Objective

This example demonstrates the operation of the PSoC® 6 MCU Serial Control Block (SCB) in I2C Slave mode. Two projects show the use of Peripheral Driver Library (PDL) functions to receive data from an I2C Master in different modes.

Overview

The SCB in I2C Slave mode accepts command packets to control the color of an RGB LED. The I2C Slave updates its read buffer with a status packet in response to the accepted command. Two projects in this example are I2C Slave using callback and polling method.

Requirements

Tool: PSoC Creator™ 4.2, Bridge Control Panel 1.17.0

Programming Language: C (Arm® GCC 5.4-2016-q2-update, Arm MDK 5.22)

Associated Parts: All PSoC 6 MCU parts

Related Hardware: CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

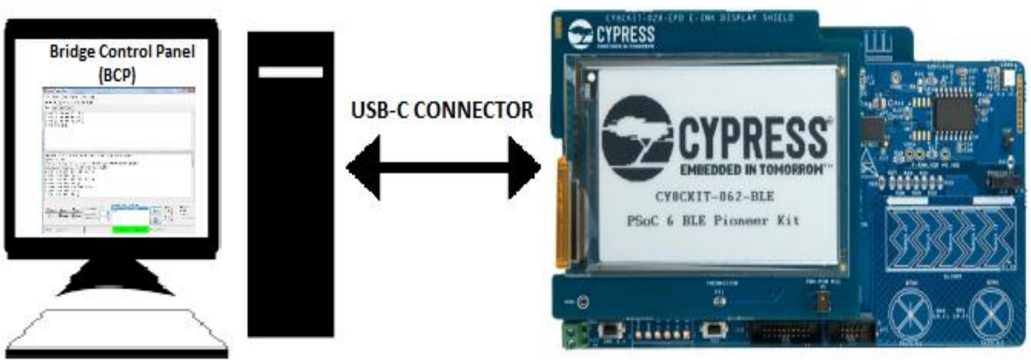
Design

The Figure 1 shows the high-level implementation of this code example. This code example implements an I2C Slave using an SCB Component. The I2C Slave is configured with a 5-bytes write buffer, which can be accessed by the I2C Master to write commands. In addition, the 3-bytes read buffer is configured to read status of the Slave by the Master. The Bridge Control Panel software as shown in Figure 1 and Figure 4 is used as the I2C Master.

The first byte in the write buffer contains the Start of Packet (SOP) value, the next three bytes contain red, green, and blue LED's TCPWM compare value, and the fifth byte in the write buffer is End of Packet (EOP). The Slave updates the Master's read buffer with the status packet. The first byte of the status packet is SOP, the second byte contains the status where the value 0x00 means success and 0xFF means failure for the command data sent by the Master.

To control the color of an RGB LED, three PWMs with a period value of 255 (~195 kHz) are used. The duty cycle of the PWMs are controlled in the firmware and specified by the I2C Master. Changing the duty cycle of the three PWM's signal will result in a change in the RGB LED color.

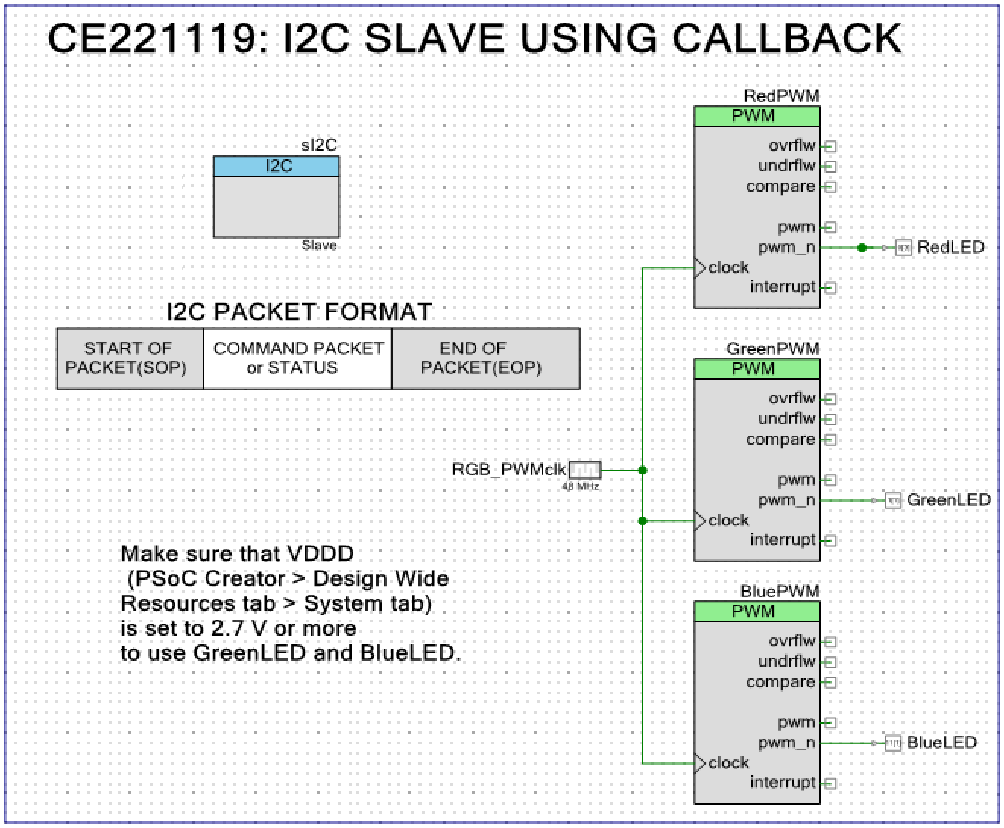
Figure 1. Interface Diagram for I2C Slave Example



