CapSense Slider and Button

About this document

Scope and purpose

This code example demonstrates the implementation of the CapSense® linear slider and button for controlling the LEDs. The CapSense Component sends the information from the linear slider and button over I2C to the CapSense Tuner tool. The response to the CapSense button can be seen on the green LED glow state. The CapSense linear slider touch position effect can be seen on the red LED brightness.

Requirements

Tool: PSoC® Creator™ 4.4

Programming Language: C (Arm® GCC 5.4.1 and Arm MDK 5.22)

Associated Parts: PSoC 4500S

Related Hardware: CY8CKIT-045S PSoC 4500S Pioneer Kit

Table of contents

About this document **1**
Table of contents **1**
1 Overview **2**
2 Hardware and Software Setup **3**
2.1 Hardware Setup 3
2.2 Software Setup 3
3 Operation **4**
4 Design and Implementation **5**
4.1 Components and Settings 6
5 Reusing This Example **11**
References **12**
Revision history **13**

1 Overview

This code example demonstrates the operation of the CapSense CSD linear slider with five segments and one CSX CapSense button. Data from the CapSense linear slider and button is sent to the CapSense Tuner using I2C communication. Touch the button to turn the green LED on/off. Use the linear slider touch position to control the red LED brightness.

Note: *See the CapSense Tuner section of the **CapSense Component Datasheet** to know how to use Tuner tool.*

2 Hardware and Software Setup

2.1 Hardware Setup

This code example is set up to use CY8CKIT-045S by default, but it can be used with other PSoC 4 devices. To see how to migrate this code example to a new device, see **Reusing This Example**.

2.2 Software Setup

No software setup required.

3 Operation

1. Connect the CY8CKIT-045S board into your computer's USB port.
2. Build the project and program it onto the PSoC 4500S device. Choose **Debug > Program**. For more information on device programming, see PSoC Creator Help.
3. Touch the CapSense button (CSB1) on the kit. The green LED connected to P0[0] will turn on. Removing finger from button, the green LED will turn off.
4. Slide your finger slowly up and down the slider and confirm that the red LED connected to P1[6] changes brightness.

4 Design and Implementation

Figure 1 shows the PSoC Creator schematics for the "CE230205_CapSense_Slider_and_Button" code example. This code example uses CapSense, EZI2C Slave (SCB mode), PWM, Pin, and Clock components. In this code example, the following functions are performed:

1. Initialize and start all hardware components.
2. Link the communication data buffer for the EZI2C Component to the CapSense data structure.
3. The CapSense Component scans all widgets, and after complete, processes all scanned CapSense widgets.
4. Control the LEDs through CapSense slider touch positions and button touch status.
5. Send all data to the tuner.
6. Scan CapSense widgets before restarting the loop.

