PSoC™ 4 CY8C4xxx-BL MCU with AIROC™ Bluetooth® LE 入門

About this document

Scope and purpose

本アプリケーション ノート (AN91267) では、Bluetooth® LE 無線システムを統合した Arm® Cortex®-M0 ベースのプログラマブル システムオンチップ (PSoC™) である PSoC™ 4 with AIROC™ Bluetooth® LE (PSoC™ 4 CY8C4xxx-BL MCU with AIROC™ Bluetooth® LE)をご紹介します。本資料は、PSoC™ 4 with AIROC™ Bluetooth® LE のアーキテクチャとその開発ツールを探究することをご支援し、PSoC™ Creator という PSoC™ 4 with AIROC™ Bluetooth® LE の開発ツールを使用してどのように簡単に Bluetooth® LE 設計を作成できるかを説明します。また、設計開発時間を短縮させ、PSoC™ 4 with AIROC™ Bluetooth® LE を深く理解できるように更なるリソースへのアクセスをご提供します。

Intended audience

AIROC™ Bluetooth® LE を搭載した PSoC™ 4 の新規ユーザーを対象としています。

関連製品ファミリ

CY8C41x7-BL、CY8C42x7-BL、CY8C41x8-BL、CY8C42x8-BL

ソフトウェア バージョン

PSoC™ Creator 4.4

さらに多くのサンプルコードが必要な場合は、以下を参照してください。

PSoC™サンプル コードのリストにアクセスするには、**サンプル コードのウェブページ**を参照してください。ビデオライブラリについては**ここ**からご覧ください。

Table of contents

About this document **1**
Table of contents **1**
1 はじめに **4**
1.1 事前準備 4
1.1.1 ハードウェア 4
1.1.2 ソフトウェア 4
2 PSoC™ 4 with AIROC™ Bluetooth® LE リソース **5**
2.1 PSoC™ Creator 5
2.2 PSoC™ Creator ヘルプ 6
2.3 サンプル コード 7
3 PSoC™ 4 with AIROC™ Bluetooth® LE の機能 **9**
4 PSoC™の Bluetooth® LE MCU より優れている点 **11**
5 Bluetooth® LE の概要 **13**

Table of contents

5.1 Bluetooth® LE 接続の確立 14
5.2 GATT データ形式 15
5.3 Bluetooth® LE プロファイル 17
5.4 Bluetooth® LE コンポーネント 19
5.4.1 特長 19
5.4.2 コンフィギュレーション 20
6 PSoC™ 4 with AIROC™ Bluetooth® LE 開発セットアップ 23
7 初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計 25
7.1 設計について 25
7.2 事前準備 26
7.3 第 1 段階: 設計を作成 26
7.4 第 2 段階: ファームウェアを書く 39
7.4.1 システム初期化 39
7.4.2 Bluetooth® LE スタック イベント ハンドラ 41
7.4.3 Bluetooth® LE サービス固有イベント ハンドラ 44
7.4.4 メインループおよび低消費電力実装 45
7.5 第 3 段階: デバイスをプログラム 47
7.6 第 4 段階: 設計をテスト 49
7.6.1 CySmart モバイル アプリ 49
7.6.2 CySmart ホスト エミュレーション ツール 51
8 まとめ 55
9 付録 A: Bluetooth® LE デバイス ファミリの比較 56
10 付録 B : インフィニオンの専門用語 60
11 付録 C: インフィニオン Bluetooth® LE 開発ツール 61
11.1 CySmart ホスト エミュレーション ツール 62
11.2 CySmart モバイル アプリ 63
12 付録 D: PSoC™ 4 with AIROC™ Bluetooth® LE デバイス 65
12.1 Bluetooth Low Energy サブシステム (BLESS) 65
12.1.1 ディープスリープ モード 65
12.1.2 スリープ モード 65
12.1.3 アイドル モード 65
12.1.4 送信モード 66
12.1.5 受信モード 66
12.2 Arm® Cortex®-M0, メモリ, DMA 66
12.3 プログラマブルなデジタル ペリフェラル 66
12.3.1 プログラマブルな SCB 66
12.3.2 プログラマブルな TCPWM 67
12.3.3 ユニバーサル デジタル ブロック 67
12.3.4 アプリケーション 68
12.4 プログラマブルなアナログ 68
12.4.1 連続時間ブロック ミニ (CTBm) 68
12.4.2 ハードウェア シーケンサを備えた SAR ADC 68
12.4.3 低消費電力コンパレータ 69
12.4.4 静電容量タッチ センシング (CapSense™) 69
12.4.5 セグメント LCD 直接ドライバー 69
12.4.6 アプリケーション 69
12.5 システム全体のリソース 70
12.5.1 低リーク電力モード 70
12.5.2 電源と電圧の監視 71

Table of contents

12.5.3 クロック供給システム 71
12.5.4 デバイス セキュリティ 72
12.6 プログラマブルな GPIO 72
13 付録 E: Bluetooth® LE プロトコル 73
13.1 概要 73
13.2 Physical Layer (PHY) 73
13.3 Link Layer (LL) 74
13.4 Host Control Interface (HCI) 75
13.5 Logical Link Control and Adaptation Protocol (L2CAP) 75
13.6 Security Manager (SM) 75
13.7 Attribute Protocol (ATT) 76
13.7.1 アトリビュート階層 77
13.7.2 アトリビュート動作 79
13.8 Generic Attribute Profile (GATT) 80
13.9 Generic Access Profile (GAP) 80
関連アプリケーション ノート 83
改訂履歴 84

1 はじめに

サイプレスの PSoC™ 4 with AIROC™ Bluetooth® LE デバイスは、Bluetooth® Low Energy 無線ブロック、プログラマブル アナログとデジタル ペリフェラル, メモリ, および Arm® Cortex®-M0 マイクロコントローラーを単一のチップに集積したプログラマブル組み込みシステムオンチップです。Bluetooth® LE は、Bluetooth Special Interest Group (SIG) により策定された短距離通信向けの超低消費電力無線規格です。この規格は、低消費電力を実現するために設計および最適化された物理層、プロトコル スタック、アプリケーションの使用事例という要素から成り立っています。

PSoC™ 4 with AIROC™ Bluetooth® LE は、MCU と Bluetooth® LE 無線ブロックの組合せの、費用対効果の高く小さいフットプリントの代替品です。プログラマブルなアナログとデジタル サブシステムにより、PSoC™ Creator (PSoC™ 4 with AIROC™ Bluetooth® LE アプリケーションを開発するための回路図ベースの設計ツール) を使用して、設計の柔軟性および動的な微調整を実現できます。Bluetooth® LE アプリケーションを開発するには、Bluetooth® LE の複雑なプロトコル スタックの実用的な知識は必要ありません。インフィニオンは PSoC™ Creator で、プロトコルの複雑さを簡素化する、設定しやすく無償のグラフィカル ユーザー インターフェース (GUI) ベースの Bluetooth® LE コンポーネントを提供します。GUI を使用して Bluetooth® LE コンポーネントを数分で設定でき、Bluetooth® LE 設計を迅速に開始できます。PSoC™ 4 with AIROC™ Bluetooth® LE は、SRAM 内容、プログラマブルなロジック、および割り込みからウェイクアップする機能を保持しながら、150 nA の消費電流を実現します。ディープスリープ モードでは、アクティブな Bluetooth® LE 接続を維持しながら 1.3 µA を消費します。これらの低消費電力モードの組合せは、ウェアラブル フィットネス モニターやワイヤレス センサー インターフェースなどのバッテリ駆動 Bluetooth® LE 設計に対してベストインクラスのシステム消費電力を提供します。

PSoC™ 4 with AIROC™ Bluetooth® LE はバラン回路を内蔵しており、アンテナ整合に必要な外部部品数を削減し、RF 基板設計を簡素化します。

PSoC™ 4 with AIROC™ Bluetooth® LE の CapSense™という静電容量タッチ センシング機能は、比類のない信号対ノイズ比、クラス最高の耐水性、およびボタンやスライダー、トラック パッド、近接センサーなど多種多様なセンサー タイプを提供しています。CapSense™ユーザー インターフェースは、活動モニターやヘルスおよびフィットネス機器などのウェアラブル電子機器で大幅に採用されています。

本アプリケーション ノートは、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスの機能および開発ツールの概要を含む Bluetooth® LE の基礎知識について説明します。高度なアプリケーション開発については、「**Designing Bluetooth® LE Applications**」および「**Creating a Bluetooth® LE Custom Profile**」アプリケーション ノートを参照ください。

1.1 事前準備

Bluetooth® LE デザインの作成に先立ち、必ず開発キットを入手し、必要なソフトウェアをインストールしたことをご確認ください。

1.1.1 ハードウェア

- **PSoC™ 4 Bluetooth® Low Energy 準拠のパイオニアキット**
- Windows 7 以降の PC／ノートパソコン (PC 向け **CySmart ホスト エミュレーション ツール** アプリケーションを使用する場合)
- Android 5 以降または iOS 8 以降の携帯電話 (**iOS**／**Android** 向け CySmart アプリを使用する場合)

1.1.2 ソフトウェア

- **PSoC™ Creator 4.4** および **PSoC™ Programmer 3.29**
- **CySmart ホスト エミュレーション ツール**または **iOS**／**Android** 向け CySmart アプリ

2 PSoC™ 4 with AIROC™ Bluetooth® LE リソース

インフィニオンは、**www.cypress.com** に大量のデータを掲載しており、ユーザーがデザインに適切な PSoC™デバイスを選択し、迅速かつ効率的にデザインに統合する手助けをしています。PSoC™の初心者のお客様は、「**付録 B : インフィニオンの専門用語**」に記載されている一般的に使用する用語一覧をご覧になるようお勧めします。リソースの包括的なリストについては、「**KBA86521, How to Design with PSoC™ 3, PSoC™ 4, and PSoC™ 5LP**」を参照ください。

以下は PSoC™ 4 with AIROC™ Bluetooth® LE のリソースの要約です。

- **概要: PSoC™ポートフォリオ**、**PSoC™ RF ロードマップ**
- **製品セレクタ: PSoC™ 1**、**PSoC™ 3**、**PSoC™ 4**、**PSoC™ 5LP**。さらに、**PSoC™ Creator** にはデバイス選択ツールが含まれています。
- **データシート**は **PSoC™ 4XX7-BL** および **PSoC™ 4XX8-BL** デバイス ファミリの電気的仕様を説明します。
- **アプリケーション ノートおよびサンプル コード**は基本的なレベルから高度なレベルまでの幅広いトピックに触れています。アプリケーション ノートの多くはサンプル コードを含んでいます。PSoC™ Creator は追加のサンプル コードをご提供します。詳細は、「**サンプル コード**」を参照ください。
- **テクニカル リファレンス マニュアル (TRM)** は各 PSoC™ 4 with AIROC™ Bluetooth® LE デバイス ファミリのアーキテクチャとレジスタの詳細な説明を提供します。
- **CapSense™デザイン ガイド:** PSoC™ 4 with AIROC™ Bluetooth® LE ファミリのデバイスを使用して静電容量タッチセンシング アプリケーションを設計する方法について説明します。
- **開発ツール**
  - **CY8CKIT-042-BLE Bluetooth® Low Energy Pioneer Kit** は使いやすくて安価な Bluetooth® LE 向けのプラットフォームです。このキットは、Arduino™準拠シールドおよび Digilent® Pmod™ドーターカード用コネクタを搭載します。
  - **Windows**、**iOS**、および **Android** 用 CySmart Bluetooth® LE ホスト エミュレーション ツールはご使用の Bluetooth® LE ペリフェラル アプリケーションをテストおよびデバッグするための使い勝手の良いアプリです。

その概要については、「**付録 C: インフィニオン Bluetooth® LE 開発ツール**」を参照ください。

- **トレーニング ビデオ:** 1 つのビデオは千もの言葉に匹敵するといわれています。始めて Bluetooth® LE 設計を作成する際に、以下の PSoC™ 4 with AIROC™ Bluetooth® LE ビデオをご覧になるようお勧めします。
  - **PSoC™ 4 with AIROC™ Bluetooth® LE のご紹介**
  - **PSoC™ Creator 入門**
  - **PSoC™ 4 Bluetooth® Low Energy Compliant Pioneer Kit**
  - **Bluetooth® Low Energy のご紹介**
  - **最初の Bluetooth® Low Energy 設計の設定**
  - **最初の Bluetooth® Low Energy 設計のファームウェアを書く**
- 設計、互換性、機能、タイミング、データシート、および一般的なアプリケーションのサポートに関連する**テクニカルサポート**ページの質問や関連項目にアクセスしてください。

2.1 PSoC™ Creator

PSoC™ Creator は無償の Windows ベースの統合設計環境 (IDE) です。このツールにより、お客様は PSoC™ 4 with AIROC™ Bluetooth® LE、PRoC-BLE を基にして、ハードウェアとファームウェアを同時に設計できます。**Figure 1** に示すように、PSoC™ Creator を使用すれば、以下のことを実現できます。

1. コンポーネントをドラッグ＆ドロップして、メイン デザイン ワークスペースでハードウェア システム デザインを構築
2. アプリケーションのファームウェアと PSoC™ ハードウェアを相互設計
3. コンフィギュレーション ツールを用いてコンポーネントを設定
4. 100 以上のコンポーネントを含むライブラリを利用
5. コンポーネント データシートをレビュー

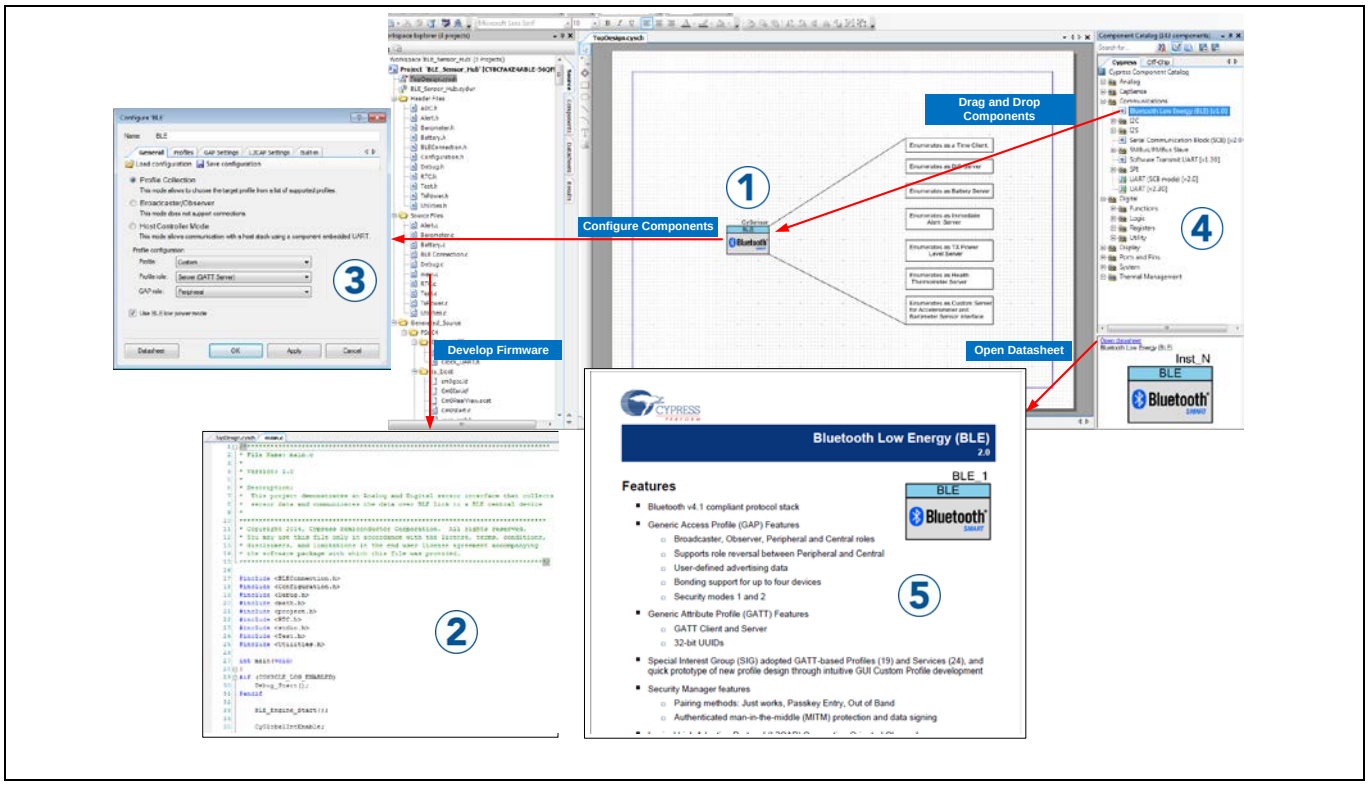


Figure 1　PSoC™ Creator の回路図エントリ入力コンポーネント

2.2　PSoC™ Creator ヘルプ

PSoC™ Creator ホームページへアクセスし、PSoC™ Creator の最新版をダウンロードしてインストールしてください。次に、PSoC™ Creator を起動して、以下の項目を開きます。

- **Quick Start Guide** (クイック スタート ガイド): **Help** > **Documentation** > **Quick Start Guide** を選択します。このガイドは PSoC™ Creator プロジェクトを開発するための基礎知識を提供します。
- **Simple Component Code Examples** (シンプルなコンポーネントのサンプル コード): **File** > **Code Example** を選択します。これらのサンプル コードは、PSoC™ Creator のコンポーネントの設定と使用方法を示します。
- **System Reference Guide** (システム リファレンス ガイド): **Help** > **System Reference Guide** を選択します。このガイドは、PSoC™ Creator が提供するシステム機能を記載し説明します。
- **Component datasheets** (コンポーネント データシート): コンポーネントを右クリックして「Open Datasheet」を選択します。すべての PSoC™ 4 with AIROC™ Bluetooth® LE のコンポーネント データシートの一覧を表示するには、**PSoC™ 4 with AIROC™ Bluetooth® LE コンポーネント データシート**のページへアクセスしてください。

2.3 サンプル コード

PSoC™ Creator は多数のサンプル コードを提供しています。これらのプロジェクトは、**Figure 2** に示すように、PSoC™ Creator のスタート ページからアクセスできます。

