CE211423 - CY8CKIT-042-BLE – F-RAM™ I²C

Objective

This example project is based on a PSoC® Creator™ starter design for the PSoC 4 device. It demonstrates how an F-RAM device can be used with a PSoC device to read and write data. In this project, the PSoC device scans though the F-RAM address space writing the address and then reading it back.

Development Kit Configuration

This example project is designed to run on the CY8CKIT-042-BLE kit from Cypress Semiconductor. A description of the kit, along with more example programs and ordering information, can be found at http://www.cypress.com/documentation/development-kitsboards/cy8ckit-042-ble-bluetooth-low-energy-ble-pioneer-kit.

The project requires configuration settings changes to run on other kits from Cypress Semiconductor. Table 1 lists supported kits. To use any other kit, change the project's device with the help of Device Selector called from the project's context menu.

Table 1. Development Kits and Parts

Development Kit	Device
CY8CKIT-042-BLE	CY8C4247LQI-BL483
CY8CKIT-044	CY8C4247AZI-M485
CY8CKIT-046	CY8C4248BZI-L489

Pin assignments for the supported kits are in Table 2.

Table 2. Pin Assignments for Supported Kits

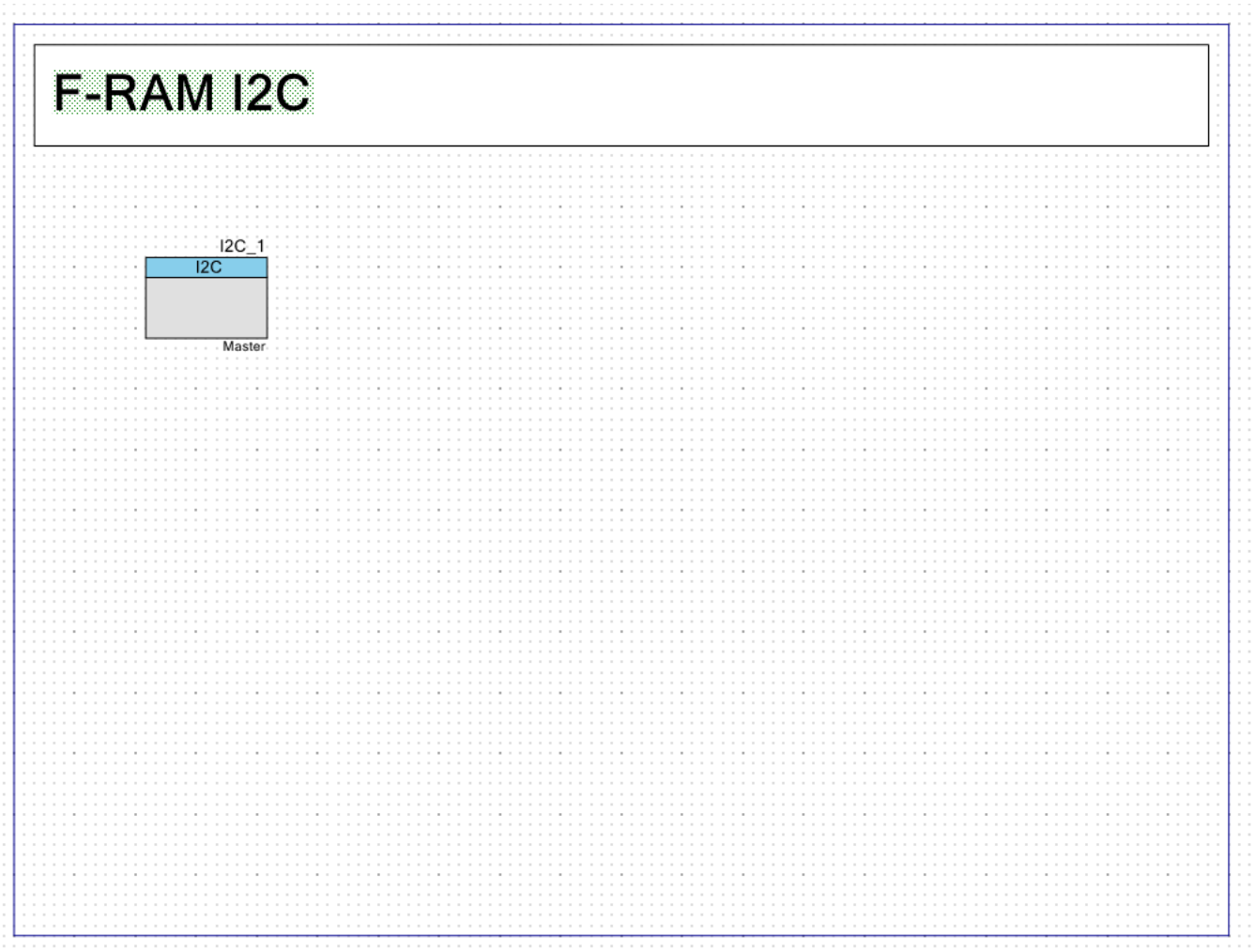
Pin Name	Development Kit			
	CY8CKIT-042	CY8CKIT-042 BLE	CY8CKIT-044	CY8CKIT-046
\I2C_1:scl\	P5[1]	P5[1]	P5[1]	P5[1]
\I2C_1:sda\	P5[0]	P5[0]	P5[0]	P5[0]

Set jumper J9 (J16 for CY8CKIT-042-BLE) to 3.3V position.

