

AN1025: SD CARD EXAMPLE

APPLICATION NOTE

Tuesday, 22 September 2015

Version 1.0



Contents

APPLICATION NOTE	1
1 File Manager Project	3
1.1 Project Configuration	3
1.2 BGScript	3
2 Accessing the SD Card Content	4
2.1 Accessing the file manager with a browser	4
2.2 Accessing the file manager with REST API commands	4

1 File Manager Project

The *filemgr* example project included in the latest official version of the SDK (version 1.3.0 build 83) demonstrates how files can be uploaded, indexed, accessed, retrieved, and deleted to/from an SD card which is connected to a module via a web page based user interface. The project includes a BGScript source file, called *main.bgs*, an html file, *index.html*, a style sheet for the html file, *style.css*, the project configuration file, *project.xml*, and three image files used by *index.html*.

The html page, in conjunction with the style sheet and the image files, makes up the web page that provides the user interface allowing for files to be uploaded, indexed, accessed, retrieved, or deleted to/from the SD card connected to the SD card hardware interface on the WF121 module (SPI3 in the case of this example project). The BGScript file, "main.bgs" implements the application logic involved with the operation of the WF121 hardware.

1.1 Project Configuration

The project configuration includes the 4 tags: `<scripting>`, `<hardware>`, and `<files>`. The `<scripting>` tag is used to specify the main BGScript file to use for the project. In this example the main.bgs will including another script file, *api.bgs*, with import command.

```
<?xml version="1.0" encoding="UTF-8" ?>
<project>
  <scripting>
    <script in="main.bgs" />
  </scripting>
  <hardware>
    <uart channel="1" baud="115200" api="true" handshake="true" />
    <sdhc enable="1" spi_port="3" cs_port="3" cs_pin="6" />
  </hardware>
  <files>
    <file path="index.html"/>
    <file path="style.css"/>
    <file path="bluegiga.png"/>
    <file path="loader.gif"/>
    <file path="background.jpg"/>
  </files>
</project>
```

Project tags:

<hardware> tag is used to define the hardware configuration of the WF121.

<sdhc> tag within the hardware section configures the SD card interface. In order to use the SD Card interface, the `<sdhc>` settings shown above must be used.

<uart> tag configures the UART interface, which has to be connected and buffers read, for example using the WF121 GUI, when the API is enabled.

<files> Defines the HTML, CSS and image files used by the embedded HTTP server.

1.2 BGScript

The BGScript source file, *main.bgs*, is responsible for the application logic primarily concerned with the hardware operation of the WF121 and importing the REST API functionality from the *api.bgs* file. Specifically, *main.bgs* is responsible for turning WiFi on and setting the operating mode to Access Point when the module boots, then starting Access Point mode, and adding HTTP server accessible URL paths to the SD card and module's Flash. The *https_add_path* API method is used to add new URL paths to resources accessible by the HTTP sever. Such resources can be any of the following: the device flash, the SD card, or the API.

