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.

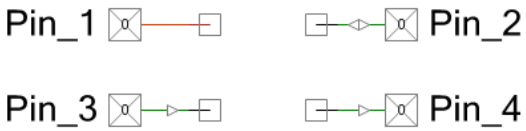
Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

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.

General Purpose Input / Output (GPIO)

1.0

Features



- Rapid setup of GPIO parameters
- Automatic place and route of signals to and from GPIOs
- Supports Software and Hardware connections
- Supports Analog, Digital I/O and Bidirectional signal types

General Description

The GPIO Component is a graphical configuration entity built on top of the cy_gpio driver available in the Peripheral Driver Library (PDL). It allows schematic-based connections and hardware configuration as defined by the Component Configure dialog.

The GPIO Component allows hardware resources to connect to a physical port-pin. It provides access to external signals through an appropriately configured physical IO pin. It allows electrical characteristics (e.g., Drive Mode) to be chosen for one or more pins; these characteristics are then used by PSoC Creator to automatically place and route the signals within the Component.

GPIO can be used with schematic wire connections, software, or both. It can be configured into many combinations of types. For convenience, the Component Catalog provides four preconfigured GPIO Components: Analog, Digital Bidirectional, Digital Input, and Digital Output.

When to Use a GPIO Component

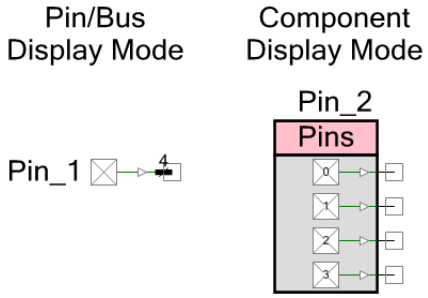
Use the GPIO Component when a design must generate or access an off-device signal through a physical IO pin. It is a convenient way to configure the pins in a graphical manner and allows the PSoC Creator IDE to perform the pin routing and multiplexing.

Input/Output Connections

This section describes the various input and output connections for the GPIO Component.

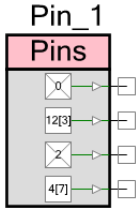
Display of GPIOs

GPIO can be configured into complex combinations of digital input, digital output, digital bidirectional, and analog. Simple configurations are generally shown as single pins. More complex types of pins are shown as standard Components with a bounding box.



Display of Locked Pins

When a GPIO Component is assigned to a physical pin using the PSoC Creator Design-Wide Resources Pin Editor, the tooltip for the GPIO Component shows the specific pin assignments. If pin assignment is locked, the display of the Component indicates the assignment, as shown in the following example:



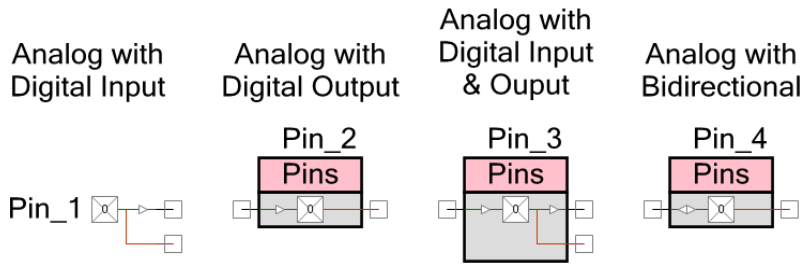
Note If the GPIO Component is set to **Display as Bus**, the display of the Component does not display any locked pin assignments; however, the tooltip still displays this information.

Analog

Configure the GPIO Component as Analog any time the design requires a connection between a device pin and an internal analog terminal connected with an analog wire. When configured as analog, the terminal is shown on the right side of the symbol with the connection drawn in the color of an analog wire.



An analog GPIO Component may also support digital input or output connections, or both, as well as bidirectional connections. It is possible to short together digital output and analog signals on the same pin. This can be useful in some applications, but is not a general use case, and should be used with care.