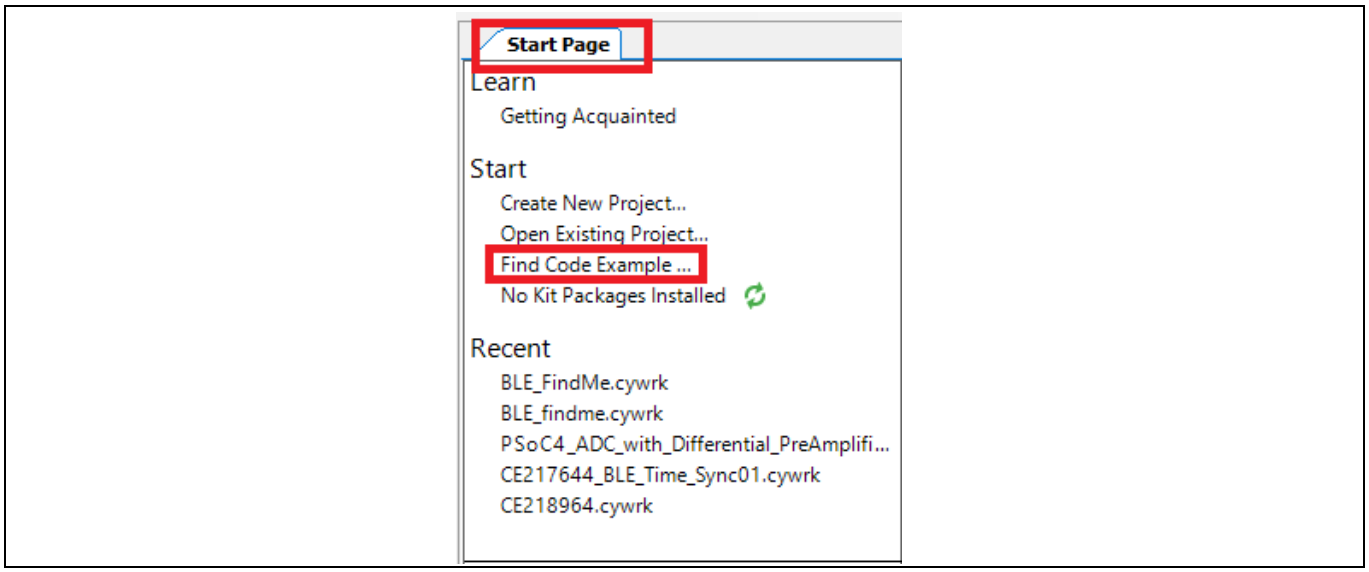


Figure 2 PSoC™ Creator 内のサンプル コード

サンプル コードにより、空のページの代わりに完成した設計から始めることで設計時間を短縮させることができます。サンプル コードはまた、PSoC™ Creator コンポーネントを様々なアプリケーションに使用する方法を示します。

Figure 3 に示す **Find Example Project** ダイアログにはいくつかのオプションがあります。

- アーキテクチャ、デバイス ファミリ (PSoC™ 4、PSoC™ 4 with AIROC™ Bluetooth® LE、PRoC-BLE など)、カテゴリまたはキーワードに基づいてサンプル コードを検索します。
- Filter Options に基づいて提供されたサンプル コードのメニューから選択します。**Figure 3** に示すように、ご参考のための 30 以上の Bluetooth® LE サンプル コードがあります。
- 選択したドキュメントを表示します (**Documentation** タブ)
- 選択したサンプル コードをレビューします。このウィンドウからコードをプロジェクトにコピー & ペーストして、コード開発時間を短縮させることができます。
- さらに、選択に応じて新規プロジェクト (また、必要な場合は新規ワークスペース) を作成できます。完成した基本的な設計から始めることで設計時間を短縮させます。その設計をアプリケーションに適用できます。
- PSoC™ Creator サンプル コードの他には、この GitHub リポジトリで多くの Bluetooth® LE の参考例を見つけられます。

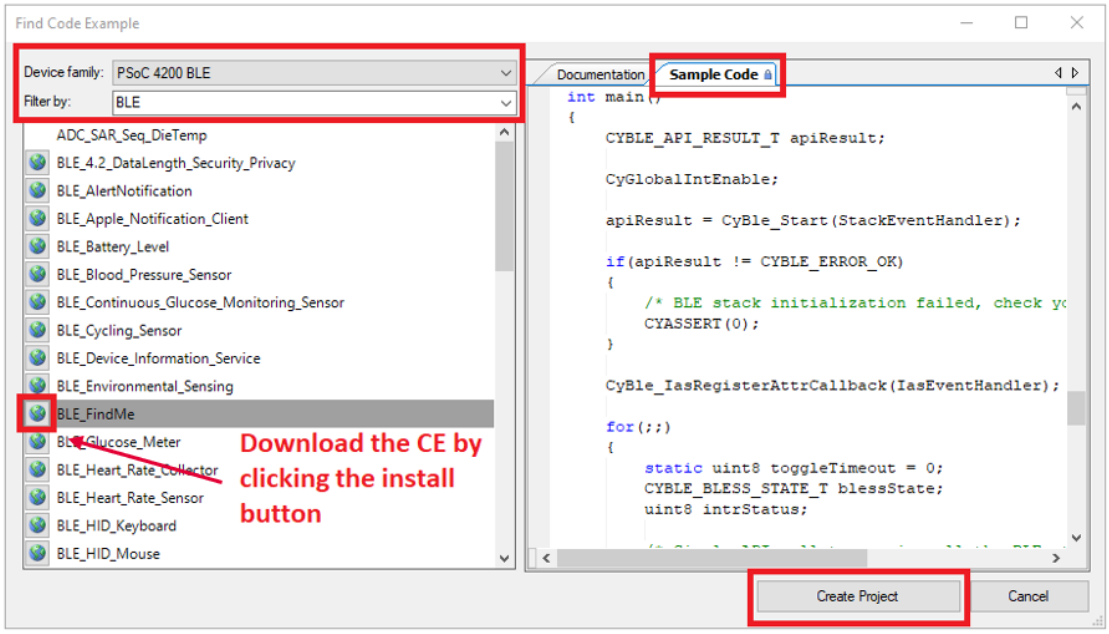


Figure 3　コード例およびサンプル コード

3 PSoC™ 4 with AIROC™ Bluetooth® LE の機能

Figure 4 に示すように、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスの主な特長は以下の通りです。

- **32 ビット MCU サブシステム**
  - 48 MHz Arm® Cortex®-M0 CPU
  - 最大 256 KB フラッシュおよび 32 KB SRAM
- **SmartSense™自動チューニングを備えた CapSense™**
  - タッチパッド機能を備えたインフィニオン Capacitive Sigma-Delta™ (CSD) コントローラー1 個
- **プログラマブルなアナログ フロント エンド (AFE)**
  - PGA やコンパレータ、フィルターなどに設定可能なオペアンプ 4 個
  - 12 ビットの 1 Msps SAR2 ADC 1 個
- **プログラマブル デジタル ロジック**
  - ユニバーサル デジタル ブロック (UDB) 4 個: カスタム デジタル ペリフェラル
  - 設定可能な TCPWM ブロック 4 個: 16 ビット タイマー、カウンターまたは PWM
  - 設定可能なシリアル通信ブロック (SCB) 2 個: I2C マスター／スレーブ、SPI マスター／スレーブ、または UART
- **Bluetooth 4.2 準拠の Bluetooth Smart 接続に対応**
  - 集積バランを備えた 2.4 GHz Bluetooth® LE 無線通信
  - Bluetooth 4.2 仕様に準拠したコントローラーとホスト実装

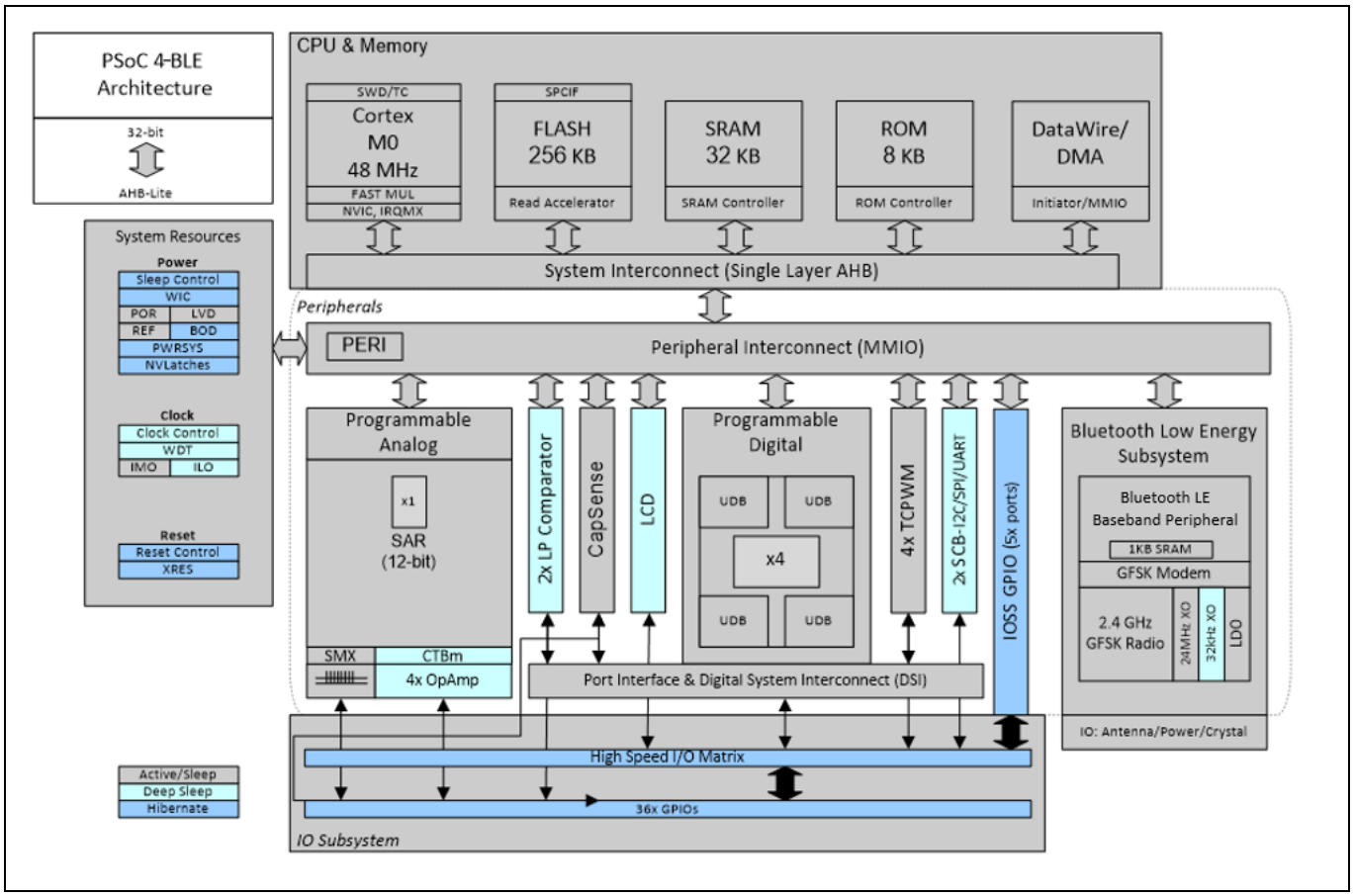


Figure 4 PSoC™ 4 with AIROC™ Bluetooth® LE アーキテクチャ (CY8C4248-BL)

PSoC™ 4 with AIROC™ Bluetooth® LE の機能

各機能の簡単な説明については、「**付録 D: PSoC™ 4 with AIROC™ Bluetooth® LE デバイス**」を参照ください。詳細な情報については、PSoC™ 4 with AIROC™ Bluetooth® LE ファミリの**データシート**, **TRM**, および**アプリケーション ノート**を参照ください。**Figure 4** は CY8C4248-BL デバイス ファミリの特長を示します。これらの機能のサブセットは他のデバイス ファミリで使用可能です。**Table 2** を参照ください。

4 PSoC™の Bluetooth® LE MCU より優れている点

Figure 5 に示すように、標準的な MCU は CPU (8051 または Arm® Cortex®など)、ADC や DAC, UART, SPI, Bluetooth® LE などのペリフェラル機能, および汎用 I/O を含みます。これらはすべて CPU のレジスタ インターフェースにリンクされています。MCU の中で、CPU はデバイスの「心臓」です。CPU はセットアップ, データ移動, タイミングなどすべての処理を管理します。CPU がないと、MCU は動作不可能です。

Figure 6 に示すように、PSoC™はかなり異なります。PSoC™では、CPU、アナログ、デジタルおよび I/O は、プログラマブルなシステムで同等に重要なリソースです。 PSoC™の心臓は CPU ではなく、システムの相互接続とプログラマビリティです。ペリフェラル アナログとデジタルは高度に設定可能な信号マトリックスおよびデータ バス メッシュを介して相互接続しています。これにより、ユーザーはアプリケーションの要件を満たすカスタム設計を作成できます。PSoC™をプログラムして MCU をエミュレートできますが、MCU をプログラムして PSoC™をエミュレートできません。

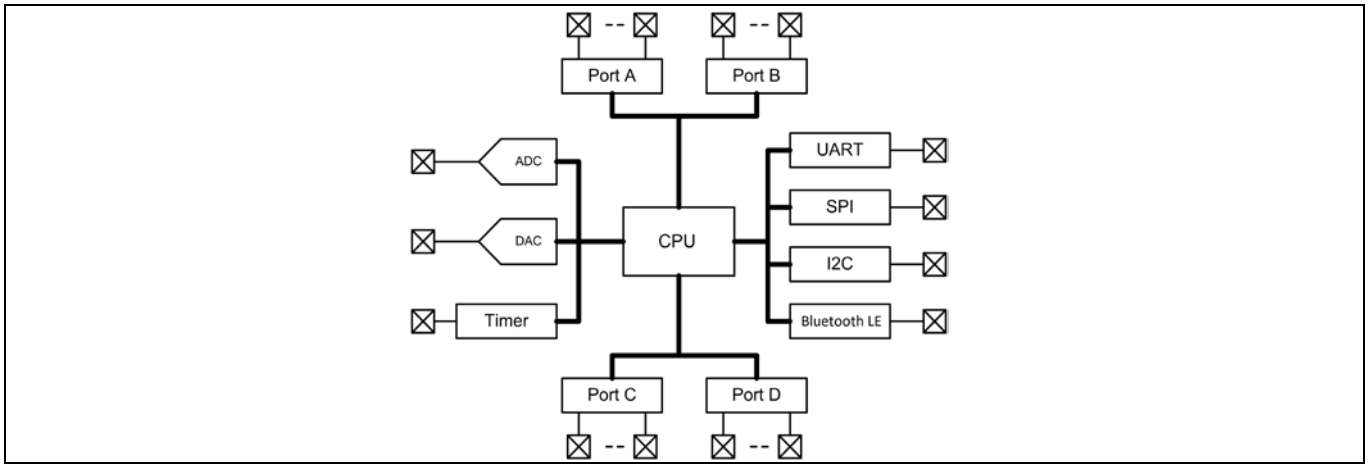


Figure 5 標準 Bluetooth® LE MCU のブロック図

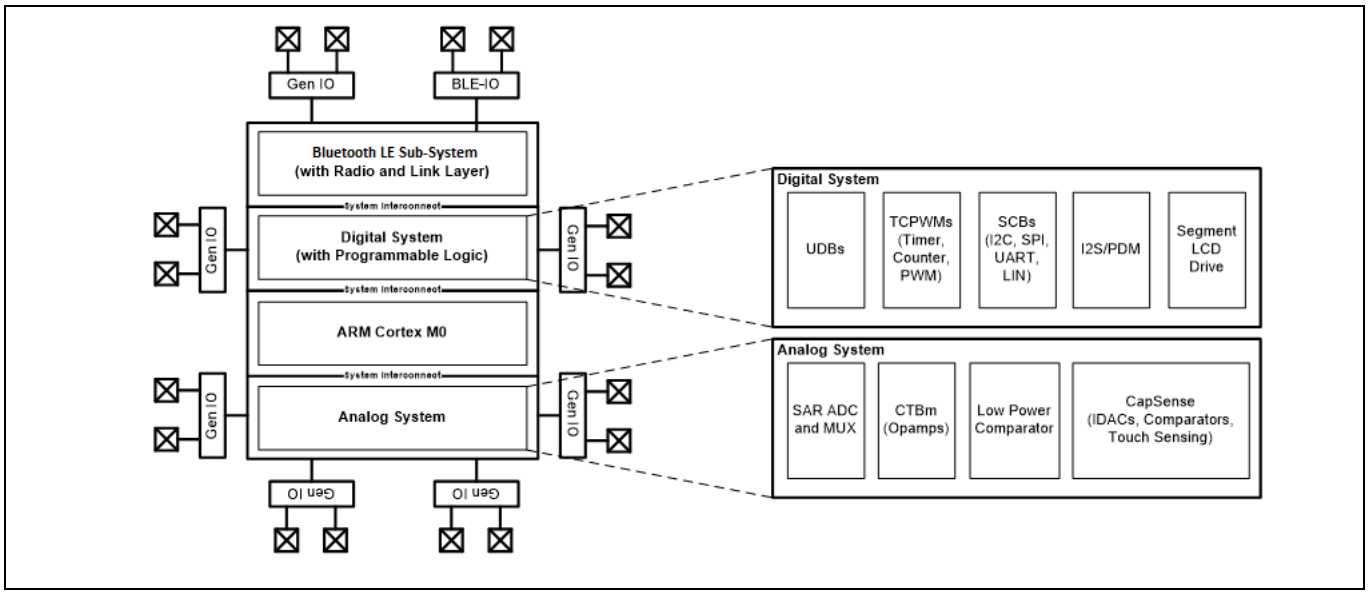


Figure 6 PSoC™ 4 with AIROC™ Bluetooth® LE のブロック図

標準的な MCU は、ステートマシンを処理したり、タイミングにタイマーを使用したり、出力ピンを駆動したりするには、CPU ファームウェアを必要とします。従って、機能的なパスはほとんどの場合、

PSoC™の Bluetooth® LE MCU より優れている点

CPU を介します。しかし、PSoC™では非同期の平行処理が可能です。PSoC™を設定して CPU から独立して動作する要素を取得できます。PSoC™ Creator は、アナログ ブロック、デジタル ブロックとプログラマブルな相互接続ブロック間のシームレスなインターフェースを作成します。また PSoC™ Creator で、すべてのアナログとデジタル ペリフェラルはコンポーネントと呼ばれる仮想 IC の形で提供されています。本アプリケーション ノートのサンプル プロジェクトでは、PSoC™ Creator およびそのコンポーネントを使用する方法を説明します。

PSoC™には、ユニバーサル デジタル ブロック (UDB) として知られるプログラム可能なデジタル ブロックもあります。PSoC™ Creator はまた、UART, SPI, I^2S, タイマー, PWM, カウンター, デジタル ゲート (AND, OR, NOT, XOR) など、UDB から成る様々なコンポーネントを提供しています。PSoC™ Creator で UDB を使用して、カスタム ステートマシンおよびデジタル ロジックを作り上げることが可能です。カスタム PSoC™ Creator コンポーネントを作り上げる方法については、「**PSoC™ Creator Component Author Guide**」を参照ください。

