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.

CE218964 - PSoC 6 MCU RTC Daily Alarm

Objective

This example demonstrates the RTC Alarm function of the PSoC® 6 MCU Real-Time Clock (RTC).

Overview

This code example demonstrates how to configure RTC registers for a daily alarm using the RTC driver API in the Peripheral Driver Library (PDL). A GPIO output is included for an LED to notify alarm expiration. A UART is used to show the current and alarm times.

Requirements

Tools: PSoC Creator™ 4.2; Peripheral Driver Library (PDL) 3.0.2

Programming Language: C (Arm® GCC 5.4-2016-q2-update, Arm MDK Generic)

Associated Parts: All PSoC 6 MCU parts

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

Hardware Setups

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

Software Setup

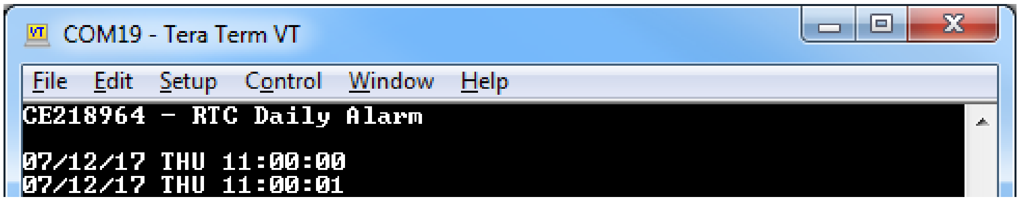
This code example requires a PC terminal software.

Operation

1. Connect the CY8CKIT-062-BLE Pioneer Kit to your computer's USB port.
2. Open a PC terminal using a tool like Tera Term or PuTTY. Configure it for 115,200 baud at data bit 8, no parity and 1 stop bit to match the UART Component.
3. 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.

   Note: Do not delete or replace the *stdio_user.h* file, if prompted by PSoC Creator.
4. Confirm that the terminal program is working. It should show a starting message in the terminal window as shown in Figure 1.

Figure 1. UART Display Start Message

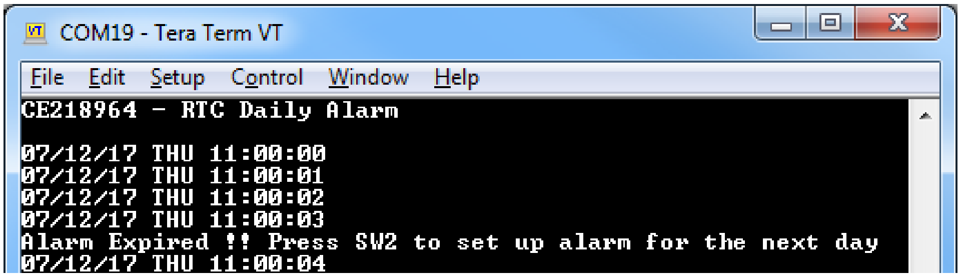


```
CE218964 - RTC Daily Alarm

07/12/17 THU 11:00:00
07/12/17 THU 11:00:01
```

5. Wait three seconds. It should show the message `Alarm Expired !! Press SW2 for to set alarm for next day` after three seconds (Figure 2).

Figure 2. UART Display Message After RTC Alarm

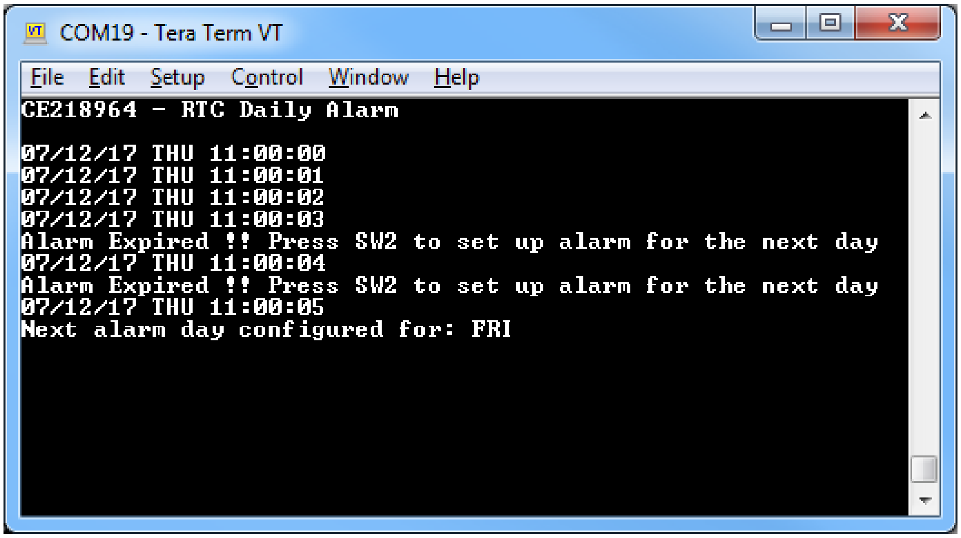


```
CE218964 - RTC Daily Alarm

07/12/17 THU 11:00:00
07/12/17 THU 11:00:01
07/12/17 THU 11:00:02
07/12/17 THU 11:00:03
Alarm Expired !! Press SW2 to set up alarm for the next day
07/12/17 THU 11:00:04
```

