



**LaACES  
Student  
Ballooning  
Course**

# The Ultimate GPS Logger Shield - SD

LaACES Student Ballooning Course



**LaACES  
Student  
Ballooning  
Course**

# SD Card

- The GPS shield has a microSD card slot. Use this to save the GPS and flight data
- Communication between the Mega and SD card uses SPI
- The microSD can be any capacity, but be aware of the limitations of the SD library being used
- When inserting the microSD, ensure that it latches. If it does not latch, no data will be logged
- Data saved in files on SD card

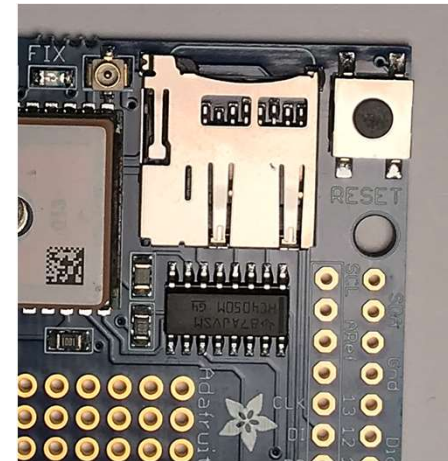


Figure 1: Shown is the Adafruit Ultimate GPS Logger Shield's microSD socket. It is next to the Reset button. When inserting a microSD card, it will latch once fully in



LaACES  
Student  
Ballooning  
Course

# Adafruit SD Library: Limitations and Characteristics

- For the MegaSat, the Adafruit SD library will be used. It can be found at <https://github.com/adafruit/SD>
- This library has limitations:
  - Despite SD card size, only ~2GB of data will be saved. If you attempt to save more, the data will be lost
  - File names must follow 8.3 format



LaACES  
Student  
Ballooning  
Course

# Adafruit SD Library: SPI

- SPI communication requires 4 lines: MISO, MOSI, clock, and chip select
- Chip select is hardwired on the Adafruit Shield to digital pin 10
- MISO, MOSI, and SCK are digital pins 50, 51, and 52 on the Mega but digital pins 12, 11, and 13 on the Shield
  - For this SD library, the pins can be declared in software. Other libraries may require jumps to connect pins 50 to 12, 51 to 11, and 52 to 13 in order to work



**LaACES  
Student  
Ballooning  
Course**

# Filenames

- File names must follow 8.3 format – FILENAME.EXT
  - Filenames can be shorter than 8 characters but cannot be longer
  - Common extension types are .txt and .csv
- If possible, use filenames that convey information
  - Ex: Use a timestamp for the filename
- Approved characters in Adafruit SD library filenames
  - Letters, numbers, \_, - (not all inclusive)
- Characters not allowed in filename by Adafruit library
  - Spaces, periods (not all inclusive)



# Extension Types

- Some common extension types are .txt and .csv
- CSV is a comma-separated values file
  - Figure 2 shows an example of what this data could look like
  - Excel automatically separates each row by the delimiter (,)
- TXT file can have the same data format, but user must manually tell Excel what the delimiter is

|    |   |    |   |                     |          |             |             |           |            |
|----|---|----|---|---------------------|----------|-------------|-------------|-----------|------------|
| 1  | ***Logging in file 01153040.csv created at          | A  | B   | C                   | D        | E           | F           | G         |            |
| 2  | START, Timestamp, Altitude, # Satellites, Fix Q     | 1  | ***Logging in file 01153040.csv created at 09/01/2019 15:30:40 for code version FlightC |                     |          |             |             |           |            |
| 3  | START, _____ No Fix! _____, 0.00, 0, 0, 5142, 0, 0  | 2  | START   | Timestamp           | Altitude | # Satellite | Fix Quality | FC Millis | LC Upper L |
| 4  | START, _____ No Fix! _____, 0.00, 0, 0, 10142, 0, 0 | 3  | START   | _____ No Fix! _____ | 0        | 0           | 0           | 5142      | 0          |
| 5  | START, _____ No Fix! _____, 0.00, 0, 0, 15142, 0, 0 | 4  | START   | _____ No Fix! _____ | 0        | 0           | 0           | 10142     | 0          |
| 6  | START, _____ No Fix! _____, 0.00, 0, 0, 20142, 0, 0 | 5  | START   | _____ No Fix! _____ | 0        | 0           | 0           | 15142     | 0          |
| 7  | START, 09/01/2019 15:31:04, 1204.10, 6, 1, 28116    | 6  | START   | _____ No Fix! _____ | 0        | 0           | 0           | 20142     | 0          |
| 8  | START, 09/01/2019 15:31:09, 1207.20, 6, 1, 33116    | 7  | START   | 9/1/2019 15:31      | 1204.1   | 6           | 1           | 28116     | 0          |
| 9  | START, 09/01/2019 15:31:14, 1206.80, 6, 1, 38116    | 8  | START   | 9/1/2019 15:31      | 1207.2   | 6           | 1           | 33116     | 0          |
| 10 | START, 09/01/2019 15:31:19, 1208.10, 6, 1, 43116    | 9  | START   | 9/1/2019 15:31      | 1206.8   | 6           | 1           | 38116     | 0          |
| 11 | START, 09/01/2019 15:31:24, 1209.80, 6, 1, 48116    | 10 | START   | 9/1/2019 15:31      | 1208.1   | 6           | 1           | 43116     | 0          |
|    |   | 11 | START   | 9/1/2019 15:31      | 1209.8   | 6           | 1           | 48116     | 0          |

Figure 2: This is an example of a .csv file. Left – opened in a text editor. Right – opened in Excel, the commas are used to separate the columns.



# Process of Writing Data to SD Card

- SD card communication initialized in setup()
  - `SD.begin(CS, MOSI, MISO, CLK);`
- Open/create SD file
  - `myFile = SD.open(filename, FILE_WRITE);`
- Write to SD card – same as printing to Serial Monitor
  - `myFile.println("This sentence will be written to my SD file");`
- Flush the data
  - `myFile.flush();`
- When finished, close the file
  - `myFile.close();`



**LaACES  
Student  
Ballooning  
Course**

# SdFat

- SdFat removes the size and filename limitations
- SPI connections must be hard wired.
  - MISO, MOSI, and SCK (50, 51, and 52) must be hardwired to pins (12, 11, and 13)
  - These hardwire connections are not currently implemented on the MegaSat
- The SdFat library can be found at <https://github.com/greiman/SdFat>





**LaACES**  
**Student**  
**Ballooning**  
**Course**

# Information and Image References

- Information
  - <https://learn.adafruit.com/adafruit-ultimate-gps-logger-shield/overview>
  - <https://www.arduino.cc/en/Reference/SDCardNotes>