call or do something else.

10. 
11. Plug your USB to TTL converter into your PC USB port
12. For using this USB to UART Converter you need a software tool. Most of the development tools like MikroC Pro, Arduino has a UART Tool along with it. You can use different software tools: Terminal, sscom32E, USR-TCP232-Test_V1.4. **We recommend you to use sscom32E, as you can save your AT commands and it's very easy to use.**
13. 

14. 

15. 


3. Updating firmware of the GSM GPRS GPS Bluetooth module

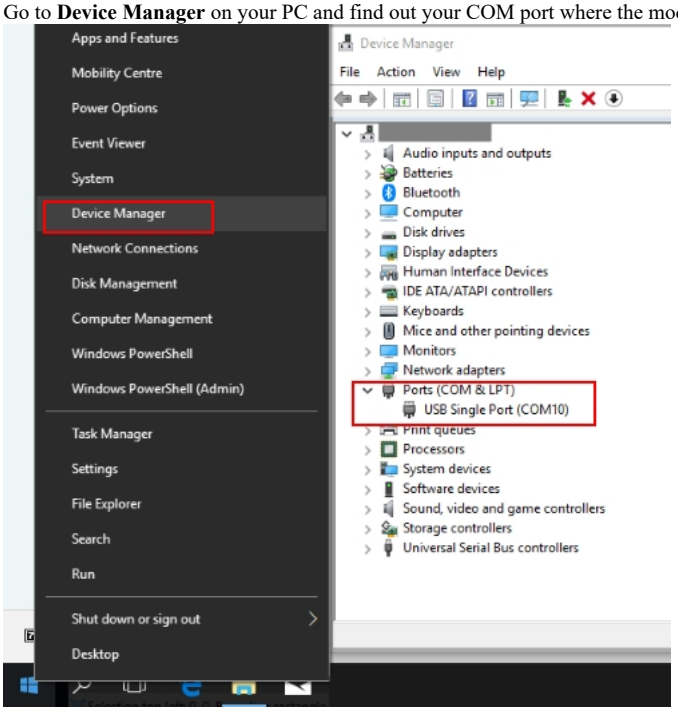
**VERY IMPORTANT NOTE:**
**Before doing firmware update:**

1. Follow the steps as described above in 1. Configuring the GSM GPRS GPS Bluetooth module with Arduino board or 2. Configuring the GSM GPRS GPS Bluetooth module *USB to TTL converter*. Check your module firmware version. Enter "**AT+GMR**" and press button **Send**. Returns the TA Revision Identification of Software Release of the module, for example: "Revision:1418B03SIM808M32 OK". Set baud rate to 115200 bps (It's 9600 by default) Enter "**AT+IPR=115200**" and press button **Send**. Returns the: " OK". If you check it with command "**AT+ITR?**" - it will return: "+IPR:115200 OK"
2. Always make sure that you have your firmware version on hands so you can restore previous version if next firmware version is bad, for example we had previous version 1418B03SIM808M32.zip.
3. Read all information about firmware update available on Web.
4. Just think if you realy need to do it as it is a risk to damage the module!!!
5. If you decided to do it make sure that the next firmware version which you are planing to upload is not older than existing (For example we have two newer firmware versions - 1418B03SIM808M32_BT_EAT.zip and 1418B04SIM808M32.zip). You can check for new versions here.
6. Make sure that you have **Media Tek driver** for USB to Micro USB cable. You can download here.
7. Make sure that you have USB to Micro USB data cable or USB to TTL converter.

