

Volume

1

A-WIT TECHNOLOGIES INC.

---

... a passion for execution ...

# C Stamp SD Card Disk Drive Manual

---

Version 1.0

A-WIT TECHNOLOGIES INC.

# C Stamp SD Card Disk Drive Manual

---

© A-WIT Technologies Inc.  
Phone (800) 985-AWIT • Fax (800) 985-2948

---

# Table of Contents

Notices	1
Getting Support	2
C Stamp Program Example for SD Card Disk Drive	2
Downloading and Running Your Program	4
Developing Your Own Programs and Projects	4
SDD_open	5
SDD_read	5
SDD_write	6
Terms and Conditions	7



## Introduction

The A-WIT C Stamp SD Disk Drive provides a very simple way for C Stamp based projects to store data to and retrieve data from an SD card. Only four function calls are necessary; the user need not worry about the details of communicating with the SD card.

The A-WIT C Stamp SD card support is a hardware and software interface through which C Stamps can read and write sectors to SD cards using the SPI protocol, which is supported by all SD cards. This is a low-level API, which is interoperable with the A-WIT C Stamp FAT32 file system support.

The A-WIT C Stamp FAT32 file system support provides a simple API through which the users can read, write, and create files. The system can read and write 512-byte blocks.

## Notices

CSTAMP™ and CSTAMP™ Related Hardware Products, Software Products and Documentation are developed and distributed by A-WIT Technologies, Inc. All rights reserved by A-WIT Technologies, Inc. A-WIT SOFTWARE OR FIRMWARE AND LITERATURE IS PROVIDED “AS IS,” WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL A-WIT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY ARISING OUT OF OR IN CONNECTION WITH THE SOFTWARE OR FIRMWARE OR THE USE OF OTHER DEALINGS IN THE SOFTWARE OR FIRMWARE.

MPLAB C-18 and MPLAB C-18 Users Guide is reproduced and distributed by A-WIT Technologies, Inc. under license from Microchip Technology Inc. All rights reserved by Microchip Technology Inc. MICROCHIP SOFTWARE OR FIRMWARE AND LITERATURE IS

PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL MICROCHIP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY ARISING OUT OF OR IN CONNECTION WITH THE SOFTWARE OR FIRMWARE OR THE USE OF OTHER DEALINGS IN THE SOFTWARE OR FIRMWARE.

## Getting Support

If possible, please check the C Stamp website [www.c-stamp.com](http://www.c-stamp.com) under SUPPORT for any updates to documentation, changes, or notices that may have become available since your Installation CD was produced. If you continue to have any issues for which a solution is not found in the aforementioned website, please e-mail [tech\\_support@a-wit.com](mailto:tech_support@a-wit.com) for help.

## C Stamp Program Example for SD Card Disk Drive

The following program is a demo of how to use the C Stamp software infrastructure for SD card disk drive. It shows how a system could perform write and read operations on files.

The sample program is as follows:

```
// SD Card Disk Drive demo for A-WIT C Stamp
//
// This file contains a demo main routine that lets
// the user access an SD card via a serial connection

#include "CS110000.h"

// Prototypes
BYTE receive (BYTE buffer[], BYTE length);
void send (BYTE buffer[], BYTE length);

// Program entry point
void main (void)
{
    RAM SDD_FILE file[1];

    RAM BYTE prompt_1[] = "\n\rFile: ";
    RAM BYTE prompt_2[] =
        "\n\r(1) Read\n\r(2) Write\n\r";
```

```

RAM BYTE prompt_3[] = "\n\rData: ";
RAM BYTE newline[] = "\n\r";
RAM BYTE response1[128];
RAM BYTE response2[1];

int count;
BYTE command;

// Initialize memory for Disk Drive operations
SRAMInitHeap();

// Take commands from the user
while (TRUE){
    // Get filename
    send(prompt_1, sizeof(prompt_1));
    receive(response1, sizeof(response1));

    // Get action
    send(prompt_2, sizeof(prompt_2));
    receive(response2, 1);

    // Initialize with SD card on pin 13, and
    // Open the file
    if (!SDD_open(13, file, response1,
                 (response2[0] == '2'))) STOP();

    // Read
    if (response2[0] == '1'){
        send(newline, sizeof(newline));
        do{
            count = SDD_read(file, response1,
                             sizeof(response1));
            send(response1, count);
        }while (count > 0);
    }

    // Write
    else if (response2[0] == '2'){
        send(prompt_3, sizeof(prompt_3));
        count = receive(response1, sizeof(response1));
        send(newline, sizeof(newline));
        SDD_write(file, response1, count);
    }

    // Invalid

```

```

        else
        {
        }
    }
}

// Send data over the serial port with predefined
// settings
void send (BYTE buffer[], BYTE length)
{
    SEROUT(0, 0, 9.6, 0, 8, NOPAR, 0, buffer, length);
}

// Receive data (until [enter] is pressed) from the
// serial port with predefined settings
BYTE receive (BYTE buffer[], BYTE length)
{
    BYTE i = 0;
    BYTE c[1];

    do{
        SERIN(0, 0, 9.6, 8, 0, 0, c, 1, 0);
        if (c[0] != '\r'){
            buffer[i] = c[0];
            i = i+1;
        }
    }while ((c[0] != '\r') && (i < length));
    buffer[i] = '\0';

    return i;
}

```

## Downloading and Running Your Program

Download and start the program by pushing and letting go of the RESET button while pushing the START button. Then you can let go of the START button.

