

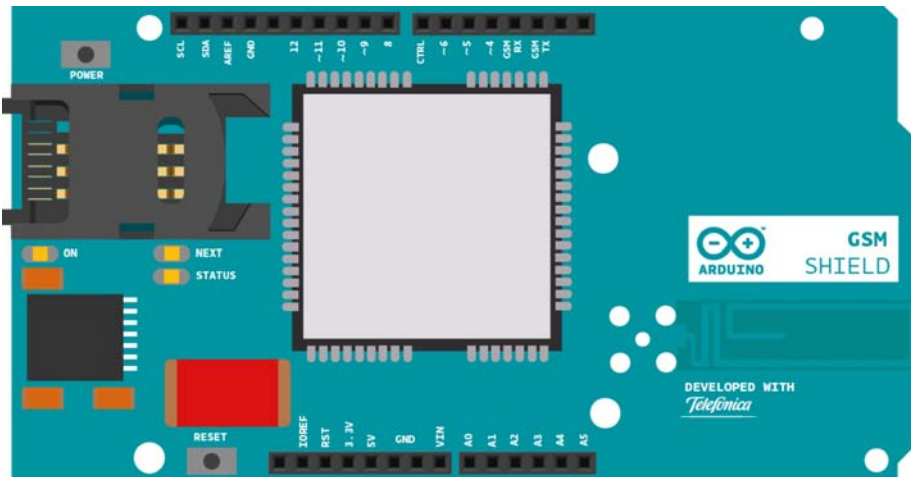
Receive SMS

This sketch waits for an SMS message and prints it to the serial monitor. It uses the GSM library of the Arduino GSM Shield and an active SIM card. To operate, the SIM card doesn't need a data plan.

Hardware Required

- Arduino or Genuino Board
- Arduino + Telefonica GSM/GPRS Shield ([//www.arduino.cc/en/Main/ArduinoGSMShield](http://www.arduino.cc/en/Main/ArduinoGSMShield))
- SIM card

Circuit



([//www.arduino.cc/en/uploads/Tutorial/GSMShield_ArduinoUno.jpg](http://www.arduino.cc/en/uploads/Tutorial/GSMShield_ArduinoUno.jpg))

image of the Arduino GSM Shield on top of an Arduino or Genuino board

Code

First, import the GSM library

```
#include <GSM.h>
```

SIM cards may have a PIN number that enables their functionality. Define the PIN for your SIM. If your SIM has no PIN, you can leave it blank :

```
#define PINNUMBER ""
```

Initialize instances of the classes you're going to use. You're going to need both the GSM and GSM_SMS class.

```
GSM gsmAccess;  
GSM_SMS sms;
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=2>)

Create a `char` array to hold the number that is sending the message :

```
char remoteNumber[20];
```

In `setup` , open a serial connection to the computer. After opening the connection, send a message indicating the sketch has started.

```
void setup(){  
  Serial.begin(9600);  
  Serial.println("SMS Messages Receiver");
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=3>)

Create a local variable to track the connection status. You'll use this to keep the sketch from starting until the SIM is connected to the network :

```
boolean notConnected = true;
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=4>)

Connect to the network by calling `gsmAccess.begin()` . It takes the SIM card's PIN as an argument. By placing this inside a `while()` loop, you can continually check the status of the connection. When the modem does connect, `gsmAccess()` will return `GSM_READY` . Use this as a flag to set the `notConnected` variable to `true` or `false` . Once connected, the remainder of `setup` will run.

```
while(notConnected)  
{  
  if(gsmAccess.begin(PINNUMBER)==GSM_READY)  
    notConnected = false;  
  else  
  {  
    Serial.println("Not connected");  
    delay(1000);  
  }  
}
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=5>)

Finish `setup` with some information to the serial monitor.

```
Serial.println("GSM initialized.");  
Serial.println("Waiting for messages");  
}
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=6>)

SMS messages are received by the modem. SIM cards have some memory space to store incoming SMS. The number of SMS the card can hold can be as few as 10, or as many as 200, depending on the SIM. You should check with your provider to determine how many your SIM can keep in memory.

In `loop()` , create a variable of type `char` to temporarily hold characters from any SMS received. Use `sms.available()` to check for the presence of any messages on the SIM :

```
void loop()  
{  
  char c;  
  if (sms.available())  
  {
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=7>)

If a SMS is available, retrieve the remote sender's number by calling `sms.remoteNumber(remoteNumber, 20)` . the `remoteNumber` argument is the `char` array you declared in the beginning of the sketch, it can be no longer than 20 characters. Send this number to the serial monitor.

```
Serial.println("Message received from:");  
  sms.remoteNumber(remoteNumber, 20);
```

```
Serial.println(remoteNumber);
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=8>)

It's possible to delete SMS messages by calling `sms.flush()`. Using `sms.peek()` it's possible to identify the message index number, which could be helpful for removal

The code below won't remove any from the SIM, but you could iterate through a `for` loop, or identify a specific index number to remove, instead of the dummy `#` used below

```
if(sms.peek()=='#')
{
    Serial.println("Discarded SMS");
    sms.flush();
}
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=9>)

To read a message, use `sms.read()`. Here, you'll store each character from the message into the variable `c` and print it out as it gets read.

```
while(c=sms.read())
    Serial.print(c);
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=10>)

Indicate the message is complete and remove it from memory with `sms.flush()`.

