AN79953

PSoC® 4 入门

作者：Ronny Liu
相关项目：有
适用器件：PSoC 4100/4200 系列
软件版本：PSoC® Creator™ 2.2 SP1
相关应用文档：点击本链接

关于本应用文档如果您有任何问题或者反馈意见，请发送邮件至：rliu@cypress.com

摘要

AN79953 初步介绍了 PSoC® 4，一个以 ARM® Cortex™-M0 作为核心的单芯片可编程器件。这篇应用文档介绍了芯片架构以及 ARM 芯片是如何与 PSoC 的可编程模拟和数字子系统一起构建应用。本文档还介绍了开发工具以及何处可以获得相关的开发资料。文中还提供了一个示例项目，示范如何使用 PSoC Creator 集成开发环境来进行开发。

目录

简介 1
PSoC 4 特性 2
ARM Cortex-M0 处理器和存储器 2
可定制的数字外设 2
创新的模拟外设 3
低功耗以和模式切换 4
系统资源 5
PSoC 开发工具 6
软件工具 6
硬件开发套件 7
入门项目实践 8
项目简介 8
第一部分 – 建立项目 8
第二部分 – 器件编程 12
总结 13
PSoC 开发者资源 14
入门资源 14
进阶资料 14
高级主题 15
全球销售和设计支持 167

简介

PSoC 是一颗真正的可编程单芯片系统，它集成了可配置的模拟和数字外设，内嵌存储器以及微处理器。PSoC 家族包括有 PSoC 1、PSoC 3、PSoC 4 和 PSoC 5LP 系列产品。

PSoC 内置了处理器，但是它有别于传统的单片机。一个典型的单片机往往包含一个处理器单元（例如 8051 或者 ARM），同时还有一系列的外设，处理器在芯片中扮演着核心的角色，它管理着所有的数据流和时序。离开了这个核心，单片机不再是一个独立的单片机。

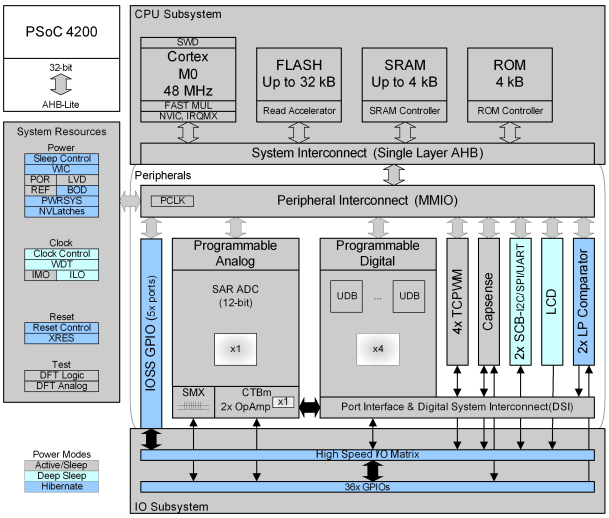
PSoC 则完全不同。如图 1 所示，系统中的处理器，模拟子系统，数字子系统和输入/输出系统都是同等重要，都能相对独立的工作。系统的核心不再是处理器本身，而是各个子系统的互联以及可编程性。换言之，PSoC 是通用单片机的一个超集。你可以通过编程用 PSoC 来模拟通用单片机，但是无法使用一个通用单片机来模拟实现 PSoC 的功能。

这篇应用文档介绍了可编程单芯片系统的概念，尤其是着重介绍了基于 ARM Cortex-M0 的 PSoC 4 产品系列。通过本文档，您可以理解 PSoC 4 的基本情况，以及它可能适用的项目类型。您还可以通过本文档了解强大的图形化集成开发环境 PSoC Creator，这个软件支持 PSoC 家族的 3，4 和 5LP 系列。

PSoC 4 特性

如图 1 所示，PSoC 4 具有完备的功能集合，包括处理器和存储器系统，数字子系统，模拟子系统以及其它系统资源。以下章节简要介绍 PSoC 4 的主要特性，要了解详细信息，请参考 PSoC 4 的器件手册，技术参考手册和其他相关的应用文档。

图 1. PSoC 4200架构



ARM Cortex-M0 处理器和存储器

PSoC 4 系列内置 32 位的 ARM Cortex-M0 处理器，它支持 16 位指令集和 Thumb-2 指令集合的子集，因此它向上兼容更高性能的 Cortex 处理器。Cortex-M0 支持 48M 的最大工作频率，提供最高 43 DMIPS 的处理能力。

PSoC 4 在处理器中纳入了一个单周期的 32 位硬件乘法器，一个唤醒中断控制器，以及嵌套向量中断管理控制器（NVIC），支持 32 个中断源。

PSoC 4 还内置了 4 KB 的静态存储器（SRAM）和 32 KB 的闪存（FLASH）。专门的闪存访问加速模块和处理器紧密合作，使得闪存的平均访问速度达到单周期 SRAM 访问速度的 85%。

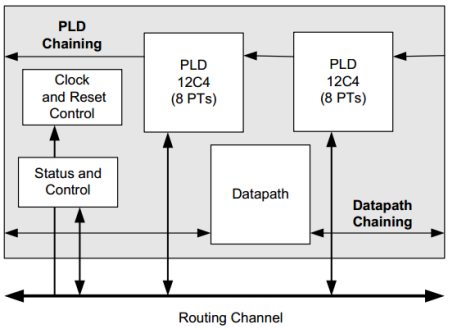
可定制的数字外设

PSoC 4 系列芯片提供了丰富的数字外设，它们包括赛普拉斯独有的通用数字模块（UDB），可复用的串行通信模块，可配置的定时/技数/脉宽调制（TCPWM）模块，等等。所有这些外设都是高度可配置的，用户可以定制这些模块的功能来满足不同的项目需求。

基于 PLD 的 UDB

UDB模块提供了一种更灵活的设计手段。使用UDB，你可以实现多种数字逻辑功能，例如定时器，计数器，脉宽调制，串行通信，以及其他逻辑。这些功能都已用组件的形式实现，在PSoC的软件开发工具中提供给用户。

图 2. 基于 PLD 的 UDB



如图 2 所示，每个 UDB 内有两个“12C4”的 PLD，每个 PLD 有 8 个乘积项，支持 12 个输入，以及 4 个宏单元输出。UDB 中的 PLD 从布线模块或者寄存器获得得输入信号，来控制状态机以及数据通道单元（Datapath），最终输出信号。需要指出的是，用户并不需要精通硬件描述语言来使用 UDB，赛普拉斯公司的软件开发工具能够从你的硬件原理设计直接生成目标代码。