## Developing Your Own Programs and Projects

Now that you have successfully developed and run your program, it is easy to move on to more complex and elaborate projects and circuits of your own.

## Infrastructure for SD Card Disk Drive Reference

This chapter describes the elements of the C Stamp Infrastructure for SD Card Disk Drive, which consists of 3 functions. While this manual is generally written in a variable size font, code and its comments are written with a “fixed font”. Additionally, keywords, functions, and commands of the infrastructure are in “**bold**”. Generic syntax is in a “***bold and italics fixed font***”.

### SDD\_open

```
BIT SDD_open(BYTE cs, SDD_FILE file[], BYTE pathname[],  
             BIT create);
```

Initializes the Disk Drive system. Returns **TRUE** on success, **FALSE** on error. **cs** is the pin on the C Stamp to which the SD card’s Slave Select (SS) or Chip Select (CS) pin is connected. This system supports multiple files on multiple SD cards.

Opens a file for reading/writing, from an absolute **pathname**. **pathname** is the absolute pathname of the file to be opened. Must be defined in RAM (i.e. not as a constant).

**file** is an **SDD\_FILE** object that refers to a valid, declared file.

**pathname** is the complete path and filename of the file to be opened. No root directory prefix is required. For example, “mydir1/mydir2/myfile.txt”.

If **create** is **TRUE** and the specified file does not exist, it will be created.

### SDD\_read

```
int SDD_read(SDD_FILE file[], BYTE buffer[],  
             int length);
```

**SDD\_read** reads the requested number of bytes (or until end-of-file) from an open **file**, and returns the number of bytes read.

**file** is an **SDD\_FILE** object that refers to a valid, opened file (see **SDD\_open**).

**buffer** is an array of data to which data will be stored.

**length** is the number of bytes to read (until end-of-file or error).

## SDD\_write

```
int SDD_write(SDD_FILE file[], BYTE buffer[],  
              int length);
```

**SDD\_write** writes the requested number of bytes from the provided **buffer** to an open **file**, and returns the number of bytes written successfully.

**file** is an **SDD\_FILE** object that refers to a valid, opened file (see **SDD\_open**).

**buffer** is a **BYTE** array buffer from which data will be sourced.

**length** is the number of bytes to write.

## Terms and Conditions

### **Quality Assurance**

A-WIT has stringent quality control procedures in place to insure the best quality products.

### **90-Day Limited Warranty**

A-WIT Technologies, Inc warrants its products against defects in materials and workmanship for a period of 90 days. If you discover a defect, A-WIT Technologies, Inc. will, at its option, repair, replace, or refund the purchase price. After 90 days, products can still be sent in for repair or replacement, but there will be a \$10.00USD minimum inspection/labor/repair fee (not including return shipping and handling charges).

### **14-Day Money-Back Guarantee**

If, within 14 days of having received your product, you find that it does not suit your needs, you may return it for a refund. A-WIT will refund the purchase price of the product in the form of a check, excluding shipping/handling costs, once the product is received. This refund does not apply if the product has been altered or damaged. If you decide to return the products after the 14-day evaluation period, a 20% restocking fee will be charged against a credit.

### **Disclaimer**

Warranty does not apply if the product has been altered, modified, or damaged. A-WIT makes no other warranty of any kind, expressed or implied, including any warranty of merchantability, fitness of the product for any particular purpose even if that purpose is known to A-WIT, or any warranty relating to patents, trademarks, copyrights or other intellectual property. A-WIT shall not be liable for any injury, loss, damage, or loss of profits resulting from the handling or use of the product shipped.

### **How to Return a Product**

When returning, you must first e-mail [sales@a-wit.com](mailto:sales@a-wit.com) for a Return Merchandise Authorization number. No packages will be accepted without the RMA number clearly marked on the outside of the package. After inspecting and testing, we will return your product, or its replacement using the same shipping method used to ship the product to A-WIT within 30 days. In your package, please include a daytime telephone number and a brief explanation of the problem.

Please contact our Sales Department at [sales@a-wit.com](mailto:sales@a-wit.com) if you have any questions regarding our warranty policy or if you are requesting an RMA number.