```
Serial.println("\nEND OF MESSAGE");
sms.flush();
Serial.println("MESSAGE DELETED");
}
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=11>)

Add a brief delay and close the loop.

```
delay(1000);
}
```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=12>)

Once your code is uploaded, open the serial monitor. With a phone, or other SMS enabled service, send a SMS to the number of your SIM. You should see the message print out on screen when it is received.

The complete sketch is below.

```
/*
SMS receiver

This sketch, for the Arduino GSM shield, waits for a SMS message
and displays it through the Serial port.
```

```
Circuit:
* GSM shield attached to and Arduino
* SIM card that can receive SMS messages
```

```
created 25 Feb 2012
by Javier Zorzano / TD
```

```
This example is in the public domain.
```

```
http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS
```

```
*/
```

```
// include the GSM library
#include <GSM.h>
```

```
// PIN Number for the SIM
#define PINNUMBER ""
```

```
// initialize the library instances
```

```

GSM gsmAccess;
GSM_SMS sms;

// Array to hold the number a SMS is retrieved from
char senderNumber[20];

void setup() {
  // initialize serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port only
  }

  Serial.println("SMS Messages Receiver");

  // connection state
  boolean notConnected = true;

  // Start GSM connection
  while (notConnected) {
    if (gsmAccess.begin(PINNUMBER) == GSM_READY) {
      notConnected = false;
    } else {
      Serial.println("Not connected");
      delay(1000);
    }
  }

  Serial.println("GSM initialized");
  Serial.println("Waiting for messages");
}

void loop() {
  char c;

  // If there are any SMSs available()
  if (sms.available()) {
    Serial.println("Message received from:");

    // Get remote number
    sms.remoteNumber(senderNumber, 20);
    Serial.println(senderNumber);

    // An example of message disposal
    // Any messages starting with # should be discarded
    if (sms.peek() == '#') {
      Serial.println("Discarded SMS");
      sms.flush();
    }

    // Read message bytes and print them
    while (c = sms.read()) {
      Serial.print(c);
    }

    Serial.println("\nEND OF MESSAGE");

    // Delete message from modem memory
    sms.flush();
    Serial.println("MESSAGE DELETED");
  }

  delay(1000);
}

```

[Get Code] (<http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveSMS?action=sourceblock&num=1>)

See Also

- [GSM \(<http://www.arduino.cc/en/Reference/GSMConstructor>\) Constructor](http://www.arduino.cc/en/Reference/GSMConstructor)
- [GSM.begin \(<http://www.arduino.cc/en/Reference/GSMBegin>\(\)\)](http://www.arduino.cc/en/Reference/GSMBegin)
- [GSM.shutdown \(<http://www.arduino.cc/en/Reference/GSMShutdown>\(\)\)](http://www.arduino.cc/en/Reference/GSMShutdown)

