Xilinx Standalone Library Documentation

XilFFS Library v3.9

UG1032 (2018.2) June 6, 2018

XILINX.

Table of Contents

Overview

The Xilinx fat file system (FFS) library consists of a file system and a glue layer.
This FAT file system can be used with an interface supported in the glue layer.
The file system code is open source and is used as it is. Currently, the Glue layer implementation supports the SD/eMMC interface and a RAM based file system.
Application should make use of APIs provided in ff.h. These file system APIs access the driver functions through the glue layer.

The file system supports FAT16 and FAT32. The APIs are standard file system APIs. For more information, see the http://elm-chan.org/fsw/ff/00index_e.html.

Note

> The XilFFS library uses `Revision R0.10b` of the generic FAT filesystem module.

Library Files

The table below lists the file system files.

File	Description
ff.c	Implements all the file system APIs
ff.h	File system header
ffconf.h	File system configuration header – File system configurations such as READ_ONLY, MINIMAL, can be set here. This library uses _FS_MINIMIZE and _FS_TINY and Read/Write (NOT read only)

The table below lists the glue layer files.

File	Description
diskio.c	Glue layer – implements the function used by file system to call the driver APIs
ff.h	File system header
diskio.h	Glue layer header

Selecting a File System with an SD Interface

To select a file system with an SD interface:

1. Launch Xilinx SDK. Xilinx SDK prompts you to create a workspace.

2. Select **File** > **New** > **Xilinx Board Support Package**. The **New Board Support Package** wizard appears.

3. Specify a project name.

4. Select **Standalone** from the **Board Support Package OS** drop-down list. The **Board Support Package Settings** wizard appears.

5. Select the **xilffs** library from the list of **Supported Libraries**.

6. Expand the **Overview** tree and select **xilffs**. The configuration options for xilffs are listed.

7. Configure the xilffs by setting the `fs_interface = 1` to select the SD/eMMC. This is the default value. Ensure that the SD/eMMC interface is available, prior to selecting the `fs_interface = 1` option.

8. Build the bsp and the application to use the file system with SD/eMMC. SD or eMMC will be recognized by the low level driver.

Selecting a RAM based file system

To select a RAM based file system:

1. Launch Xilinx SDK. Xilinx SDK prompts you to create a workspace.

2. Select **File** > **New** > **Xilinx Board Support Package**. The b>New Board Support Package appears.

3. Specify a project name.

4. **Standalone** from the **Board Support Package OS** drop-down list. The **Board Support Package Settings** wizard appears.

5. Select the **xilffs** library from the list of **Supported Libraries**.

6. Expand the **Overview** tree and select xilffs. The configuration options for xilffs are listed.

7. Configure the xilffs by setting the `fs_interface = 2` to select RAM.

8. As this project is used by LWIP based application, select `lwip library` and configure according to your requirements. For more information, see the LwIP Library documentation.

9. Use any lwip application that requires a RAM based file system - TCP/UDP performance test apps or tftp or webserver examples.

10. Build the bsp and the application to use the RAM based file system.

Library Parameters in MSS File

XilFFS Library can be integrated with a system using the following code snippet in the Microprocessor Software Specification (MSS) file:

```
BEGIN LIBRARY
PARAMETER LIBRARY_NAME = xilffs
PARAMETER LIBRARY_VER = 3.9
PARAMETER fs_interface = 1
PARAMETER read_only = false
PARAMETER use_lfn = false
PARAMETER enable_multi_partition = false
PARAMETER num_logical_vol = 2
PARAMETER use_mkfs = true
PARAMETER use_strfunc = 0
PARAMETER set_fs_rpath = 0
END
```

The table below describes the libgen customization parameters.

Parameter	Default Value	Description
LIBRARY_NAME	xilffs	Specifies the library name.
LIBRARY_VER	3.9	Specifies the library version.
fs_interface	1 for SD/eMMC 2 for RAM	File system interface. SD/eMMC and RAM based file system are supported.
read_only	False	Enables the file system in Read Only mode, if true. Default is false. For Zynq® UltraScale+™ MPSoC devices, sets this option as true.
use_lfn	False	Enables the long file name (LFN) support, if true.
enable_multi_partitio	False	Enables the multi partition support, if true.

Parameter	Default Value	Description
num_logical_vol	2	Number of volumes (logical drives, from 1 to 10) to be used.
use_mkfs	True	Enables the mkfs support, if true. For Zynq UltraScale+ MPSoC devices, set this option as false.
use_strfunc	0	Enables the string functions (valid values 0 to 2). Default is 0.
set_fs_rpath	0	Configures relative path feature (valid values 0 to 2). Default is 0.
ramfs_size	3145728	Ram FS size is applicable only when RAM based file system is selected.
ramfs_start_addr	0x10000000	RAM FS start address is applicable only when RAM based file system is selected.

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support .

Solution Centers

See the Xilinx Solution Centers for support on devices, software tools, and intellectual property at all stages of the design cycle. Topics include design assistance, advisories, and troubleshooting tips.

Please Read: Important Legal Notices

Automotive Applications Disclaimer

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.