AN91184

PSoC® 4 BLE — 设计 BLE 应用

作者： Pushek Madaan

相关项目：有

相关器件系列：CY8C4XX7-BL、CY8C4XX8-BL、CYBL10X6X、CYBL10X7X

软件版本：PSoC Creator™ 3.2

相关应用笔记： 请点击此处

AN91184 介绍了如何使用标准配置文件（由 PSoC Creator 的 BLE 组件中所包含的蓝牙 SIG 定义）在 PSoC 4 BLE 基础上设计一款 Bluetooth® Low Energy（低功耗蓝牙 — BLE）应用。它展示了如何使用 CY8CKIT-042-BLE 套件上的 BLE 健康温度计配置文件来构建一个应用。本应用笔记也适用于 PRoC BLE 器件。

目录

1 简介 1
2 PSoC 资源 2
2.1 PSoC Creator 2
2.2 PSoC Creator 帮助 3
2.3 示例代码 3
2.4 技术支持 4
3 标准服务与自定义服务 5
3.1 BLE 健康温度计 5
4 PSoC Creator 项目：健康温度计 6
4.1 配置组件 7
4.2 配置固件 20
4.3 硬件配置 24
4.4 构建并编程器件 26
5 应用测试 27
5.1 CySmart 中心仿真工具 27
5.2 CySmart 手机应用 31
5.3 总结 33
6 相关应用笔记 33
全球销售和设计支持 35

1 简介

低功耗蓝牙（BLE）是由蓝牙技术联盟（SIG）介绍的一个超低功耗无线标准，使用于短距离通信。对 BLE 物理层、协议栈和配置文件架构进行的设计和优化，都是为最大限度降低功耗。类似于传统蓝牙，BLE 的工作频段为 2.4 GHz ISM，但其带宽较较低（1 Mbps）。

赛普拉斯的 PSoC 4 BLE 是可编程嵌入式片上系统（SoC），在单芯片中集成了 BLE、可编程模拟和数字外设功能、存储器以及 ARM® Cortex®-M0 微控制器。

本应用笔记介绍了如何使用 PSoC Creator BLE 组件来设计一款 BLE 健康温度计应用（该应用使用健康温度计标准配置文件），然后使用 CySmart 中心仿真工具和 CySmart 手机应用来验证该应用。由于 PSoC Creator BLE 组件具有预建标准配置文件，因此它易于使用支持 BLE 的项目中的这些器件。

该应用笔记是针对基本了解 BLE、PSoC、PSoC Creator IDE 以及使用热敏电阻测量温度方法的客户。请参考以下各链接：

- AN91267 — PSoC 4 BLE 入门
- PSoC Creator 主页
- AN66477 — PSoC® 3、PSoC 4 和 PSoC 5LP — 使用热敏电阻测量温度

2 PSoC 资源

在赛普拉斯网站 www.cypress.com 上提供了大量资料，有助于选择符合您设计的 PSoC 器件，并能够快速有效地将该器件集成到您的设计中。有关使用资源的完整列表，请参考 KBA86521 — 如何使用 PSoC 3、PSoC 4 和 PSoC 5LP 进行设计。

下面提供了 PSoC 4 BLE 的简要列表：

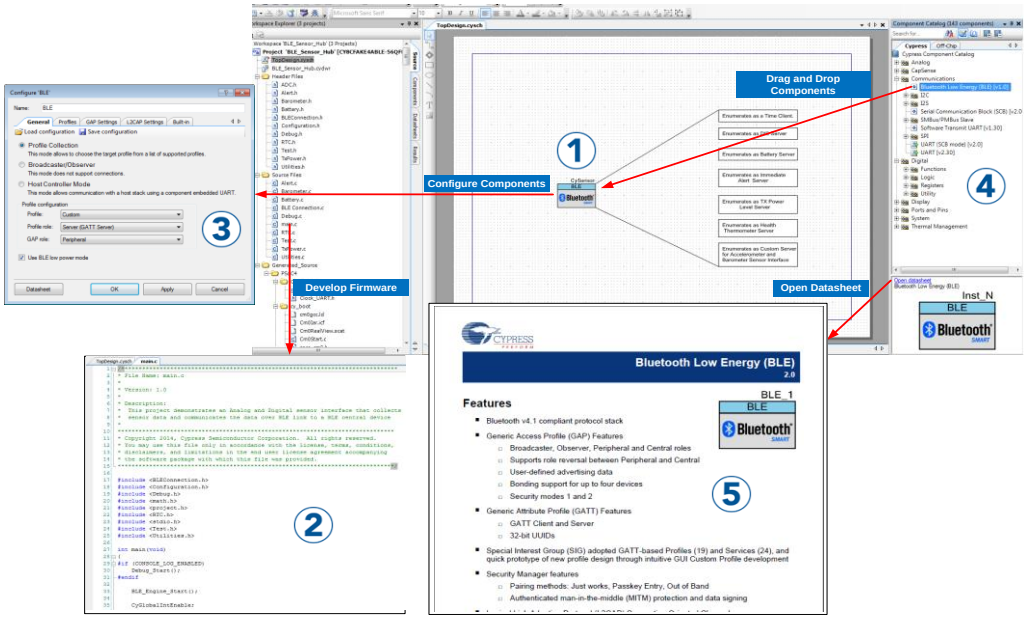
- **概况：**PSoC 产品系列、PSoC 产品路线图
- **产品选择器：**PSoC 1、PSoC 3、PSoC 4 或 PSoC 5LP。此外，PSoC Creator 还包含了一个器件选择工具。
- **数据手册**描述并提供了适用于 PSoC 41XX-BL 和 PSoC 42XX-BL 器件系列的电气规格。
- **应用笔记和代码示例：**包括从基本到高级的广泛主题。许多应用笔记还提供了代码示例。PSoC Creator 提供了其他代码示例 — 请参考示例代码一节。
- **技术参考手册（TRM）**详细说明了每个 PSoC 4 BLE 器件系列中的架构和寄存器。
- **CapSense 设计指南：**了解如何在 PSoC 4 BLE 器件系列中设计电容式触摸感应应用。
- **开发工具**
  - CY8CKIT-042-BLE 低功耗蓝牙（BLE）套件包括 Arduino™兼容屏蔽和 Digilent® Pmod™子卡的连接器。
  - 适用于 Windows、iOS 和 Android 版本的 CySmart BLE 主机仿真工具是一种易于使用的 GUI，通过它您可以进行测试和调试 BLE 外设应用。

2.1 PSoC Creator

PSoC Creator 是一个基于 Windows 的免费集成开发环境（IDE）。通过它可以同时在 PSoC 4 BLE 和 PRoC BLE 的系统中设计硬件和固件。如图 1 所示，通过 PSoC Creator，您可以进行下面的操作：

1. 将组件图标拖放到主设计工作区中，以进行您的硬件系统设计。
2. 对您的应用固件和 PSoC 硬件进行协同设计。
3. 使用配置工具配置各组件。
4. 研究包含 100 多个组件的库。
5. 查看组件数据手册。

图 1. PSoC Creator 原理图输入项和组件



2.2 PSoC Creator 帮助

请访问 PSoC Creator 主页以下载并安装 PSoC Creator 的最新版本。然后，启动 PSoC Creator 并将它导航到下列各项：

- **快速入门指南**：依次选择 **Help** > **Documentation** > **Quick Start Guide**。本指南提供了开发 PSoC Creator 项目的基本知识。
- **简单组件的示例项目**：请依次选择 **File** > **Open** > **Example projects**。这些示例项目展示了如何配置及使用 PSoC Creator 组件。
- **入门设计**：依次选择 **File > New** > **Project** >**PSoC 4 Starter Designs**。这些入门设计展示了 PSoC 4 BLE 的特有性能。
- **系统参考指南**：请依次选择 **Help** > **System Reference** > **System Reference Guide**。该指南列出并描述了 PSoC Creator 提供的系统功能。
- **组件数据手册**：右键单击组件，然后选择“Open Datasheet”项。请访问 PSoC 4 BLE 组件的数据手册网页，获取所有 PSoC 4 BLE 组件的数据手册列表。
- **文档管理器**：PSoC Creator 提供了一款文档管理工具，便于寻找和查看文件资源。 要想打开文档管理工具，请选择菜单项：**Help** >**Document Manager**。

2.3 示例代码

PSoC Creator 包含了多个代码示例项目。可以从 PSoC Creator 的“Start Page”（起始页）上获取这些项目，如图 3 所示。

各个示例项目通过为您提供完整的设计（并非一个空白页），可以加快您的设计过程。示例项目还介绍了如何将 PSoC Creator 组件使用于不同应用中。此外，它还包含了多个代码示例和数据手册，如图 3 所示。

在图 3 显示的 **Find Example Project**（查找示例项目）对话框中，您可以选择以下选项：

- 根据 architecture（架构）或 device family（器件系列）（例如：PSoC 4、PSoC 4 BLE、PRoC BLE 等）；category（类型）或 keyword（关键词）等选项筛选示例。
- 从 **Filter Options**（滤波选项）的示例菜单中进行选择。您可以从拥有 20 多个 BLE 示例项目的库开始，如图 3 所示。
- 通过 **Documentation**（文档）选项卡，查看选中的数据手册。
- 查看选中的代码示例。您可以复制该窗口中的代码然后将其粘贴在您的项目内，从而加快代码开发过程。
- 或根据已选项目创建一个新的项目（若需要可添加新的工作区）。通过向您提供一个完整的基本设计，它可以加快您的设计进程。然后，您可以根据自己的应用来调整该设计。

图 2. PSoC Creator 中的代码示例

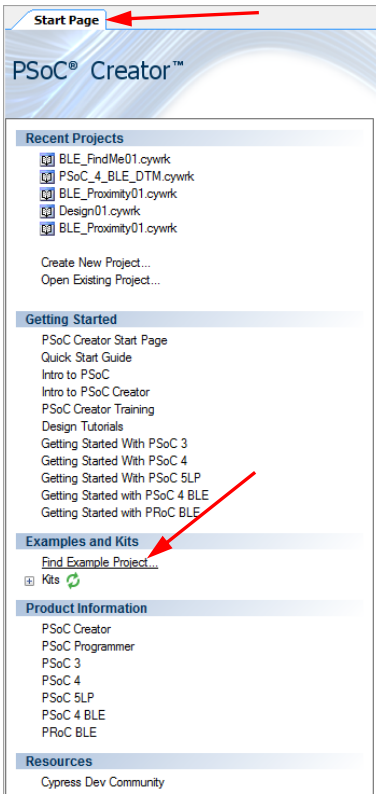
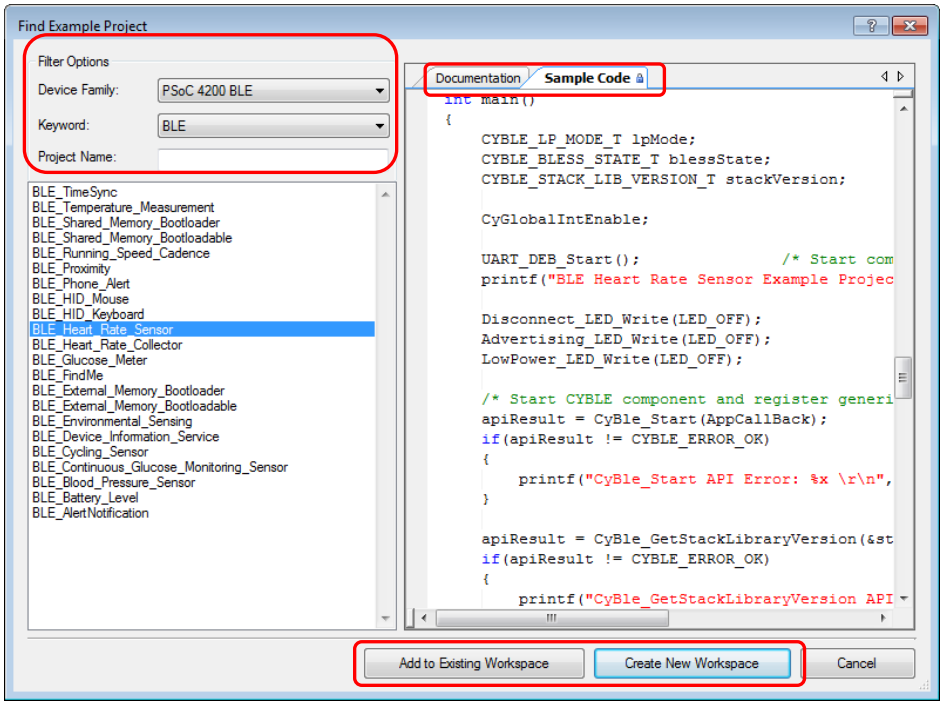


图 3. 带样本代码的代码示例项目



2.4 技术支持

- 常见问题（FAQ）：加深了解我们的 BLE 套件
- BLE 论坛：查看您在 PSoC 4 BLE 和 PRoC BLE 论坛上咨询的内容是否得到了各开发伙伴方的回复。
- 赛普拉斯支持：仍未解决？请访问我们的支持页面，并创建一个技术支持情况或联系本地销售人员代表。如果您在美国，可以通过拨打我们的免费电话，直接与技术支持团队联系：+1-800-541-4736。在提示符处选择第 8 项。

