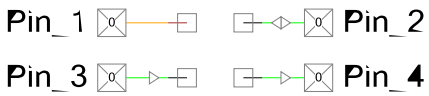
引脚
2.0

特性



- 快速设置所有引脚参数和驱动模式
- 允许 PSoC Creator 自动放置和路由信号
- 允许与一个或多个引脚同时交互

概述

引脚组件允许硬件资源连接到物理端口引脚。该组件通过合理配置的物理 IO 引脚来访问外部信号。它允许电气特性与一个或多个引脚相互关联；PSoC Creator 随后使用这些特性自动放置和路由组件中的信号。

这些引脚可以分别用原理图导线连接和软件来控制，也可以由二者共同控制。要利用组件 API 访问引脚组件，该组件必须是相邻的非跨端口组件。这样可以确保引脚被正确映射到单个物理端口。仅可以从原理图中或通过全局单引脚 API 访问跨端口或非相邻的引脚组件（有关详细信息，请参考应用编程接口部分中的内容）。

注意：在与全局 API 一起使用的引脚组件中包含为每个引脚创建的#defines。

引脚组件可以被配置成多种类型的组合。为方便起见，组件目录中提供了 4 种预先配置的引脚组件：模拟引脚、数字双向引脚、数字输入引脚和数字输出引脚。

何时使用引脚组件

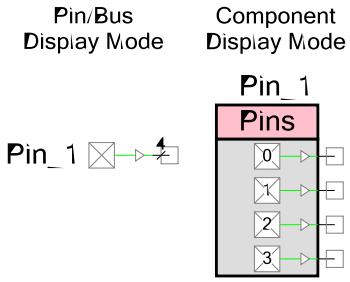
当设计必须通过物理 IO 引脚生成或访问器件外的信号时，使用该引脚组件。引脚组件是目录中最通用的组件。例如，引脚组件可用于连接电位器、按钮、LED 和外设传感器，如接近传感器和加速传感器接口。

输入/输出连接

本节介绍引脚组件的各种输入和输出接口。

引脚显示

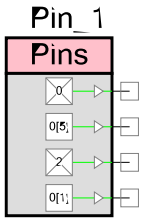
引脚可以配置为数字输入、数字输出、数字双向和模拟的复杂组合。简单配置通常作为单一引脚显示。更复杂的引脚类型作为带有边界框的标准组件显示（请参考下面框图）。



最常见的默认配置在以下章节给出。

锁定引脚的显示

使用 PSoC Creator Design-Wide Resources Pin Editor 将引脚组件分配到物理 GPIO 或 SIO 引脚时，组件的工具提示会显示出特定引脚的分配。如果锁定引脚分配，组件显示会指明引脚分配，如以下示例所示：



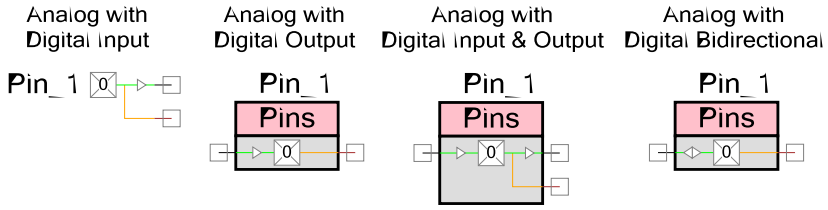
注意：如果将引脚组件设置为 **Display as Bus**（显示为总线），组件显示不予显示任何锁定引脚分配；然而，工具提示仍然显示此信息。

模拟

设计过程中，任何需要在器件引脚与连有模拟导线的内部模拟终端之间进行的连接，均要将引脚组件配置为模拟组件。配置为模拟时，终端将显示在符号右侧，并且连接有模拟导线颜色的连接线。



此外，模拟引脚组件还支持数字输入和/或输出连接及双向连接。数字输出与模拟信号可以组合在同一个引脚组件中。这可以用于某些应用中；然而，这不是通用的使用情况，应谨慎使用。



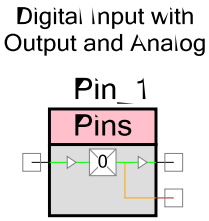
数字输入

设计过程中，任何与内部数字输入之间连接，或任何需要由 CPU/DMA 读取状态的引脚，均需要被配置为数字输入。任何使用数字输入模式的引脚状态均可以由 CPU/DMA 来读取。此外，如果在原理图上显示出引脚终端（HW 连接），则用户可以将之连接到原理图中的其他组件。

引脚终端属性为可视时，会被显示在组件符号的右侧。该连接使用数字导线的颜色绘制，其导线上连接的小输入缓冲器用来提示信号方向。



此外，数字输入引脚组件还支持数字输出和模拟连接。

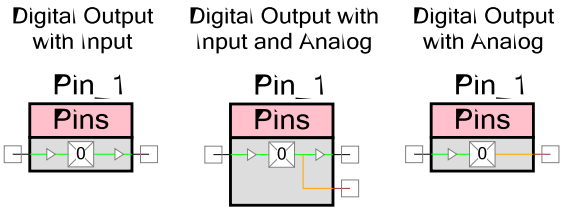


数字输出

任何被驱动到逻辑高电平或低电平的引脚，均要被配置为数字输出。任何数字输出引脚的状态均可以由 CPU/DMA 来写入。此外，如果在原理图上显示出引脚终端（HW 连接），则用户可以将之连接到原理图中的其他组件。引脚终端属性为可视时，会被显示在组件符号的右侧。该连接使用数字导线的颜色绘制，其导线上连接的小输入缓冲器用来提示信号方向。



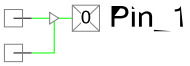
此外，数字输出引脚组件还支持数字输入和模拟连接。



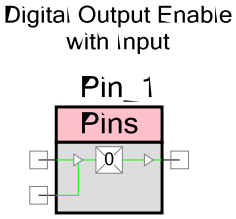
数字输出使能

当需要使用数字逻辑快速控制引脚的输出驱动，而又不需要 CPU 对其进行干预时，可以采用数字输出使能。在此终端上的逻辑高电平使能引脚输出驱动，其配置方式是通过 **General**（通用）子选项卡上的 **Drive Mode**（驱动模式）参数。此终端的逻辑低电平禁用引脚输出驱动，并使引脚采用 HI-Z 驱动模式。当使用原理图连接配置带有数字输出的组件并选中数字输出使能时，此终端将被显示。数字输出使能显示在组件符号左侧，连接到数字输出缓冲器。该连接使用数字导线颜色进行绘制。

当该引脚组件被设置为 **Display as Bus**（显示为总线）时，由于所有引脚共享同一个输出使能，所以无论引脚组件的宽度如何，都仅提供一个输出使能。相反的，引脚组件未被配置为总线时，则根据每个引脚提供单个输出使能。



数字输出使能引脚组件还同时支持输入和模拟连接。

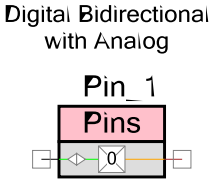


数字双向

设计过程中，任何需要在器件引脚和内部数字双向终端之间进行的连接都采用数字双向模式。数字双向模式最常用于通信组件，如 I^2C。当引脚被配置为数字双向模式时，终端会显示在符号左侧，并带有数据导线颜色的连接，该连接带有显示信号为双向模式的输入和输出缓冲器。



双向引脚组件还支持模拟连接。



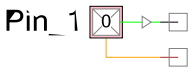
参考电压（Vref）

配置引脚组件以使用 Vref 信号：

- 使用数字输入或双向终端，并将 **Input**（输入）子选项卡上的 **Threshold**（阀值）参数设置为 **Vref**，或
- 使用数字输出或双向终端，并将 **Output**（输出）子选项卡上的 **Drive Level**（驱动电平）配置为 **Vref**

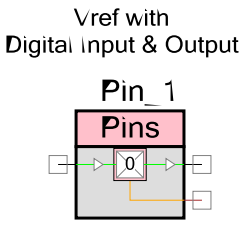
使用 Vref 需要一个 SIO 引脚，它显示为粉红色外形。所有引脚均可以提供各自的 V_{DDIO} 供电电压。此外，SIO 引脚还可为器件接口提供可编程或模拟路由电压，来配合可能不同于 SIO 的 Vddio 电压的接口。Vref 终端提供向 SIO 引脚供电的模拟路由电压。SIO 引脚还使用 Vref 输入作为 SIO 的输入阀值。

Vref 信号显示在组件右侧，从 SIO 单个引脚或 SIO 引脚对底侧伸出，这取决于该信号的配置方法。每个 SIO 引脚对共享单个 Vref 输入。



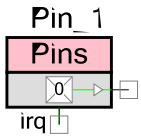
Vref 仅可以与另一个数字输入或输出连接一起使用。

注意：使用 Vref 时，无法选择 **Analog**（模拟）。



IRQ

要配置带有中断模式的引脚组件，必须使用数字输入，然后在 **Input**（输入）子选项卡上配置 **Interrupt**（中断）参数。当使用中断模式时，显示引脚组件，带有边界框，IRQ 从组件底部向外伸出。典型的使用情况是将中断组件与此终端连接。如果使用引脚中断将此模块从睡眠或休眠低功耗模式唤醒，连接至引脚 irq 终端的中断组件可能不会将 InterruptType 设置为“RISING_EDGE”。