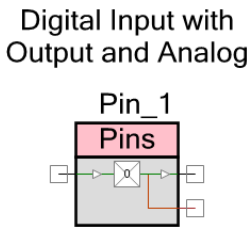
Digital Input

Configure a GPIO Component as digital input any time your design requires a connection between a device pin and an internal digital input terminal, or if the pin's state is read by the CPU/DMA. In all cases when using digital-input pins, the pin state is readable by the CPU/DMA. Additionally, if the schematic terminal (HW Connection) is displayed it can be routed to other Components on the schematic.

When visible, the terminal is shown on the right side of the symbol. The connection is drawn as a digital wire with a small input buffer to show signal direction.



A digital-input GPIO Component may also support digital output and analog connections.

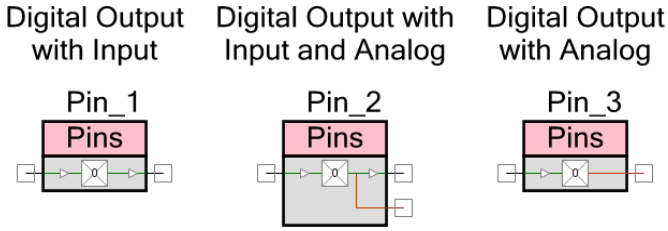


Digital Output

Configure a GPIO Component as digital output any time a device pin is to be driven to a logic high or low. For software pins, the pin output driver is writeable by the CPU/DMA. If the terminal is displayed, it can be routed from other Components on the schematic. When visible, the terminal is shown on the left side of the symbol. The connection is drawn as a digital wire with a small output buffer to show signal direction.



A digital-output GPIO Component may also support digital input and analog connections.



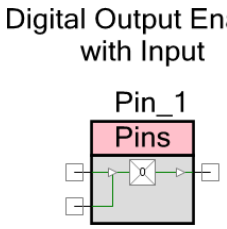
Digital Output Enable

Select digital output enable when digital logic is to be used to quickly control the pin output driver without CPU intervention. A high logic level on this terminal enables the pin output driver as configured by the **Drive Mode** parameter on the **General** subtab. A logic low level on this terminal disables the pin output driver and makes the pin assume the HI-Z drive mode. This terminal is shown when a Component is configured with digital output using a schematic connection, and when the digital output enable has been selected. The digital output enable appears on the left side of the symbol and connects to the digital output buffer. It is drawn as a digital wire.

When the pin is set to **Display as Bus**, only one output enable is provided regardless of the GPIO Component width because all of the pins share the same output enable. When not displayed as a bus, individual output enables are provided per pin.



A digital output enable GPIO Component may also support input and analog connections.



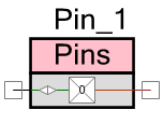
Digital Bidirectional

Configure a GPIO Component as digital bidirectional any time a connection between a device pin and an internal digital bidirectional terminal is required. Digital bidirectional mode is most often used with a communication Component like I^2C. When configured as digital bidirectional, the terminal is shown on the left side of the symbol with the connection drawn as a digital wire with input and output buffers showing that the signal is bidirectional.



A bidirectional GPIO Component may also support analog connections.

Digital Bidirectional
with Analog



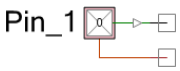
Vref

To configure a GPIO Component to use a Vref signal:

- Use a digital input or bidirectional terminal and set the **Threshold** parameter to **Vref** on the **Input** subtab, or
- Use a digital output or bidirectional terminal and configure the **Drive Level** to **Vref** on the **Output** subtab

Using a Vref requires an SIO pin, indicated with a pink outline. All pins can supply their respective V_{DDIO} supply voltages. SIO pins can also supply a programmable or analog-routed voltage for interfacing with devices at a different potential than the pin's Vddio voltage. The Vref terminal provides the analog routed voltage supplied to the SIO pin. SIO pins can also use the Vref input as the input threshold.

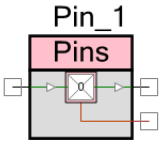
The Vref signal displays on the right side of the Component, extending from the bottom of the SIO single pin or the SIO pin pair, depending on how it is configured. Each SIO pin pair shares a single Vref input.



Vref can only be used in conjunction with another digital input or output connection.

Note When using Vref, **Analog** pin type cannot be selected.