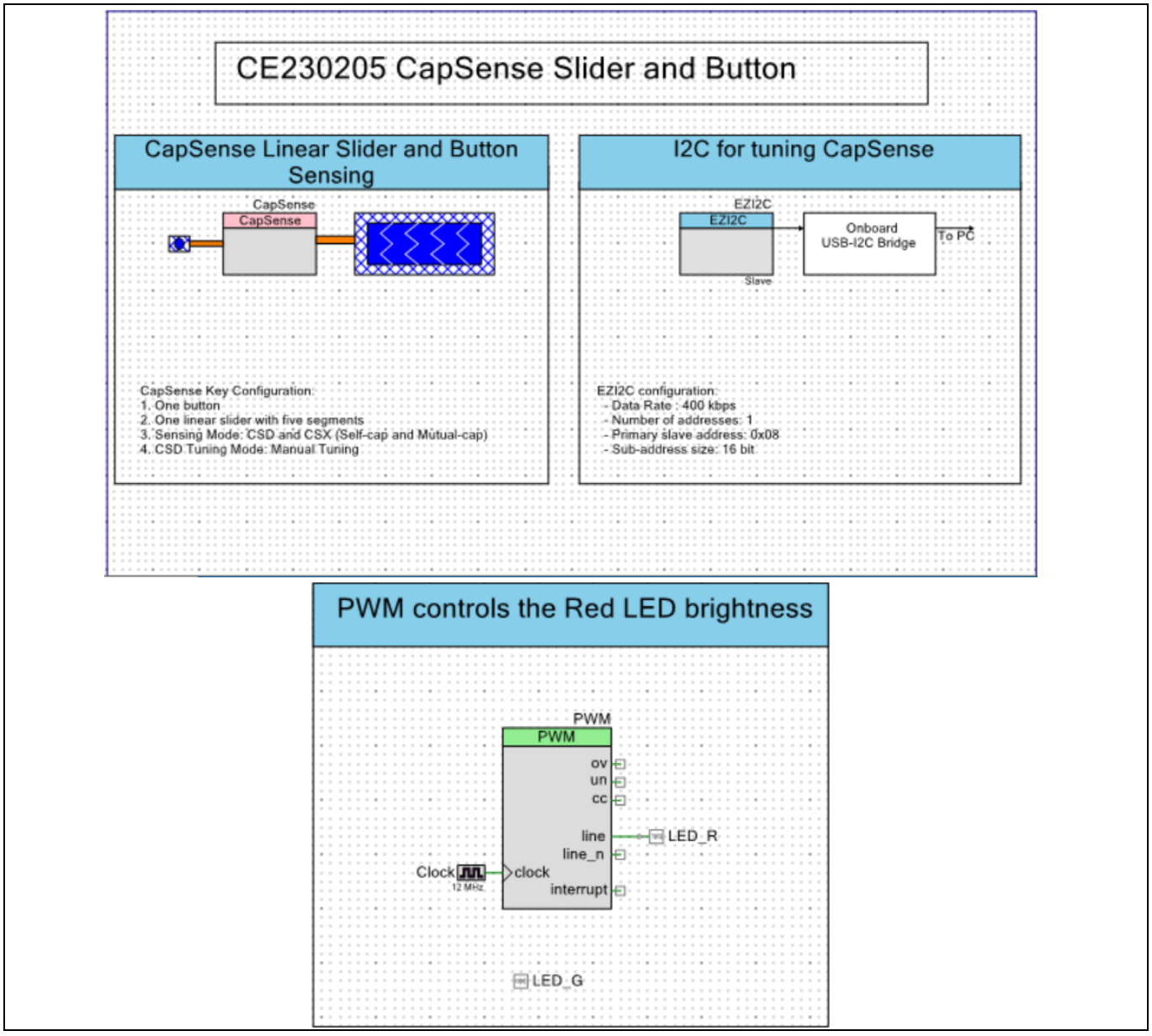


Figure 1 PSoC Creator Schematics for CE230205_CapSense_Slider_and_Button

4.1 Components and Settings

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 PSoC Creator Components

Component	Instance Name	Purpose	Non-default Settings
CapSense	CapSense	Scan 5-segement linear slider and one CapSense button	See **Figure 2**, **Figure 3**, **Figure 4**
EZI2C Slave (SCB Mode)	EZI2C	Transmits data from the kit to the tuner	See **Figure 5**
PWM	PWM	Controls the duty cycle of the LED_R	See **Figure 6**
Clock	Clock	Drive the PWM at 12MHz	See **Figure 7**
Digital Output Pin	LED_R	Drive the PWM output to red LED	See **Figure 8**
	LED_G	Firmware controlled digital output pin that control green color LED	See **Figure 9**

