CE218129 – PSoC 6 MCU Wakeup from Hibernate Using a Low-Power Comparator

Objective

This code example demonstrates the functionality of wakeup from the Hibernate mode using the LPComp Component in PSoC® 6 MCU.

Overview

This code example demonstrates how to set the Component options for the LPComp internal reference voltage and how to set the external input from a GPIO using the LPComp driver.

The code example uses one GPIO input to compare the input voltage and internal reference voltage to wake the PSoC 6 MCU from Hibernate mode. The LED indicates the current power mode.

Requirements

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

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 Setup

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

Software Setup

None.

Operation

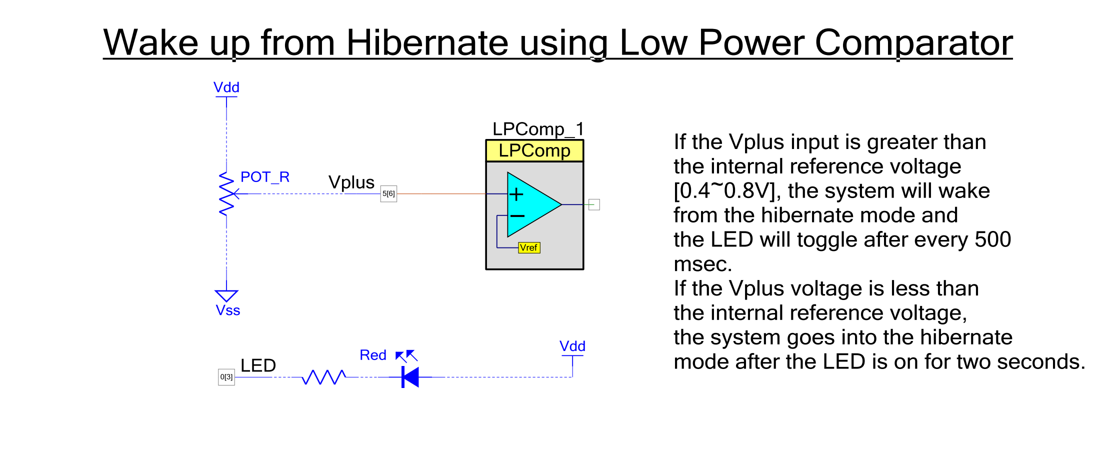
Follow the instructions that came with your kit to make sure that your kit is connected to your PC.

1. On CY8CKIT-062-BLE, place a potentiometer on P5[6] to change the Vplus input voltage.
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. Turn the knob of the potentiometer until the Vplus voltage is greater than the internal V_{ref} (Vplus > 0.8 V).
4. Confirm that the LED is toggling.
5. Turn the knob of the potentiometer until the Vplus voltage is less than the internal V_{ref} (Vplus < 0.4 V).
6. Confirm that the LED is ON for the first two seconds, and then it is OFF during the Hibernate mode.
7. Repeat the operations 3 to 6 for the wakeup and the Hibernate mode.

Design and Implementation

This code example features one Low-Power Comparator (LPComp) Component, one status LED, one GPIO for the wakeup input, and one potentiometer on the Vplus pin, as Figure 1 shows.