Vref with
Digital Input & Output

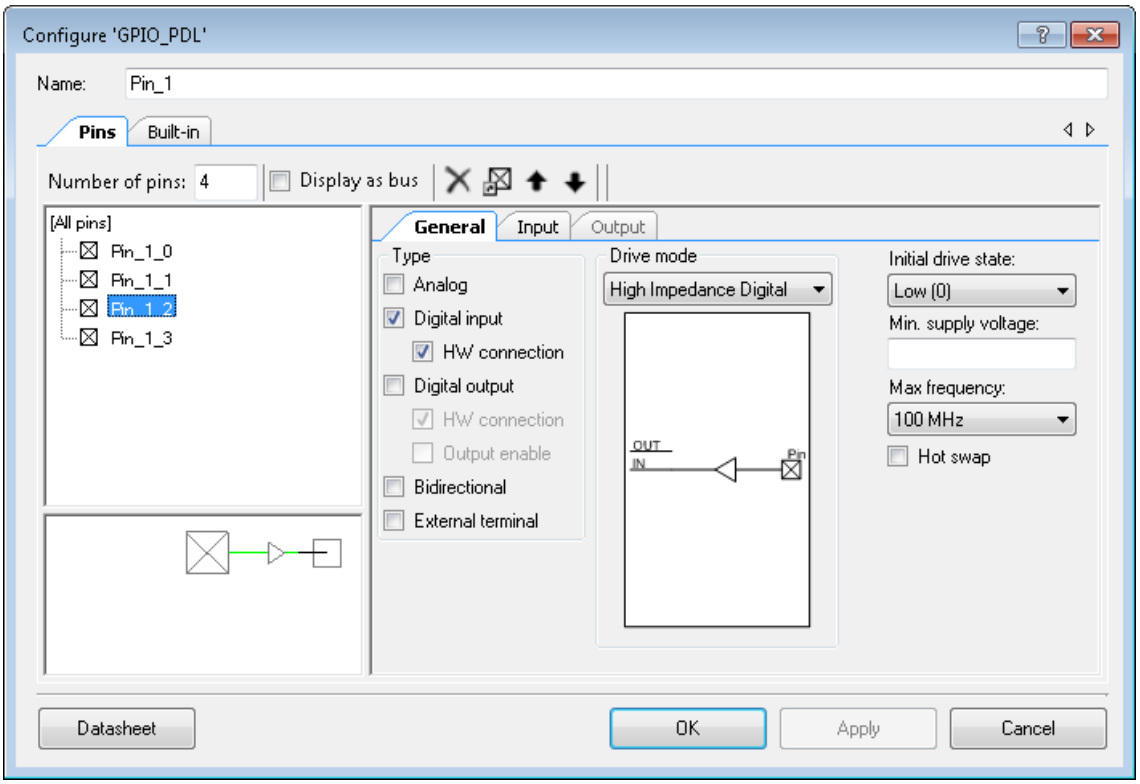


Component Parameters

Drag a GPIO Component onto the design schematic and double click it to open the **Configure** dialog. This dialog is used to set Component-wide parameters, which are organized into separate tabs.

Pins Tab

The **Pins** tab has three areas: a toolbar, pin tree, and a set of subtabs. The toolbar is used to determine how many physical pins are managed by the Component and determine their order. The subtabs are used to set the pin-specific attributes, such as pin type, drive mode, and initial drive state. The pin tree works with the subtabs to allow you to choose the specific pins to which these attributes are applied.



Toolbar

The toolbar contains these commands:

- **Number of pins** – The number of device pins controlled by the Component. The default value is **1**.
- **Display as Bus**

  This parameter selects whether to display individual terminals for each pin or a single wide terminal (bus). The bus option is only valid when pins are homogeneous. That

means all pins in the Component have the same pin type, output/input HW connections, and SIO grouping. They also must all either use or not use the SIO Vref.

-  **Delete Pin –** Deletes selected pins from the tree.
-  **Add/Change Alias –** Opens a dialog to add or change the alias name for a selected pin in the tree. You can also double-click a pin or press [**F2**] to open the dialog.
-  **Move Up/Down –** Moves the selected pins up or down in the tree.
-  **Pair/Unpair SIOs –** Pairs or unpairs selected SIO pins (identified by a pink outline) in the tree.