5 Bluetooth® LE の概要

Bluetooth® LE、すなわち Bluetooth Smart™ は、Bluetooth SIG により策定された、低消費電力、短距離、低データ レートの無線通信プロトコルです。**Figure 7** に示すように、Bluetooth® LE は、低消費電力で小さなデータ量を効率的に転送するために設計された階層式プロトコル スタックを有するもので、バッテリ式デバイス向けの推奨の無線プロトコルとなります。

Bluetooth® LE スタックは以下の要素から成ります。

- Physical Layer (PHY): 2.4 GHz RF、データ レートが 1 Mbps
- Link Layer (LL): PHY のタイミングおよびパケット形式を定義
- Host Control Interface (HCI): スタックのハードウェア コントローラー (PHY + LL) の層とファームウェア ホスト層をリンクさせる
- Logical Link Control and Adaptation Protocol (L2CAP): パケット アセンブリ／ディスアセンブリおよびプロトコル マルチプレクサ層として動作
- Attribute Protocol (ATT): アプリケーション データの構成とアクセス方法を定義
- Security Manager (SM): Bluetooth® LE 接続を介した安全なデータ転送を行うためのツールボックス
- Generic Attribute Profile (GATT): ATT 層により定義されたデータにアクセスする方式を定義
- Generic Access Profile (GAP): デバイスが Bluetooth® LE リンク マスターかスレーブかを決定し、それに応じて下層を設定するアプリケーション指向インターフェース
- Bluetooth® LE プロトコルの詳細説明については、「**付録 E: Bluetooth® LE プロトコル**」を参照ください。

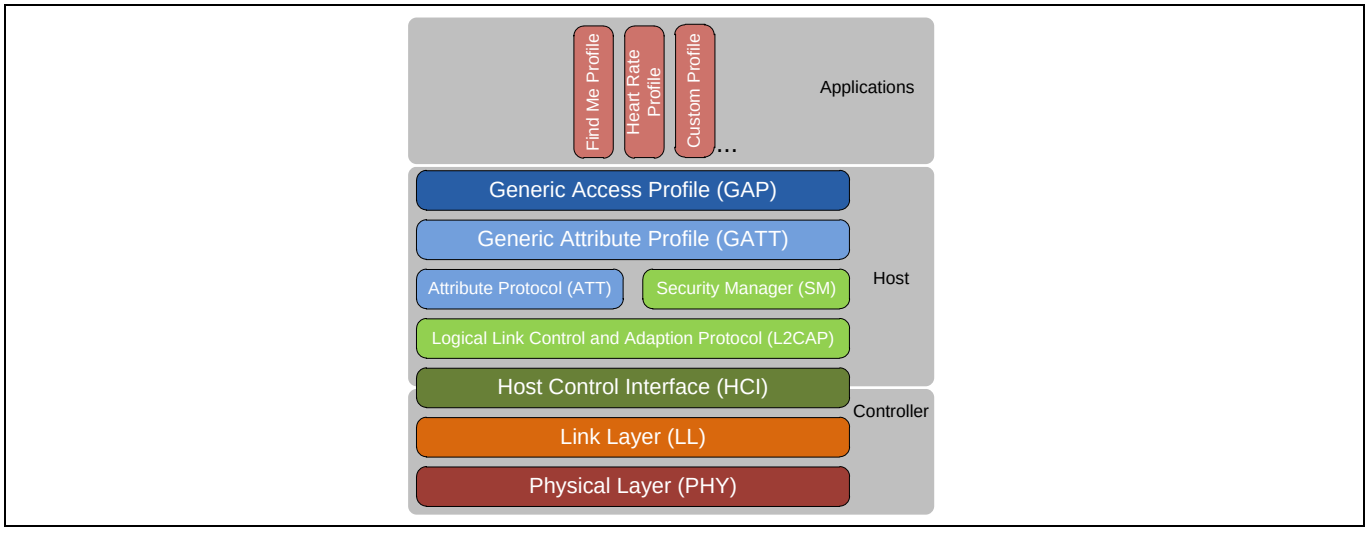


Figure 7 Bluetooth® LE プロトコル スタック

Bluetooth® LE アプリケーションを開発するには、この複雑なプロトコル スタックの実用的な知識は必要ありません。インフィニオンは PSoC™ Creator で、プロトコルの複雑さを簡素化する、設定しやすい GUI ベースの Bluetooth® LE コンポーネントを提供します。Bluetooth® LE を使い始めるにあたって、以下を理解すれば充分です。

- Bluetooth® LE 接続の確立手順
- アプリケーション データの表示と抽象化
- アプリケーション要件の Bluetooth® LE GAP と GATT 層コンフィギュレーションへのマッピング

5.1 Bluetooth® LE 接続の確立

Bluetooth® LE Pinoneer Kit とスマートフォンのような 2 つのデバイス間の Bluetooth® LE 接続を確立するために、Figure 8 に示すように、2 つの Generic Access Profile (GAP) デバイス ロールを理解する必要があります。

- **GAP ペリフェラル:** 自分の存在を宣伝し、GAP セントラル デバイスからの接続を許可するデバイスです。心拍数測定機能を実装する Bluetooth® LE Pioneer Kit が GAP ペリフェラル デバイスの一例です。
- GAP セントラル: GAP ペリフェラルからの宣伝をスキャンし、それらのペリフェラルとの接続を確立するデバイスです。心拍数測定機器に接続するスマートフォンが GAP セントラル デバイスの一例です。

セントラル デバイスがペリフェラルとの接続を確立した後、両方のデバイスは Bluetooth® LE 接続を介して接続されていると言えます。Bluetooth® LE 通信が接続されると、Generic Attribute Profile (GATT) は Figure 8 に示すように、GAP のロールとは関係なく、データの転送元と転送先に基づいて以下の 2 つのプロファイル ロールを定義します。

- **GATT サーバー:** データまたは状態情報を格納するデバイスです。GATT クライアントによって設定されると、GATT クライアントにデータを送信するか、または自分のローカル状態を変更します。例えば、心拍数測定機器は、GATT クライアントであるスマートフォンに心拍数データを送信する GATT サーバーです。同様に、スマート電球は GATT クライアントであるスマート スイッチによって設定される電球の状態 (オン／オフ) を格納する GATT サーバーです。
- **GATT クライアント:** GATT サーバーの状態を設定するか、または GATT サーバーからデータを受信するデバイスです。例えば、心拍数測定機器から心拍数情報を受信するスマートフォンは GATT クライアントです。

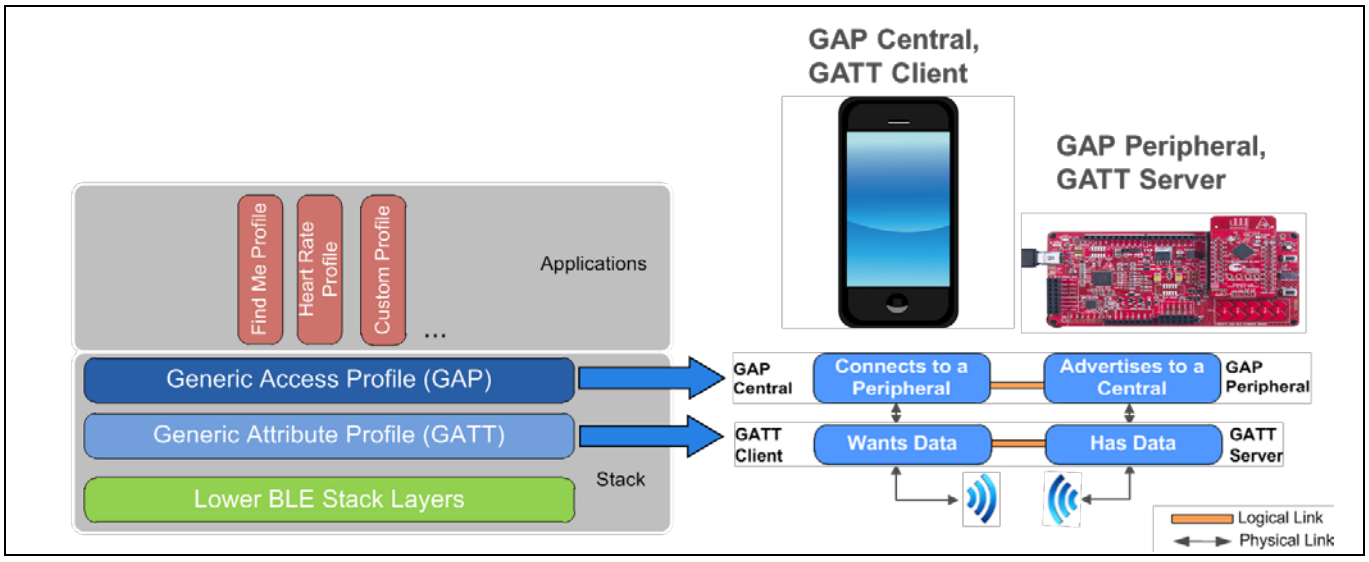


Figure 8 Bluetooth® LE アプリケーションの概要

Bluetooth® LE 接続が確立された後、GATT クライアントは GATT サーバーに存在するすべてのデータおよび状態を検出します。データを検出すると、GATT クライアントは GATT サーバーを設定する、および／または GATT サーバーのデータや状態を読み書きできます。

5.2 GATT データ形式

Bluetooth® LE GAP／GATT ロールを理解した後、次のステップは、データを GATT サーバーに格納する方法、および GATT クライアントがデータを取り込む方法を理解することです。GATT サーバーはアトリビュート, キャラクタリスティック, およびサービスを使用して Bluetooth® LE デバイスのデータを表示および抽象化します。本節で説明するように、サービスは 1 つ以上のキャラクタリスティックを含み、各キャラクタリスティックは実際のデータを含む複数のアトリビュートで構成されています。

- **アトリビュート:** GATT 層の基本的なデータ コンテナであり、一件の個別情報を表します。**Figure 9** に示すように、アトリビュートは以下の要素から構成されています。
  - アトリビュート ハンドル: アトリビュートをアドレス指定するために使用します。
  - アトリビュート タイプ: Bluetooth SIG が割り当てる 16 ビットの UUID (Universally Unique Identifier) であり、アトリビュートに格納されているデータを定義します。
  - アトリビュート バリュー: 実際のデータを格納します。
  - アトリビュート パーミッション: アトリビュートの読み出し／書き込みのパーミッション、およびセキュリティ要件を定義します。

心拍数測定、電池残量、電池残量単位、デバイス名などがアトリビュートの例です。

GATT サーバーには、「アトリビュート データベース」と呼ばれるデータベースに格納される数多くのアトリビュートが含まれます。GATT クライアントは、アトリビュート ハンドルを使用して GATT サーバーのアトリビュート データベースから 1 つ以上のアトリビュートを読み書きします。ユーザーとして、読み出し／書き込み動作を実行するアトリビュート ハンドルを知れば十分です。他の詳細は、インフィニオンの Bluetooth® LE コンポーネント API 関数によって抽象化されています。

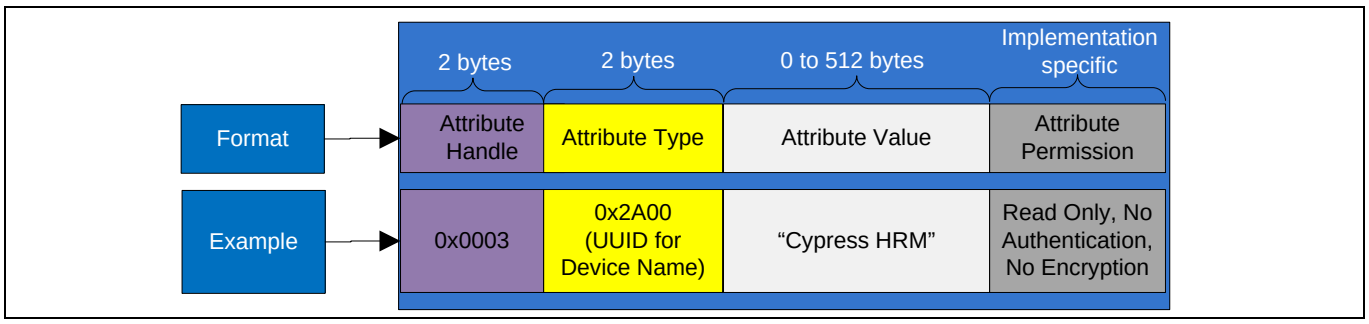


Figure 9 GATT アトリビュートの例

- **キャラクタリスティック:** キャラクタリスティックは、複数の個別アトリビュートで構成されています。これらのアトリビュートを組み合わせるとシステム情報や意味のあるデータを形成します。キャラクタリスティックは **Figure 10** に示すように、1 つのキャラクタリスティック デクレレーション アトリビュート、1 つのキャラクタリスティック バリュー アトリビュート、および 1 つ以上 (任意) のキャラクタリスティック ディスクリプタ アトリビュートで構成されています。例えば、「90」の電池残量のアトリビュートと、「%」の電池残量ディスクリプタのアトリビュートを組み合わせると、90%のシステムの電池残量情報を作ります。

Bluetooth SIG は、開発アプリケーションに有用な定義済みの**標準キャラクタリスティック**一式を提供します。また、独自のカスタム キャラクタリスティックを定義することもできます。スマート電球の「オン／オフ」状態はカスタム キャラクタリスティックの一例です。

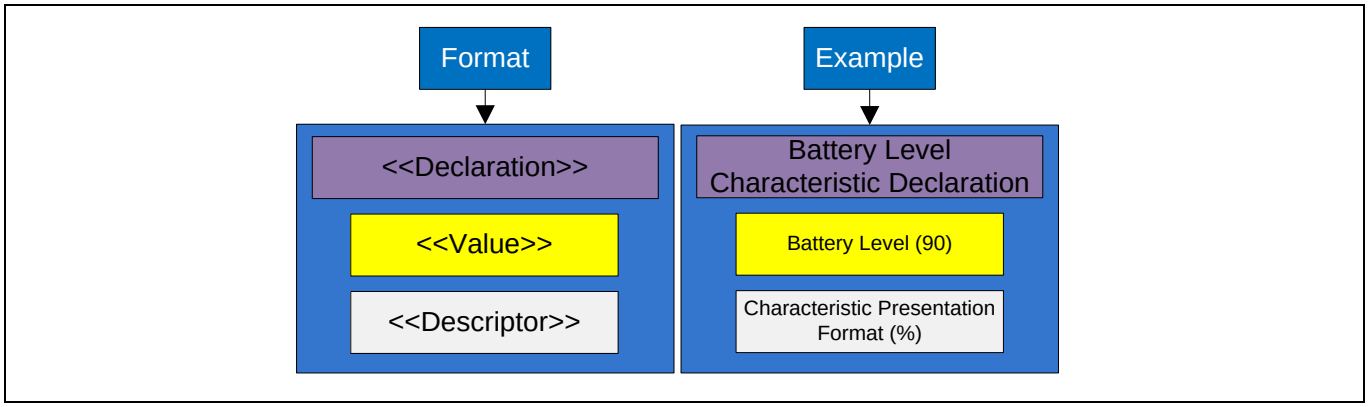


Figure 10　GATT キャラクタリスティックの例

- **サービス:** サービスは、デバイスの特定の機能または特長を定義する 1 つ以上の関連するキャラクタリスティックで構成されています。**Figure 11** に、心拍数測定に関連する情報を説明した 3 つのキャラクタリスティックの例 (心拍数サービス) を示します。

Bluetooth SIG は、Bluetooth® LE デバイスの一般的な機能を実装するための定義済み**標準サービス**一式を提供しています。また、標準またはカスタム キャラクタリスティックから成る独自のカスタム サービスを定義することもできます。スマート電球サービスは電球の「オン／オフ」状態キャラクタリスティックを含むカスタム サービスの一例です。

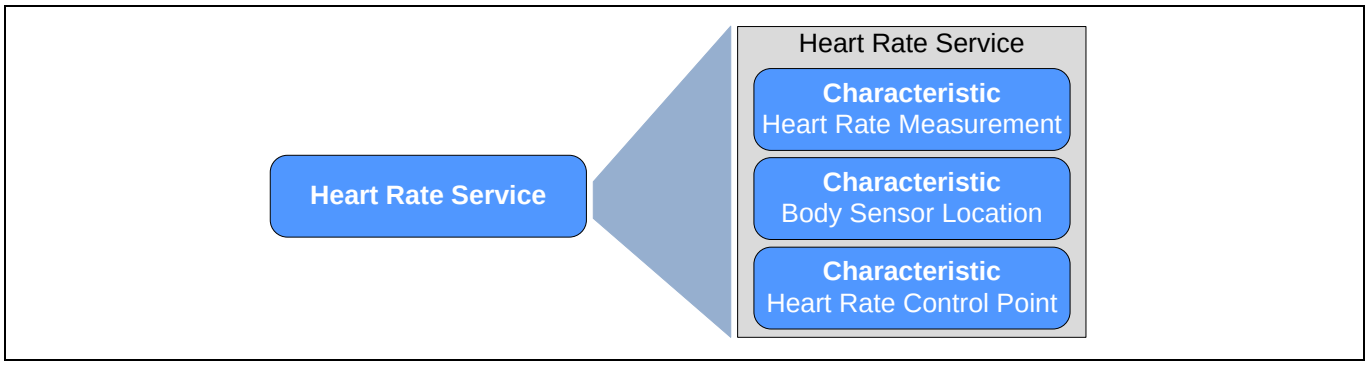


Figure 11　GATT サービスの例

GATT 層はデータ フォーマットに加えて、GATT 層のデータを検出し、アクセスする一連の手順や方式も定義します。Bluetooth® LE 接続の確立後、GATT クライアントはこれらの GATT 手順を使用して、接続された GATT サーバーの完全なアトリビュート データベースを検出します。検出が完了したら、GATT クライアントはキャラクタリスティック読み出し／書き込み GATT 手順で、アトリビュート ハンドルを使用してアトリビュートへの読み出し／書き込み動作を実行します。

Figure 12 に、心拍数センサーで心拍数データをモデル化する方法、およびスマートフォンが心拍数データを取得する GATT 手順を示します。普通の心拍数センサーは、宣伝してスマートフォンのような GAP セントラル デバイスに接続する GAP ペリフェラル ロールを実装します。接続の確立後、心拍数センサー デバイスは心拍数データを格納する GATT サーバーになります。GATT サーバーは、実際に測定された心拍数値を心拍数バリュー アトリビュートの形式で格納する心拍数測定キャラクタリスティックを含む心拍数サービスをサポートします。

スマートフォン側では、接続の確立後、スマートフォンでの GATT クライアントはまず、GATT 検出の手順で接続された GATT サーバーがサポートするすべてのサービス、キャラクタリスティック、アトリビ