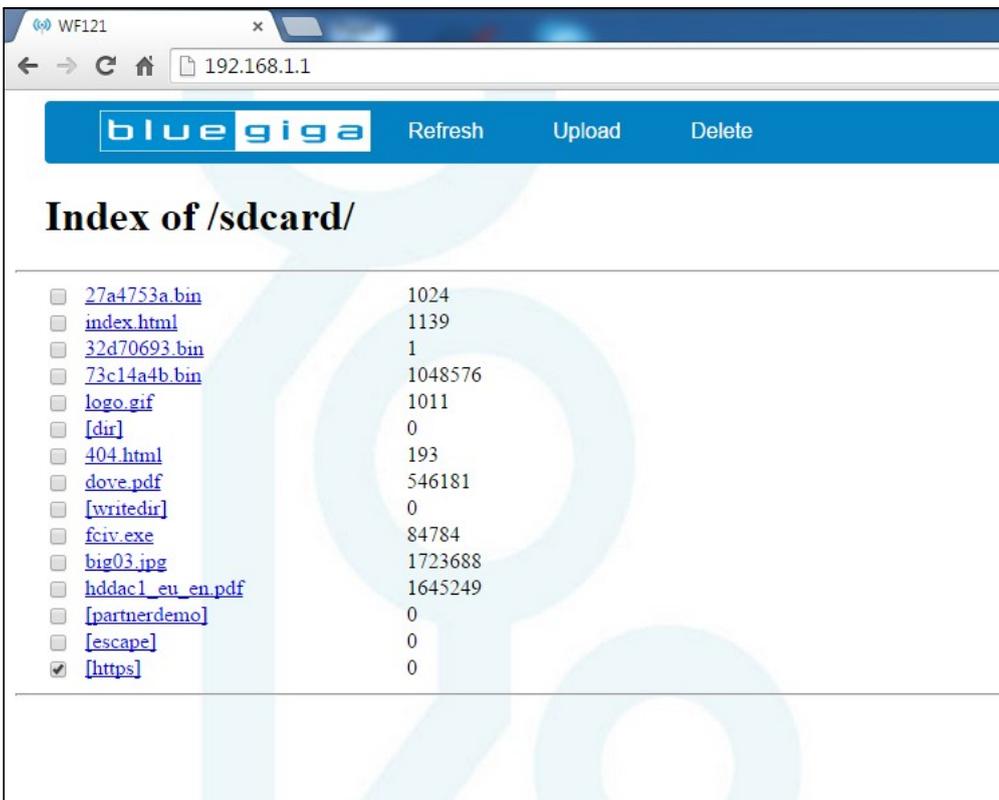
2 Accessing the SD Card Content

2.1 Accessing the file manager with a browser

The html source file, *index.html*, implements the web based interface allowing files to be uploaded, indexed, accessed, retrieved, or deleted to/from the SD card connected to the WF121 module.

After associating the remote device with the module's Access Point called "WF121 Example", browse with using Chrome,IE, Safari etc. to <http://192.168.1.1/>

The files currently stored on the SD card will be listed on the body of the page. New files, located on the device accessing the page, may be uploaded to the SD Card using the "Upload" button. Existing files on the SD Card can be deleted using the "Delete" button. Files listed under the index can be selected and viewed within the browser if they are of a format the browser being used can interpret such as html files.



2.2 Accessing the file manager with REST API commands

The web page is powered by Javascript code which exposes the HTTP server's built-in functionality meant to manipulate the files in the SD card filesystem. The below operations are available, assuming the usage of `call https_add_path(2, 7, "/sdcard")` in the script:

Read a file called 123.txt which exists in the root directory of the SD Card:

```
| GET /sdcard/123.txt
```

Write a file called 123.txt into the root directory of the SD Card

```
| PUT /sdcard/123.txt
```

Read the content of the directory /mydir:

```
|
```

```
| GET /sdcard/mydir?dir
```