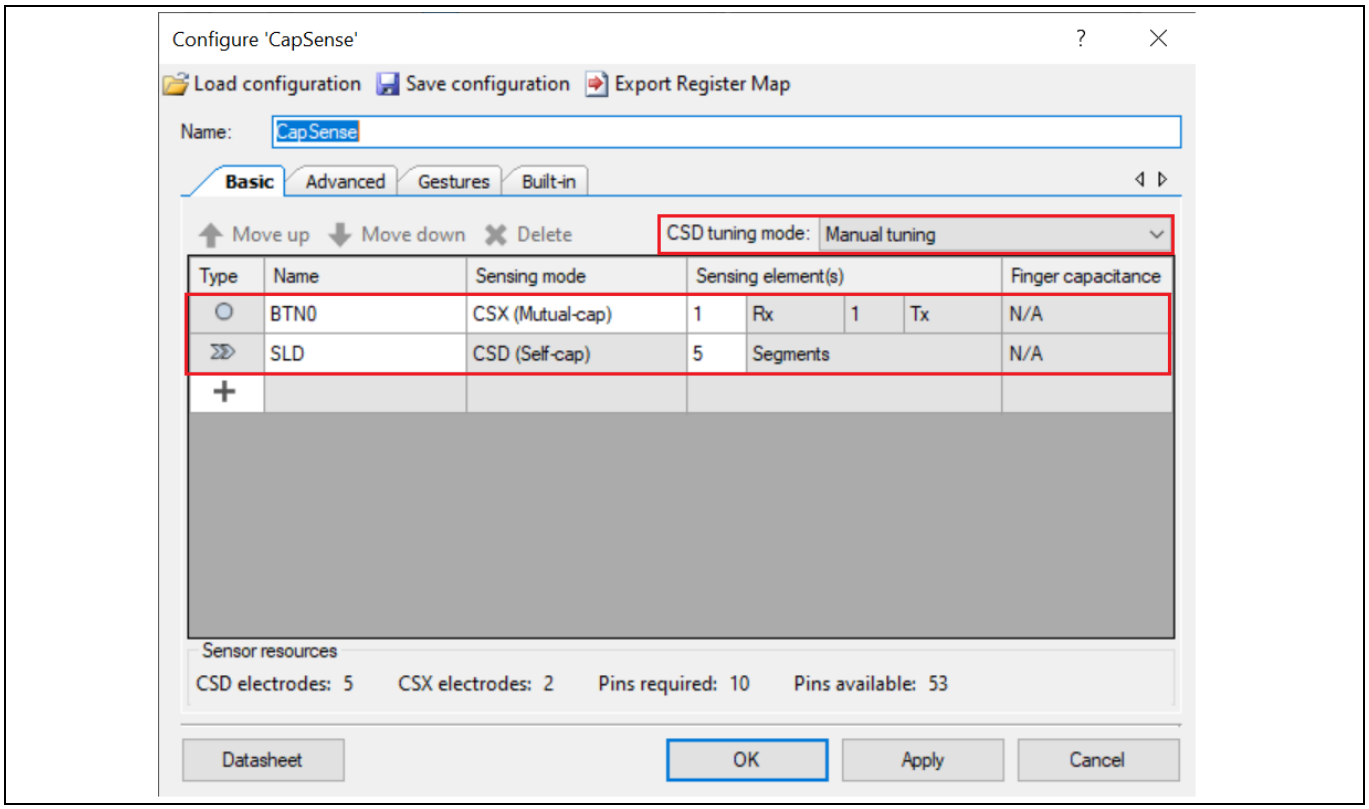


Figure 2 CapSense Component Setting

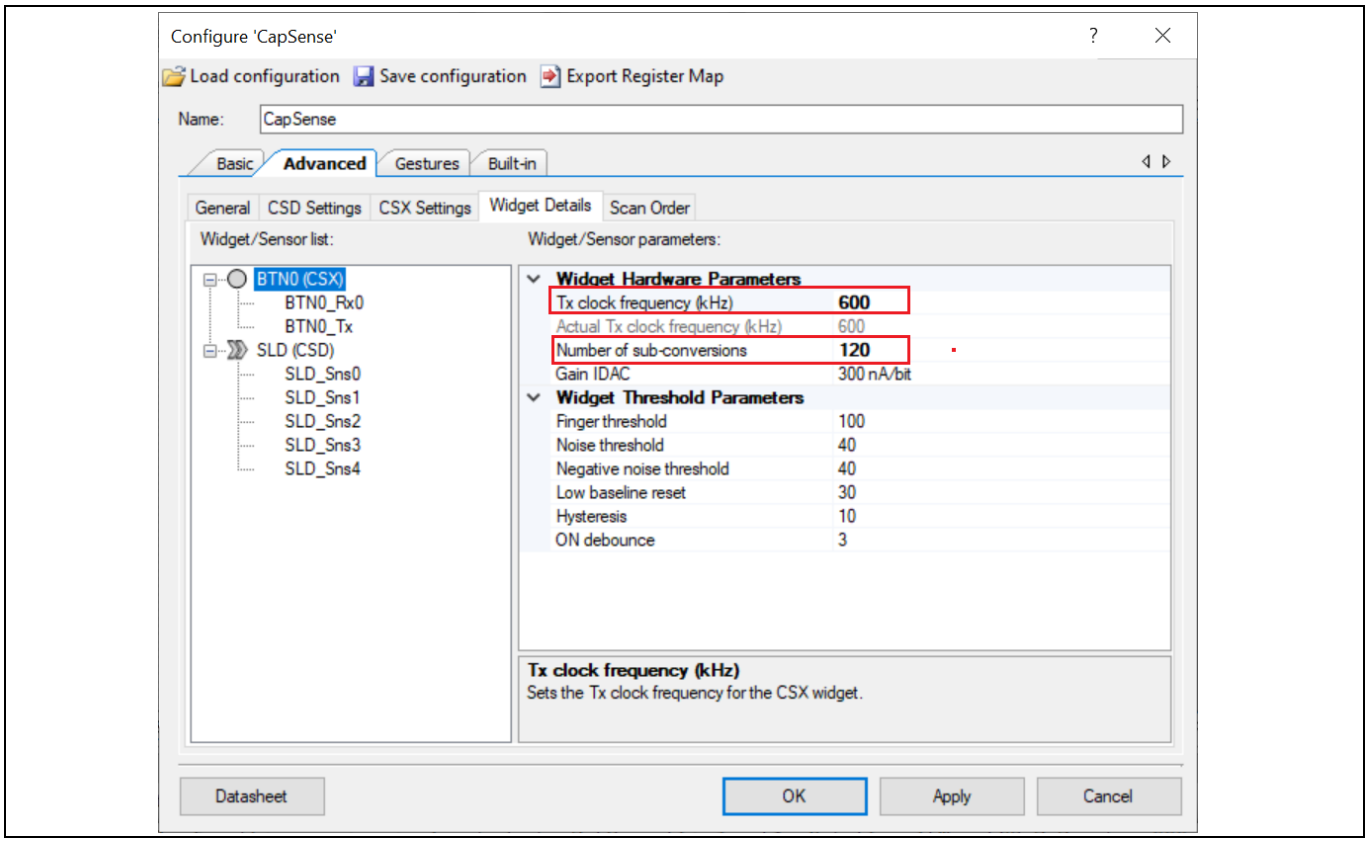


Figure 3 Button Widget Details

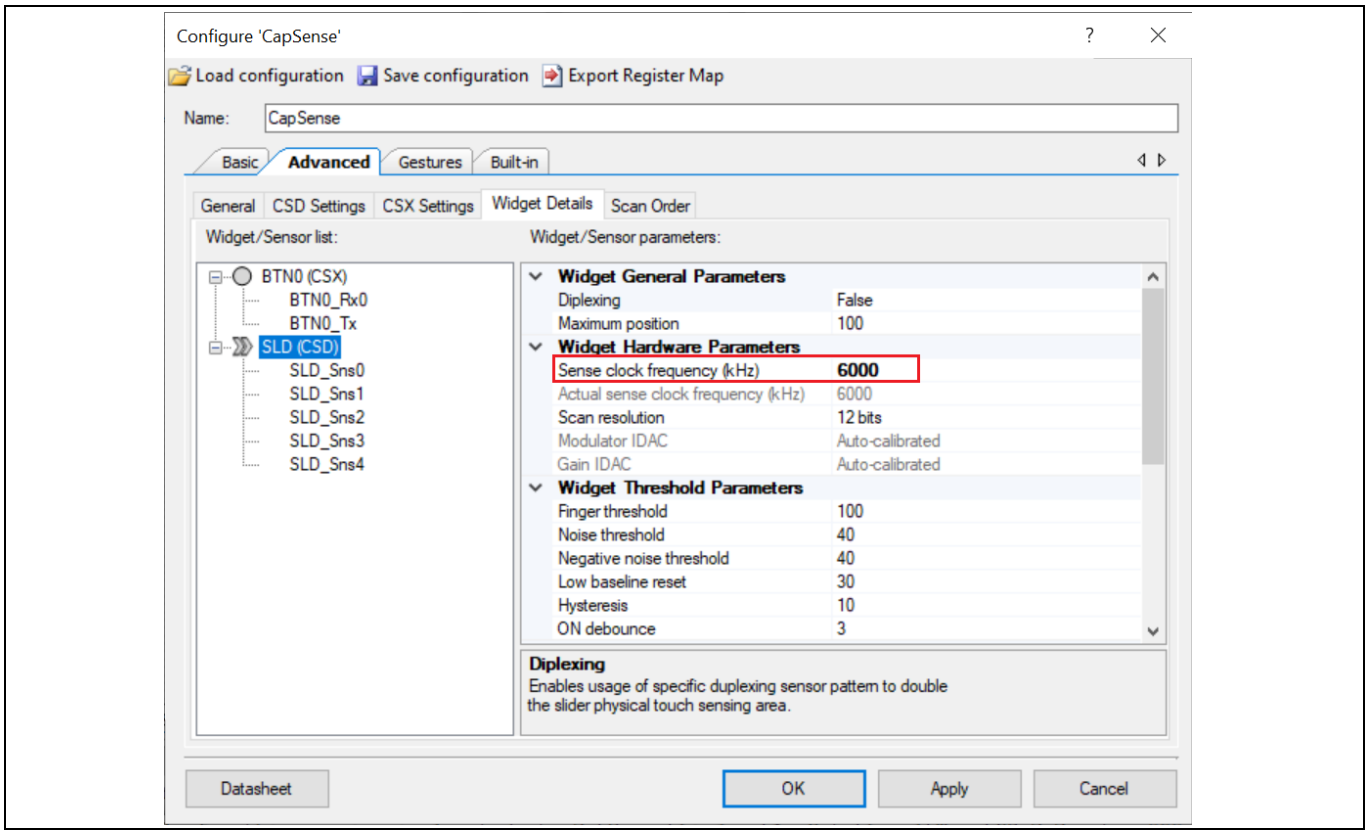


Figure 4 Linear Slider Widget Details

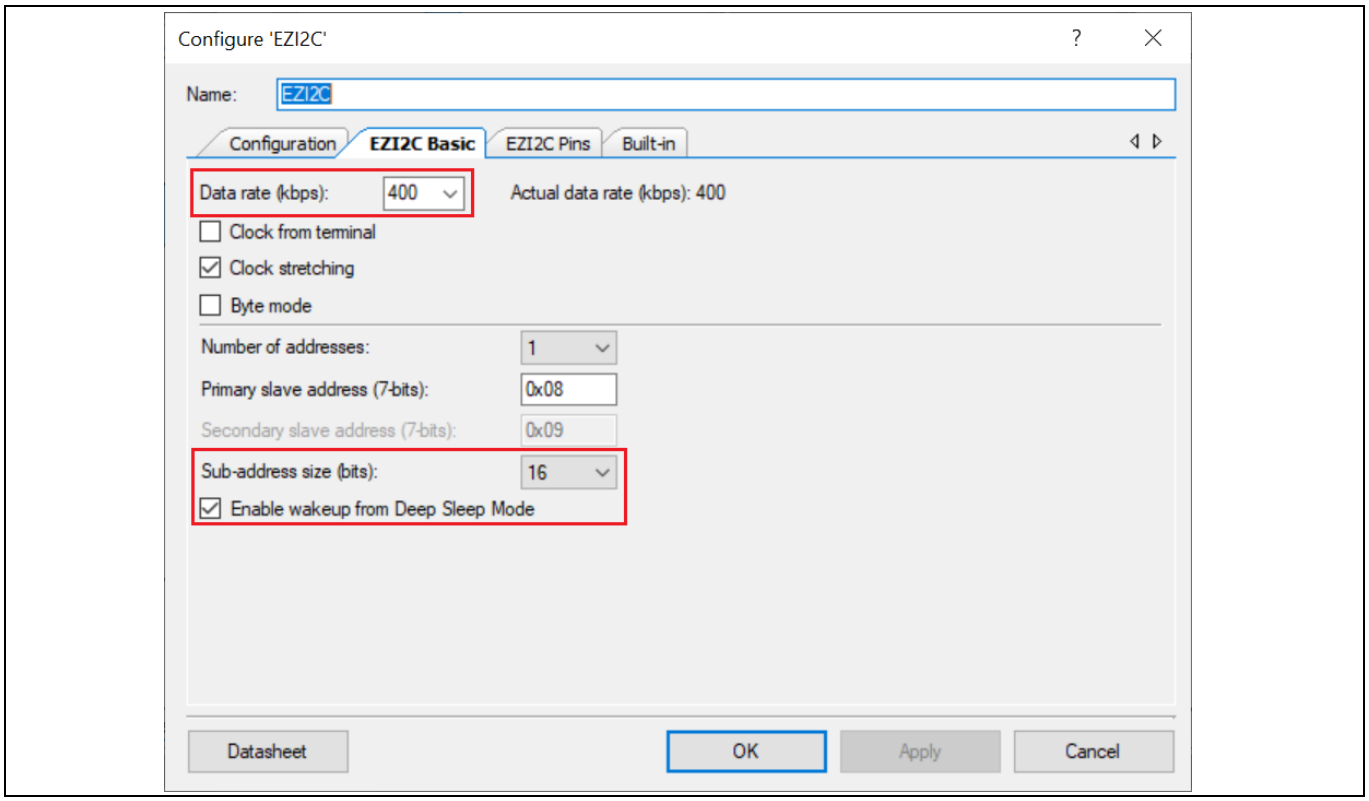


Figure 5 EZI2C Component Setting

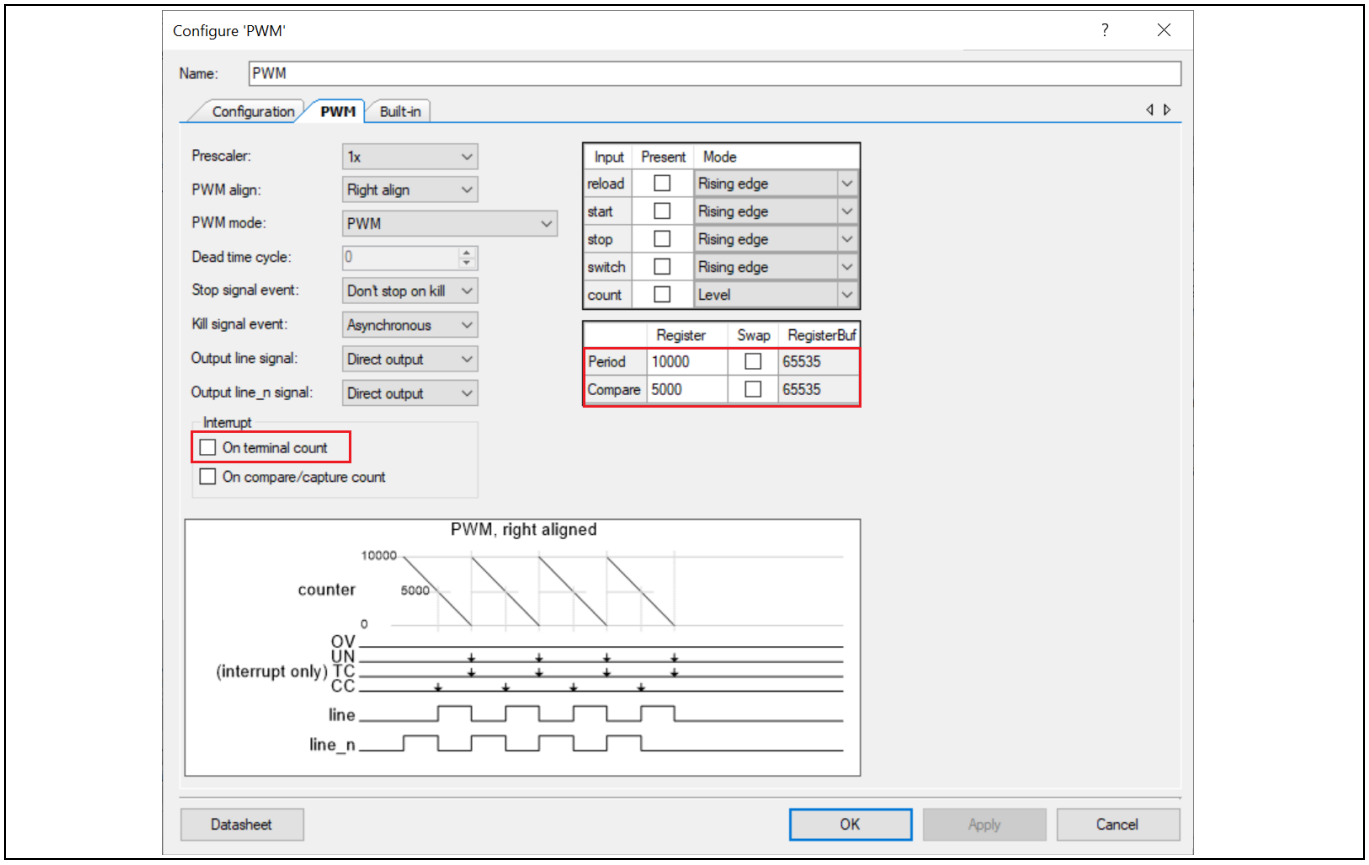


Figure 6 PWM Component Setting

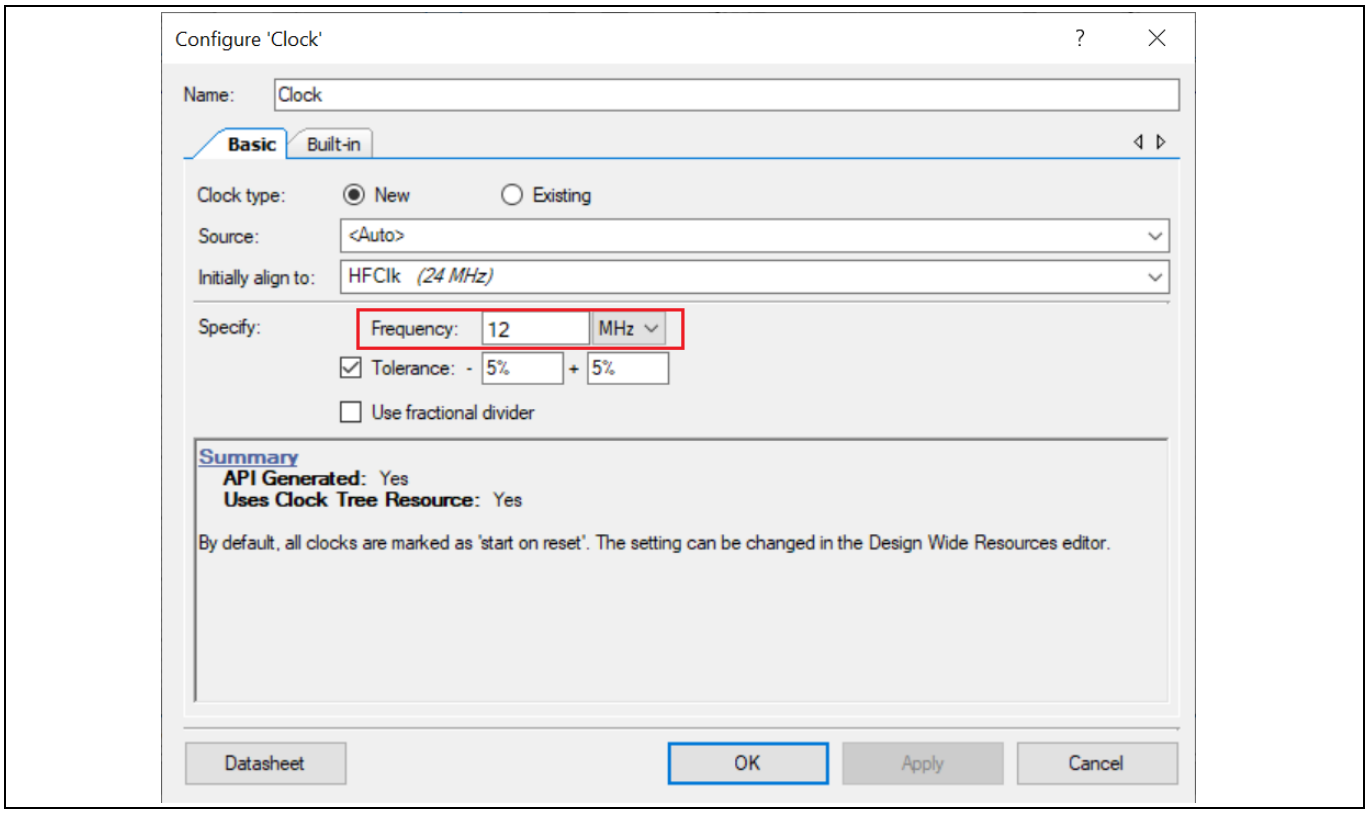


Figure 7 PWM Clock Setting

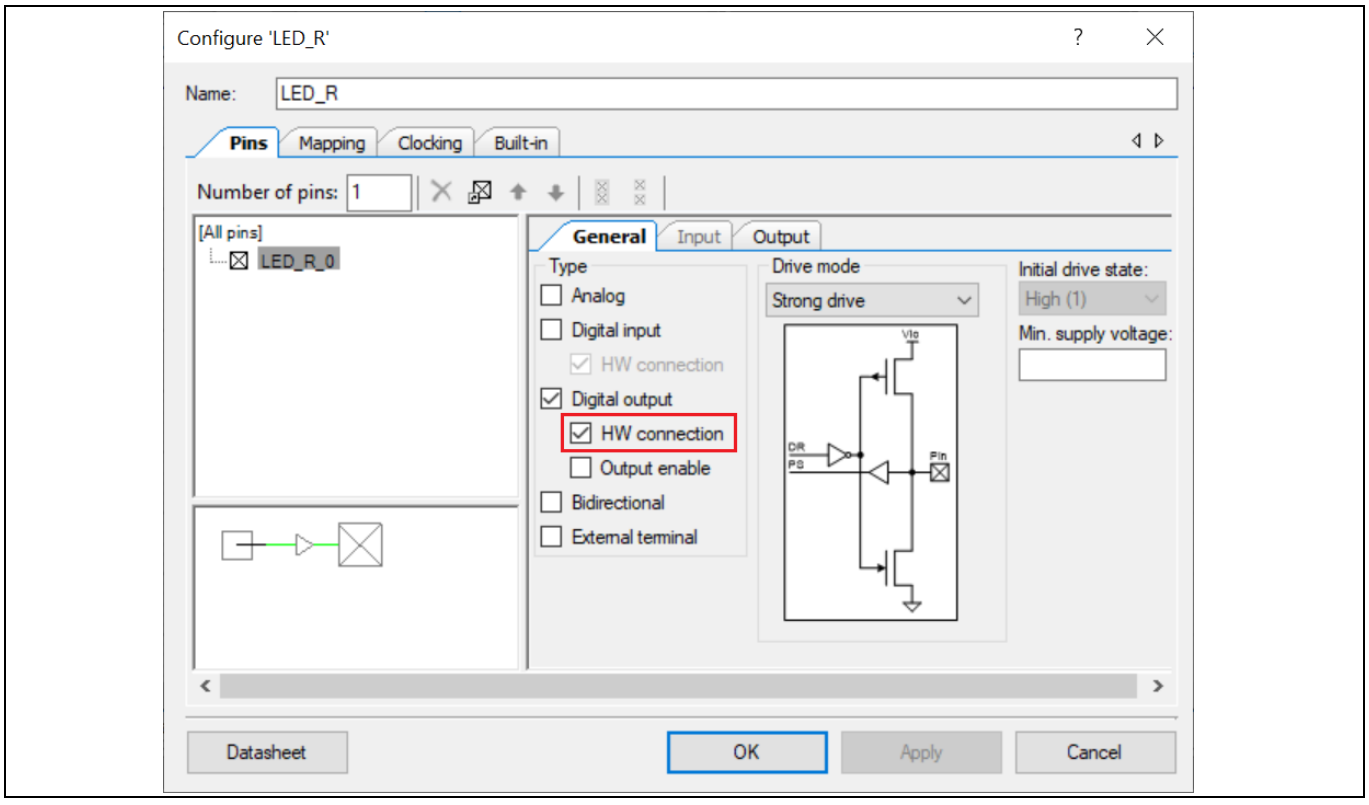


Figure 8 LED_R Setting

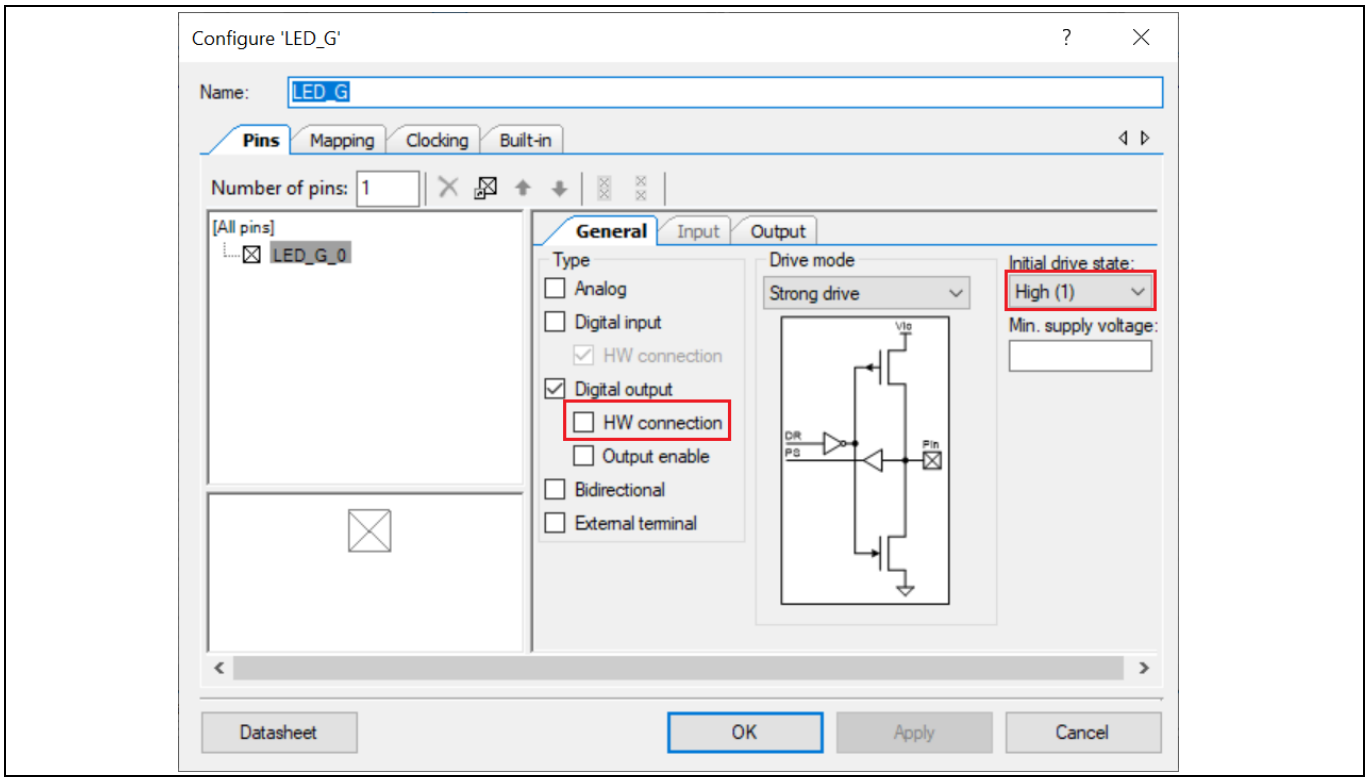