中断模式可以用于所有引脚组件配置，只要配置包含数字输入即可。

有关配置该中断的更多信息，请参考 **Input（输入）子选项卡**中的 **Interrupt**（中断）参数的说明。

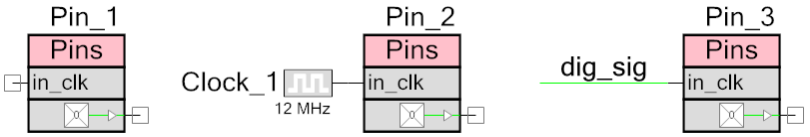
任何数字输入硬件连接都可以连接到 ISR 组件，提供用来生成高或低逻辑电平及边沿事件引脚中断的功能。使用电平中断的数字输入连接时，不使用通过此参数配置的专用引脚中断逻辑，反而消耗 DSI 路由资源。

PSoC 4 特定连接

以下终端仅适用于 PSoC 4 器件。当未将 **Sync Mode** 设置为“透明”时，可以使用这些终端。

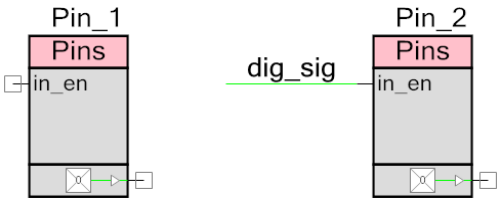
In Clock

在 PSoC 4 上，引脚组件可以将时钟组件或数字信号作为输入同步逻辑的时钟信号使用。如果位于 **Clocking**（时钟）选项卡上的 **In Clock** 参数指定为 **External**（外部），**in_clk** 终端将被显示，以便原理图连接。



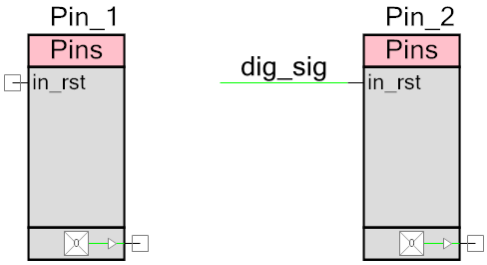
In Clock Enable

在 PSoC 4 上，引脚组件可以将数字信号作为输入同步逻辑的时钟使能信号使用。如果位于 **Clocking**（时钟）页面的 **In Clk En** 参数指定为 **External**（外部），**in_en** 终端将被显示，以便原理图连接。



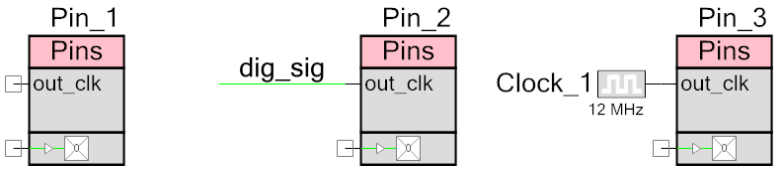
In Reset

在 PSoC 4 上，引脚组件可以将数字信号作为输入同步逻辑的复位信号使用。如果位于 **Clocking**（时钟）页面的 **In Reset** 参数指定为 **External**（外部），**in_rst** 终端将被显示，以便原理图连接。



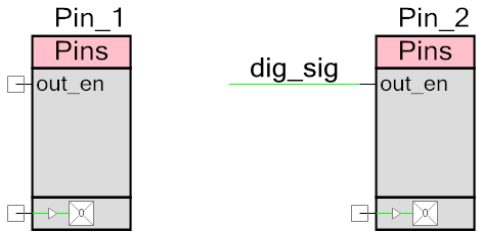
Out Clock

在 PSoC 4 上，引脚组件可以将时钟组件或数字信号作为输出同步逻辑的时钟信号使用。如果位于 **Clocking**（时钟）页面的 **Out Clock** 参数指定为 **External**（外部），**out_clk** 终端将被显示，以便原理图连接。请注意，通过该配置可以将时钟信号驱动到 PSoC 4 器件上的引脚。更多有关其的使用情况，请参考 **Output Mode** 参数。



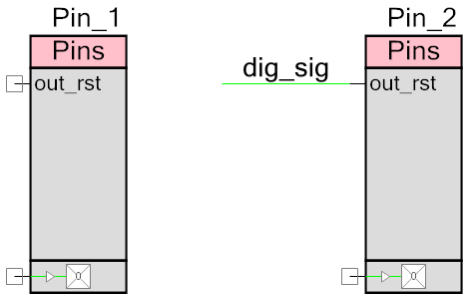
Out Clock Enable

在 PSoC 4 上，引脚组件可以将数字信号作为输入同步逻辑的时钟使能信号使用。如果位于 **Clocking**（时钟）页面上的 **Out Clk En** 参数指定为 **External**（外部），**out_en** 终端将被显示，以便原理图连接。如果 **Output Mode** 设置为 **Clock** 或 **Clock-Inverted**，该信号将作为时钟的使能信号使用。



Out Reset

在 PSoC 4 上，引脚组件可以将数字信号作为输出同步逻辑的复位信号使用。如果位于 **Clocking**（时钟）页面的 **Out Reset** 参数指定为 **External**（外部），**out_rst** 终端将被显示，以便原理图连接。

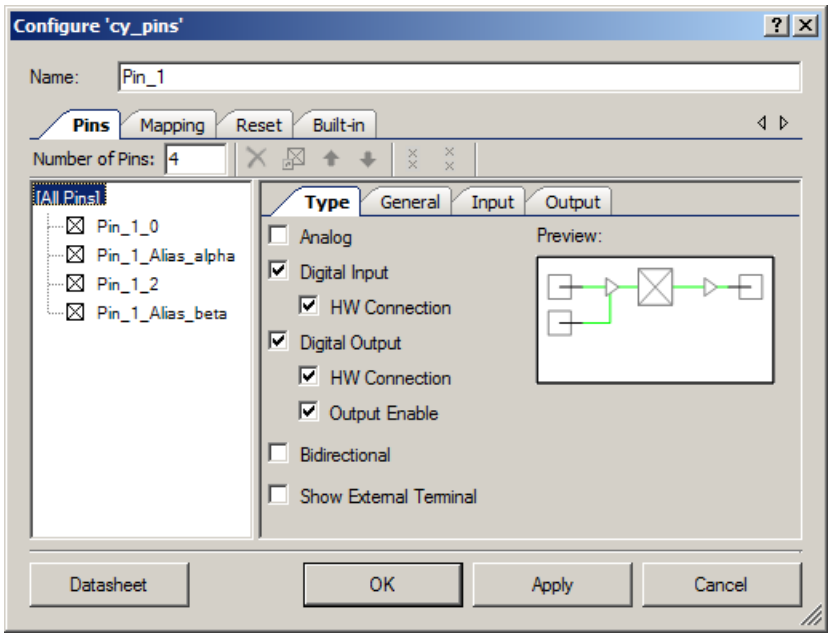


组件参数

将引脚组件拖放到您的设计原理图上，并双击以打开 **Configure**（配置）对话框。此对话框用于设置组件参数，例如，上电复位状态和物理引脚映射约束。所有参数被分配到子选项卡中。

Pins（引脚）选项卡

该 **Pins**（引脚）选项卡有 3 个区域：工具栏、引脚树和另外一组子选项卡。工具栏用于确定由组件管理的物理引脚数量及顺序。子选项卡用来设置引脚特定的属性，例如，类型、方向、驱动模式和初始状态。引脚树配合子选项卡，以便允许您选择适合这些属性的特定引脚。



Toolbar（工具栏）

包含以下命令：

- **Number of Pins**（引脚数） — 由组件控制的器件引脚数。有效值范围介于 1-64 之间。默认值为 **1**。

  注意：某些配置仅可以放置到单独的物理端口中；因此，默认最大引脚数被限定到 8 或更少。将组件配置为非相邻或跨端口引脚时，最大引脚数可以设置为 64，因为这些引脚不再需要放置到单独物理端口中。

- **Delete Pin**（清除引脚） — 从树中清除所选引脚。
- **Add/Change Alias**（添加/更改别名） — 打开对话框，可以在树中添加或更改所选引脚的别名。还可以双击引脚或按**[F2]**打开对话框。

-  **Move Up/Down**（上移/下移） — 在树中上移或下移所选引脚。
-  **Pair/Unpair SIOs**（成对/不成对 SIO） — 在树中选择的成对或不成对 SIO 引脚（标识为粉红色外形）。

  此控制表示需要 SIO 的引脚是否应放置在器件上相同的 SIO 引脚对中。成对配置引脚会产生较少的物理 SIO 引脚闲置。这是因为需要 SIO 的不成对引脚无法与另一个需要 SIO 的引脚共享器件上的 SIO 对。要在器件上共享 SIO 对，这些引脚必须具有单对设置，这对设置的配置方法相同，并且是互邻的。