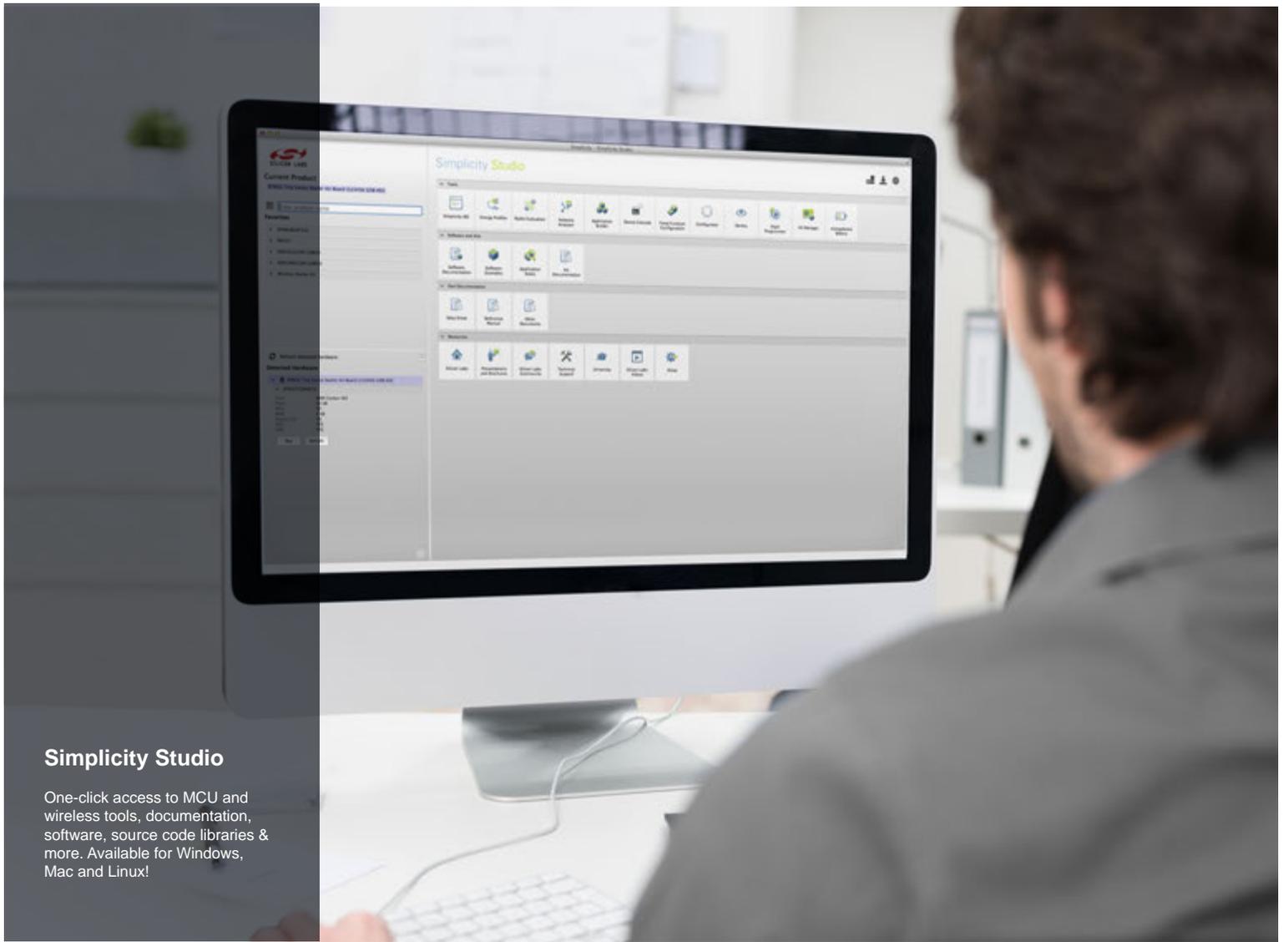
Delete a file called 123.txt which exists in the root directory of the SD Card:

```
| DELETE /sdcard/123.txt
```

Any of the above HTTP requests could be also issued for example by the "curl" program in a Linux system, like in the following example:

```
curl -v -X GET http://192.168.1.1/sdcard/123.txt
```

Since the HTTP server running in the module has no capability to create a directory, the solution in this case would be to intercept a custom HTTP request and adapt the BGScript to parse it and then launch the *call* *sdhc_fmkdir()* which is one of the BGAPI/BGScript commands in the SD memory card interface class meant to fully manipulate the filesystem of an attached SD card



Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



IoT Portfolio
www.silabs.com/IoT



SW/HW
www.silabs.com/simplicity



Quality
www.silabs.com/quality



Support and Community
community.silabs.com

Disclaimer

Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are not designed or authorized for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information

Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR®, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOModem®, Precision32®, ProSLIC®, Simplicity Studio®, SIPHY®, Telegesis, the Telegesis Logo®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



SILICON LABS

Silicon Laboratories Inc.
 400 West Cesar Chavez
 Austin, TX 78701
 USA

<http://www.silabs.com>