  This control specifies whether pins that require SIO should be placed in the same SIO pair on the device. Pairing pins results in fewer physical SIO pins being "wasted." This is because an unpaired pin that requires SIO cannot share its SIO pair on the device with another pin that requires SIO. For pins to share an SIO pair on the device, they must have their per-pair settings configured the same way and be adjacent.

  A pin requires SIO if **Hot Swap** is selected, **Threshold** is set to anything but "LVTTL" or "CMOS," **Drive Level** set to "Vref," and/or **Current** is set to "25mA sink."

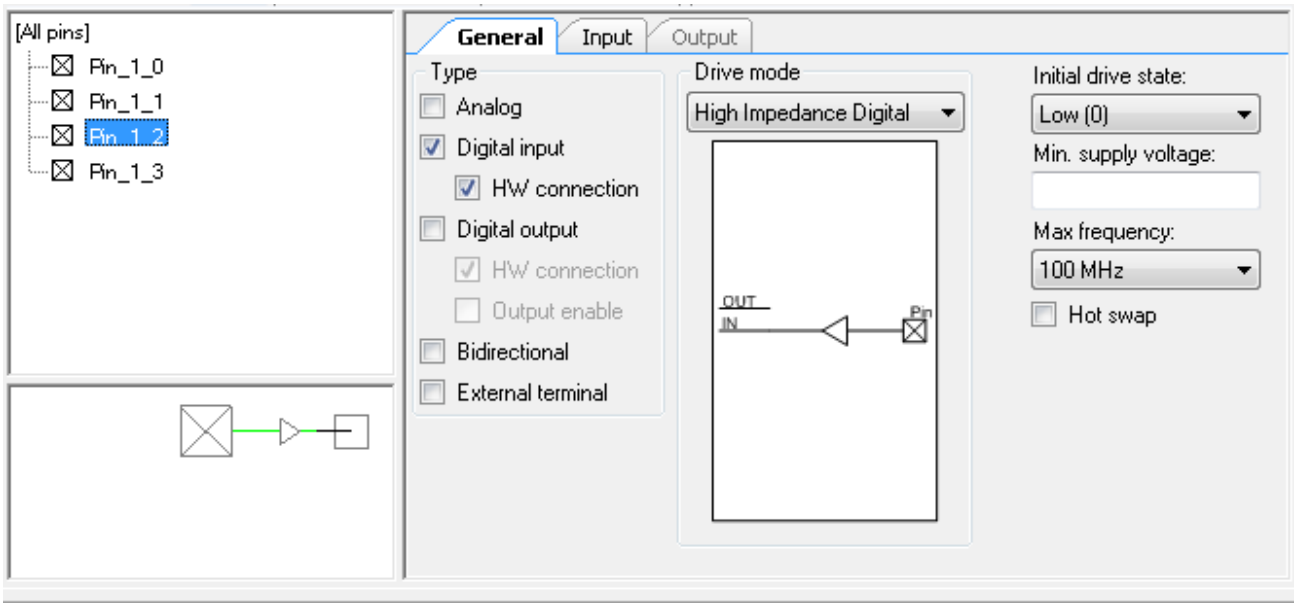
Pin Tree

This area displays all of the pins for the Component. You can individually select one or more pins to use with the toolbar commands and subtabs. Each pin displays its name, which consists of the <GPIO instance name>_<individual pin alias>.

Below the tree is a preview area that shows what the selected GPIO Component symbol will look like with various options selected for that specific pin.

General Subtab

This is the default subtab displayed for the **Pins** tab.



It contains the following parameters:

Type

This is where you choose the type of pins for your Component using the check boxes.

- **Analog** – Select **Analog** to enable the analog pin terminal to allow analog signal routing to other Components. Selecting analog forces the pin to be physically placed on a GPIO pin but not an SIO pin.
- **Digital Input** – Select **Digital Input** to enable the digital input pin terminal (optional) and enable the **Input** subtab for additional configuration options related to inputs.
  - **HW Connection** – This parameter determines whether the digital input terminal for an input pin is displayed in the schematic. If displayed, the pin provides a digital signal that can connect to other digital sinks such as Component terminals. If this option is not selected, the terminal is not displayed and it can only be read by the CPU/DMA.
- **Digital Output** – Select **Digital Output** to enable the digital output pin terminal (optional) and enable the **Output** subtab for additional configuration options related to outputs.
  - **HW Connection** – This parameter determines whether the digital output terminal for a given output pin is displayed on the schematic. If displayed, the pin outputs the digital signal supplied by a hardware source. If not displayed, the output logic level is determined by CPU writes. If this option is not selected, the terminal is not displayed and it is controlled only by the CPU/DMA.