ュートを検出します。検出が完了した後、GATT クライアントであるスマートフォンは GATT キャラクタリスティックの手順で、GATT サーバーからの心拍数データのプッシュ通知を有効にします。

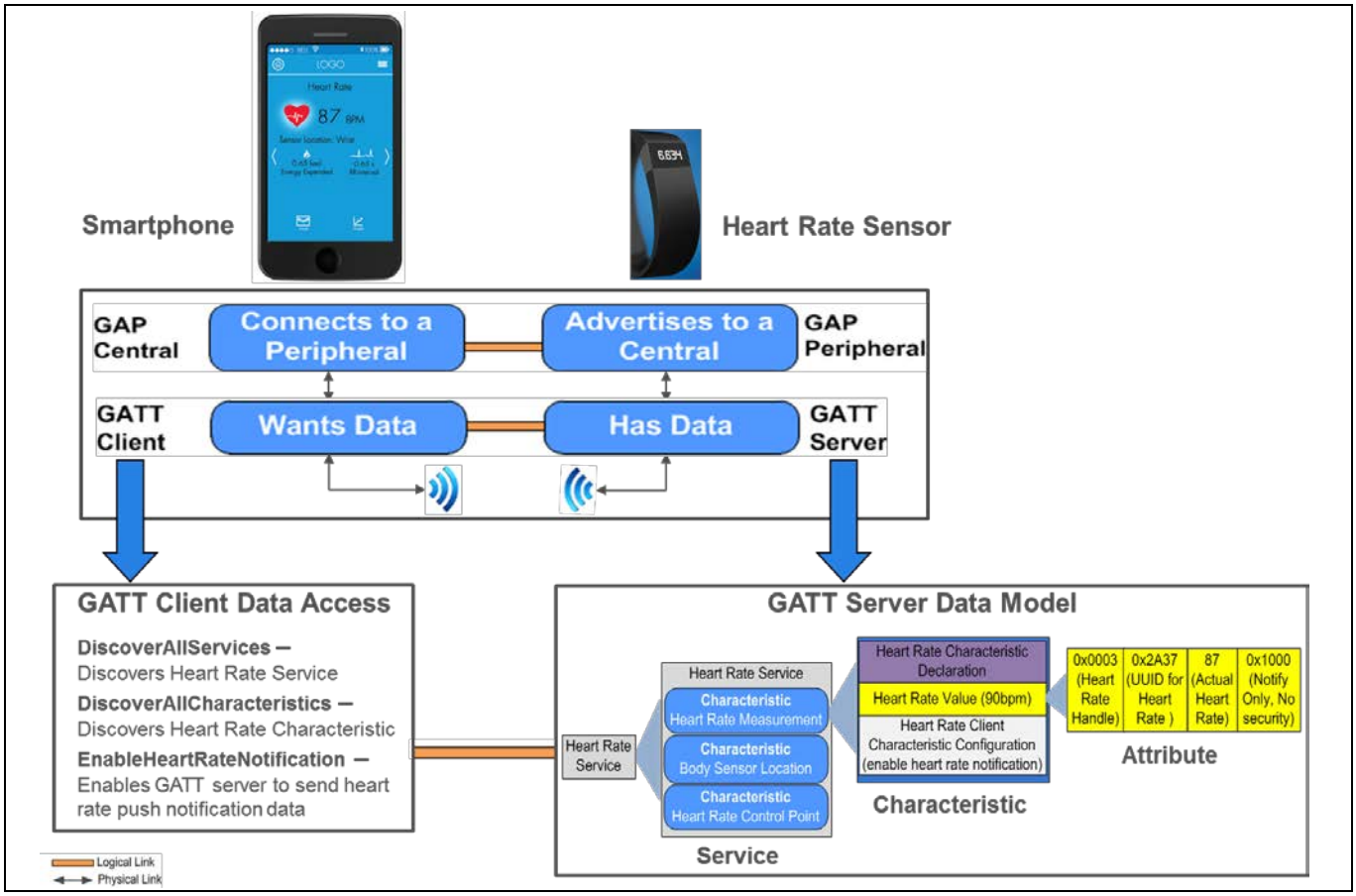


Figure 12　GATT データ形式

5.3　Bluetooth® LE プロファイル

Bluetooth® LE プロファイルは、プロファイル準拠デバイス間のアプリケーション レベルでの相互運用性を確保するための仕様です。これは、特定のエンド アプリケーションやユースケースを作成するための異なる Bluetooth® LE 層のロールとコンフィギュレーションおよびサポートされる GATT サービスを定義します。例えば、心拍数監視機器の場合、Bluetooth® LE 心拍数プロファイルは、相互運用可能な心拍数監視機器を作成するために必要な GAP と GATT ロール、および心拍数監視機器がサポートする GATT サービスを定義します。Bluetooth SIG は、一般的な Bluetooth® LE エンド アプリケーション用の定義済み標準プロファイル一式を提供しています。また、標準またはカスタム サービスから成る独自のカスタム プロファイルの作成もできます。

Figure 13 に示すように、プロファイルは GATT 層と同様に、2 つのアプリケーション ロールを定義します。

- **センサーまたはサーバー:** センサー プロファイル ロールはデータを持っているアプリケーションによってサポートされます。センサー プロファイル仕様は、センサー アプリケーションのユースケースに対応するために必要な Bluetooth® LE デバイスのロール (例えば、GATT／GAP ロール) および動作 (例えば、宣伝間隔、サポートされる GATT サービス) を定義します。センサー プロファイル仕様に従うと、対応するコレクター プロフィールを実装する他のすべてのデバイスとの相互運用性を保証します。例えば、心拍数センサー プロファイルを実装する心拍数監視デバイスは、心拍数コレクター プロファイルを実装するすべてのスマートフォンと相互運用可能です。

- **コレクターまたはクライアント:** コレクター プロファイル ロールはデータを要求するアプリケーションによってサポートされます。コレクター プロファイル仕様は、対応するセンサー プロファイル仕様を実装するあらゆるデバイスと相互運用し、それらから情報を収集するために必要な Bluetooth® LE デバイス ロードと動作を定義します。

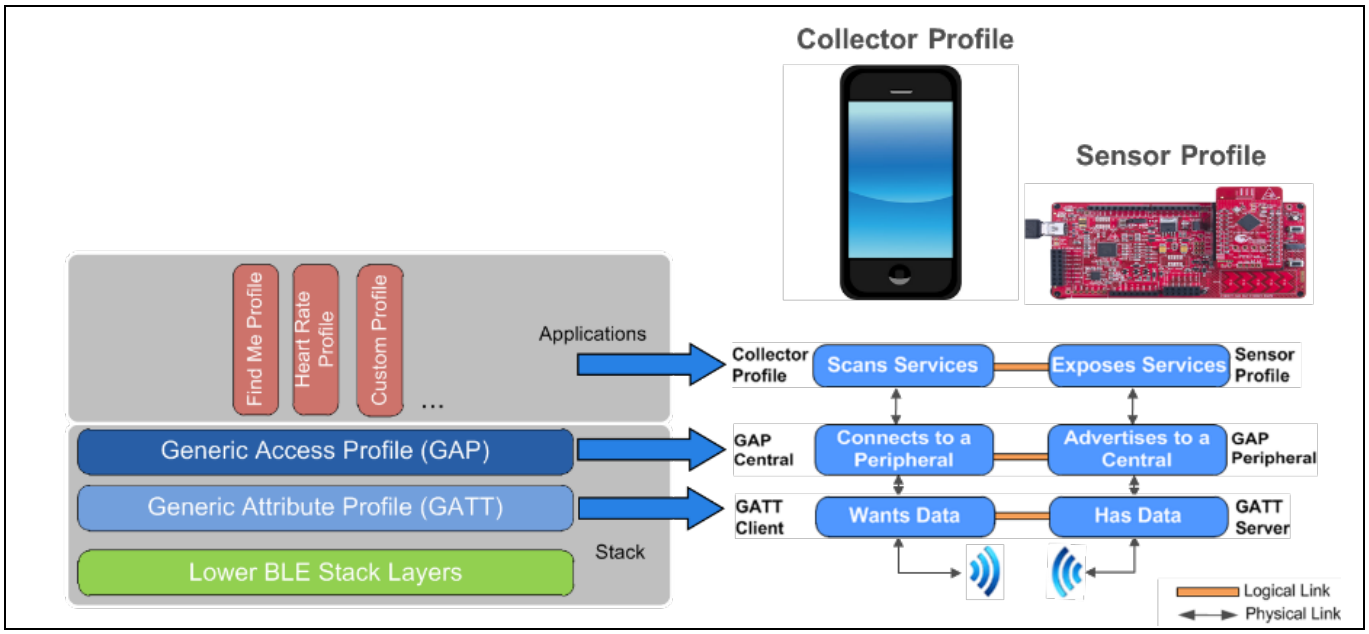


Figure 13 Bluetooth® LE プロファイルの例

Bluetooth® LE の概要

Figure 14 に、Bluetooth® LE デバイスにおけるデータ抽象化とデータ階層の概要を示します。

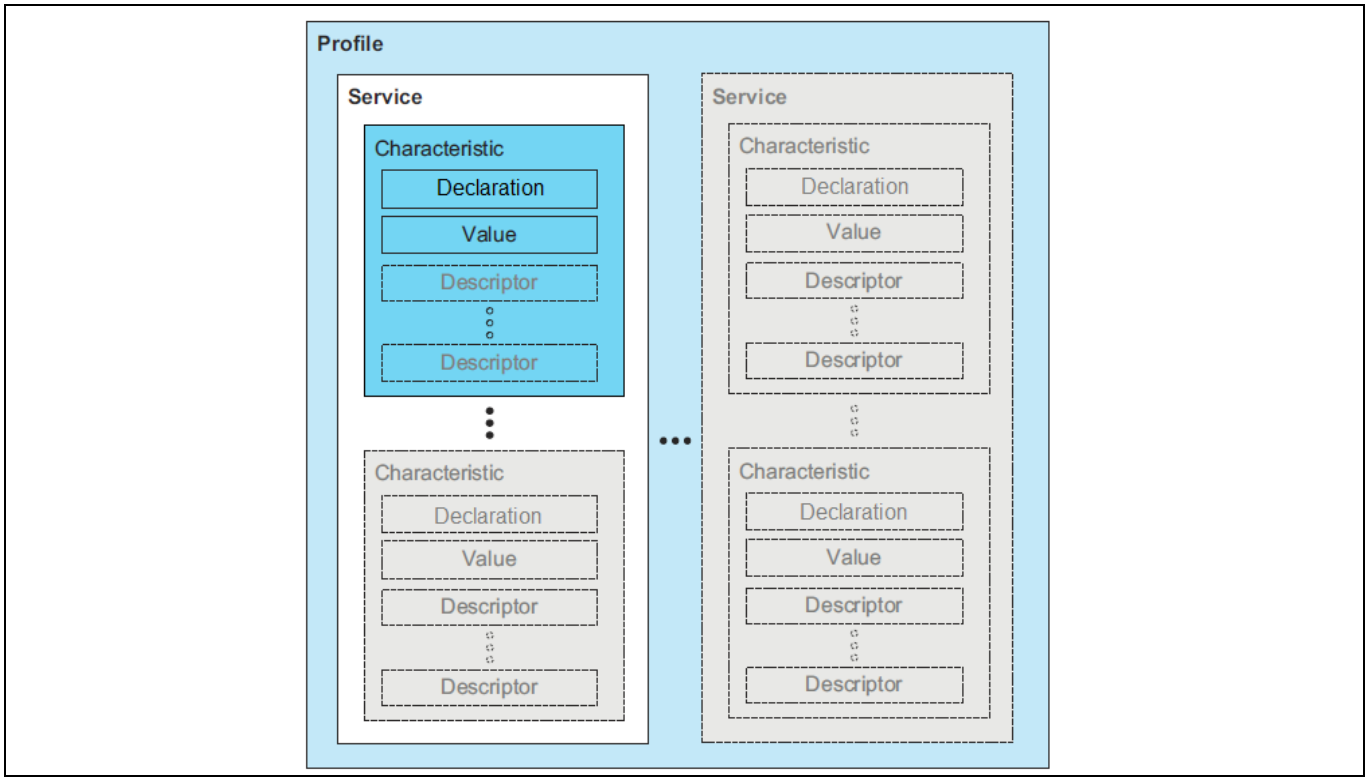


Figure 14 Bluetooth® LE データ階層*

*Bluetooth SIG により提供された無料イメージ

5.4 Bluetooth® LE コンポーネント

5.4.1 特長

PSoC™ Creator の Bluetooth® LE コンポーネントは、Bluetooth® LE プロトコルをシンプルかつ使いやすい GUI およびわずか数関数の API に簡素化します。Bluetooth® LE コンポーネントの特長は以下のとおりです。

- Bluetooth 4.2 準拠のプロトコル スタック
- すべての GAP ロール (セントラル、ペリフェラル、ブロードキャスター、オブザーバー) に対応。限定された GAP の同時組合せ (セントラルとオブザーバー、ペリフェラルとブロードキャスターなど) にも対応
- Bluetooth SIG によって採用されたすべての GATT ベース プロファイルおよびサービスに対応 (プロファイルごとに少なくとも 1 サンプル コードがある)
- Bluetooth® LE コンポーネント GUI によりカスタム プロファイルの作成および使用を簡素化
- L2CAP 接続指向チャネル、Link Layer の低デューティ サイクル宣伝、および LE ping 機能に対応
- 最大 4 個のボンディングされたデバイスおよび 8 個のデバイス ホワイトリスト フィルターに対応
- すべての Bluetooth 4.2 セキュリティ モードに対応
- Bluetooth 4.2 仕様の Link Layer データ長拡張、Link Layer プライバシー、および LE セキュア接続機能に対応

詳細については、Bluetooth® LE コンポーネントのデータシートを参照ください。

5.4.2 コンフィギュレーション

Bluetooth® LE コンポーネントを使用し、Bluetooth SIG によって定義されたプロファイル (標準プロファイル) ベース アプリケーションを作成するには、5 つのステップがあります。例えば、「**初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計**」節で記述されている Find Me Target 例を作成するために、以下のステップに従ってください。

ステップ 1: ご使用の設計に適切なプロファイルとプロファイル ロールを選択します。Bluetooth® LE コンポーネントは、選択したプロファイル ロールに必要な GAP と GATT ロールを自動的に選択します。**Figure 15** に示すように、Bluetooth SIG によって定義されたターゲット ロールの Find Me Profile の場合、**Find Me Profile** の仕様の通りに GAP role が Peripheral にセットされ、GATT role が Server にセットされます。

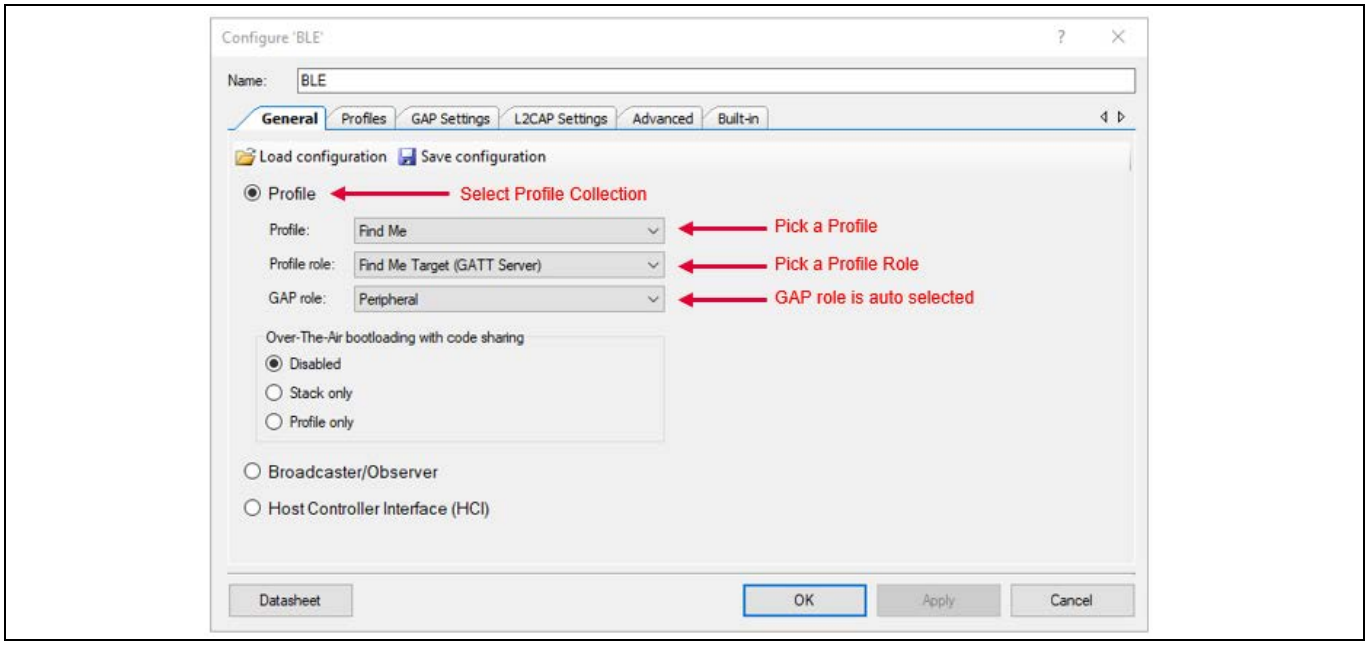


Figure 15 Bluetooth® LE コンポーネントのプロファイル コンフィギュレーション

ステップ 2: 選択したプロファイル ロールに応じて、サポートされるすべての GATT サービスと該当するキャラクタリスティックはプロファイル仕様に従って自動生成されます。設計に基づいてサービスとキャラクタリスティックの値を確認し、必要に応じて編集します。**Profiles** タブ内の設定情報は、選択した設計のアトリビュート データベースを形成します。