Figure 9 LED_G Setting

For information on the hardware resources used by a Component, see the Component datasheet.

Table 2 shows the pin assignment for the project done through the Pins tab in the Design Wide Resource window.

Table 2 Pin Assignment for the CE230205_CapSense_Slider_and_Button

Pin Name	CY8CKIT-045S (CY8C4548AZI-S485)
\CapSense:CintA\	P4[2]
\CapSense:CintB\	P4[3]
\CapSense:Cmod\ (Cmod)	P4[1]
\CapSense:Rx[0]\ (BTN0_Rx0)	P0[4]
\CapSense:Sns[0]\ (SLD_Sns0)	P4[5]
\CapSense:Sns[0]\ (SLD_Sns1)	P4[6]
\CapSense:Sns[0]\ (SLD_Sns2)	P4[7]
\CapSense:Sns[0]\ (SLD_Sns3)	P7[0]
\CapSense:Sns[0]\ (SLD_Sns4)	P7[1]
\CapSense:Tx[0]\ (BTN0_Tx0)	P4[4]
\EZI2C:scl\	P1[0]
\EZI2C:sda\	P1[1]
LED_G	P0[0]
LED_R	P1[6]

5 Reusing This Example

This example is designed for the supported CY8CKIT-045S board. To port the design to a different PSoC 4 device or kit, change the target device using the Device Selector and update the pin assignments in the Design Wide Resources Pins settings as needed.

In some cases, a resource used by a code example is not supported on another device. In that case, the