Welcome!
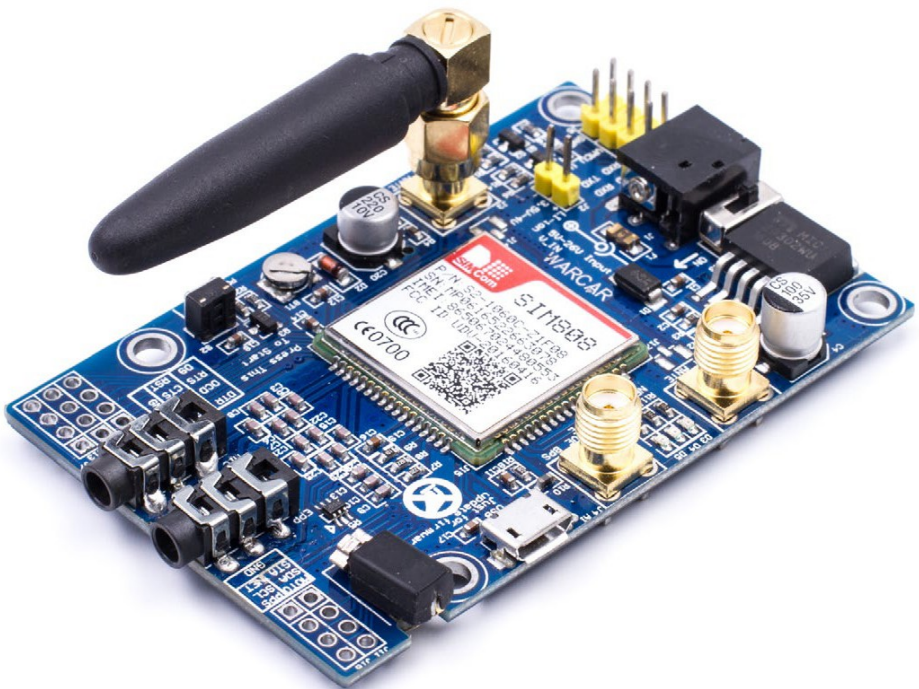
And thank you for purchasing our **AZ-Delivery SIM-808-Boards with GSM, GPRS and GPS**! On the following pages, we will take you through the first steps of the installation process to the first SMS and the first GPS tracking. We wish you a lot of fun!

http://flyt.it/SIM808

The **AZ-Delivery SIM-808-Board** comes with one external antenna for GSM / GPRS and GPS. Unlike the **AZ-Delivery GPRS shield,** it cannot be plugged into an Arduino UNO, but it can be connected to only with three cables.

The electricity supply is best provided by a 5V voltage-stabilizer with at least 10W power!

Overview of the most important information

- » SIM-808 chipset
- » Standard SIM card slot
- » External antennas for GSM/GPRS and GPS
- » Separate 3,5 mm jack connector for microphone and headphones
- » Serial communication via hard- and software serial
- » Power supply with 5V via external 5/2,5 mm connection (to turn on, press switch inwards)

The SIM-808 chip sometimes requires a current with strength of up to 2A, which the Arduino cannot afford or manage. The result can be, functional breaks and damages of the Arduino. It is therefore advised to always use a voltage stabilized 5V / 10W-power supply, as an external care application of the shield.

On the following pages, you will find information about
- » *Structure of the circuit*

And a guide for
- » *sending an SMS* and
- » *the query of your GPS coordinates*.

It is assumed by this tutorial, that you are familiar with uploading sketches to an Arduino and you know how to use the Serial Monitor!

