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.

CE218636 – PSoC 6 MCU Inter-IC Sound (I2S) Example

Objective

This example demonstrates how to use the I2S hardware block in PSoC® 6 MCU to interface with an audio codec.

Requirements

Tool: PSoC Creator™ 4.3; Peripheral Driver Library (PDL) 3.0.1 or higher

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

Associated Parts: All PSoC 6 MCU parts with I2S

Related Hardware: CY8CKIT-062-WiFi-BT or CY8CKIT-062-BLE, CY8CKIT-028-TFT

Overview

This code example shows how to play a short audio on a speaker or headphone. The application stores the audio data in the flash memory. This data is written to the output TX FIFO of the I2S hardware block, which interfaces with an audio codec chip.

The audio codec chip converts a digital audio stream into analog. The chip is configured by PSoC 6 MCU over an I^2C interface. The code example includes the files *codec.c/h*, which wrap the configuration of the audio codec.

Hardware Setup

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

Software Setup

None

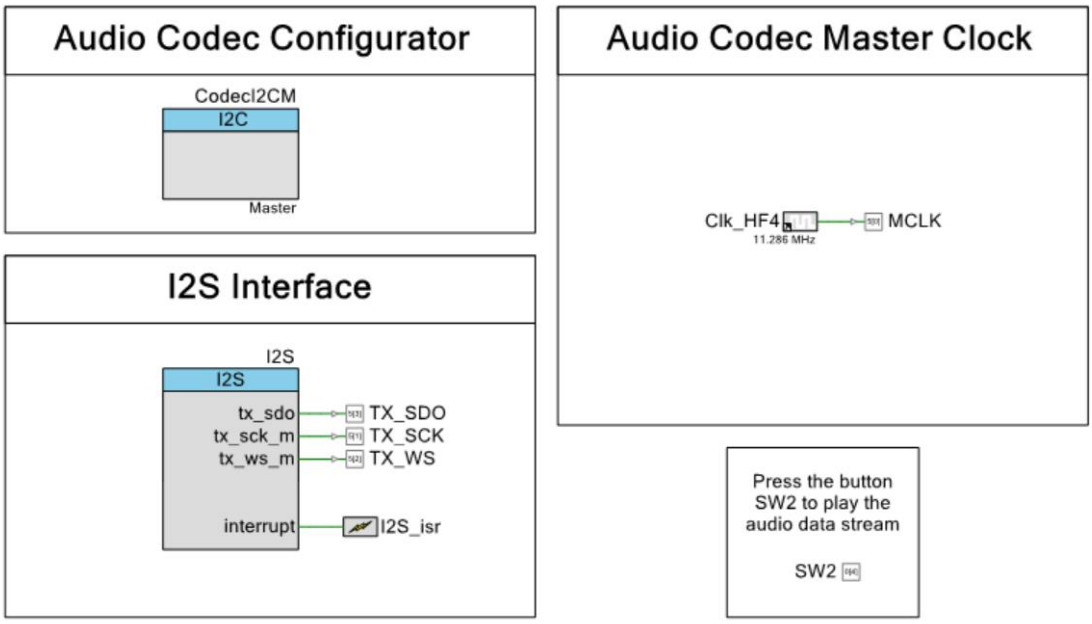
Operation

1. Connect the CY8CKIT-028-TFT shield to the CY8CKIT-062 kit.
2. Build the “CE218636_I2S” project and program the CY8CKIT-062 kit. For more information on building projects and device programming, see PSoC Creator Help.
3. Connect a headphone or speaker to the audio jack of the CY8CKIT-028-TFT shield.
4. Press the CY8CKIT-062 SW2 button to play the short audio “PSoC Rocks”.

Design and Implementation

Figure 1 shows the PSoC Creator schematic for interfacing an audio codec with the I²S Component.