When this option is selected, the initial drive state of the pin is forced to "High (1)." This setting cannot be changed. This setting may lead to a glitch on the pin during device initialization. To avoid this behavior, Cypress recommends to configure the pin as "analog high-z" in the Component and then change the pin drive mode to "strong drive" in firmware after the Component to which it is connected has been initialized and enabled.

- **Output Enable** – This parameter allows the use of the output enable feature of pins and displays the output enable input terminal. The output enable feature allows a hardware (DSI) signal to control the pin's output drivers without requiring the CPU. A high logic level configures the output drivers, as set in the **Drive Mode** parameter. A low logic level disables the output drivers and places the pin in HI-Z drive mode.

- **Bidirectional** – Enabling the **Bidirectional** parameter is functionally equivalent to enabling the **Digital Input** with **HW Connection** and the **Digital Output** with **HW Connection** parameters. The difference is that only a single bidirectional terminal is displayed on the Component symbol rather than separate input and output terminals. The terminal can then connect to other bidirectional type connections. Both the **Input** subtab and **Output** subtab are enabled for further configuration.

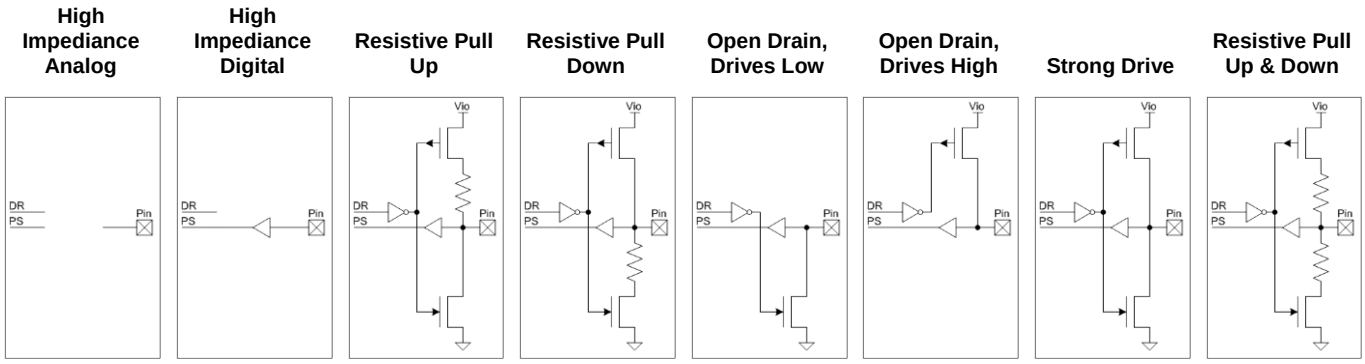
- **Show External Terminal** – Allows connections to Off-Chip Components in the Component Catalog to illustrate circuitry external the chip.

Drive Mode

This parameter configures the pin to provide one of the eight available pin drive modes. The defaults and legal choices are influenced from the **Type** selections. Refer to the device datasheet for more details on each drive mode. A diagram shows the circuit representation for each drive mode as it is selected.

- If the **Type** is Digital Input or Digital Input/Analog, the default is High Impedance Digital.
- If the **Type** is Analog, the default is High Impedance Analog. This is the only pin drive mode that can support purely Analog pins.
- If the **Type** is Bidirectional or Bidirectional/Analog, the default is Open Drain, Drives Low.
- All other pin types default to Strong Drive.

The diagram for each drive mode is as follows:



- The "OUT" connection is driven from either the digital system (when the Digital Output terminal is connected) or the port OUT Register (when HW Connection is disabled).
- The "IN" connection drives the port IN register. It also drives the digital system if the Digital Input terminal is enabled and connected.
- The analog connection connects directly to the pin and does not go through the digital logic.

