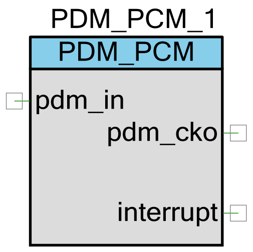
Pulse-Density Modulation to Pulse-Code Modulation Decoder (PDM_PCM_PDL)

1.0

Features



- Supports stereo/mono dual mode PDM to PCM conversion programmable word length
- 1-bit PDM input, 16/18/20/24-bit PCM data output
- Low digital microphone clock rates for low power applications
- PCM sampling rate: from 8 KHz to 48 KHz
- Volume control: programmable gain amplifier (PGA) from -12 dB to +10.5 dB in 1.5 dB steps
- Hybrid Component (Peripheral Driver Library (PDL) and Component Application Programming Interface (API))

General Description

The PDM_PCM_PDL Component converts a bit stream from a PDM source to PCM, which is similar to the output of an ADC. Various sample rates and data formatting options are supported.

The PDM_PCM_PDL Component is a graphical configuration entity built on top of the pdm_pcm driver available in the PDL. It allows schematic-based connections and hardware configuration as defined by the Component Configure dialog.

The generated configuration structures are to be passed into the pdm_pcm driver. The Component-specific API can also be used to access higher level functionalities built on top of the base driver.

When to Use a PDM_PCM_PDL Component

The PDM_PCM_PDL Component is typically part of a system to implement digital voice/audio recording and processing such as:

- Wearable Device
- IoT/IoE System
- Virtual Reality Gaming System

PRELIMINARY

- Tablet
- Smart Automobile
- Voice-in-Command Appliance
- Smart Home System
- PMP, MID
- Voice Recorder

Quick Start

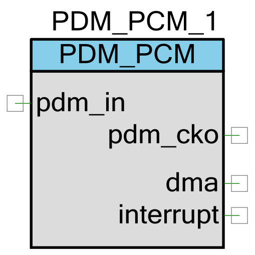
1. Drag a PDM_PCM [v1.0] Component from the Component Catalog Cypress/Communications/PDM_PCM folder onto your schematic (the placed instance takes the name PDM_PCM_1).
2. Double-click to open the Configure dialog.
3. Set up the desired PDM_PCM settings (clock dividers, audio channels, word lengths, interrupts, etc.).
4. Connect all the input/output digital pins (if using the base Component and not a macro).
5. Build the project in order to verify the correctness of your design. This will add the required PDL modules to the Workspace Explorer, and generate the configuration data for the PDM_PCM_1 instance.
6. In *main.c*, initialize the peripheral and start the application (example for transmission only):

```
(void)Cy_PDM_PCM_Init(PDM_PCM_1_HW, &PDM_PCM_1_config);
Cy_PDM_PCM_ClearFifo(PDM_PCM_1_HW);
Cy_PDM_PCM_EnableDataStream(PDM_PCM_1_HW);
```

7. Build and program the device.

Input/Output Connections

This section describes the various input and output connections for the PDM_PCM_PDL Component. An asterisk (*) in the following list indicates that it may not be shown on the Component symbol for the conditions listed in the description of that I/O.