3 标准服务与自定义服务

蓝牙 SIG 定义了一组服务，这些服务可被配置为 GATT 客户端或 GATT 服务器。这些服务被称为标准服务。标准服务示例包括：心率服务、健康温度计服务、血压服务以及即时警报服务。请参考蓝牙开发者网站，获取标准服务的完整列表。

定义了这些标准服务，以便满足众多应用的要求。例如，可以配置心率服务以报告从腕带或胸带监视仪中心率传感器中获得的数据。它还显示了在指定时间间隔内所消耗的能量。

通过 BLE 标准您还能够创建自己的服务，又称为自定义服务。顾名思义，这些服务用于定义不属于 BLE 标准服务的内容。这些服务与 BLE 标准服务同样重要，因为通过它们您能够展开具有自定义应用的 BLE 器件。

3.1 BLE 健康温度计

在 BLE 健康温度计应用中（图 4），温度计设备作为 GAP 外设工作并实现了健康温度计传感器配置文件，接收数据的移动设备则作为 GAP 中心工作并实现了健康温度计收集器配置文件。在该示例中，健康温度计传感器配置文件实现了两个标准服务：健康温度计服务和器件信息服务。健康温度计服务包含温度测量、温度类型和测量间隔三项功能。器件信息服务包含九项功能，具体情况在本文档后面的内容中进行介绍。

图 4. BLE 系统设计



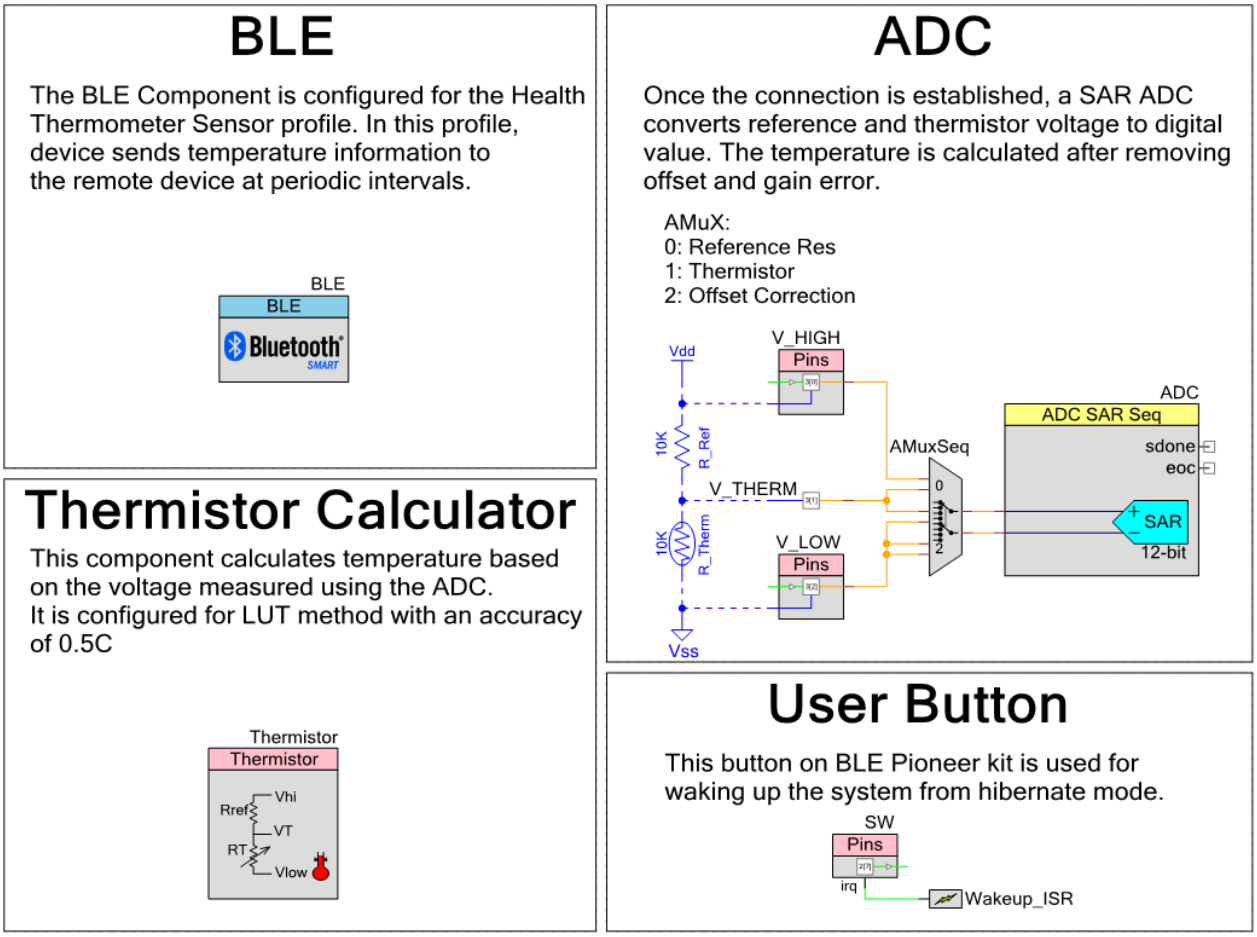
4 PSoC Creator 项目：健康温度计

在该项目中，PSoC 4 BLE 器件集成了下列各项：

- 一个 BLE 组件，在 GAP 层中它作为外设工作，在 GATT 层中它作为 GATT 服务器工作。
- 一个模数转换器（ADC），用于测量热敏电阻上的电压。
- 一个热敏电阻计算器，通过 ADC 所测量的电压计算温度。
- 一个用户按键，用于使系统从休眠模式唤醒。

图 5 显示的是健康温度计项目的 PSoC Creator 原理图。

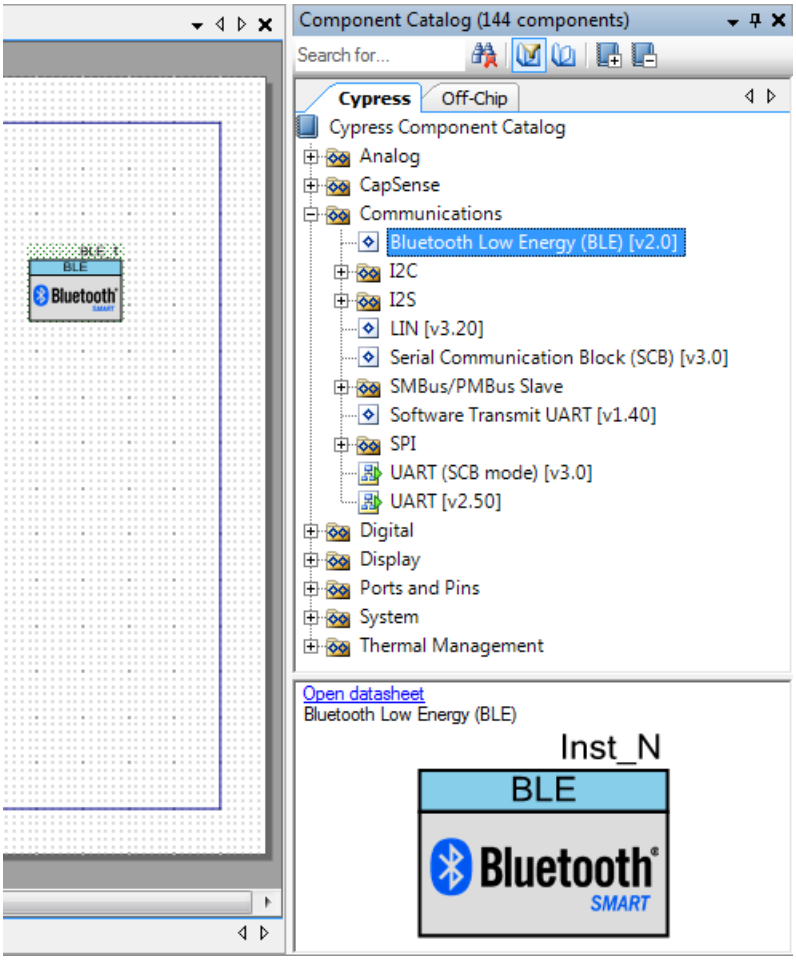
图 5．PSoC Creator 原理图



4.1 配置组件

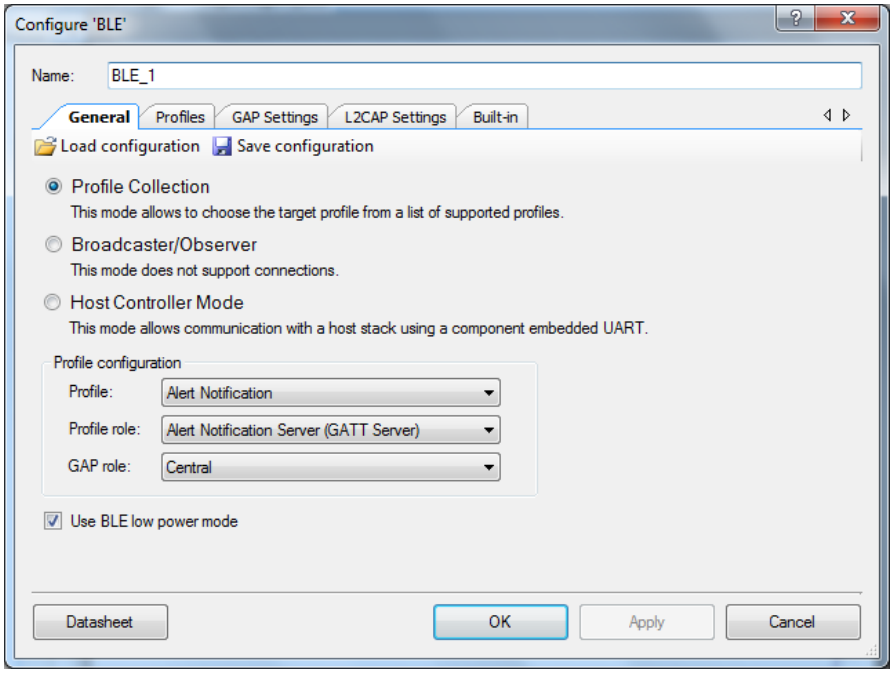
1. 创建一个新的 PSoC 4100/PSoC 4200BLE 设计项目。如果您尚未了解 PSoC Creator，请参考 PSoC Creator 主页中的内容。
2. 将一个 BLE 组件（Component Catalog > Communications）拖放到顶级设计原理图（请参考图 6）内。

图 6. BLE 组件



3. 双击 BLE 组件以配置它。将弹出配置窗口，如图 7 所示。

图 7. BLE 组件 — 配置窗口



4. 在组件配置窗口中的 **General** 选项卡（请参考图 8）上，配置以下各项设置：

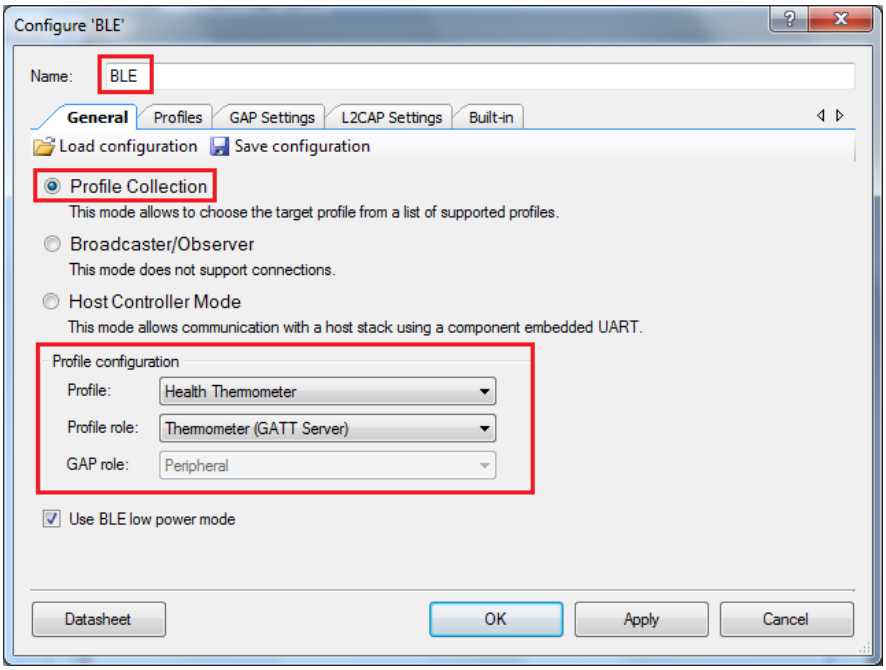
   名称：BLE

   配置：配置文件集

   配置文件：健康温度计

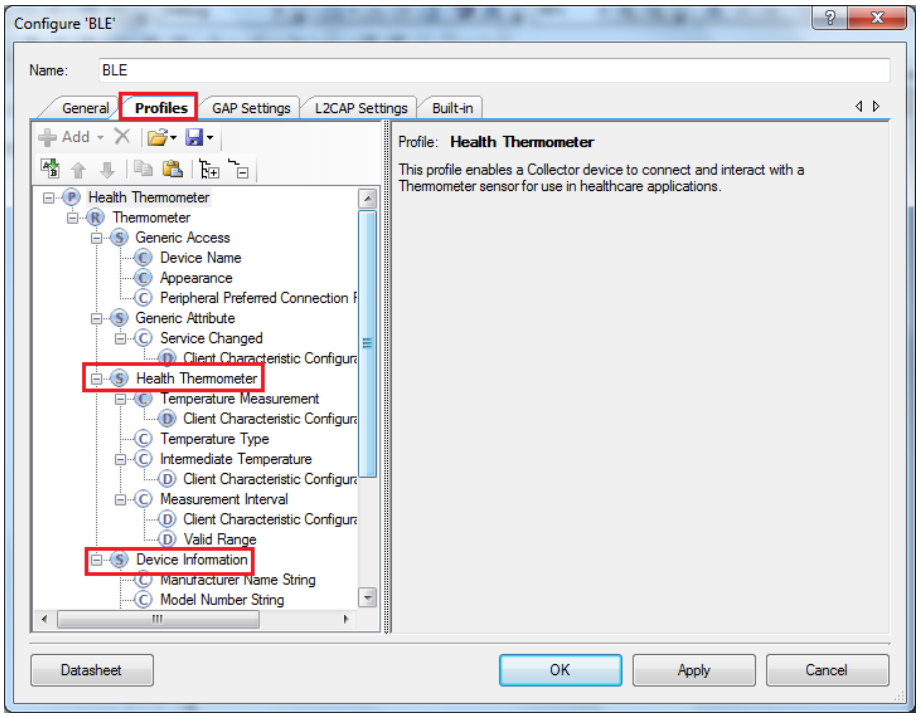
   配置文件作用：温度计（GATT 服务器）

图 8. General 选项卡



注意： 对于蓝牙 SIG，健康温度计标准配置文件包含了健康温度计服务和器件信息服务。因此，默认添加了这些服务，如图 9 所示。请参考蓝牙采用的规范，深入了解健康温度计配置文件或健康温度计服务。

