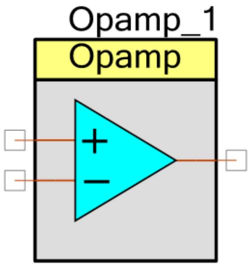
PSoC® Creator™ Component Datasheet

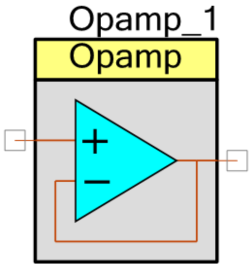
# Operational Amplifier (OpAmp_PDL)

1.0

## Features



- Follower or Opamp configuration
- Rail-to-rail inputs
- Direct output connection to pin (low resistance)
- 1 mA or 10 mA output current drive
- Internal connection for Follower configuration
- Multiple power levels
- Operates in Deep Sleep mode



## General Description

The Opamp operates as an off-the-shelf operational amplifier. A direct connection is made between the Opamp output to a GPIO pin for low output resistance. Two output current levels (1 mA and 10 mA) are provided to drive internal or external signals respectively. The 10 mA may drive both internal (SAR component) and external signals. The user also has control of different power levels that provide a trade-off between power and bandwidth.

**Note** External resistors are required to perform amplification.

### When to Use the Opamp

The following is a list of common use cases for the OpAmp_PDL component:

- Gain for Scan_ADC
- High impedance buffer for Scan_ADC
- General purpose signal amplifier
- Active filter

## Quick Start

1. Drag an OpAmp_PDL Component from the Component Catalog Cypress/Analog/Amplifiers folder onto your schematic (placed instance takes the name **Opamp_1**).

2. Double-click to open the Configure dialog.

3. Set up the desired settings (Mode, Output Drive, Power, Enable Deep Sleep Operation). Rename the instance name to **Opamp** for readability.

4. Open the Design-Wide Resources Pin Editor, and assign the input and output pins for your design. Note that the choice of pins that can be used for the **Opamp** inputs and output is limited.

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 configuration data for the **Opamp** instance.

6. In the main.c file, initialize the peripheral and start the application:

```
/* Initialize and enable the opamp. */
(void) Cy_CTB_OpampInit(Opamp_CTB_HW, Opamp_OPAMP_NUM, &
  Opamp_opampConfig);
Cy_CTB_Enable(Opamp_CTB_HW);
```

7. Build the project and program the device.

# Input/Output Connections

This section describes the various input and output connections for the OpAmp_PDL Component. An asterisk (∗) after the terminal name indicates that the terminal may not be shown on the Component symbol for the conditions listed in the description of that I/O terminal.