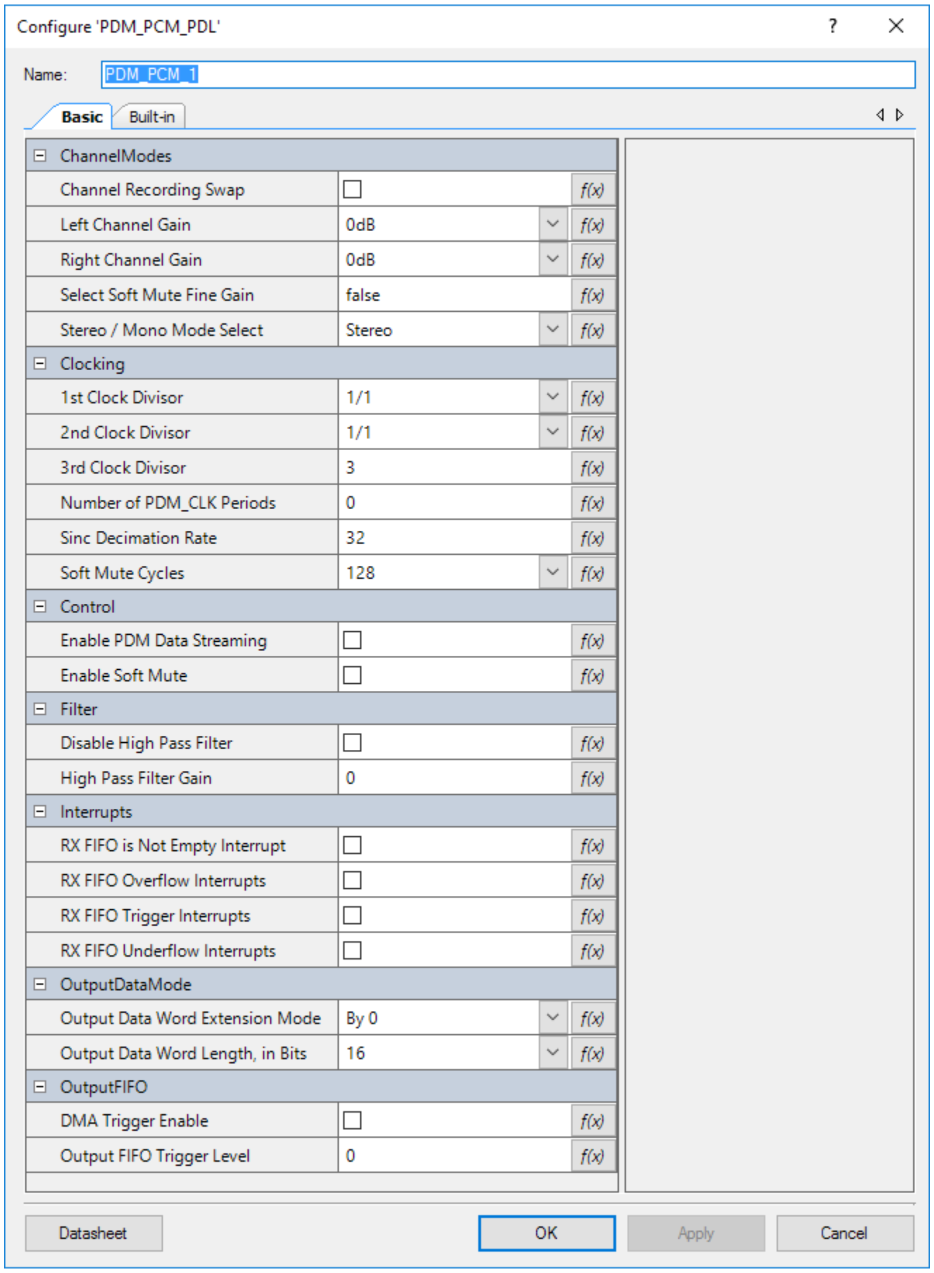
Terminal Name	I/O Type	Description
pdm_in	Digital Input	PDM input signal from PDM device for conversion. Can be connected to external input pin.
pdm_cko	Digital Output	Clock output signal for PDM sampling. Can be connected to external output pin.
interrupt	Digital Output	Interrupt signal output.
dma*	Digital Output	DMA transfer request signal. Displays when you select the **DMA Trigger Enable** option.

Component Parameters

The PDM_PCM_PDL 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 general peripheral initialization settings.



Name: PDM_PCM_1

ChannelModes	
Channel Recording Swap	☐
Left Channel Gain	0dB
Right Channel Gain	0dB
Select Soft Mute Fine Gain	false
Stereo / Mono Mode Select	Stereo
Clocking	
1st Clock Divisor	1/1
2nd Clock Divisor	1/1
3rd Clock Divisor	3
Number of PDM_CLK Periods	0
Sinc Decimation Rate	32
Soft Mute Cycles	128
Control	
Enable PDM Data Streaming	☐
Enable Soft Mute	☐
Filter	
Disable High Pass Filter	☐
High Pass Filter Gain	0
Interrupts	
RX FIFO is Not Empty Interrupt	☐
RX FIFO Overflow Interrupts	☐
RX FIFO Trigger Interrupts	☐
RX FIFO Underflow Interrupts	☐
OutputDataMode	
Output Data Word Extension Mode	By 0
Output Data Word Length, in Bits	16
OutputFIFO	
DMA Trigger Enable	☐
Output FIFO Trigger Level	0