I2C Slave Using Callback Method

Figure 2 shows the PSoC Creator project schematic for the callback method. In the callback method, data write and read complete events from the Master are handled through interrupts. PDL functions are used to configure the SCB Component to act as an I^{2}C Slave, and to configure its relevant interrupts to handle data write and read complete events by the Master.

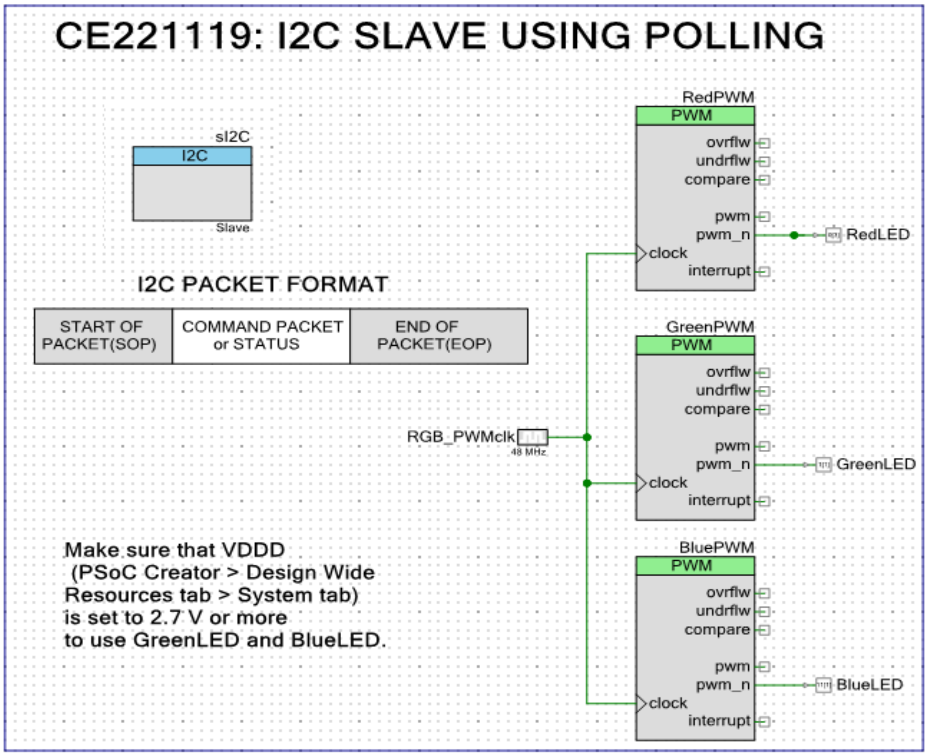
Figure 2. I^{2}C Slave Schematic for Callback Method



I^{2}C Slave Using Polling Method

Figure 3 shows the PSoC Creator project schematic for the polling method. In the polling method, the system checks the status of the I^{2}C Slave continuously. PDL functions are used to configure the SCB Component to act as an I^{2}C Slave and to know the status of the Slave. Through the status information, the Slave gets to know whether the Master has completed writing data to the write buffer or completed reading status from the read buffer. If the Master has completed writing data to the write buffer, the data written is checked for SOP and EOP, and if it is correct, the red, green and blue LED's PWM compare value is updated with the data written by the Master.

Figure 3. I2C Slave Schematic for Polling Method



Design Considerations

This code example is designed to run on CY8CKIT-062-BLE with PSoC 6 MCU. To port the design to other devices and kits, change the target device in Device Selector, and change the pin assignments in the *cydwr* settings.

I2C Slave projects designed in this example can be used by Master devices located on a different or the same board. In this example, the Master is a host PC running the Cypress Bridge Control Panel (BCP) software shown in Figure 4.

