Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

CE219431 - PSoC 6 MCU PDM-to-PCM Example

Objective

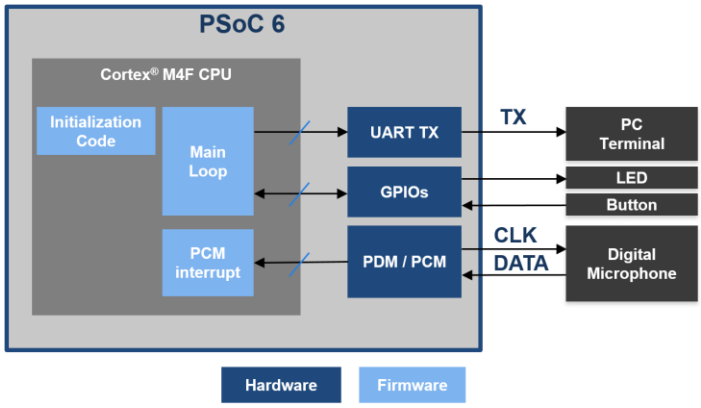
This example demonstrates how to use the pulse-density modulation/pulse-code modulation (PDM/PCM) hardware block in PSoC 6 with a digital microphone.

Overview

This code example shows how to use a digital microphone with the PDM/PCM block. It measures the sound intensity (volume), and turns ON an LED when the volume exceeds a threshold. You can reset the threshold by pressing a switch; you can use this to capture the environment noise and set a new threshold above the noise. A debug UART reports the current volume.

Figure 1 shows the high-level block diagram of the interface between PSoC 6 and the microphone and user interface elements.

Figure 1. Block Diagram



Requirements

Tool: PSoC® Creator™ 4.2

Programming Language: C (ARM® GCC 5.4-2016-q2-update)

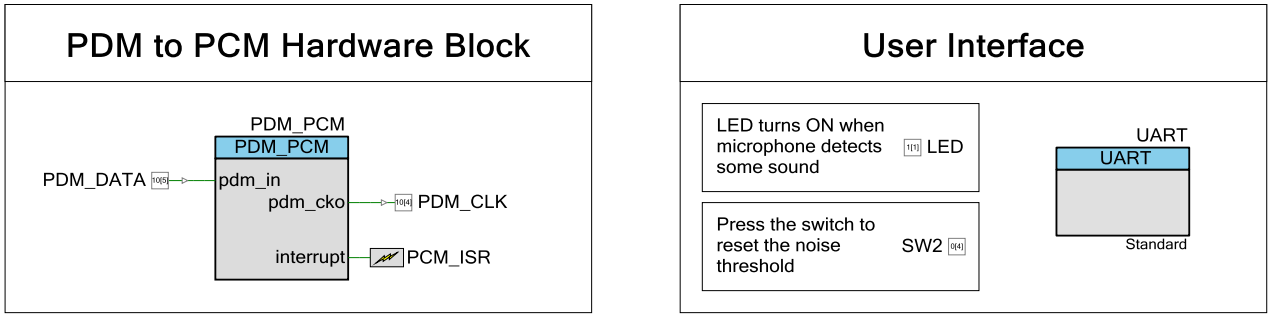
Associated Parts: All PSoC 6 parts with the PDM/PCM hardware block

Related Hardware: CY8CKIT-062 or CY8CKIT-062-BLE, CY8CKIT-028-EPD

Design

Figure 2 shows the PSoC Creator schematic for interfacing a digital microphone with the PDM_PCM Component.

Figure 2. Project Schematics



The CY8CKIT-028-EPD shield has a microphone with a single-bit PDM output. This allows you to convert any audio captured by the microphone into a digital signal (PDM). The PSoC 6 device in CY8CKIT-062 converts this digital signal to a quantized 16-bit value (PCM). An interrupt is triggered when there is enough data to be processed (at least 128 samples). The absolute values of the samples are summed and divided by the number of samples. This represents the overall loudness of the sound, which is proportional to the volume.

The PDM_CLK clock is configured for a sampling rate of 8 ksps. Because this application only calculates the volume, a higher sample rate for better audio quality is not required. Ideally, the audio wave captured should oscillate around zero. The PDM/PCM hardware block is also configured to enable a high-pass filter. This filter attenuates low frequencies that add offset to the conversion.

