TCPWM_Dead_Time_Mode for KIT_T2G-B-H_LITE

Customer training workshop

Q3 2024

Scope of work

- This code example demonstrates the TCPWM dead-time mode function. The PWM line and PWM COMPL pins output the 1 kHz frequency wave and 1 kHz frequency wave with PWM left-align 250 µs in PWM dead-time mode.

- **Device**
  - The TRAVEO™ T2G CYT4BF8CDS device is used in this code example

- **Board**
  - The TRAVEO™ T2G KIT_T2G-B-H_LITE board is used for testing

Introduction

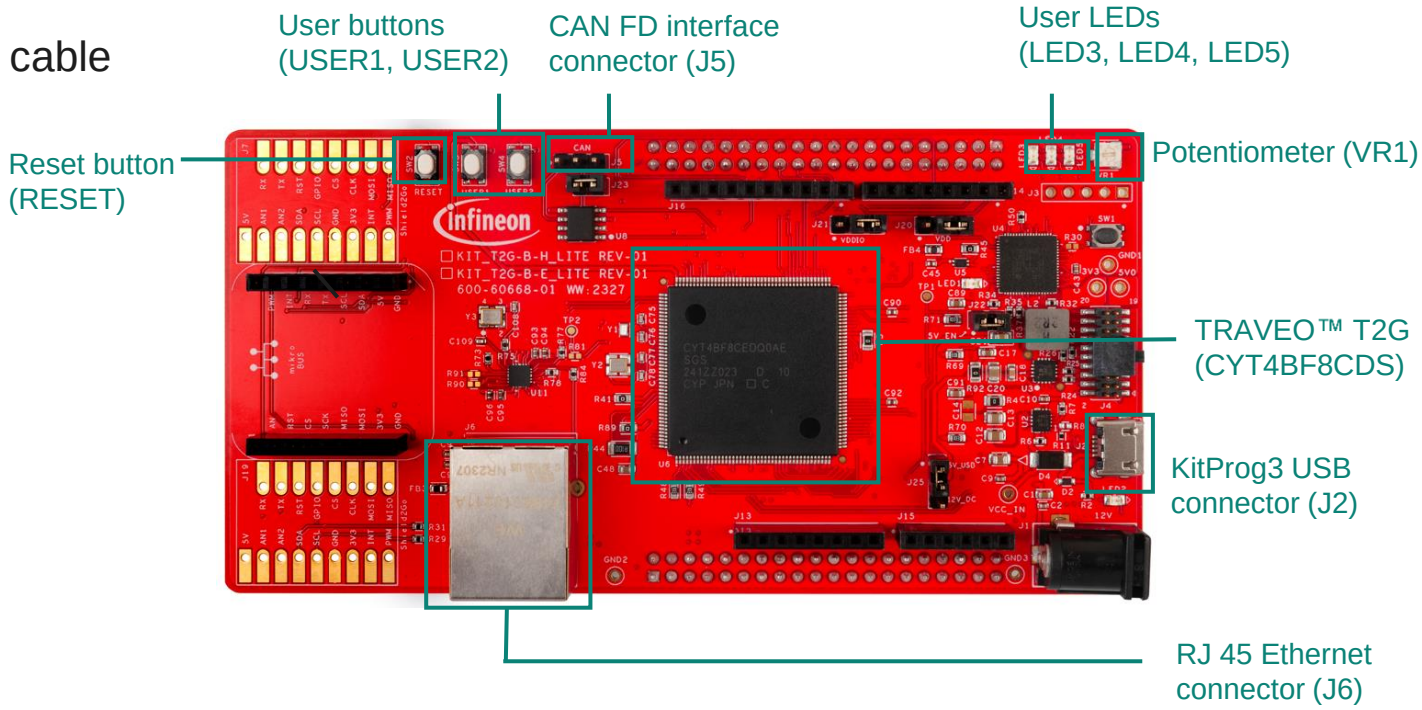
- **TCPWM has the following features:**
  - Supports up to four counter groups (device-specific)
  - Each counter group consists of up to 256 counters (counter group-specific)
  - Each counter can run in one of the following seven function modes:
    - Timer-counter with compare
    - Timer-counter with capture
    - Quadrature decoding
    - Pulse width modulation (PWM)/stepper motor control (SMC) for pointer instruments
    - PWM with dead time/three-phase motor control (Brushless-DC, BLDC)
    - Pseudo-random PWM
    - Shift Register mode
  - 16-bit or 32-bit counters (counter group-specific)
  - Up, down, and up/down counting modes
  - Clock prescaling (division by 1, 2, 4, ... 64, 128)
  - Up to two capture and compare functions (counter group-specific)
  - Double buffering of all Compare/Capture and Period Registers
  - Two output trigger signals for each counter to indicate underflow, overflow, and capture/compare events; they can also directly connect with the line output signal
  - Supports interrupt on:
    - Terminal count - Depends on the mode; typically occurs on overflow or underflow
    - Capture/compare - The count has been captured in the capture registers or the counter value equals the value in the compare register
  - Line out selection feature for stepper motor application including two complementary output lines with dead time insertion