Figure 1. Wakeup from Hibernate Using LPComp

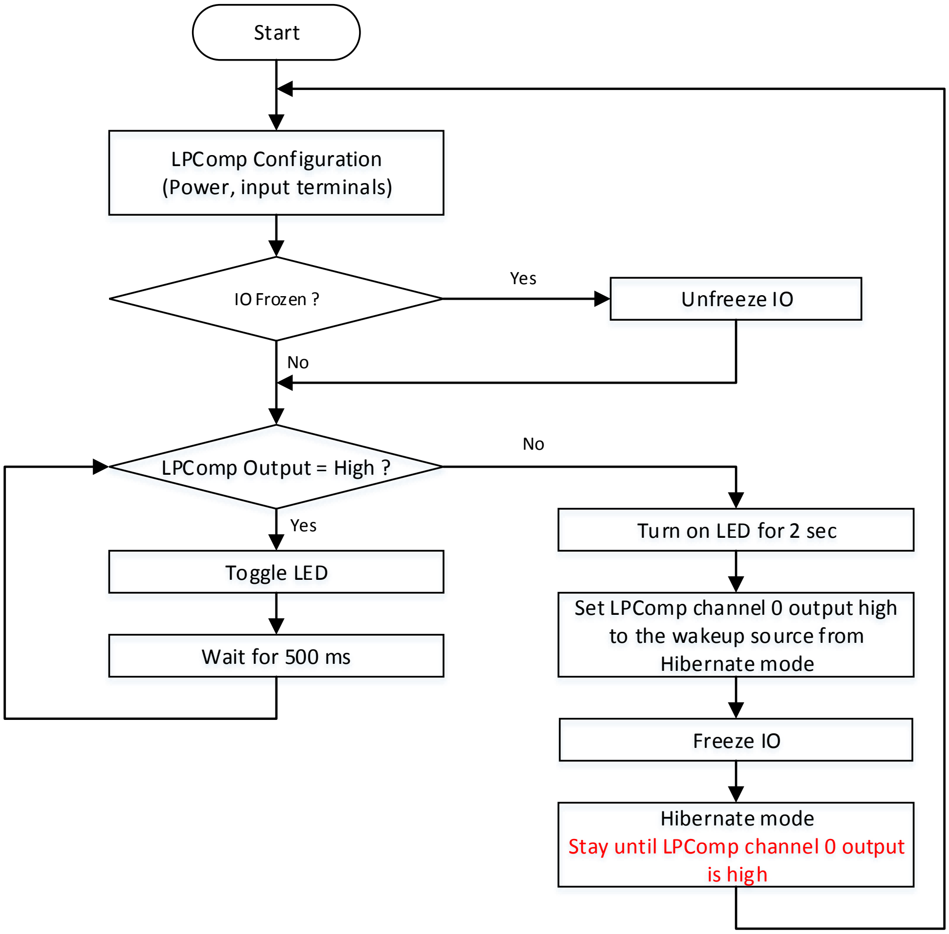


Wake up from Hibernate using Low Power Comparator

If the Vplus input is greater than the internal reference voltage [0.4~0.8V], the system will wake from the hibernate mode and the LED will toggle after every 500 msec.
If the Vplus voltage is less than the internal reference voltage, the system goes into the hibernate mode after the LED is on for two seconds.

Figure 2 shows the firmware flow. The main loop checks the output of LPComp channel 0 and toggles the LED when the output is high. Otherwise, the system goes into the Hibernate mode after turning the LED ON for two seconds. The system will wake up immediately if the LPComp channel 0 output goes high during Hibernate mode.

Figure 2. Flowchart of Wakeup from Hibernate Mode Using the LPComp Input



Design Considerations

The internal reference voltage can vary from 0.4 V to 0.8 V, so the Vplus input should be high enough to cause a wakeup.

In Hibernate mode, the LPComp Component cannot use the AMUXBUS, so the Vplus input should be either P5[6] (LPComp Channel 0) or P6[2] (LPComp Channel 1). These pins are dedicated to the LPComp GPIO positive terminal.

Components and Settings

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 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	Purpose	Non-default Settings
Low Power Comparator	LPComp_1	Provide low power voltage comparison	Hysteresis: Disable Power/Speed: Ultra Low Power/Slow Local VREF input: Checked
Digital Output Pin	LED	Provide visual feedback	Drive mode: Strong Drive Uncheck HW connection
Analog Input Pin	Vplus	Provide user input	Default

To configure LPComp Component using the PDL driver, set the PDL_CONFIGURATION #define to 1; otherwise leave it to 0.

Table 2 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.

Table 2. DWR Pin Assignment Table

Component	Instance Name	Pin
Digital Output Pin	LED_R	P0[3]
Analog Pin	Vplus	P5[6]

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. For single-core PSoC 6 MCU devices, port the code from *main_cm4.c* to *main.c*.

Related Documents

Application Notes	
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
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
AN219434 – Importing PSoC Creator Code into an IDE for a PSoC 6 MCU Project	Describes how to import the code generated by PSoC Creator into your preferred IDE
PSoC Creator Component Datasheets	
Pins	Supports connection of hardware resources to physical pins
Low Power Comparator	Supports low power comparators
Related Code Example	
CE218472 - PSoC 6 MCU Comparing External Voltages Using a Low-Power Comparator	
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet	PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual
Development Kit (DVK) Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	

Document History

Document Title: CE218129 – PSoC 6 MCU Wakeup from Hibernate Using a Low-Power Comparator

Document Number: 002-18129

Revision	ECN	Orig. of Change	Submission Date	Description of Change
*A	5917910	AJYA	10/27/2017	Initial Public Release
*B	6002191	AJYA	12/21/2017	Updated LPComp Component to v1.1
*C	6124049	AJYA	04/05/2018	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

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

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.