Parameter Name	Description
Channels Modes	
Channel recording swap	Right-left channel recording swap.
Enable soft mute gain	Set fine gain step for smooth PGA or Soft-Mute attenuation transition.
Left channel gain	Left channel gain, in dB.
Right channel gain	Right channel gain, in dB.
Stereo / mono mode select	Stereo / mono mode select.
Clocking	
1-st clock divisor	This configures a frequency of PDM CLK.
2-st clock divisor	MCLKQ divider (2nd divider).
3-st clock divisor	PDM CKO clock divider (3rd divider). Range: 1 - 15.
Number of PDM_CLK periods	Phase difference from the rising edge of internal sampler clock (CLK_IS) to that of PDM_CKO clock. Range: 0 - 7.
Sinc decimation rate	SINC Decimation Rate. Range: 0 - 127.
Soft mute cycles	Extra PDM clock delay during gain-setting or soft mute operation.
Control	
Enable PDM data streaming	Enable PDM data streaming.
Enable soft mute	Enable Soft Mute.
Filter	
Disable high pass filter	Disables high pass filter.
High pass filter gain	High pass filter gain. Range: 0 - 15.
Interrupts	
RX FIFO is Not Empty Interrupt	RX FIFO is not empty.
RX FIFO Overflow Interrupts	Attempt to write to a full RX FIFO.
RX FIFO Trigger Interrupts	RX FIFO has more entries than the value specified by “Output FIFO Trigger Level”
RX FIFO Underflow Interrupts	Attempt to read from an empty RX FIFO.
Output Data Mode	
Output Data Word Extension Mode	When reception word length is shorter than the word length of RX_FIFO_RD, extension mode of upper bit should be set. '0': Extended by '0' '1': Extended by sign bit (if MSB word is "1", then it is extended by "1", if MSB is "0" then it is extended by "0")
Output Data Word Length, in Bits	PCM Word Length in number of bits.

Parameter Name	Description
Output FIFO	
DMA Trigger Enable	Enable Trigger event from output RX FIFO.
Output FIFO Trigger Level	Trigger level. When the RX FIFO has more entries than the number of this field, a receiver trigger event is generated. Range: 0 - 255.

Application Programming Interface

API routines allow you to configure the Component using software.

By default, PSoC Creator assigns the instance name PDM_PCM_1 to the first instance of a Component in a given design. You can rename it to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol.

This Component uses the pdm_pcm driver module from the PDL. The driver is copied into the “pdl\drivers\peripheral\pdm_pcm\” directory of the application project after a successful build.

Refer to the PDL documentation for a detailed description of the complete API. To access this document, right-click on the Component symbol on the schematic and choose the “**Open PDL Documentation…**” option in the drop-down menu.

The Component generates the configuration structures and base address described in the Global Variables and Preprocessor Macros sections. Pass the generated data structure and the base address to the associated pdm_pcm driver function in the application initialization code to configure the peripheral. Once the peripheral is initialized, the application code can perform run-time changes by referencing the provided base address in the driver API functions.

In addition to the PDL API, the PDM_PCM component provides an instance-based component API that provide additional functionality available through PSoC Creator.

Global Variables

The PDM_PCM_PDL 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. PDM_PCM_1.c). Each variable is also prefixed with the instance name of the Component.

uint8 PDM_PCM_initVar

Indicates whether or not the PDM_PCM Component has been initialized. The variable is initialized to 0 and set to 1 the first time PDM_PCM_1_Start() is called. This allows the Component to restart without reinitialization after the first call to the PDM_PCM_1_Start() routine.

const cy_stc_pdm_pcm_config_t PDM_PCM_config

The Component generates an instance-specific configuration structure called PDM_PCM_1_config. This should be passed to the PDM_PCM_Init function to initialize the Component settings selected in the GUI.

Preprocessor Macros

The PDM_PCM_PDL Component generates the following preprocessor macro(s). Note that each macro is prefixed with the instance name of the Component (e.g. "PDM_PCM_1").

#define PDM_PCM_1_HW   PDM_PCM_1_cy_mxs40_pdm__HW

The pointer to the base address of the HW PDM_PCM instance.

#define PDM_PCM_1_RX_FIFO_DMA_PTR   (void *)&(PDM_PCM_1_HW->RX_FIFO_RD)

Rx FIFO Read register pointer for DMA initialization.

Data in RAM

The generated configuration 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 CY_CONST_CONFIG to make your selection. The default option is to place the data in flash.

Code Examples and Application Notes

This section lists the projects that demonstrate the use of the Component.

Code Examples

PSoC Creator provides access to code examples in the Code Example dialog. For Component-specific examples, open the dialog from the Component Catalog or an instance of the Component in a schematic. For general examples, open the dialog from the Start Page or **File** menu. As needed, use the **Filter Options** in the dialog to narrow the list of projects available to select.

Refer to the "Code Example" topic in the PSoC Creator Help for more information.