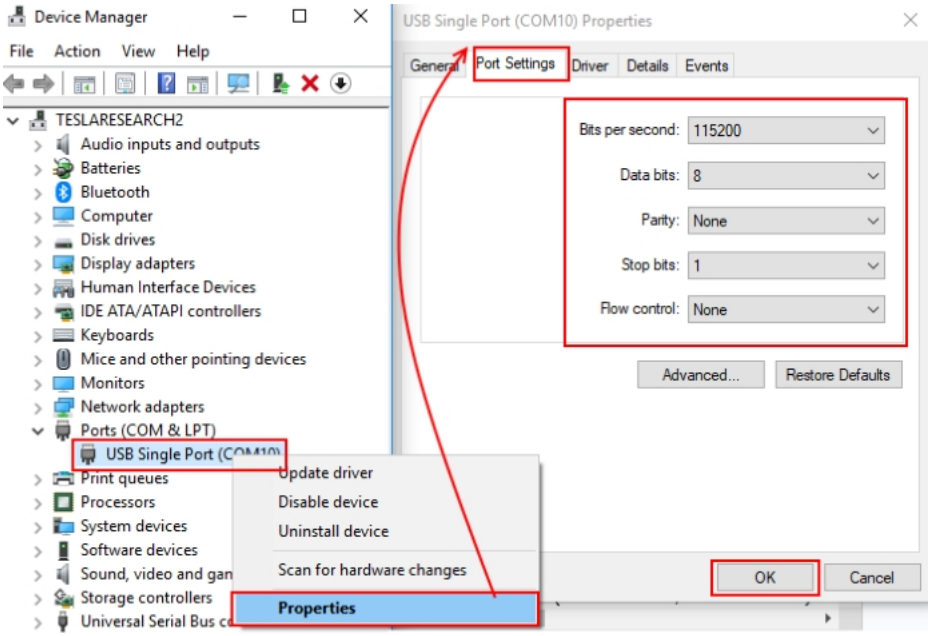
**Preparations and upgrade**

- Both UART TTL interface (serial port) and USB interface can be used for firmware upgrade.
- It is recommended to use the USB interface as it's faster then the UART TTL interface.
- There are SEVERAL options how to do upgrading of firmware: 1. Connect USB to Mirco USB cable, Press Start Download button, Power on the module (USB interface); 2 Power on the module, Press Start Download button, Connect USB to Mirco USB cable (USB interface); 3. Connect USB to TTL converter, Press Start Download button, Power on the module (UART TTL interface). We will show you the FIRST option here. See below.

**Firmware versions:**
1418B03SIM808M32.zip
1418B03SIM808M32_BT_EAT.zip
1418B04SIM808M32.zip
**USB interface:**

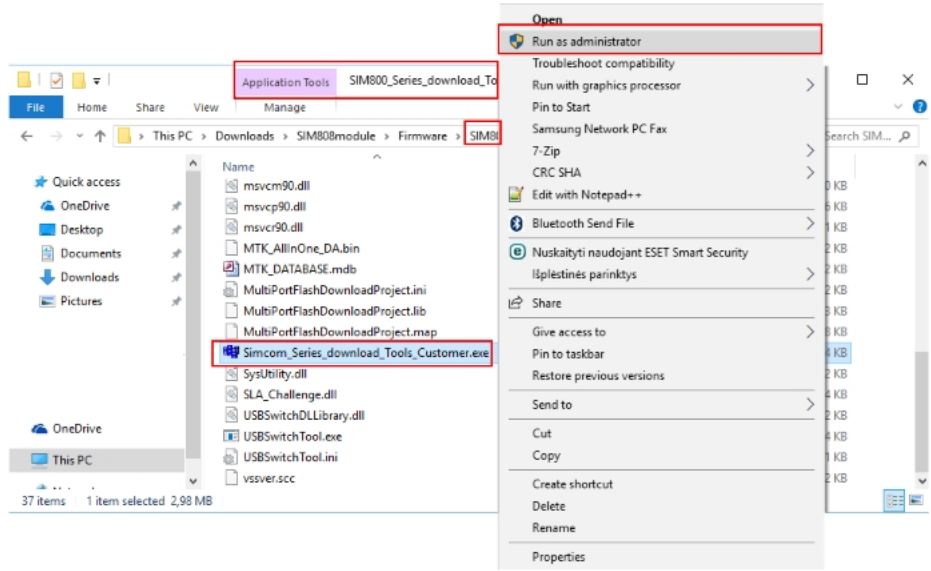

1. Download and install the **Media Tek driver** for USB to Micro USB cable. You can download here.
2. Download and unzip the firmware upgrade tool **SIM800_Series_download_Tools_Customer_v1.10**. You can find it here.
3. Make sure that you have USB to Micro USB data cable. Connect the micro usb port on the module with your PC USB port.
4. Power up GSM GPRS GPS Bluetooth module with power supply adapter (5-26VDC 2A) through the DC044 interface.
5. 
6. Go to **Device Manager** on your PC and find out your COM port where the module connected. **We have COM10 and we are using Windows 10.**
7. 

8. Go to **COM10 port settings** - click with right mouse button on COM10 port. Make sure that the COM10 port (serial port of your PC where the module connected) set as follows: 115200 bps, 8 bit, No parity bit, 1 stop bit, no flow control. Press **OK** button. See on the picture below.