- [GSM_SMS](http://www.arduino.cc/en/Reference/GSMSMSConstructor) ([//www.arduino.cc/en/Reference/GSMSMSConstructor](http://www.arduino.cc/en/Reference/GSMSMSConstructor)) Constructor
- [beginSMS](http://www.arduino.cc/en/Reference/GSMSMSBeginSMS) ([//www.arduino.cc/en/Reference/GSMSMSBeginSMS](http://www.arduino.cc/en/Reference/GSMSMSBeginSMS)())
- [ready](http://www.arduino.cc/en/Reference/GSMSMSReady) ([//www.arduino.cc/en/Reference/GSMSMSReady](http://www.arduino.cc/en/Reference/GSMSMSReady)())
- [endSMS](http://www.arduino.cc/en/Reference/GSMSMSEndSMS) ([//www.arduino.cc/en/Reference/GSMSMSEndSMS](http://www.arduino.cc/en/Reference/GSMSMSEndSMS)())
- [available](http://www.arduino.cc/en/Reference/GSMSMSAvailable) ([//www.arduino.cc/en/Reference/GSMSMSAvailable](http://www.arduino.cc/en/Reference/GSMSMSAvailable)())
- [remoteNumber](http://www.arduino.cc/en/Reference/GSMSMSRemoteNumber) ([//www.arduino.cc/en/Reference/GSMSMSRemoteNumber](http://www.arduino.cc/en/Reference/GSMSMSRemoteNumber)())
- [read](http://www.arduino.cc/en/Reference/GSMSMSRead) ([//www.arduino.cc/en/Reference/GSMSMSRead](http://www.arduino.cc/en/Reference/GSMSMSRead)())
- [write](http://www.arduino.cc/en/Reference/GSMSMSWrite) ([//www.arduino.cc/en/Reference/GSMSMSWrite](http://www.arduino.cc/en/Reference/GSMSMSWrite)())
- [print](http://www.arduino.cc/en/Reference/GSMSMSPrint) ([//www.arduino.cc/en/Reference/GSMSMSPrint](http://www.arduino.cc/en/Reference/GSMSMSPrint)())
- [peek](http://www.arduino.cc/en/Reference/GSMSMSPeek) ([//www.arduino.cc/en/Reference/GSMSMSPeek](http://www.arduino.cc/en/Reference/GSMSMSPeek)())
- [flush](http://www.arduino.cc/en/Reference/GSMSMSFlush) ([//www.arduino.cc/en/Reference/GSMSMSFlush](http://www.arduino.cc/en/Reference/GSMSMSFlush)())
- [Arduino GSM Shield](http://www.arduino.cc/en/Main/ArduinoGSMShield) ([//www.arduino.cc/en/Main/ArduinoGSMShield](http://www.arduino.cc/en/Main/ArduinoGSMShield)) – Complete product description.
- [Getting started with the GSM Shield](http://www.arduino.cc/en/Guide/ArduinoGSMShield) ([//www.arduino.cc/en/Guide/ArduinoGSMShield](http://www.arduino.cc/en/Guide/ArduinoGSMShield)) – Get everything set up in minutes.
- [GSM library](http://www.arduino.cc/en/Reference/GSM) ([//www.arduino.cc/en/Reference/GSM](http://www.arduino.cc/en/Reference/GSM)) – Your reference for the GSM Library.
- [GSMExamplesMakeVoiceCall](http://www.arduino.cc/en/Tutorial/GSMExamplesMakeVoiceCall) ([//www.arduino.cc/en/Tutorial/GSMExamplesMakeVoiceCall](http://www.arduino.cc/en/Tutorial/GSMExamplesMakeVoiceCall)) - How to make a voice call with mic and speaker.
- [GSMExamplesReceiveVoiceCall](http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveVoiceCall) ([//www.arduino.cc/en/Tutorial/GSMExamplesReceiveVoiceCall](http://www.arduino.cc/en/Tutorial/GSMExamplesReceiveVoiceCall)) - The call is received and connected, the number that is calling is shown on serial monitor and then the call is hung up.
- [GSMExamplesSendSMS](http://www.arduino.cc/en/Tutorial/GSMExamplesSendSMS) ([//www.arduino.cc/en/Tutorial/GSMExamplesSendSMS](http://www.arduino.cc/en/Tutorial/GSMExamplesSendSMS)) - How to send an SMS entering number and text through the serial monitor.
- [GSMExamplesWebServer](http://www.arduino.cc/en/Tutorial/GSMExamplesWebServer) ([//www.arduino.cc/en/Tutorial/GSMExamplesWebServer](http://www.arduino.cc/en/Tutorial/GSMExamplesWebServer)) - A web server that when gets a request from a client sends back the value read on Analog inputs.
- [GSMExamplesWebClient](http://www.arduino.cc/en/Tutorial/GSMExamplesWebClient) ([//www.arduino.cc/en/Tutorial/GSMExamplesWebClient](http://www.arduino.cc/en/Tutorial/GSMExamplesWebClient)) - Connect to the Arduino.cc home and print the contents on the serial monitor window.
- [GSMToolsTestGPRS](http://www.arduino.cc/en/Tutorial/GSMToolsTestGPRS) ([//www.arduino.cc/en/Tutorial/GSMToolsTestGPRS](http://www.arduino.cc/en/Tutorial/GSMToolsTestGPRS)) – Tries to access the internet over GPRS with supplied APN and credentials.

Last revision 2015/08/13 by SM

Share



NEWSLETTER



©2017 Arduino [Copyright Notice](http://www.arduino.cc/en/Main/CopyrightNotice) ([//www.arduino.cc/en/Main/CopyrightNotice](http://www.arduino.cc/en/Main/CopyrightNotice)) [Contact us](http://www.arduino.cc/en/Main/ContactUs) ([//www.arduino.cc/en/Main/ContactUs](http://www.arduino.cc/en/Main/ContactUs))
[About us](http://www.arduino.cc/en/Main/AboutUs) ([//www.arduino.cc/en/Main/AboutUs](http://www.arduino.cc/en/Main/AboutUs)) [Careers](http://www.arduino.cc/Careers) ([//www.arduino.cc/Careers](http://www.arduino.cc/Careers))

<https://twitter.com/arduino> <https://www.facebook.com/official.arduino> <https://plus.google.com/+Arduino>
https://www.flickr.com/photos/arduino_cc <https://youtube.com/arduinoteam>

