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.

CE219339 – PSoC 6 MCU – MCWDT and RTC Interrupts (Dual CPU)

Objective

This example demonstrates how to configure multiple peripherals to generate interrupts on multiple CPUs in PSoC 6 MCU.

Overview

PSoC 6 MCU has dual CPUs: an ARM® Cortex®-M4 (CM4) and a Cortex-M0+ (CM0+). This example uses a PSoC 6 MCU multi-counter watchdog timer (MCWDT) and a real-time clock (RTC) to generate the interrupts. The interrupts are assigned to separate CPUs.

Requirements

Tool: PSoC Creator™ 4.2; Peripheral Driver Library (PDL) 3.0.1

Programming Language: C (Arm® GCC 5.4.1 and Arm MDK 5.22)

Associated Parts: All PSoC 6 MCU parts

Related Hardware: CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

Hardware Setup

This example uses the kit's default configuration. Refer to the kit guide to ensure the kit is configured correctly.

Software Setup

None.

Operation

1. Plug the CY8CKIT-062-BLE kit board into your computer's USB port.
2. Build the project and program it into the PSoC 6 MCU device. Choose **Debug** > **Program**. For more information on device programming, see PSoC Creator Help. Flash for both CPUs is programmed in a single program operation.
3. Confirm that orange LED toggles every second and red LED toggles every 5 seconds.

Design and Implementation

The MCWDT generates an interrupt every second, with the interrupt assigned only to CM0+. On each interrupt, the orange LED on the kit toggles.

The RTC interrupt is assigned to CM4; the RTC generates an interrupt every 5 seconds. This interrupt toggles the red LED on the kit.

Figure 1 shows the PSoC Creator Schematics of this code example. This example uses MCWDT, RTC, Interrupt, and Pins Components.