6. Confirm red LED (LED_R) toggles every second.
7. Press switch SW2. It should show the next alarm information as shown in Figure 3.

Figure 3. UART Display Message After Switch Is Pressed



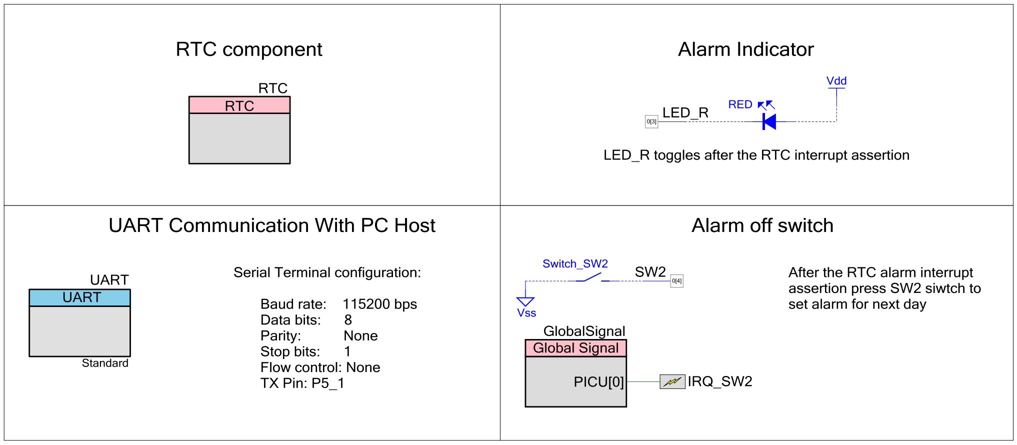
```
CE218964 - RTC Daily Alarm

07/12/17 THU 11:00:00
07/12/17 THU 11:00:01
07/12/17 THU 11:00:02
07/12/17 THU 11:00:03
Alarm Expired !! Press SW2 to set up alarm for the next day
07/12/17 THU 11:00:04
Alarm Expired !! Press SW2 to set up alarm for the next day
07/12/17 THU 11:00:05
Next alarm day configured for: FRI
```

Design

As Figure 4 shows, this code example features one RTC, one GPIO for LED alarm indicator, one UART for user interface, and one GPIO for alarm switch SW2.

Figure 4. RTC Daily Alarm Schematic


RTC component
RTC
RTC
Alarm Indicator
Vdd
RED
LED_R
0[3]
LED_R toggles after the RTC interrupt assertion
UART Communication With PC Host
UART
UART
Standard
Alarm off switch
Switch_SW2
SW2
0[4]
Vss
GlobalSignal
Global Signal
PICU[0]
IRQ_SW2
After the RTC alarm interrupt assertion press SW2 siwtch to set alarm for next day

