

# Simple Arduino SD-Card GPS/NMEA Datalogger

---

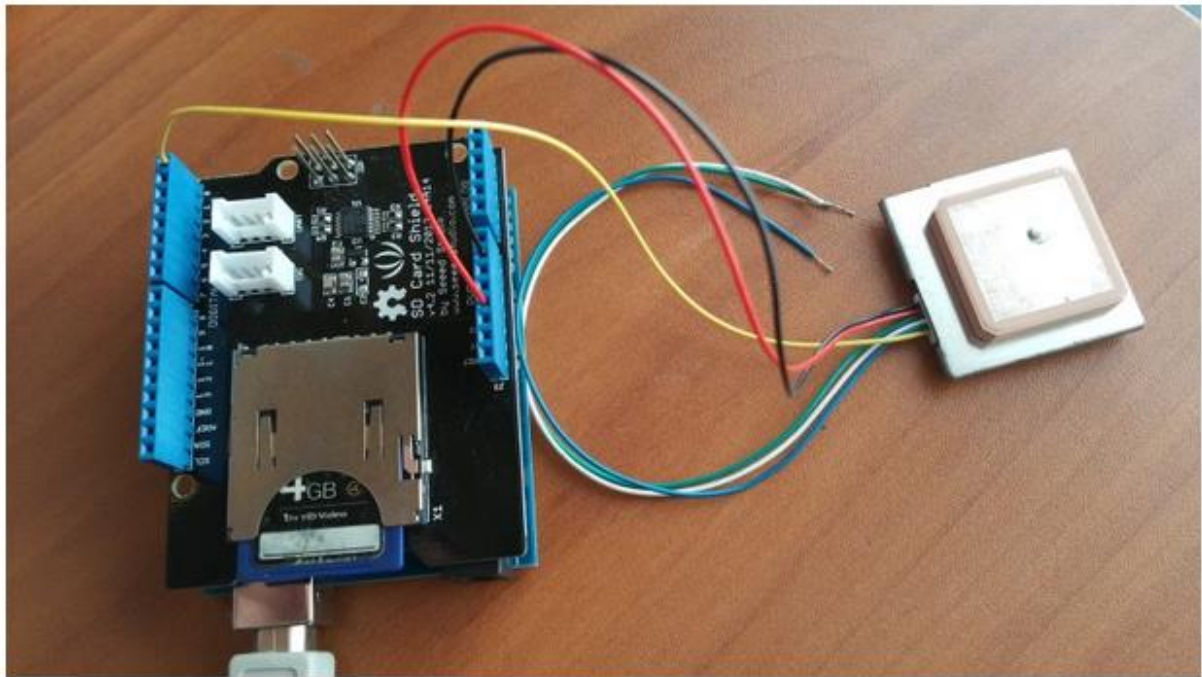
*This article is going to show how to build a minimalistic GPS datalogger. The GPS logger is based on the Arduino platform and stores raw NMEA sentences from pretty much any GPS module to a SD card.*

Posted on January 22, 2016 by KF5OBS

<http://jaunty-electronics.com/blog/2016/01/simple-arduino-sd-card-gpsnmea-datalogger/>

For a project I needed to log GPS information. I had various GPS modules and plenty of Arduinos laying around the lab. At first I intended for the Arduino to capture data from the GPS module, process it and then store it onto a SD card. However, I discarded that idea in favor of more flexibility and now use the arduino merely as pass-thru device for the raw GPS data.

Many GPS receivers spit out GPS information in a format called NMEA 0183, or NMEA for short. It is a standardized format for all kinds of navigation related instruments [1]. It's primarily used in the marine world. Nearly any GPS module on the market can spit out NMEA sentences. As stated before, originally I wanted the Arduino to extract position, speed, heading and other values from said NMEA sentences and store the result on the SD card. But storage is junk cheap these days so I decided to dumb the entire NMEA sentences directly on the SD card.



Very simple Arduino SD-Card GPS NMEA Logger

### Very simple Arduino SD-Card GPS NMEA Logger

The primary advantage of that idea is that any available bit of information is logged. Some information, like how many satellites are currently in view, may not be as interesting but again, storage is cheap. There's various programs out there that can visualize NMEA data so logging the raw NMEA sentences also had compatibility advantages. NMEA 0183 is electrically a serial bus. Therefore, this project is basically a serial logger. It takes anything coming in on the Arduino's hardware UART and dumps it unprocessed on the SD card. My GPS module uses a data rate of 4800 baud. If yours is different or you are logging data from a different device, keep in mind to adjust the code to the necessary data-rate.

The code is quite simple:

\*

GPS NMEA SD Card Data Logger

This example shows how to log data from three analog sensors to an SD card using the SD library.

The circuit:

GPS Module TX connected to the Arduino RX pin.

SD card attached to SPI bus as follows:

MOSI - pin 11  
MISO - pin 12  
CLK - pin 13  
CS - pin 4

Created: 01/22/2016  
by Sebastian Westerhold  
KF5OBS

This example code is in the public domain.

```
*/

#include
#include

// Set up necessary variables
const int chipSelect = 4;
char SerRead = 0;

// Setup routine
void setup() {

// Initiate Serial at 4800 baud
Serial.begin(4800);
while (!Serial) {
}

// Now initialize the SD card
Serial.print("Initializing SD card...");
if (!SD.begin(chipSelect)) {
Serial.println("Card failed, or not present");
return;
}
Serial.println("card initialized.");
}

// Main routine
void loop() {

// our buffer for serial data
String dataString = "";

// If there's serial data in the buffer, add to dataString
while(Serial.available() >0)
{
SerRead=Serial.read();
dataString += SerRead;
}

// If there's data in dataString, dump to SD card
if (dataString.length() >0) {

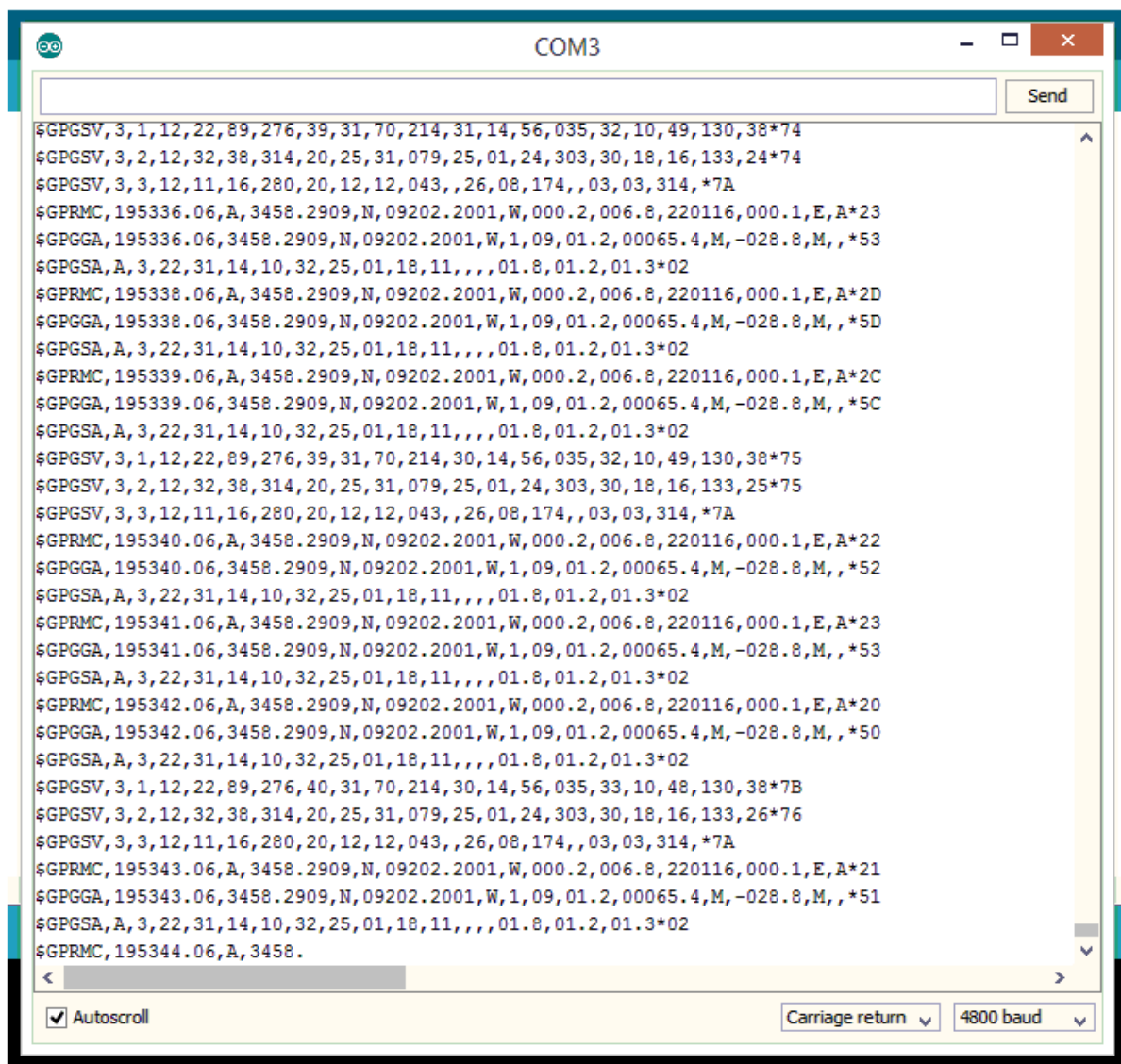
// Open file on SD Card
File dataFile = SD.open("NMEA.txt", FILE_WRITE);

// Dump data to file and serial port as well
if (dataFile) {
dataFile.print(dataString);
dataFile.close();
Serial.print(dataString);
}
}
```

```
// If there's any errors, so say via serial.
else {
Serial.println("error opening NMEA.txt");
}
}
}
```

You can download the project file here: [Arduino\\_GPS\\_NMEA\\_Logger](#)

For debugging purposes, the Arduino also relays the received serial information on its own serial port. This can be used for debugging purposes. Without a GPS connected, the arduino should echo everything set to it on its serial port and write the received data to a file called NMEA.txt on the SD card. If everything is connected correctly, you should get an output like this:



If the card can not be initialized or the file can not be opened / written to, the Arduino will throw error messages over the serial port. Please ensure the SD card is formatted as FAT32 or FAT16. I used the SD card V4.0 Shield [2] from seed. However, interfacing with a SD card is rather simple and SD card sockets are easy to come by.

From this point forward it is super easy to pair sensor data with position information. This could be useful for RC planes, model rockets or weather balloon payloads. You could easily make up your own custom NMEA sentence to include values from analog and digital inputs. The opportunities are endless. Enjoy experimenting!

Links and Sources:

[1] NMEA 0183, Wikipedia: [https://en.wikipedia.org/wiki/NMEA\\_0183](https://en.wikipedia.org/wiki/NMEA_0183)

[2] SD Card Shield V 4.0, SEED: [http://www.seeedstudio.com/wiki/SD\\_Card\\_shield\\_V4.0](http://www.seeedstudio.com/wiki/SD_Card_shield_V4.0)