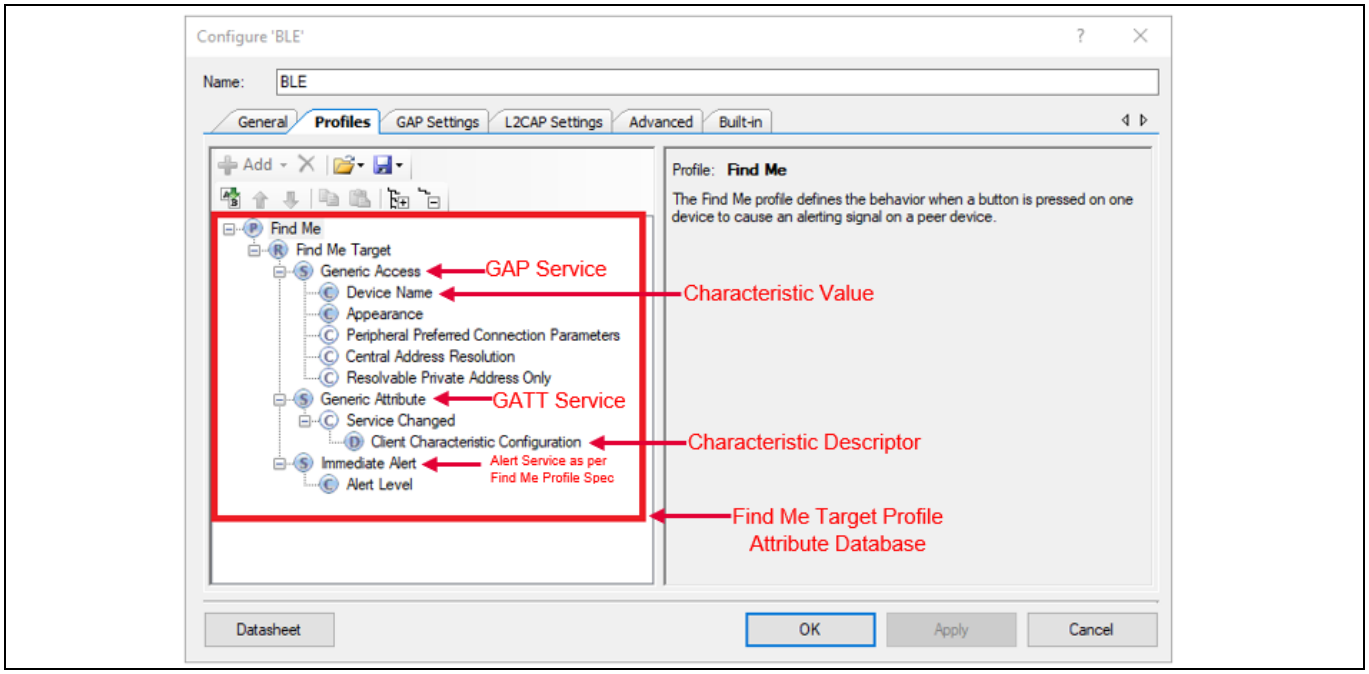


Figure 16 Bluetooth® LE アトリビュート データベースのコンフィギュレーション

ステップ 3: Figure 17 と **Figure 18** に示すように、14 ページの上部で説明したように、ご使用の設計の GAP の一般設定と宣伝データ設定を行います。

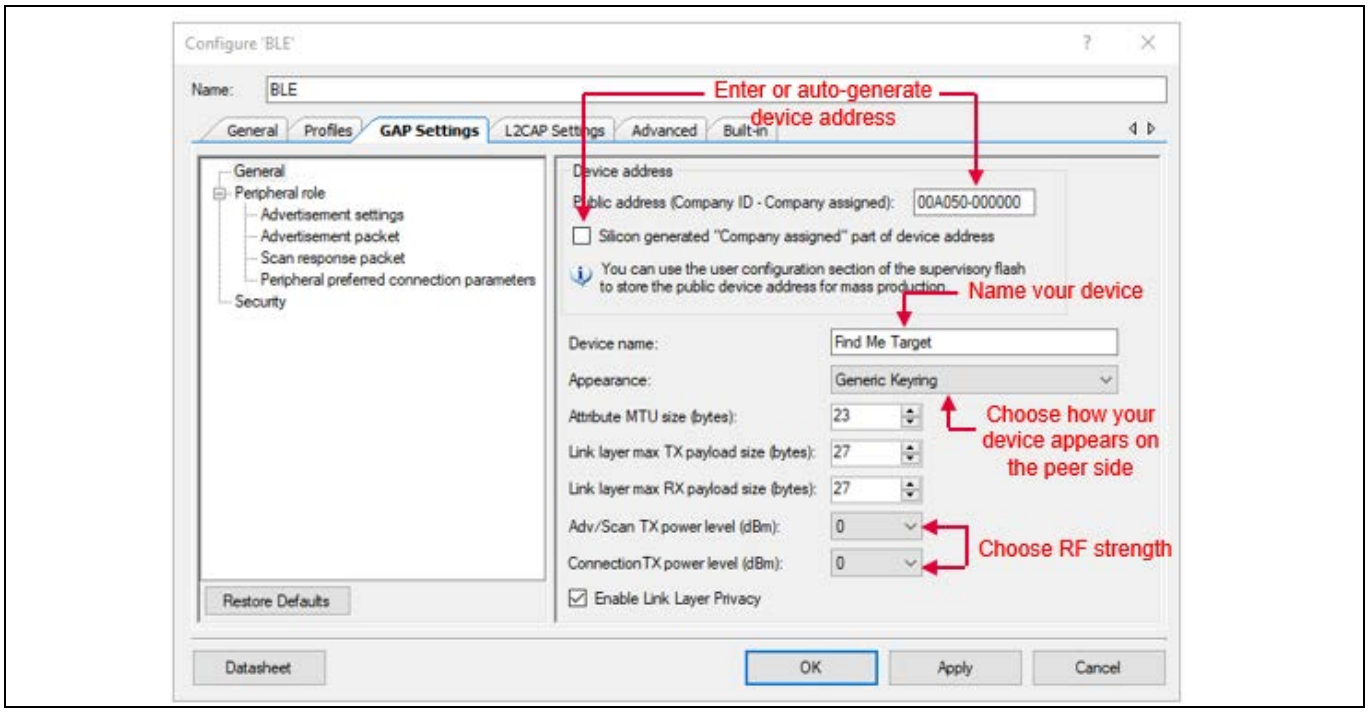


Figure 17 Bluetooth® LE GAP のコンフィギュレーション

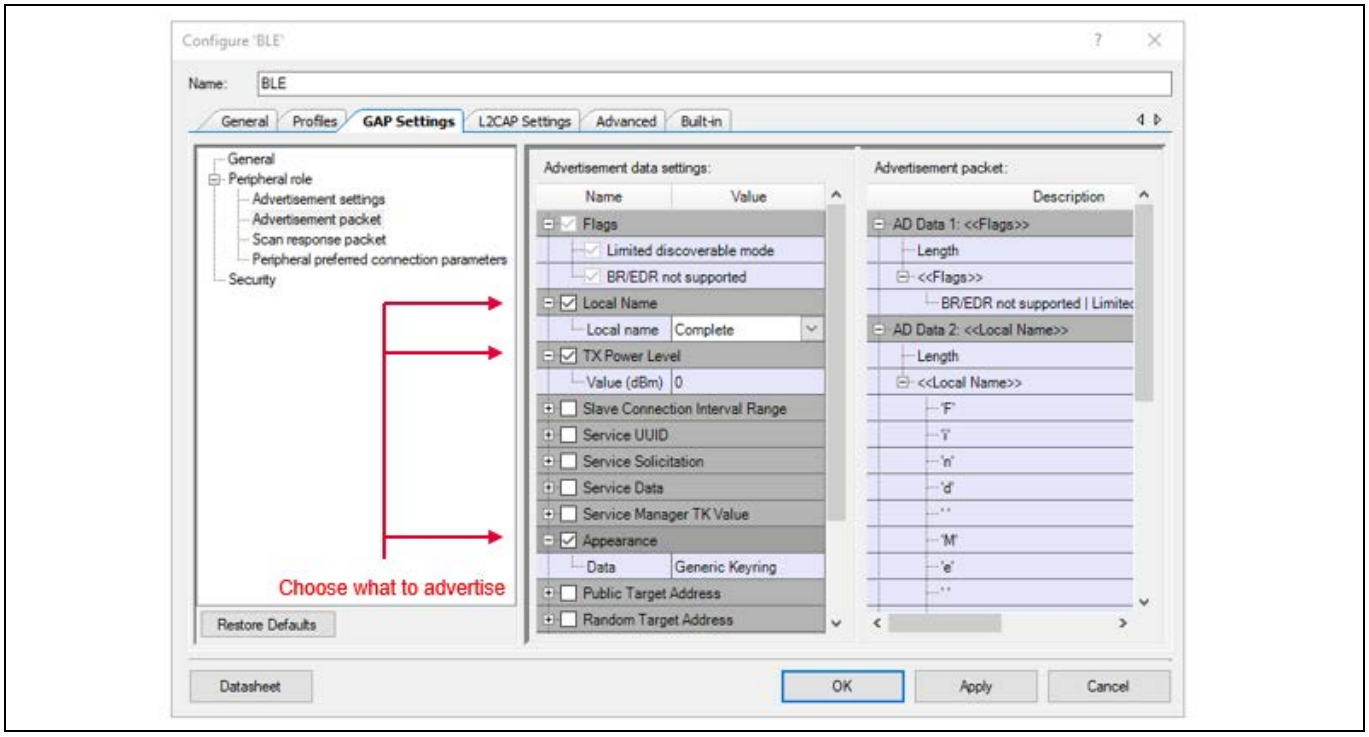


Figure 18　Bluetooth® LE 宣伝データのコンフィギュレーション

ステップ 4: ファームウェアを書いて、設定した設計を初期化します。データとイベントの受信のために Bluetooth® LE コンポーネントにイベント ハンドラを登録します。イベント ハンドラの詳細は、本アプリケーション ノートの**後半**で説明されています。

ステップ 5: プログラムのメイン ループで待機し、Bluetooth® LE コンポーネントからイベントが発生したら、Bluetooth® LE コンポーネントの API 関数を使って必要な処置を取るか、またはセントラル デバイスにデータを送信します。

「**初めての First PSoC™ 4 with AIROC™ Bluetooth® LE 設計**」節で、シンプルなペリフェラル アプリケーションを作成するための Bluetooth® LE コンポーネントの設定手順を説明します。Bluetooth® LE 標準とカスタム プロファイルを用いたアプリケーション開発のために Bluetooth® LE コンポーネントを使用する方法の詳細な説明については、アプリケーション ノート **AN91184** と **AN91162** を参照ください。

6 PSoC™ 4 with AIROC™ Bluetooth® LE 開発セットアップ

Figure 19 に、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスを用いて Bluetooth® LE ペリフェラル設計を評価するために必要なハードウェアとソフトウェア ツールを示します。一般的に、Bluetooth® LE Pioneer Kit (**Figure 19** に示す赤色の基板) は、CySmart iOS／Android アプリまたは CySmart ホスト エミュレーション ツールなどのセントラル デバイスと通信できるペリフェラルとして設定されます。 また、CySmart ホスト エミュレーション ツールの動作には Bluetooth® LE ドングル (**Figure 19** に示す黒色の基板) が必要です。

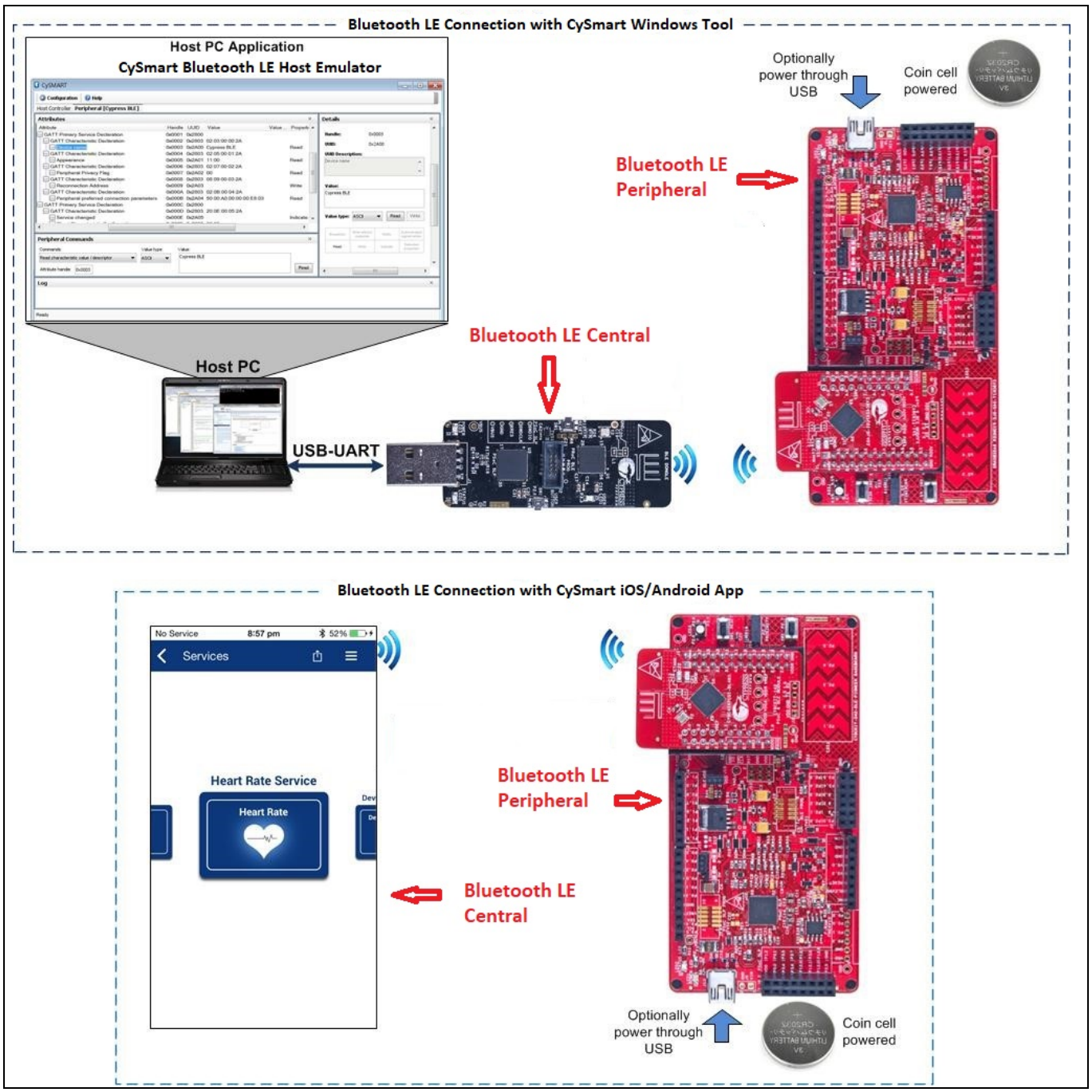


Figure 19 Bluetooth® LE 機能的セットアップ

PSoC™ 4 with AIROC™ Bluetooth® LE 開発セットアップ

Figure 20 に示すように、Bluetooth® LE ドングルは Windows 向け CySmart ホスト エミュレーション ツールで動作するようにプログラムされています。Bluetooth® LE Pioneer Kit は、Bluetooth® LE 設計をプログラムまたはデバッグするために PSoC™ Creator と併用する基板搭載 USB プログラマを備えています。Bluetooth® LE Pioneer Kit には、USB インターフェースを介して、またはコイン電池で電源を供給できます。Bluetooth® LE ドングルと Bluetooth® LE Pioneer Kit の両方は、開発とテストのために同時に共通のホスト PC に接続できます。

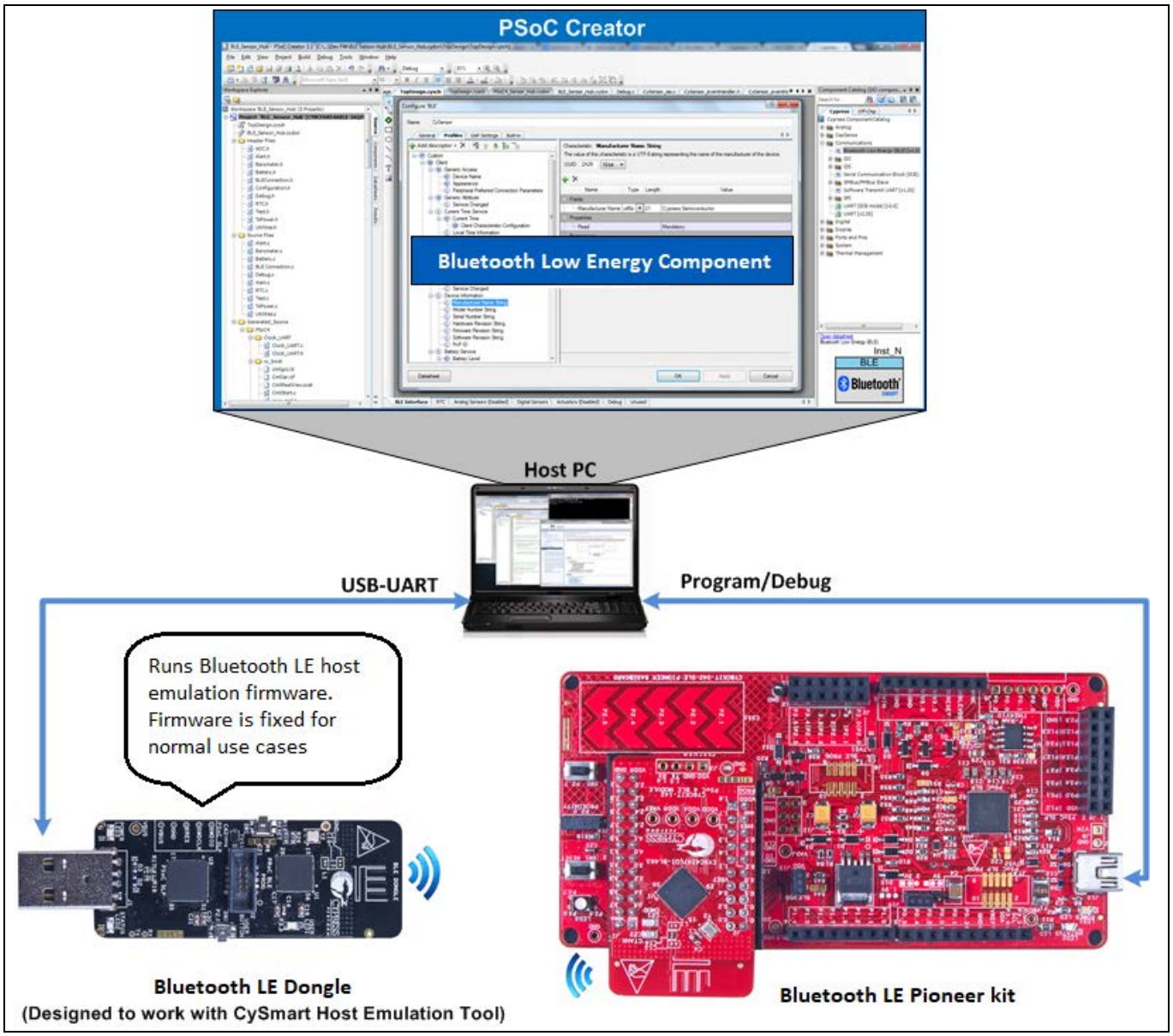


Figure 20　Bluetooth® LE 開発セットアップ

7 初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計

本節で、PSoC™ Creator を使用して PSoC™ 4 with AIROC™ Bluetooth® LE デバイスのシンプルな Bluetooth® LE Pioneer Kit ベース設計を作成する手順を説明します。本節で、Bluetooth SIG によって定義されたシンプルな標準プロファイル デザインの作成を説明します。高度な標準またはカスタム プロファイル デザインを作成するには、「**Designing Bluetooth® LE Applications**」および「**Creating a Bluetooth® LE Custom Profile**」アプリケーション ノートを参照ください。