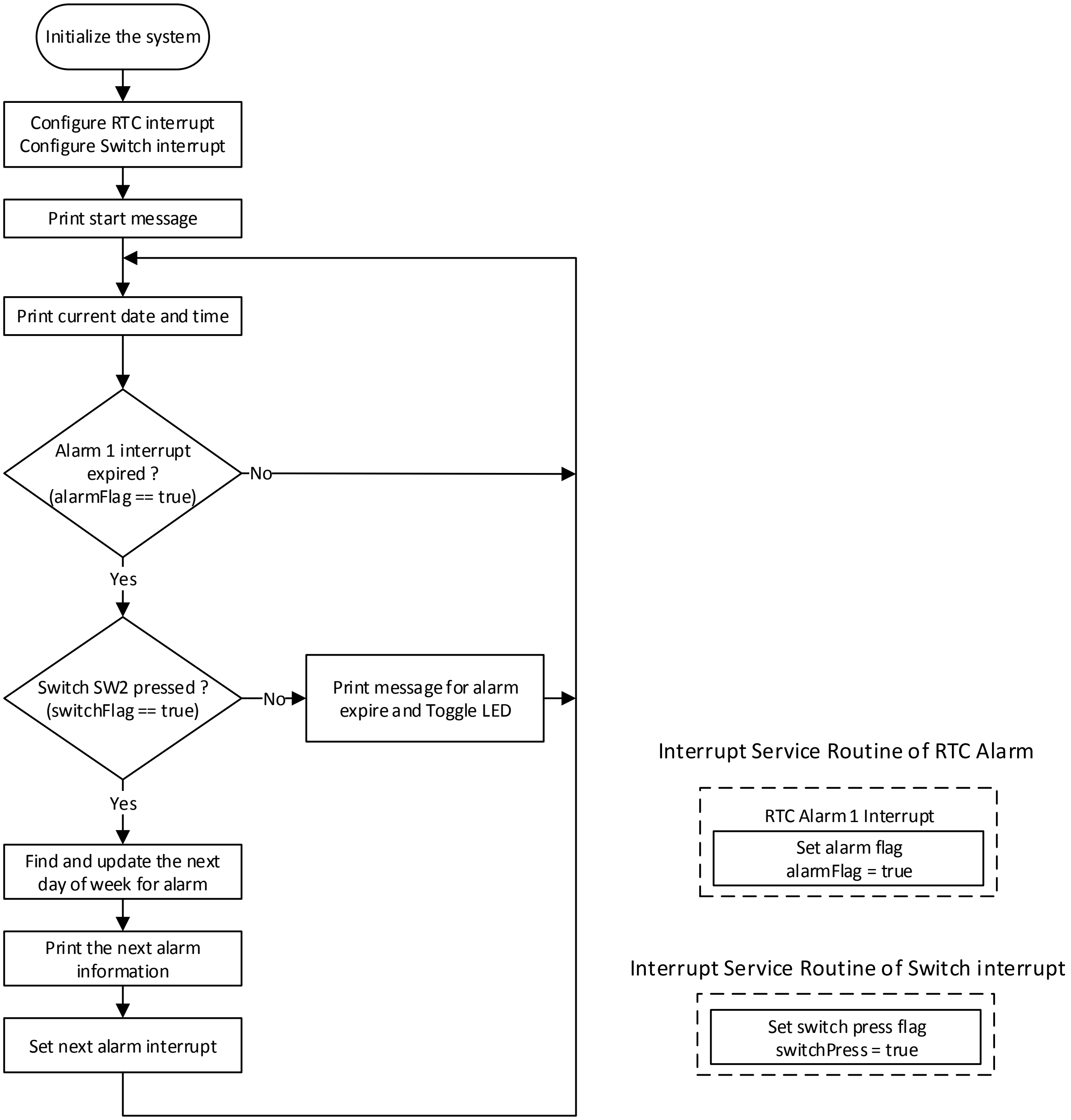
Serial Terminal configuration:

Baud rate: 115200 bps
Data bits: 8
Parity: None
Stop bits: 1
Flow control: None
TX Pin: P5_1

The PSoC 6 RTC is a hardware-based function; the alarm time can be configured by the alarm register fields. The daily alarm needs to enable the hour, minute, and second time fields. Each alarm field is paired with its own enable field. For example, sec (second) field is paired with secEn (second enable) and dayOfWeek field is paired with dayOfWeekEn field. If an enable field is set, the field value will be used for matching the alarm time; otherwise the field value will be ignored. For more information, see PSoC 6 BLE Register Technical Reference Manuals.

As Figure 5 shows, the alarm function uses the RTC alarm 1 interrupt. After an alarm has expired, the code prints the alarm expiration message and toggles the red LED (LED_R, P0[3]) every second until the SW2 button is pressed.