Project Configuration

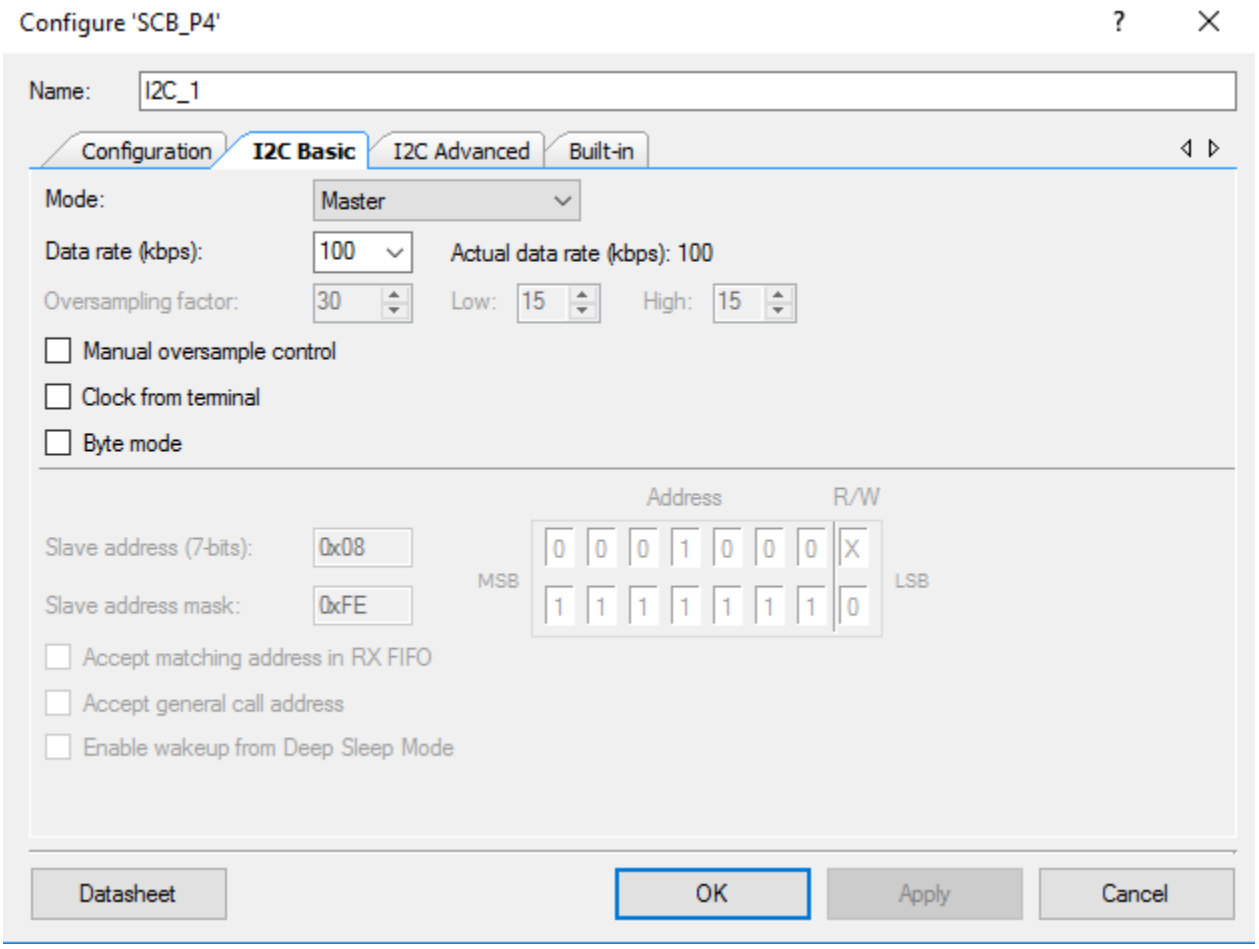
This example project consists of an I²C master and a slave F-RAM device. The top design schematic is shown in Figure 1.

Figure 1. Top Design Schematic



The I²C master is used to move data to and from the F-RAM device. The I²C Component configuration is shown in Figure 2.

