Customer training workshop: Switching_Power_Modes for KIT_T2G-B-H_EVK

TRAVEO™ T2G CYT4BF series Microcontroller Training
V1.0.1 2023-03

Please read the **Important notice and warnings** at the end of this document

Scope of work

- This example demonstrates how to transition MCU to the Active, Sleep, DeepSleep, and Hibernate power states, and how to wake up and return to Active state. The main function of this multi-core application runs on the CM7_0 core; it shows the power mode transition.

- Device
  - The TRAVEO™ T2G CYT4BFBCH device is used in this code example.

- Board
  - The TRAVEO™ T2G KIT_T2G-B-H_EVK board is used for testing.

Copyright © Infineon Technologies AG 2023. All rights reserved.

Introduction

- **MCU supports the following power modes:**
  - Active: All peripherals are available
  - Low-Power Active (LPACTIVE) profile: Low-power profile of Active mode where all peripherals including the CPU are available, but with limited capability
  - Sleep: All peripherals except the CPU are available
  - Low-Power Sleep (LPSLEEP) profile: Low-power profile of Sleep mode where all peripherals except the CPU are available, but with limited capability
  - DeepSleep: Only low-frequency peripherals are available
  - Hibernate: The device and I/O states are frozen and the device resets on wakeup
  - XRES: The device enters this state when the XRES_L pin is asserted

Copyright © Infineon Technologies AG 2023. All rights reserved.

Introduction (contd.)

- **Power modes have the following features**
  - Software can use power modes to optimize power consumption in an application.
  - Low-power DeepSleep mode with support for multiple wakeup sources and configurable amount of SRAM retention
  - Ultra-low-power Hibernate mode with wakeup from I/O and timer alarms

The power consumption in different power modes is controlled by using the following methods:

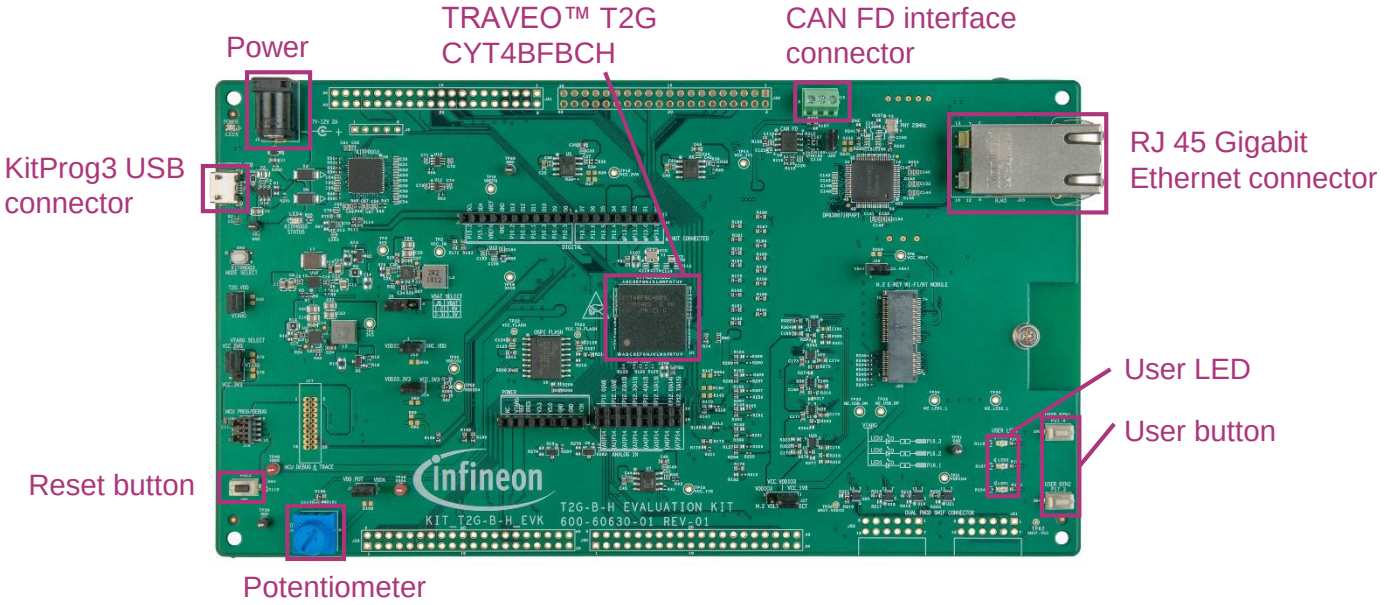
- Enabling and disabling clocks to peripherals
- Powering on/off clock sources
- Powering on/off peripherals and parts inside the device