  如果 **Hot Swap**（热插拔）设置为“真”、**Threshold**（阀值）设置为除 **LVTTL** 或 **CMOS** 以外的值、**Drive Level**（驱动电平）设置为 **Vref** 和/或 **Drive Current**（驱动电流）设置为 **25 mA sink**（25 mA 灌电流），那么引脚需要 SIO。

Pin Tree（引脚树）

该区域显示组件的所有引脚。您可以使用工具栏命令和子选项卡单独选择要使用的一个或多个引脚。每个引脚显示名称的命名方式为引脚组件名称+ ‘_’ +单个引脚别名。

Type（类型）子选项卡

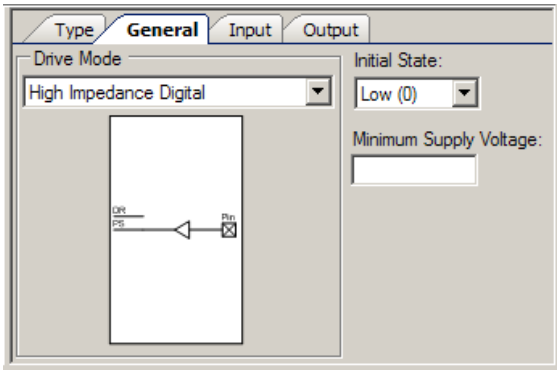
这是针对 **Pins**（引脚）选项卡显示的默认子选项卡。这是使用复选框选择组件引脚类型的位置。预览区显示所选引脚组件符号的外观，其中包含为该特定引脚选择的各种选项。

- **Analog**（模拟）— 选择 **Analog**（模拟），可以使能模拟引脚终端，以便支持路由到其他组件的模拟信号。选择模拟会强制把引脚配置在 GPIO，而非 SIO 上。
- **Digital Input**（数字输入） — 选择 **Digital Input**（数字输入），可以使能数字输入引脚终端（可选）和与输入相关的其他配置选项的 **Input**（输入）子选项卡。
  - **HW Connection**（HW 连接）— 此参数确定输入引脚的数字输入终端是否在原理图上显示。若显示，则该引脚向数字系统互联（DSI）提供数字信号，以供硬件组件使用。独立于此选择，所有引脚始终由 CPU 通过寄存器或 API 来读取。如果不选择此选项，则终端不会显示，并仅由软件 API 来控制。
- **Digital Output**（数字输出）— 选择 **Digital Output**（数字输出），可以使能数字输出引脚终端（可选）和与输出相关的其他配置选项的 **Output**（输出）子选项卡。
  - **HW Connection**（HW 连接） — 此参数确定指定输出引脚的数字输出终端是否在原理图上显示。若显示，引脚通过 DSI 输出硬件组件所提供的数字信号。若未显示，则通过 CPU 寄存器或 API 写入来确定输出逻辑电平。如果不选择此选项，则终端不会显示，并仅由软件 API 来控制。
  - **Output Enable**（输出使能）— 此参数支持引脚使用是你使能输出功能，并显示输出使能输入终端。输出使能功能支持硬件信号来控制引脚输出驱动，无需 CPU 写入

寄存器。高逻辑电平配置输出驱动，这是使用 **Drive Mode**（驱动模式）参数所设置的。低逻辑电平禁用输出驱动，并将引脚放置到高阻驱动模式中。

- **Bidirectional**（双向） — 使能 **Bidirectional**（双向）参数在功能上等效于使能 **Digital Input**（数字输入）与 **HW Connection**（HW 连接）和 **Digital Output**（数字输出）与 **HW Connection**（HW 连接）参数。区别仅在于单个双向终端显示在组件符号上，而非单独的输入和输出终端上。使能 **Input**（输入）和 **Output**（输出）子选项卡，用于进一步配置。
- **Show External Terminal**（显示外部终端） — 支持连接到片外组件以显示连接到 PSoC 的外部电路。

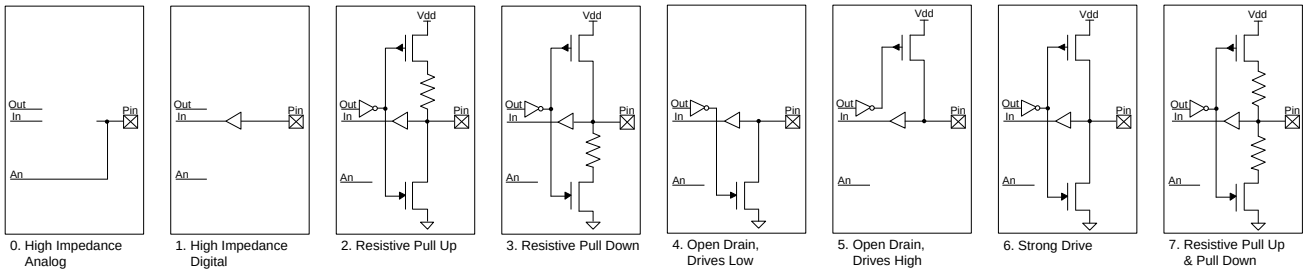
General（通用）子选项卡



General（通用）子选项卡可以设置适用于每个引脚的参数，例如，所选引脚的驱动模式、初始状态和最低供电电压。在此子选项卡上的设置包括：

- **Drive Mode**（驱动模式） — 此参数用于配置引脚以提供 8 个可用引脚驱动模式之一。在 **Type**（类型）子选项卡上的选择影响默认和法定选项。有关每个驱动模式的详细信息，请参见器件数据手册。子选项卡中的图形显示了被选驱动模式的电路实现。
  - 如果类型是 **Digital Input**（数字输入）或 **Digital Input**（数字输入）**/Analog**（模拟），则默认设置为 **High Impedance Digital**（高阻抗数字）。
  - 如果引脚类型为 **Analog**（模拟），则默认设置为 **High Impedance Analog**（高阻抗模拟）。这是可支持模拟引脚的唯一引脚驱动模式。
  - 如果引脚类型为 **Bidirectional**（双向）或 **Bidirectional**（双向）**/Analog**（模拟），则默认设置为 **Open Drain**，**Drives Low**（开漏驱低）。
  - 其他所有引脚类型默认为 **Strong Drive**（强驱动）。

  每个驱动模式如下图所示：



Out 连接可以从数字系统（数字输出终端被连接）或数据寄存器（HW 连接被禁用）驱动。

如果数字输入终端被使能并处于连接状态，**In** 连接将驱动引脚状态寄存器和数字系统。

An 连接与模拟系统相连。

注意：如果使用 3 种电阻驱动模式（**Resistive Pull Up**（上拉电阻）、**Resistive Pull Down**（下拉电阻）、**Resistive Pull Up/Down**（上拉/下拉电阻））中的任意一种，则无法将输出驱动电平设置为 **Vref**。

注意：PSoC 4000 系列不支持下列模式 — **Resistive Pull Up**（电阻上拉）、**Resistive Pull Down**（电阻下拉）和 **Resistive Pull Up/Down**（电阻上拉/下拉）。

注意：为了使过压容差正常工作，引脚必须处于以下驱动模式之一：**High Impedance Analog**（高阻抗模拟）、**High Impedance Digital**（高阻抗数字）或 **Open Drain, Drives Low**（开漏驱低）。

注意：USBIO 引脚仅支持 **Open Drain，Drives Low**（开漏驱低）和 **Strong Drive**（强驱动）驱动模式。为了将 USBIO 引脚作为输入使用，应使用 **Open Drain, Drives Low**（开漏驱低）驱动模式。

- **Initial State**（初始状态）— 此参数指定加电复位（POR）之后写入引脚数据寄存器中的引脚特定的初始值。上电复位时，所有引脚在硬件中均默认为逻辑低电平（0）。只有在完成配置驱动模式之后，才将初始状态写入引脚，这作为整个器件配置的部件而触发。

默认情况下，在 PSoC 3 和 PSoC 5LP 上，初始状态仅针对 **Resistive Pull Up**（上拉电阻）和 **Resistive Pull Up/Down**（上拉/下拉电阻）驱动模式配置为高电平，从而确保上拉电阻保持活动状态。

默认情况下，在 PSoC 4 上，初始状态针对不露出输出使能终端的数字输出引脚配置为高电平。在这些配置中，数据寄存器用于控制输出使能。

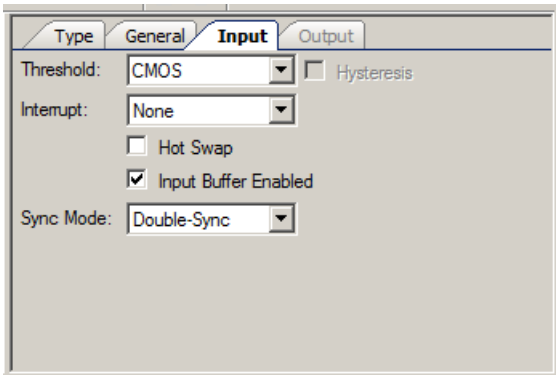
注意：这不应与 **Reset**（复位）主选项卡下的复位状态相互混淆。该属性影响该引脚作为其成员的整个端口的状态，即从其他任何器件配置（包括初始状态配置）之前的复位时开始。