Figure 5. RTC Daily Alarm Flowchart


Initialize the system
Configure RTC interrupt
Configure Switch interrupt
Print start message
Print current date and time
Alarm 1 interrupt expired ? (alarmFlag == true)
No
Yes
Switch SW2 pressed ? (switchFlag == true)
No
Print message for alarm expire and Toggle LED
Yes
Find and update the next day of week for alarm
Print the next alarm information
Set next alarm interrupt
Interrupt Service Routine of RTC Alarm
RTC Alarm 1 Interrupt
Set alarm flag
alarmFlag = true
Interrupt Service Routine of Switch interrupt
Set switch press flag
switchPress = true

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. List of PSoC Creator Components

Component	Instance name	Propose	Parameter
Real Time Clock	RTC	Provide date and time information	**[General Tab]:** Enable Interrupt: Check
UART (SCB)	UART	used for printing terminal messages	**[General Tab]:** TX/RX Mode: TX only
Digital Output Pin	LED_R	Provide visual feedback	**[General Tab]:** HW connection: Uncheck Drive mode: Strong drive
Digital Input Pin	SW2	Provide user interaction	**[General Tab]:** HW connection: Uncheck Drive mode: Resistive Pull Up
Global Signal Reference	Global Signal	Configure the interrupt	**[General Tab]:** Global signal name: Port interrupt 0 (PICU[0])
Interrupt	IRQ_SW2	Configure the interrupt	Default

Design-Wide / Global Resources

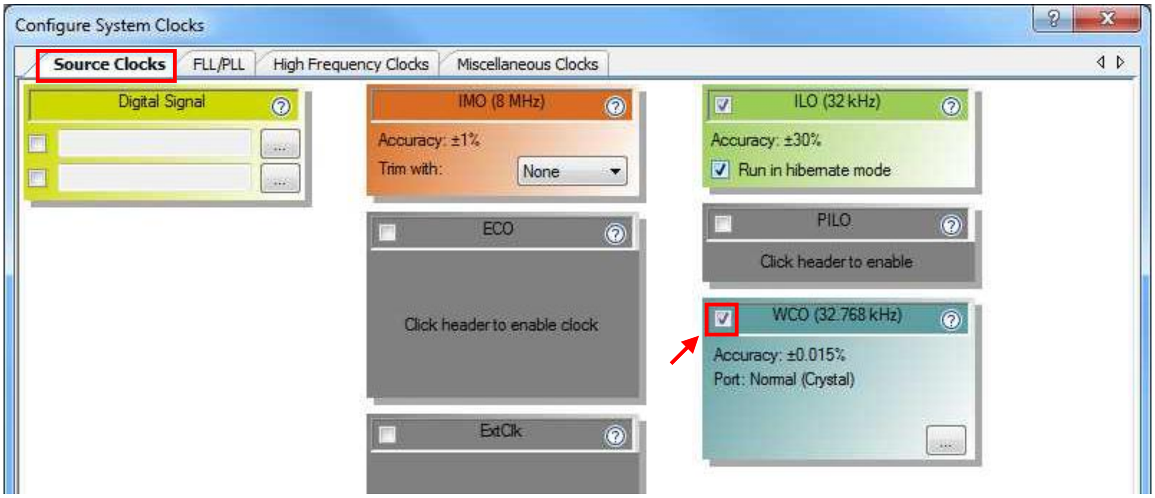
This code example runs on CY8CKIT-062-BLE, which has a PSoC 6 MCU device.

A backup clock is necessary for the RTC to function. For accurate RTC operation, it is recommended that you use a WCO.

Do the following to configure the RTC clock (BakClk) as WCO.

1. Double-click **Clocks** in **Design Wide Resources**.
2. Click **Edit Clock…** and open **Configure System Clocks**.
3. Enable WCO clock for the backup clock source in **Source Clocks**, as Figure 6 shows.