The calculated volume is sent over the UART. Use a terminal program on your PC to observe the data. The code example also turns ON a kit LED when the calculated volume is greater than a predefined threshold. The threshold can be used as a noise threshold so that this example can run in noisy environments. By pressing the SW2 button, you can set a new noise threshold based on the current volume. Note that you should remove any source of sound and avoid speaking close to the microphone while pressing the button.

Design Considerations

This code example runs on CY8CKIT-062-BLE or CY8CKIT-062 kits, which have a PSoC 6 device. To port the design to other PSoC 6 devices and kits, change the target device using PSoC Creator **Project** > **Device Selector**, and pin assignments in the Design Wide Resources window.

Hardware Setup

This example requires the CY8CKIT-028-EPD shield to be connected to CY8CKIT-062 (-BLE).

Operation

1. Connect the CY8CKIT-028-EPD shield to CY8CKIT-062.
2. Build the “CE218636_PDM_PCM” project and program CY8CKIT-062. For more information on building projects and device programming, see PSoC Creator Help.
3. Touch, speak, or play any sound over the microphone. Observe that the Green LED on CY8CKIT-062 turns ON.
4. If the LED is always ON or blinking, without playing any sound over the microphone, press the CY8CKIT-062 SW2 button to reset the noise threshold.
5. Open a serial terminal in the PC and connect to the COM port created by CY8CKIT-062. Configure the baud rate at 115200 bps. Observe the volume being reported in real time.

Components

Table 1 lists the PSoC Creator Components used in this example, as well as the hardware resources used by each.

Table 1. List of PSoC Creator Components

Component	Instance Name	Hardware Resources
PDM_PCM	PDM_PCM	PDM/PCM Block
UART	UART	SCB
Interrupt	PCM_ISR	Interrupt [CM4]
Digital Output Pin	PDM_CLK	GPIO
Digital Output Pin	LED	GPIO
Digital Input Pin	PDM_DATA	GPIO
Digital Input Pin	SW2	GPIO

Parameter Settings

This section shows the changed settings for various Components as well as the system clocks.

PDM_PCM

The PDM_PCM Component is configured for a sampling rate of 8 ksps. Configure the HFClk1 clock to a specific frequency (see below). The PDM_CLK frequency is calculated as:

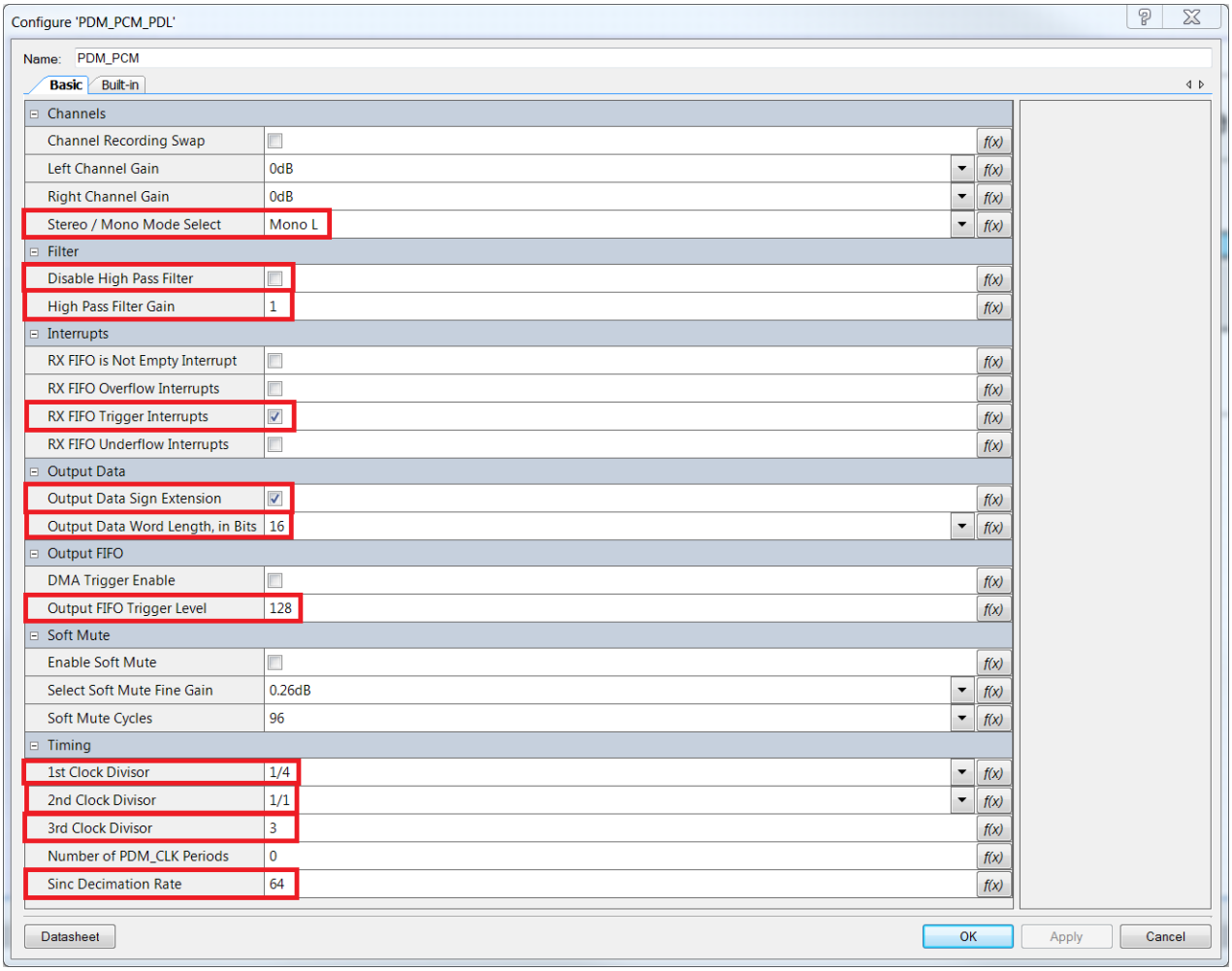
$$\text{PDM CLK (kHz)} = \frac{\text{HFClk1 (kHz)}}{\text{1st Clock Divisor} \times \text{2nd Clock Divisor} \times (\text{3rd Clock Divisor} + 1)}$$

The sampling rate is calculated as:

$$\text{Sampling Rate (ksps)} = \frac{\text{PDM CLK}}{2 \times \text{Sinc Decimation Rate}}$$

Figure 3 shows the PDM_PCM Component configuration window.

Figure 3. PDM_PCM Component Configuration Window



PCM_ISR

The PCM Interrupt Type is configured to be **Auto-Select Trigger**.

UART

The UART is configured to be **TX only** and baud rate (bps) equal to **115200**.

SW2

The pin drive mode is configured to be **Resistive Pull Up** and the Interrupt to be **Rising Edge**.

LED

The pin drive mode is configured to be **Strong Drive** and the Initial drive state to be **High (1)**.

Design-Wide Resources

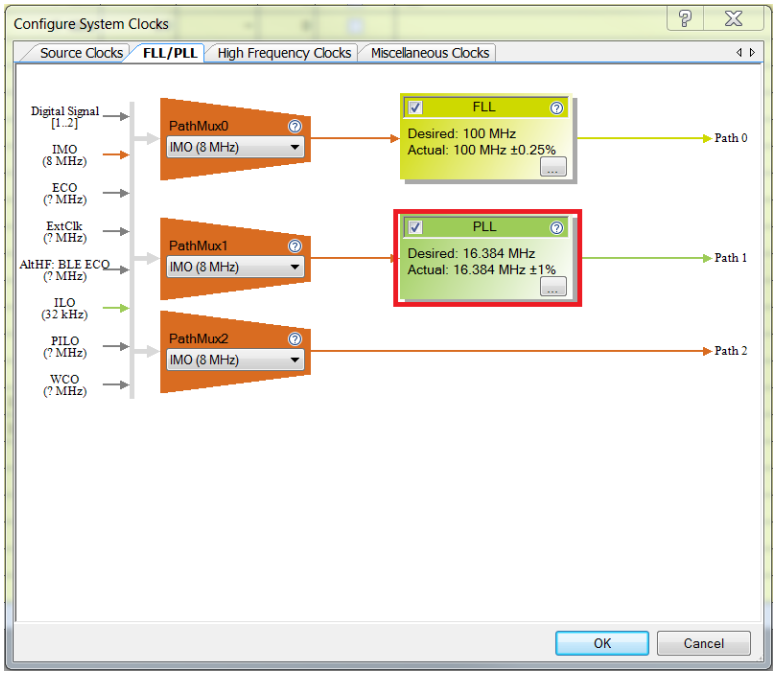
In the Configure System Clocks FLL/PLL tab, enable the PLL to be 16.384 MHz. This clock frequency comes from the following equation:

$$\text{HFClk1} = \text{Sampling Rate} \times \text{1st Clock Divisor} \times \text{2nd Clock Divisor} \times (\text{3rd Clock Divisor} + 1) \times 2 \times \text{Sinc Decimation Rate}$$

$$\text{HFClk1} = 8\text{k} \times 4 \times 1 \times (3 + 1) \times 2 \times 64 = 16.384\ \text{MHz}$$

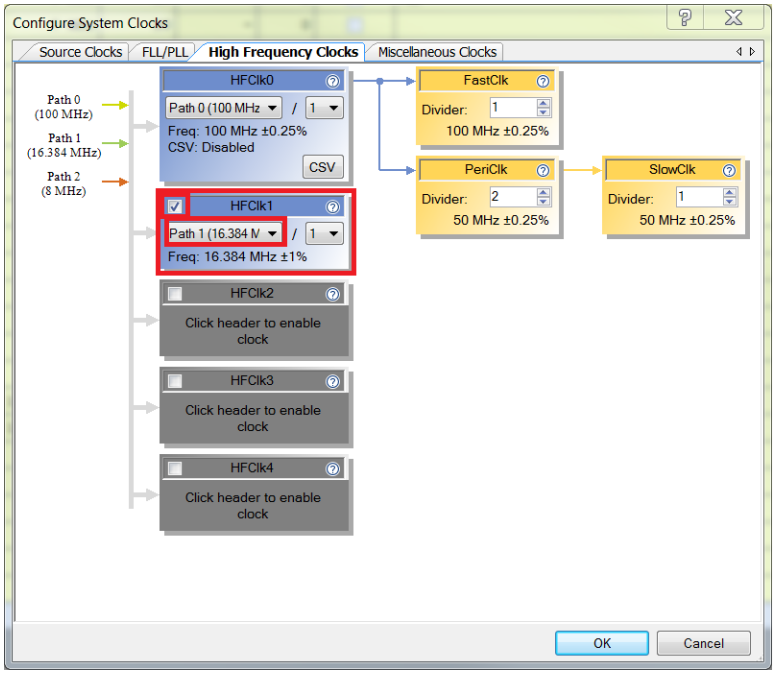
Figure 4 shows the configuration of the PLL.

Figure 4. Clock Configuration Window



Configure the High Frequency Clocks HFClk1 to be linked to Path 1. Figure 5 shows how the high-frequency clocks are configured.

Figure 5. High Frequency Clock Configuration



Related Documents

Table 2 lists all relevant application notes, code examples, knowledge base articles, device datasheets, and Component / user module datasheets.

Table 2. Related Documents

Application Notes	
AN210781 Getting Started with PSoC 6 MCU with BLE Connectivity	Describes the PSoC 6 BLE, and how to build this code example
AN217666 PSoC 6 MCU Interrupts	Describes how to use interrupts in PSoC 6
PSoC Creator Component Datasheets	
PSoC 6 Pulse-Density Modulation to Pulse-Code Modulation Decoder (PDM_PCM) Component	Converts a bit stream from a PDM source to PCM, which is similar to the output of an ADC
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet (PRELIMINARY)	
PSoC 6 MCU: PSoC 62 Datasheet	
PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual	
PSoC 6 MCU: PSoC 63 with BLE Register Technical Reference Manual	
PSoC 6 MCU Programming Specifications	
Development Kit (DVK) Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	

Document History

Document Title: CE219431 - PSoC 6 MCU PDM-to-PCM Example

Document Number: 002-19431

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5731230	RLOS	05/09/2017	New Code Example
*A	5842526	RLOS	08/02/2017	Updated Project to PSoC Creator 4.2

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

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