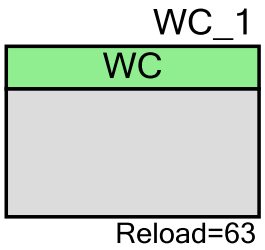
Watch Counter (PDL_WC)

1.0



Features

- 6-bit down counter
- Underflow interrupt request capability
- One of the four types of clock can be selected (WCCK0..3)

General Description

The Peripheral Driver Library (PDL) Watch Counter component is a timer that counts down starting from a specified value. It generates an interrupt request at the time that the 6-bit down counter enters an underflow condition.

The Watch Counter wakes up the microcontroller from low power consumption mode. The clock source can be selected from the main clock, the sub clock, the built-in high-speed CR clock or the built-in low-speed CR clock.

This component uses firmware drivers from the PDL_WC module, which is automatically added to your project after a successful build.

When to Use a PDL_WC Component

Use the PDL_WC component when you need to generate an interrupt after passed some period of time.

Quick Start

1. Drag a PDL_WC component from the Component Catalog FMx/Digital/Watch Counter folder onto your schematic. The placed instance takes the name WC_1.
2. Double-click to open the component's Configure dialog.
3. On the "**Basic**" tab set the following parameters:
    - reload value
    - select clock source
    - initialize interrupt if needed, and callback function

4. Build the project to verify the correctness of your design. This will add the required PDL modules to the Workspace Explorer and generate configuration data for the WC_1 instance.
5. In the *main.c* file, initialize the peripheral and start the application.

```
/* Use this structure to configure input and output prescaler */
stc_wc_pres_clk_t stcWcPresClk = { WcPresInClkMainOsc, WcPresOutClkArray0 };

Wc_Pres_SelClk(&WC_1_HW, &stcWcPresClk);
Wc_Init(&WC_1_HW, &WC_1_Config);
Wc_Pres_EnableDiv(&WC_1_HW);
Wc_EnableCount(&WC_1_HW);
```

6. Build and program the device.

Component Parameters

The PDL_WC component Configure dialog allows you to edit the configuration parameters for the component instance.

Basic Tab

This tab contains the component parameters used in the basic peripheral initialization settings.

Parameter Name	Description
bIrqEnable	Enable interrupt on counter underflow.
bTouchNvic	Update the NVIC with the WC interrupt.
enCntClk	Select the clock source from prescaler to Watch Counter. Note – you must set up the prescaler with Wc_Pres_SelClk() before using the watch counter.
pfnIrqCallback	Callback function for counter underflow interrupts. Note: this generates a declaration only - USER must implement the function.
u8ReloadValue	6-bit reload value for down counter.

Component Usage

After a successful build, firmware drivers from the PDL_WC module, are added to your project in the pdl/drivers/wc folder. Pass the generated data structures to the associated PDL functions in your application initialization code to configure the peripheral.

Generated Data

The PDL_WC component populates the following peripheral initialization data structure(s). The generated code is placed in C source and header files that are named after the instance of the component (e.g. *WC_1_config.c*). Each variable is also prefixed with the instance name of the component.

Data Structure Type	Name	Description
stc_wc_config_t	WC_1_Config	Configuration structure.

Once the component is initialized, the application code should use the peripheral functions provided in the referenced PDL files. Refer to the PDL documentation or the list of provided API functions. To access this document, right-click on the component symbol on the schematic and choose "**Open API Documentation…**" in the drop-down menu.

Preprocessor Macros

The WC component generates the following preprocessor macro(s). Note that each macro is prefixed with the instance name of the component (e.g. "WC_1").

Macro	Description
WC_1_HW	Hardware pointer to the block instance in the device. This should be used in all API calls when specifying the block to access.

Data in RAM

The generated data may be placed in flash memory (const) or RAM. The former is the more memory-efficient choice if you do not wish to modify the configuration data at run-time. Under the **Built-In** tab of the Configure dialog set the parameter CONST_CONFIG to make your selection. The default option is to place the data in flash.

Interrupt Support

If the PDL_WC component is specified to trigger interrupts, it will generate the callback function declaration that will be called from the WC ISR. The user is then required to provide the actual callback code. If a null string is provided the struct is populated with zeroes and the callback declaration is not generated. In that case it is the user's responsibility to modify the struct in firmware.

The component generates the following function declarations.

Function Callback	Description
WC_1_WcIrqCallback	Interrupt callback function.

Code Examples and Application Notes

There are numerous code examples that include schematics and example code available online at the Cypress Code Examples web page.

Cypress also provides a number of application notes describing how FMx devices can be integrated into your design. You can access the Cypress Application Notes search web page at www.cypress.com/appnotes.

Resources

The PDL_WC component uses the WC Watch Counter peripheral block.

References

- FM0+ Family of 32-bit ARM® Cortex®-M0+ Microcontrollers Peripheral Manuals
- Cypress FM0+ Family of 32-bit ARM® Cortex®-M0+ Microcontrollers

Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
1.0.a	Minor datasheet edits.	
1.0	Initial Version	