图中 Datapath 包含了一个 8 位的单周期算术逻辑运算组件，它带有一个比较和条件跳转的逻辑电路。Datapath 还包括了一个基于存储器（RAM）的控制单元，用来选择某个指令周期需要执行的操作。Datapath 能被用来实现很多的功能，比如定时器，计数器，PWM，PRS，CRC 校验等等。
除了 PLD 和 Datapath，UDB 里还有时钟、复位控制单元，以及状态和控制寄存器，它们被用来实现处理器和 UDB 之间的通信以及同步。

PSoC 4 共有 4 个 UDB 模块，每个 UDB 都能通过智能布线系统互联，PLD 和 Datapath 能够被串联以实现更高性能的一些功能模块。

可配置的串行通信模块（SCB）

PSoC 4 有两个串行通信模块（SCB），通过简单的配置，同一个硬件接口可以被配置为 SPI、I2C 或者 UART。值得一提的是重配置可以实时发生，例如在一个设计中，SCB 被配置为 UART 来进行固件更新，更新完成后固件可以把该接口重新配置为 I2C，整个过程不需要芯片复位。

PSoC 4 的 SCB 模块支持广泛的串口通信协议，以下为支持的协议列表：

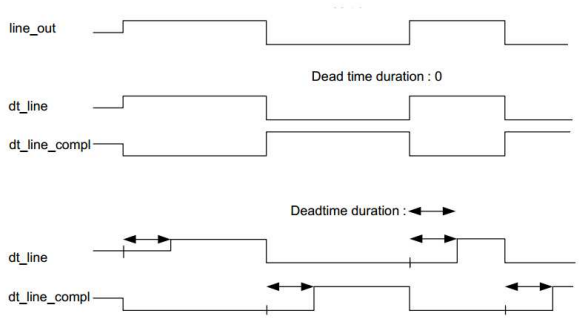
- 可作为标准的 SPI 主器件或从器件，并支持 Motorola、Texas Instruments 和 National Semiconductor 协议。
- 标准的 UART 协议，并支持 SmartCard 读卡器、本地互联网络和 IrDA 协议。
- 可作为标准的 I2C 主器件或从器件。
- 支持 SPI 和 I2C 的 EZ 模式。
- 支持 SPI 和 I2C 的低功耗（深度睡眠）模式（需使用外部时钟）。

参阅技术参考手册以获得详细信息。

可编程的定时/技术/脉宽调制模块（TCPWM）

PSoC 4 有四个定时/计数/脉宽调制模块（TCPWM），每个 TCPWM 使用专用的硬件逻辑实现了一个 16 位的定时器，计数器或者脉宽调制器。每个 TCPWM 提供了一对互补的输出。在死区模式下，通过插入一段死区，可以获得两个互不重叠的 PWM 输出，参见图 3。

图 3. PWM 的死区控制



除了实现通用的定时、计数或者脉宽调制以外，PSoC 4 还提供了捕获寄存器（Capture register）来记录某个时间发生时的计数值，时间寄存器（Period register）用来当计数值达到时间寄存器预设值时，停止或者重新装载计数器，还有比较寄存器（Compare register），用来产生比较数值从而控制 PWM 的占空比。

PSoC 4 还提供了关断（Kill）信号以方便控制。例如在电机控制系统中，Kill 信号可以在过流发生时被用于立刻关闭 FETs，免除了软件的处理的额外开销。

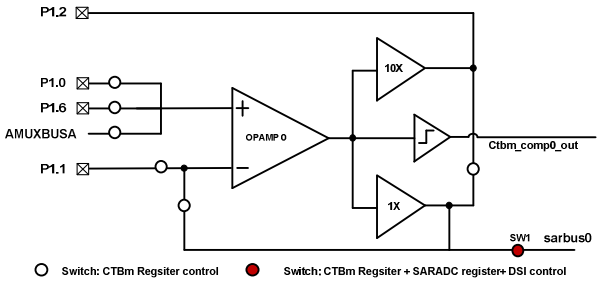
创新的模拟外设

PSoC 4 的模拟子系统包括了精密电流/电压参考电路、连续时间模块（CTBm）、一个高速的 12 位逐次逼近型模数转换（SAR ADC）、触控感应模块（CapSense）、片上温度传感器以及段式液晶控制器。以下对部分模块进行了介绍。

连续时间模块（CTBm）

如图 4 所示，CTBm 包括有一个开关矩阵，两个完全相同的运算放大器（OpAmp）-它可以被配置为比较器或者电压跟随器，每个运算放大器中有一个电荷泵，还有数字信号接口。

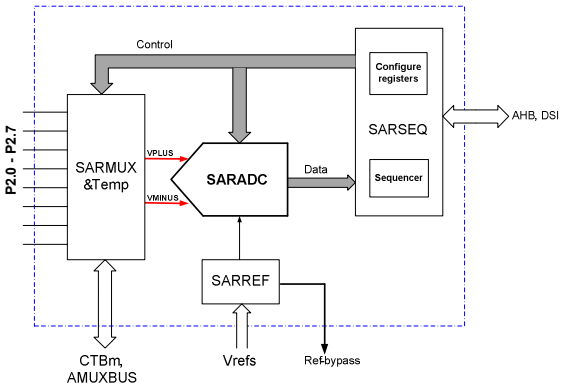
图 4. CTBm 中的运算放大器



带有硬件定序的高精度模数转换（ADC）

PSoC 4 内置一个逐次逼近型模数转换器（SAR ADC），它的核心支持 12 位的分辨率，在工作在 18 MHz 时每秒能采样高达 100 万次。

图 5. SAR ADC 结构图



如图 5 所示，SAR ADC 前端是一个多路选择器（SARMUX），最高支持 8 路外部管脚信号或者内部信号（来自模拟总线、温度传感器、CTBm 等等）。图中 SARREF 模块为 ADC 提供了多种参考电压选择。

定序器 SARSEQ 可用于控制 SARMUX 和 SARADC 以实现按照配置的模式进行扫描。SARSEQ 的存在使得所有的控制可完全不依赖于处理器，它同时还提供了一些简单的预处理功能，比如对输出数据进行平均，超限处理，等等。

PSoC 4 支持总共 8 路可独立配置的输入通道，另外还支持一路额外的插入通道。每个通道都可以独立配置来选择单端/差分模式、分辨率等参数。模数转换结果经过双缓冲结构，可以实现简单的预处理。转换的触发可以来自于多种方式-软件、定时器、外部管脚、或者 UDB。这个特性使得用户可以非常灵活的进行系统开发。