Overview of all Links

Arduino library:
  » *https://github.com/MarcoMartines/GSM-GPRS-GPS-Shield*

SIM808 hardware documentation:
  » *http://simcom.ee/documents/?dir=SIM808*

Application programming interfaces:
  » Arduino IDE: *https://www.arduino.cc/en/Main/Software*
  » Web-Editor: *https://create.arduino.cc/editor*
  » Arduino extension for SublimeText:
    https://github.com/Robot-Will/Stino
  » Arduino extension "Visual Micro" for Atmel Studio or Microsoft Visual Studio:
    http://www.visualmicro.com/page/Arduino-for-Atmel-Studio.aspx

Interesting information from AZ-Delivery
  » Arduino compatible boards:
    https://az-delivery.de/collections/arduino-kompatible-boards
  » Arduino accessories:
    https://az-delivery.de/collections/arduino-zubehor
  » AZ-Delivery G+Community:
    https://plus.google.com/communities/115110265322509467732
  » AZ-Delivery on Facebook:
    https://www.facebook.com/AZDeliveryShop/

Structure of the circuit

For the serial communication between the board and the microcontroller, only three connections are necessary. The **TX**-pin of one device is connected to the **RX**-pin of the other, like the **RX**- to the **TX** pin. Since the board and the controller have different voltage sources, the masses **(GND)** must also be adjusted.

RX and **TX** are on the UNO's "**D0**" and "**D1**" pins. This connection, namely "**Hardware Serial**", is also used for the serial monitor of the Arduino IDE. It is possible, nevertheless, to emulate a so-called "**Software Serial**" via a library. Since the library used here, is also used for the **AZ-Delivery SIM-900-shield**, we should adjust the pin assignment to function for both modules: "**D7**" for **TX** and "**D8**" for **RX**.

As previously mentioned, the board should be powered by a **voltage stabilized 5V power supply unit** with at least **10W power**. When sending signals, a brief increase in the power consumption may occur, which cheap power supply units will not be able to handle. It is also not recommendable to get power supply from the Arduino as, in the worst case scenario, this could lead to its damage.
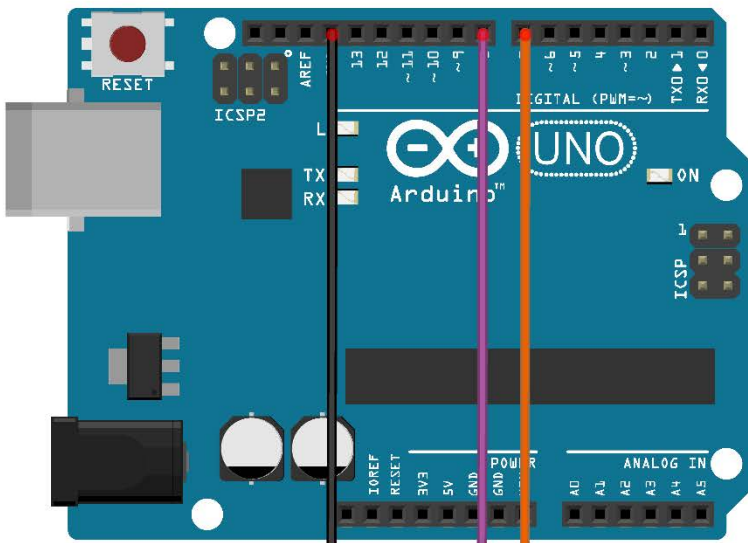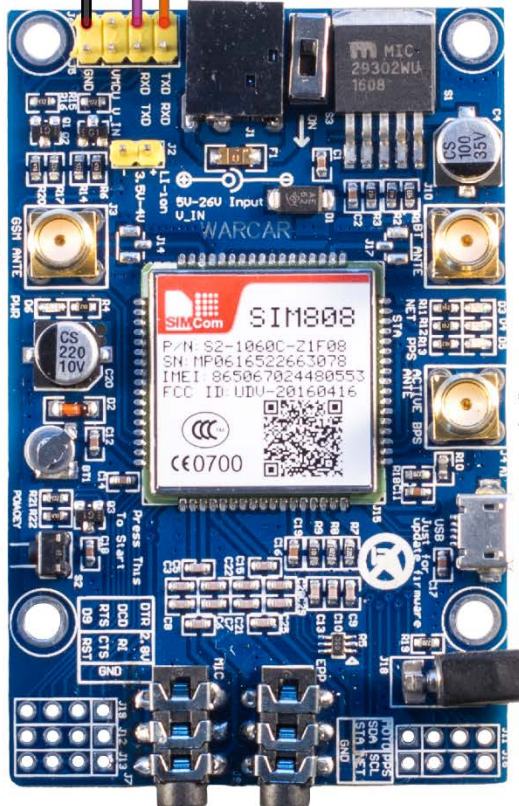
GND

RX (D8)

TX (D7)

GSM
Antenne

GPS
Antenne

Power

fritzing

The last step is to insert the SIM card. The slot for insertion can be found on the bottom side of the board. **But first, make sure that your card does not require a PIN!** Otherwise, you will have to deactivate it via the security settings of a mobile phone.