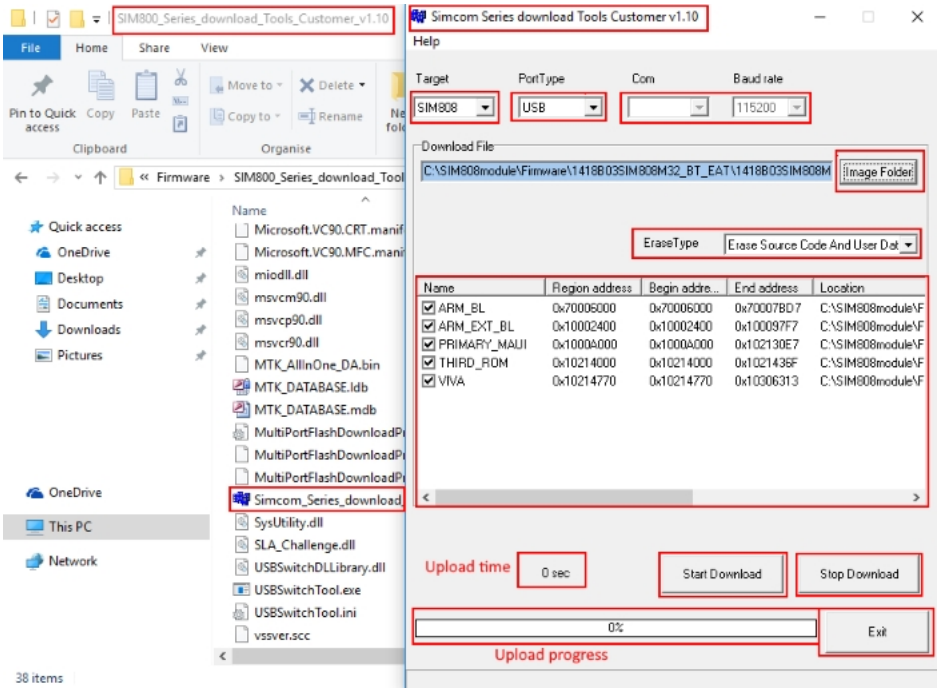


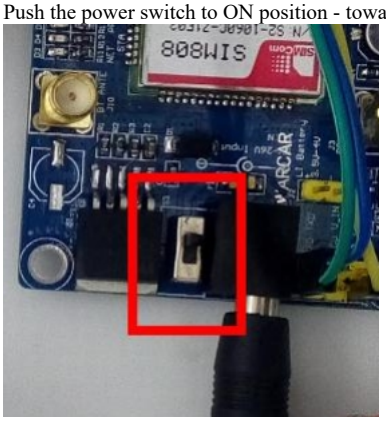

9. Opens propert

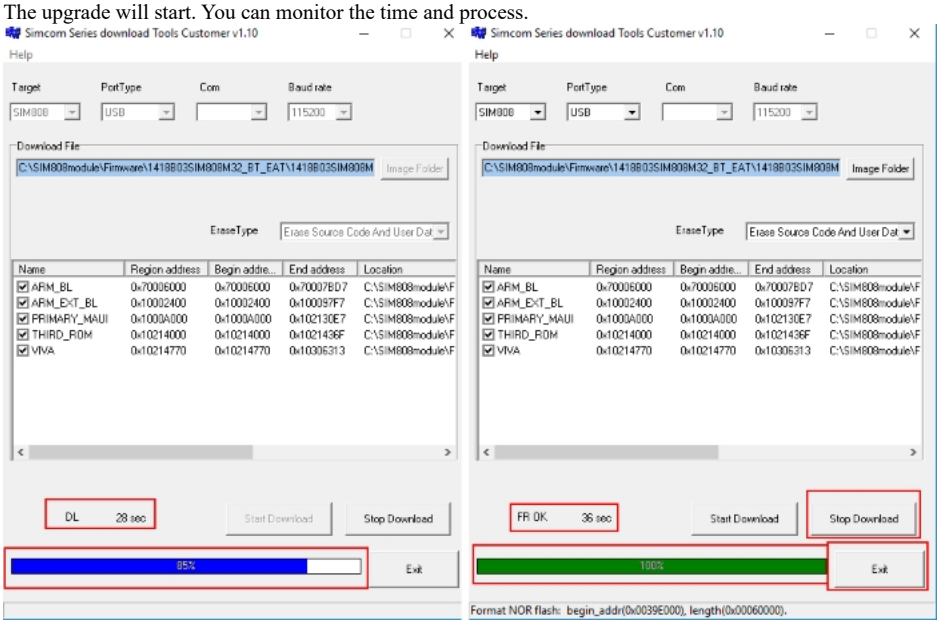
10. Open **SIM800_Series_download_Tools_Customer_v1.10** folder and find the file **Simcom_Series_download_Tools_Customer.exe**. Right-click on this file and select **Run as Administrator** and press **Yes** button**.**