- **Minimum Supply Voltage**（最低供电电压） — 此参数选择所请求的最低高逻辑电平输出电压。这必须通过一个 V_{DDIO} 供电输入来供应所需的电压。此选项确保引脚组件映射到支持

所需输出电压的引脚上。如果保留为空，则该组件没有电压需求，允许放置到由可用的 V_{DDIO} 电压来供电的引脚。

通过在 *<project>.cydwr* 文件的 **System**（系统）选项卡中进行设置可以确定 $V_{DDIO0}/V_{DDIO1}/V_{DDIO2}/V_{DDIO3}$ 的有效值。在没有独立 V_{DDIO} 的器件上，可以确定 V_{DDD} 的有效值。根据所选器件，您可能会有两个 USB 引脚，使用 V_{DDD} 作为放置供电电压。如果该值不小于或等于这些设置的最大值，则无法放置引脚。在此对话框之外执行此范围检查；如果检查失败，在 Notice List（通知列表）窗口显示结果。

Input（输入）子选项卡



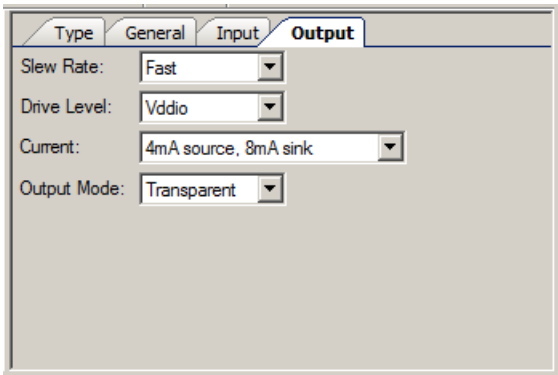
Input（输入）子选项卡用来指定输入设置。如果引脚类型不是在 **Type**（类型）子选项卡中选择的 **Digital Input**（数字输入）或 **Bidirectional**（双向）类型，则无需指定输入信息而禁用此子选项卡。

- **Threshold**（阀值）— 此参数用来选择定义逻辑高电平（1）逻辑低电平（0）的阀值电平。**CMOS** 是默认值，应适用于较大数量的应用连接。为了器件与自定义接口的有效互联，其他阀值级别区别于 CMOS 的阀值。指定为 **CMOS or LVTTL** 的引脚将默认为 CMOS，但可能配置为 LVTTL 以放入配置为 LVTTL 的端口。衍生于 Vddio 或 Vref 的阀值需要使用 SIO 引脚。
  - **CMOS** — 默认值
  - **LVTTL**
  - **CMOS 或 LVTTL**
  - **0.4 x Vddio** — 需要 SIO
  - **0.5 x Vddio** — 需要 SIO
  - **0.5 x Vref** — 需要 SIO
  - **Vref** — 需要 SIO

- **Hysteresis**（迟滞）– 启用或禁用引脚的 SIO 差分迟滞。如果 **Threshold**（阀值）为 **CMOS**、**LVTTL** 或 **CMOS or LVTTL**，则禁用此功能。迟滞控制要求您使用 SIO 引脚。GPIO 引脚始终启用迟滞。
  - **Disabled**（禁用）— 默认设置
  - **Enabled**（使能）
- **Interrupt**（中断） — 此参数用于选择引脚是否生成中断，若选中，则生成中断类型。引脚中断可以在上升沿和/或下降沿触发。如果使用除 **None**（无）以外的任何设置，则须将该组件配置为相邻引脚，以使该组件可以映射到单个物理端口。由于端口中的所有引脚与其中断是逻辑“或”关系，这些引脚生成单个中断信号和中断终端符号，因此需要单个端口。**Interrupt**（中断）参数使用专用引脚中断逻辑，这样可以锁存生成中断事件的引脚。中断触发后，必须调用 Pin_ClearInterrupt()函数来清除锁存的引脚事件，从而可以继续检测将来的事件。如果引脚组件中有不止一个引脚可以生成中断，则可以解码 Pin_ClearInterrupt()返回值以确定哪个引脚生成了中断事件。
  - **None**（无）— 默认
  - **Rising Edge**（上升沿）
  - **Falling Edge**（下降沿）
  - **Both Edges**（上升沿和下降沿）
- **Hot Swap**（热插拔）— 针对热插拔功能配置的引脚映射到支持硬件中此功能的 SIO 引脚。热插拔功能支持引脚上电压的上升沿高于引脚的 V_{DDIO} 电压，高达 6.0 V。此外，热插拔也不允许 6.0 V 以下任何电压下的引脚电流泄漏到 PSoC 器件，甚至在 PSoC 器件未供电时也如此。热插拔功能用于在通信总线（如 I^2C）未供电时连接 PSoC 器件，无需短接总线或向 PSoC 器件反向供电。
  - **Disabled**（禁用） — 默认设置
  - **Enabled**（使能） — 需要 SIO
- **Input Buffer Enabled**（使能输入缓冲区） — 此参数用于使能或禁用引脚的数字输入缓冲区。数字缓冲区需要用来通过 DSI 路由或 CPU 读取的方式来读取或使用引脚上的逻辑电平。引脚作为数字输入时将需要输入缓冲区。默认情况下，模拟引脚禁用数字输入缓冲区以减少引脚在低供电模式下的漏电流。如果引脚类型为 **Analog**（模拟）引脚，则默认设置为 **Disabled**（禁用）。其他所有引脚类型包括含有 **Analog**（模拟）引脚的组合，默认设置为 **Enabled**（使能）。您应禁用输入缓冲区以降低不必要的电流消耗，特别是在有模拟信号的情况下更应如此。
  - **Enabled**（使能）
  - **Disabled**（禁用）

- **Sync Mode**（同步模式）— 默认情况下，在引脚使用双同步器将进入器件的所有信号同步到输入时钟时发生输入同步。在 PSoC 3 和 PSoC 5LP 上，输入时钟始终为 BUS_CLK。在 PSoC 4 上，输入时钟默认为 HFCLK，但可以通过 **Clocking**（时钟）页面上的 **In Clock** 参数进行选择。在有限的情况下，例如应用性能需要异步信号且不违反器件工作要求，可以选择在引脚上禁用输入同步（透明）。在 PSoC 4 上，同步还可以使用单个触发器执行（单独同步）。
  - **Double-Sync**（双重同步）— 默认设置
  - **Single-Sync**（单独同步）（仅针对 PSoC 4）
  - **Transparent**（透明）

Output（输出）子选项卡



该 **Output**（输出）子选项卡指定输出设置。如果引脚类型不是 **Digital Output**（数字输出）或 **Bidirectional**（双向）模式，则由于无需指定输出信息而禁用此选项卡。

- **Slew Rate**（摆率）— 此参数用于确定引脚随输出逻辑电平的变化而发生的上升和下降速率。信号切换频率大于 1 MHz 时需要快速模式。可以选择低速模式用于信号切换频率低于 1 MHz 的，利用较低边沿速率的优点，因为这样可以降低辐射性 EMI 并防止与相邻信号耦合。在 PSoC 4 上，位于同一端口的所有引脚必须具有相同的摆率。
  - **Fast**（快速）— 默认设置
  - **Slow**（慢速）
- **Drive Level**（驱动电平）— 此参数用于选择源自引脚的输出驱动电压供应。所有引脚均可以提供各自的 V_{DDIO} 供电电压。此外，SIO 引脚还为器件接口提供可编程或模拟路由电压，用于可能不同于 SIO V_{DDIO} 电压的接口。
  - **Vddio** — 默认
  - **Vref** — 需要 SIO

注意：如果使用 3 种电阻驱动模式（**Resistive Pull Up**（上拉电阻）、**Resistive Pull Down**（下拉电阻）、**Resistive Pull Up/Down**（上拉/下拉电阻））中的任意一种，则无法将输出驱动电平设置为 **Vref**。

- **Current**（电流）— 驱动电流选择用于确定特定引脚所需的最大额定逻辑电平电流。引脚可以供应更多电流，但要遵从逻辑电平的要求，或最大值应小于系统电压基础上列出的值。有关驱动电流的更多信息，请参见器件数据手册。
  - **4 mA source，8 mA sink**（4 mA 拉电流、8 mA 灌电流） — 默认
  - **4 mA source，25 mA sink**（4 mA 拉电流、25 mA 灌电流） — 需要 SIO
- **Output Mode**（输出模式）— 输出同步可减少高速信号（要求最低信号时滞）中引脚到引脚的信号时滞。默认情况下，输出模式为 Transparent（透明）且不发生同步。如果选择了单独同步，输出信号同步到输出时钟。
  - **Transparent**（透明） — 默认
  - **Single-Sync**（单独同步）
  - **Clock**（时钟）（仅针对 PSoC 4）
  - **Clock-Inverted**（时钟反相）（仅针对 PSoC 4）