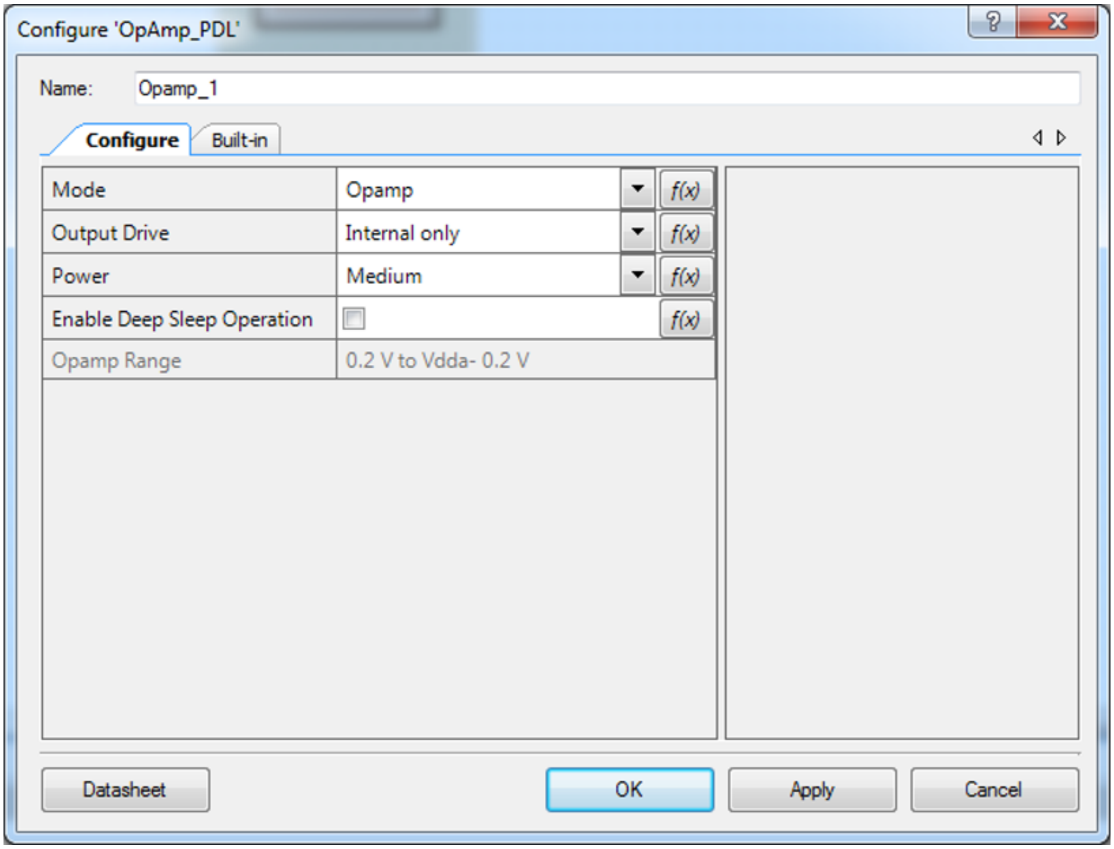
| Terminal | I/O Type | Description |
|---|---|---|
| Positive Input | Analog Input | When the Opamp is configured in Opamp Mode, this I/O is the standard Opamp non-inverting input. When the Opamp is configured in Follower Mode, this I/O is the voltage input. |
| Negative Input∗ | Analog Input | When the Opamp is configured in Opamp Mode, this I/O is the standard Opamp inverting input. When the Opamp is configured in Follower Mode, this I/O is connected internally to the output and the I/O is not visible. |
| Vout | Analog Output | The output can be directly connected to a pin and/or routed to an internal load using the analog routing fabric. The drive strength is selectable as either 10x or 1x. Connections to pins require the 10x setting. Internal connections can operate with either the 1x or 10x setting, but should normally be configured for 1x. |

# Component Parameters

This section covers the various parameters available from the Component's Configure dialog. Drag an Opamp_PDL Component from the Component Catalog onto your schematic, and double-click it to open the dialog.

For any selectable parameter, the option shown here in **bold** is the default.

## Basic Tab



Name: Opamp_1

| | |
|---|---|
| Mode | Opamp |
| Output Drive | Internal only |
| Power | Medium |
| Enable Deep Sleep Operation | □ |
| Opamp Range | 0.2 V to Vdda- 0.2 V |

### Mode

This parameter allows you to select between two configurations:

- **Opamp**
- Follower

In Opamp mode, both the inverting and non-inverting input terminals are visible. In the Follower mode, the inverting input is connected internally to the output.

### Output Drive

This parameter selects the output drive mode, either to an internal connection or to a device pin.

- Internal only
- **Output to pin**

### Power

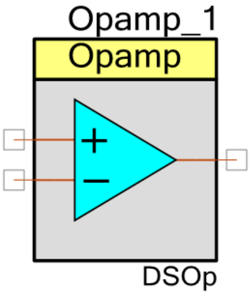
The opamp works over a wide range of operating currents. Higher operating currents increases the opamp bandwidth. Refer to the device datasheet for the current consumption and bandwidth specifications.

- Low
- **Medium**
- High

### Enable Deep Sleep Operation

This check box enables the opamp hardware to operate while in Deep Sleep mode. It also ensures that only Deep Sleep capable switches are used for analog routing during build time and that the Analog Reference (AREF) block that supplies the reference currents for the opamp is also enabled in Deep Sleep mode.