图 9. 健康温度计配置文件



5. 通过以下各项设置配置 **Profiles** 选项卡：

   服务：健康温度计

   特性：温度测量

   字段：温度单位标识

   数值：温度测量值（单位为摄氏度）

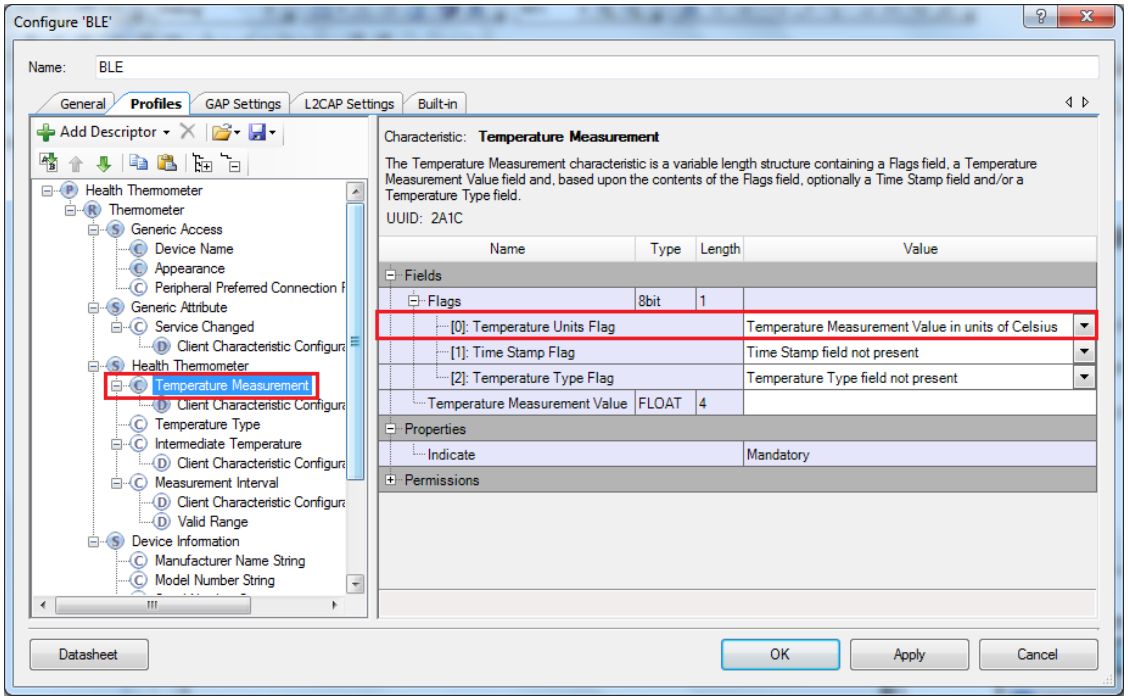
   特性：温度类型

   字段：温度文本介绍

   值： body（通用）

   图 10 显示的是温度测量特性配置。

图 10. 温度测量特性



同样，根据表 1 更新其他服务和特性。描述符列中的“N/A”表示与特性相应的字段和数值，并非表示描述符。例如，字段 **Temperature Text Description**（温度文本介绍）属于 **Health Thermometer**（健康温度计）服务中的特性 **Temperature Type**（温度类型），但字段 **Lower Inclusive Value**（较低的包容值）和 **Upper Inclusive Value**（较高的包容值）则属于 **Measurement Interval**（测量间隔）特性的 **Valid Range** （有效范围）描述符。

表 1. 特性配置

服务	特性	描述符	字段	值	备注
健康温度计	温度测量	N/A	温度单位标识	温度测量值（单位为摄氏度）	可能值由蓝牙 SIG 定义。BLE 组件提供了选择其中一个可能值的选项。
	温度类型	N/A	温度文本介绍	值：body（通用）	可能值由蓝牙 SIG 定义。BLE 组件提供了选择其中一个可能值的选项。
	测量间隔	N/A	测量间隔	1	用户定义 单位为秒 取值范围：1 - 65535
	测量间隔	有效范围	较低的包容值	1	用户定义 单位为秒 取值范围：1 - 65535
			较高的包容值	60	用户定义 单位为秒 取值范围：1 - 65535
器件信息	制造商名称字符串	N/A	制造商名称	赛普拉斯半导体公司	用户定义
	型号字符串	N/A	型号	1.0	用户定义

服务	特性	描述符	字段	值	备注
	序列号字符串	N/A	序列号	**	用户定义
	硬件版本字符串	N/A	硬件版本	CY8CKIT-042-BLE	用户定义
	固件版本字符串	N/A	固件版本	**	用户定义
	软件版本字符串	N/A	软件版本	PSoC Creator 3.2	用户定义

保留其他设置为其默认值。

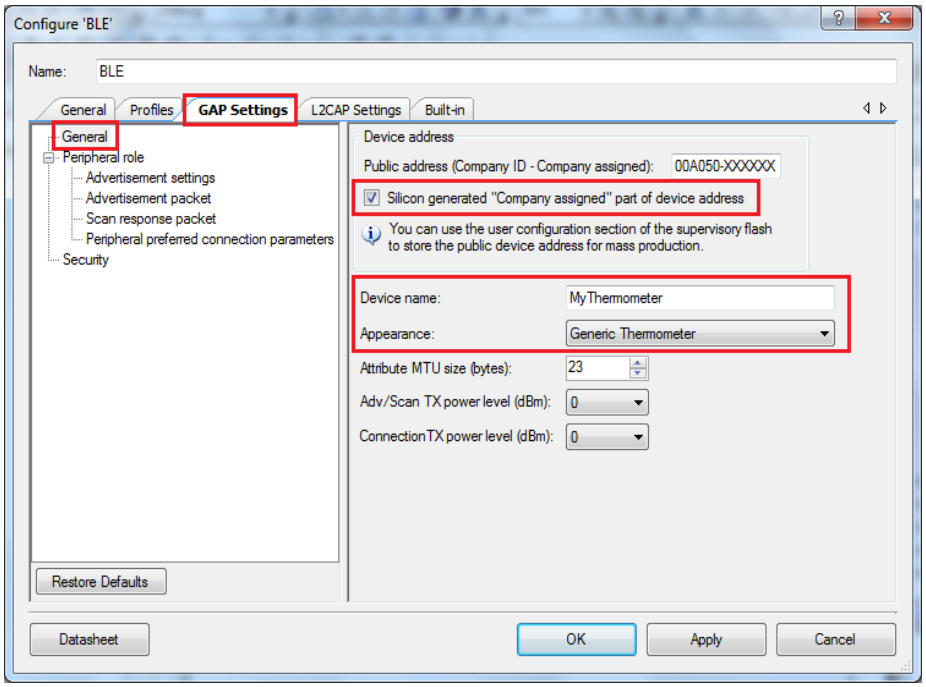
6. 根据表 2 配置 **GAP Settings**（GAP 设置）选项卡中 **General**（通用）项的内容，设置下面的 **Bluetooth Device Address (BD_ADDR)**（蓝牙器件地址）、**Device name**（器件名称）以及 **Appearance**（外观）。

表 2. 常规设置

名称	值	备注	图
公共地址	Check Silicon Generated Address（检查芯片生成地址）	将您公司的 ID 以及公司分配的各项值作为地址使用。如果您没有这些信息，那么可以将所需地址添加到该字段中。	图 11
器件名称	MyThermometer（我的温度计）	用户定义	图 11
外观	Generic Thermometer（通用温度计）	可能值由蓝牙 SIG 定义。BLE 组件提供了选择其中一个可能值的选项。	图 11

注意：公共地址指的是唯一的 48 位 BD_ADDR，用于识别器件。它被分为两部分：公司 ID（24 位）与公司分配值（24 位）。公共地址默认被设置为赛普拉斯半导体的公司 ID。您应该使用由 IEEE 分配的 24 位公司 ID。

图 11. 器件名称与外观



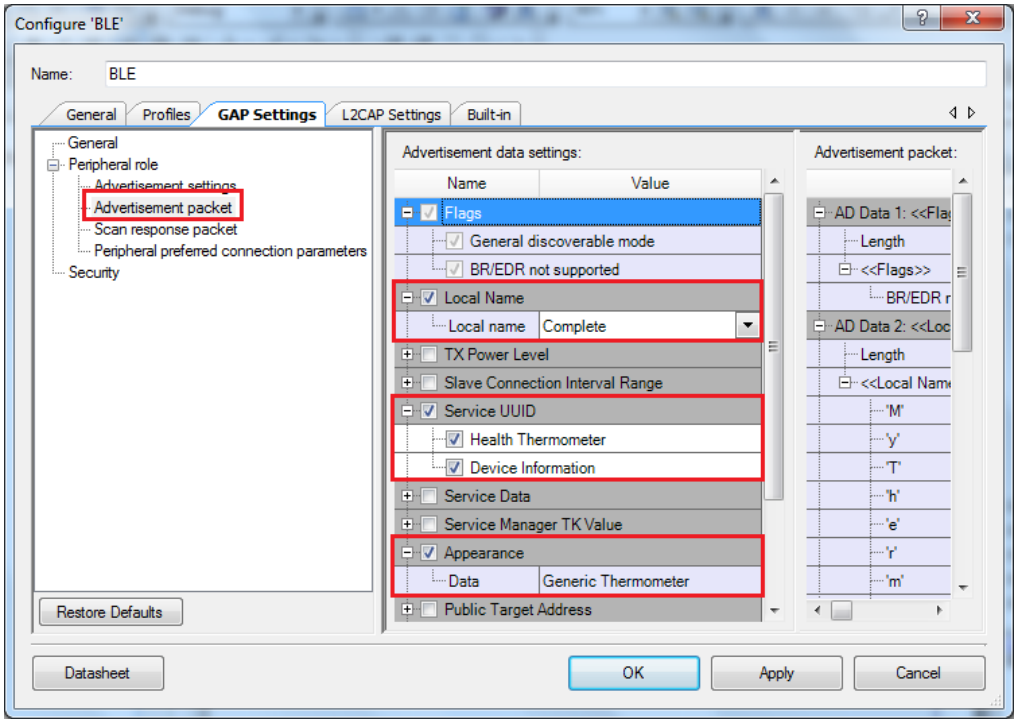
7. 在该项目中，保持 **Peripheral Role**（外设角色）下方的 **Advertisement Settings**（广播设置）默认值。

8. 根据表 3 配置 **Peripheral role**（外设角色）下方的 **Advertisement packet**（广播数据包）。

表 3. 广播数据包设置

名称	选框	值	备注	图
Local Name（本地名称）	使能	完成	将完整名称作为广播数据包的一部分进行传输。您可以选择将要发送的字符数，从而发送缩短的名称。	图 12
Service UUID（服务 UUID）	使能	N/A	将服务 UUID（通用唯一标识符）作为广播数据包的一部分进行传输。	图 12
Service UUID（服务 UUID） > Health Thermometer（健康温度计）	使能	N/A	将健康温度计服务 UUID 作为广播数据包的一部分进行传输。	图 12
Service UUID（服务 UUID） > Device Information（器件信息）	使能	N/A	将器件信息服务 UUID 作为广播数据包的一部分进行传输。	图 12
Appearance（外观）	使能	N/A	将 Appearance（外观）值作为广播数据包的一部分进行传输。	图 12

