
Adafruit SD Card Library Documentation

Release 1.0

Phillip Moyer

Mar 03, 2021

Contents

1	Dependencies	3
2	Usage Example	5
3	Sharing the SPI bus with other devices	7
4	Contributing	9
5	Documentation	11
6	Table of Contents	13
6.1	Simple test	13
6.2	adafruit_sdcard - SD card over SPI driver	14
6.2.1	Implementation Notes	14
7	Indices and tables	17
	Python Module Index	19
	Index	21

CircuitPython driver for SD cards. This implements the basic reading and writing block functionality needed to mount an SD card using `storage.VfsFat`.

CHAPTER 1

Dependencies

This driver depends on:

- [Adafruit CircuitPython 2.0.0+](#)
- [Bus Device](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).

CHAPTER 2

Usage Example

Mounting a filesystem on an SD card so that its available through the normal Python ways is easy.

Below is an example for the Feather M0 Adalogger. Most of this will stay the same across different boards with the exception of the pins for the SPI and chip select (cs) connections.

```
import adafruit_sdcard
import busio
import digitalio
import board
import storage

# Connect to the card and mount the filesystem.
spi = busio.SPI(board.SCK, board.MOSI, board.MISO)
cs = digitalio.DigitalInOut(board.SD_CS)
sdcard = adafruit_sdcard.SDCard(spi, cs)
vfs = storage.VfsFat(sdcard)
storage.mount(vfs, "/sd")

# Use the filesystem as normal.
with open("/sd/test.txt", "w") as f:
    f.write("Hello world\n")
```

Sharing the SPI bus with other devices

Important: If the same SPI bus is shared with other peripherals, it is important that the SD card be initialized before accessing any other peripheral on the bus. Failure to do so can prevent the SD card from being recognized until it is powered off or re-inserted.

CHAPTER 4

Contributing

Contributions are welcome! Please read our [Code of Conduct](#) before contributing to help this project stay welcoming.

CHAPTER 5

Documentation

For information on building library documentation, please check out [this guide](#).

6.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/sd_read_simpletest.py

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  import os
5  import busio
6  import digitalio
7  import board
8  import storage
9  import adafruit_sdcard
10
11  # The SD_CS pin is the chip select line.
12  #
13  #     The Adalogger Featherwing with ESP8266 Feather, the SD CS pin is on board.D15
14  #     The Adalogger Featherwing with Atmel M0 Feather, it's on board.D10
15  #     The Adafruit Feather M0 Adalogger use board.SD_CS
16  #     For the breakout boards use any pin that is not taken by SPI
17
18  SD_CS = board.SD_CS # setup for M0 Adalogger; change as needed
19
20  # Connect to the card and mount the filesystem.
21  spi = busio.SPI(board.SCK, board.MOSI, board.MISO)
22  cs = digitalio.DigitalInOut(SD_CS)
23  sdcard = adafruit_sdcard.SDCard(spi, cs)
24  vfs = storage.VfsFat(sdcard)
25  storage.mount(vfs, "/sd")
26
27  # Use the filesystem as normal! Our files are under /sd
```

(continues on next page)

(continued from previous page)

```
28
29 # This helper function will print the contents of the SD
30 def print_directory(path, tabs=0):
31     for file in os.listdir(path):
32         stats = os.stat(path + "/" + file)
33         filesize = stats[6]
34         isdir = stats[0] & 0x4000
35
36         if filesize < 1000:
37             sizestr = str(filesize) + " bytes"
38         elif filesize < 1000000:
39             sizestr = "%0.1f KB" % (filesize / 1000)
40         else:
41             sizestr = "%0.1f MB" % (filesize / 1000000)
42
43         prettyprintname = ""
44         for _ in range(tabs):
45             prettyprintname += "  "
46         prettyprintname += file
47         if isdir:
48             prettyprintname += "/"
49         print("{0:<40} Size: {1:>10}".format(prettyprintname, sizestr))
50
51         # recursively print directory contents
52         if isdir:
53             print_directory(path + "/" + file, tabs + 1)
54
55
56 print("Files on filesystem:")
57 print("=====")
58 print_directory("/sd")
```

6.2 adafruit_sdcard - SD card over SPI driver

CircuitPython driver for SD cards using SPI bus.

Requires an SPI bus and a CS pin. Provides readblocks and writeblocks methods so the device can be mounted as a filesystem.

- Author(s): Scott Shawcroft

6.2.1 Implementation Notes

Hardware:

- Adafruit MicroSD card breakout board+ (Product ID: 254)
- Adafruit Assembled Data Logging shield for Arduino (Product ID: 1141)
- Adafruit Feather M0 Adalogger (Product ID: 2796)
- Adalogger FeatherWing - RTC + SD Add-on For All Feather Boards (Product ID: 2922)

Software and Dependencies:

- Adafruit CircuitPython firmware for the ESP8622 and M0-based boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's Bus Device library: https://github.com/adafruit/Adafruit_CircuitPython_BusDevice

class `adafruit_sdcard.SDCard` (*spi, cs, baudrate=1320000*)
 Controls an SD card over SPI.

Parameters

- **spi** (*SPI*) – The SPI bus
- **cs** (*DigitalInOut*) – The chip select connected to the card
- **baudrate** (*int*) – The SPI data rate to use after card setup

Example usage:

```
import busio
import storage
import adafruit_sdcard
import os
import board

spi = busio.SPI(SCK, MOSI, MISO)
sd = adafruit_sdcard.SDCard(spi, board.SD_CS)
vfs = storage.VfsFat(sdcard)
storage.mount(vfs, '/sd')
os.listdir('/')
```

count ()

Returns the total number of sectors.

Returns The number of 512-byte blocks

Return type `int`

readblocks (*start_block, buf*)

Read one or more blocks from the card

Parameters

- **start_block** (*int*) – The block to start reading from
- **buf** (*bytearray*) – The buffer to write into. Length must be multiple of 512.

writeblocks (*start_block, buf*)

Write one or more blocks to the card

Parameters

- **start_block** (*int*) – The block to start writing to
- **buf** (*bytearray*) – The buffer to write into. Length must be multiple of 512.

`adafruit_sdcard.calculate_crc` (*message*)

Calculate the CRC of message[0:5], using a precomputed table in CRC_TABLE. :param bytearray message:
 Where each index is a byte

CHAPTER 7

Indices and tables

- `genindex`
- `modindex`
- `search`

a

adafruit_sdcard, 14

A

adafruit_sdcard (*module*), 14

C

calculate_crc() (*in module adafruit_sdcard*), 15

count() (*adafruit_sdcard.SDCard method*), 15

R

readblocks() (*adafruit_sdcard.SDCard method*), 15

S

SDCard (*class in adafruit_sdcard*), 15

W

writeblocks() (*adafruit_sdcard.SDCard method*),
15