Hardware Setup

The code example works with the default settings on the CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit. If the settings are different from the default values, see the 'Selection Switches' table in the kit guide to reset to the default settings.

Table 4 lists the PSoC Creator pin connection settings required on CY8CKIT-062-BLE with PSoC 6 MCU.

Software Setup

This section describes how to set up the BCP software as the I2C Master to send commands to an I2C Slave.

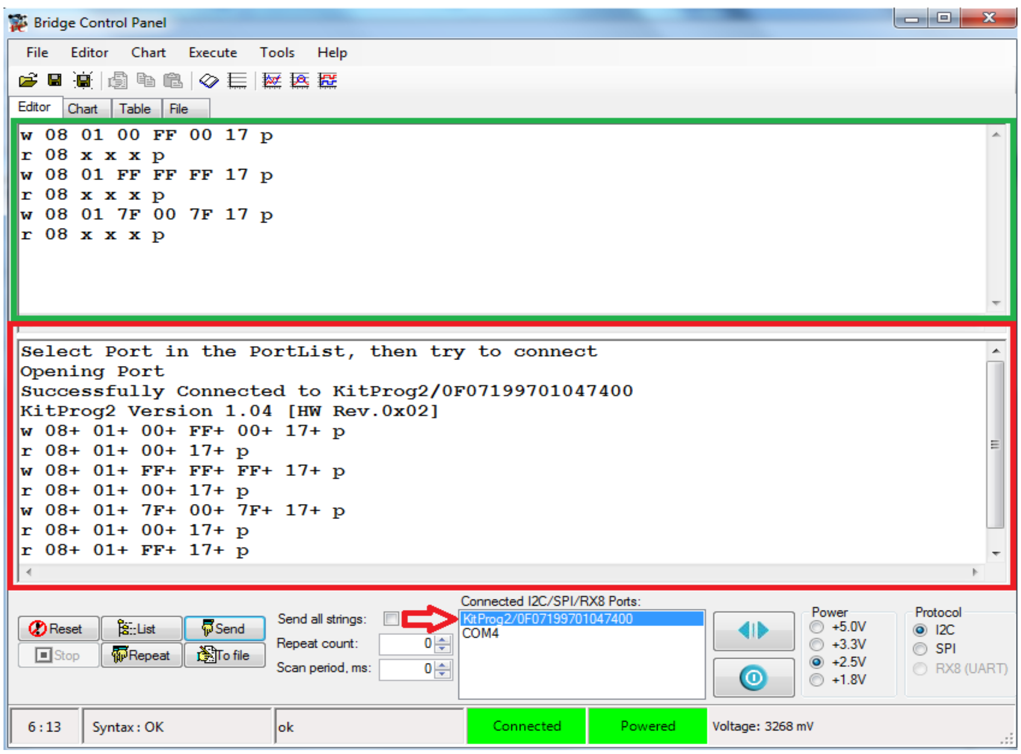
The BCP software is installed automatically as part of the PSoC Creator installation. Follow these steps to configure BCP:

1. Open BCP from **Start** > **All Programs** > **Cypress** > **Bridge Control Panel <version>** > **Bridge Control Panel <version>.** For explanation regarding symbols used for writing and reading data from the I2C buffer, click **Help** in BCP, select Communication in Help contents and select Entering Commands topic. Explanation is given in the BCP Help contents and in Table 1 for symbol 'w', 'r', 'x' and 'p' used in the commands shown in Figure 4.
2. Select **KitProg2/<serial number>** under **Connected I2C/SPI/RX8 Ports** (see Figure 4). Note that the PSoC 6 BLE Pioneer Kit must be connected to the USB port of your computer.
3. Select **Tools** > **Protocol Configuration**, navigate to the **I2C** tab, and set the **I2C speed** to '100 kHz', and click **'OK'**. BCP is now ready for sending the command packets and reading the status from the Slave.

Table 1. Symbols for Commands

Symbol Type	Symbol	Purpose
Starting symbol of command	w	Define the start of "write data" command. Generate Start condition on the I^{2}C bus. After this start symbol the 7-bit address of the I^{2}C Slave should follow.
Starting symbol of command	r	Defines the start of "read data" command. Generates the Start condition on the I^{2}C bus. Used instead of "s" symbol for start definition. After this start symbol the 7-bit address of the I^{2}C Slave should follow.
Read data	x	Reserved symbol, which means that 1 byte of data should be read. Used only in "read data" commands.
Stop condition	p	Generates Stop condition on the I^{2}C bus. This symbol can occur at the end of current command. The absence of the Stop condition symbol at the end of the current command means generation of Restart condition for the following command.