图 12. GAP 设置 — 广播数据包



保持其他设置项（**Scan response packet**、**Peripheral preferred connection parameters** 以及 **Security** 设置）的默认值。同样，保持 **L2CAP Settings** 选项卡中各设置的默认值。

9. 点击 **Apply**，然后选择 **OK**。
10. 依次放置并配置数字输入引脚组件，如图 13 和图 14 中显示的内容。

图 13. 引脚配置

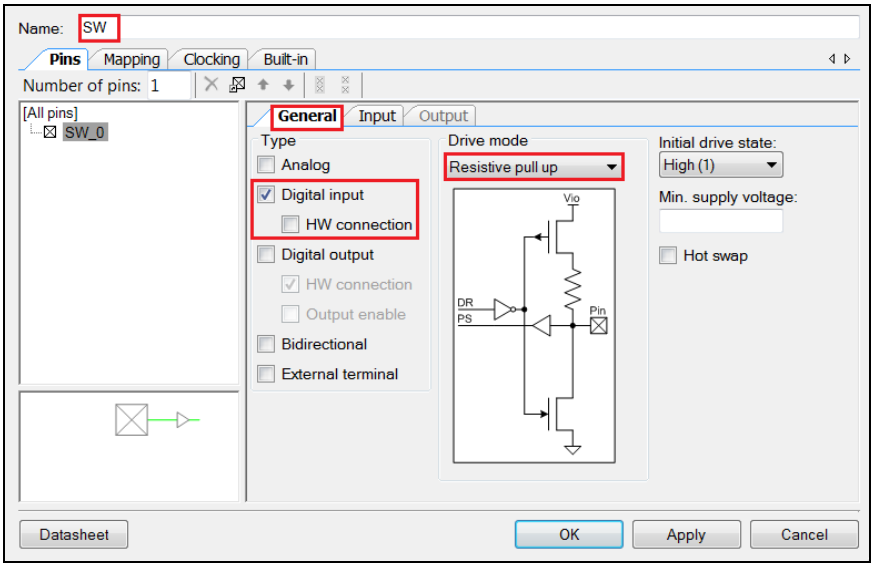
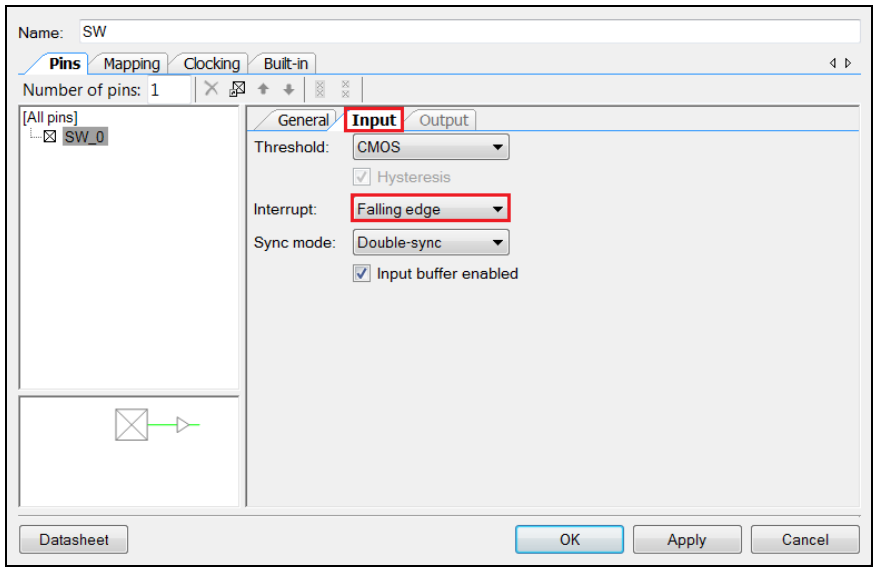


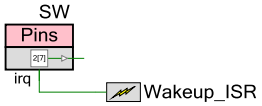
图 14. 引脚中断配置



注意：用户开关的驱动模式将被选定为**电阻上拉**，以保持信号的默认状态为逻辑高电平。当按下该开关时，引脚将被接地，使引脚上信号的状态从逻辑高电平转为逻辑低电平。因此，该中断被设置在**下降沿**上。

11. 放置中断组件并将它连接到 SW 组件的“irq”引脚上，将它重新命名为“Wakeup_ISR”，如图 15 所示。该中断组件用于记录中断信号和触发相应功能。

图 15. SW 引脚



12. 要想测量温度，请按照步骤 13 到步骤 19 中的介绍进行操作。
13. 根据表 4 拖放序列 SAR ADC 组件并配置它。

表 4. SAR ADC 组件配置

选项卡	名称	复选框	值	备注	图
普通	名称	N/A	ADC	可将该名称作为组件的 API 的前缀。	图 16
普通	时钟频率（kHz）	N/A	6000	该应用要求的精度不高。应用的采样率越快，则活动时间越短，也更节省功耗。	图 16
普通	Vref 选择	N/A	VDDA	输入电压范围为 0 到 V_{DDA}。	图 16
普通	进行求平均的采样	N/A	16	平均采样值为 16 使高频噪声达到平均标准。	图 16
通道	已定序通道	N/A	1	将要扫描的通道数	图 17
通道	AVG	已检查	N/A	使能相应 ADC 通道的求平均	图 17

图 16. SAR ADC 配置 — 普通

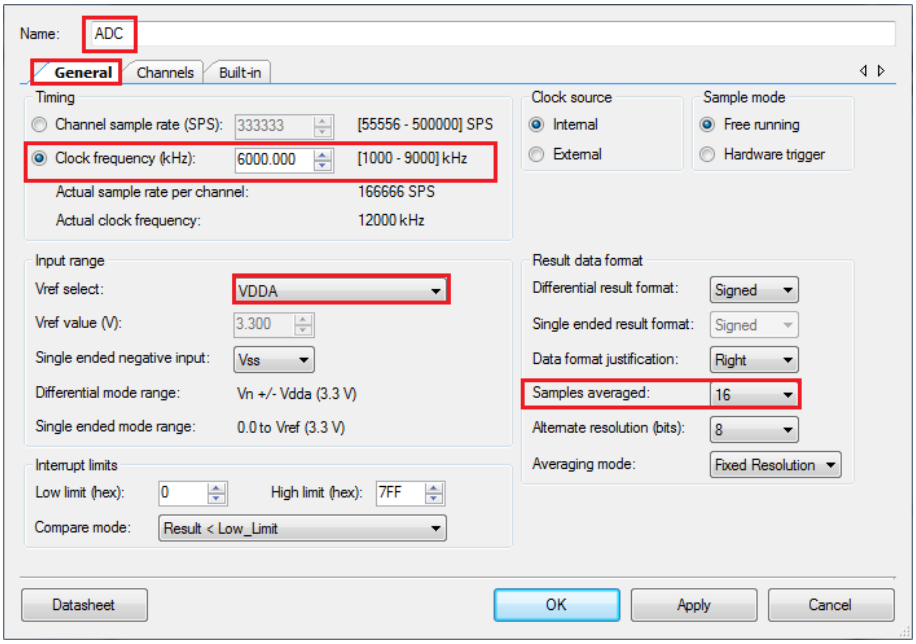
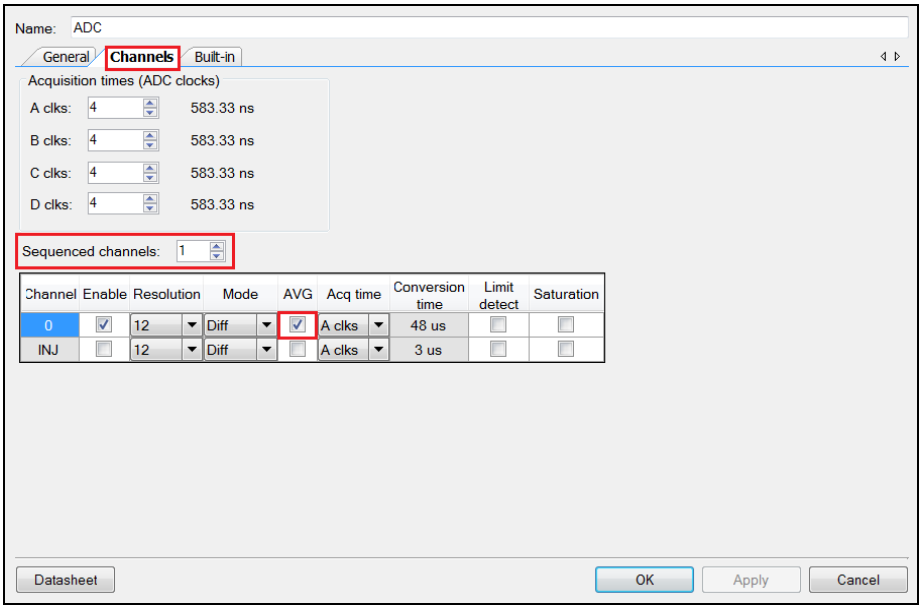
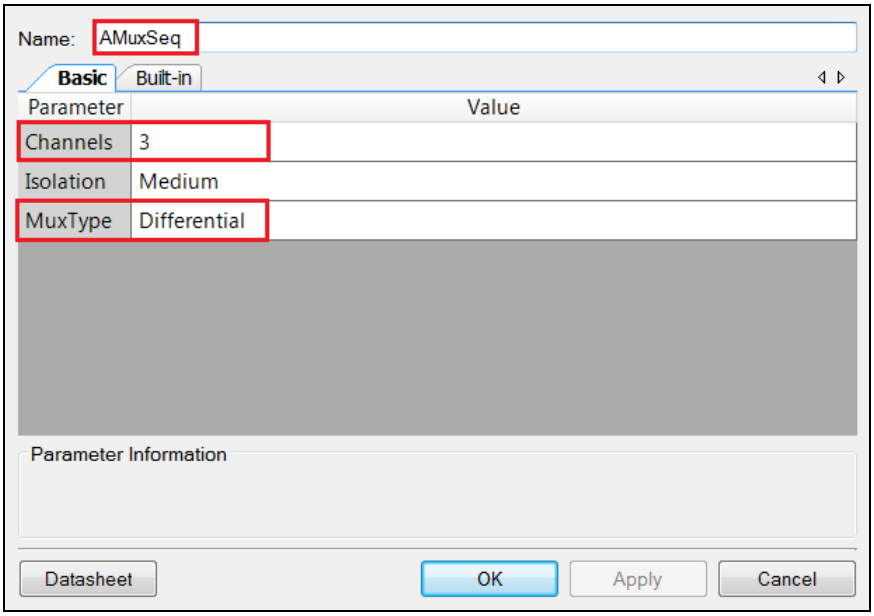


图 17. SAR ADC 配置 — 通道



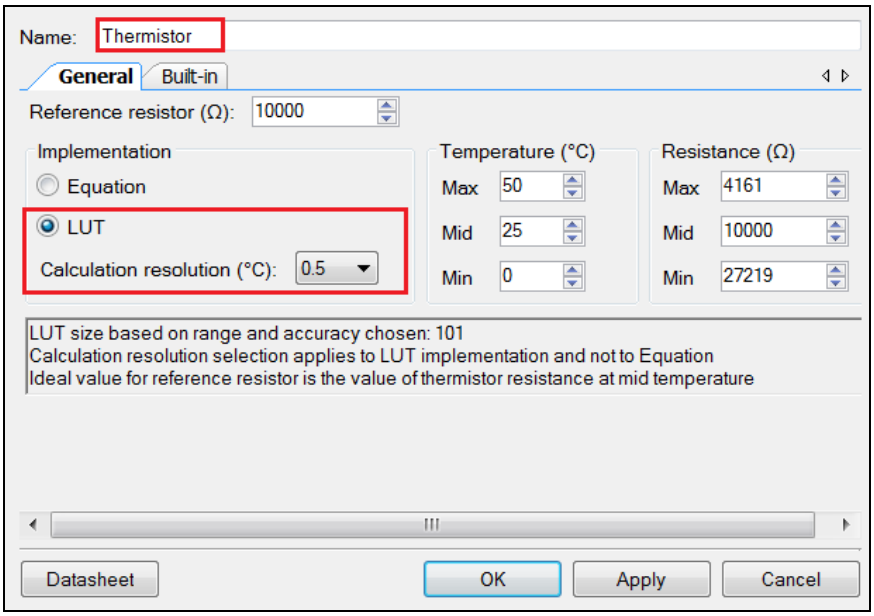
14. 根据图 18 放置和配置模拟复用序列发生器组件。AMUX 组件用于将多个模拟信号复用到 ADC 输入端。

