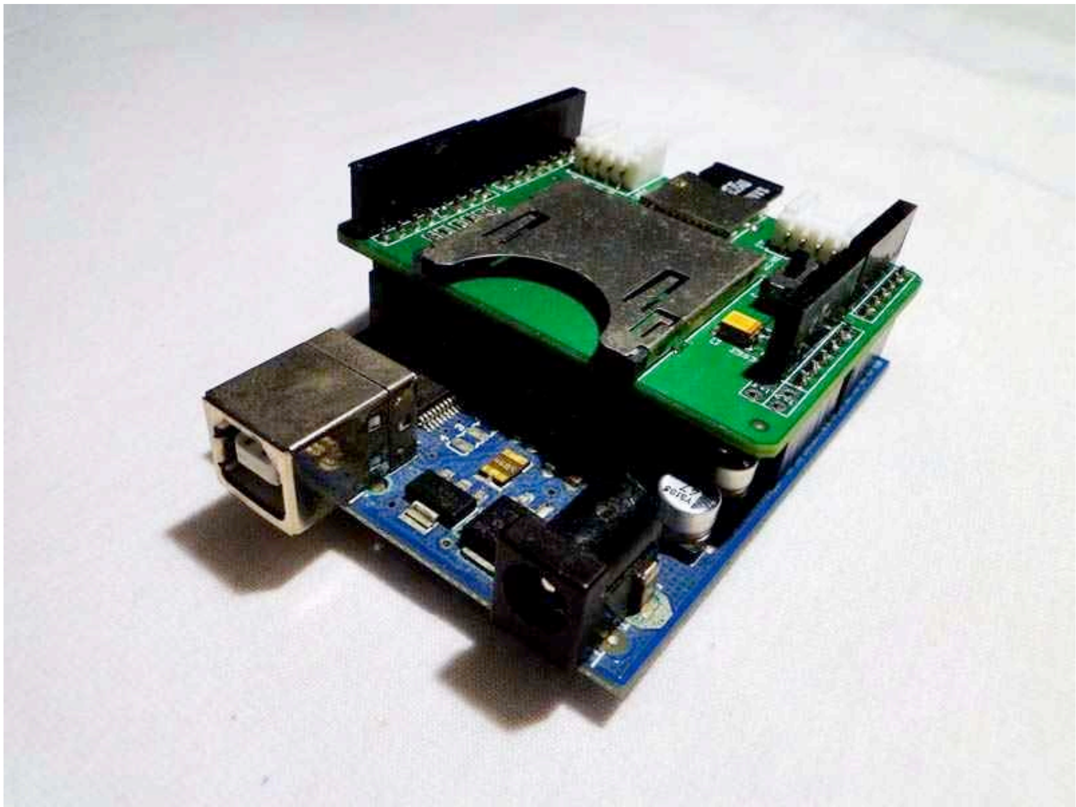
SD Card Arduino Shield Data Logger Tutorial



Note: Before going through this tutorial, please read getting Started tutorial, Click here to Download

Using the SD library to log data

This example shows how to use the SD card Library to log data from three analog sensors to a SD card. Please click here for more information on the SD library.