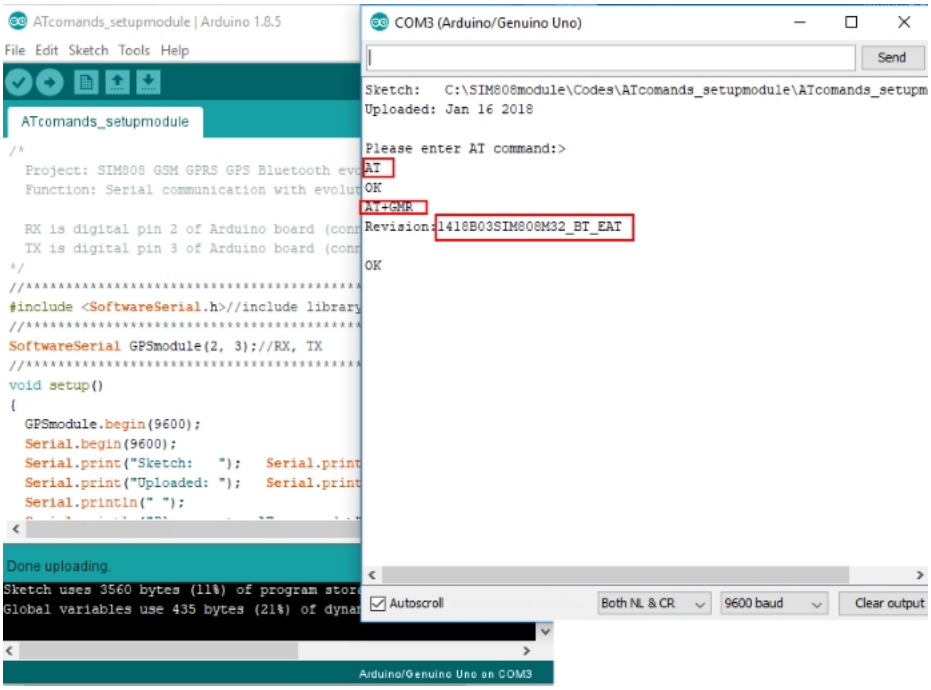

11.
12. Select **Target - SIM808** (you have SIM808 IC), **PortType - USB**. **Com** and **Baud rate** are not available (grey color).
13. Press **Image Folder** button and find the location of firmware. Firmware configuration file has the **CFG** extension. We selected **1418B03SIM808M32_BT_EAT.cfg** file.
14. Select **Erase Source Code and User Data** in **EraseType** field.

15. 

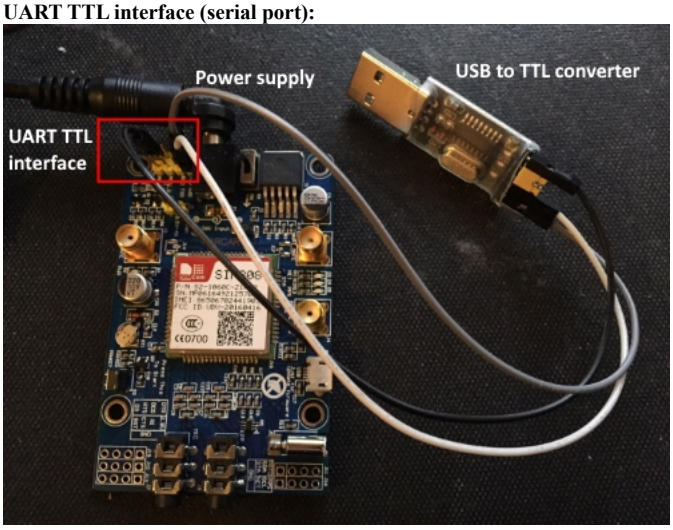
16. Make sure that your PC and the module have a steady power supply.
17. Press **Start Download** button.
18. Push the power switch to ON position - towards the SIM808 IC. When the module powered on, the **LED power on indication (PWR)** will be ON.
19. 

20. The upgrade will start. You can monitor the time and process.
21. 

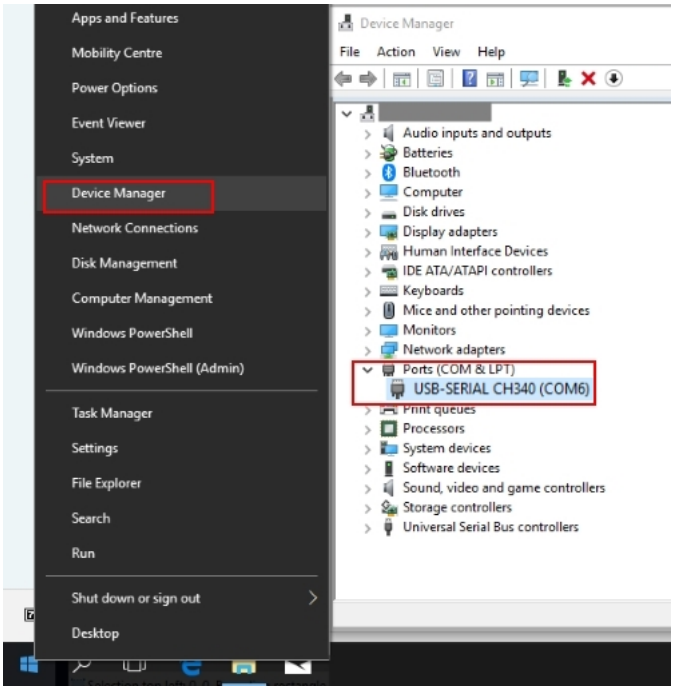
22. Press **Stop Download** button when the bar will be green (100%) and FR OK.
23. Press **Exit** button.
24. Make sure that you follow the instruction precisely. When you finish upgrade you need to reset the module: unplug the module from PC USB port and power off.
25. You can also check that you module has a newer version. Follow the steps as described above in 1. Configuring the GSM GPRS GPS Bluetooth module with Arduino board or 2. Configuring the GSM GPRS GPS Bluetooth module *USB to TTL converter*. Use AT command "**AT+GMR**" for that, for example, you can see that we have reply: "Revision:1418B03SIM808M32_BT_EAT OK"

26. 

