



APPLICATION NOTE 4074

Adding a FAT-Formatted SD Card to TINI's File System

Abstract: By using the SPI™ interface in the TINI® platform, Secure Digital (SD) cards can easily be accessed. This application note includes all of the necessary code for reading from and writing to an SD card that has been formatted with a FAT file system. That SD card will then provide extra file space which the TINIOS can access.

Introduction

The standard file system for any embedded system is usually small in size. The storage space of the Tiny Internet Interface (TINI) platform is no exception. TINI's file system resides in the on-board RAM, and is limited by the size of this RAM. Because of this memory constraint, it is important to provide alternatives for data storage. This application note describes how to install an additional file system that resides on a FAT-formatted Secure Digital (SD) card.

By accessing this alternate FAT-formatted file system, TINI users are no longer limited in the type and amount of information that they can process. Since TINI also uses its RAM for dynamic memory allocations during its runtime, using a supplemental FAT-formatted file system makes more of TINI's RAM available for running applications. Since the files stored on the SD card are stored in a FAT file system, the card can easily be read by other systems as well. A standard PC equipped with an SD card reader, for example, could read and write to the same card, thereby allowing easy transfer of files.

Once the SD card's file system has been mounted, there will be no apparent difference between a file that resides on the local file system and one located on the SD card. All interactions with files occur through the standard java.io classes and the com.dalsemi.fs.DSFile class.

Connecting the SD Card

Using SPI is the most convenient way to interface an SD card to the TINI platform on the [DS80C400](#) networked microcontroller. Application note 4068 "[Interfacing SD Cards with the TINI System](#)" details many of the specifics of this interface.

First consider the physical connections between the SPI in the TINI platform and the SD card. SPI requires six connections to the card. Wire the SD card to the DSTINIS400 board as described in **Table 1**.

Table 1. SD Card Connections to the [DSTINIS400](#) Sockets Evaluation Board

SD Card	DSTINIS400 SPI Header J21
CLK	Pin 1
DI	Pin 3
V _{CC}	Pin 4
DO	Pin 5
GND	Pin 6
CS	Pin 7

Building and Installing the Example

All of the code required for this application note is available for [download](#) (ZIP). The steps for building and installing the code are outlined below. The download also includes prebuilt binaries for TINI SDK version 1.17. If you want to use these prebuilt files, skip to step 9.

1. Download and install the latest [TINI 1.1 Software Development Kit](#).
2. Download and extract the source code for this example.
3. Open a command prompt window and change to the directory where you extracted the example. All of the commands for steps 4 through 10 should be entered at this prompt.

4. Set a `TINI_HOME` environment variable to the path where you installed the TINI SDK:

```
set TINI_HOME=C:\tini1.15
```

(Note: Steps 5 through 9 below can be performed by running the `build.bat` file included with the example.)

5. Extract the `SlushSrc.jar` file found in the TINI SDK to the `src` directory:

```
cd src
jar -xf %TINI_HOME%\src\SlushSrc.jar
cd ..
```

6. Extract the `MountCommand.java` and `UnmountCommand.java` files from the `OptionalSlushCommandsSrc.jar` file found in the TINI SDK:

```
cd src
jar -xf %TINI_HOME%\src\OptionalSlushCommandsSrc.jar
  com\dalsemi\slush\command\MountCommand.java
  com\dalsemi\slush\command\UnmountCommand.java
cd ..
```

7. Add the above `Mount` and `Unmount` commands to the list of the commands loaded when Slush boots. To do this, edit the `com\dalsemi\slush\Slush.java` file. Add the following lines to the `initializeShellCommands` method:

```
CommandInterpreter.addCommand("mount", new MountCommand());
CommandInterpreter.addCommand("unmount", new UnmountCommand());
```

8. Rebuild Slush, adding in the native library:

```
dir src\*.java /B /S /ON > files
javac -bootclasspath %TINI_HOME%\bin\tiniclasses.jar;%TINI_HOME%\bin\modules.jar -d bin @files
java -classpath %TINI_HOME%\bin\tini.jar BuildDependency -f bin
  -d %TINI_HOME%\bin\tini.db -o bin\slush_fat32.tbin -l -p
  %TINI_HOME%\bin\modules.jar -add FTPCLIENT;MAILTO;SPI400 -
  fake -noreflect -ref com.dalsemi.slush.util.FakeMainMaker -
  ref com.dalsemi.slush.command.SlushCommand
  -ref com.dalsemi.protocol.mailto.Handler -t 470100 -ref FATFS -x %TINI_HOME%\bin\owapi_dep.txt
```

9. Load the new `slush_fat32.tbin` file into your TINI by using `JavaKit`:

```
java -classpath %TINI_HOME%\bin\tini.jar;%classpath% JavaKit -flash 40 -m400
```

10. Log in to Slush and run the `Mount` command to add the FAT32 file system:

```
mount mnt FATFS
```

You will now see a directory called `mnt` in the root of your TINI's file system. This directory and its files can be accessed exactly like any of TINI's traditional files, however, all files and directories found here will be stored in the attached SD card.

Conclusion

By using TINI's built-in support for mounting additional file systems, TINI's storage capacity can be extended without much overhead. This process frees the applications developed for TINI from the constraints of the internal file system. Using an SD card also allows for easy sharing of data files and log files with other FAT-compatible systems.

References

The example code used by this application note can be obtained from Maxim at: files.dalsemi.com/microcontroller/app_note_software/an4074_sw.zip.

Further information on the SD media format can be obtained from the Secure Digital Association at: www.sdcard.org/.

The SD media breakout board used in this project can be ordered from Spark Fun Electronics at: www.sparkfun.com/commerce/product_info.php?products_id=204.

Further information on the FAT file system can be obtained from Microsoft at: www.microsoft.com/whdc/system/platform/firmware/fatgen.msp.

SPI is a trademark of Motorola, Inc.

TINI is a registered trademark of Maxim Integrated Products, Inc.

Application Note 4074: www.maxim-ic.com/an4074

More Information

For technical support: www.maxim-ic.com/support

For samples: www.maxim-ic.com/samples

Other questions and comments: www.maxim-ic.com/contact

Automatic Updates

Would you like to be automatically notified when new application notes are published in your areas of interest? [Sign up for EE-Mail™](#).

AN4074, AN 4074, APP4074, Appnote4074, Appnote 4074

Copyright © by Maxim Integrated Products

Additional legal notices: www.maxim-ic.com/legal