Figure 1. Project Schematics

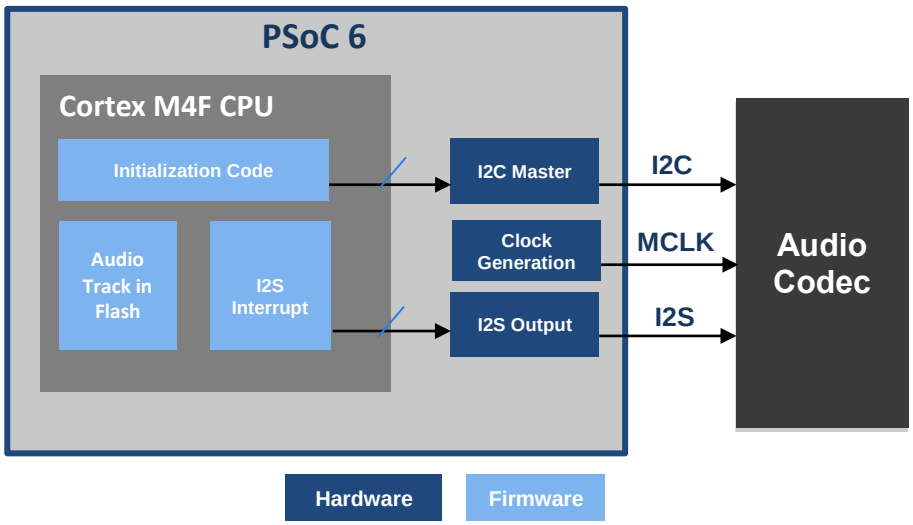


The CY8CKIT-028-TFT shield contains the audio codec AK4954A and an audio jack. This allows you to listen to any audio data stream transmitted to the audio codec over the I2S interface. You can connect a speaker or headphone to the audio jack. PSoC 6 MCU in the CY8CKIT-062 kit generates the data over the I2S. This is done by storing a short audio track in the flash memory, and writing it to the TX FIFO of the I2S hardware block. The I2S interface requires a continuous stream of data, which can be satisfied by feeding the TX FIFO with DMA transfers or interrupt service routine (ISR). In this example, an ISR is configured to be triggered when the TX FIFO is not full.

PSoC 6 MCU also generates the clock fed to the audio codec. Based on the AK4954A datasheet, this codec requires an 11.2896-MHz MCLK and a 2.822 MHz BCLK to sample at 44.1 kHz. The code example contains an I²C Master, through which PSoC 6 MCU configures the audio codec. The code example provides an API to easily configure the AK4954A.

Figure 2 shows the high-level block diagram of the interface between PSoC 6 MCU and the audio codec chip.

Figure 2. Block Diagram



Design Considerations

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

Components

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. List of PSoC Creator Components

Component	Instance Name	Hardware Resources
I^2S	I2S	I2S Block
I^2C (Master)	CodecI2CM	SCB
Clock	HFClk4	High Frequency Clock
Interrupt	I2S_isr	Interrupt [CM4]
Digital Output Pin	MCLK	GPIO
Digital Output Pin	TX_SDO	GPIO
Digital Output Pin	TX_SCK	GPIO
Digital Output Pin	TX_WS	GPIO
Digital Input Pin	SW2	GPIO