If this option is enabled, a "DSOp" label will be displayed under the symbol. If two opamps of the same CTB block are used, both must have the same Deep Sleep settings or the Creator project will not build.



The Cy_CTB_SetDeepSleepMode() function can be used during runtime to disable operation in Deep Sleep mode. However, this function should only be used if this check box is checked.

**Note** For correct operation in Deep Sleep mode, Vdda should be at least 2.7 V.

### Opamp Range (display only)

The opamp output range is 0.2 V to Vdda - 0.2 V for a typical load. The output range will reduce for higher loads (see the device datasheet for specifications). The opamp input range is rail-to-rail if Deep Sleep operation is disabled. If Deep Sleep operation is enabled, the pump is disabled which reduces the input range to 0 V to Vdda - 1.5 V. The overall opamp range is therefore

- **0.2 V to Vdda - 0.2 V** when Deep Sleep operation disabled
- 0.2 V to Vdda - 1.5 V when Deep Sleep operation enabled

# Application Programming Interface

The Application Programming Interface (API) is provided by the cy_ctb driver module from the PDL. The driver is copied into the “pdl\” 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 cy_ctb 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.

By default, PSoC Creator assigns the instance name **Opamp_1** to the first instance of the OpAmp_PDL 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. For readability, the instance name used in the following section is **Opamp**.

## Global Variables

The OpAmp_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., Opamp.c). Each variable is also prefixed with the instance name of the Component.

### Variables

- const cy_stc_ctb_opamp_config_t Opamp_opampConfig

### Variable Documentation

*const cy_stc_ctb_opamp_config_t Opamp_opampConfig*

Configuration structure for initializing one opamp using the CTB PDL.

## Preprocessor Macros

Preprocessor macros used in the OpAmp_PDL.

### Macros

- #define Opamp_CTB_HW
- #define Opamp_OPAMP_NUM

### Macro Definition Documentation

*#define Opamp_CTB_HW*

The pointer to the base address of the CTB instance

*#define Opamp_OPAMP_NUM*

The specific opamp of the CTB instance

## Data in Ram

The generated 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, enable or disable the "Config Data in Flash" checkbox to make your selection. The default option is to place the data in flash.

## Interrupt Service Routine

This component does not support any interrupts.

## Code Examples and Applications

### 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.

There are also numerous code examples that include schematics and example code available online at the Cypress Code Examples web page.

### Application Notes

Cypress provides a number of application notes describing how PSoC can be integrated into your design. You can access the Cypress Application Notes search web page.

# Functional Description

The component is a basic operational amplifier.

The opamps can operate in an ultra low power mode by setting the Opamp Reference Current in the Design-Wide Resources System Editor to 100 nA with a gain bandwidth trade-off. Refer to the device datasheet for the opamp specifications. This selection in the Design-Wide Resources will affect all opamps on the device. In ultra low power mode, the opamp input range is reduced to 0 V to Vdda - 1.5 V.

# MISRA Compliance

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

| Deviation Type | Description |
|---|---|
| Project deviations | Deviations that are applicable for all PSoC Creator Components |
| Specific deviations | Deviations that are applicable only for this Component |

Refer to **PSoC Creator Help** > **Building a PSoC Creator Project** > **Generated Files (PSoC 6)** for information on MISRA compliance and deviations of the files generated by PSoC Creator.

The OpAmp_PDL Component has no specific deviations.

# Resources

The OpAmp_PDL Component uses a single opamp from the CTB block.

# DC and AC Electrical Characteristics

**Note** Final characterization data for PSoC 6 devices is not available at this time. Once the data is available, the Component datasheet will be updated on the Cypress web site.

# Component Changes

This section lists the changes in the OpAmp_PDL Component from the previous versions.

| Version | Description of Changes | Reason for Changes / Impact |
|---|---|---|
| 1.0.b | Minor datasheet edits | |
| 1.0.a | Production qualified, remove "Prototype" designation | |
| 1.0 | Initial version | |

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. 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. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, 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 or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.