Application Notes

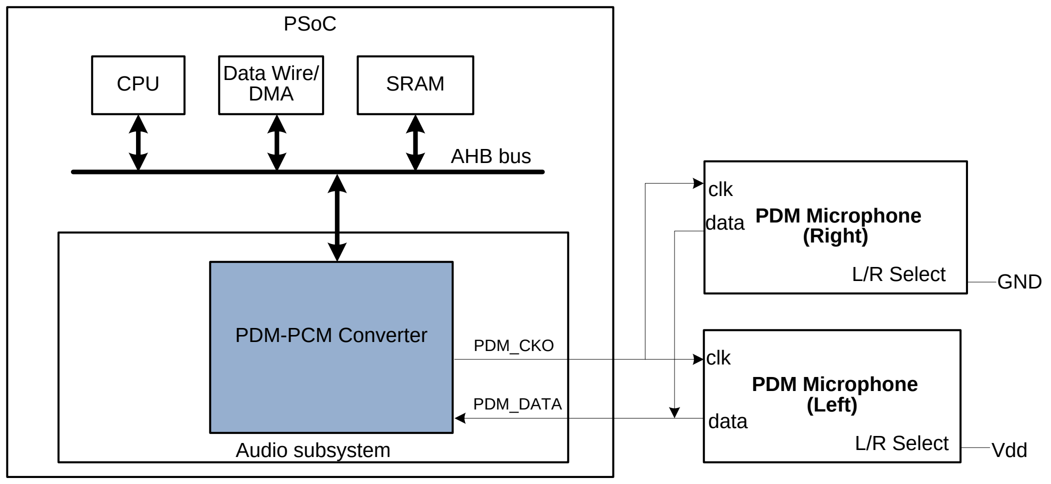
TBD

Functional Description

The PDM-PCM unit takes in a stereo or mono serial stream coming from e.g. an external digital PDM microphone.

Block Diagram and Configuration

The following is a simplified diagram of the PDM_PCM hardware:



Clock Selection

The PDM interface is a de-facto industry standard used widely for digital microphones. In the PDM interface the left/right channels are sampled at the rising/falling edges of the PDM clock (PDM_CKO).

The internal clock source for PDM_PCM is the HFCLK1.

The main clock input is divided by the **1st Clock divider** (PDM_PCM_1_CLK_DIV) and **2nd Clock divider** (PDM_PCM_1_MCLK_DIV) settings. This is again divided by the **3rd Clock divider** (PDM_PCM_1_CKO_DIV) setting to generate the PDM_CKO output.

In order to compensate for PDM_IN delay, the PDM_PCM_1_CKO_DELAY field is used to delay the internal PDM_IN sampling clock.

DMA Support

The PDM_PCM_PDL Component supports Direct Memory Access (DMA) transfers. The Component may transfer from the following source.

Name of DMA Source / Destination	Length	Direction	DMA Req Signal	DMA Req Type	Description
PDM_PCM_1_RX_FIFO_DMA_PTR	32	Source	dma	N/A	RX Fifo data register

PRELIMINARY

API Memory Usage

The Component is designed to use API from the pdm_pcm PDL module. That is why the Component itself only consumes resources necessary to allocate structures for driver operation and start the Component.

Industry Standards

MISRA Compliance

This section describes the MISRA-C:2004 compliance and deviations for the Component. There are two types of deviations defined:

- project deviations – deviations that are applicable for all PSoC Creator Components
- specific deviations – deviations that are applicable only for this Component

This section provides information on Component-specific deviations. Project deviations are described in the MISRA Compliance section of the *System Reference Guide* along with information on the MISRA compliance verification environment.

The PDM_PCM_PDL Component does not have any specific deviations.

This Component uses firmware drivers from the pdm_pcm PDL module. Refer to the PDL documentation for information on their MISRA compliance and specific deviations.

Registers

See the PDM to PCM decoder (PDM_PCM) Registers section in the chip *Technical Reference Manual (TRM)* for more information about the registers.

Resources

The PDM_PCM_PDL Component uses the PDM_PCM part of the AUDIOSS peripheral block.

DC and AC Electrical Characteristics

Characterization data of the PDM_PCM_PDL Component.

DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
TBD after characterization						

PRELIMINARY

AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
TBD after characterization						

References

List of references related to PDM_PCM_PDL Component:

- PDM – “pulse density modulation” data format
- PCM – “pulse code modulation” data format

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.b	Additional datasheet edits.	
1.0.a	Updated the datasheet.	
1.0	Initial Version	