Parameter Settings

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

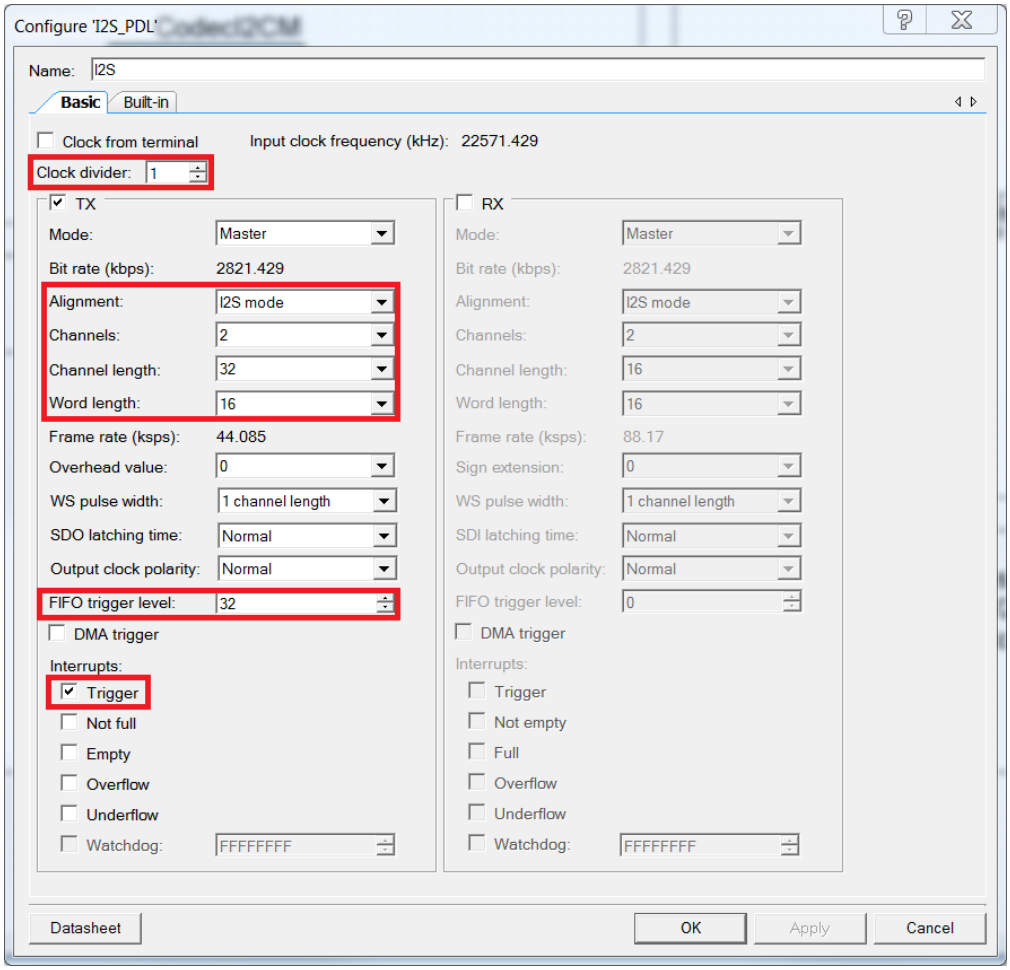
I2S

The I2S Component is configured for a frame rate of 44.1 ksps. Configure the HFClk1 clock to a specific frequency (see below). The Input clock frequency displayed in the Component is derived from HFClk1. The frame rate is calculated as:

$$Frame\ Rate\ (ksps) = \frac{HFClk1}{Clock\ divider \times 16 \times Channel\ Length}$$

Figure 3 shows the I2S configuration window. Note that the frame rate is not exactly 44.1 ksps due to the PLL output frequency cannot achieve the desired frequency of 22.5792 MHz.

Figure 3. I²S Component Configuration Window



I2S_isr

The I2S Interrupt Type is configured to be **Auto-Select Trigger**.

HFClk4

The Component clock is configured to be 11.286 MHz, which is sourced from the PLL output and divided by 2.

SW2

The pin drive mode is configured to be Resistive Pull Up.

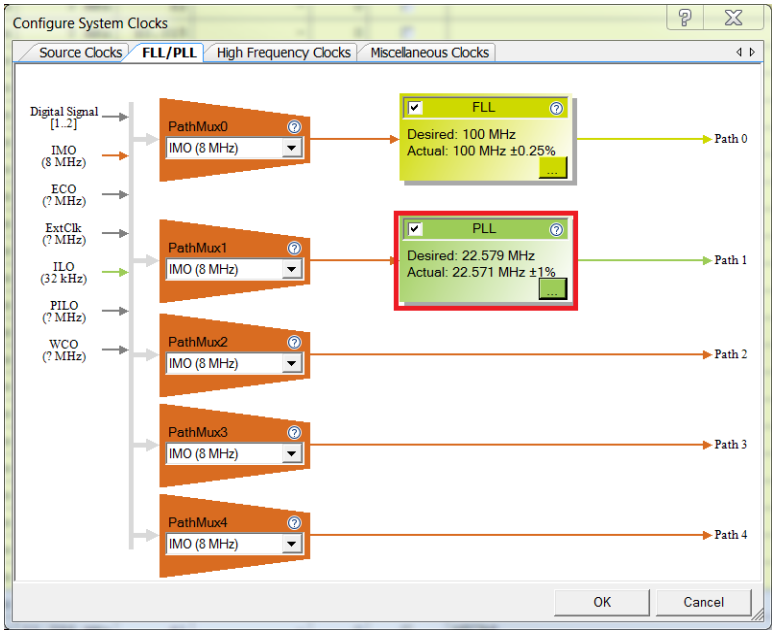
Design-Wide Resources

In the FLL/PLL tab, enable the PLL to be 22.5792 MHz. This clock frequency comes from the following equation:

$$HFClk1 = Frame\ Rate \times Clock\ divider \times 16 \times Channel\ Length = 44.1k \times 1 \times 16 \times 32 = 22579.2\ kHz$$