低功耗比较器

PSoC 4 的休眠电源域有两个低功耗比较器，它们能够在除了关闭模式以外的其它所有功耗模式下工作。这个模块也是高度可配置的，你可以选择功耗模式和响应速度、选择使能/关闭迟滞功能以适应不同的噪声环境，也可以定制修正值。

在关注功耗的应用中，当器件进入休眠模式时，低功耗比较器仍然能够工作，它可以被用来监控输入并产生一个输出，作为中断或者唤醒源来唤醒处理器。

触控感应模块（CapSense）

PSoC 4 的 CapSense 模块支持全面的触控感应。这个特性利用用电容感应来实现按键、滑条、触摸板以及接近感应。其基本原理是通过测量手指和感应平面之间的电容大小来获知位置和动作信息。

PSoC 4 的 CapSense 采用了 delta-sigma 运算，实现了大于 5：1 的信噪比（SNR）。另外提供的屏蔽电极，使得接近感应的实现和防水设计更为方便。所有的通用输入输出管脚（GPIO）都可配置为 CapSense 的电极或者屏蔽电极。另外 PSoC 4 还使用了伪随机序列（PRS）时钟源，大大改善了系统的电磁干扰（EMI）。

段式液晶驱动

PSoC 4 的段式液晶驱动器使用了不同的脉宽调制驱动模式（1/2 偏置或 1/3 偏置波形）来产生不同的液晶驱动波形（A 类或者 B 类）。这样的灵活设计兼顾了低功耗模式（例如深度睡眠）下的驱动需求。

PSoC 4 的段式液晶驱动支持最多 4 个公共端（Common），所有的 GPIO 都可以配置为公共端（Common）或者区段端（segment）。

低功耗以及模式切换

PSoC 4 系列芯片支持业界最宽广的供电范围，它允许的工作电压从 1.71v 到 5.5v。这个芯片系列支持高达 5 种功耗模式，其中非保持的关闭模式消耗仅 20nA 的电流。所有的功耗模式以及模式切换可以参考表 1：

模式	描述	唤醒源	唤醒动作
工作	工作模式，所有外设均可用	不适用	不适用
睡眠	处理器进入睡眠，内存（SRAM）数据能够保持，所有外设均工作	任何中断	中断
深度睡眠	所有内部供电来自深度睡眠电压调制器，主时钟和高速外设均关闭。只有低速时钟（32 kHz）仍然可用。来自低速外设、异步或者低功耗的模拟外设能够唤醒系统。	端口中断、低功耗比较器、SCB、看门狗定时器	中断
休眠	只有 SRAM 和 UDB 的数据能保持，多数内部电源均关断，唤醒智能来自端口中断或者低功耗比较器输出	端口中断、低功耗比较器输出	复位
关闭	所有内部供电均关断，只有GPIO状态能保持，唤醒智能来自系统复位或者外部专用唤醒管脚 (P0.7)	系统复位、专用唤醒管脚（P0.7）	复位

系统资源

PSoC 4 系统资源包括一些常用的功能，例如看门狗定时器、复位系统等等。另外还有其他的特别的设计，包括电源系统、输入输出系统、可编程时钟、器件安全等等。

供电与电源监测

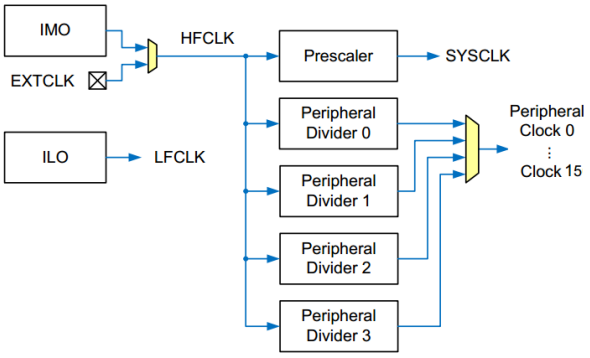
PSoC 4 允许业界最为宽广的供电电压范围（1.71v-5.5v）。为了支持不同的功耗模式，内部还有多种调压器来支持不同的功耗模式，这包括工作模式调压器、精密源调压器、深度睡眠调压器和休眠调压器。
丰富的电源检测手段是 PSoC 4 的一个特别设计。各种电压检测，包括上电复位、各种条件的低掉电检测、低压检测等等。上电复位电路为系统提供复位脉冲信号，掉电检测电路则保护了器件在掉电是免受损害。低压检测则准确的监控外部供电的电压波动。所有以上的检测都会被记录在芯片的寄存器中，使得用户能够进行系统分析。

时钟系统

PSoC 4提供了两种内部时钟，一个是精度为±2%的，3-48 MHz的内部主时钟（IMO），另一个是32 kHz的低速时钟（ILO）。赛普拉斯的PSoC Creator软件提供了ILO校准组件，使用它可以使ILO达到±10%的实时精度。它还允许外接1-48 MHz的外部时钟。

PSoC 4 带有四个外设时钟分频模块，每个内置了三个可串联的 16 位分频器。这样的设计使得时钟系统可以通过编程来满足大多数的应用。时钟系统的架构请参见图 6：

图 6. 时钟系统



器件安全性

PSoC 4 提供了一系列的安全特性来保护开发者的知识产权。其内置的 Flash 的每个行都有一个单独的保护位，这些保护信息保存在 Flash 的管理区块。

为了达到更高的安全级别，器件的调试接口可以被固件禁止-当禁止后，要重新打开调试功能的办法就是清除 Flash 的内容并且重新烧写新的带有使能调试的固件。

另外，所有的器件接口可以被永久禁止，这样可以防止通过恶意编程或者利用中断 Flash 的烧写来进行攻击。注意永久禁止接口会使得芯片不能够进行缺陷分析，因为即使是赛普拉斯公司也不能够再访问这些接口。因此在大多数的应用中并不推荐采用这样的措施。

输入输出系统（I/O）

输入输出系统提供了处理器内核以及周边器件和外界通信的桥梁。PSoC 的灵活性使得信号可以被路由至任意的管脚，这大大方便了电路设计和布线。通过智能路由系统，大多数的数字外设（除了串行通信模块）、LCD、和 CapSense 都能把信号路由至任何一个通用输入输出（GPIO）管脚。其他的模拟外设也可以通过两根模拟总线来控制其信号通路。

