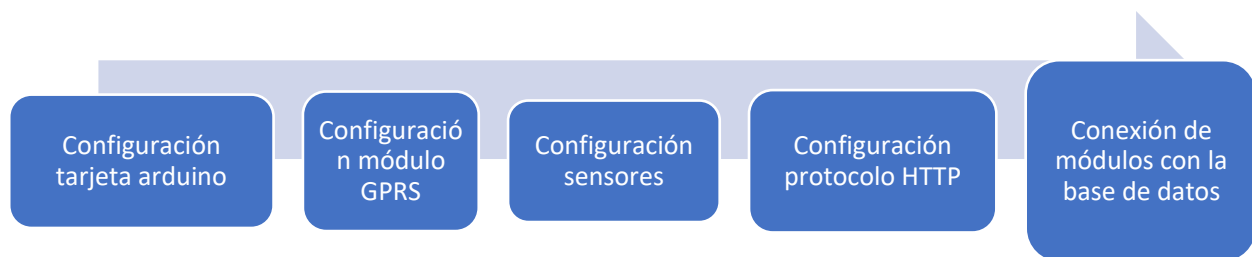


## APÉNDICE F

### 1. Programación de los módulos

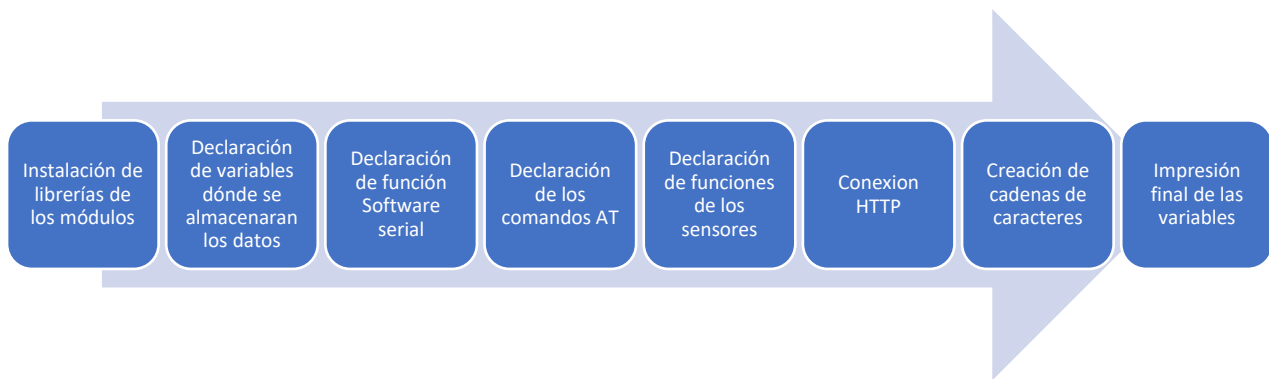
En esta sección se presenta el proceso de configuración de las dos grandes partes del sistema de adquisición de datos: la tarjeta arduino y módulo shield SIM900. Se desarrolló un programa en C(arduino) dónde se hizo la comunicación entre la placa, el módulo GRPS y la base de datos. Además, se describe cada uno de los ciclos de lectura de datos de cada uno de los sensores, tratamiento de variables para la transmisión y el envío de variables por medio del módulo GRPS por medio del protocolo http.



*Figura 1 Diagrama de bloques del proceso de programación de los módulos*

#### 1.1. Configuración Tarjeta arduino

La configuración del arduino constó del desarrollo de un código que permitiese a la placa leer todos los elementos que se encuentran conectados a los pines análogos y digitales, seguido de la agrupación de estos con los comandos AT que permitieran la comunicación con el módulo SIM900 (Para conocer más sobre la programación de arduino ver Anexo H).



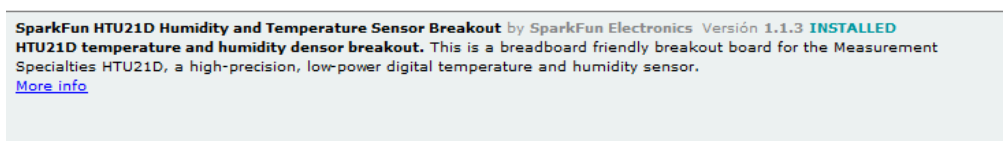
*Figura 2 Diagramas de bloques del proceso de programación del arduino*

Primero se realizó la instalación de las librerías del módulo weather shield y los sensores del weather kit de sparkfun (“SparkfunMPL3115A2.h” y “SparkfunHTU21D”). Seguido se realizó la declaración de los pines análogos y digitales en los que iban conectados cada uno de los sensores.