Copyright © Infineon Technologies AG 2023. All rights reserved.

Hardware setup

› This code example has been developed for the KIT-T2G-B-H-EVK board.

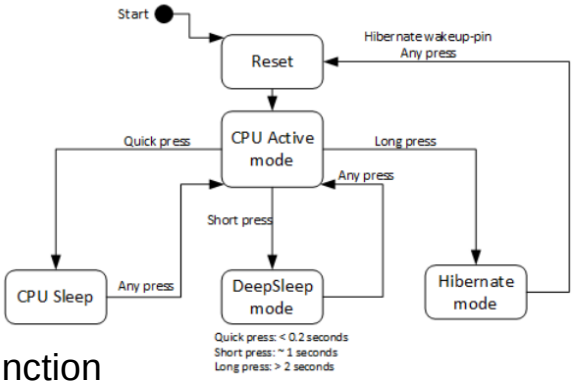
› Connect your PC to the board using the provided USB cable through the KitProg3 USB connector



Copyright © Infineon Technologies AG 2023. All rights reserved.

Implementation

This code example shows how to transition the MCU to Active, Sleep, and DeepSleep power states. After transitioning to Sleep, DeepSleep, or Hibernate power state, the example also shows how to wake up and return to Active state. The project uses a kit button to change the power state. The following figure shows the state machine implemented in the firmware to execute the transitions.



Follow these steps to configure this code example:

› CM0P core enables CM7_0 core and CM7_1 core

› CM0P core and CM7_1 core goes to DeepSleep

› Running the Power modes function on the CM7_0 core

CM0P core enables CM7_0 core and CM7_1 core

› Enable CM7_0 core and CM7_1 core by calling ***Cy_SysEnableCM7()*** function

CM0P core and CM7_1 core goes to DeepSleep

› Both CM0P and CM7_1 core goes to DeepSleep by calling ***Cy_SysPm_CpuEnterDeepSleep()***

Copyright © Infineon Technologies AG 2023. All rights reserved.

Implementation (contd.)

Running the power modes function on the CM7_0 core

- Initialize the UART block for display messages on the terminal
  - Call the ***cy_retarget_io_init()*** function to use UART.
  - Initialize P13.1 as UART TX, P13.0 as UART RX (these pins are connected to KitProg3 COM port).
  - The serial port parameters changes to 8N1 and 115200 baud.
- Initialize the user button
  - Call the ***cyhal_gpio_init()*** function to initialize the User Button P21.4.
  - Call the ***cyhal_gpio_enable_event()*** function to enable User button to wake-up the MCU.
- Initialize the PWM block that controls the LED brightness
  - Call the ***cyhal_pwm_init()*** function to initialize the PWM out peripheral and configure the LED pin P16.1.
  - Call the ***cyhal_pwm_set_duty_cycle()*** function to set the PWM duty cycle and frequency.
  - Call the ***cyhal_pwm_start()*** function to start the PWM generation and outputs.
- Register power management callbacks
  - Call the ***cyhal_syspm_init()*** function to initialize system power management that is needed for future operations.
  - Call the ***cyhal_syspmregister_callback()*** function to register the specified handler with the power manager to be notified of power state changes.