Introduction (contd.)

- Selectable start, reload, stop, count, and two capture event signals for each TCPWM with the rising edge, falling edge, both edges, and level trigger options
- Each counter with up to 254 (device-specific) synchronized input trigger signals and two constant input signals: '0' and ‘1’
- Two types of input triggers for each counter:
  - General-purpose triggers used by all counters
  - One-to-one triggers for specific-counter
- Synchronous operation of multiple counters
- Supports Debug mode

Hardware setup

– This code example has been developed for the KIT_T2G-B-H_LITE board
– Connect the PC to the board using the provided USB cable through the KitProg3 USB connector (J2)



Implementation

- **Code example design**
  - Configure the PWM in dead-time mode with HAL APIs
    Enable the PWM line and PWM COMPL lines to output 1 kHz frequency wave and 1 kHz frequency wave with PWM left-align 250 µs in dead-time mode
  - The application uses the UART to print messages on the UART terminal
    The UART resource initialization and retargeting of standard I/O to the UART port uses the retarget-io library

- **Follow these steps to configure this code example:**
  - TCPWM initialization
  - Enable TCPWM

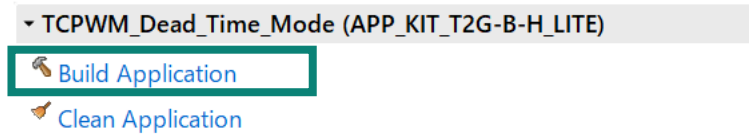
- **TCPWM initialization**
  - Call the ***cyhal_pwm_init_adv()*** function to initialize TCPWM
    - The left-aligned; continuous running PWM signal
    - Assigns normal and inverted outputs to pins
    - Change the dead-time by specifying the ***DEAD_TIME*** value in microseconds (µs)
  - Call the ***cyhal_pwm_set_duty_cycle()*** function to initialize TCPWM
    - PWM frequency is 1 kHz, 50% duty
    - Change these parameters by setting the ***PWM_DUTY_CYCLE*** or ***PWM_TARGET_FREQUENCY***

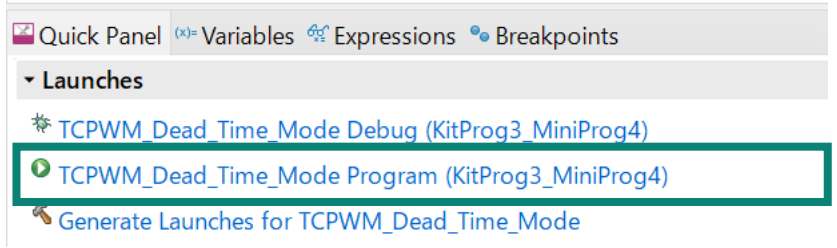
- **Enable TCPWM**
  - Call the ***cyhal_pwm_start()*** function to start the TCPWM output

Compiling and programming

1. Connect to power and USB cable
2. Use Eclipse IDE for ModusToolbox™ software for compiling and programming
3. For compilation:
   a. Select the target application project in the Project Explorer
   b. In the Quick Panel, scroll down, and click **Build Application** in TCPWM_Dead_Time_Mode (APP_KIT_T2G-B-H_LITE)
4. For programming:
   a. Select the target application project in the Project Explorer
   b. In the Quick Panel, scroll and click **TCPWM_Dead_Time_Mode Program (KitProg3_MiniProg4)** in the Launches

