GSM library

The GSM library is included with Arduino IDE 1.0.4 and later.

With the Arduino GSM Shield, this library enables an Arduino board to do most of the operations you can do with a GSM phone: place and receive voice calls, send and receive SMS, and connect to the internet over a GPRS network.

The GSM shield has a modem that transfers data from a serial port to the GSM network. The modem executes operations via a series of AT commands. The library abstracts low level communications between the modem and SIM card. It relies on the Software Serial library for communication between the moden and Arduino.

Typically, each individual command is part of a larger series necessary to execute a particular function. The library can also receive information and return it to you when necessary.

Library structure

As the library enables multiple types of functionality, there are a number of different classes.

- The *GSM* class takes care of commands to the radio modem. This handles the connectivity aspects of the shield and registers your system in the GSM infrastructure. All of your GSM/GPRS programs will need to include an object of this class to handle the necessary low level communication.
- Voice call handling, managed by the*GSMVoiceCall* class.
- Send/receive SMS messages, managed by the *GSM_SMS* class.
- The *GPRSClass* is for connecting to the internet.
- *GSMClient* includes implementations for a client, similar to the Ethernet and WiFilibraries.
- *GSMServer* includes implementations for a server, similar to the Ethernet and WiFilibraries. *NB : A number of network operators do not allow for incoming connections from the*

GSM class

This class prepares the functions that will communicate with the modem.

- GSM
- begin()
- shutdown()

GSMVoiceCallclass

Enables voice communicat through the modem. A microphone and speaker r to be added for full use.

- GSMVoiceCall
- getVoiceCallStatus()
- ready()
- voiceCall()
- answerCall()
- hangCall()
- retrieveCallingNumber()

GSM_SMS class

Facilitates sending and receiving Short Message Service (SMS) messages.

- GSM_SMS
- beginSMS()
- ready()
- endSMS()
- available()
- remoteNumber()

A number of utility classes such as*GSMScanner* and *GSMModem*

Ethernet library compatibility

The library tries to be as compatible as possible with the current Ethernet library. Porting a program from an Arduino Ethernet or WiFi library to an Arduino with the GSM Shield should be fairly easy. While it is not possible to simply run Ethernet-compatible code on the GSM shield as-is, some minor, library specific, modifications will be necessary, like including the GSM and GPRS specific libraries and getting network configuration settings from your cellular network provider.

Examples

There are two groups of examples for the GSM shield. There are examples to illustrate the possibilities of the shield, like how to send SMS messages and connect to the internet. There is also set of example tools that you can use to debug the functionality of the library and the hardware at lower level.

- Make Voice Call: get your shield to make phone calls from the Serial Monitor
- Receive Voice Call: check the status of the modem while getting voice calls
- Send SMS: use the Serial Monitor to type in SMS messages to different phone numbers
- Receive SMS: read SMS messages and prompt them to the Serial Monitor
- Web Client: download the content of a website to your Arduino board through GPRS
- Web Server: create a wireless web server through GPRS
- Xively Client: communicate to the Xively sensor backbone
- Xively Client String: communicate to the Xively sensor backbone

Tools

- Test Modem: read the IMEI of your modem
- Test GPRS: test the proper functionality of the GPRS network using your SIM card
- GSM Scan Networks: check for the available networks

- print()
- peek()
- flush()

GPRS class

This class is responsible fo including the files that are part of the library that involve TCP communicatic

- GPRS
- attachGPRS()

GSMClient class

The client class creates cli that can connect to server and send and receive data

- GSMClient
- ready()
- connect()
- beginWrite()
- write()
- endWrite()
- connected()
- read()
- available()
- peek()
- flush()
- stop()

GSMServer class

The Server class creates servers which can send da to and receive data from connected clients (prograr running on other compute or devices).

- GSMServer
- ready()

connects to.

Test Web Server: Create a webserver with your GSM shield

For additional information on the GSM shield, see the Getting Started page and theGSM shield hardware page.

endWrite()

read()

available()

stop()

GSMModem class

The GSMModem class facilitates diagnostic communication with the modem.

GSMModem

begin()

getIMEI()

GSMScanner class

The GSMScanner class provides diagnostic information about the network and carrier.

GSMScanner

begin()

getCurrentCarrier()

getSignalStrength()

readNetworks()

GSMPIN class

The GSMPIN class has utilities for communicatin with the SIM card.

GSMPIN

begin()

isPIN()

checkPIN()

checkPUK()

changePIN()

switchPIN()

checkReg()

GSMBand class

The GSMBand class provic
information about the
frequency band the model
connects to. There are als
methods for setting the ba

- GSMBand
- begin()
- getBand()
- setBand()