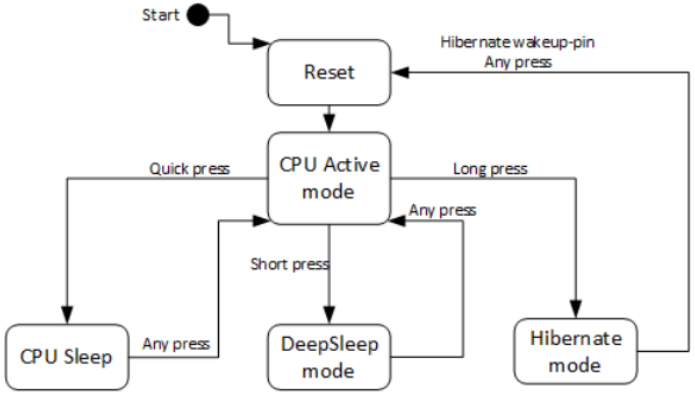
Copyright © Infineon Technologies AG 2023. All rights reserved.

Implementation (contd.)

Running the power modes function on the CM7_0 core

- Go to Sleep mode/DeepSleep mode/Hibernate mode by pressing the user button
  - Call the ***cyhal_syspm_sleep()*** function to set CPU to Sleep mode.
  - Call the ***cyhal_syspm_deepsleep()*** function to set CPU to DeepSleep mode
  - Call the ***cyhal_syspm_hibernate()*** function to set the system mode to Hibernate
- Wakeup from Sleep mode/DeepSleep mode/Hibernate mode
  - Press the user button to enable interrupt to wake-up.

Quick Press : < 0.2 seconds	CPU Sleep	LED blinks (3Hz / duty 100%)
Short Press : ~1 seconds	DeepSleep mode	LED dimmed ON (100Hz / duty 10%)
Long Press : >2 seconds	Hibernate mode	LED OFF



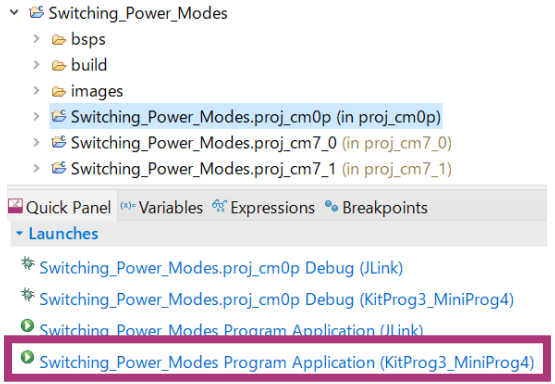
Copyright © Infineon Technologies AG 2023. All rights reserved.

Compiling and programming

1. Connect to power and USB cable
2. Use Eclipse IDE for ModusToolbox™ software for compiling and programming
3. Compile
   a) Select the target application project in the Project Explorer.
   b) In the Quick Panel, scroll down, and click "Build Application" in Switching Power Modes (APP_KIT-T2G-B-H-EVK)



4. Open a terminal program and select the KitProg3 COM port. Set the serial port parameters to 8N1 and 115200 baud.
5. Programming
   a) Select the target application project in the Project Explorer
   b) In the Quick Panel, scroll down, and click "Switching_Power_Modes Program Application (KitProg3_MiniProg4)[1]" in Launches



[1] Multicore is used in this code example. Thus it needs to program 3 times by each core.

Copyright © Infineon Technologies AG 2023. All rights reserved.