example will not work. If you build the code targeted at such a device, you will get errors. See the device datasheet for information on what a particular device supports.

References

For a comprehensive list of PSoC 3, PSoC 4, and PSoC 5LP resources, see **KBA86521** in the Cypress community.

Application Notes

[1] **AN79953** - Getting Started with PSoC 4: Describes PSoC 4 devices and how to build your first PSoC Creator project

[2] **AN85951** - PSoC4 and PSoC 6 MCU CapSense Design Guide: Describes how to design capacitive touch sensing applications with PSoC 4 and PSoC 6 MCU devices

PSoC Creator Component Datasheets

[3] **CapSense**: Supports various interfaces such as Button, Matrix Buttons, Slider, Touchpad, and Proximity Sensor

[4] **EZI2C Slave**: Supports one or two address decoding with independent memory buffers

[5] **PWM**: Supports terminal count output for 8-, 16-, 24-, and 32-bit sequence lengths

[6] **Pins**: Supports connection of hardware resources to physical pins

[7] **Clock**: Supports clock component

Device Documentation

[8] **PSoC 4500S Datasheets**

[9] **PSoC 4500S Technical Reference Manuals**

Development Kit Documentation

[10] **CY8CKIT-045S PSoC 4500S Pioneer Kit**

Revision history

Document version	Date of release	Description of changes
**	2020-11-18	New code example

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

Edition 2020-11-18
Published by
Infineon Technologies AG
81726 München, Germany

© 2020 Infineon Technologies AG. All Rights Reserved.

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

Document reference
002-30205 Rev. **

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.