Figure 1. PSoC Creator Project Schematic

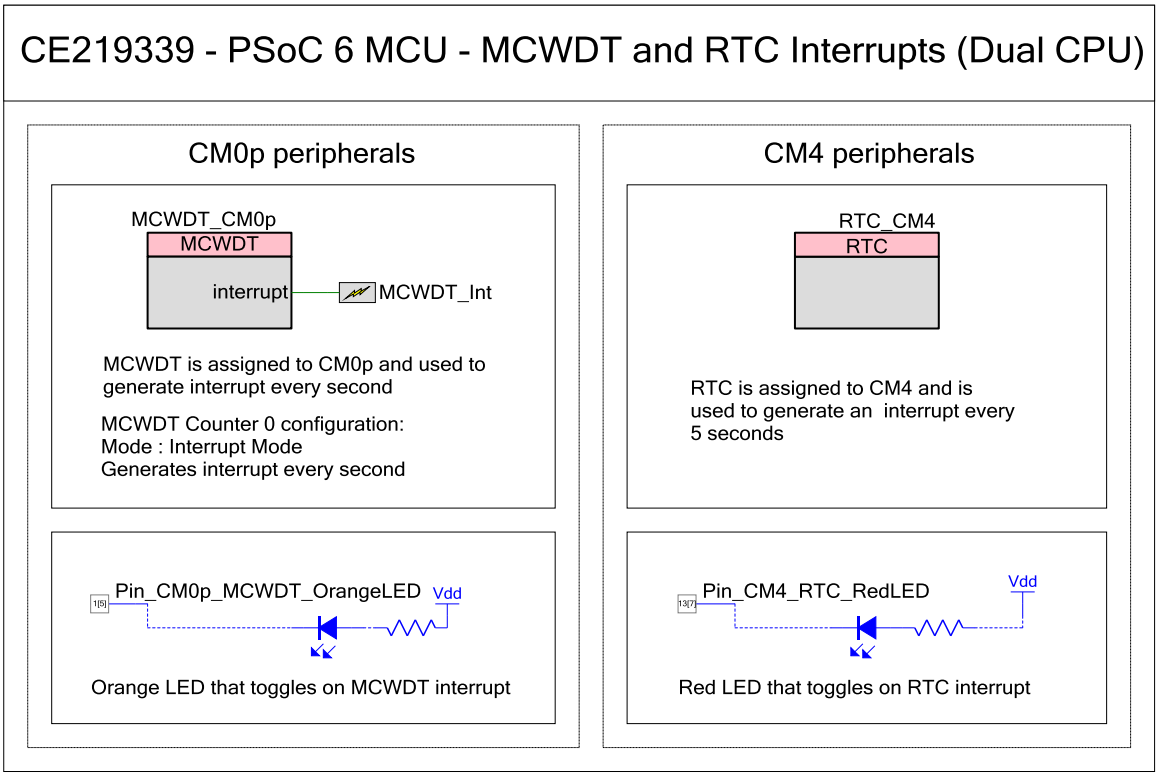
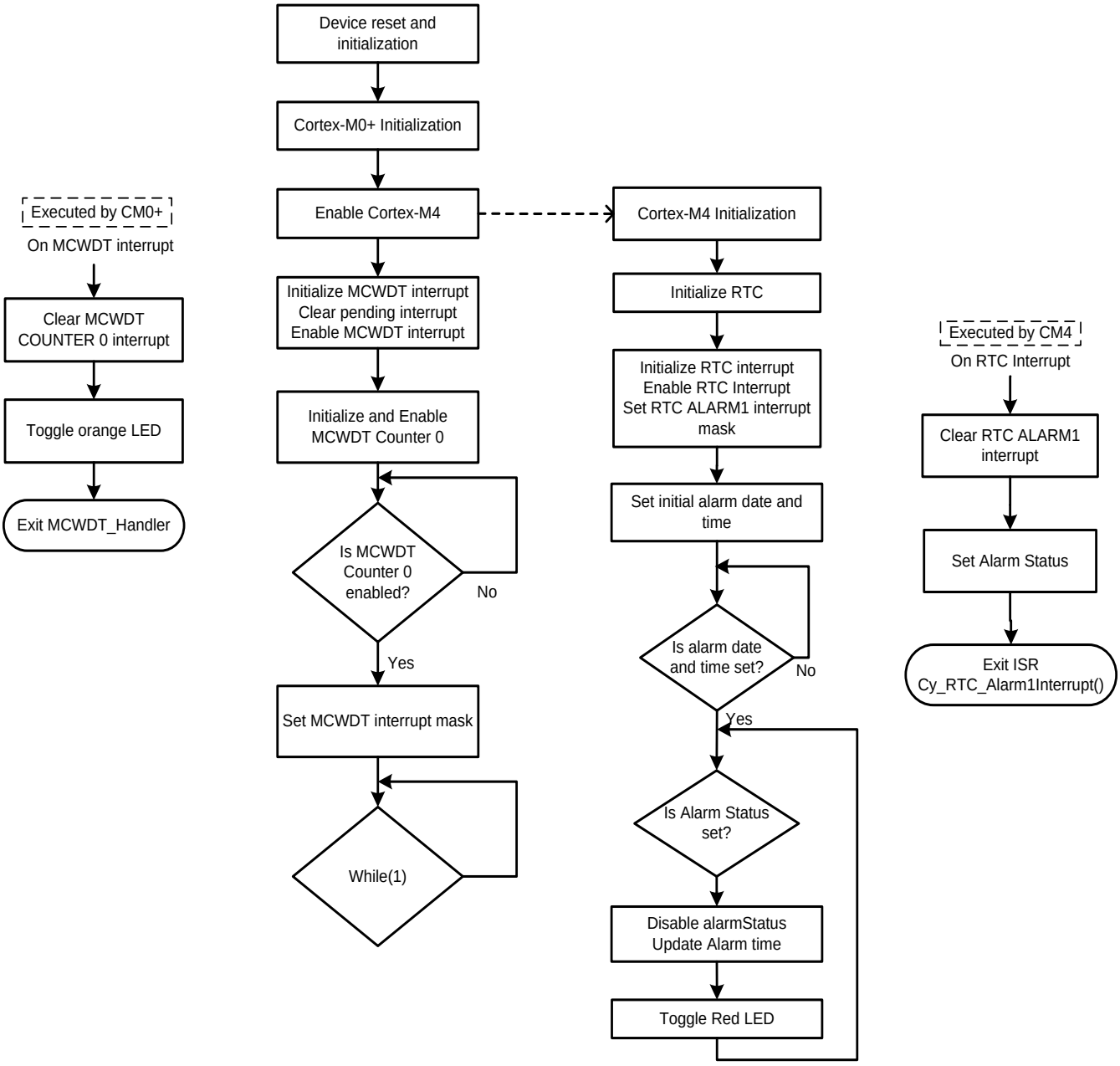