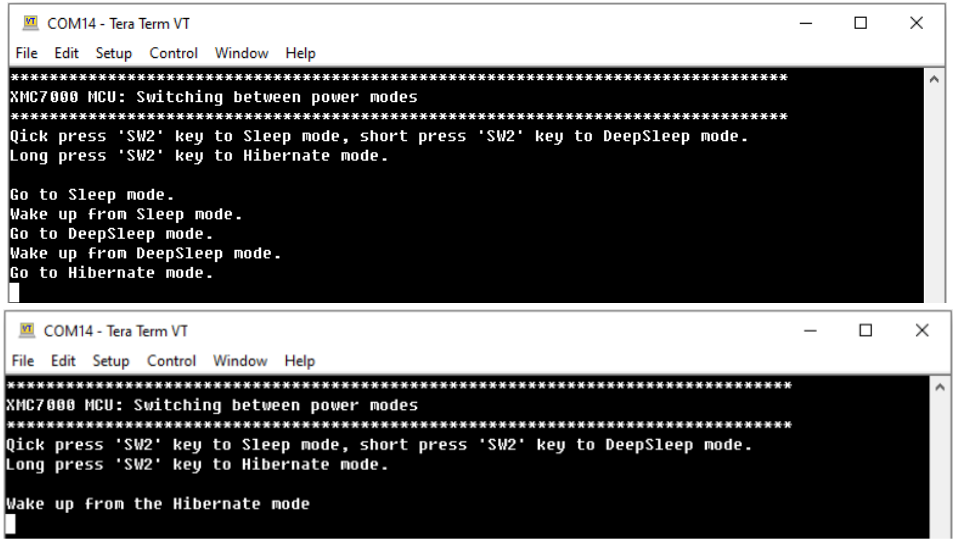
Run and test

1. After programming, the application starts automatically. Confirm that the kit LED blinks at 3 Hz (approximately). Take note of the current consumption. The device is in the Active state at this moment.
2. Press the kit button quickly to transition the device into the Sleep state. Observe that the LED is ON and dimmed. Confirm that the current consumption drops to dozens of milliamperes. The CPU is in the Sleep state at this moment.
3. Quickly press the kit button to return to the Active state. Observe that the LED blinks again.
4. Press the kit button for approximately one second and release it. Observe that the LED is OFF and that the current consumption has dropped to over a hundred microamperes. The device is in DeepSleep state at this moment.
5. Press the kit button quickly to return to the Active state. Observe that the LED blinks again and that the current consumption has increased to the same level measured before.
6. Press the kit button for at least two seconds and release it. Observe that the LED is OFF and that the current consumption has dropped to approximately 10 microamperes. The device is in Hibernate state at this moment.

Copyright © Infineon Technologies AG 2023. All rights reserved.

Run and test (contd.)

7. Press the kit button quickly to wake up from the Hibernate state, and then MCU reset. Observe that the LED is blinking again. When woken up from Hibernate state, the terminal application displays the following message:





Copyright © Infineon Technologies AG 2023. All rights reserved.

References

Datasheet

› CYT4BF datasheet 32-bit Arm® Cortex®-M7 microcontroller TRAVEO™ T2G family

Architecture Technical reference manual

› TRAVEO™ T2G automotive body controller high family architecture technical reference manual

Registers Technical reference manual

› TRAVEO™ T2G automotive body controller high registers technical reference manual

PDL/HAL

› PDL

› HAL

Training

› TRAVEO™ T2G Training

Copyright © Infineon Technologies AG 2023. All rights reserved.

Revision History

Revision	ECN	Submission Date	Description of Change
**	7782653	2022/07/07	Initial release
*A	7876639	2023/03/01	Changed CySysEnableCM7 to Cy_SysEnableCM7 in “Implementation” Removed Cy_SysPm_CpuSleepOnExit() in “Implementation” Changed figures in “Compiling and programming”

Copyright © Infineon Technologies AG 2023. All rights reserved.

Important notice and warnings

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2023-03
Published by
Infineon Technologies AG
81726 Munich, Germany

© 2023 Infineon Technologies AG.
All Rights Reserved.

Do you have a question about this document?
Go to:
www.infineon.com/support

Document reference
002-35580 Rev. *A

IMPORTANT NOTICE

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie") .

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.

Copyright © Infineon Technologies AG 2023. All rights reserved.