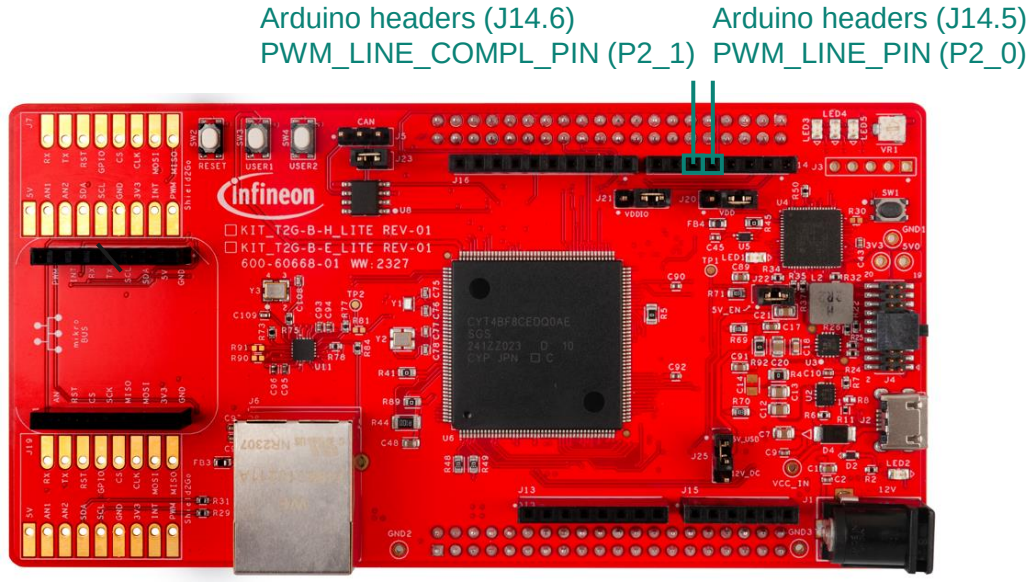
KitProg3 USB connector (J2)

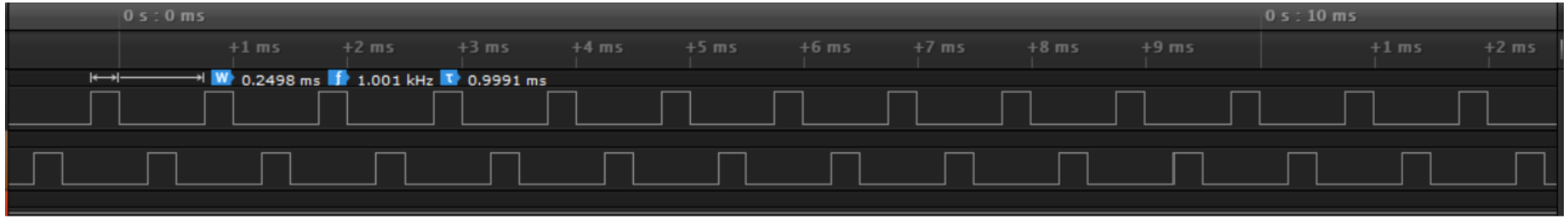




Run and test

1. After successful programming, the application starts automatically.
2. Use the logic analyzer to observe the PWM output wave.
3. Connect the PWM_LINE_PIN (P2_0) and PWM_LINE_COMPL_PIN (P2_1) of Arduino headers (J14.5 and J14.6), and GND lines of the hardware board to the logic analyzer.





References

- **Datasheet**
  - CYT4BF TRAVEO™ T2G 32-bit Automotive MCU based on Arm® Cortex®- M7 dual
- **Architecture reference manual**
  - TRAVEO™ T2G Automotive MCU body controller high architecture reference manual
- **Registers reference manual**
  - TRAVEO™ T2G Automotive MCU: TVII-B-H-8M body controller high registers reference manual
- **PDL/HAL**
  - Peripheral driver library (PDL)
  - Hardware abstraction layer (HAL)
- **Training**
  - TRAVEO™ T2G training

Revision History

Revision	ECN	Submission Date	Description of Change
**	7782103	2022/07/05	Initial release
*A	8080724	2024/10/03	Replaced development board from KIT_T2G-B-H_EVK to KIT_T2G-B-H_LITE