Figure 4. Bridge Control Panel



Operation

1. Open the CE221119 code example in PSoC Creator.
2. Connect the CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit to your computer using the USB-C cable provided.
3. Build and program each I^{2}C Slave project into the CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit. For more information on building a project or programming a device, see PSoC Creator Help.
4. Configure the BCP software as described in Software Setup.
5. In the **Editor** tab of BCP marked in **green** color shown in Figure 4, type the command to be written or data to be read from the I^{2}C buffer. After typing the correct command format, click **Send**.
6. Data written and read from the Slave can be seen in the editor column of Figure 4 marked in **red** color.

7. The command format that is used to write the data to the Slave if BCP is used as the I²C Master is shown below. The symbol ‘SOP’ means ‘start of packet’ and ‘EOP’ means ‘end of packet’.

Start for Write	Slave Address	SOP	Red LED TCPWM Compare Value	Green LED TCPWM Compare Value	Blue LED TCPWM Compare Value	EOP	Stop
w	(0x08)	(0x01)	(0x00 to 0xFF)	(0x00 to 0xFF)	(0x00 to 0xFF)	(0x17)	p

Some of the example commands sent by the I²C Master are shown below for changing the color of the RGB LED. You can change the value between ‘0x00’ to ‘0xFF’ for red, green, and blue LED's TCPWM compare value.

- **‘w 08 01 FF 00 00 17 p’** changes color to Red.
- **‘w 08 01 00 FF 00 17 p’** changes color to Green.
- **‘w 08 01 00 00 FF 17 p’** changes color to Blue.
- **‘w 08 01 00 FF FF 17 p’** changes color to Cyan.
- **‘w 08 01 7F 00 7F 17 p’** changes color to Purple.
- **‘w 08 01 FF FF 00 17 p’** changes color to Yellow.
- **‘w 08 01 FF FF FF 17 p’** changes color to White.

The following is the command format to read the status form the Slave’s read buffer. The symbol ‘x’ denotes one byte to read from the Slave’s read buffer. In this example, three bytes are read from the Slave.

Start for Read	Slave Address	Read SOP (0x01)	Read Status (0x00 = Success) (0xFF =Fail)	Read EOP (0x17)	Stop
r	(0x08)	x	x	X	p

8. After each command is sent, the status packet must be read from the read buffer of the Slave by sending the **‘r 08 x x x p’** command. If the packet read is in the format **‘r 08 01 00 17 p’**, then the status is set as 'success'; if the packet read is **‘r 08 01 FF 17 p’**, the status is set as 'fail' for the command sent by the I²C Master.

Components

Table 2 lists the PSoC Creator Components used in both projects and the hardware resources used by each Component.

Table 2. PSoC Creator Components

Component	Instance Name	Hardware Resources
I2C (SCB_I2C_PDL)	sI2C	One SCB peripheral block
TCPWM(TCPWM_PWM_PDL)	RedPWM, GreenPwm, BluePWM	Three TCPWM peripheral blocks
Clock(SysClk_PDL)	RGB_PWMclk	One Clock peripheral
GPIO(GPIO_PDL)	RedLED, GreenLED, BlueLED	Three GPIO peripherals

Parameter Settings

Non-default settings for each Component are listed in Table 3.

Table 3. Parameter Settings

Component	Instance Name	Non-default Parameter Settings
I2C (SCB_I2C_PDL)	sI2C	Checked boxes: Use TX FIFO, Use RX FIFO
TCPWM(TCPWM_PWM_PDL)	RedPWM, GreenPWM, BluePWM	Period 0: 255u, Compare 0 : 0u
Clock(SysClk_PDL)	RGB_PWMclk	Frequency: 48 MHz

Design-Wide Resources

Make sure that V_{DDD} (**PSoC Creator** > **Design Wide Resources** tab > **System** tab) is set to 2.7 V or more to use GreenLED and BlueLED.

Table 4 shows the pin assignment for this code example.

Table 4. Pin Names and Location

Pin Name	Location
sI2C:scl	P6[0]
sI2C:sda	P6[1]
RedLED	P0[3]
GreenLED	P1[1]
BlueLED	P11[1]

Related Documents

Application Notes	
AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity	Describes PSoC 63 with Bluetooth Low Energy (BLE) Connectivity and how to build your first PSoC Creator project
PSoC Creator Component Datasheets	
I2C	Supports I²C communication
TCPWM	Supports PWM Signal Generation
Clock	Supports clock signal Generation
GPIO	Supports Input/Ouput Communication through pins
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet	PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual
Development Kit (DVK) Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	

Document History

Document Title: CE221119 – PSoC 6 MCU I^2C Slave

Document Number: 002-21119

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5894788	VJYA	11/10/2017	New Code Example

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709