Figure 6. Enable WCO for the RTC Clock



4. Select WCO for BakClk in **Miscellaneous Clocks**, as Figure 7 shows.

Figure 7. Set the Backup Clock Source to WCO

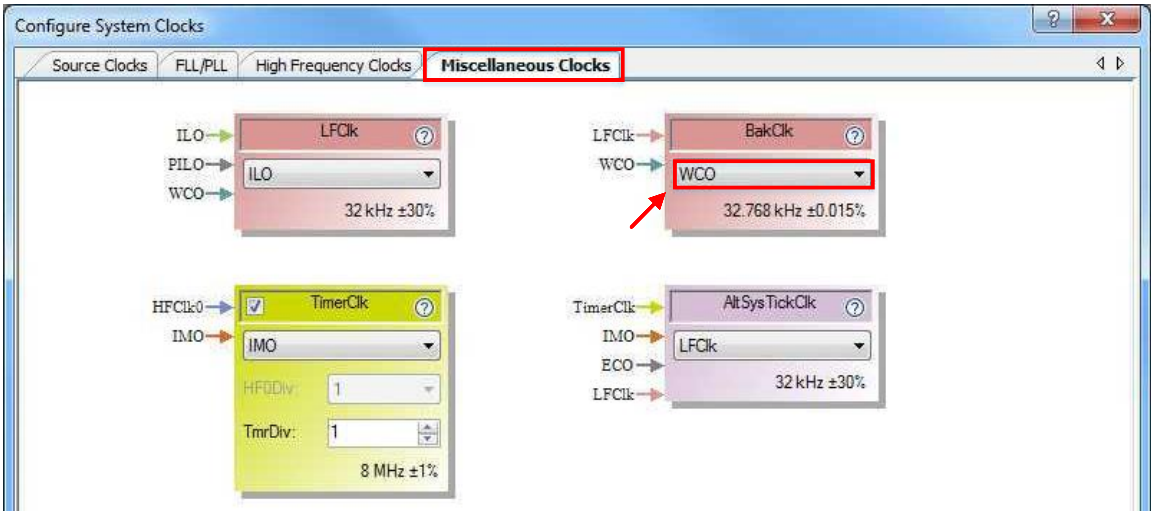
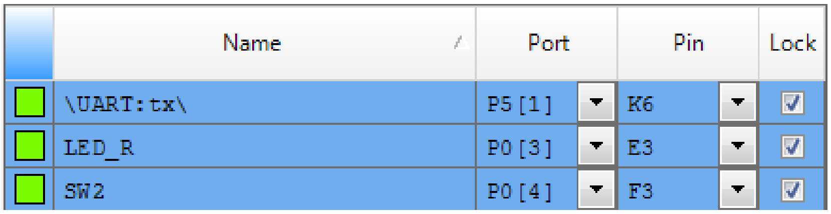


Figure 8 shows the pin assignment for the project done through the **Pins** tab in the **Design Wide Resources** window. These assignments are compatible with CY8CKIT-062-BLE.

Figure 8. Pin Assignments



	Name	Port	Pin	Lock
🟩	\UART:tx\	P5[1]	K6	☑
🟩	LED_R	P0[3]	E3	☑
🟩	SW2	P0[4]	F3	☑

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 kits, change the target device using the Device Selector and update the pin assignments in the Design Wide Resources Pins settings as needed. For single-CPU PSoC 6 MCU devices, port the code from *main_cm4.c* to *main.c*.

Related Documents

Application Notes	
AN221774 – Getting Started with PSoC 6 MCU	Describes PSoC 6 MCU devices and how to build your first PSoC Creator project
AN210781 – Getting Started with PSoC 6 MCU with BLE Connectivity	Describes PSoC 6 MCU with BLE Connectivity devices and how to build your first PSoC Creator project
Code Examples	
CE216825 PSoC 6 Real-Time Clock Basics	
CE218542 PSoC 6 Customer Tick Timer Using RTC Alarm Interrupt	
CE219339 PSoC 6 MCU - MCWDT and RTC Interrupts (Dual Core)	
PSoC Creator Component Datasheets	
Pins	Supports connection of hardware resources to physical pins
RTC	Component provides an application interface for keeping track of time and date
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet	PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual PSoC 6 MCU: PSoC 63 with BLE Registers Technical Reference Manual
Development Kit (DVK) Documentation	
CY8CKIT-062-BLE Pioneer Kit	

Document History

Document Title: CE218964 - PSoC 6 MCU RTC Daily Alarm

Document Number: 002-18964

Revision	ECN	Orig. of Change	Submission Date	Description of Change
*A	5993969	AJYA	12/14/2017	Initial Public Release
*B	6207367	AJYA	06/13/2018	Updated to PSoC Creator 4.2

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

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

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.