图 18. AMUX 配置



15. 根据图 19 放置和配置热敏电阻计算器组件。通过这些设置，可以配置该组件为使用热敏电阻测量温度的组件。

图 19. 热敏电阻计算器配置



注意：要想准确测量温度，必须使用热敏电阻数据手册中给定的值来更新温度（°C）和电阻值（Ω）列。

16. 根据图 20 和图 21 分别放置和配置两个模拟引脚组件。

注意：可以将这些引脚配置为模拟引脚和数字输出。数字输出使用户能够将引脚从固件驱动到 Vdd 或 Vss。

图 20. 模拟引脚配置 — V_HIGH

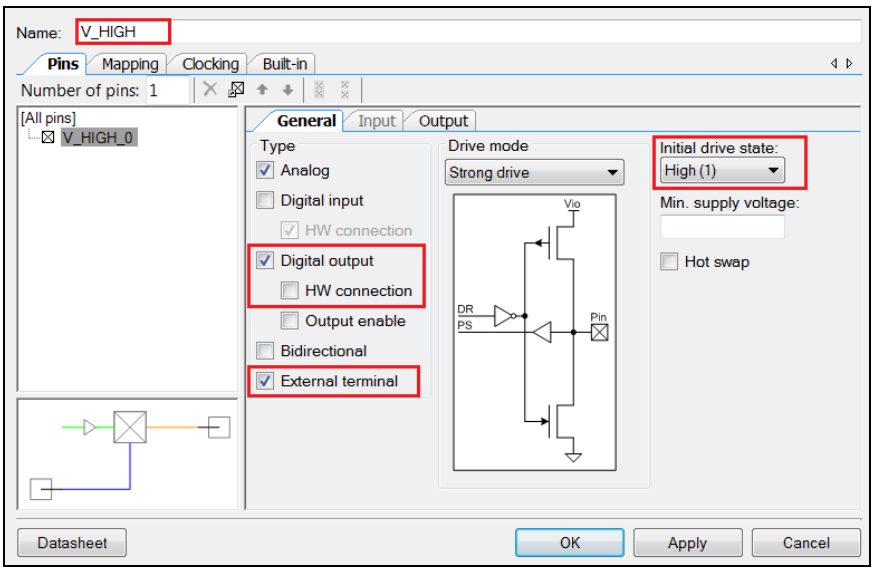
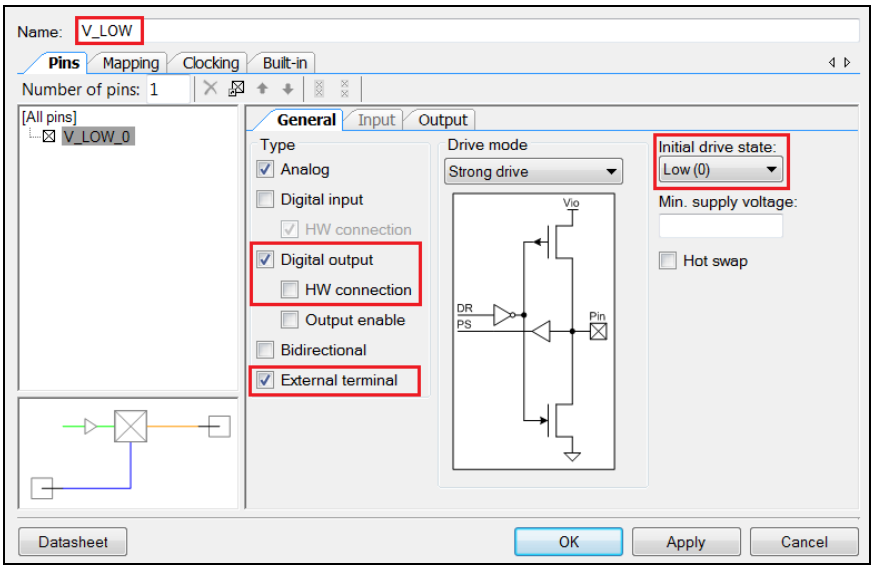


图 21. 模拟引脚配置 — V_LOW

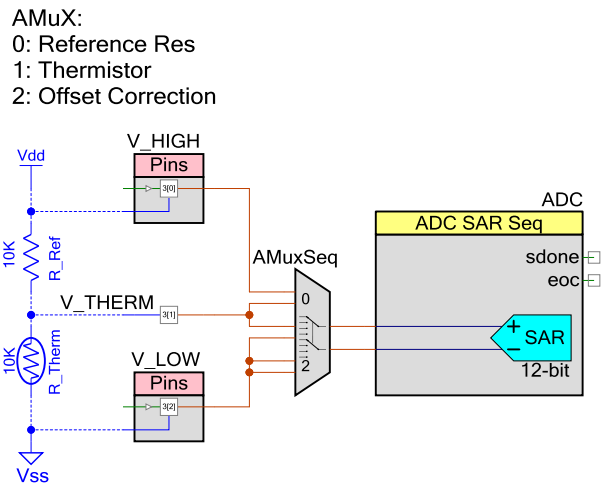


17. 放置模拟引脚组件，将其命名为“V_THERM”。使能**外部终端**。

18. 根据图 22 所示的原理图进行连接模拟复用序列发生器、序列 SAR ADC 和模拟引脚组件。

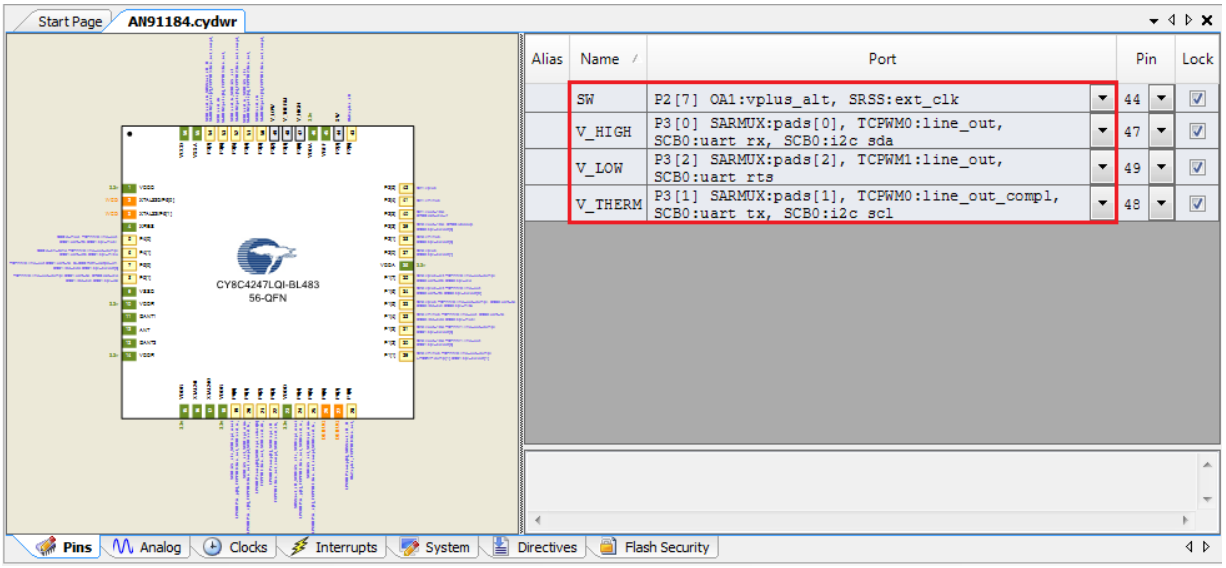
    注意：所有蓝色组件都包含在“片外”目录中。这些组件仅供参考使用，但通过它们可以使原理图更加明确地表示硬件的结构。

图 22. 原理图 — 温度测量



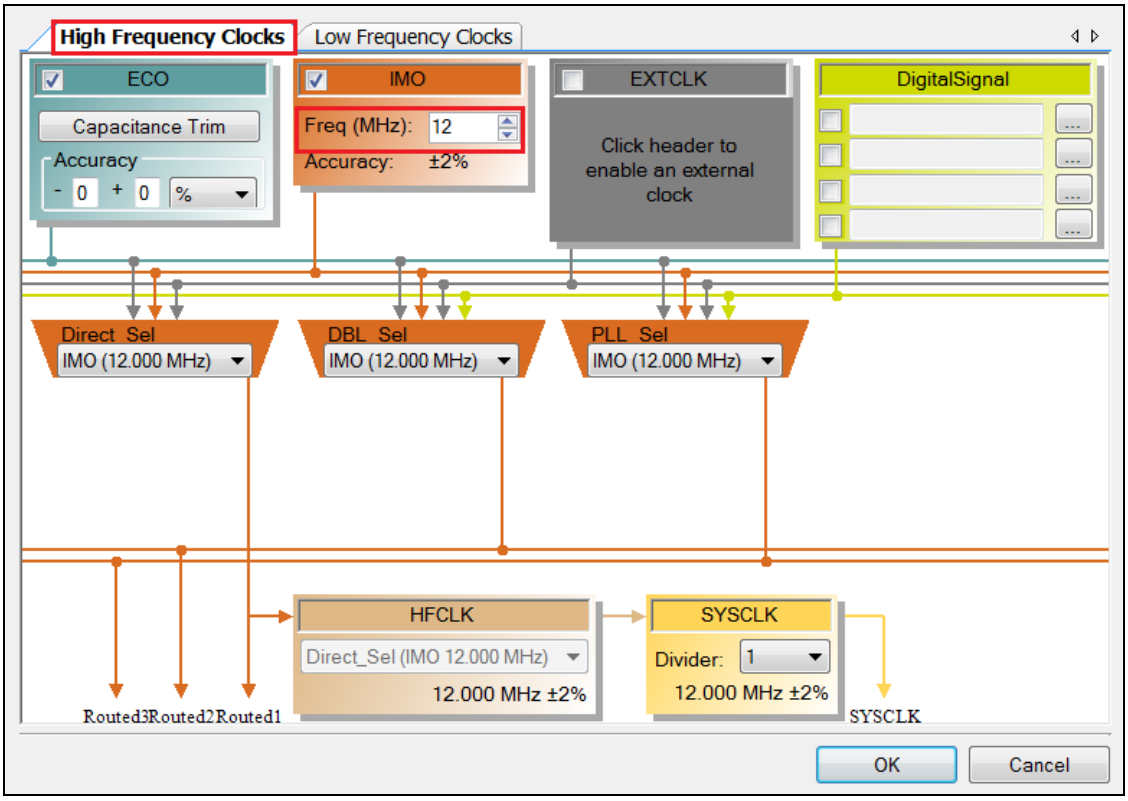
19. 在设计范围资源窗口的 Pins（引脚）选项卡中，根据图 23 所示内容连接引脚。

图 23. 设计范围资源窗口的 Pins（引脚）选项卡



20. 在设计范围资源窗口的 Clocks（时钟）选项卡中，将 IMO 频率配置为 12 MHz，如图 24 所示。

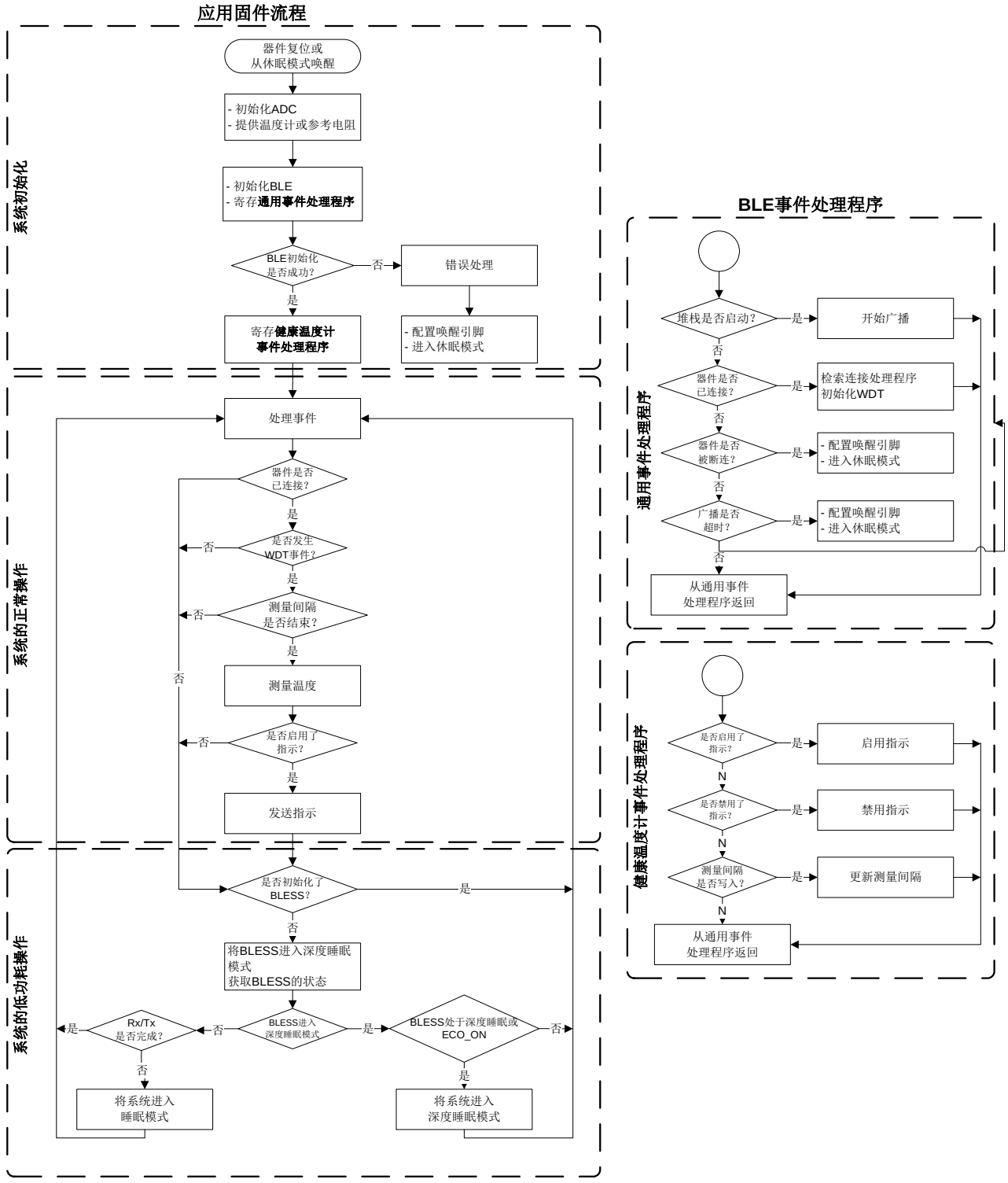
图 24. 时钟配置



4.2 配置固件

图 25 显示的是健康温度计应用的固件流程。

图 25. 系统流程图



注意：应用固件的源文件包含在示例项目（本应用笔记附带）中。您可以将源文件添加到您自己的项目中，或使用原来完整的示例项目。示例项目有 9 个源文件，详细列在表 5 中。