Notes

- If any of the three resistive drive modes (Resistive Pull Up, Resistive Pull Down, Resistive Pull Up/Down) is used, setting the output **Drive Level** to "Vref" does not work.
- For over voltage tolerance to work on pins with such capability (GPIO_OVT and SIO), the pin must be in one of the following **Drive Modes**:
  - High Impedance Analog
  - High Impedance Digital
  - Open Drain, Drives Low

Initial Drive State

This parameter specifies the pin-specific initial value written to the pin's OUT Register after a device reset/power-on. This happens during port configuration process in device start-up code. Unless changed manually or automatically configured to logic high (1) by the pin **Drive Mode** parameter, all pins default to logic low (0).The initial drive state is configured high (1) by default only for the "Resistive Pull Up" and "Resistive Pull Up/Down" **Drive Modes** to ensure the pull-up resistor is active.

Note the initial drive state is not configurable for hardware digital output pins. This is driven by the signal attached to it and hence the initial drive state does not get set at the output.

Minimum Supply Voltage

This parameter selects the requested minimum high logic level output voltage. The requested voltage must be provided by the V_{DDIO} supply input. If left blank, the Component has no special voltage requirements.

Maximum Frequency

The maximum frequency parameter determines the maximum rise and fall ramp rate for the pin as it changes output logic levels. Slower switching signals can benefit from slower transition edge rates, reducing radiated EMI and coupling with neighboring signals.

- 200 MHz – Requires HSIO
- 100 MHz – **(Default)** Requires GPIO, GPIO_OVR or HSIO
- 80 MHz – Requires SIO, HSIO, GPIO_OVT, GPIO_5V or GPIO_5V_SMC
- 50 MHz – Requires SIO, HSIO, GPIO_OVT, GPIO_5V or GPIO_5V_SMC
- 1 MHz – Requires GPIO, GPIO_OVT, GPIO_5V or GPIO_5V_SMC
- I2C SM (100kbps) – Requires GPIO_OVT
- I2C FM (400 kbps) – Requires GPIO, GPIO_OVT, GPIO_5V, GPIO_5V_SMC or SIO
- I2C FM+ (1.0 Mbps) – Requires GPIO_5V_SMC
- I2C HS (3.4 Mbps) – Requires GPIO_5V_SMC

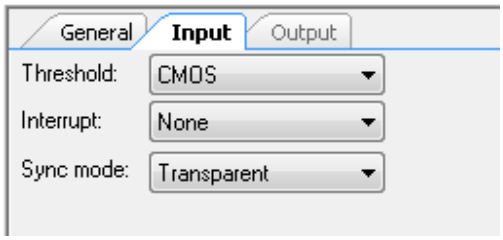
Hot Swap

A pin configured for hot swap capability is mapped to an SIO or a GPIO Over-Voltage Tolerance (GPIO_OVT) pin that supports this capability in hardware. Hot swap capability allows the voltage present on the pin to rise above the pin's V_{DDIO} voltage, up to 6.0 V. Hot swap also does not allow a pin with any voltage up to 6.0 V present to leak current into the device even when the device is not powered. Hot swap is useful for connecting the device when unpowered to a communications bus like I^2C without shorting the bus or back powering the device.

- Disabled – **Default**
- Enabled – Requires SIO or GPIO_OVT

Input Subtab

The **Input** subtab specifies input settings. If the **Type** is not "Digital Input" or "Bidirectional," this subtab is disabled because you do not need to specify input information.



Threshold

This parameter selects the threshold levels that define a logic high level (1) and a logic low level (0) for the entire port on which that pin is placed. "CMOS" is the default and should be used for the vast majority of application connections. The other threshold levels allow for easy interconnect with devices with custom interface requirements that differ from those of CMOS. A pin specified as "CMOS or LVTTL" will default to CMOS, but may be configured as LVTTL in order to be placed in a port configured as LVTTL. Thresholds that are derived from Vddio, a routed Vref, or an Internal Vref (1.2 V) require the use of an SIO pin. This setting is on a port, so if it is in the same group where there are different settings, the port cannot be contiguous.