每个 GPIO 都能被单独配置为表 1 中 8 种模式中的一种。每个 GPIO 管脚都能被单独的配置为快或慢的输出压摆率（Slew rate），每个 GPIO 管脚都能被配置为 CMOS 兼容或者 LVTTL 兼容模式。所有上述的控制都是独立可编程的。

表 1. GPIO 驱动模式

Fig.	Drive Mode	Data = 1	Data = 0
0	High-Impedance Analog	High Z	High Z
1	High-Impedance Digital	High Z	High Z
2	Resistive Pull-up	Res 1 (5k)	Strong 0
3	Resistive Pull-down	Strong 1	Res 0 (5k)
4	Open Drain, Drives Low	High Z	Strong 0
5	Open Drain, Drives High	Strong 1	High Z
6	Strong Drive	Strong 1	Strong 0
7	Resistive Pull-up and -down	Res 1 (5k)	Res 0 (5k)

PSoC 开发工具

软件工具

Cypress 强大的软件集成开发环境，PSoC Creator, 支持 PSoC 3/4/5LP 系列器件。这个革命性的软件提供了友好的图形界面，代码编辑管理，以及强大的软硬件综合设计能力。你可以从 www.cypress.com/go/creator 免费获得该软件。

以下章节介绍了 PSoC Creator 的基本概念。

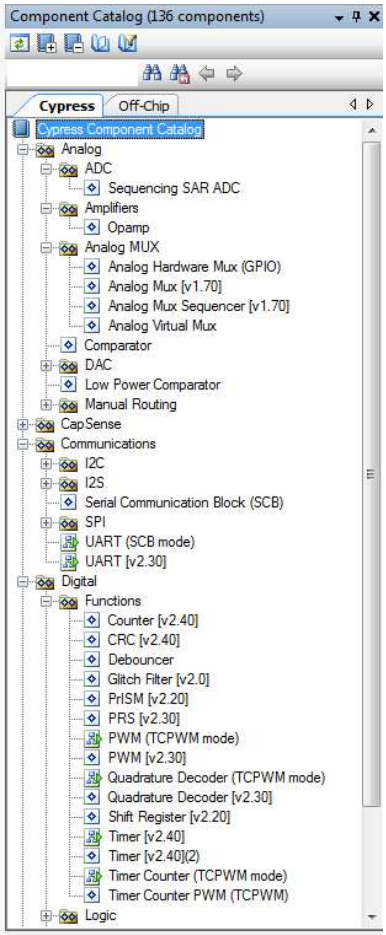
工作空间/项目 – 项目管理了设计的所有资源，包括原理图，资源，源代码，以及生成的代码。工作空间则管理着一个或多个项目。项目总是一个工作空间的一部分，而一个工作空间可以包括一个或多个可以同时操作的项目。

组件/实例 – 组件定义和实现了某个特定的功能，它包括一整套方便开发者使用的资源，如原理图符号，相关代码和函数接口，以及参考文档。

一个常见的组件的例子是定时器/计数器。PSoC Creator 提供了丰富的组件库（参见图 7），方便用户进行快速的硬件和软件开发。用户也可以自己定制组件，便于项目的复用管理。

当用户从组件库中选择应用一个组件时，一个实例就会生成。在一个项目中你可以有同一个组件的多个实例，只要你选择的器件有足够多的硬件资源。

图 7. PSoC Creator 组件库

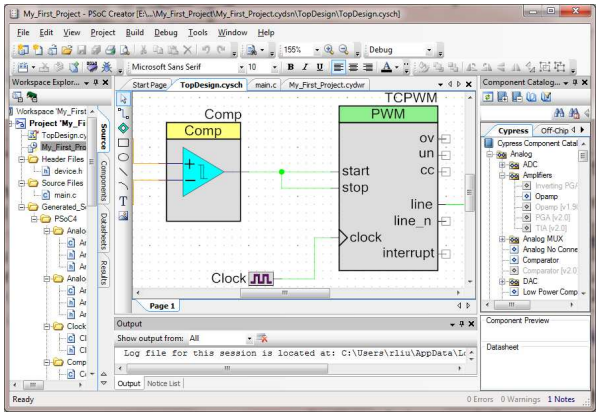


工具集合 – PSoC Creator 工具集合允许你使用抽象的符号来进行设计， 从而使你能专注在高层的架构工作。

文本编辑器，或者叫代码编辑器，提供了源代码查看和编辑的功能。

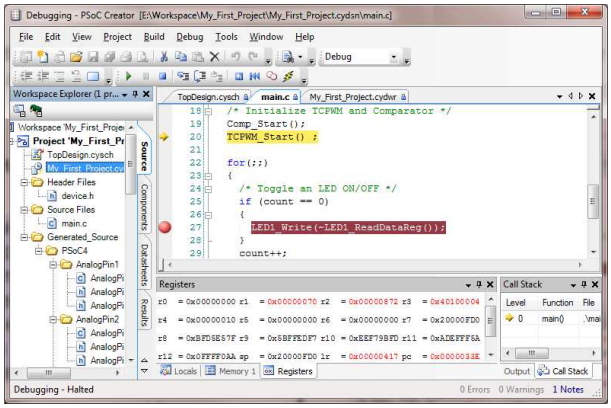
就像和其它厂商的单片机开发一样，你总是可以通过写代码来开发固件。然而，PSoC Creator 还提供了更加方便的图形化开发界面，你可以用电路原理图的方式来实现你的设计并且让 PSoC Creator 自动生成大部分的底层代码。原理图窗口可以方便你创建和修改设计的草图（如图 8）。全局设计资源(DWR)窗口提供了用户界面来管理所有的设计资源如管脚，时钟，中断，DMA 等等。

图 8. 原理图工具



编译/编程/调试 – 当你创建一个项目时，PSoC Creator 选择合适的工具链（代码生成，编译器，连接工具等）来构建项目。PSoC Creator 还能将生成的代码烧写到器件中。PSoC Creator 支持不同的烧写工具，比如 CY8CKIT-002 MiniProg3。当然，PSoC Creator 也完全支持调试功能，参见图 9.

