WICED Studio

WICED® - FRAM

Associated Part Family: CYW943907WAE4

Doc. No.: 002-19004 Rev. *B

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

Contents

About This Document

Purpose and Scope

This document provides instructions for how to use the WICED® F-RAM™ APIs to store data in the 32-KB F-RAM of the CYW943907WAE4 audio board. The F-RAM device can act as a non-volatile storage for storing critical configuration variables or the Device Configuration Table (DCT) itself. Using the APIs, you can save the data on your device between power cycles.

Note: This document applies to **WICED SDK 6.2.0** or higher and CYW943907WAE4 Audio board only.

Audience

This document is for software developers who are using the WICED Development System to create applications for secure embedded wireless networked devices.

Acronyms and Abbreviations

In most cases, acronyms and abbreviations are defined on first use.

For a comprehensive list of acronyms and other terms used in Cypress documents, go to www.cypress.com/glossary.

IoT Resources and Technical Support

Cypress provides a wealth of data at www.cypress.com/internet-things-iot to help you to select the right IoT device for your design, and quickly and effectively integrate the device into your design. Cypress provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates. Customers can acquire technical documentation and software from the Cypress Support Community website (community.cypress.com/).

Terminology

Bootloader	The initial program run when power is applied. Initializes hardware and decides which application to run.
DCT	Device Configuration Tables. Data stored to flash, both system DCT information and application information.
LUT	Look Up Table – in the WICED Multi-Application Framework, this is a simple directory where the system (App, DCT, Resources) is located in flash.
F-RAM	Ferroelectric Random-Access Memory

1 Introduction

1.1 F-RAM and CYW943907WAE4 Board Details

The CYW943907WAE4 Audio board has a 256-Kb Cypress SPI F-RAM FM25W256 connected to SPI Port 0 of the BCM4390x controller.

F-RAM offers an easy-to-use non-volatile memory for storing the configuration data or DCT data. F-RAM offers byte-wise access to the memory as well as near-infinite endurance of 100 trillion read/write cycles, making it ideal for storing local variables that need to be retained over several power cycles. The F-RAM device can be used for application developers to place the entire DCT, thereby freeing the SFLASH to remain accessible during the periodic DCT update intervals. The footprint on the audio board allows for upward migration up to 4-Mb SPI F-RAM.

1.2 F-RAM Read/Write Console Commands

The test console app has two commands that enable F-RAM write/read operations (`fram_read` and `fram_write`). The write/read commands use the underlying SPI driver to communicate with the F-RAM. F-RAM commands can be enabled at the build time by using the following build string:

```
test.console-CYW943907WAE4 ENABLE_SPI_0=1 download run
```

Console commands perform a write/read speed test into F-RAM with the following arguments:

`fram_write` `[address] [No. of Kbytes to write]` → `fram_write 0 10`

This command writes a predefined data pattern of 16 Kbytes into the F-RAM, starting at address 0.

`fram_read` `[address] [No. of Kbytes to read]` → `fram_read 1 8`

This command reads and compares 8 Kbytes data from location 1.

Note: All arguments are in hexadecimal format.

Console commands merely provide an estimate of SPI Write/read speeds in F-RAM. These commands perform a write (or read) operation 100 times and return the average time per operation. Writing and reading from different locations are expected to fail; you must ensure that these commands write to the memory before reading from it.

Figure 1 shows the expected result. The SPI bus can provide a sustained throughput of ~1.9 MB/s for both write and read operations.

```
Starting WICED v4.x-DEVELOPMENT
Platform CYW943907WAE4 initialised
Started ThreadX v5.8
Initialising NetX_Duo v5.10_sp3
Creating Packet pools
WLAN MAC Address : 00:A0:50:57:C6:7B
WLAN Firmware    : wl0: Apr 16 2018 13:50:53 version 7.15.168.111 (r687968) FW
ID 01-fa2066f0
WLAN CLM         : API: 12.2 Data: 9.10.74 Compiler: 1.31.3 ClmImport: 1.36.3
Creation: 2018-04-16 13:45:26
Console app
> fram_write 1 9
Average time taken for Write of 9 Kbytes = 5 ms
> fram_read 1 9
Average time taken for Read of 9 Kbytes = 5 ms
Data Write Read matched
>
```

Figure 1. F-RAM Write/Read Console Commands

Table 1 compares typical write read periods between the native SPI driver and the F-RAM driver in WAE4 audio board:

Data Packet Size	Performance with Native SPI Driver		Performance with Enhanced SPI Driver	
	Write period	Read period	Write Period	Read Period
6 KB	6 ms	6 ms	3 ms	3 ms
8 KB	8 ms	8 ms	5 ms	4 ms
16 KB	16 ms	16 ms	9 ms	9 ms
32 KB	32 ms	32 ms	18 ms	17 ms
Throughput	**~1 MB/s**		**~1.89 MB/s**	

Table 1. F-RAM Driver Performance

2 Customizing the F-RAM Driver

The console commands explained in Section 1 provide an estimate of write/read speeds in F-RAM devices over the SPI bus. This section explains how to use the F-RAM driver to perform application data storage in the F-RAM device. On the WAE4 audio board, F-RAM is included in the platform option through the following global define:

```
#define WICED_PLATFORM_INCLUDES_SPI_FRAM
```

The F-RAM driver uses the built-in SFLASH handle and defines a new interface on SPI port 0 of the WLAN chip. During the build operation, enabling SPI port 0 ensures that the F-RAM handle is defined correctly. Refer to the *spi_flash.mk* file for more details on enabling F-RAM driver permanently in the application for the WAE4 audio board.

Set `ENABLE_SPI_0: = 1`

Calling the `init_sflash` function for the F-RAM handle initializes the hardware interface.

```
hw_interface_t hw_interface = INTERFACE_4390x_SPI_0;

init_sflash( &fram, &hw_interface, 1);
```

Use the sflash_write and sflash_read functions to write/read to the F-RAM:

- `int sflash_write ( const sflash_handle_t* handle, unsigned long device_address, const void* data_addr, unsigned int size )`
  - `const slash_handle* handle:= pointer to F-RAM handle`
  - `unsigned long device_address:= physical F-RAM memory address in Hexadecimal (16-bit address)`
  - `const void* data_addr:= pointer to array containing data to be written to F-RAM`
  - `unsigned int size:= size of the data to be written in the F-RAM`
- `int sflash_trsf ( const sflash_handle_t* handle, unsigned long device_address, const void* data_addr, unsigned int size )`
  - `const slash_handle* handle:= pointer to F-RAM handle`
  - `unsigned long device_address:= physical F-RAM memory address in Hexadecimal (16-bit address)`
  - `const void* data_addr:= pointer to array in which data read from F-RAM will be stored`
  - `unsigned int size:= size of the data to be read from the F-RAM`

Refer to console command functions (`fram_write`, `fram_read`) defined in the *command_console_platorm.c* file for an example of the F-RAM SPI driver (*Wiced-SDK/libraries/utilities/command_console/platform/*)

.

Document Revision History

Document Title: Enabling F-RAM Storage in WICED

Document Number:

Revision	ECN	Issue Date	Description of Change
**			Initial Release

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Community | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709