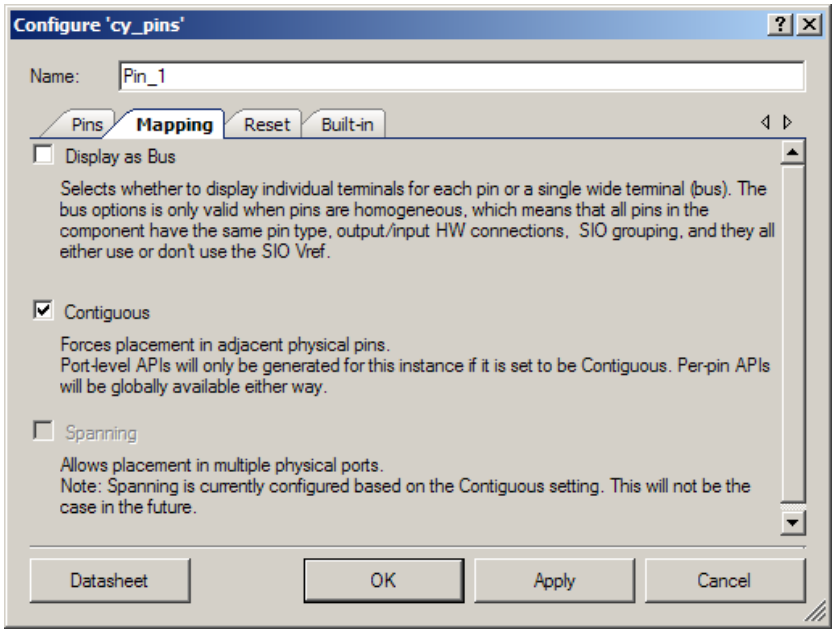
  在 PSoC 3 和 PSoC 5LP 上，输出时钟始终为 BUS_CLK。

  在 PSoC 4 上，输出时钟默认为 HFCLK，但可以通过 **Clocking**（时钟）页面上的 **Out Clock** 参数进行修改。

  如果将 PSoC 4 器件的输出模式设置为 **Clock** 或 **Clock-Inverted**，则外部连接时钟或信号将可以驱动该引脚。在该配置中，数据寄存器（DR）值作为 out_clock 终端的使能值使用，而且必须在引脚定制器中将该值置 1 或通过使用软件设置该值，以将该值设置为高电平。请注意，如果这是一个 HW 输出引脚，那么在该模式下，连接到输出引脚终端的信号值不会影响到该引脚的操作。反而，使用时钟页面上的 **Out Clock Enable** 信号时，如果将 DR 设置为高电平，那么将可以使能时钟的硬件控制。
- **OE Synchronized**（OE 同步）（仅针对 PSoC 4）— 输出使能同步允许将输出使能信号同步到输出时钟。
  - **Disabled**（禁用） — 默认设置
  - **Enabled**（使能）

Mapping（映射）选项卡

该 **Mapping**（映射）选项卡包含用于定义引脚组件在原理图中如何显示及如何映射到物理引脚的参数。



显示为总线

此参数用于选择是否显示各个引脚的单个终端或单个宽终端（总线）。总线选项仅在引脚为同一类型时才有效。这意味着组件中的所有引脚均有相同的引脚类型、输出/输入 HW 连接和 SIO 分组。此外，这些引脚还必须全部使用或全部不使用 SIO Vref。需要大量相同引脚类型时，可以使用显示为总线选项。这样可以节约原理图空间和时间以便于配置和路由。

Contiguous（连续）

此参数用于强制在端口中放置实际相邻的引脚。根据器件数据手册，实际的引脚布局是程序包依赖关系。此选项有下列限制：

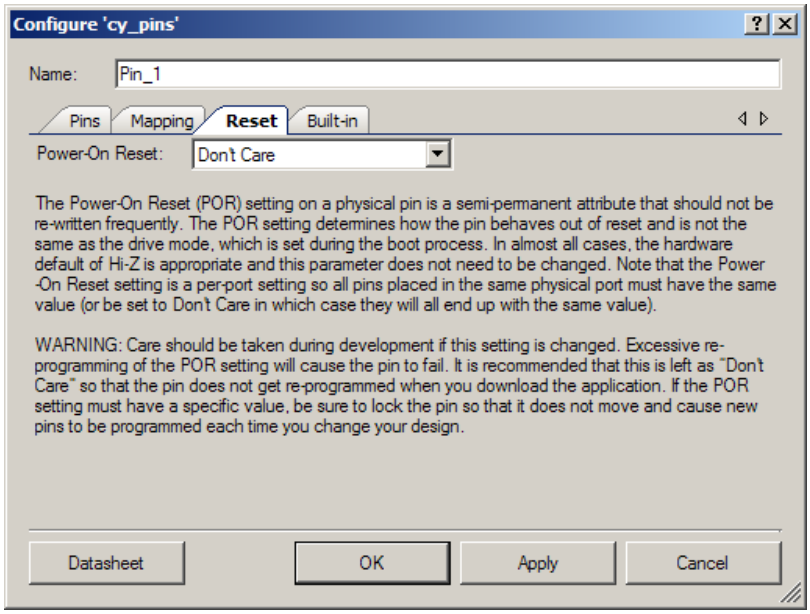
- 若连续，则为组件生成端口级别的 API。若非连续，则不生成端口级别的 API。
- 若连续，组件中的引脚数必须小于或等于 8。

Spanning（跨端口）

此参数用于在多个物理端口中实现引脚布局。目前，这由连续选项所控制，其中，连续布局意味着非跨距，而非连续布局意味着跨距。以后的软件版本将支持 **Spanning**（跨端口）参数的单独控制。

Reset（复位）选项卡

仅在 PSoC 3 和 PSoC 5LP 中，**Reset**（复位）选项卡才可用。



Power-On Reset（上电复位）

物理引脚上的上电复位（POR）设置是非永久性属性，不应频繁重写。POR 设置确定引脚退出复位状态后的行为方式。它不同于驱动模式，后者是在启动过程中设置的。几乎在所有情况下，默认为高阻的硬件均是正确的，无需更改此参数。注意：上电复位设置是单端口设置，因此在同一个物理端口中放置的所有引脚均有相同的值（或设置为 **Don't Care**（无需关注），在此情况下，这些引脚最终均有相同的值）。

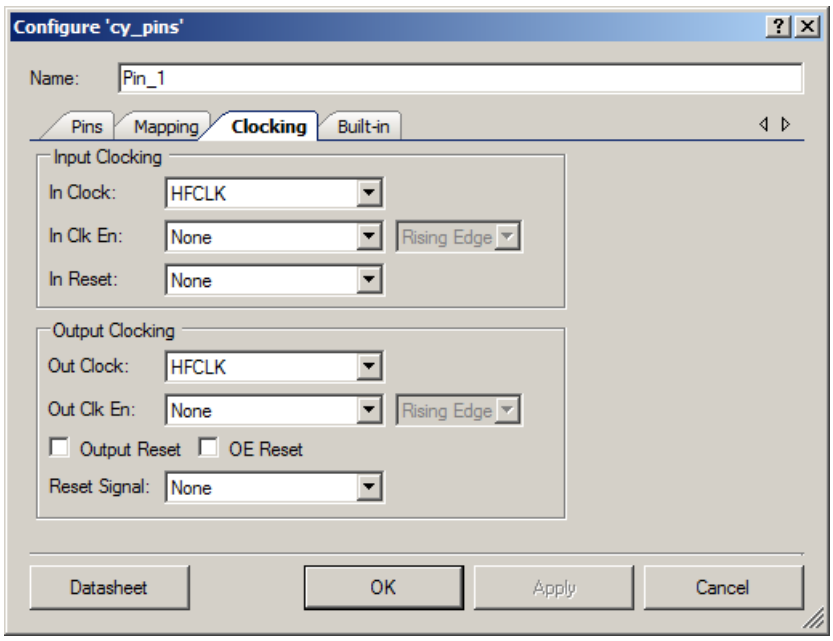
警告：POR 的设置是被编程到 PSoC 的非易失性锁存器（NVL）内的。这些 NVL 的写周期数量有限（请参考器件技术参考手册）。对该设置进行过多的更改及重新编程可导致引脚失效。推荐将其保留为 **Don't Care**（无需关注），这样，下载应用时无需重新编程引脚。如果加电复位（POR）设置必须具有特定值，请确保锁定该引脚，这样，每次更改设计时，无需移动该引脚，从而不会产生需要编程的新引脚。

- **Don't Care**（无需关注） — 默认。保留设置为 **Don't Care**（无需关注），通过放置此组件的物理端口来确定加电复位（POR）设置。如果在端口上放置的所有引脚均设置为 **Don't Care**（无需关注），则该端口默认使用加电复位（POR）设置。否则，无论是否对该物理端口中放置的其他引脚指定 POR（它们必须全部匹配），每个引脚都使用 **Don't Care**（无需关注）设置。
- High-Z Analog（高阻模拟）
- Pulled-Up（上拉）