图 9. 使用 PSoC Creator 进行调试



硬件开发套件

Cypress 提供了丰富的开发套件来满足不同的应用。其中最基本的平台为 CY8CKIT-001 开发套件，它可以通过增加处理器模块来支持 PSoC 1、PSoC 3、PSoC 4 和 PSoC 5LP 系列。

PSoC 4 Pioneer 开发板(CY8CKIT-042)是特别支持 PSoC 4200 的，具有非常低的成本的评估套件。

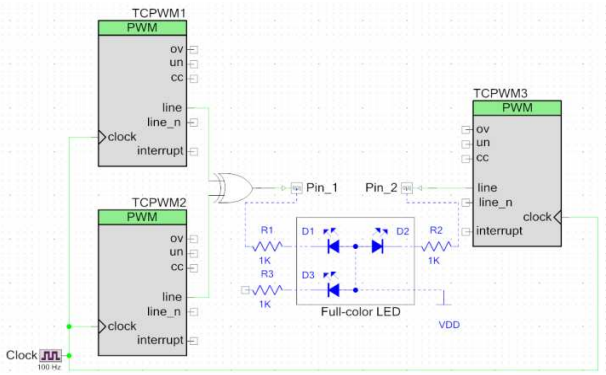
入门项目实践

这个环节将带领读者使用 PSoC Creator 来开发一个实际的项目。通过这个实践，读者可以获得 PSoC Creator 的基本信息，初步了解 PSoC 4，以及系统可编程的基本概念。同时，读者可以了解 PSoC 相比较通用 MCU 的优越性。

项目简介

这个设计简单的驱动一个全彩 LED。如图 10 所示，全彩 LED 本质上由三个独立的红色、绿色和蓝色的单色 LED 组成。根据混色原理，当改变任意一个单色 LED 的驱动电流的时候，全彩 LED 的显示颜色会发生变化。

图 10. PWM 驱动彩色 LED



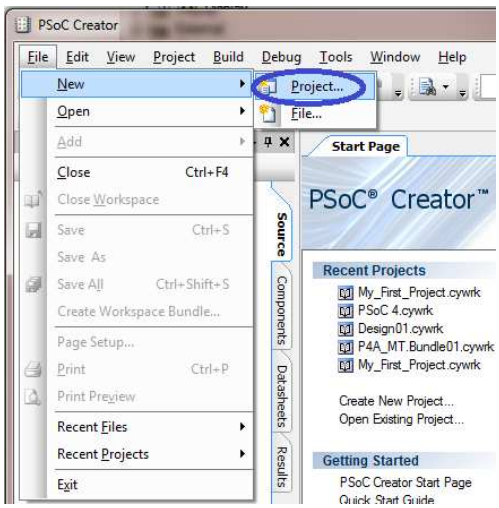
在这个设计中，PSoC 4 用一路 PWM 来驱动全彩 LED 中的一个单色 LED，这个 PWM 有着固定的占空比。另外一路 PWM 驱动第二个单色 LED，这路 PWM 我们设计为占空比可变的，这样，我们可以看到全彩 LED 的颜色是周期性变化的。

第一部分 – 建立项目

这个部分带领你一步一步的遍历项目创建的过程，包括从创建空的项目，到进行硬件和软件设计。您也可以跳过一些简单步骤直接演示项目：

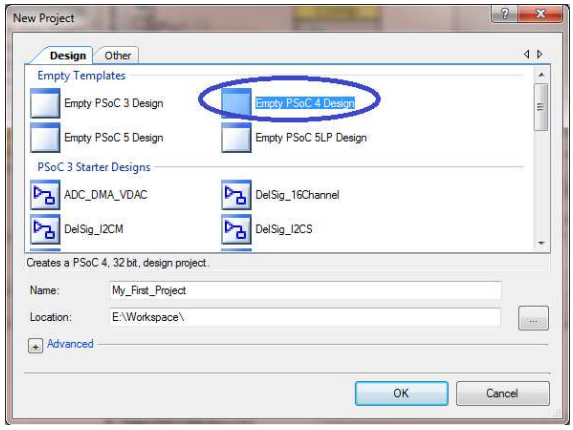
1. 首先，您需要安装 PSoC Creator 2.2 SP1 开发软件。你可以从 www.cypress.com/go/creator 免费获得该软件。
2. 启动 PSoC Creator，从 File 菜单中选择 New→Project，如图 11 所示。

图 11. 选择 File→New→Project



3. 选择空的 PSoC 4 的项目模板，输入项目名称 *My_First_Project*，如图 12 所示。选择您的项目路径，然后点击 **OK**，一个项目文件夹会自动在您指定的路径产生，这个文件夹包括了所有项目文件。

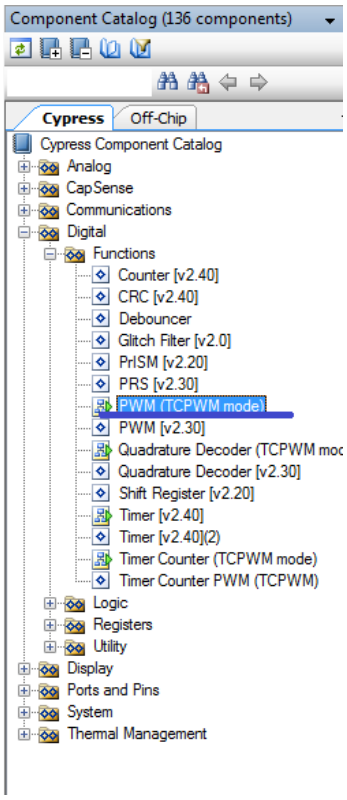
图 12. 创建空的 PSoC 4 项目



4. 在工作空间窗口，打开文件 *TopDesign.cysch*，也就是项目的原理图文件。

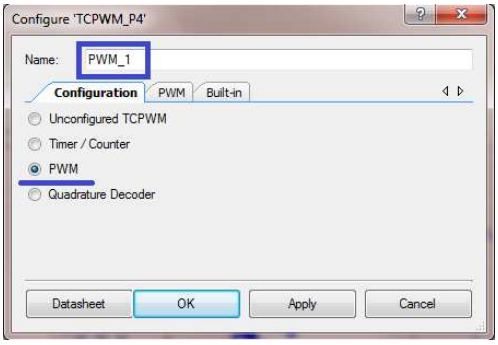
   从组件库中，拖拽一个 PWM 的组件到设计页面上，如图 13 所示。

   图 13. 将组件拖放到原理图中



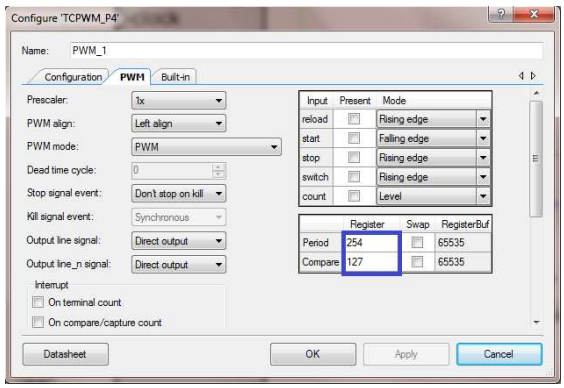
5. 双击原理图上放置的 PWM 组件，在弹出的属性页面进行配置，如图 14 所示。在这个例子中，改变 PWM 组件的名字为 PWM_1，并且选择 PWM 模式。