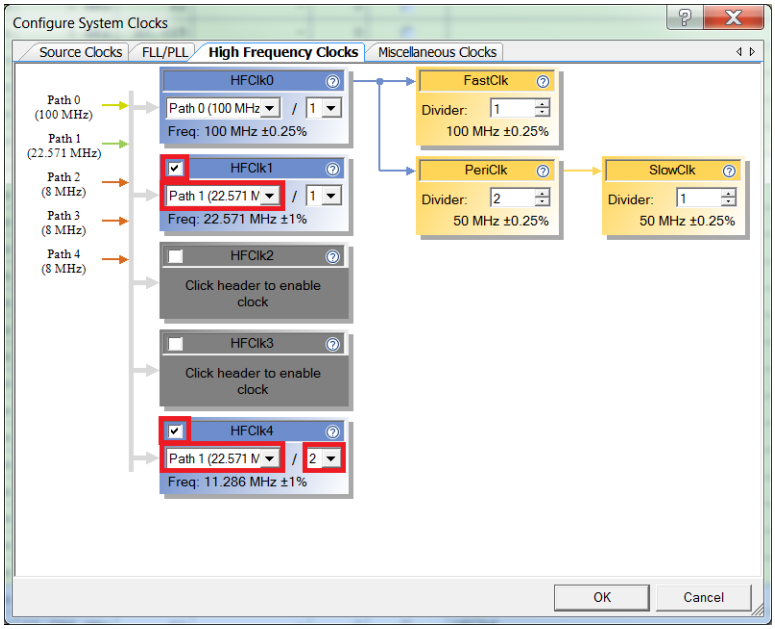
Figure 4 shows the configuration of the PLL. Note that the actual frequency is 22.571 MHz, instead of the desired frequency of 22.5792 MHz.

Figure 4. Clock Configuration Window



Configure the High Frequency Clocks HFClk1 and HFClk4 to be linked to the Path 1. Note that ideally both the clocks created for the I2S block (MCLK and BCLK) should come from the same source. Figure 4 shows how the high frequency clocks are configured.

Figure 5. High Frequency Clock Configuration



Related Documents

Application Notes	
AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (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	
Inter-IC Sound Bus (I2S) Component	Sends digital audio streaming data to external I2S devices
Intra-Integrated Circuit (I2C) Component	Supports I^2C slave, master, and master-slave operation configurations.
Device Documentation	
PSoC® 6 MCU: PSoC 63 with BLE Datasheet Programmable System-on-Chip (PSoC®)	
PSoC 6 MCU: CY8C62x6, CY8C62x7 Datasheet	
PSoC® 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual	
PSoC® 6 MCU: CY8C63x6, CY8C63x7 : CY8C63X6, CY8C63X7 REGISTERS TECHNICAL REFERENCE MANUAL (TRM)	
PSoC 6 Programming Specifications	
Development Kit (DVK) Documentation	
PSoC® 6 BLE Pioneer Kit (CY8CKIT-062-BLE)	

Document History

Document Title: CE218636 – PSoC 6 MCU Inter-IC Sound (I2S) Example

Document Number: 002-18636

Revision	ECN	Submission Date	Description of Change
**	5697907	06/21/2017	New Code Example
*A	5849304	08/10/2017	Updated Project to PSoC Creator 4.2 Updated Clock Frequencies to use slower values
*B	5891795	09/21/2017	Updated Audio Codec part Updated TopDesign screenshot
*C	6899239	06/17/2020	Updated Tool and Related Hardware under Requirements. Updated Figure 1. Updated template.

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

Community | Code Examples | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
An Infineon Technologies Company
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2017-2020. This document is the property of Cypress Semiconductor Corporation and its subsidiaries ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.
TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress shall have no liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. CYPRESS DOES NOT REPRESENT, WARRANT, OR GUARANTEE THAT CYPRESS PRODUCTS, OR SYSTEMS CREATED USING CYPRESS PRODUCTS, WILL BE FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (collectively, "Security Breach"). Cypress disclaims any liability relating to any Security Breach, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from any Security Breach. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. "High-Risk Device" means any device or system whose failure could cause personal injury, death, or property damage. Examples of High-Risk Devices are weapons, nuclear installations, surgical implants, and other medical devices. "Critical Component" means any component of a High-Risk Device whose failure to perform can be reasonably expected to cause, directly or indirectly, the failure of the High-Risk Device, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from any use of a Cypress product as a Critical Component in a High-Risk Device. You shall indemnify and hold Cypress, its directors, officers, employees, agents, affiliates, distributors, and assigns harmless from and against all claims, costs, damages, and expenses, arising out of any claim, including claims for product liability, personal injury or death, or property damage arising from any use of a Cypress product as a Critical Component in a High-Risk Device. Cypress products are not intended or authorized for use as a Critical Component in any High-Risk Device except to the limited extent that (i) Cypress's published data sheet for the product explicitly states Cypress has qualified the product for use in a specific High-Risk Device, or (ii) Cypress has given you advance written authorization to use the product as a Critical Component in the specific High-Risk Device and you have signed a separate indemnification agreement.
Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.