*Figura 3 SparkfunMPL3115A2.h en el administrador de librería de arduino*

Después, se realizó la declaración de variables globales para la recolección y transmisión de datos sensados (dirección del viento, velocidad del viento, humedad, temperatura, etc).



*Figura 4 SparkfunHTU21D en el administrador de librería de arduino*

### 1.1.1. Configuración módulo GPRS

Tras realizar la declaración de variables, se prosiguió a realizar la configuración del puerto serial para poder transmitir los datos al módulo GPRS. Por medio del comando `#include <SoftwareSerial.h>` se realizó la inserción de la librería que permite realizar comunicaciones por el puerto serial.

```
#include <SoftwareSerial.h>
SoftwareSerial gprsSerial(10,11);
```

*Figura 5 Librería y línea de comando inicial encargada de la comunicación con el puerto serial*

Después de verificar el funcionamiento, se empezó a realizar la configuración del módulo GPRS para transmitir por protocolo HTTP por medio de los comandos AT. (Ver comandos en el anexo A)

### 1.1.2. Configuración sensores

Teniendo la conexión del módulo GPRS establecida, se continua con la programación de la placa weather shield. Se inicia con la configuración del sensor de presión barométrica que tiene la placa, por medio de las instrucciones `setModeBarometer`, `setOversampleRate` y `enableEventFlags`. Además, se inicia el ciclo y configuración del sensor de humedad por medio de la instrucción `myHumidity.begin`. (Ver comandos en el anexo A)

Tras configurar inicialmente los módulos que vienen con la placa weather shield, se empiezan a realizar los ciclos void loop en los que se configuran los sensores del weather meter kit (anemómetro, pluviómetro, veleta de viento) (Ver comandos en el anexo A)

### **1.1.3. Configuración protocolo HTTP**

La configuración del protocolo HTTP se desarrolló al final, primero se inicializa el servicio http por medio de las líneas `gprsSerial.println("AT+HTTPINIT")` y `("AT+HTTPPARA=\"CID\",1")`.

Tras configurar el protocolo, se realiza hace el link con el index en el que se encuentra la conexión con la base de datos y dónde se utiliza el método get para enviar datos de un formulario a otro. (Ver comandos en el anexo B)

Habiendo realizado la configuración, se inicia la comunicación con el servidor HTTP por medio de comandos AT como `"AT+HTTPACTION=0"`, `"AT+HTTPREAD"`, `"AT+HTTPTERM"`.

## **1.2. Establecimiento de la conexión**

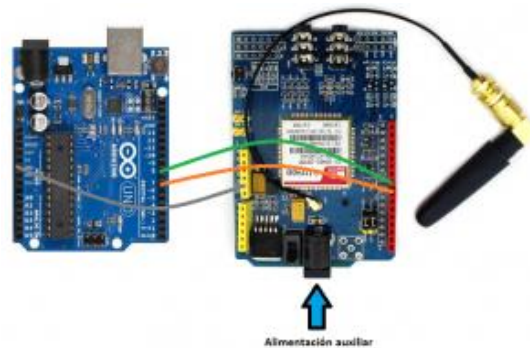
En esta sección se presenta el procedimiento que se siguió para la conexión de los módulos del sistema (Arduino MEGA y SIM900) y la constitución de la base de datos. Los puntos mencionados abarcan la explicación inicial del modelado de la base de datos, su estructura inicial, creación y presentación de los datos que se llevaran a esta. Además, se mostrará el procedimiento que se siguió para la presentación de los datos en una interfaz web y la forma en la que se alojan los archivos de esta en un hosting.

### **1.2.1. Conexión módulos con la base de datos**

Una de las partes vitales del proyecto es la conexión entre el Arduino MEGA, SIM900 y la base de datos. El desarrollo de la conexión entre estos módulos se divide en tres partes: conexión arduino MEGA con SIM900, conexión de SIM900 con la red móvil y finalmente la conexión entre los módulos y la base de datos.

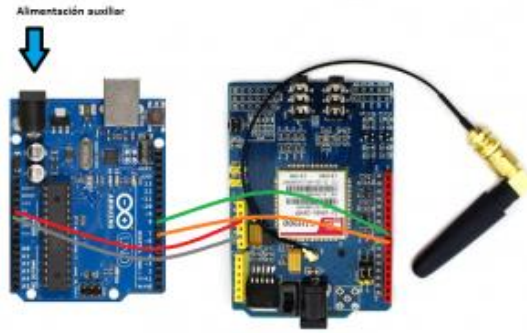
### 1.2.2. Conexión Arduino MEGA con módulo SIM900

Para la conexión de estos módulos se necesitan fuentes de alimentación externa, debido a que con conectar el arduino por USB será suficiente para encender el módulo, pero no para alimentar a la tarjeta también. Si se conecta la alimentación externa al shield GPRS y se alimenta el arduino con USB o una fuente aparte, sólo se tendrán que unir los pines 7 y 8 en el GRPS, 10 y 11 en el arduino Mega, y las tierras entre ambas tarjetas. Además, también se debe cambiar la instrucción en la definimos la instancia “Software Serial” que creamos y colocamos los pines respectivos. [42]



*Figura 6 Conexión entre el arduino y el módulo SIM900 [42]*

Ahora, si se conecta la alimentación externa al arduino a parte de las tres conexiones anteriores, se deberán unir los pines de 5V del arduino y del módulo GPRS.