   图 14. 配置 PWM



继续点击配置窗口的 PWM 页面，更改部分属性如图 15 所示。在本例中，禁止中断和所有输入信号，设置 period 值为 254， compare 值为 127。这将产生一个 50%占空比的 PWM。点击 **Apply** 或者 **OK** 以保存设置。

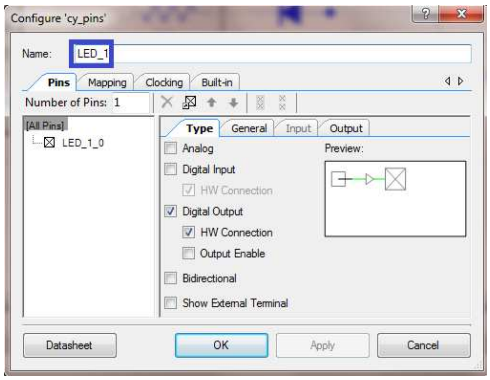
图 15. 配置 PWM_1 属性



6. 现在给 PWM 提供时钟和输出管脚。从组件库中选择时钟（Clock）组件和数字输出管脚（Digital Output Pin）。

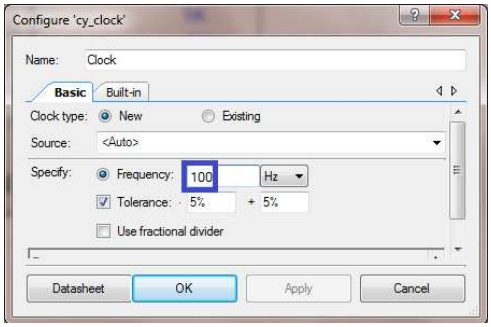
7. 配置数字输出管脚，仅将其名字改为 LED_1，保留其他默认设置。如图 16 所示。

   图 16. 配置数字输出管脚



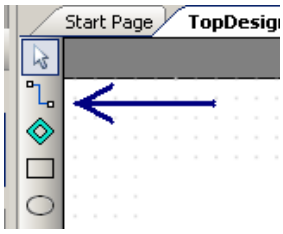
8. 配置时钟频率为 100 赫兹，保留其他默认属性。如图 17 所示。

图 17. 配置时钟组件



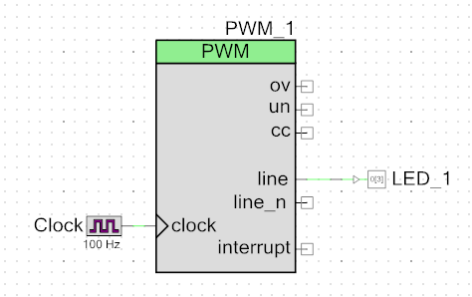
9. 选择连线工具，如图 18 所示。（你也可以使用快捷键'w'）。

图 18. 选择连线工具



将所选用的组件进行连线。如图 19 所示。

图 19. 组件连线

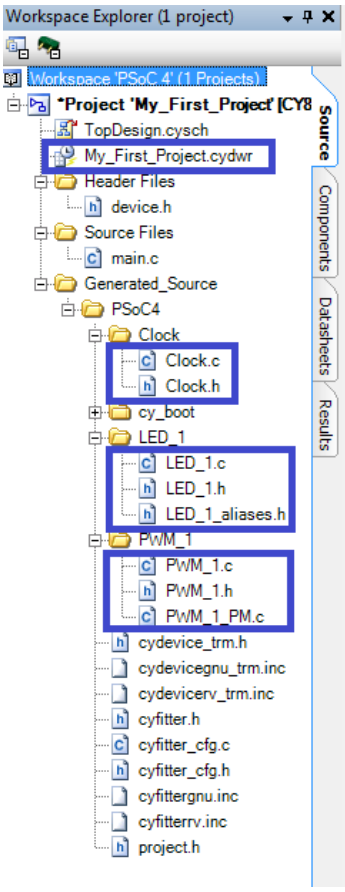


10. 现在我们可以添加一些代码来点亮 LED 了。从 **Build** 菜单选择 Build My_First_Project，这时在工作空间的索引窗口里您会注意到 PSoC Creator 为 PWM、Clock 和数字输出管脚等组件自动生成了一些代码，如图 20 所示。这些文件实现了 PWM、Clock、数字输出管脚等相关的所有底层代码和编程接口。参见组件数据手册获得所有编程接口的信息。

在本例中，仅使用了 PWM 的两个编程接口来初始化 PWM。从工作空间索引中打开 main.c 文件，添加以下代码：

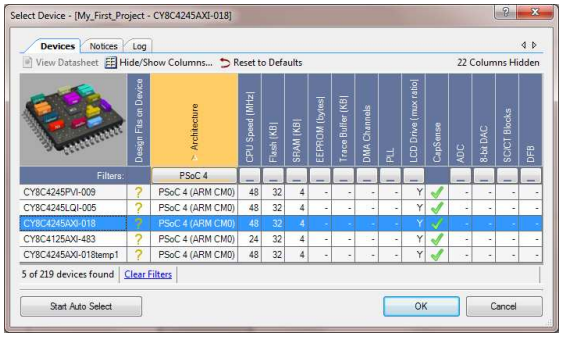
```
void main()
{
  /* Initialize PWM */
  PWM_1_Start ();
  PWM_1_TriggerCommand (
          PWM_1_MASK, PWM_1_CMD_START);

  for (;;)
  {
    /* Place your application code
       here. */
  }
}
```

图 20. 工作空间索引



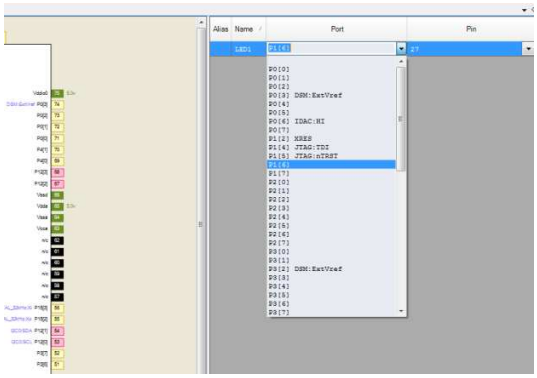
11. 在最终完成这个初始设计之前，我们还需要为设计分配物理管脚。首先选择您所使用的器件-点击菜单 **Project → Device Selector** 并选择合适的器件型号，如图 21 所示。