表 5. 示例项目中的源文件

名称	说明
main.c	它是一个主固件文件。它只有一个函数： ■ main() — 该函数控制着完整应用的流程。该函数的主要任务是初始化系统、应用控制（包括处理 BLE 事件、应用流程控制以及低功耗实现）。
CommonFunctions.c/.h	该文件实现了用于进行控制的通用函数。该文件具有以下函数： ■ InitializeSystem() — 初始化所有系统的模块。 ■ PrepareForDeepSleep — 通过使硬件模块进入深度睡眠模式，从而降低系统功耗。 ■ WakupFromDeepSleep() — 为正常操作恢复硬件模块。
Temperature.c/.h	该函数会从 ADC 读取测量到的数据并使用热敏电阻计算器组件计算温度，从而实现温度测量。它还为用户提供仿真温度选项，无需从温度传感器测量温度。该文件具有以下函数： ■ MeasureSensorVoltage() — 该函数测量了传感器电压。如果用户选择了仿真选项，那么该函数不可用。更多详细信息，请参考传感器仿真一节。 ■ ProcessTemperature() — 该函数测量或仿真温度值。
WatchdogTimer.c/.h	它执行看门狗定时器功能并跟踪系统时间。该文件具有以下函数： ■ WatchdogTimer_Start() — 启动看门狗定时器（WDT0）（以 1 s 周期为单位），如果该定时器的值与某个值相匹配，会产生中断。 ■ WatchdogTimer_Isr() — 为 WDT 服务 ISR 的函数，用于跟踪测量间隔。该函数是看门狗定时器的回调函数。 ■ WatchdogTimer_Stop() — 终止看门狗定时器（WDT0）。
BLE_HTSS.c/.h	该文件执行了项目的 BLE 指定功能。它处理了由 BLE 堆栈生成的事件，在事件处理程序一节中详细说明该过程。该文件具有以下函数： ■ GenericEventHandler() — 处理了由 BLE 堆栈生成的通用事件。 ■ HtssEventHandler() — 处理了为健康温度计服务生成的事件。 ■ ProcessBLE() — 将温度数据作为指示并发送到 GATT 客户端。 ■ EnableBLE() — 启动 BLE 组件并寄存事件处理程序函数。

以下章节介绍的是健康温度计应用的操作。请注意，本文档不包含完整固件。只是详细说明了各种主要概念。请参考所包含的示例项目，了解完整固件。

健康温度计应用状态机包括以下四种状态：

- 系统初始化
- 事件处理程序
  - 通用事件处理程序
  - 健康温度计事件处理程序
- 系统的正常操作
- 系统的低功耗操作

这些注意事项将在后面的章节中详细说明。

4.2.1 系统初始化

当器件复位或从休眠模式唤醒时，固件将先执行初始化，包括启动 SAR ADC、使能全局中断、启动运算放大器和看门狗定时器。系统被初始化后，固件将初始化 BLE 组件，从而实现初始化整个 BLE 子系统。

注意： 在进行 BLE 组件初始化过程中，用户代码需要将一个指针传输给事件处理函数（需要调用该函数以接收 BLE 堆栈中的事件）。图 25 中所显示的通用事件处理程序被注册为 BLE 初始化过程的一部分。代码 1 显示了启动 BLE 组件和寄存通用事件处理程序的代码。

代码 1. BLE 初始化

```
apiResult = CyBle_Start(GenericEventHandler);
```

如果 BLE 组件初始化成功，固件会存储被调用以接收健康温度计服务事件的函数，并切换到正常工作模式。代码 2 显示的是用于寄存健康温度计服务的代码段。

代码 2. 健康温度计服务事件处理程序

```
CyBle_HtsRegisterAttrCallback(HealthThermometetEventHandler);
```

4.2.2 事件处理程序

在 BLE 组件中，通过一系列事件，在 BLE 堆栈上执行的所有操作的结果都将被反馈到应用固件。这些事件提供了 BLE 接口状态和数据。可以将这些事件分为以下几类：

- 常用事件

  在 GAP 层、GATT 层和堆栈上 L2CAP 层执行操作会生成这些事件。例如，当初始化并启动 BLE 堆栈时，将接收 CYBLE_EVT_STACK_ON 事件；当建立了与远程器件的连接时，将接收 CYBLE_EVT_GAP_DEVICE_CONNECTED 事件；当收到从客户端发送的写指令时，将生成 CYBLE_EVT_GATTS_WRITE_CMD_REQ 事件。更多有关常用事件的详细信息，请参考 BLE 组件的 API 文档（右击 PSoC Creator 中的 BLE 组件并选择 **Open API Documentation**（打开 API 文档））。

  应用固件必须包含一个事件处理程序函数，用于成功建立和维持 BLE 链接。代码 3 显示的是 GenericEventHandler 函数的实现情况，其中在初始化 BLE 堆栈过程中所生成的事件、器件连接、器件断连以及超时都经过处理。

- 服务特定事件

  服务特定事件是通过在标准服务（由蓝牙 SIG 定义）上执行操作生成的。例如，当客户端写入客户端配置特性描述符（以使能温度测量特性的指示）时，服务器将收到 CYBLE_EVT_HTSS_INDICATION_ENABLED 事件。更多有关服务特定事件的详细信息，请参考 BLE 组件的 API 文档。

  BLE 组件可以将这些事件路由到一个服务特定的事件处理程序。应用固件应包含一个服务特定事件处理程序函数，用于处理这些事件。如果应用固件不支持服务特定事件处理程序函数，那么常用事件处理程序函数（GenericEventHandler）将处理这些事件。代码 4 显示的是服务特定事件处理程序（即 HtssEventHandler）的实现情况。

代码 3. 通用事件处理程序

```
void GenericEventHandler(uint32 event, void *eventParam)
{
  switch(event)
  {
    /* This event is received when component is Started */
    case CYBLE_EVT_STACK_ON:
      /* Stop watchdog to reduce power consumption during advertising */
      WatchdogTimer_Stop();
      /* Start Advertisement and enter Discoverable mode*/
      CyBle_GappStartAdvertisement(CYBLE_ADVERTISING_FAST);
    break;
```

```
            /* This event is received when device is disconnected or advertising times out*/
            case CYBLE_EVT_GAP_DEVICE_DISCONNECTED:
            case CYBLE_EVT_TIMEOUT:
              /* Sets the ENABLE_HIBERNATE flag to put system in Hibernate mode */
              SystemFlag |= ENABLE_HIBERNATE;
            break;

            /* This event is received when connection is established */
            case CYBLE_EVT_GATT_CONNECT_IND:
                /* Start watchdog timer with 1s refresh interval */
                /* Note: For this application, wakeup should be 1s because htssInterval
                *  resolution is configured as 1s */
                WatchdogTimer_Start(REFRESH_INTERVAL);
                /* Retrieve BLE connection handle */
                connectionHandle = *(CYBLE_CONN_HANDLE_T *) eventParam;
            break;

            default:
              /* Error handling */
            break;
          }
        }
```

代码 4. 健康温度计服务事件处理程序

```
        void HtssEventHandler(uint32 event, void* eventParam)
        {
          CYBLE_HTS_CHAR_VALUE_T *interval;
          switch(event)
          {
            /* This event is received when indication are enabled by the central */
            case CYBLE_EVT_HTSS_INDICATION_ENABLED:
              /* Set the htssIndication flag */
              htssIndication = true;
            break;

            /* This event is received when indication are disabled by the central */
            case CYBLE_EVT_HTSS_INDICATION_DISABLED:
              /* Reset the htssIndiciation flag */
              htssIndication = false;
            break;

            /* This event is received when measurement interval is updated by
            *  the central */
            case CYBLE_EVT_HTSS_CHAR_WRITE:
              /* Retrive interval value */
              interval = ((CYBLE_HTS_CHAR_VALUE_T *)eventParam);
              htssInterval = interval->value->val[1];
              /* Update htssInterval with the updated value */
              htssInterval = (htssInterval << 8) | interval->value->val[0];
            break;

            default:
              /* Error handling */
            break;
          }
        }
```

系统的正常操作模式

在系统正常工作状态下，固件会周期性调用 `CyBle_ProcessEvents()`（以便处理 BLE 堆栈相关的操作）并检查连接是否已经建立。

注意： 将 BLE 堆栈相关的操作（从链路层发送数据、将数据发送到链路层、将所生成的事件传送到应用层）作为 `CyBle_ProcessEvents()` 函数调用的一部分执行。在该应用中，代码 1 会初始化堆栈，但仅在调用 `CyBle_ProcessEvents()` 函数时才生成与堆栈相关的事件。同样，仅在调用 `CyBle_ProcessEvents()` 函数时，才会生成与器件连接、器件断连、广告超时以及健康温度计服务相关的其他事件。

如果建立了连接，固件会定期（常规间隔）测量温度，该间隔由健康温度计服务的测量间隔特性配置的。测量温度后，如果中心器件使能了指示，则固件会将温度数据作为指示发送给 BLE 中心器件。

在 BLE 应用中，根据 BLE 连接的情况，器件会传输或接收数据，并仅在周期间隔（又称为广告间隔或连接间隔）进行该操作。因此，当完成系统正常工作任务时，为了节省功耗，器件会进入系统低功耗工作模式，并在下一个连接/广告间隔被唤醒。

4.2.3 系统的低功耗工作状态

在系统低功耗工作状态下，器件将在三种可能功耗模式中的一种下进行操作。

- 睡眠模式

  当 CPU 闲置但 BLE 子系统（BLESS）处于活动状态并忙于传输或接收数据时，器件会进入该模式。在这种情况下，CPU 将被置于睡眠模式，内核其余部分（如时钟和电压调节器）则保持活动状态，从而能够随时执行正常 BLE 操作。

  为了节省功耗，内部主振荡器（IMO）的频率降低到 3 MHz，器件唤醒时，它将被切换回 12 MHz。

- 深度睡眠模式

  固件连续尝试使 BLESS 进入深度睡眠模式。成功将 BLESS 置于深度睡眠模式后，系统其余部分也会进入深度睡眠模式。

  注意： 使 BLESS 进入深度睡眠模式后，器件将立即进入深度睡眠模式。如果器件没有立即进入该模式，固件应该禁用中断（以免服务 ISR）并重新检查 BLESS 仍处于深度睡眠模式还是 ECO_ON 模式。如果 BLESS 处于这两种模式中的一种，那么器件可以安全进入深度睡眠模式；否则，它必须等到 Rx/Tx 事件完成为止。

- 休眠模式

  如果器件被断开连接或广告间隔超时，它会进入超低功耗模式，该模式又称为“休眠模式”。从该模式唤醒后，即使 RAM 中的内容被保留，固件仍从 *main.c* 的开头部分开始执行。

4.2.4 传感器仿真

如果没有温度计或参考电阻用于测量温度，您可以使用温度仿真模式测试该应用。在该模式下，温度数据被仿真，并且每经过一个测量间隔（默认值为 1 秒）它会递增 1 ^{0}C。通过将 *Temperature.h* 文件中的 MEASURE_TEMPERATURE_SENSOR 常数值从 1 变为 0，如代码 5 中所示，可以在应用代码中完成该操作。