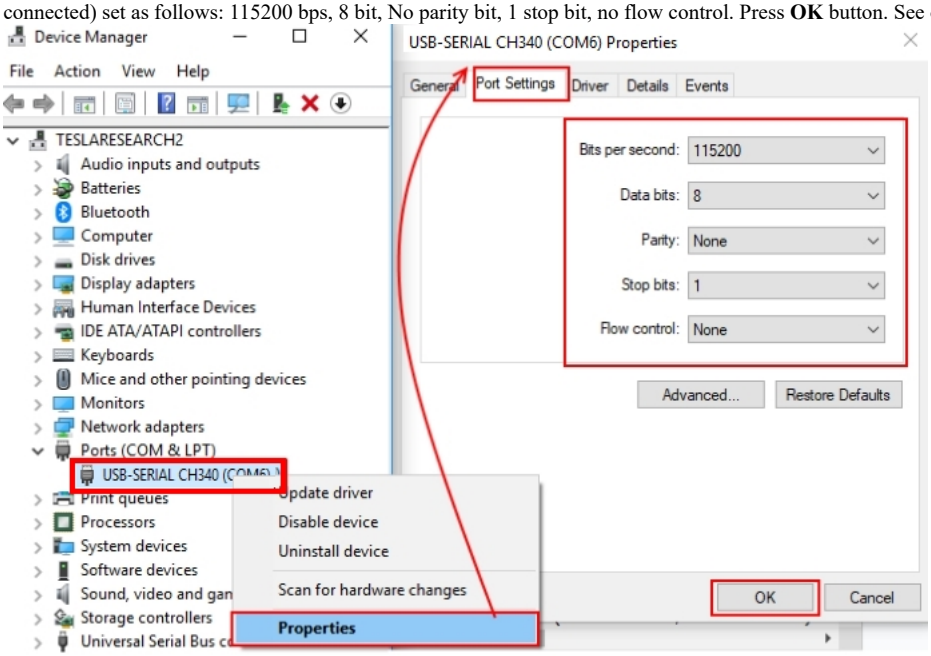

**UART TTL interface (serial port):**



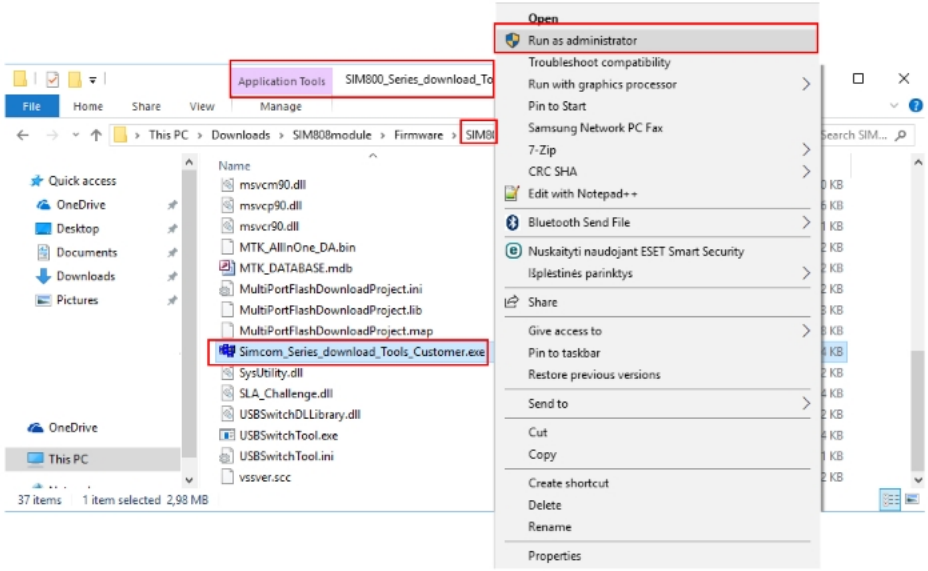

1. Download and install the **driver** for your USB to TTL converter. Read about it here.
2. Download and unzip the firmware upgrade tool **SIM800_Series_download_Tools_Customer_v1.10**. You can find it here.
3. Power up GSM GPRS GPS Bluetooth module with power supply adapter (5-26VDC 2A) through the DC044 interface.
4. 
5. Do wiring and plug in USB to TTL converter to PC USB port. See **2. Using USB to TTL converter** diagram above for reference.
6. Go to **Device Manager** on your PC and find out your COM port where the module connected. **We have COM6 and we are using Windows 10.**

7. 