Figure 2. I²C Component Configuration Window



Project Description

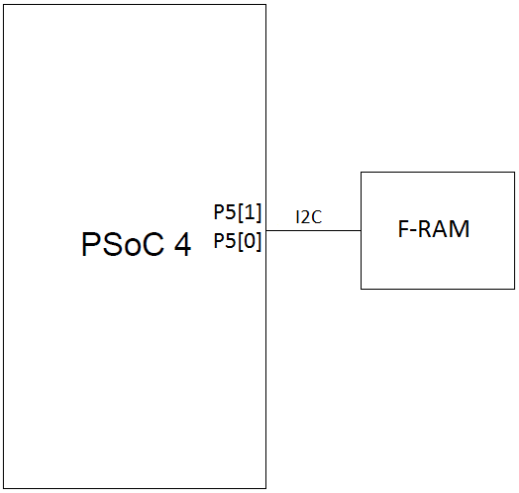
In the main function, the I²C master is started. The “for” loop in *main.c* loops the address space of the F-RAM and writes the address value. . Please refer to the project files included with this project.

Expected Results

There is no output from the PSoC device. This project is designed to be a starting template to show how the F-RAM device is accessed via the PSoC device.

Schematic

Figure 3. Connection Schematic



Using I²C to Communicate with the F-RAM Device

This example project communicates with the F-RAM device using I²C. I²C is a simple Clock/Data bus using 8-bit serial words. Code 1 and Code 2 perform the Write and Read operations for the I²C master to access the F-RAM device.

Code 1. I²C F-RAM Write

```
/*******************************************************************************

* Function Name: FRAM_Write
********************************************************************************
*
* Summary:
*  F-RAM Byte Write Command
* uint16 address - address in the F-RAM array
* uint8 data - data to be written
*
*******************************************************************************/
void FRAM_Write(uint16 address, uint8 data)
{
    uint8 txBuffer[BUFFER_SIZE];

    txBuffer[0] = address >> 8;
    txBuffer[1] = address;
    txBuffer[2] = data;


    /* Clear any previous status */
    I2C_1_I2CMasterClearStatus();

    /* I2C Wrtie Command */
    status = I2C_1_I2CMasterWriteBuf(SLAVE_ADDRESS, (uint8 *) txBuffer, BUFFER_SIZE,
I2C_1_I2C_MODE_COMPLETE_XFER);

    for(;;)
    {
        if(0u != (I2C_1_I2CMasterStatus() & I2C_1_I2C_MSTAT_WR_CMPLT))
        {
            /* Transfer complete. Check Master status to make sure that transfer
            completed without errors. */
            break;
```

```
        }
    }
}
```

Code 2. I²C F-RAM Read

```
/*******************************************************************************
* Function Name: FRAM_Read
********************************************************************************
*
* Summary:
*  F-RAM Byte Read Command
* uint16 address - address in the F-RAM array
* uint8 retrun - data to be read
*
*******************************************************************************/
uint8 FRAM_Read(uint16 address)
{
    uint8 rxBuffer[BUFFER_SIZE-2];
    uint8 txBuffer[2];

    txBuffer[0] = address >> 8;
    txBuffer[1] = address;

    /* Clear any previous status */
    I2C_1_I2CMasterClearStatus();

    /* I2C Wrtie Command to Set the Address*/
    status = I2C_1_I2CMasterWriteBuf(SLAVE_ADDRESS, (uint8 *) txBuffer, 0x2,
I2C_1_I2C_MODE_COMPLETE_XFER);
    for(;;)
    {
        if(0u != (I2C_1_I2CMasterStatus() & I2C_1_I2C_MSTAT_WR_CMPLT))
        {
            /* Transfer complete. Check Master status to make sure that transfer
            completed without errors. */
            break;
        }
    }

    /* Clear any previous status */
    I2C_1_I2CMasterClearStatus();


    /* I2C Read Command */
    status = I2C_1_I2CMasterReadBuf(SLAVE_ADDRESS, (uint8 *) rxBuffer, 1, I2C_1_I2C_MODE_COMPLETE_XFER);
    for(;;)
    {
        if(0u != (I2C_1_I2CMasterStatus() & I2C_1_I2C_MSTAT_RD_CMPLT))
        {
            /* Transfer complete. Check Master status to make sure that transfer
            completed without errors. */
            break;
        }
    }
    return rxBuffer[0];
}
```

Document History

Document Title: CE211423 - CY8CKIT-042-BLE – F-RAM™ I²C

Document Number: 002-11423

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5160314	JLTO	03/23/2016	New code example
*A	5734122	AESATMP9	05/11/2017	Updated logo and copyright.

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

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

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

©Cypress Semiconductor Corporation, 2016-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.