

# Customer training workshop: HAL\_GPIO\_Interrupt for KIT\_T2G-B-H\_EVK

TRAVEO™ T2G CYT4BF series Microcontroller Training  
V1.0.1 2023-02



Please read the [Important notice and warnings](#) at the end of this document

## Scope of work

---

- › This code example demonstrates the use of a GPIO configured as an input pin to generate interrupts on an Infineon MCU. The GPIO signal interrupts the CPU and executes a user-defined Interrupt Service Routine (ISR). The GPIO interrupt acts as a wakeup source to wake the CPU from DeepSleep.
  
- › 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.

# Introduction

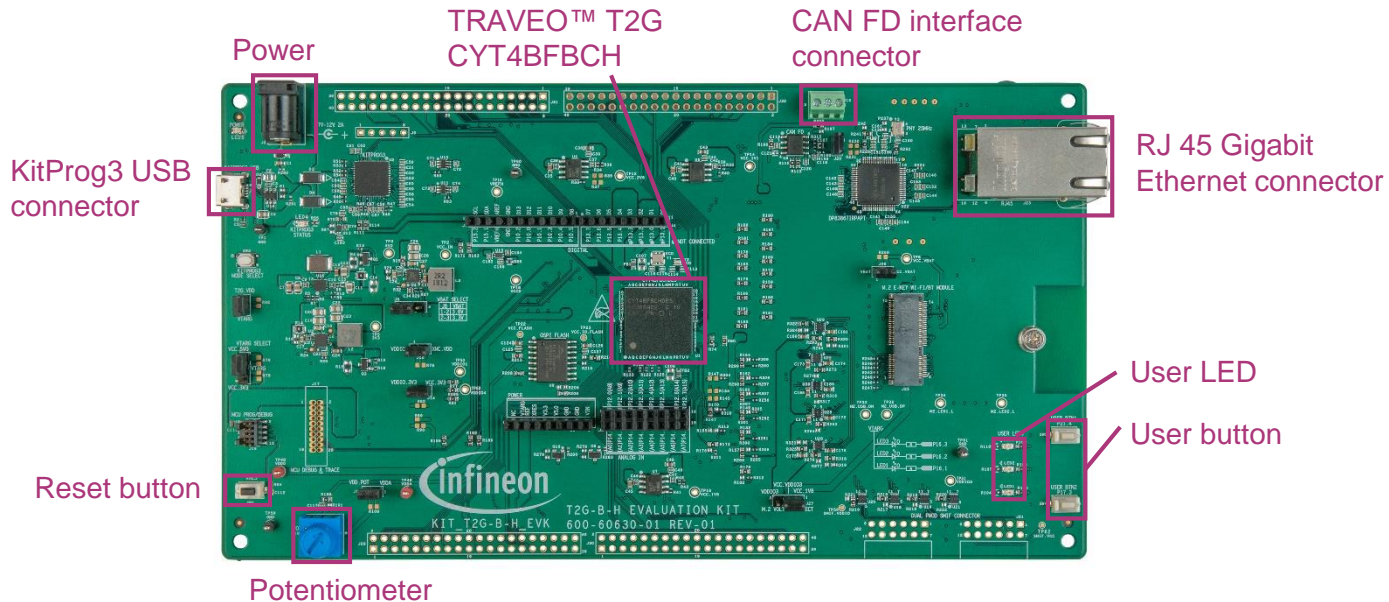
---

## › **GPIO has the following features**

- Analog and digital input and output capabilities
- Eight drive strength modes
- Separate port read and write registers
- Edge-triggered interrupts on rising edge, falling edge, or on both the edges, on all GPIOs
- Slew rate control
- Hold mode for latching the previous state (used to retain the I/O state in DeepSleep mode)
- Selectable CMOS, TTL, and automotive input buffer mode
- Smart I/O provides the ability to perform Boolean functions in the I/O signal path

# 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.



# Implementation

- › This code example uses a GPIO interrupt to wake the Arm® Cortex®-M4 or Cortex®-M7 CPU from DeepSleep. An LED is connected to an output pin; it is used for indicating the current state of the CPU. A blinking LED indicates that the CPU is active. After four successive blinks, the CPU is instructed to enter DeepSleep. Since the GPIO state is retained during DeepSleep, the LED stops blinking and stays OFF to indicate that the CPU is in DeepSleep.
- › An input pin, externally connected to a switch, is configured to generate an interrupt when the switch is pressed. The interrupt triggers the following actions:
  1. Generates a signal that wakes the CPU from DeepSleep
  2. Executes an ISR: Once ISR is executed, a flag is updated, which is used to change the rate at which the LED blinks. With every press of the switch, the LED alternates the blinking interval.

## **Follow these steps to configure this code example:**

- › STDOUT setting
- › GPIO port pin initialization
- › Interrupt configuration
- › Configure the LED
- › Go to DeepSleep

## Implementation (contd.)

---

### STDOUT setting

- › The [cy\\_retarget\\_io\\_init\(\)](#) function initializes the GPIO for UART.
  - Initializes P13.1 as UART TX, P13.0 as UART RX (these pins are connected to the KitProg3 COM port).
  - The serial port parameters change to 8N1 and 115200 baud.