*Figura 7 Conexión de los pines de 5V [42]*

### **1.2.3. Conexión de SIM900 con la red móvil**

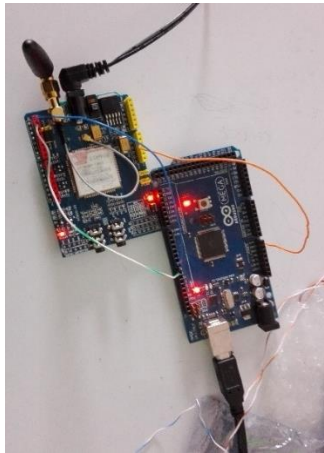
La conexión entre la SIM900 y la red móvil se desarrolló en el entorno de arduino, ya que tras realizar la declaración de librerías propias de los sensores de weather Meters y la declaración de variables en las que se almacenarían cada uno de los datos sensados, se realizó un void setup en el que se declaró la librería ‘<SoftwareSerial.h>’ con la que se abrió el puerto serial. Además, se realizó la declaración de los puertos.

Al declarar las primeras declaraciones, se incluyeron una lista de comandos AT que tienen diversas funciones:

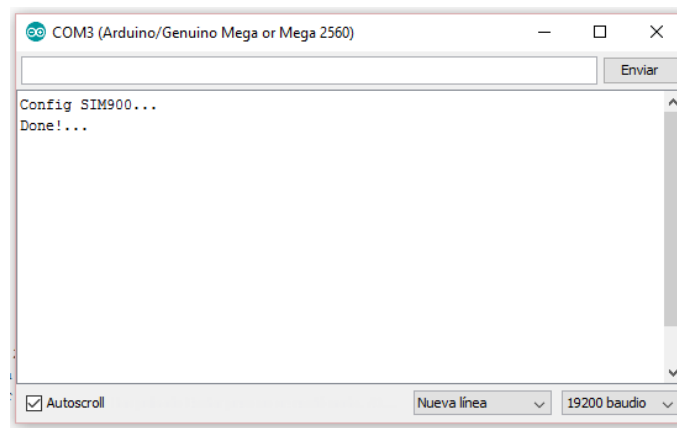
- `GprsSerial.begin(19200)`: Comunicación con puerto serial a 19200 baudios
- `Serial.println("Config SIM900")`: Comando que comprueba la configuración del módulo
- `GprsSerial.println("Done!")`: Comando de comprobación de finalización de la configuración del módulo
- `GprsSerial.println("AT")`: Comando AT de verificación de funcionamiento del módulo SIM900

- `GprsSerial.println("AT+CGATT?")`: Comando de comprobación del servicio GPRS
- `gprsSerial.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"")`: Establecimiento de la conexión con el servicio GPRS
- `gprsSerial.println("AT+SAPBR=3,1,\"APN\",\"internet.movistar.com.co\"")`:

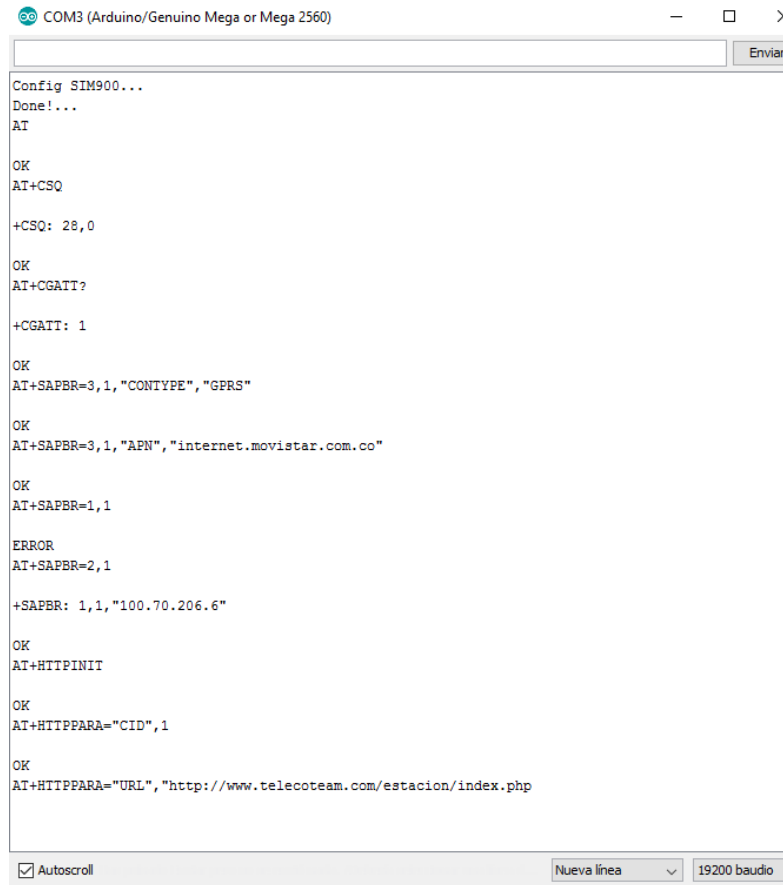
Comando AT dónde se digita el APN del operador móvil al que se está conectando el sistema, el cual en nuestro caso es Movistar.



*Figura 8 Conexión entre el arduino y SIM900*



*Figura 9 Comprobación del funcionamiento del módulo SIM900*



```
COM3 (Arduino/Genuino Mega or Mega 2560)
Config SIM900...
Done!...
AT
OK
AT+CSQ
+CSQ: 28,0
OK
AT+CGATT?
+CGATT: 1
OK
AT+SAPBR=3,1,"CONTYPE","GPRS"
OK
AT+SAPBR=3,1,"APN","internet.movistar.com.co"
OK
AT+SAPBR=1,1
ERROR
AT+SAPBR=2,1
+SAPBR: 1,1,"100.70.206.6"
OK
AT+HTTPINIT
OK
AT+HTTPPARA="CID",1
OK
AT+HTTPPARA="URL","http://www.telecoteam.com/estacion/index.php"
Autoscroll Nueva línea 19200 baudio
```

*Figura 10 Respuestas entregadas por el módulo SIM900*

### 1.3. Conexión final entre los módulos y la base de datos

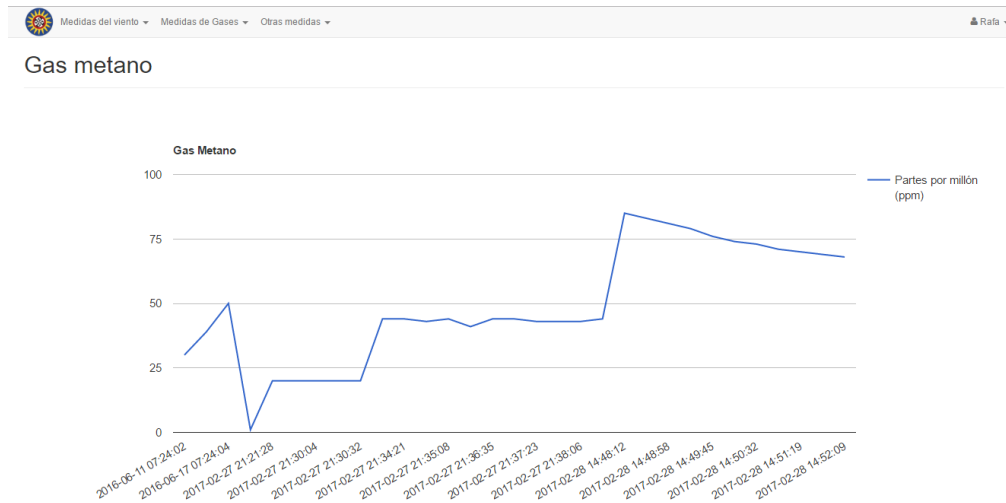
Habiendo realizado una satisfactoria conexión entre el Arduino MEGA y el módulo SIM900, se procedió a desarrollar la conexión entre dichos módulos y la base de datos.

Este proceso constó de una serie de cadenas que se desarrollaron en el código arduino, estas se igualaron a las variables en las que se almacenan los valores que se están sensando.

Tras realizar dichas cadenas, se desarrolló un código en php dónde por medio del método GET se pudiesen enviar las cadenas desarrolladas en el código c hacia un archivo en php. Una vez recibidos dichos datos en el código php, se desarrollaron nuevas cadenas que se igualaron a una



serie consultas en mysql, para así insertar en la base de datos los valores que se están exportando desde el otro programa. (Ver comandos en el anexo A)



*Figura 11 Datos graficados después de ser enviados desde el SIM900 hacia la base de datos*