图 21. 器件选择



参考，从工作空间索引打开 *My_First_Project.cydwr*（全局设计资源文件）-点击管脚页面您将会看到图形化的器件描述。为设计中的数字输出管脚分配所需要的物理管脚（本例中使用 P0[2]），如图 22 所示。

图 22. 分配物理管脚



12. 现在你可以最终编译项目。点击菜单 **Build → Generate Application**。如果没有编译错误的话 PSoC Creator 会产生相关的项目文件。您可以现在就在硬件平台上验证这个设计。

或者，继续下一步来对第二个单色 LED 进行控制。

13. 为了产生一个占空比持续改变的 PWM，我们使用两路 PWM，他们都是 50%的占空比，但是他们的 period 值有一点差异，这样如果使他们经过一个异或门，异或门将输出一个占空比变化的 PWM。

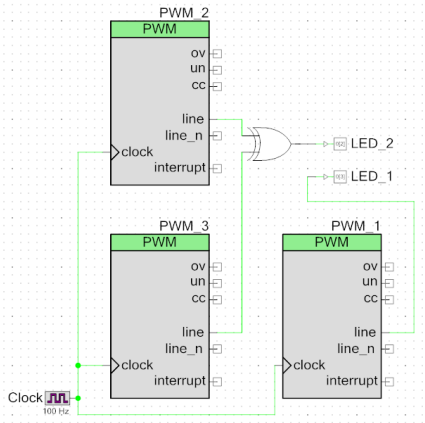
在原理图上放置另外两个 PWM 组件、一个异或门逻辑组件和另外一个数字输出管脚。

14. 改变新增的数字输出管脚的名字为 LED_2，保留其他默认设置。

15. 改变两个新增的 PWM 的名字为 PWM_2 和 PWM_3。保持着两个组件的其它配置与步骤 5 完全一致。唯一的改动是将 PWM_3 的 period 值配置为 256，compare 值配置为 128。

16. 如图 23 所示连接各个组件。

图 23. 最终原理图



17. 打开 *My_First_Project.cydwr* 选择管脚页面，为原理图中的管脚分配物理管脚，如所示。

图 24. 分配物理管脚

Alias	Name	Port	Pin	Lock
	LED_1	P0[2] SCB0:SPI:SS3	26	☑
	LED_2	P1[6]	43	☑

18. 在 *main.c* 中增加代码，初始化 新增的 PWM 组件。最终代码如下:

```
void main()
{
  /* Initialize PWMs */
  PWM_1_Start ();
  PWM_1_TriggerCommand (
          PWM_1_MASK, PWM_1_CMD_START);
  PWM_2_Start ();
  PWM_2_TriggerCommand (
          PWM_2_MASK, PWM_2_CMD_START);
  PWM_3_Start ();
  PWM_3_TriggerCommand (
          PWM_3_MASK, PWM_3_CMD_START);

  for(;;)
  {
    /* Place your application code
       here. */
  }
}
```

19. 最后，如步骤 8 所述，重新编译项目。如果您有赛普拉斯 Pioneer（CY8CKIT-042）开发板的话，您可以直接在开发板上运行生成的代码，不需要任何改动。

第二部分 – 器件编程

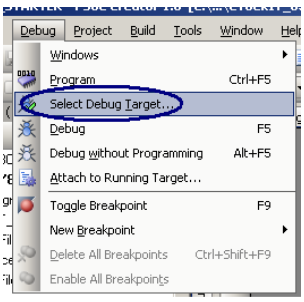
这个部分介绍了怎样使用 PSoC Programmer 进行编程，PSoC Creator 中也集成了菜单来调用该软件。

如果您在开发自己的硬件，以下信息会帮助你了解如何使用硬件调试器/编程器，例如赛普拉斯的 MiniProg3。

如果您在评估赛普拉斯 Pioneer 开发板（CY8CKIT-042），您只需要连接 USB 线，而不需要额外的调试编程器件。

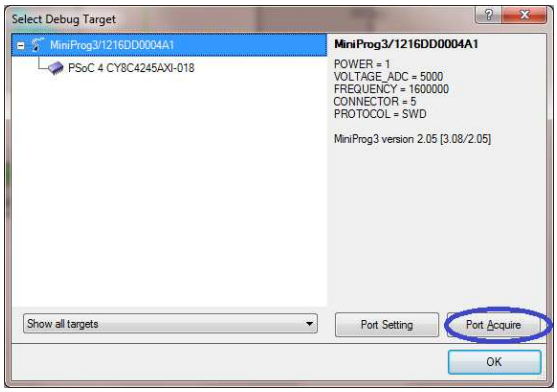
1. 从 PSoC Creator 的 **Debug** 菜单中，选择 **Select Debug Target**，如图 25 所示。

   图 25. 选择调试目标



2. 在弹出的 **Select Debug Target** 对话框中，点击 **Port Acquire**，如所示。点击 **OK** 关闭对话框。

   图 26. 设定调试目标 Connect the Pins



3. 如图 27 所示，从 **Debug** 菜单，选择 **Program** 对目标器件进行编程，这里使用的是第一部分教程中产生的目标文件。编程状态会显示在 PSoC Creator 窗口的左下角的状态栏中，如图 28 所示。

   图 27. 对目标器件进行编程

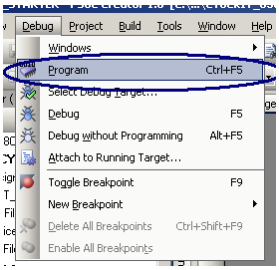
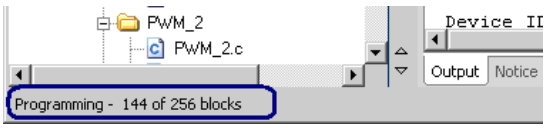


   图 28. 编程状态



4. 当编程结束，开发板上的 PSoC 会自动复位，处理器会开始运行代码。

总结

这篇应用文档概述了 PSoC 4 的基本情况和它的开发环境。从这篇文档你可以感受到 PSoC 和通用单片机的不同之处。使用 PSoC 您可以突破传统单片机的限制，着眼在单个器件上进行完整系统的开发。

要了解关于 PSoC 4 的更多信息，请参考附录中的 PSoC 开发者资源章节。

.