Threshold	Allowed Multiplier values	
	SIO not present	SIO present
CMOS	1	1
LVTTL	1	1
CMOS or LVTTL	1	1
CMOS 1.8V	1	1
Vddio	N/A	0.4, 0.5
Vref	N/A	1
0.5 x Vref	N/A	All
Vref (Internal)	N/A	1
0.5 x Vref (Internal)	N/A	All

The list of possible multiplier values include:

- 1.00 x – **Default**
- 1.25 x
- 1.49 x

- 1.67 x
- 2.08 x
- 2.50 x
- 2.78 x
- 4.16 x

Note Vref and Internal Vref share resources with the Drive Level parameter in the **Output** subtab. Therefore, they need to match if using these options in a pin configured as both Digital input pin and a Digital output.

Note When using **0.5 x Vref** or **0.5 x Vref (Internal)** options, the voltage on Vref or the 1.2 V Vref (Internal) must adhere to the lower limit of 1 V and the upper limit of (VDD – 0.4 V). That is,

1 V < (Vref x multiplier) < (VDD - 0.4 V)

Operating beyond this limit will result in incorrect operation. The Vref Component in the relation above is the full Vref, **NOT** 0.5 x Vref.

Interrupt

This parameter selects whether the pin can generate an interrupt and, if selected, the interrupt type. All pins on a port logically OR their interrupts together and generate a single interrupt signal via a dedicated Port Interrupt. A device level Combined Port Interrupt (**AllPortInt**) signal can also be used. Both types are available through the Global Signal Reference Component.

After an interrupt occurs, the interrupt source must be cleared in software to clear the latched pin events to enable detection of future events. This is accomplished by calling the Cy_GPIO_ClearInterrupt() function.

The following options are supported:

- None – **Default**
- Rising Edge
- Falling Edge
- Both Edges

Sync Mode

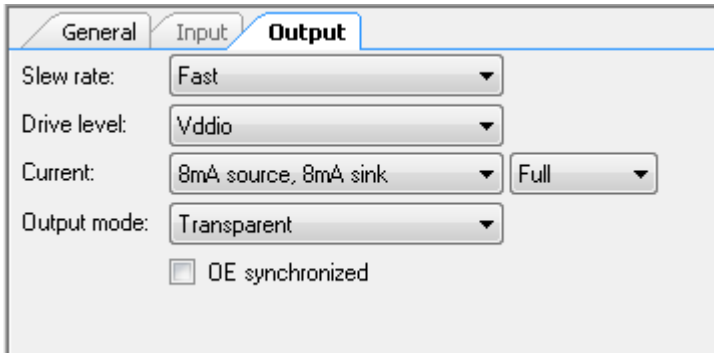
Input synchronization can be applied at the pins to synchronize all signals entering the device to the HFCLK. The synchronization is recommended for signals entering UDB logic.

- Transparent – **Default**
- Single-Sync

- Double-Sync

Output Subtab

The **Output** subtab specifies output settings. If the **Type** is not **Digital Output** or **Bidirectional**, this tab is disabled as the output information does not need to be specified.



Slew Rate

The slew rate parameter determines the rise and fall ramp rate for the pin as it changes output logic levels.

Fast mode is chosen by default and should be used for signals that switch at greater than 1 MHz. You can select slow mode for signals less than 1 MHz switching rate and benefit from slower transition edge rates, which reduce radiated EMI and coupling with neighboring signals.

- Fast – Default
- Slow

Drive Level

This parameter selects the output drive voltage supply sourced by the pin. All pins can supply their respective V_{DDIO} supply voltages. SIO pins can also supply a programmable or analog routed voltage for interface with devices at a different potential than the SIOs V_{DDIO} voltage. SIOs can use a routed external Vref or an internal Vref (1.2 V) as the drive voltage supply.

Threshold	Allowed Multiplier values		Notes
	SIO not present	SIO present	
Vddio	1	1	**Default** – Applicable for all devices
Vref	N/A	All	Requires SIO
Vref (Internal)	N/A	All	Requires SIO

The list of multiplier values include:

- 1.00 x – **Default**
- 1.25 x
- 1.49 x
- 1.67 x
- 2.08 x
- 2.50 x
- 2.78 x
- 4.16 x

Note External Vref and Internal Vref share resources with the **Threshold** parameter. Therefore they need to match if using these options in a pin configured as both Digital input pin and a Digital output pin.