Figure 2 shows the firmware flowchart. Note that after a PSoC 6 MCU device reset, CM0+ always executes first while CM4 is held in a reset state.

Figure 2. MCWDT+RTC Firmware Flow



Keep the time between interrupts and LED blinking time sufficiently high to visually observe the transition in the LED state.

Components and Settings

Table 1 lists the PSoC Creator Components used in this example, how they are used in the design, and the non-default settings required so they function as intended.

Table 1: PSoC Creator Components

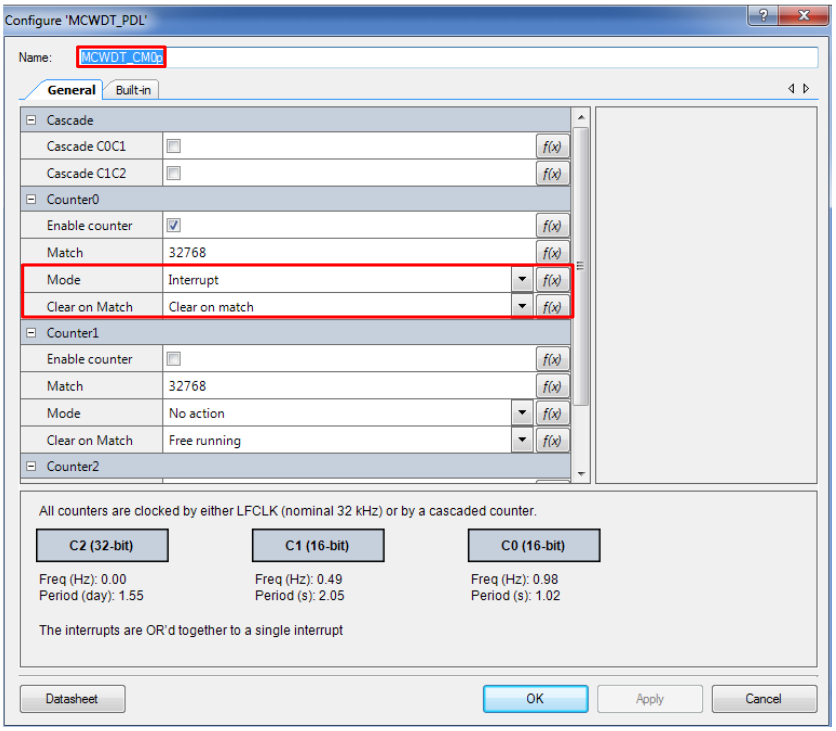
Component	Instance Name	Purpose	Non-default Settings
MCWDT	MCWDT_CM0p	Used to generate an interrupt every second	Uncheck enable counter of counter 0 and counter 1
RTC	RTC_CM4	Used to generate an interrupt every 5 seconds	Check Enable Interrupts
SysInt	MCWDT_Int	Sets the CPU and priority for the interrupt	Select interrupt type as Level Triggered
Digital Output Pin	Pin_CM0p_MCWDT_OrangeLED	Drives the Orange LED	Uncheck HW connection Change Initial Drive state to High
Digital Output Pin	Pin_CM4_RTC_RedLED	Drives the Red LED	Uncheck HW connection Change Initial Drive state to High

For information on the hardware resources used by a Component, see the Component datasheet.

Figure 3 through Figure 6 highlight the non-default settings for each component in this example.

MCWDT Counter 0 is configured to generate interrupt every second as shown in Figure 3. Also, observe only Counter 0 is enabled.