代码 5. 仿真温度传感器

```
#define MEASURE_TEMPERATURE_SENSOR (0u)
```

4.3 硬件配置

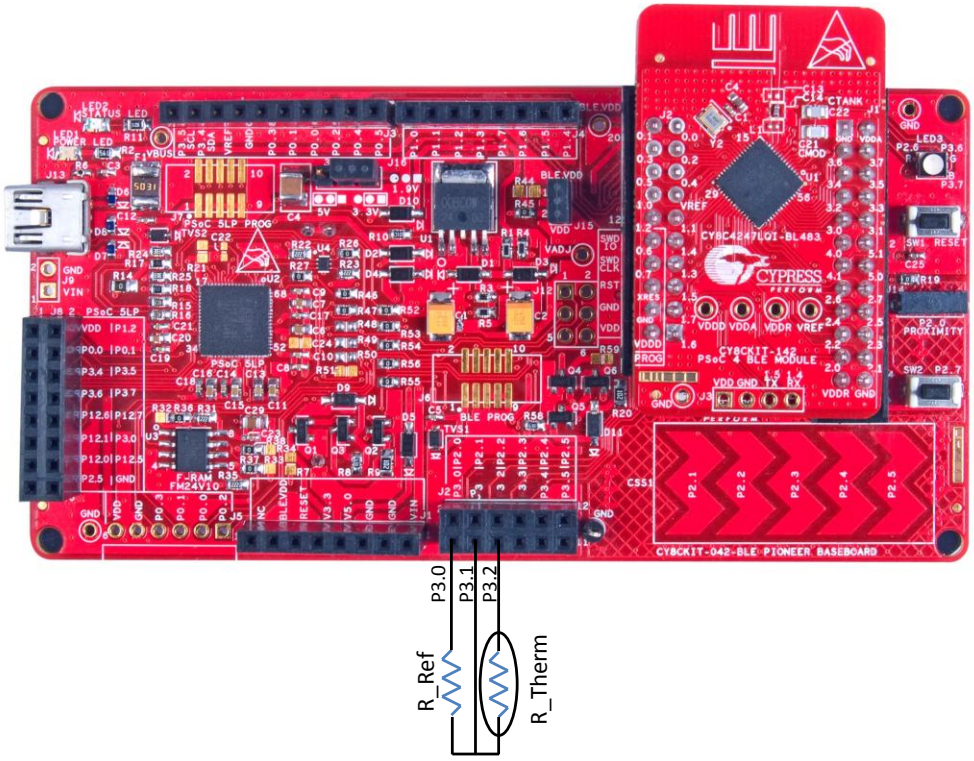
CY8CKIT-042-BLE 低功耗蓝牙（BLE）Pioneer 套件是赛普拉斯 PSoC 4 BLE 和 PRoC BLE 器件的开发套件。以灵活性为设计目标，本套件提供了与很多第三方的 Arduino™扩展板相兼容的插座。该套件提供了一个额外的插座，用于支持 Digilent® Pmod™的外设模块。另外，该电路板还集成了 CapSense®滑条、板上 1 Mb FRAM、RGB LED、按键开关、集成 USB 编程器、编程调试插座及 USB-UART/I^{2}C 桥接器等功能。

1. 将 CY8CKIT-142 PSoC 4 BLE 模块（红色的模块）放置在 BLE Pioneer 基板上。

   该套件不包含温度计，因此要想测试健康温度计应用，您可以使用 BLE Pioneer 套件和下面各选项中的一个：

   - 外部温度计 — 将一个大小为 10 kΩ 的外部温度计和一个参考电阻连接至 BLE 套件，如图 26 所示。

图 26. 外部温度计的连接情况

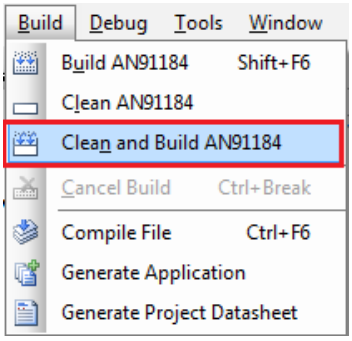


- 仿真传感器 – 固件不使用硬件传感器，它仿真了温度数据，并且每经过一个测量间隔（默认值为 1 秒），温度递增 1 ºC。更多详细信息，请参考传感器仿真一节。

4.4 构建并编程器件

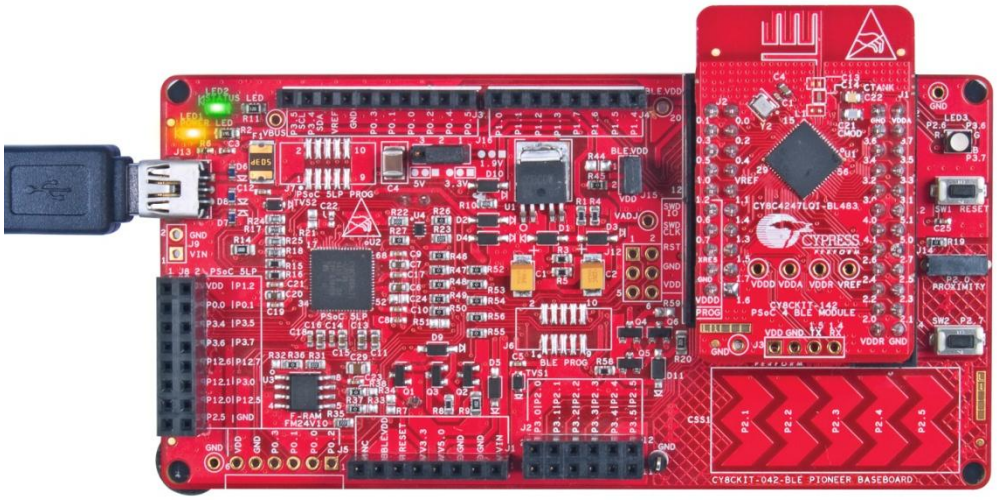
1. 依次选择 **Build > Build AN91184**，以便构建和编译固件，如图 27 所示。项目应该在无警告或错误的情况下完成构建流程。

图 27. 构建项目



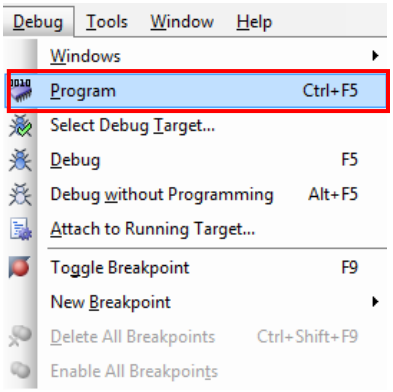
2. 将 PSoC 4 BLE 模块（红色模块）插入到 BLE Pioneer 基板上，然后通过使用 USB 标准 A 转 Mini B 型线缆将该套件连接至您的 PC（请参见图 28）。您可以在 PC 上完成 USB 枚举。

图 28. 将 BLE Pioneer 基板连接到 PCB（通过使用 USB 线缆）上



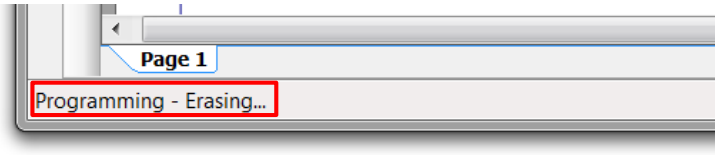
3. 依次选择 **Debug** > **Program**，如图 29 所示。如果只有一个套件连接至 PC，将自动开始编程。如果有多个套件连接至 PC，那么 PSoC Creator 将提示您选择要编程的套件。

图 29. 编程器件



可在 PSoC Creator 状态栏上查看编程状态（窗口的左下角），如图 30 所示。

图 30. 编程状态



5 应用测试

可以使用 CySmart 中心仿真工具或 CySmart 手机应用测试健康温度计应用。

5.1 CySmart 中心仿真工具

您可以使用 CySmart 中心仿真工具以及 CY5670 CySmart USB 收发器（BLE 收发器）来测试和验证 BLE 外设器件的操作。

您可以从 www.cypress.com/cysmart 网站上下载最新的 CySmart 中心仿真工具。

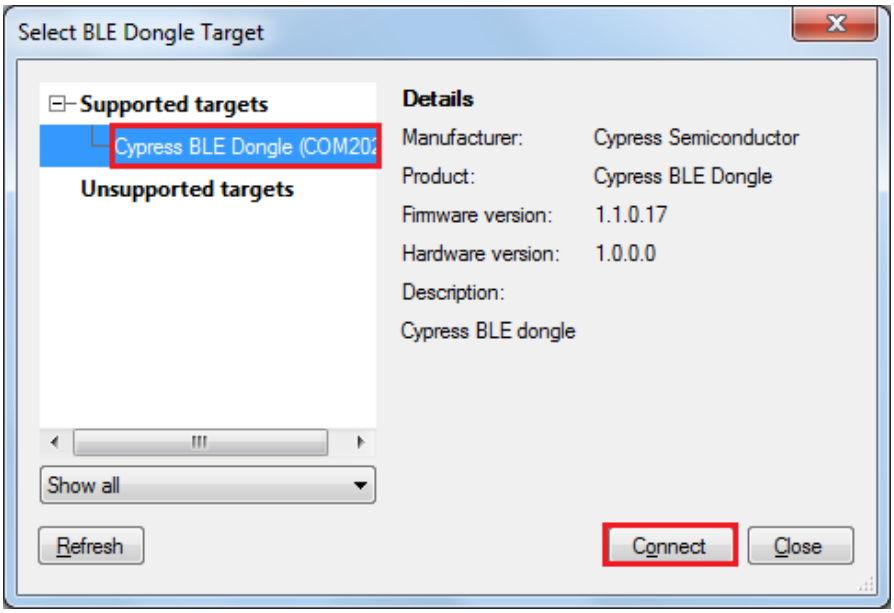
要想使用 CySmart 中心仿真工具验证健康温度计应用，请依次执行以下操作：

1. 将 BLE 收发器连接到 PC 上，并通过依次选择 **Start** > **All Programs** > **CySmart 1.1** > **CySmart 1.1.**启动 CySmart 中心仿真工具

   CySmart 中心仿真工具会检测 USB 驱动器所连接的 BLE 收发器。

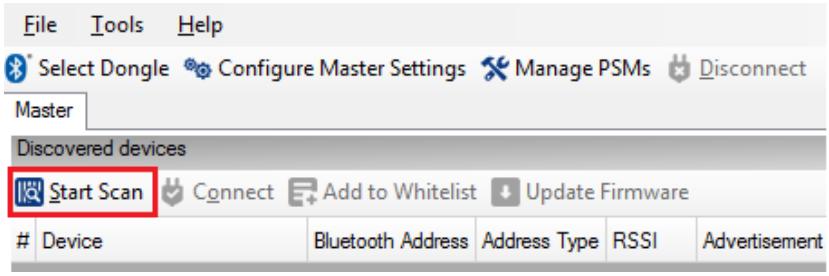
2. 点击 **Connect** 以连接至 BLE 收发器，如图 31 所示。

图 31. 选择 BLE 收发器的目标



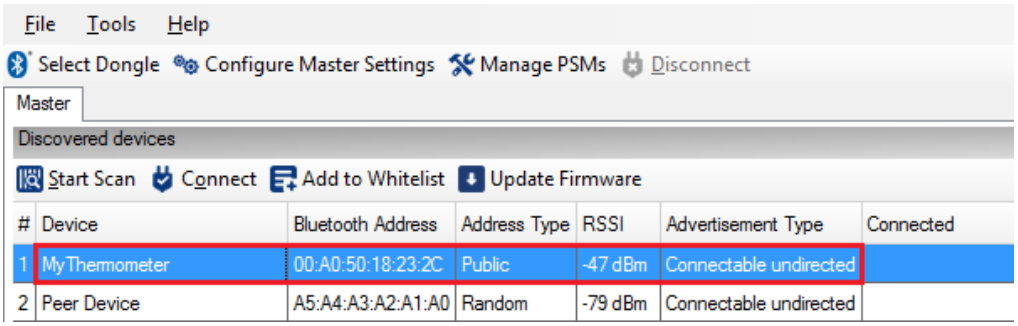
3. 当 PC 与 BLE 收发器相连时，点击 **Start Scan** 以寻找 BLE 器件，如图 32 所示。

图 32. 启动扫描



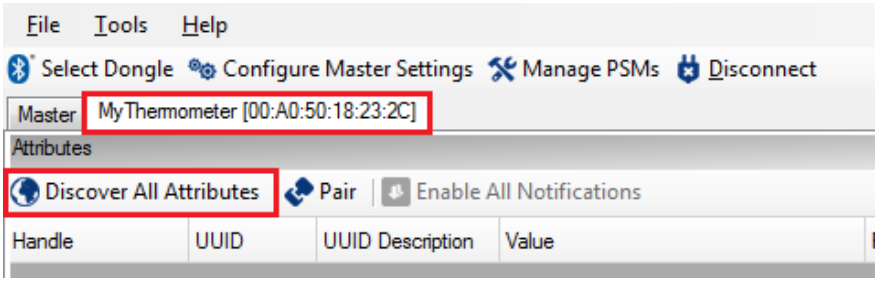
在 CySmart 中心仿真工具窗口中列出了检测出的 BLE 器件，如图 33 所示。