相关应用文档

AN75320 - PSoC 1 入门

AN54181 - PSoC 3 入门

AN77759 - PSoC 5LP 入门

AN85951 - PSoC 4 CAPSENSE 设计指南

关于作者

姓名:	Ronny Liu
职位:	应用工程师
背景:	电子工程硕士
电邮:	rliu@cypress.com

附录

PSoC 开发者资源

开发者可以获取的资源包括数据手册，参考手册，视频，应用文档，等等。

入门资源

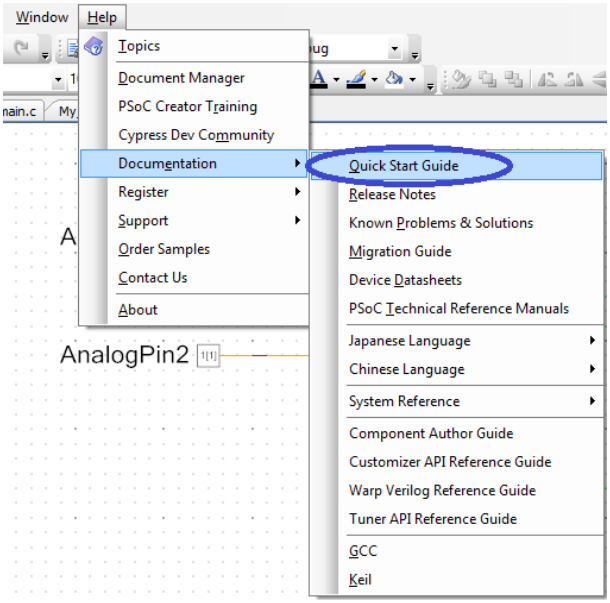
除了本应用文档以外，表 2 还列出了其他的一些资源以方便入门者了解 PSoC 4。

表 2. 入门者资源

资源	何处获得
PSoC 4 主页	www.cypress.com→ Products → Programmable System-on-Chip → PSoC® 4
快速入门指南	PSoC Creator 菜单： Help → Documentation → Quick Start Guide
视频介绍	www.cypress.com→ Support → Videos & Technical Training

快速入门指南 – PSoC Creator 包含了大量的文档和示例，以及入门指南，请参考图 29。

图 29. 快速入门指南



进阶资料

如果你想进一步了解 PSoC 4，或者准备开始进行项目设计，表 3 列出了更多的详细资料。

表 3. 进阶资源

资源	何处获得
器件数据手册	www.cypress.com→ Support → Documentation → Datasheets
器件勘误表	www.cypress.com→ Support → Documentation → Errata
PSoC Creator 帮组文档	PSoC Creator 菜单： Help → Topics
PSoC Creator 组件数据手册	www.cypress.com→ Support → Documentation → Component Datasheets
PSoC Creator 系统参考手册	PSoC Creator 菜单： Help → Documentation → System Reference Guide
应用文档	www.cypress.com→ Support → Application Notes → Programmable System-on-Chip
知识库	www.cypress.com→ Support → Knowledge Base → Programmable System-on-Chip
开发板	www.cypress.com→ Support → Development Kits/Boards → Programmable System-on-Chip
培训	www.cypress.com/go/training

器件数据手册 – 器件数据手册提供了器件功能的简介、管脚描述、器件的性能指标和固定外设的电气特性。

勘误表 – 提供了数据手册数据的勘误信息。

PSoC Creator 帮助主题 – PSoC Creator 软件集成的帮助菜单能方便用户实时的获得帮助。

组件数据手册 – Cypress 提供的每个组件都会带有数据手册，它提供了使用这个组件所需要的所有信息，比如功能描述，接口函数介绍，电气特性等等。当你在 PSoC Creator 中打开组件的配置窗口，你会看到数据手册的链接。你也可以在赛普拉斯网站获得这些资料。

PSoC Creator 系统指南 – 详细描述了赛普拉斯器件提供的功能，以方便用户获得相关信息，如时钟，存储，低功耗模式等等。这些功能不属于任何组件但是会被组件所使用。

基本培训 – 赛普拉斯网站提供了一系列的文档，示例代码，视频等等来帮助用户理解 PSoC 器件。

高级主题

如表 4 所示，对于有经验的开发者，有很多深入的材料来帮助他们进行开发。

表 4. 资深开发者资源

资源	何处获得
器件技术参考手册	www.cypress.com→ Support → Documentation → Technical Reference Manuals
PSoC Creator 组件作者指南	PSoC Creator 菜单： *Help → Documentation → Component Author Guide*

技术参考手册 – 详述了所有的周边资源，包括寄存器级别的描述。该文档分为两个部分，架构参考手册和寄存器参考手册。

组件开发者指南 – 为组件开发者提供了指南。

修正记录

文档标题： PSoC® 4 入门 – AN79953

文档编号：001-87170

版本	ECN 编号	发起者	提交日期	变动描述
**	3974382	RLIU	4/20/2013	首次翻译自 AN79953 *A

全球销售和设计支持

赛普拉斯公司拥有一个由办事处、解决方案中心、工厂和经销商组成的全球性网络。要找到距您最近的办事处，请访问赛普拉斯所在地。

产品

汽车	cypress.com/go/automotive
时钟与缓冲器	cypress.com/go/clocks
接口	cypress.com/go/interface
照明和电源控制	cypress.com/go/powerpsoc cypress.com/go/plc
存储器	cypress.com/go/memory
PSoC	cypress.com/go/psoc
触摸传感	cypress.com/go/touch
USB 控制器	cypress.com/go/usb
无线/射频	cypress.com/go/wireless

PSoC®解决方案

psoc.cypress.com/solutions

PSoC 1 |PSoC 3 | PSoC 4|PSoC 5LP

Cypress 开发者社区

Community | Forums | Blogs | Video | Training

技术支持

cypress.com/go/support

PSoC 是赛普拉斯半导体公司的注册商标。此处引用的所有其他商标或注册商标归其各自所有者所有。

赛普拉斯半导体
198 Champion Court
San Jose, CA 95134-1709

电话 ：408-943-2600
传真 ：408-943-4730
网站 ：www.cypress.com

© 赛普拉斯半导体公司，2013。此处所包含的信息可能会随时更改，恕不另行通知。除赛普拉斯产品的内嵌电路之外，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中，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。

该源代码（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。除上述指定的用途之外，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：赛普拉斯不针对此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。赛普拉斯保留在不做出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中，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。

产品使用可能受适用的赛普拉斯软件许可协议限制。