Figure 3. MCWDT Component Configuration



RTC is configured to generate interrupts as shown in Figure 4.

Figure 4. RTC Component Configuration

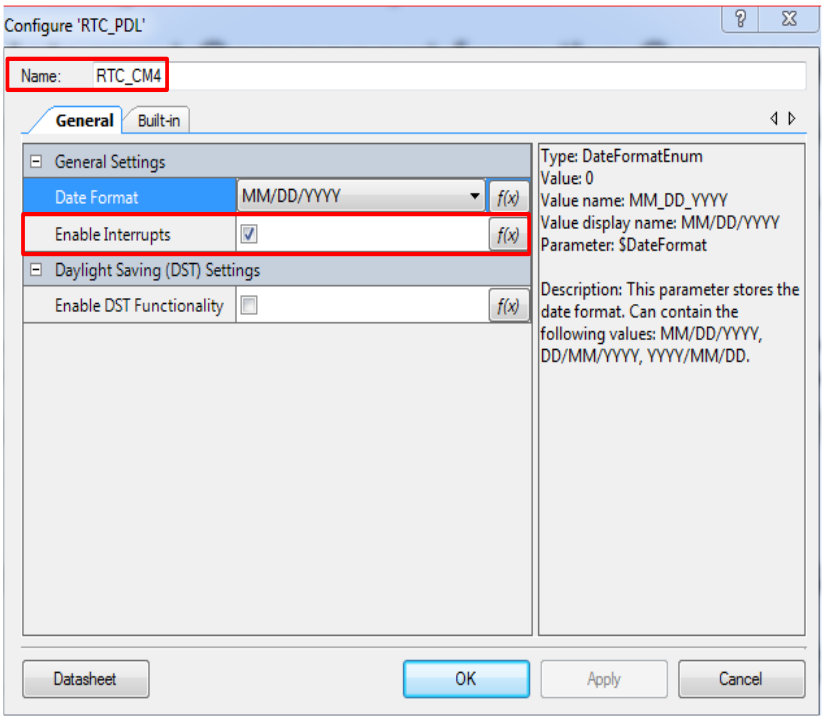
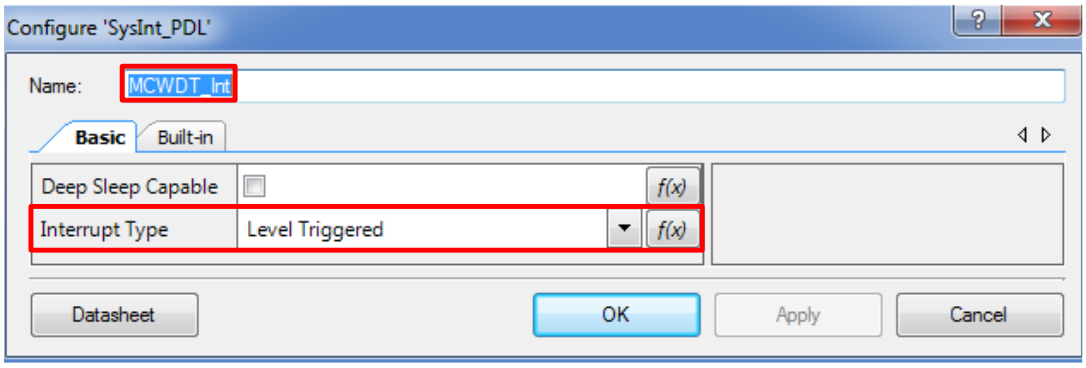