Note: When using **Vref** or **Vref (Internal)** options, the voltage on Vref or the 1.2V Vref (Internal) must adhere to the lower limit of 1V and the upper limit of (VDD – 0.4V). i.e.

1V < (Vref x multiplier) < (VDD - 0.4V)

Operating beyond this limit will result in incorrect operation.

Note The drive mode must be Strong Drive if you are using either **Vref** or **Vref (Internal)**.

Current

The drive current selection determines the maximum nominal logic level current required for a specific pin. GPIOs can supply more current at the cost of logic level compliance or can have a maximum value that is less than listed, based on system voltages. See the device datasheet for more details on drive currents.

- 5mA source, 5mA sink
- 8mA source, 8mA sink – **Default**
- 16mA source, 16mA sink – Requires HSIO or GPIO_5V_SMC
- 40mA source, 40mA sink – Requires GPIO_5V_SMC

Drive strength can be defined to select how much of the IO driver is enabled to provide additional control of the rising and falling edge slew rates. Drive strength options include:

- Full – **Default**
- Half

- Quarter
- Eighth

Output Mode

Output synchronization reduces pin-to-pin output signal skew in high-speed signals requiring minimal signal skew. By default, this parameter is set to **Transparent** and no synchronization occurs. If **Single-Sync** is selected, the output signal is synchronized to the HFCLK.

- Transparent – **Default**
- Single-Sync

OE Synchronized

Output Enable synchronization allows the Output Enable signal to be synchronized to the output clock.

- Disabled – **Default**
- Enabled

Application Programming Interface

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

By default, PSoC Creator assigns the instance name "Pin_1" to the first instance of a GPIO Component in a given design. You can rename it to any unique value that follows the syntactic rules for identifiers.

Driver Usage

The initial component configuration is performed at a port level in the *cyfitter_cfg.c* file and therefore there is no need to explicitly initialize the component from the application code.

PSoC Creator generates a list of #defines in the *cyfitter_cfg.c* file, which is located in the “**Pins and Interrupts**” directory of the workspace explorer. Pass the generated port definition (Pin_1_0_PORT) and the pin number definition (Pin_1_0_NUM) to the cy_gpio driver functions in the application to perform run-time changes. For example:

```
Cy_GPIO_Inv( Pin_1_0_PORT , Pin_1_0_NUM );
```

Sample Firmware Source Code

PSoC Creator provides many example projects that include schematics and example code in the Find Example Project 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 **Find Example Project** topic in the PSoC Creator Help for more information.

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

The GPIO Component does not have any specific deviations.

Functional Description

The GPIO Component allows easy configuration of common pin settings in most designs. It also provides more advanced configurations for those designs requiring settings beyond the basic functionality. This section highlights some of the more advanced pin modes that may not be obvious from the given parameter descriptions.

- **Port Interrupts –** Port interrupts allow signal transitions on the port pins to trigger an interrupt. To use these, you must connect a LEVEL or Derived interrupt type to the IRQ terminal of the Pins Component. This interrupt can be triggered in all power modes. If you choose to connect a RISING_EDGE interrupt to either the IRQ terminal or the Pin I/O terminal, then routing resources will be used and the interrupt will be limited to active and sleep power modes.

Resources

Each GPIO Component consumes one physical pin per bit of the **Number of Pins** parameter.

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 major changes in the Component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
1.0.e	Updated datasheet	Updated **HW Connection** check box information in the Type section to describe a possible glitch on the pin during device initialization
1.0.d	Minor datasheet edits.	
1.0.c	Updated datasheet.	Removed reference to unsupported interrupt clearing function. Minor update to MISRA section.
1.0.b	Updated Driver Usage section.	
1.0.a	Updated datasheet API section.	All API information is located in the PDL documentation.
1.0	Initial version	

© Cypress Semiconductor Corporation, 2016-2018. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC (“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. 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.

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.