If you only have a Micro- or Nano-SIM card, you should clip it into the respective SIM card adapter from the kit that came with your SIM-808. When you insert it into the cover of the card slot, even those connections that are loose will be pressed and will have a stable contact with the shield.

Installing the library for the SIM808-Board

There are fewer libraries for the **SIM-808** compared to the **SIM-900** chipsets. But since both of them listen to the same AT command set, they are both compatible with one another. One of the most functionally comprehensive chipsets is the one from the developer Marco Martines. You can download it here:

» *https://github.com/MarcoMartines/GSM-GPRS-GPS-Shield/archive/GSMSHIELD.zip*

Unzip the"**GSM-GPRS-GPS-Shield-GS- MSHIELD**" folder into the **libraries**-directory of your sketchbook folder. You can then shorten the legibility of the name, e.g. "**GSMSHIELD**". After, you will have to adjust the pin assignment to the software serial. Open the "**GSM.cpp**" file, located in the directory and change the values in line **27** and **28** to:

```
25    //De-comment this two lines below if you have the
26    //second version og GSM GPRS Shield
27    #define _GSM_TXPIN_ 7
28    #define _GSM_RXPIN_ 8
```

To be prepared for future experiments, you should also define the

```
45    #define GSM_ON 9 // connect GSM Module turn ON to pin 77
46    #define GSM_RESET 6 // connect GSM Module RESET to pin 35
```

start and the reset pin, located in the "**GSM.h**" file, as follows:

Then close any instances of your Arduino IDE that may still be open, and start the program anew. Now from the library, you should be able to find, under *other*, your provided examples.

The first SMS

The classic "Hello World" should be included in the first SMS of our Arduino. Start the example sketch "**GSM_GPRSLibrary_SMS**".
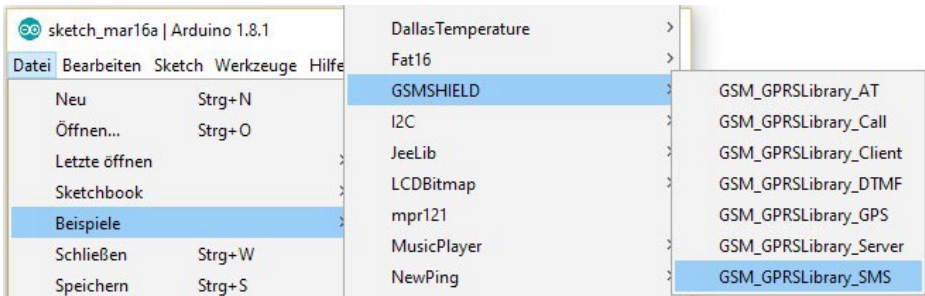
In order for something to be delivered, you must first activate the command that will send the SMS. Remove the comment characters from lines **40** and **41** and change the target numbers, as well as the message text, if necessary:

```
38      if(started) {
39          //Enable this two lines if you want to send an SMS.
40              if (sms.SendSMS("+49123456789", "Hello World!"))
41              Serial.println("\nSMS sent OK");
42      }
```

Then download the code to your UNO.

Now activate the **SIM-808-Board**, by connecting it to the power supply, pushing the switch that is next to it inwards and holding down the button on the side of the GSM antenna until the two red

LEDs, on the opposite side, light up.
When one of them is blinking fast, it shows that it is trying to connect to the mobile network. If the frequency of the blinking drops to a brief flash every three seconds, it means that the connection had been made and the board is ready.