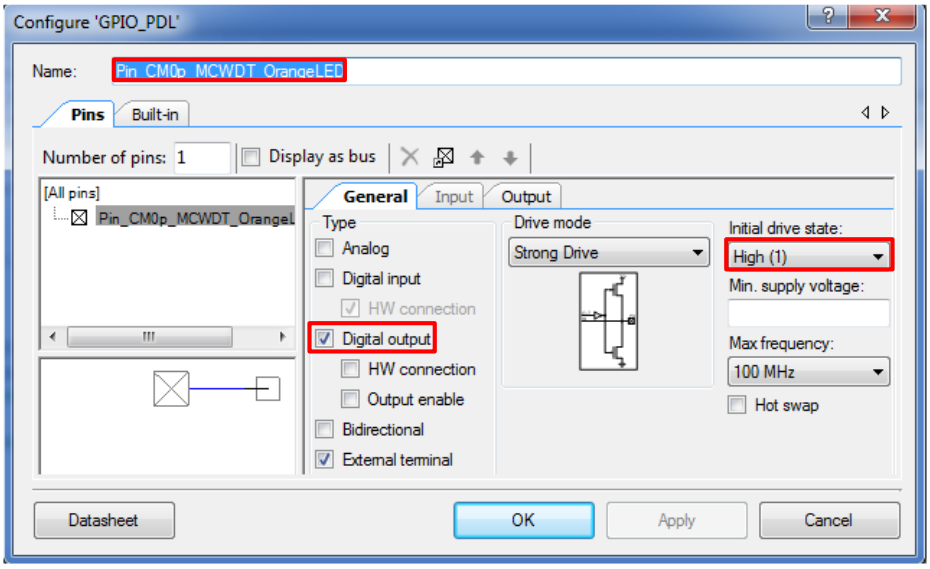
Figure 5. SysInt Component Configuration



As the Deep Sleep wake up functionality is not used, so you do not have to select the **Deep Sleep Capable** checkbox.

InterruptType is selected as **Level Triggered** because the SysInt Component is connected to a dedicated interrupt source (MCWDT interrupt) as shown in Figure 5.

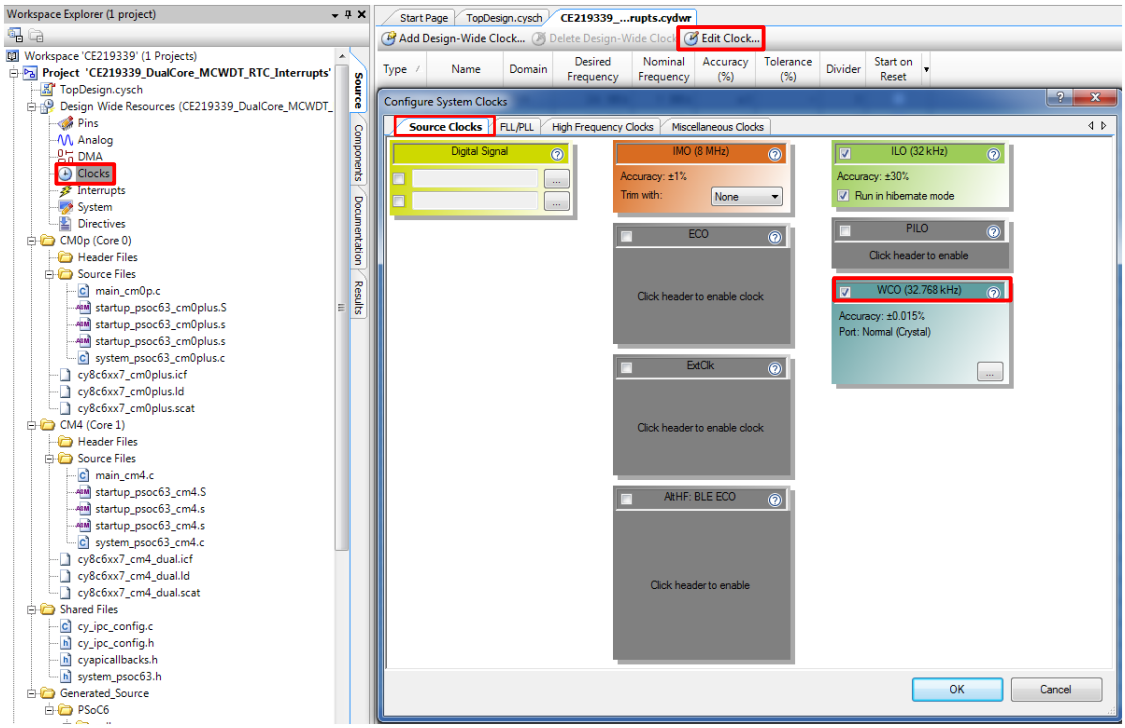
Figure 6. Pins Configuration



Pin_CM4_RTC_RedLED is configured similarly.