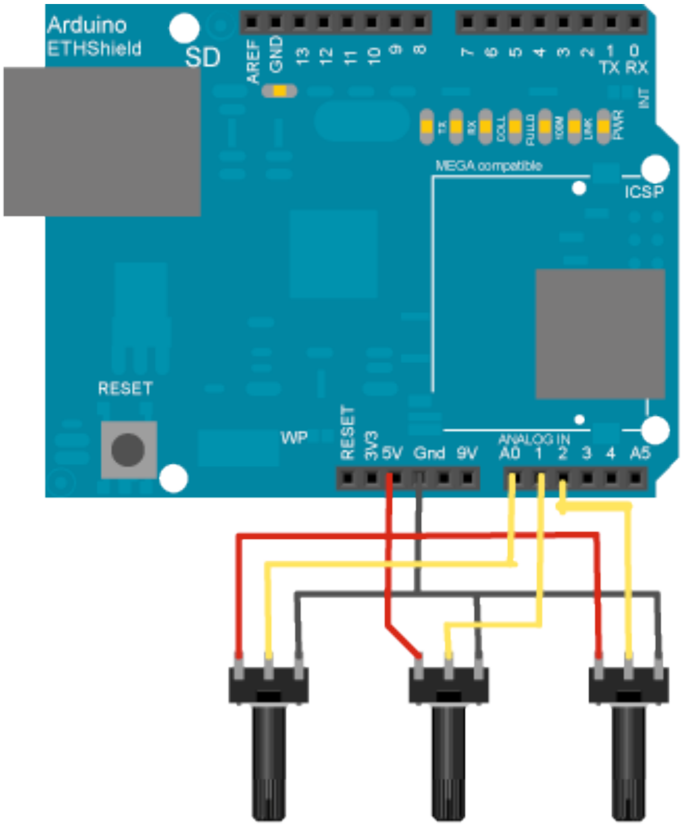
Hardware Required

- Three analog sensors
- Arduino board
- SD card board
- Formatted SD card

Using the SD library to log data

This example shows how to use the SD card Library to log data from three analog sensors to a SD card. Please click here for more information on the SD library.

Circuit (Hardware)



The code below is configured for use with an Ethernet shield, which has an onboard SD slot. In the `setup()`, call `SD.begin()`, naming pin 4 as the CS pin. This pin varies depending on the make of shield or board.

In the `loop()`, a String is created to hold the information from three analog sensors. The code iterates through the sensors, adding their data to the string.

Next, the file on the SD card is opened by calling `SD.open()`. Once available, the data is written to the card when`dataFile.println()` is used. The file must be closed with `dataFile.close()` to save the information.

```
/*
  SD card datalogger

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

 The circuit:
 * analog sensors on analog ins 0, 1, and 2
 * SD card attached to SPI bus as follows:
 ** MOSI - pin 11
 ** MISO - pin 12
 ** CLK - pin 13
 ** CS - pin 4

 created  24 Nov 2010
 updated 2 Dec 2010
 by Tom Igoe

 This example code is in the public domain.

 */

#include <SD.h>

// On the Ethernet Shield, CS is pin 4. Note that even if it's not
// used as the CS pin, the hardware CS pin (10 on most Arduino boards,
// 53 on the Mega) must be left as an output or the SD library
// functions will not work.
```

```
const int chipSelect = 4;

void setup()
{
  Serial.begin(9600);
  Serial.print("Initializing SD card...");
  // make sure that the default chip select pin is set to
  // output, even if you don't use it:
  pinMode(10, OUTPUT);

  // see if the card is present and can be initialized:
  if (!SD.begin(chipSelect)) {
    Serial.println("Card failed, or not present");
    // don't do anything more:
    return;
  }
  Serial.println("card initialized.");
}

void loop()
{
  // make a string for assembling the data to log:
  String dataString = "";

  // read three sensors and append to the string:
  for (int analogPin = 0; analogPin < 3; analogPin++) {
    int sensor = analogRead(analogPin);
    dataString += String(sensor);
    if (analogPin < 2) {
      dataString += ",";
    }
  }

  // open the file. note that only one file can be open at a time,
  // so you have to close this one before opening another.
  File dataFile = SD.open("datalog.txt", FILE_WRITE);

  // if the file is available, write to it:
  if (dataFile) {
    dataFile.println(dataString);
    dataFile.close();
    // print to the serial port too:
    Serial.println(dataString);
  }
  // if the file isn't open, pop up an error:
  else {
    Serial.println("error opening datalog.txt");
  }
}
```

Acknowledgement:
This document was written with the help of material from Arduino Italy and Seed Studio