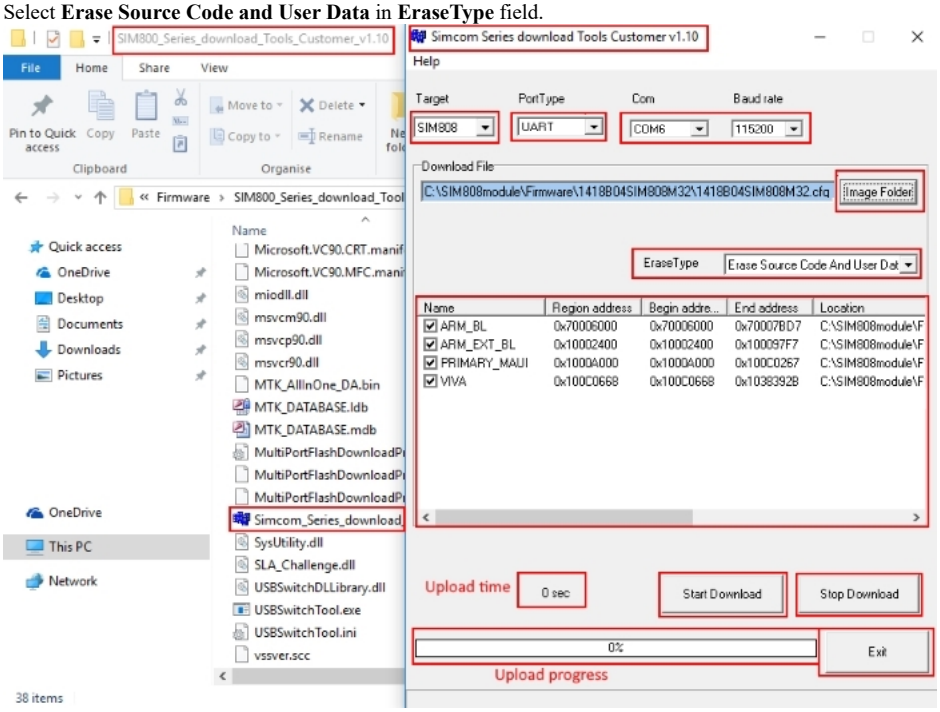
8. Go to **COM6 port settings** - click with right mouse button on COM6 port. Make sure that the COM6 port (serial port of your PC where the module connected) set as follows: 115200 bps, 8 bit, No parity bit, 1 stop bit, no flow control. Press **OK** button. See on the picture below.



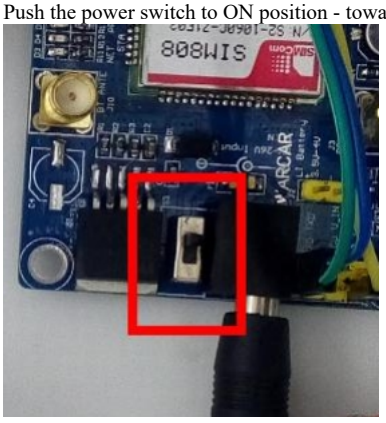

9. Opens propert
10. Open **SIM800_Series_download_Tools_Customer_v1.10**folder and find the file **Simcom_Series_download_Tools_Customer.exe**. Right-click on this file and select **Run as Administrator** and press **Yes** button**.**