7.1 設計について

本設計では、Immediate Alert Service (**IAS**；即時アラート サービス) に対応するターゲット ロールの Bluetooth® LE **Find Me Profile (FMP)** を実装します。FMP と IAS それぞれは、Bluetooth SIG によって定義された Bluetooth® LE 標準プロファイルとサービスです。**Figure 21** に示すように、Find Me Locator によってトリガーされるアラート レベルは Bluetooth® LE Pioneer Kit 上の LED の状態を変化させることで示します。2 個のステータス LED は Bluetooth® LE インターフェースの状態を示します。

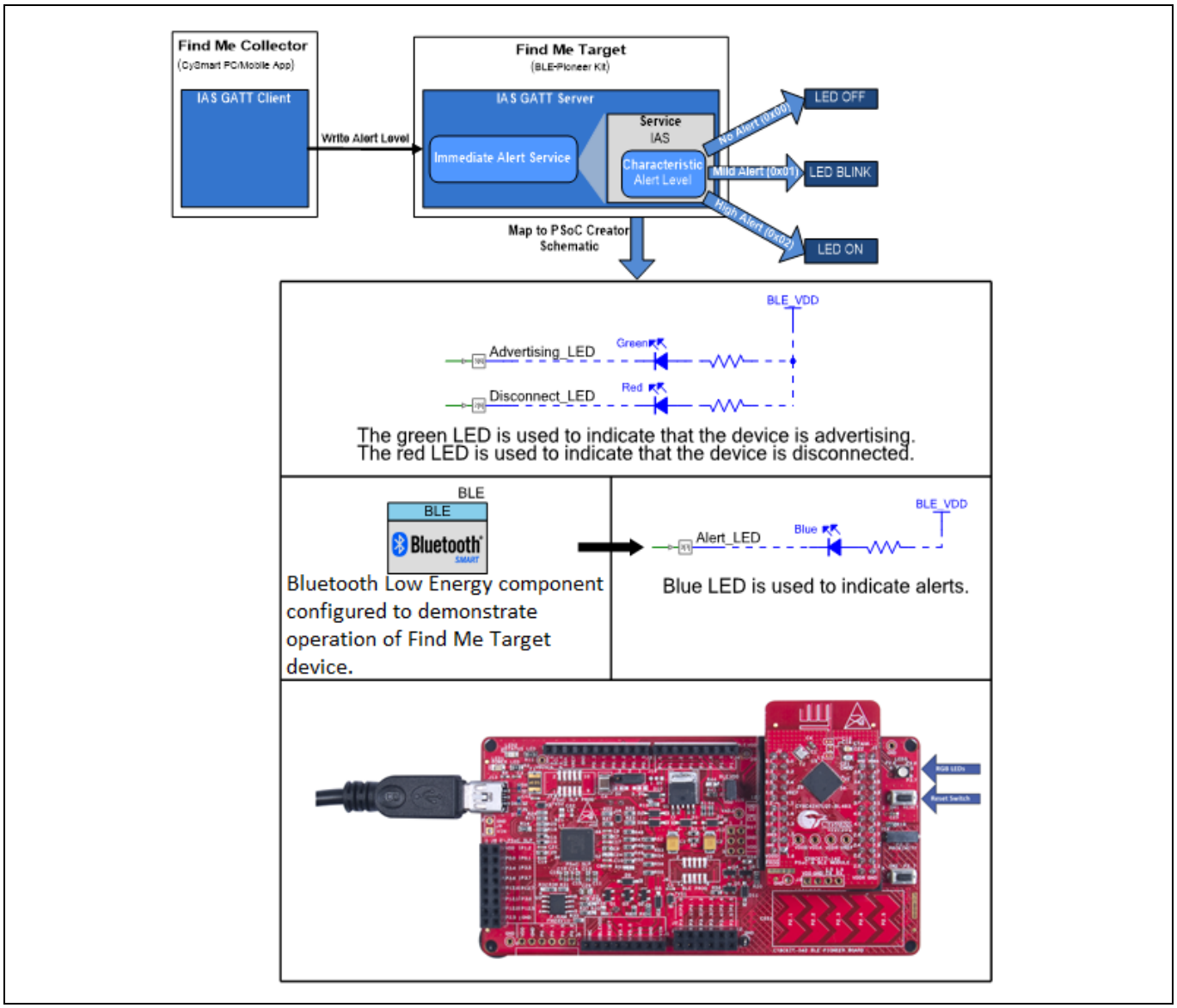


Figure 21 初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計

初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計

初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計の作成手順は以下の 4 段階を踏みます。

1. PSoC™ Creator 回路図ページで設計を作成します。
2. Bluetooth® LE イベントを初期化および処理するようにファームウェアを書きます。
3. Bluetooth® LE Pioneer Kit 上の PSoC™ 4 with AIROC™ Bluetooth® LE デバイスをプログラムします。
4. CySmart ホスト エミュレーション ツールまたはモバイル アプリを使用して設計をテストします。

Note: *本アプリケーション ノートで説明する Bluetooth® LE サンプル デザインの PSoC™ Creator 機能プロジェクトは PSoC™ Creator サンプル コードの一部として含まれています。サンプル コード (**File > Code Example** を選択して、**Filter by** を **Find Me > BLE_FindMe.** に選択すること) で開始することが可能です。サンプル コードを使用する場合、上述の回路図のコンフィギュレーション (第 1 段階) とファームウェア開発 (第 2 段階) を**飛ばして**、設計のプログラミング (**第 3 段階**) およびテスト (**第 4 段階**) に進んでください。*

7.2 事前準備

実装を行う前に、必ず **Bluetooth® LE Pioneer Kit** を入手し、以下のソフトウェアをインストールしたことをご確認ください。

- **PSoC™ Creator 4.4** および **PSoC™ Programmer 3.29**
- **CySmart ホスト エミュレーション ツール**または **iOS**／**Android** 向け CySmart アプリ

7.3 第 1 段階: 設計を作成

本節では、設計プロセスの概要について順に説明します。空のプロジェクトを作成してから、回路図設計入力について解説します。

1. PC に PSoC™ Creator 4.4 またはそれ以降版をインストールします。
2. PSoC™ Creator を起動し、**Figure 22** に示すように **File** メニューから **New** > **Project** を選択します。

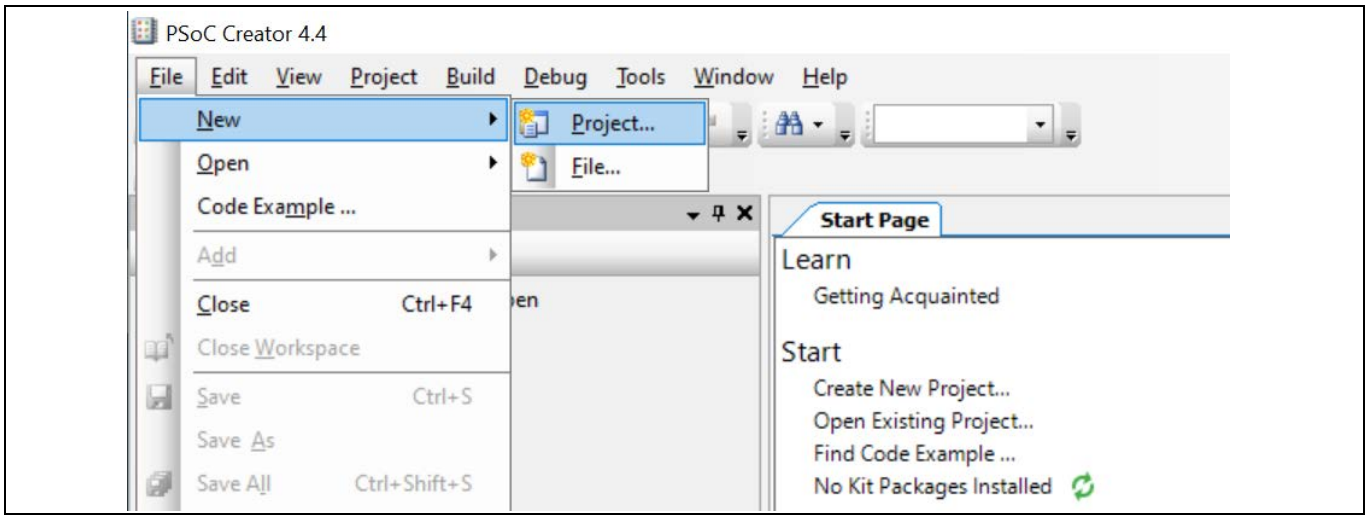


Figure 22 新規プロジェクトの作成

3. **Figure 23** に示すように、Target device を「PSoC™ 4」に選択して **Bluetooth® LE Pioneer Kit** で使用される CY8C4248LQI-BL583 デバイスを選択し、**Next** をクリックします。CY8C4248LQI-BL583 デバイスを選択するには、ドロップダウンをクリックして「Launch Device Selector」を選択し、指定されたデバイスを選択します。カスタム PSoC™ 4 with AIROC™ Bluetooth® LE ハードウェアまたは異なる PSoC™ 4 with AIROC™ Bluetooth® LE 製品番号を使用している場合、Target device で「Launch Device Selector」オプションを選択して適切な製品番号を選択します。

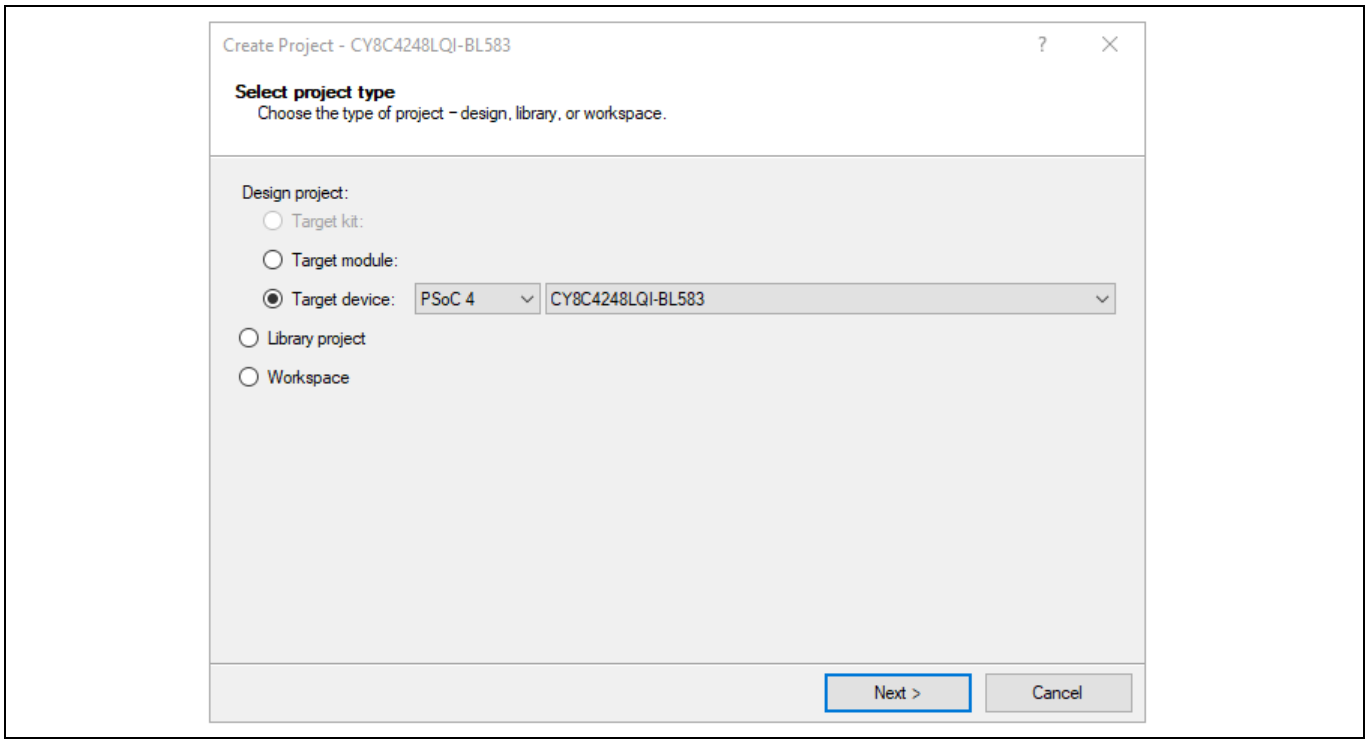


Figure 23　ターゲット デバイスの選択

4. **Figure 24** に示すように、「Empty Schematic」のプロジェクト テンプレートを選択して、**Next** をクリックします。

Note: *この設計にあらかじめ構築された例を使用したい場合、**Figure 24** の Code example を選択して、**Next** をクリックします。サンプル コード選択ウィンドウで、**Filter by** を **Find Me** にし、**BLE_FindMe** を選択して、プログラミングの節に進んでください。*

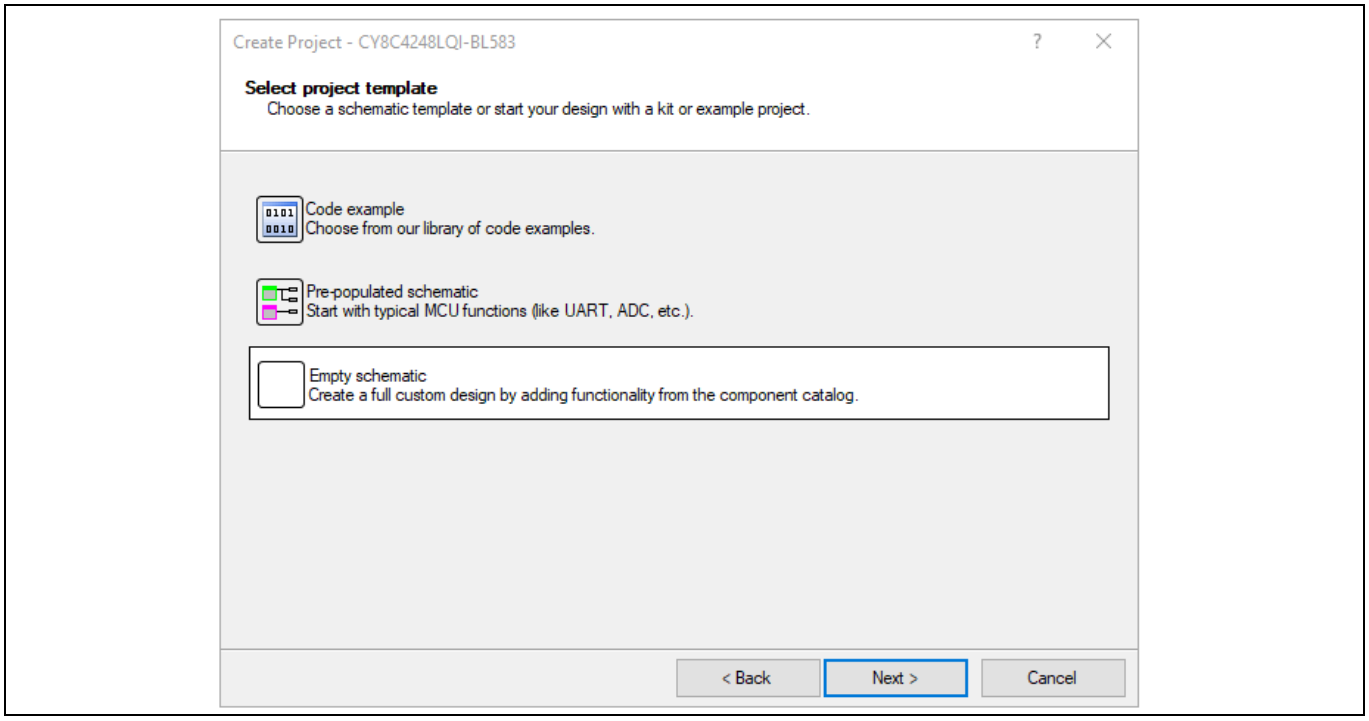


Figure 24　プロジェクト テンプレートの選択

5. **Figure 25** に示すように、ワークスペースとプロジェクトに「BLE_FindMe」などの名前を付けます。新規プロジェクトの適切な場所を選択して、**Finish** をクリックします。

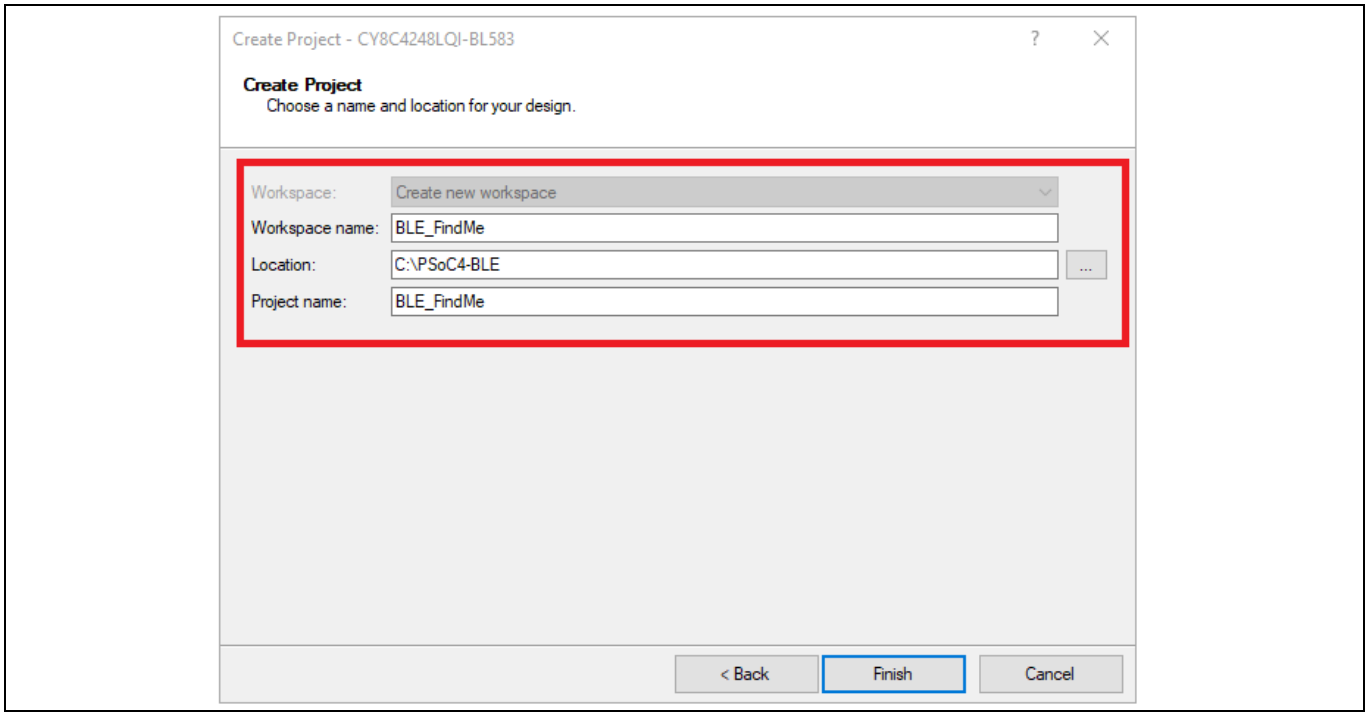


Figure 25　プロジェクトの名付け

初めての PSoC™ 4 with AIROC™ Bluetooth® LE 設計

新規プロジェクトを作成すると、基本的なファイル セットを格納するプロジェクト フォルダが作成されます。**Figure 26** に示すように、**Workspace Explorer** ウィンドウにこれらのファイルが表示されます。