图 33. 检测出的器件



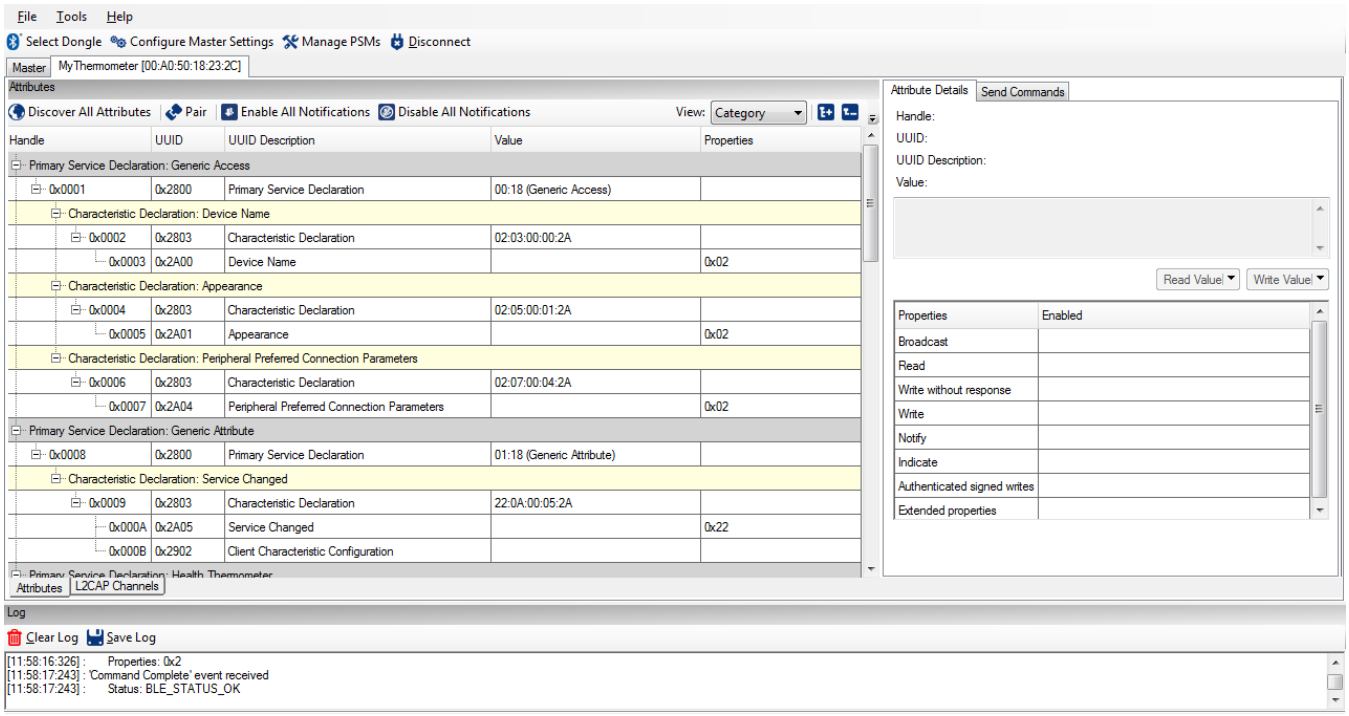
4. 点击名称，选择器件“MyThermometer”，然后点击 **Connect**，如图 33 所示。
   当 BLE 收发器被连接至 BLE 器件时，会添加带有器件名称和 BD_ADDR 的新窗口，如图 34 所示。

图 34. 连接至 MyThermometer



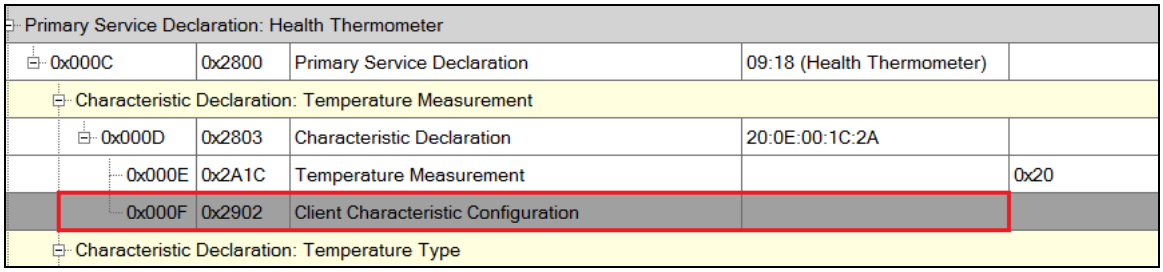
5. 要想检测出外设“MyThermomemeter”显示的所有属性，请点击 **Discover All Attributes**，如图 34 所示。
   检测出的所有属性将被分组和显示，如图 35 所示。

图 35. 检测属性



6. 要想读取测量得出的温度，选择 **Temperature Measurement** 特性的 **Client Characteristic Configuration** 描述符，如图 36 和图 37 所示。

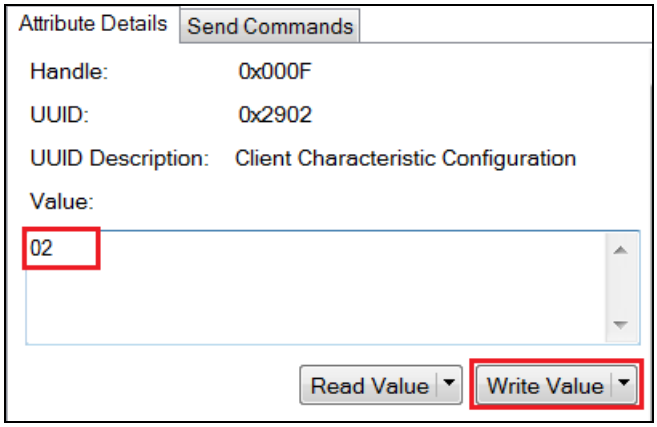
图 36. 使能指示 — 步骤 A



7. 将 **02** 写入到 **Client Characteristic Configuration** 描述符中，以使能指示，如图 37 所示。

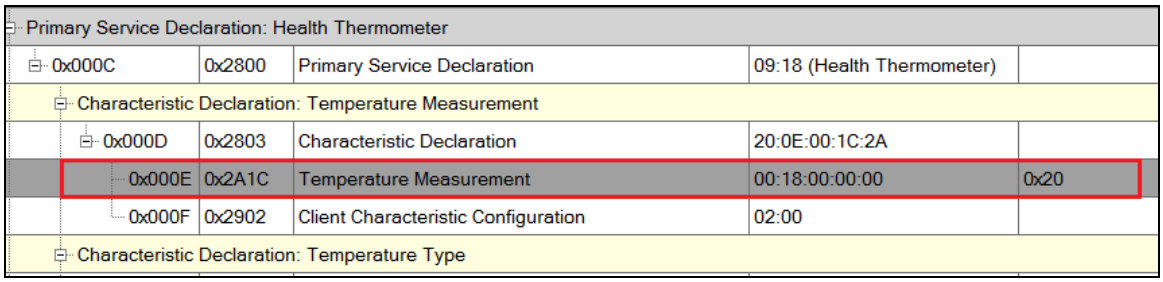
注意： 有关位定义的更多详细信息，请参考客户端特性配置描述符。

图 37. 使能指示 — 步骤 B



使用测量得出的温度值更新“温度测量”属性，如图 38 所示。

图 38. 测量得出的温度



回读值是一个 5 字节的十六进制数值，其中第一个字节是 BLE 组件中配置的标志，其余 4 个字节的数值是以从小到大的格式显示的温度值。因此，测量得出的温度值为 24 ºC（0x18）。

请参考 CySmart 用户指南，详细了解 CySmart 中心仿真工具。

5.2 CySmart 手机应用

赛普拉斯提供了一个手机应用程序，用于验证 BLE 应用。该应用支持各种标准和自定义配置文件。它还提供了能够观察 GATT 数据库的用户界面。

您可以从 iOS 设备的 Apple 应用商店和 Android 设备的 Google Play 商店下载 CySmart 应用。

- Apple 应用商店：点击此处。
- Google Play 商店：点击此处。

要想使用 CySmart 手机应用程序验证健康温度计应用，请按照下列步骤执行操作：

1. 打开您设备上的 CySmart 应用，如图 39 所示。

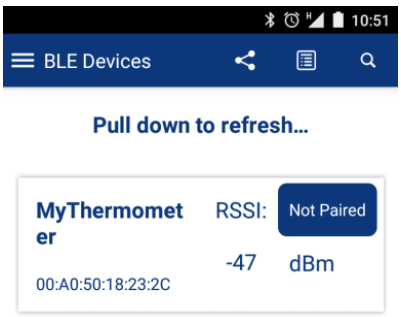
   注意： 这些屏幕截图是 Android 设备上的 CySmart 应用。iOS 设备上 CySmart 应用的外观和体验可能有点不同。

图 39. BLE 配置



2. 如果已经激活蓝牙，手机设备会扫描 BLE 器件并在屏幕上列出扫描结果；否则，它会提示用户打开蓝牙，然后搜索 BLE 器件。图 40 显示的是附近的各个 BLE 器件。

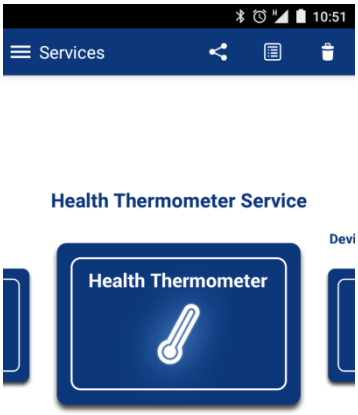
图 40. 器件列表



3. 点击器件名称，选择器件“MyThermometer”，如图 40 所示。

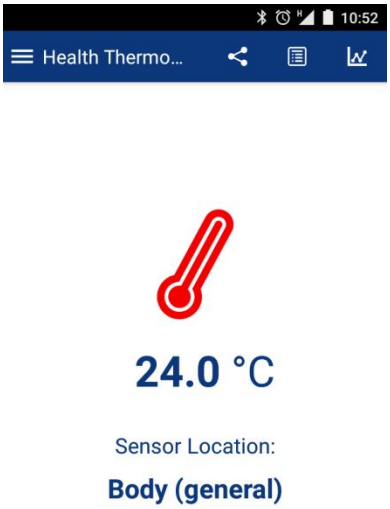
4. 如果建立了连接，则应用会自动检测所有属性，并以旋转格式显示检测出的服务，如图 41 所示。

图 41. 主页



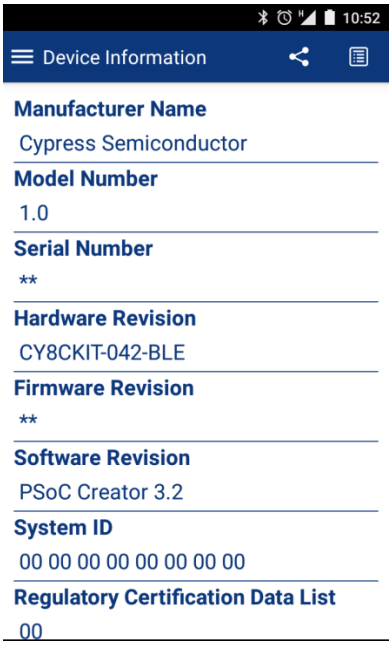
5. 选择健康温度计服务。它会报告当前的温度和传感器位置，如图 42 所示。

图 42. 健康温度计服务



6. 点击屏幕上的返回键，返回到 CySmart 应用的桌面。
7. 选择“器件信息”服务。它显示了被配置为项目一部分的器件信息，如图 43 所示。

图 43. 器件信息服务



5.3 总结

在本应用笔记中，我们了解了如何使用 PSoC Creator BLE 组件来设计 BLE 健康温度计应用（通过结合使用标准 BLE 配置文件）。然后，我们还可以使用赛普拉斯提供的 CySmart 中心仿真工具和 CySmart 手机应用程序来验证该应用。

6 相关应用笔记

- AN91267 — PSoC 4 BLE 入门
- AN91162 — 创建自定义的 BLE 配置文件
- AN92584 — 低功耗设计及估计 BLE 应用的电池使用寿命
- AN66477 — PSoC® 3、PSoC 4 和 PSoC 5LP — 使用热敏电阻测量温度

关于作者

姓名： Pushek Madaan

职务： 应用工程师

文档修订记录

文档标题：AN91184 — PSoC® 4 BLE — 设计 BLE 应用

文档编号：002-00012

版本	ECN	变更者	提交日期	变更说明
**	4938023	ROWA	09/28/2015	本文档版本号为 Rev**，译自英文版 001-91184 Rev*B。
*A	5692176	AESATMP8	04/19/2017	更新徽标和版权。

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、厂商代表和经销商组成的全球性网络。要想查找离您最近的办事处，请访问赛普拉斯所在地。

产品

ARM® Cortex® 微控制器	cypress.com/arm
汽车级产品	cypress.com/automotive
时钟与缓冲器	cypress.com/clocks
接口	cypress.com/interface
物联网	cypress.com/iot
存储器	cypress.com/memory
微控制器	cypress.com/mcu
PSoC	cypress.com/psoc
电源管理 IC	cypress.com/pmic
触摸感应	cypress.com/touch
USB 控制器	cypress.com/usb
无线连接	cypress.com/wireless

PSoC® 解决方案

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

赛普拉斯开发者社区

论坛 | WICED IoT 论坛| 项目 | 视频 | 博客 | 培训 | 组件

技术支持

cypress.com/support

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。

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

赛普拉斯半导体公司，2015-2017 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属人性质的、非独家且不可转让的如下许可（无再许可权）（1）在赛普拉斯特软件著作权项下的下列许可权（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码形式的向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。

在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。赛普拉斯保留更改本文件的权利，届时将不另行通知。在适用法律允许的限度内，赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为供参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失其他用途（“非预期用途”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何主张、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的主张，并使之免受损失。

赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，WICED，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。请访问 cypress.com 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。