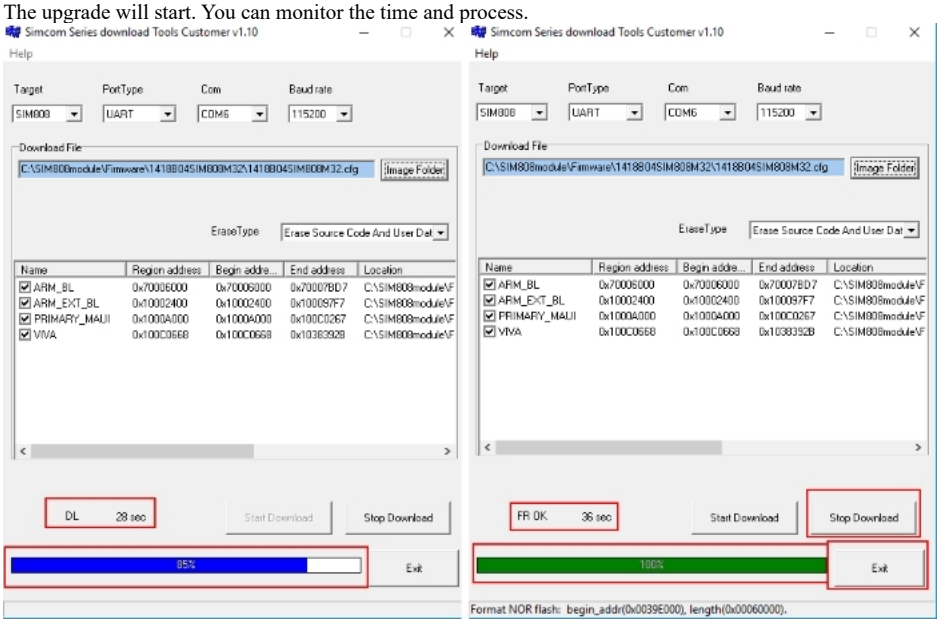

11.
12. Select **Target - SIM808** (you have SIM808 IC), **PortType - UART**, **Com - COM6**, **Baud rate - 115200**.
13. Press **Image Folder** button and find the location of firmware. Firmware configuration file has the **CFG** extension. We selected **1418B04SIM808M32.cfg** file.
14. Select **Erase Source Code and User Data** in **EraseType** field.

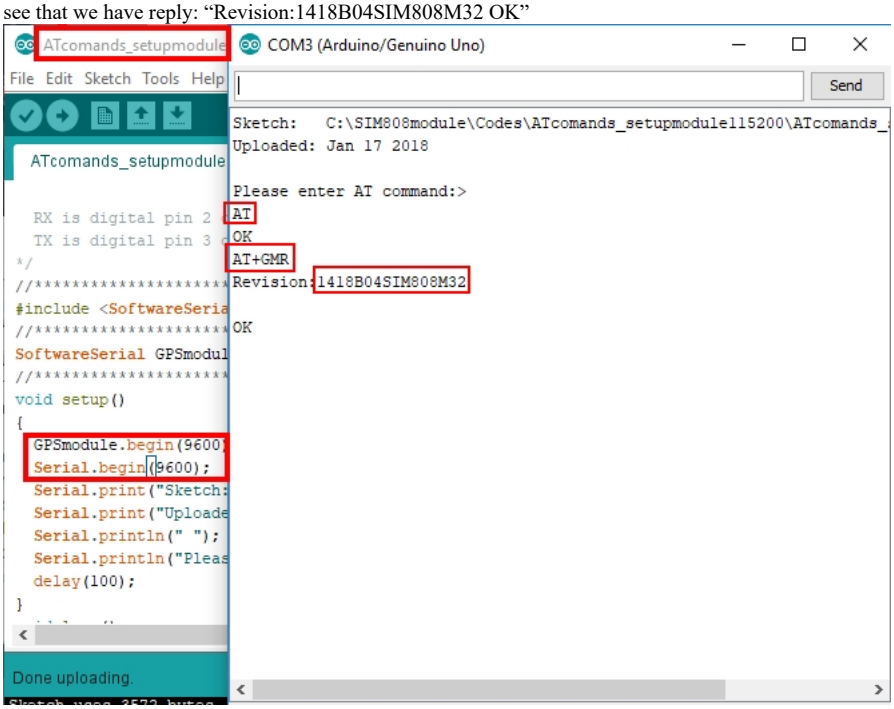



15.
16. Make sure that your PC and the module have a steady power supply.
17. Press **Start Download** button.
18. Push the power switch to ON position - towards the SIM808 IC. When the module powered on, the **LED power on indication (PWR)** will be ON.




19.
20. Press **Start button** for 2 seconds.

21. 
22. The upgrade will start. You can monitor the time and process.
23. 

24. Press **Stop Download** button when the bar will be green (100%) and FR OK.
25. Press **Exit** button.
26. Make sure that you follow the instruction precisely. When you finish upgrade you need to reset the module: unplug the module from PC USB port and power off.
27. You can also check that you module has a newer version. Follow the steps as described above in 1. Configuring the GSM GPRS GPS Bluetooth module with Arduino board or 2. Configuring the GSM GPRS GPS Bluetooth module *USB to TTL converter*. Use AT command "**AT+GMR**" for that, for example, you can see that we have reply: "Revision:1418B04SIM808M32 OK"
28. 


*Code*

**Making a CALL.** Don't forget to change in the code the phone number you want to call.To make the call, you use the *callSomeone()* function that uses the ATD command. *SIM808.println("ATD + +XXXXXXXXX;");* You need to replace the X's (highlighted in red) with the phone number you want to call. Don't forget to connect a microphone and earphones to make the call. In this code example, the call is hang up after 30 seconds, using the ATH command: *SIM808.println("ATH");* Hanging up after 30 seconds is not very useful, but it works well for an example. The idea is that you use the ATH command when an event is triggered. For example, connect a push button to the Arduino, that when pressed sends the ATH command to hang up the phone.

**Send SMS/Read SMS**. *AT+CMGF=1* - used to set module to SMS mode, *AT+CNMI=2,2,0,0,0* - used to set module to send SMS data to serial out upon receipt. The AT+CMGS command sends an SMS message to a GSM phone. AT+CMGS="+37122222222" the eneter text and end AT command with a ^Z, ASCII

code 26.