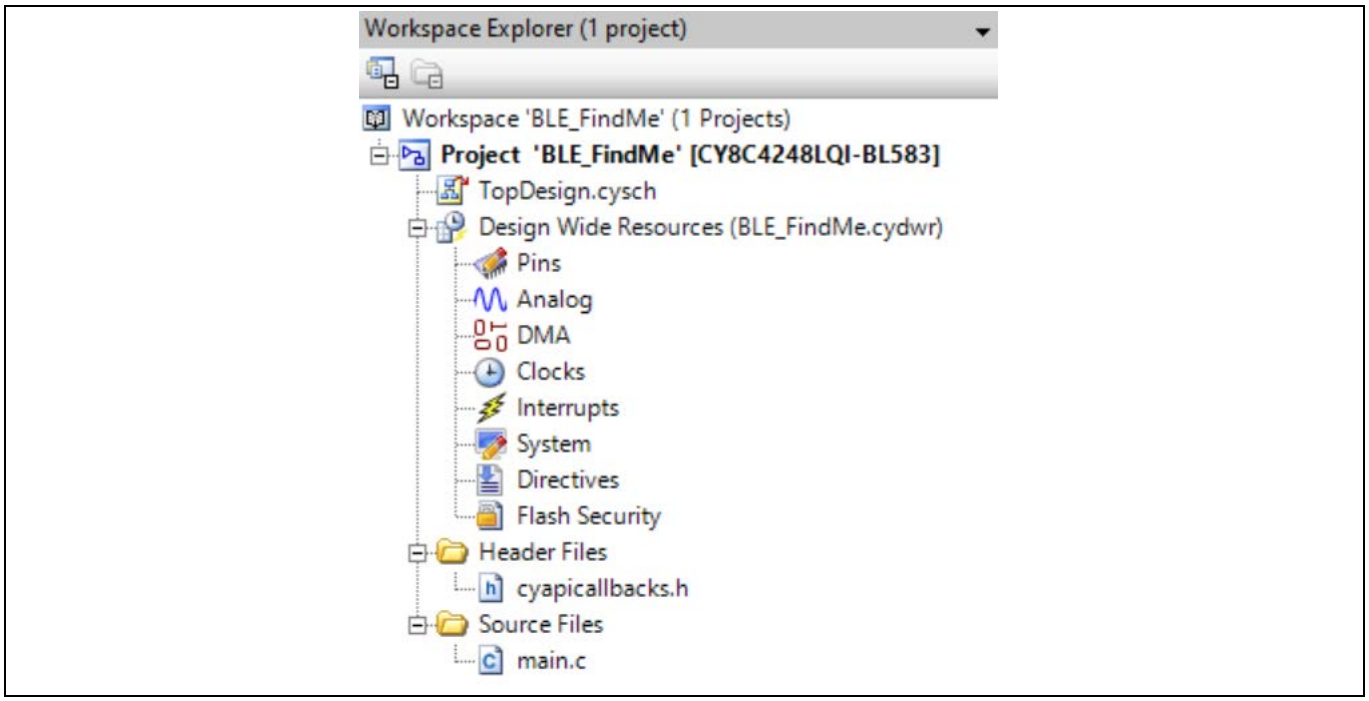


Figure 26　TopDesign 回路図の開設

6. *TopDesign.cysch* ファイルをダブルクリックして、プロジェクトの回路図を開きます。
7. プロジェクトの回路図内で、コンポーネントをドラッグ＆ドロップし、配線することでアナログ、デジタルと通信設計が作成できます。本設計では、**Figure 21** に示すように Bluetooth® LE とピンの 2 つのコンポーネントが必要です。
8. **Figure 27** に示すように、**Component Catalog** から回路図に 1 つの Bluetooth® Low Energy コンポーネントをドラッグします。

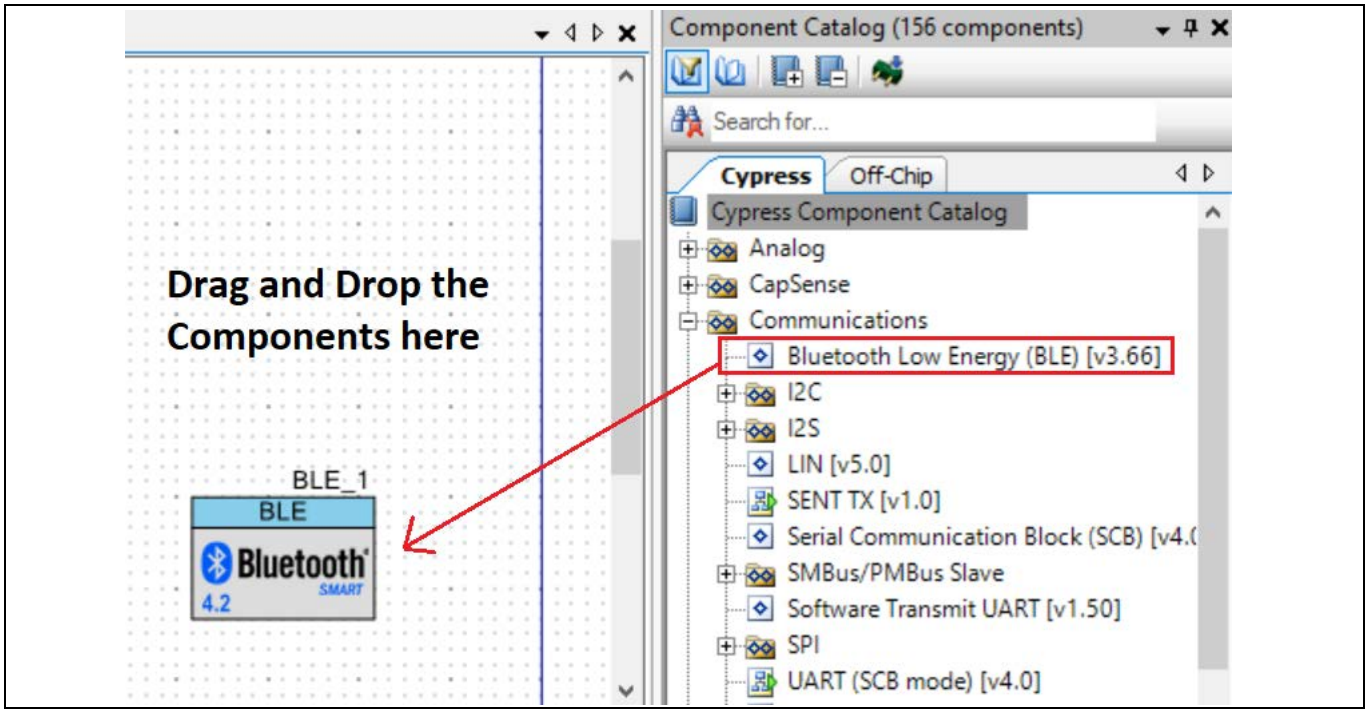


Figure 27 Bluetooth® LE コンポーネントの配置

9. 回路図内の Bluetooth® LE コンポーネントをダブルクリックし、以下の特性を持つ「BLE Find Me Target」に設定します。

- **Figure 28** に示すような GAP ペリフェラル ロール、Find Me Target (GATT サーバー) コンフィギュレーション
- **Figure 29** に示すように、GAP Device Name を「Find Me Target」にセットし、Appearance を「Generic Keyring」にセットし、「Silicon generated “Company assigned” part of device address」チェックボックスをチェックします。これにより、他のデバイスがご使用のデバイスを検出しようとし、それに固有の Bluetooth® LE デバイス アドレスを割り当てる時に現れるデバイス名とタイプを設定します。
- **Figure 30** に示すような、宣伝タイムアウトが 30 s に、高速宣伝間隔が 20 ms～30 ms に設定された制限宣伝モード。高速宣伝により、迅速に検出し接続を確立することが可能ですが、RF 宣伝パケットが増加するため消費電力が高くなります。「Slow advertising interval」チェックボックスのチェックを外します。
- **Figure 31** に示すように、Advertisement Packet で、Immediate Alert Service を有効にします。**Figure 32** に示すように、Scan Response Packet で、Local Name、Tx Power Level、Appearance フィールドを有効にします。
- GAP のセキュリティ レベルを可能な限り低いコンフィギュレーション (モード 1、セキュリティなし) に設定します。このコンフィギュレーションは、データ交換用の認証や暗号化、認可が不要です。I/O capabilities を「No Input No Output」に、Bonding requirement を「No Bonding」にセットします。

Figure 28～**Figure 34** に、この Bluetooth® LE コンポーネント コンフィギュレーションのスクリーンショットを示します。

Note: *この設計の場合、**GAP Settings > Peripheral preferred connection parameters**、**Profiles**、**L2CAP Settings** と **Advanced** タブの Bluetooth® LE コンポーネントのデフォルト コンフィギュレーションを変更する必要がありません。*