### GPIO port pin initialization

- › The [cyhal\\_gpio\\_init\(\)](#) function initializes the GPIO port pin once.
  - User LED is connected to P16.1 as output and the user button is connected to P21.4 as input.

### Configure the interrupt

- › The [cyhal\\_gpio\\_enable\\_event\(\)](#) function configures the interrupt.
  - When you push the user button, an interrupt occurs and the LED blinks again.
- › The [cyhal\\_gpio\\_register\\_callback\(\)](#) function registers the interrupt service routine (ISR).
  - The ISR name in this sample is ***gpio\_btn\_callback\_data***.

## Implementation (contd.)

### Configure the LED

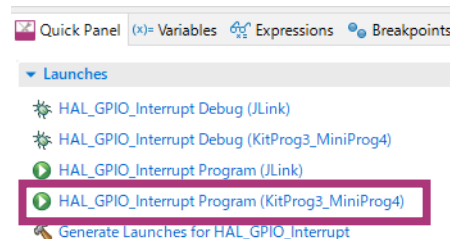
- › If the GPIO interrupt is input by pushing the user button, the ISR is executed and the ***gpio\_intr\_flag*** is set
- › When the flag is set, the main loop of the sample configures the user LED by controlling the port output value to which it is connected
  - The ***cyhal\_gpio\_write()*** function sets the output value of the GPIO pin.
  - The ***cyhal\_system\_delay\_ms()*** function creates the blinking period of the user LED.
    - You can change the blinking period of the user LED by modifying these parameters:
      - ***DELAY\_SHORT\_MS*** (default = “250”: 2Hz)
      - ***DELAY\_LONG\_MS*** (default = “500”: 1Hz)

### Go to DeepSleep

- › After controlling the user LED, the main loop of the sample calls the ***cyhal\_syspm\_deepsleep()*** function to set the CPU to DeepSleep mode
- › Once the user button is pushed after the CPU enters DeepSleep mode, the CPU wakes up from DeepSleep and controls the user LED again

# Compiling and programming

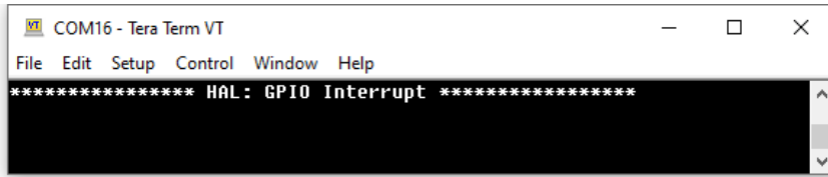
1. Connect to power and USB cable
2. Use Eclipse IDE for ModusToolbox™ software for compiling and programming
3. Compile
  - a) Select the target application project in the Project Explorer
  - b) In the Quick Panel, scroll down, and click “Build HAL\_GPIO\_Interrupt Application” in HAL\_GPIO\_Interrupt (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 “HAL\_GPIO\_Interrupt Program (KitProg3\_MiniProg4)” in Launches





# Run and test

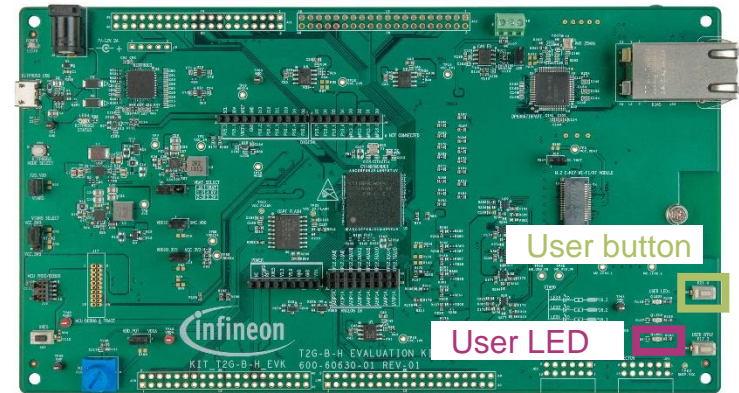
1. After successful programming, the terminal should display the following message:



```

COM16 - Tera Term VT
File Edit Setup Control Window Help
***** HAL: GPIO Interrupt *****
  
```

2. Observe that the user LED (P16.1) blinks four times and then turns OFF, indicating that the CPU has entered DeepSleep.
3. Press the user button (P21.4) to trigger an interrupt. This should wake up the device, causing the LED to resume blinking at a faster rate. (default = 2 Hz) The LED blinks four times and the device enters DeepSleep again.
4. Press the button again to repeat the wakeup cycle, and the LED resumes blinking at a slower rate. (default = 1 Hz)
5. With every interrupt and execution of ISR, the interval of blinking alternates between the slower and faster rates.



# 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](#)

# Revision History

Revision	ECN	Submission Date	Description of Change
**	7782113	2022/07/05	Initial release
*A	7874757	2023/02/20	Updated the title Updated figures in “Compiling and programming”

# Important notice and warnings

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

**Edition 2023-02**

**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](http://www.infineon.com/support)

**Document reference  
002-35566 Rev. \*A**

## **IMPORTANT NOTICE**

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

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](http://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.