Figure 7 shows the system clocks configuration for using RTC. A backup clock source is required for the RTC. The watch crystal oscillator (WCO) is turned ON for an accurate functioning of RTC.

Figure 7. System clocks Configuration for using RTC



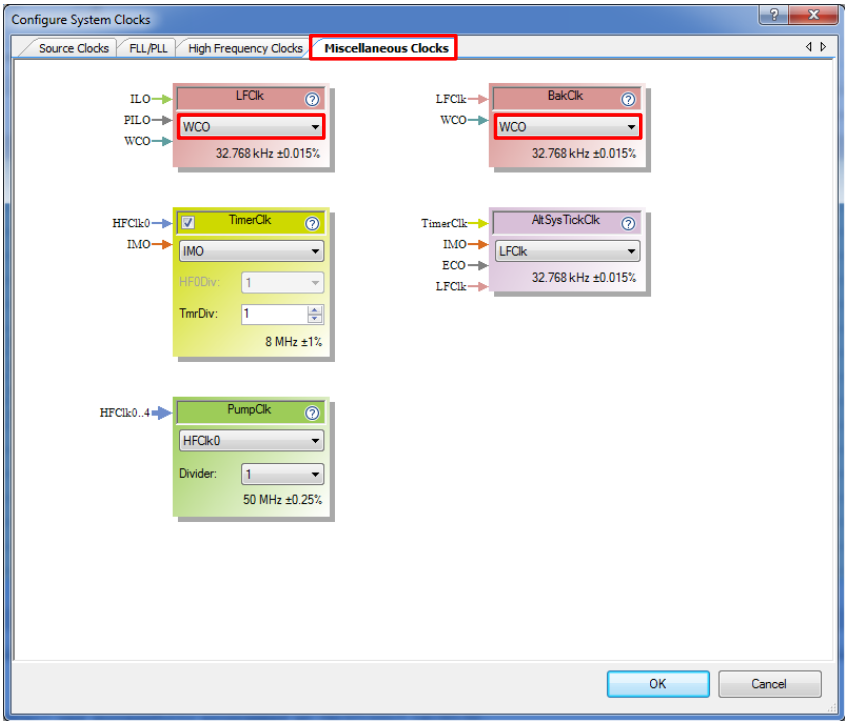
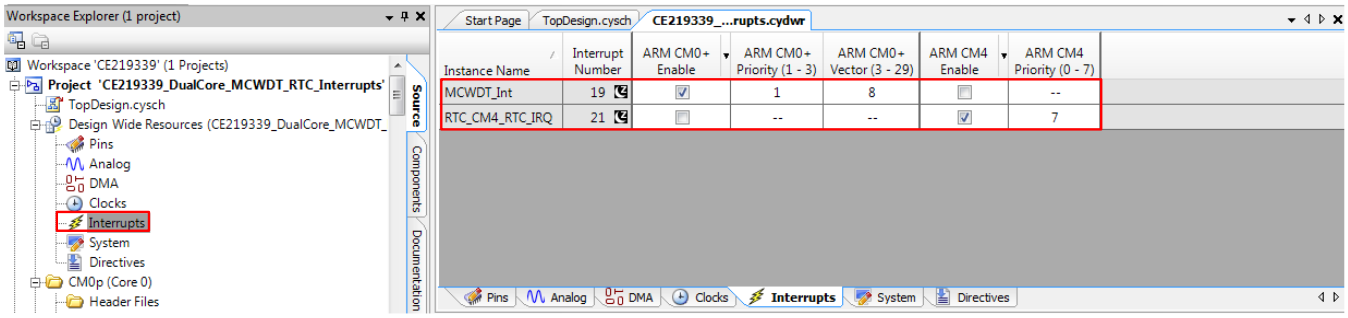


Figure 8 shows the interrupt configuration for the project. The MCWDT_Int Component is assigned to CM0+ through the checkbox. CM0+ vector number 8 is chosen to as you do not require any Deep Sleep wake up (Vectors 0 – 7 are Deep Sleep capable). Interrupt priority can be left at its default value as only one interrupt is assigned to each core. RTC has inbuilt interrupt functionality and uses dedicated interrupt number 21. Vectors 0 – 32 are Deep Sleep capable in CM4. For more information on configuring interrupts, see AN217666 – PSoC 6 MCU Interrupts.