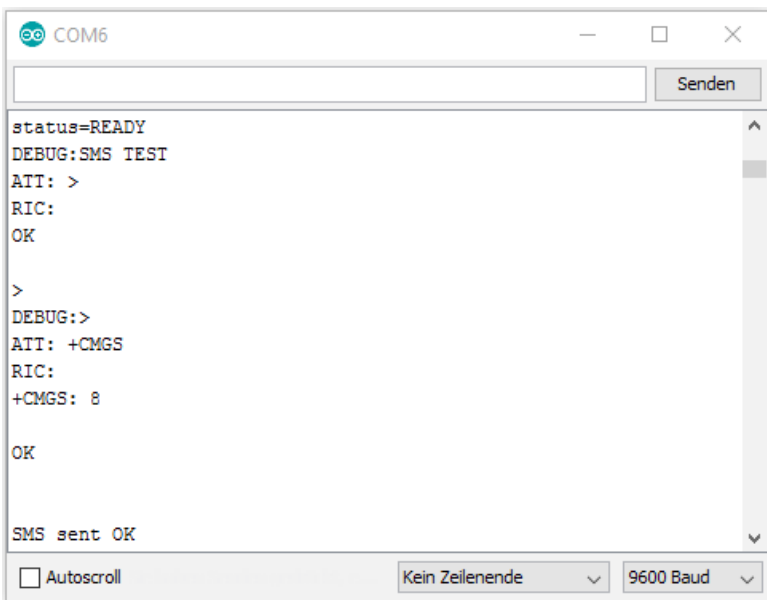Now start the Serial Monitor with a baud rate of **9600**.

```
GSM Shield testing.
ATT: OK
RIC: OK
```

The test phase begins with these lines. If that is over, then your mobile phone should receive a new "**Hello World!**" SMS. Meanwhile, the Serial Monitor displays messages about the successfully sent SMS as follows:

Read GPS coordinates

Capturing your position is also easy. To do this, start the example sketch "**GSM_GPRS- Library_GPS**" from the same directory. You do not have to change anything here, and you can load the code directly to the UNO. Then start the Serial Monitor and run the program until you see the following last lines (values will differ):
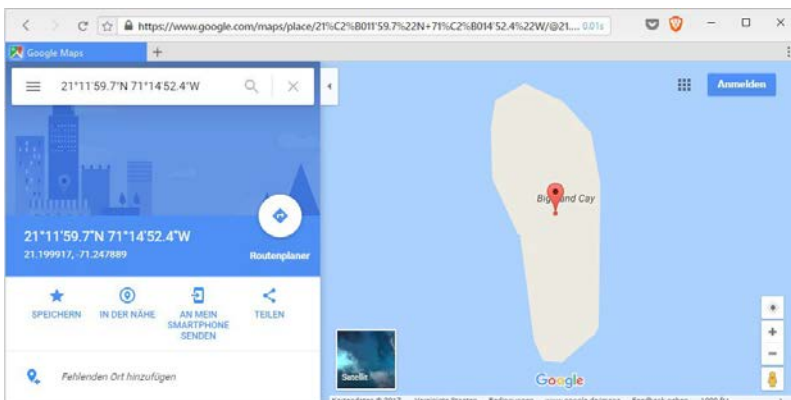
```
2111.597000
-7114.524000
73.600000
20170722120902.000
0.074080
```

The first two lines include the latitude and the longitude data. This data, unfortunately, is not interpretable. If, for example, you would like to check and verify the data on Google Maps, you should rewrite it as follows:

```
2111.597000 in 21°11'59.70"
-7114.524000 in -71°14'52.40" also:
```

Congratulations!

You have successfully completed this tutorial, sent your first SMS with an Arduino, and also found out from where the SMS was sent! Now it is time to learn and practice. It is best advised to look at the code you are using, to find out and know how to use the *send* and *read* commands. It is also worthwhile to look at the other examples that are in the library.

If you want your SIM-808-Board to be able to communicate with another, or you simply want to browse for other hardware, you will always find it in your online store at:

https://az-delivery.de

Enjoy!

Imprint

https://az-delivery.de/pages/about-us