- Pulled-Down （下拉）

Clocking（时钟）选项卡

Clocking（时钟）选项卡仅在 PSoC 4 上可用。



In Clock

此参数选择用于此引脚组件输入同步逻辑的时钟。默认情况下，HFCLK 作为输入同步时钟使用。还可以通过使能 in_clk 终端（外部）来使用时钟组件或其他信号。要将片外信号作为输入时钟且时滞最小，可以选择配置为输入或双向的此引脚组件中的任何引脚作为 In Clock （Pin_N）。下列任意选项在引脚组件中还可能为反向。

- **HFCLK** — 默认
- HFCLK （inverted）（HFCLK（反向））
- External（外部）
- External （inverted）（外部（反向））
- Pin_N
- Pin_N （inverted）（Pin_N（反向））

In Clk En

此参数选择一个信号，以作为此引脚组件输入同步逻辑的使能信号使用。默认情况下，不使用使能信号，并且始终使能输入同步。原理图中的数字信号可以通过显示并连接 in_en 终端（外部）作为使能信号使用。要将片外信号作为使能信号且延时最短，可以选择配置为输入或双向的此引脚组件中的任何引脚作为 In Clk En （Pin_N）。下列任意选项在引脚组件中还可能为反向。

- **None**（无） — 默认
- External（外部）
- External （inverted）（外部（反向））
- Pin_N
- Pin_N （inverted）（Pin_N（反向））

In Clock Enable 模式

In Clk En 参数右方的下拉菜单控制 In Clock Enable 的使能模式。如果将使能模式设置为 Rising Edge（上升沿），输入同步触发器在从低到高使能信号转换后将立即在时钟周期上转换。如果将使能模式设置为 Level（电平），则当使能信号强度高时，输入同步触发器可以在所有时钟周期上转换。

- **Rising Edge**（上升沿） — 默认
- Level （电平）

In Reset

此参数选择一个信号，以作为此引脚组件输入同步逻辑的复位信号使用。默认情况下，不使用复位。原理图中的数字信号可以通过显示并连接 in_rst 终端（外部）作为使能信号使用。要将片外信号作为复位信号且延时最短，可以选择配置为输入或双向的此引脚组件中的任何引脚作为 In Reset （Pin_N）。下列任意选项在引脚组件中还可能为反向。

- **None**（无） — 默认
- External（外部）
- External （inverted）（外部（反向））
- Pin_N
- Pin_N （inverted）（Pin_N（反向））

Out Clock

此参数选择用于此引脚组件输出同步逻辑的时钟。默认情况下，HFCLK 作为输出同步时钟使用。还可以通过使能 out_clk 终端（外部）来使用时钟组件或其他信号。要将片外信号作为输出时钟且时滞最小，可以选择配置为输入或双向的此引脚组件中的任何引脚作为 In Clock （Pin_N）。下列任意选项在引脚组件中还可能为反向。

- **HFCLK** — 默认
- HFCLK （inverted）（HFCLK（反向））
- External（外部）
- External （inverted）（外部（反向））
- Pin_N
- Pin_N （inverted）（Pin_N（反向））

Out Clk En

此参数选择一个信号，以作为此引脚组件输出同步逻辑的使能信号使用。默认情况下，不使用使能信号，并且始终使能输出同步。原理图中的数字信号可以通过显示并连接 out_en 终端（外部）作为使能信号使用。要将片外信号作为使能信号且延时最短，可以选择配置为输入或双向的此引脚组件中的任何引脚作为 In Clk En （Pin_N）。下列任意选项在引脚组件中还可能为反向。

- **None**（无） — 默认
- External（外部）
- External （inverted）（外部（反向））
- Pin_N
- Pin_N （inverted）（Pin_N（反向））

Out Clock Enable 模式

Out Clk En 参数右方的下拉菜单控制 Out Clock Enable 的使能模式。如果将使能模式设置为 Rising Edge（上升沿），输出同步触发器在从低到高使能信号转换后将立即在时钟周期上使能。如果将使能模式设置为 Level（电平），则当使能信号强度高时，输出同步触发器可以在所有时钟周期上转换。

- **Rising Edge**（上升沿）— 默认
- Level （电平）

Output Reset（输出复位）

允许将选中的 Out Reset（输出复位）信号用于复位输出同步逻辑。

- **Disabled**（禁用） — 默认设置
- Enabled（使能）

OE Reset（OE 复位）

允许将选中的 Out Reset（输出复位）信号用于复位输出使能同步逻辑。

- **Disabled**（禁用） — 默认设置
- Enabled（使能）

Out Reset Signal（输出复位信号）

此参数选择一个信号，以作为此引脚组件输出同步逻辑或输出使能同步逻辑的复位信号使用。默认情况下，不使用复位。原理图中的数字信号可以通过显示并连接 out_rst 终端（外部）作为使能信号使用。要将片外信号作为复位信号且延时最短，可以选择配置为输入或双向的此引脚组件中的任何引脚作为 Out Reset （Pin_N）。下列任意选项在引脚组件中还可能为反向。

- **None**（无） — 默认
- External（外部）
- External （inverted）（外部（反向））
- Pin_N
- Pin_N （inverted）（Pin_N（反向））

应用编程接口

应用编程接口（API）子程序允许您使用软件配置和使用组件。引脚组件在单引脚和组件范围基础上使能访问功能。

单引脚 API

通过使用 *cypins.h* 生成的文件（位于 cy_boot 目录中）中定义的全局 API，可以访问组件中的单个引脚。这些 API 记录在《*系统参考指南*》（Help（帮助）> Documentation（文档））中。

PSoC 3 和 PSoC 5LP API

- CyPins_ReadPin()
- CyPins_SetPin()
- CyPins_ClearPin()
- CyPins_SetPinDriveMode()
- CyPins_ReadPinDriveMode()
- CyPins_FastSlew()
- CyPins_SlowSlew()

这些 API 可以与物理引脚寄存器名称或组件引脚别名一起使用。不建议直接从软件中访问物理引脚，因为通过工具分配到其他函数的相同引脚未实施预防措施。即使只从软件访问引脚，但是，赛普拉斯强烈建议使用引脚组件。通过上述 API 生成组件别名，使用该别名安全访问单个引脚，而不会有性能或存储器损失。

要使用上述 API，该组件会生成 *Pin_aliases.h* 文件中引脚寄存器的别名，其中引脚表示引脚组件的示例名称。默认情况下，该别名是组件名称，其后缀是引脚数：

Pin_x — x 是组件中的引脚（基数为 0）

如果在 Pins configuration（引脚配置）对话框中提供别名，则可以创建其他#define，格式如下：

Pin_<AliasName>

可以使用任何别名。例如，示例引脚的<别名>为“MyAlias”。它将生成别名 Pin_0 和 Pin_MyAlias。要想使用单引脚 API 读取该引脚，您可以选择下列方法：

CyPins_ReadPin(Pin_0)

CyPins_ReadPin(Pin_MyAlias)

PSoC 4 API

- CY_SYS_PINS_READ_PIN()
- CY_SYS_PINS_SET_PIN()
- CY_SYS_PINS_CLEAR_PIN()
- CY_SYS_PINS_SET_DRIVE_MODE()
- CY_SYS_PINS_READ_DRIVE_MODE()

PSoC 4 与 PSoC 3 和 PSoC 5LP 的别名相同。然而，由于引脚寄存器有所不同，因此目前该器件的单引脚 API 不使用这些别名。另外，可以使用定义在 *Pin.h* 中的合适端口寄存器（其中引脚表示引脚组件的示例名称）和端口上的实际引脚位置访问这些寄存器。

例如，如果您想读取位于 P0[0]上的引脚值（假定引脚位于端口 0，引脚 0 上），您需要调用：

```
CY_SYS_PINS_READ_PIN (Pin_PS, 0)
```

组件 API

这些 API 使用简单函数调用，就可以访问组件中的所有引脚。只有在所有引脚均放置在器件上单个物理端口中时，才能有效实施组件范围的 API。只有组件配置为连续组件时，才能生成这些 API。非连续性引脚组件只允许访问单个引脚，如上所述。

默认情况下，PSoC Creator 将实例名称“Pin_1”分配给指定设计中引脚组件的第一个实例。您可以将其重命名为遵循标识符语法规则的任何唯一值。实例名称会成为每个全局函数名称、变量和符号常量的前缀。出于可读性考虑，下表中使用的实例名称为“Pin”。

下表格列出并说明了每个函数的接口。以下各节将更详细地介绍每个函数。

函数	说明
Pin_Read()	读取物理端口，并返回组件中所有引脚的当前值。
Pin_Write()	将值写入组件引脚，同时保护物理端口（假设由多个引脚组件共享）中的其他引脚。
Pin_ReadDataReg()	读取端口数据输出寄存器的当前值，然后返回组件中所有引脚的当前值。
Pin_SetDriveMode()	为引脚组件的各个引脚设置驱动模式
Pin_ClearInterrupt()	清除映射组件的端口上的任何活动中断。返回中断状态寄存器的值

uint8 Pin_Read (void)