Figure 8. System Interrupt Configuration



Reusing This Example

This example is designed for the CY8CKIT-062-BLE pioneer kit. To port the design to a different PSoC 6 MCU device and/or kit, change the target device using the Device Selector and update the pin assignments in the Design Wide Resources Pins settings as needed.

In some cases, a resource used by a code example (for example, an IP block) is not supported on another device. In that case the example will not work. If you build the code targeted at such a device, you will get errors. See the device datasheet for information on what a particular device supports.

Related Documents

Application Notes	
AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity	Describes PSoC 6 MCU with BLE Connectivity devices and how to build your first PSoC Creator project
AN215656 – PSoC 6 MCU: Dual-Core CPU system Design	Describes the dual-core CPU architecture in PSoC 6 MCU, and shows how to build a simple dual-core design
AN217666 – PSoC 6 MCU Interrupts	Presents theory and configuration details related to interrupts in PSoC 6 MCU
PSoC Creator Component Datasheets	
MCWDT	Provides MCWDT settings
RTC	Provides Real Time Clock Settings
Pins	Supports connection of hardware resources to physical pins
Interrupt	Supports generating interrupts from hardware signals
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet	PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual
Development Kit Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	

Cypress Resources

Cypress provides a wealth of data at www.cypress.com to help you to select the right PSoC device for your design, and quickly and effectively integrate the device into your design. For a comprehensive list of resources, see KBA86521. The following is an abbreviated list of resources:

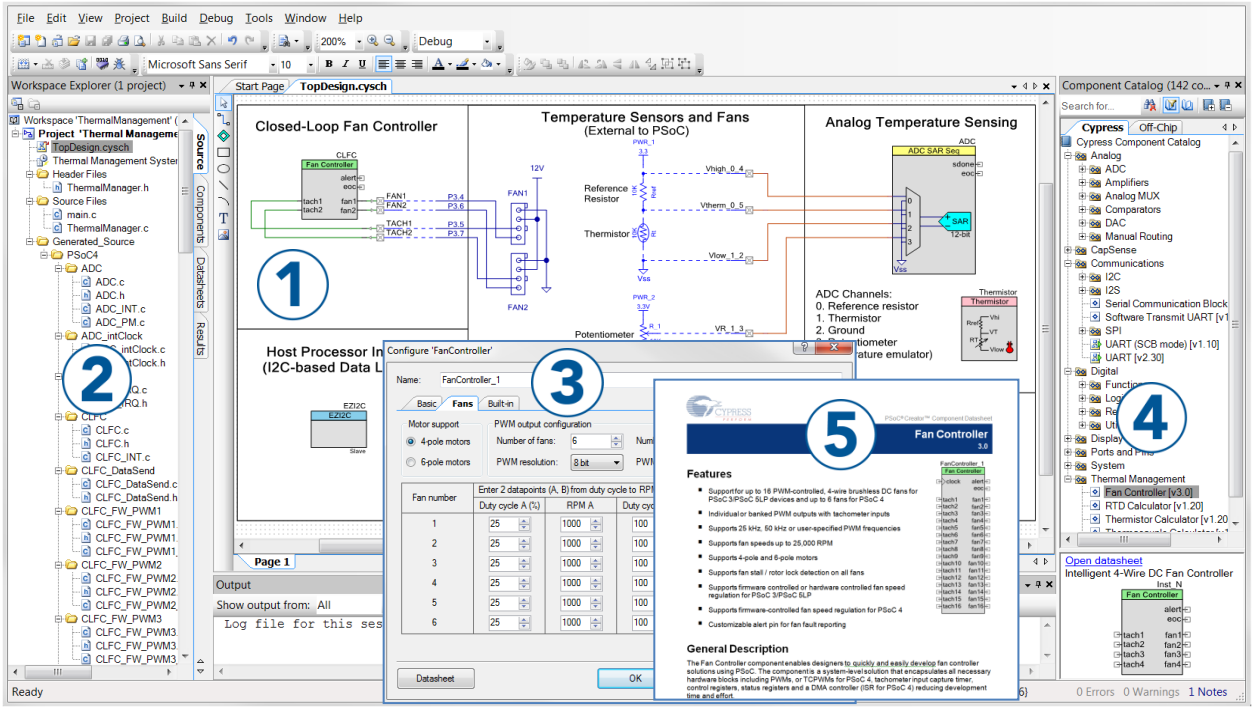
- **Overview:** MCU Portfolio, PSoC & MCU Roadmap
- **Product Selectors:** PSoC 1, PSoC 3, PSoC 4, PSoC 5LP, or PSoC 6. In addition, PSoC Creator includes a device selection tool.
- **Datasheets:** Describe and provide electrical specifications for MCU and PSoC device families.
- **CapSense Design Guides:** Learn how to design capacitive touch-sensing applications.
- **Application Notes:** Cover a broad range of topics, from basic to advanced level.
- **Code Examples**: for PSoC 3, PSoC 4, and PSoC 5LP; or for PSoC 6.
- PSoC **Technical Reference Manuals (TRM):** Provide detailed descriptions of the architecture and registers for PSoC device family.
- **Training Videos**: These videos provide guidance on getting started with various Cypress product families and tools.
- **Development Kits:** Some examples include:
  - PSoC 6 BLE Pioneer Kit is a low-cost hardware platform that enables design and debug of the PSoC 63 series. It comes with an E-Ink display shield board.
  - CY8CKIT-042 and CY8CKIT-040, Pioneer kits, are easy-to-use and inexpensive development platforms. These kits include connectors for Arduino™ compatible shields and Digilent® Pmod™ daughter cards.
  - CY8CKIT-049 is a series of very low-cost prototyping platform for sampling PSoC 4 devices.
  - CY8CKIT-030 and CY8CKIT-050 are designed for analog performance. They enable you to evaluate, develop, and prototype high-precision analog, low-power, and low-voltage applications powered by PSoC 3 and PSoC 5LP, respectively.
  - CY8CKIT-001 is a common development platform for all PSoC family devices.
- The MiniProg3 device provides an interface for flash programming and debug.

PSoC Creator

PSoC Creator is a free Windows-based Integrated Design Environment (IDE). It enables concurrent hardware and firmware design of systems based on PSoC 3, PSoC 4, PSoC 5LP, and PSoC 6 MCU. See Figure 9 – with PSoC Creator, you can:

1. Drag and drop Components to build your hardware system design in the main design workspace
2. Codesign your application firmware with the PSoC hardware
3. Configure Components using configuration tools
4. Explore the library of 100+ Components
5. Review Component datasheets

Figure 9. PSoC Creator Features



Document History

Document Title: CE219339 – PSoC 6 MCU – MCWDT and RTC Interrupts (Dual CPU)

Document Number: 002-19339

Revision	ECN	Orig. of Change	Submission Date	Description of Change
*A	5856560	JSLN/ARVI	08/17/2017	Initial Public Release
*B	6003671	JSLN	12/22/2017	Updated for PSoC Creator 4.2
*C	6065823	JSLN	02/09/2018	Updated Template. Modified title from PSoC 6 MCU – MCWDT and RTC Interrupts (Dual core) to PSoC 6 MCU – MCWDT and RTC Interrupts (Dual CPU). Enabled WCO in the project. Modified project name from CE219339_DualCore_MCWDT_RTC_Interrupts to CE219339_DualCPU_MCWDT_RTC_Interrupts.

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 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,2017-2018. 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. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress does not assume any liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. 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.