You can read more about Read CMS AT command **here**.

See the example in Basics: Project 053k SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2) - LED control by SMS
*Summary*

We learnt how to connect SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2) to Arduino board and use it. We successfully upgraded our module with latest firmware too.

Notes:

- It takes for about half a minute or one to read the data by the GPS module initially when you run it, so do not panic for this it's very usual.
- It happens in some case that it is unable to detect the data that might be the issue with antenna, so unplug the antenna( if it is detachable) and attach it again.
- If, code says "Check Connection", then you should definitely check it twice, before giving up. Also, sometimes interchanging the TX and RX pins is preferable and surprisingly works.

**Tips and Hacks**

There is a simple way of power on the SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2) without pressing the **POWKEY** - start button. There is D9 at the bottom right corner of the SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2). Just set HIGH the D9 pin (for example, attached to Arduino digital pin 9) for 1 second. See the *turnOnOFFbySoftware sketch* Then it will power the whole module. No need to press the **POWKEY** - start button any more. Make sure that you power up GSM GPRS GPS Bluetooth module with power supply adapter (5-26VDC 2A) DC044 interface or with V_IN pin (5-26VDC 2A) or with Lithium Ion battery 3,5-4VDC. If you want to turn off the SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2) just set LOW the D9 pin (for example, attached to Arduino digital pin 9) for 3 seconds. See the *turnOnOFFbySoftware sketch*

Wiring:

Arduino Uno -> SIM808 GSM GPRS GPS Bluetooth evolution board (EVB-V3.2)

Digital Pin 9 D9 pin

GND GND


**GPS Visualizer: Do-It-Yourself Mapping**

You can check the GPS data logged with GPS Visualizer, an online utility that creates maps and profiles from geographic data. It is free and easy to use, yet powerful and extremely customizable. Input can be in the form of GPS data (tracks and waypoints), driving routes, street addresses, or simple coordinates. Use it to see where you've been, plan where you're going, or quickly visualize geographic data (scientific observations, events, business locations, customers, real estate, geotagged photos, etc.).

GPS Visualizer can read data files from many different sources, including but not limited to: GPX (a standard format used with many devices and programs, including Garmin's eTrex, GPSMAP, Oregon, Dakota, Colorado, & Nüvi series), Google Earth (.kml/.kmz), Google Maps routes (URLs), FAI/IGC glider logs, Fugawi (.trk/.wpt), Furuno, Garmin Fitness (.fit), Garmin Forerunner (.xml/.hst/.tcx), Garmin MapSource/BaseCamp/HomePort (.gdb), Geocaching.com (.loc), Google Spreadsheets, IGN Rando (.rdn), iGO (.trk), Lowrance (.usr), Microsoft Excel, NMEA 0183 data, OziExplorer (.plt/.wpt), Suunto X9/X9i (.sdf), Timex Trainer, TomTom (.pgl), U-blox (.ubx), XML feeds, and of course tab-delimited or comma-separated text.

*Library*

- **SoftwareSerial** library included in Arduino IDE.
- The library has the following known limitations: If using multiple software serial ports, only one can receive data at a time;Not all pins on the Mega and Mega 2560 support change interrupts, so only the following can be used for RX: 10, 11, 12, 13, 14, 15, 50, 51, 52, 53, A8 (62), A9 (63), A10 (64), A11 (65), A12 (66), A13 (67), A14 (68), A15 (69);Not all pins on the Leonardo and Micro support change interrupts, so only the following can be used for RX: 8, 9, 10, 11, 14 (MISO), 15 (SCK), 16 (MOSI);On Arduino or Genuino 101 the current maximum RX speed is 57600bps; On Arduino or Genuino 101 RX doesn't work on Pin 13. You can read about it here.

*Sketch*

- See attachments on the begining of this project

**Other projects of Acoptex.com**

MEDIUM Basics: Project 083w Sipeed Maixduino board - Using PlatformIO IDE of Acoptex.com in Sipeed Maixduino 08-08-2019

EASY Basics: Project 083y Sipeed Maixduino board - Using MaixPy REPL interface and Micropython Editor of Acoptex.com in Sipeed Maixduino 07-08-2019

EASY Basics: Project 083x Sipeed Maixduino board - Uploading script with uPyLoader of Acoptex.com in Sipeed Maixduino 06-08-2019

MEDIUM Basics: Project 083e Sipeed Maixduino board - Uploading MaixPy of Acoptex.com in Sipeed Maixduino 04-08-2019

MEDIUM Basics: Project 083f Sipeed Maixduino board - Using MycroPython of Acoptex.com in Sipeed Maixduino 04-08-2019

jobs.published_at
jobs.viewed

SITE LINKS:
About Us
Contact Us
Privacy Policy
How to post project/information / or comment it

Post a new project
Terms and Conditions
Our Partners

Members A-Z
Boards/Modules/Shields A-Z
Sitemap
Donate