Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

CE215815 - FM0+ Cyclic Redundancy Check

This example uses the Cyclic Redundancy Check (CRC) peripheral to calculate a checksum and validate the data, using the CCIT CRC16 standard.

Overview

This code example defines sixty-four 32-bit values to use as input data to the CRC. It validates the data three times: pushing the data as 8-bit, 16-bit, and 32-bit values. The green LED is used to indicate success or failure.

Requirements

Tool: PSoC Creator 4.0

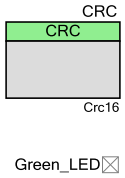
Programming Language: C (GCC 4.9.3)

Associated Parts: All S6E1C parts

Related Hardware: FM0-64L-S6E1C3

Design

The schematic includes the CRC Component and a GPIO for the LED, renamed for ease of use.



The firmware performs following functions:

1. Initializes the CRC peripheral
2. Pushes the input data to the CRC peripheral as 32-bit, 16-bit, and 8-bit values, and checks the result each time
3. Turns on the green LED if there are no errors (the calculated result matches the checksum)

Design Considerations

CRC32-bit Operation

This code example uses the CCIT CRC16 standard. The peripheral also supports IEEE 802.3 CRC32-bit. To change the standard, modify the **enMode** setting for the peripheral, and change the initial value to 0xFFFFFFFFu (32-bits). In the firmware, change the #define for the checksum, using the macros provided in the *main.c* file.

Data Size and Byte Ordering

Regardless of mode you can push the data to the CRC peripheral as 8-, 16-, or 32-bit values. You have choices for how to order the data, and the bits within the data. The code that confirms the CRC checksum must use the same configuration as the code that created the checksum.

PDL Installation

The project assumes that you have installed the PDL in the location specified in the **Project Management** panel of the **Tools** > **Options** dialog. If that location is incorrect you will see the build error “The given PDL path is invalid. Unable to find required PDSC file.” To correct this problem in a newly-created project open the **Project** > **Properties** dialog and enter the correct path to the PDL. To avoid the problem in projects you create in the future, make sure you put the correct path in the **Tools** > **Options** dialog.

Hardware Setup

The CRC Component does not require any hardware setup. Table 1 lists the pin connections required to use this code example on supported FM0+ kits.

Table 1. List of Pins

Pin	FM0-64L-S6E1C3
Green_LED:GPIO	P3E

Components

Table 2 lists the PSoC Creator Components used in this example, as well as the hardware resources used by each.

Table 2. List of PSoC Creator Components

Component	Version	Hardware Resources
PDL_CRC	1.0	CRC block
PDL_GPIO	1.0	GPIO pin

Parameter Settings

The CRC Component uses default parameter settings, with these exceptions:

Table 3: Component Settings

Tab	Setting	Value
None	Name	CRC
Basic	U32CrcInitValue	0xFFFFu

Operation

Program the kit, then click the Resume Execution button. If the green LED lights, the CRC checksum matched the calculation in all three cases (8-, 16-, and 32-bit values).

Related Documents

Table 4 lists relevant application notes, code examples, knowledge base articles, device and Component datasheets.

Table 4. Related Documents

PSoC Creator Component Datasheets	
PDL_CRC	Supports IEEE-802.3 CRC32 and CCITT CRC16 standards (right-click the Component to access)
Device Documentation	
S6E1C	FM0+ S6E1C-Series Ultra Low Power ARM® Cortex®-M0+ Microcontroller (MCU) Family
Development Kit (DVK) Documentation	
FM0-64L-S6E1C3	ARM® Cortex®-M0+ MCU Starter Kit with USB and Digital Audio Interface

Document History

Document Title: CE215815 - FM0+ Cyclic Redundancy Check

Document Number: 002-15815

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5373366	YFS	09/23/16	New Code Example.
*A	5775156	YFS	6/15/17	Added search keyword so that user can quickly find Code Examples from the component instance popup menu. Updated logo and copyright date.
*B	5987561	YFS	12/7/17	Removing S6E1B support.

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
Lighting & Power Control	cypress.com/powerpsoc
Memory	cypress.com/memory
PSoC	cypress.com/psoc
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless/RF	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

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

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com

© 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.