说明： 读取相关物理端口（引脚状态寄存器）和根据组件实例的宽度和位的位置屏蔽所需的位。引脚的状态寄存器返回物理引脚上的当前逻辑电平。

参数： 无

返回值： 组件中引脚的当前值是右对齐数字。

其他影响： 无

void Pin_Write (uint8 value)

说明： 将值写入物理端口（数据输出寄存器），适当掩码和移动各位。数据输出寄存器与驱动模式参数相互关联来控制物理引脚上的信号。此函数通过使用适当方法避免更改端口上的其他位（读取-修改-写入或位传送）。

参数： uint8 value：写入组件实例中的值。

返回值： 无

其他影响： 如果使用读取-修改-写入操作（非原子操作）；中断服务子程序（ISR）可能导致此API损坏。中断此API并向引脚组件数据寄存器执行写入的ISR可能导致端口数据损坏。为避免此问题发生，应使用单引脚API（主要方法）或禁用此API相关的中断。

uint8 Pin_ReadDataReg (void)

说明： 读取相关物理端口的数据输出寄存器并根据组件实例的宽度和位的位置屏蔽相应的位。数据输出寄存器与驱动模式参数相互关联来控制物理引脚上的信号。这不同于首选的Pin_Read() API，因为Pin_ReadDataReg()用于读取数据寄存器，而非状态寄存器。对于输出引脚而言，可以使用API来确定该引脚即时写入的值。

参数： 无

返回值： 掩码并移动至组件实例右对齐数字的数据寄存器的当前值。

其他影响： 无

void Pin_SetDriveMode (uint8 mode)

说明： 为引脚组件的各个引脚设置驱动模式。

参数： uint8 mode：所选信号的模式。定义的合法选项：

Pin_1_DM_STRONG	强驱动
Pin_1_DM_OD_HI	漏极开路，驱动高电平
Pin_1_DM_OD_LO	漏极开路，驱动低电平
Pin_1_DM_RES_UP	电阻上拉
Pin_1_DM_RES_DWN	电阻下拉
Pin_1_DM_RES_UPDWN	电阻上拉/下拉
Pin_1_DM_DIG_HIZ	高阻抗数字驱动
Pin_1_DM_ALG_HIZ	高阻抗模拟驱动

返回值： 无

其他影响： 如果使用可分割的读取-修改-写入操作，中断服务子程序（ISR）可能导致此API损坏。中断此API并向引脚组件驱动模式寄存器执行写入的ISR可能导致端口数据损坏。为避免此问题发生，应使用单引脚API（主要方法）或禁用此API相关的中断。

uint8 Pin_ClearInterrupt (void)

说明： 清除与组件连接的所有活动中断，然后返回中断状态寄存器的值，并允许确定哪个引脚生成了中断事件。

参数： 无

返回值： uint8：中断状态寄存器当前值右移。每个引脚设置一个位，假设该引脚生成一个中断事件。例如，在引脚组件中，位0用于引脚0，位1用于引脚1。

其他影响： 全部清除物理端口中断状态寄存器的各个位，不仅仅是那些与引脚组件相关的位。

功能说明

通过引脚组件，可轻松配置大部分设计中的通用引脚设置。它还为那些需要超越基本功能的设置的设计提供了更先进的配置。本节重点介绍了一些更高级的引脚模式，这些模式可能在给定的参数说明中不是那么显而易见。

- **SIO 引脚** — 特殊输入/输出 SIO 引脚（仅在 PSoC 3 和 PSoC 5LP 中才可用）提供了差分输入缓冲区以及调节高电平的输出电压（VOH）的方法。SIO 引脚可以承受比 I/O 电源电压更高的输入电压，并可灌出高达 25 mA 的电流。选择您设计中的 SIO 引脚有几种方法。一旦设置下面任何一个参数，就需要 SIO 引脚。

- **Hot Swap**（热插拔）被设为‘true’（真）
- **Threshold Level**（阈值级别）被设为 0.5 VDDIO、0.4 VDDIO、0.5 VREF、VREF
- **Drive Level**（驱动水平）被设为 Vref
- **Drive Current**（驱动电流）被设为 25 mA 的灌电

SIO 总是被放在端口 12 上，因为引脚位于此端口上。两个 SIO 引脚可从组件配置配对，使的它们能够共享一个通用的参考发生器模块。

- **POR** — 上电复位（POR）选项在 PSoC 3 和 PSoC 5LP 器件中可用。物理引脚上的 POR 设置是非永久性属性，不应频繁重写。POR 设置确定引脚退出复位状态后的行为方式。该设置是端口范围有效的；并不同于驱动模式，后者是在启动过程中设置的。几乎在所有情况下，默认为高阻态（Hi-Z）的硬件均是正确的，无需更改此参数。
- **PSoC 4 的初始状态** — 该参数指出 PSoC 4 器件上引脚的初始状态。当器件没有“Output Enable”（输出使能）终端时，也可将该参数作为一个输出引脚的“输出使能”使用。
- **PSoC 4 引脚时钟** — 通过使用 PSoC 4 引脚的时钟选项，输入和输出值可以与外部时钟同步。如果引脚状态需要与除 HFCLK 外的其他时钟同步，则应使用该选项。
- **将 PSoC 4 时钟连接到引脚** — 与 PSoC 3 和 PSoC 5LP 不同，PSoC 4 中的时钟不能直接连接到引脚终端，除非它被指定为一个时钟。为了使能该模式，可以将 **Output Mode**（输出模式）参数设为‘Clock’（时钟）或‘Clock-Inverted’（时钟反相）。
- **PSoC 端口适配器** — 若干器件（如 PSoC 4000 系列）没有端口适配器，限制了如输入 **Sync Mode**（同步模式）、**Output Mode**（输出模式）和 PSoC4 引脚时钟选项等一些引脚性能。代换器件时，请注意这一点，并参考器件的技术参考手册。

MISRA 合规性

本节介绍了MISRA-C:2004兼容性和本器件的偏差。定义了两种类型的偏差：项目偏差 — 适用于所有PSoC Creator组件的偏差；特定偏差 — 仅适用于该组件的偏差。本节提供了有关组件特定偏差的信息。*《系统参考指南》*的MISRA合规性章节中介绍了项目偏差以及有关MISRA合规性验证环境的信息。

此引脚组件没有任何特定偏差。

示例固件源代码

PSoC Creator 在“Find Example Project”（查找示例项目）对话框中提供了很多包括原理图和代码示例的示例项目。要查看特定组件实例，请打开“Component Catalog”中的对话框或原理图

中的组件示例。要查看通用示例，请打开 Start Page 或 **File** 菜单中的对话框。根据要求，可以通过使用对话框中的 **Filter Options** 选项来限定可选的项目列表。

更多有关信息，请参考《PSoC Creator 帮助》部分中主题为“Find Example Project”（查找样例项目）的内容。

资源

每一个引脚组件消耗 **Number of Pins**（引脚数）参数每个位的一个物理引脚。

API 存储器的使用情况

根据编译器、器件、所使用的 API 数量以及组件的配置情况的不同，组件所用的存储器大小也不一样。下表提供了在某一器件配置中的所有 API 使用的存储器大小。

下表中的存储器大小是在将相应编译器设置为 Release（释放）模式并且优化选项为 Size 的情况下测得的。有关特定的设计，可分析编译器生成的映射文件以确定存储器使用情况。

Configuration （配置）	PSoC 3 （Keil_PK51）		PSoC 4 （GCC）		PSoC 5LP （GCC）	
	闪存 字节	SRAM 字节	闪存 字节	SRAM 字节	闪存 字节	SRAM 字节
默认提供中断	50	0	92	0	96	0

直流和交流电气特性

下表中的值是基于初始特性数据的预估值。

引脚的直流指标

参数	说明	条件	最小值	典型值	最大值	单位
V_{INMAX}	最大输入电压	V_{DDIO}和V_{DDD}的所有容许值	–	–	5.5	V
V_{INREF}	输入电压参考（差分输入模式）		0.5	–	0.52 x V_{DDIO}	V
V_{OUTREF}	输出电压参考（稳压输出模式）					
		V_{DDIO} > 3.7	1	–	V_{DDIO} – 1	V
		V_{DDIO} < 3.7	1	–	V_{DDIO} – 0.5	V
V_{IH}	输入电压上限阈值					

参数	说明	条件	最小值	典型值	最大值	单位
	GPIO模式	CMOS输入	0.7 x V_{DDIO}	–	–	V
	差分输入模式	禁用迟滞	SIO_ref + 0.2	–	–	V
V_{IL}	输入电压下限阈值					
	GPIO模式	CMOS输入	–	–	0.3 x V_{DDIO}	V
	差分输入模式	禁用迟滞	–	–	SIO_ref – 0.2	V
V_{OH}	输出电压高电平					
	非稳压模式	I_{OH} = 4 mA，V_{DDIO} = 3.3 V	V_{DDIO} – 0.4	–	–	V
	稳压模式	I_{OH} = 1 mA	SIO_ref – 0.65	–	SIO_ref + 0.2	V
	稳压模式	I_{OH} = 0.1 mA	SIO_ref – 0.3	–	SIO_ref + 0.2	V
V_{OL}	输出电压低电平					
		V_{DDIO} = 3.30 V，I_{OL} = 25 mA	–	–	0.8	V
		V_{DDIO} = 1.80 V，I_{OL} = 4 mA	–	–	0.4	V
R_{PULLUP}	上拉电阻		3.5	5.6	8.5	kΩ
$R_{PULLDOWN}$	下拉电阻		3.5	5.6	8.5	kΩ
I_{IL}	输入漏电流（绝对值）[1]					
	GPIO	25 °C，V_{DDIO} = 3.0 V	–	–	2	nA
	SIO: $V_{IH} \le V_{DDSIO}$	25 °C，V_{DDSIO} = 3.0 V，V_{IH} = 3.0 V	–	–	14	nA
	SIO: $V_{IH} > V_{DDSIO}$	25 °C，V_{DDSIO} = 0 V，V_{IH} = 3.0 V	–	–	10	µA
C_{IN}	输入电容[1]		–	–	7	pF
V_H	输入电压迟滞（施密特触发器）[1]	单端模式（GPIO模式）	–	40	–	mV
		差分模式	–	35	–	mV
I_{DIODE}	通过保护二极管连到V_{SSIO}的电流		–	–	100	µA

1. 基于器件特性（未经过生产测试）。

引脚的交流指标

参数	说明	条件	最小值	典型值	最大值	单位
TriseF	快速强（Fast Strong）驱动模式下的上升时间(90/10%)[1]	Cload = 25 pF， V_{DDIO} = 3.3 V	–	–	12	ns
TfallF	快速强（Fast Strong）驱动模式下的下降时间(90/10%)[1]	Cload = 25 pF， V_{DDIO} = 3.3 V	–	–	12	ns
TriseS	慢速强（Slow Strong）驱动模式下的上升时间(90/10%)[1]	Cload = 25 pF， V_{DDIO} = 3.0 V	–	–	75	ns
TfallS	低速强（Slow Strong）驱动模式下的下降时间(90/10%)[1]	Cload = 25 pF， V_{DDIO} = 3.0 V	–	–	60	ns
Fsioout	SIO输出工作频率					
	3.3 V < V_{DDIO} < 5.5 V，非稳压输出（GPIO）模式，快速强驱动模式	90/10% V_{DDIO}， Cload = 25 pF	–	–	33	MHz
	1.71 V < V_{DDIO} < 3.3 V，非稳压输出（GPIO）模式，快速强驱动模式	90/10% V_{DDIO}， Cload = 25 pF	–	–	16	MHz
	3.3 V < V_{DDIO} < 5.5 V，非稳压输出（GPIO）模式，低速强驱动模式	90/10% V_{DDIO}， Cload = 25 pF	–	–	5	MHz
	1.71 V < V_{DDIO} < 3.3 V，非稳压输出（GPIO）模式，低速强驱动模式	90/10% V_{DDIO}， Cload = 25 pF	–	–	4	MHz
	3.3 V < V_{DDIO} < 5.5 V，稳压输出模式，快速强驱动模式	在25pF负载时连续开关输出	–	–	20	MHz
	1.71 V < V_{DDIO} < 3.3 V，稳压输出模式，快速强驱动模式	在25pF负载时连续开关输出	–	–	10	MHz
	1.71 V < V_{DDIO} < 5.5 V，稳压输出模式，低速强驱动模式	在25pF负载时连续开关输出	–	–	2.5	MHz
Fsioin	SIO输入工作频率					
	1.71 V < V_{DDIO} < 5.5 V	90/10% V_{DDIO}	–	–	66	MHz

组件更改

本节列出了各版本的主要组件更改内容。

版本	更改说明	更改原因/影响
2.0.b	纠正了PSoC 3和PSoC 5LP上SIO引脚对的缺陷。如果引脚组件用于配置一对SIO，第一个引脚的设置会静静地应用于第二个引脚。因此，不能设置其参数不同的SIO引脚对。	引脚组件版本2.0允许在引脚上独立设置各参数。但是要注意，一些参数（如输入阈值）仍需要保持一致。在定制器中检查的正常参数值将捕捉这些错误，并强迫您作出适当的修正。
	更新了SIO引脚对的处理。从更早版本的引脚组件迁移时，引脚对可显示为不成对。	如果引脚不成对，请删除该对并添加引脚。
2.0.a	编辑了数据手册。	更新了Reset选项卡的屏幕抓图。 添加了“功能说明”一节的内容。
2.0	添加了PSoC 4000器件的支持。	PSoC 4000器件引脚的布线和同步特性有限制。
	添加了用于PSoC 4单引脚API的文档。	PSoC 4单引脚API与PSoC 3和PSoC 5LP有所不同
	添加了有关时钟驱动引脚的信息，用于PSoC 4器件。	需要特殊配置
1.90.a	更新了驱动模式框图。	更新
	对数据手册进行了少量编辑。	
1.90	添加了PSoC4支持。	PSoC 4使能了新的引脚时钟选项。
1.80	添加了MISRA合规性章节。	该组件没有任何特定偏差。
	添加了有关isr终端连接中断类型的注释。	更新
	添加了有关USBIO驱动模式支持的注释。	更新
	将Input（输入）页面上的Input Synchronized（输入同步）复选框更改为Sync Mode（同步模式）下拉菜单。	下拉菜单可以添加更多模式。
	将Output（输出）页面上的Output Synchronized（输出同步）复选框更改为Output Mode（输出模式）下拉菜单。	下拉菜单可以添加更多模式。
1.70	对数据手册进行了少量编辑及更新	
1.60.a	对数据手册进行了少量编辑及更新	
1.60	添加了外部终端功能	允许引脚连接到片外组件。
	在数据手册中补充了有关PSoC 5加电复位的注解	澄清
	在数据手册中补充了有关PSoC 3 ES2和PSoC 5上P15[7:6]的API可用性	更新

版本	更改说明	更改原因/影响
1.50.a	摘要已经对每4个引脚宏进行了更改。	提高可读性
	向数据手册添加了特性数据	
	优化了数据手册的中断信息	
	在数据手册中补充了有关Vref驱动电平的注解	
	对数据手册进行了少量编辑和更新	
1.50	向API中添加了Keil函数可重入支持。	添加此功能，以便使客户能够指定各个生成函数可重入。
	在Configure（配置）对话框的Reset（复位）选项卡中补充了语句，澄清加电复位适用于整个物理端口。	更新
1.20	“显示为总线”此时给出检查错误，并且引脚组件不同类。同类检查已经得到扩展，以包含HW连接设置。 由于‘显示为总线’检查错误及未选中某些HW连接，因而需要对早期版本和新版本进行适当更改。	

© 赛普拉斯半导体公司，2014。此处，所包含的信息可能会随时更改，恕不另行通知。除赛普拉斯产品内嵌的电路以外，赛普拉斯半导体公司不对任何其他电路的使用承担任何责任。也不根据专利或其他权利以明示或暗示的方式授予任何许可。除非与赛普拉斯签订明确的书面协议，否则赛普拉斯产品不保证能够用于或适用于医疗、生命支持、救生、关键控制或安全应用领域。此外，对于合理预计会发生运行异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯将不批准将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

PSoC®是赛普拉斯半导体公司的注册商标，PSoC Creator™和 Programmable System-on-Chip™是赛普拉斯半导体公司的商标。该处引用的所有其他商标或注册商标归其各自所有者所有。

所有源代码（软件和/或固件）均归赛普拉斯半导体公司（赛普拉斯）所有，并受全球专利法规（美国和美国以外的专利法规）、美国版权法以及国际条约规定的保护和约束。赛普拉斯据此向获许可者授予适用于个人的、非独占性、不可转让的许可，用以复制、使用、修改、创建赛普拉斯源代码的派生作品、编译赛普拉斯源代码和派生作品，并且其目的只能是创建自定义软件和/或固件，以支持获许可者仅将其获得的产品依照适用协议规定的方式与赛普拉斯集成电路配合使用。除上述指定的用途外，未经赛普拉斯的明确书面许可，不得对此类源代码进行任何复制、修改、转换、编译或演示。

免责声明：赛普拉斯不针对该材料提供任何类型的明示或暗示保证，包括（但不仅限于）针对特定用途的适销性和适用性的暗示保证。赛普拉斯保留在不另行通知的情况下对此处所述材料进行更改的权利。赛普拉斯不对此处所述之任何产品或电路的应用或使用承担任何责任。对于合理预计可能发生运转异常和故障，并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

从赛普拉斯或其获得分许可的其中一个联营公司处购买 I^2C 组件，即可根据 Philips I^2C 专利权获得一份许可，以便在 I^2C 系统中使用这些组件，但前提是该系统符合 Philips 定义的 I^2C 标准规范。

产品使用可能受适用的赛普拉斯软件许可协议限制。