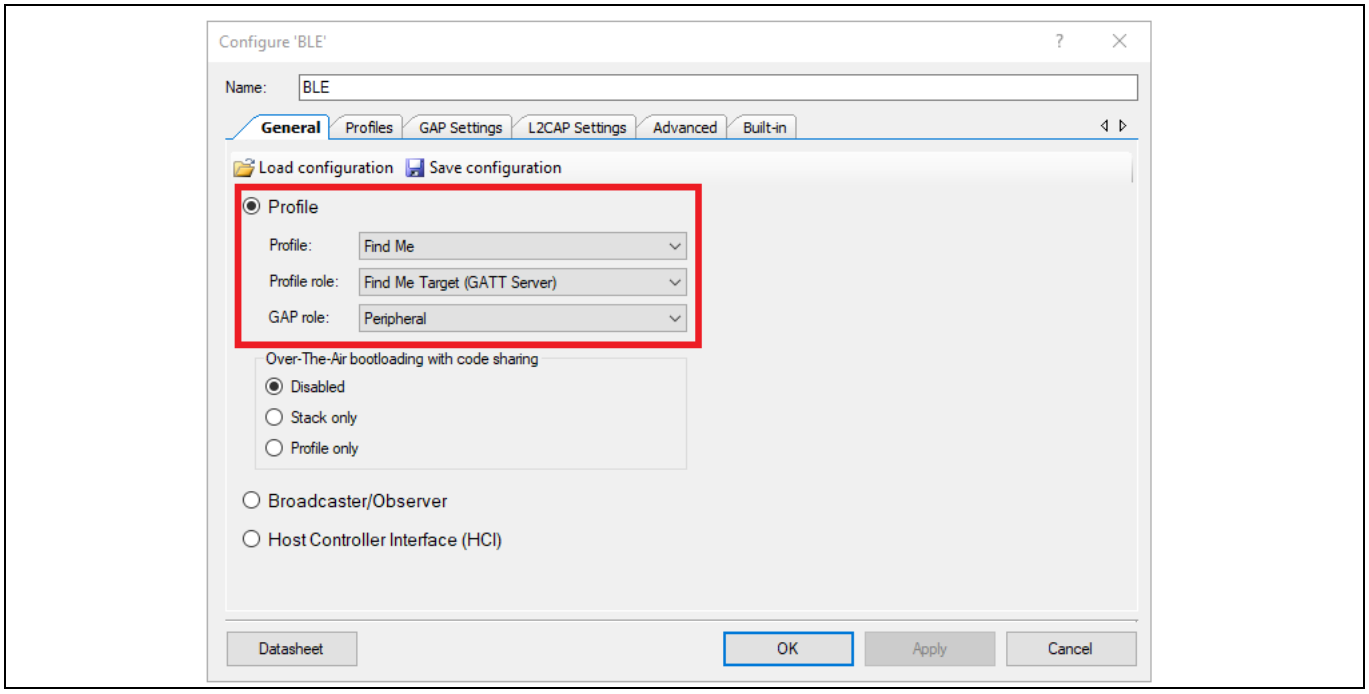


Figure 28 Bluetooth® LE コンポーネントの一般コンフィギュレーション

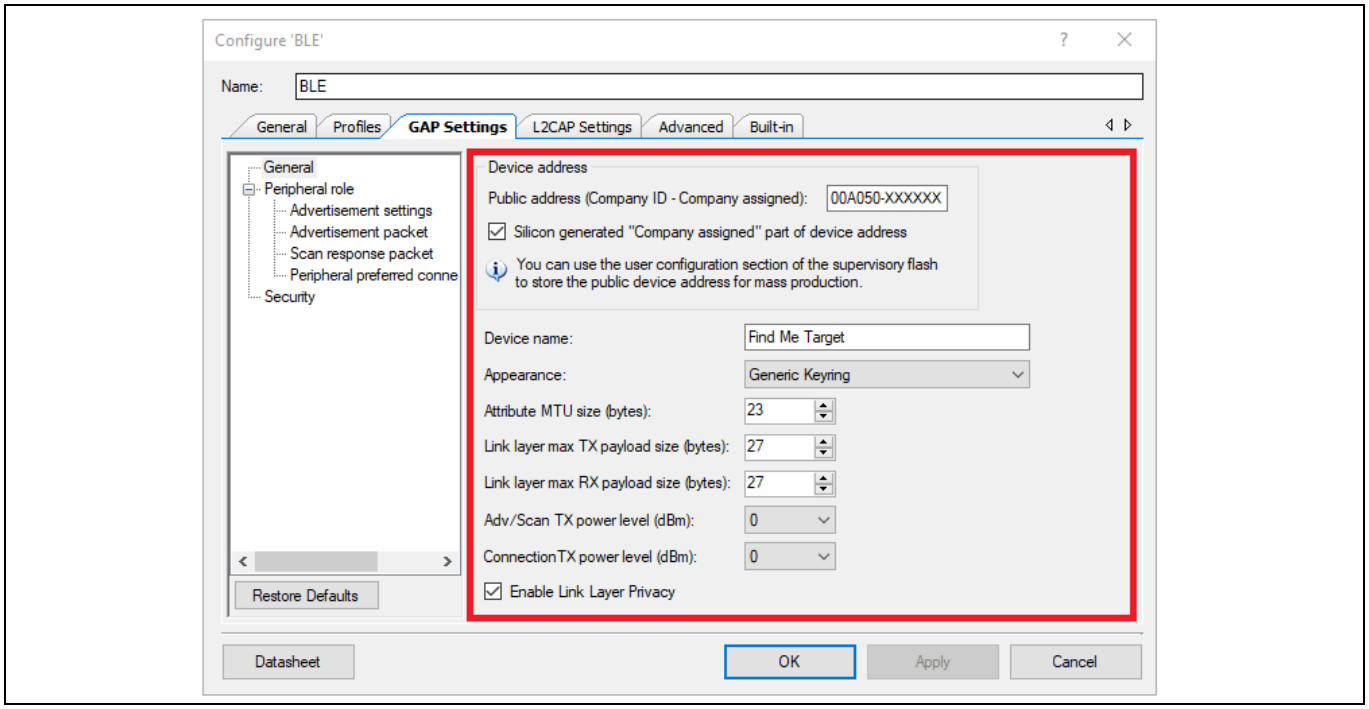


Figure 29 Bluetooth® LE コンポーネントの GAP 一般設定

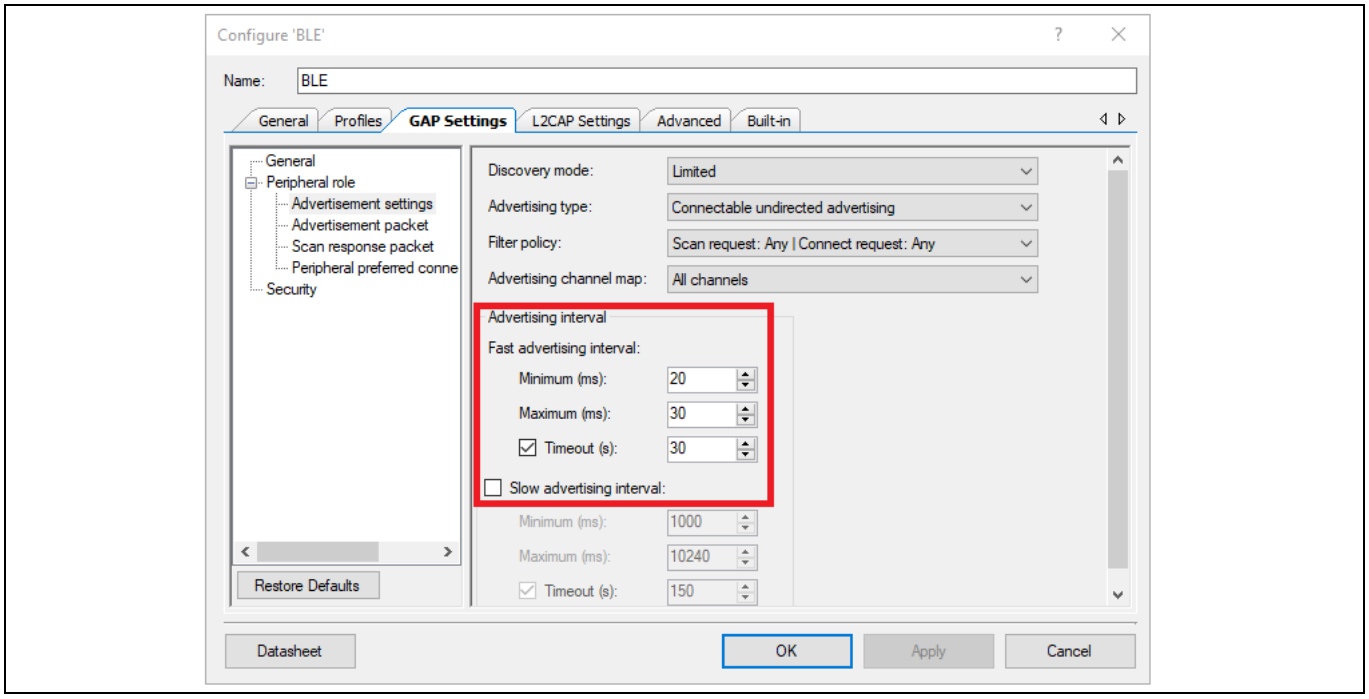


Figure 30 Bluetooth® LE コンポーネントの GAP 宣伝設定

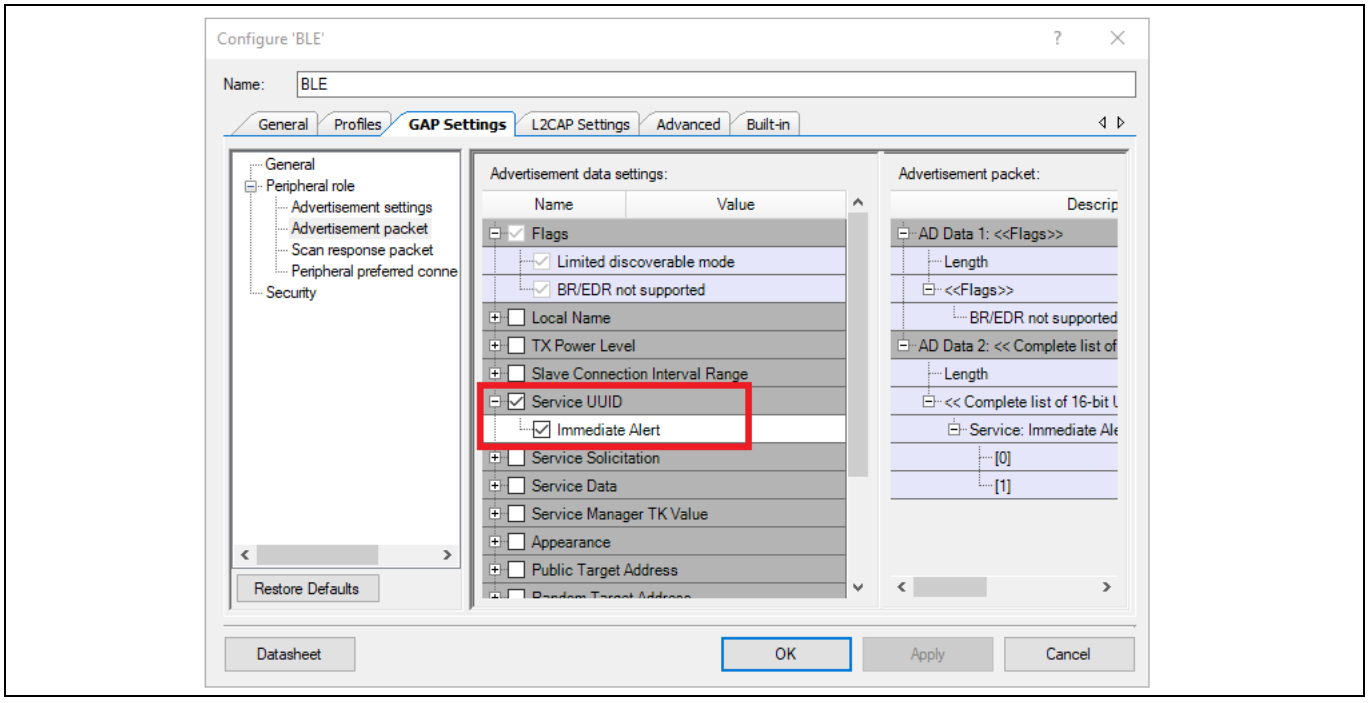


Figure 31 Bluetooth® LE コンポーネントの GAP 宣伝パケット

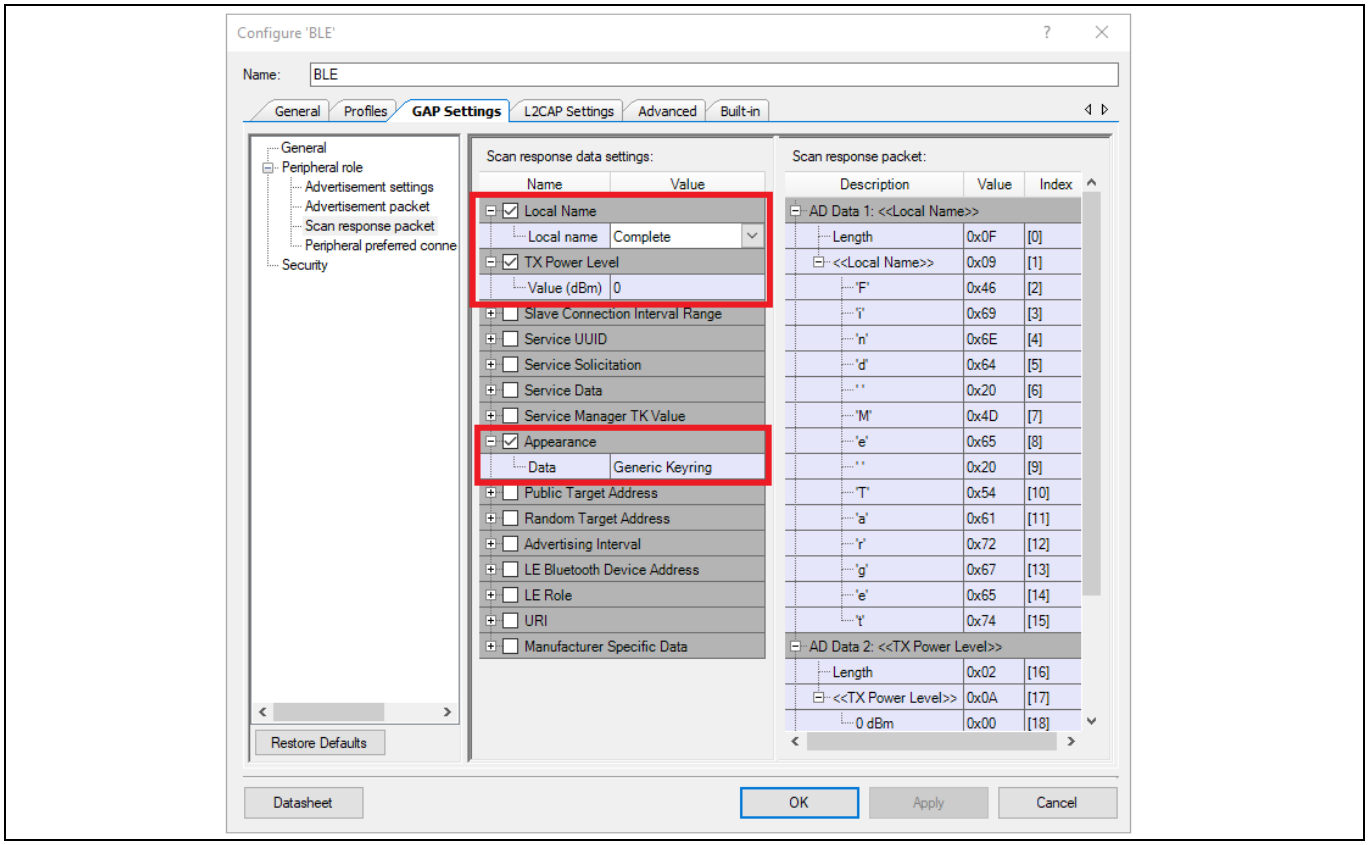


Figure 32　Bluetooth® LE コンポーネントの GAP スキャン応答パケット

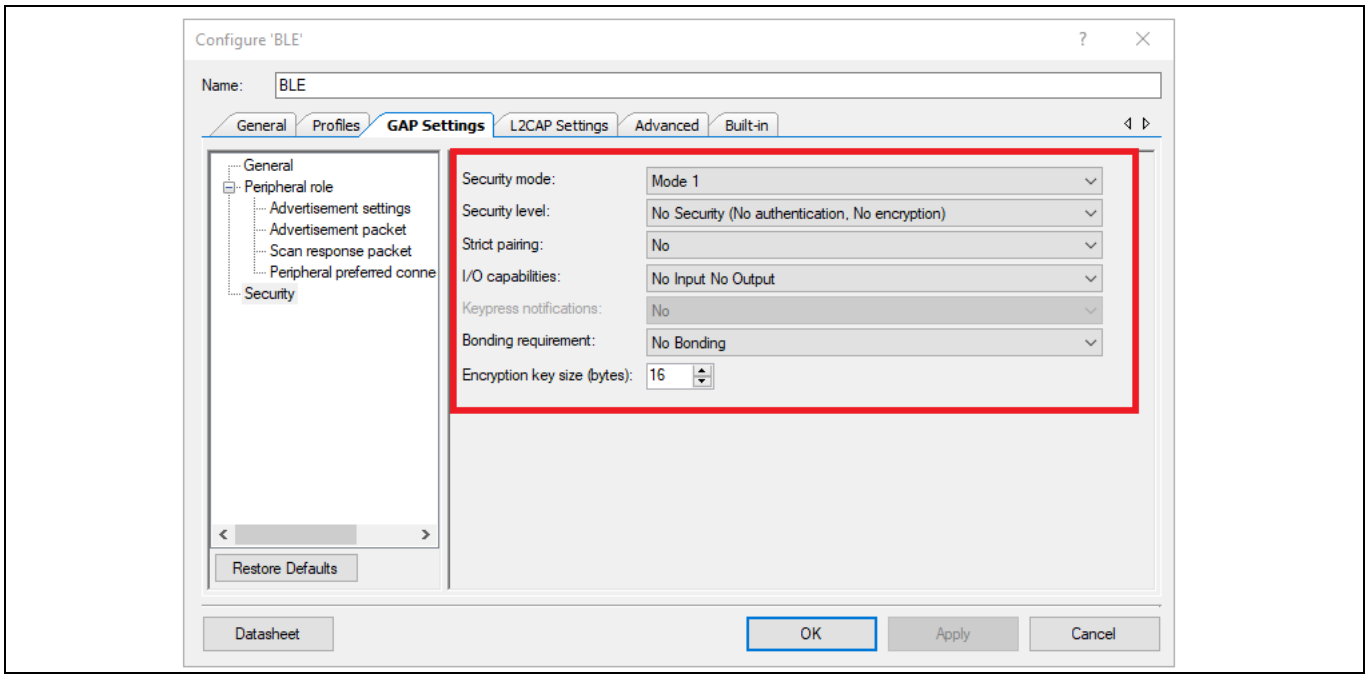


Figure 33　Bluetooth® LE コンポーネントの GAP セキュリティ設定

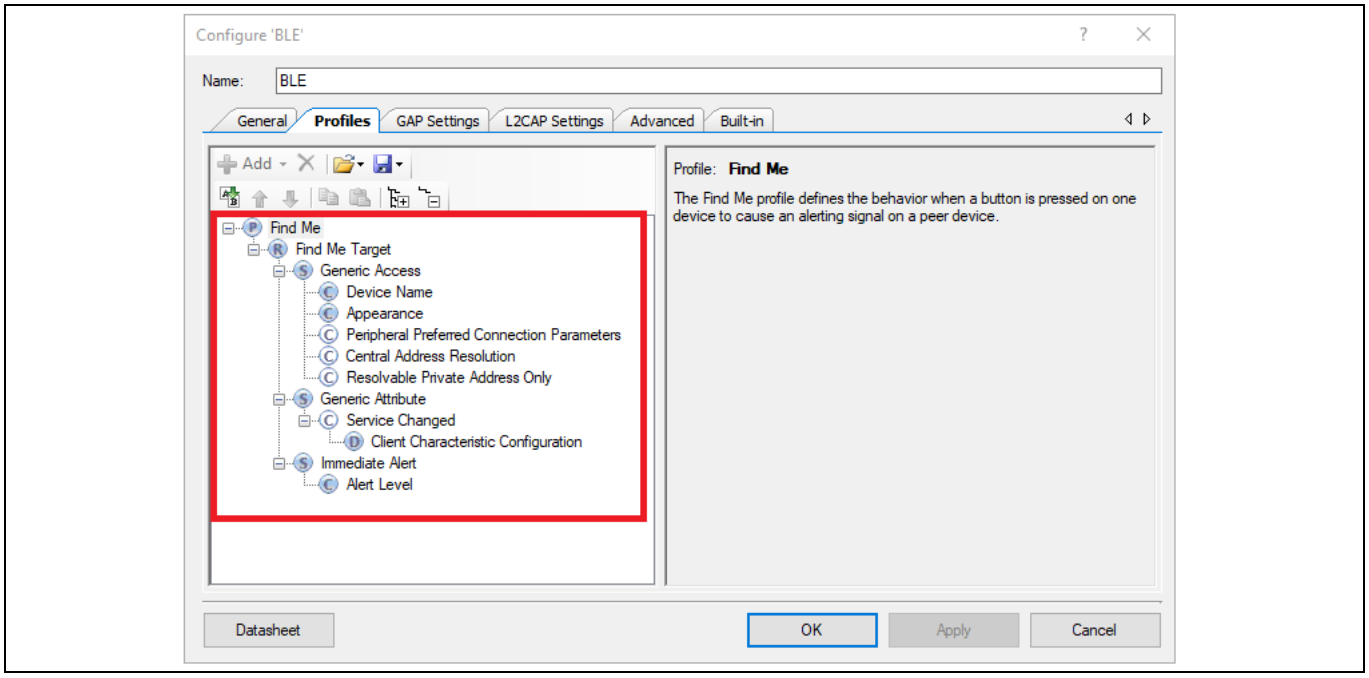


Figure 34　Bluetooth® LE コンポーネントのプロファイル コンフィギュレーション

10. **Figure 35** に示すように、Digital Output Pin コンポーネントをドラッグ＆ドロップします。

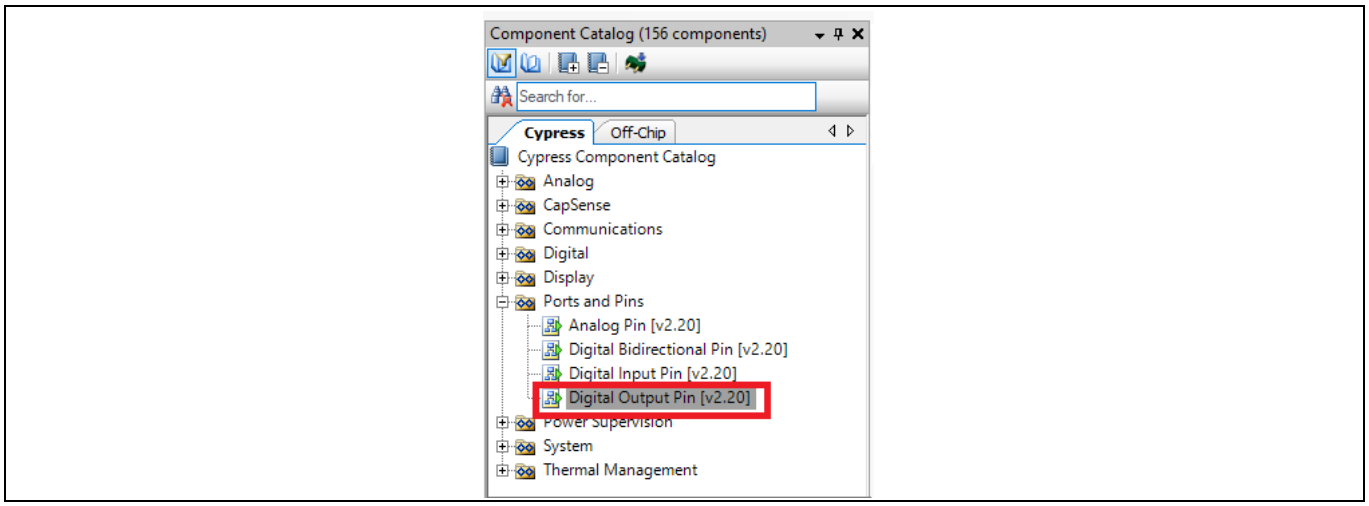


Figure 35　デジタル出力ピン コンポーネントの配置

11. **Figure 36** に示すように、コンポーネントをダブルクリックして以下の変更を行います。

- **HW Connection** チェックボックスのチェックを外します。
- Name を **Alert_LED** に変更します。
- **Initial drive state** を「High (1)」にセットします。

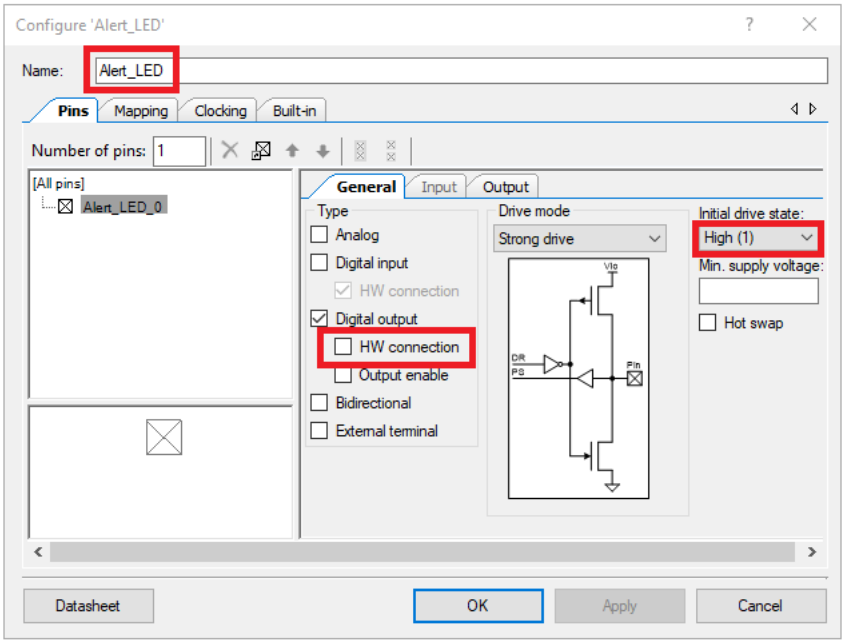


Figure 36　ピン コンポーネントの名前変更

12. 2 つの Digital Output Pin コンポーネントを追加します。**Figure 37** と **Figure 38** に示すように、コンポーネントをダブルクリックして以下の変更を行います。

- コンポーネント名を「Advertising_LED」と「Disconnect_LED」に変更します。
- **HW Connection** チェックボックスのチェックを外します。
- **Initial drive state** を「High (1)」にセットします。

これらのピンを使用して Bluetooth® LE 宣伝と切断状態指示 LED を駆動します。Bluetooth® LE Pioneer Kit 上の LED はアクティブ LOW です。すなわち、ピンを HIGH に駆動すると LED は消灯し、ピンを LOW に駆動すると LED は点灯します。

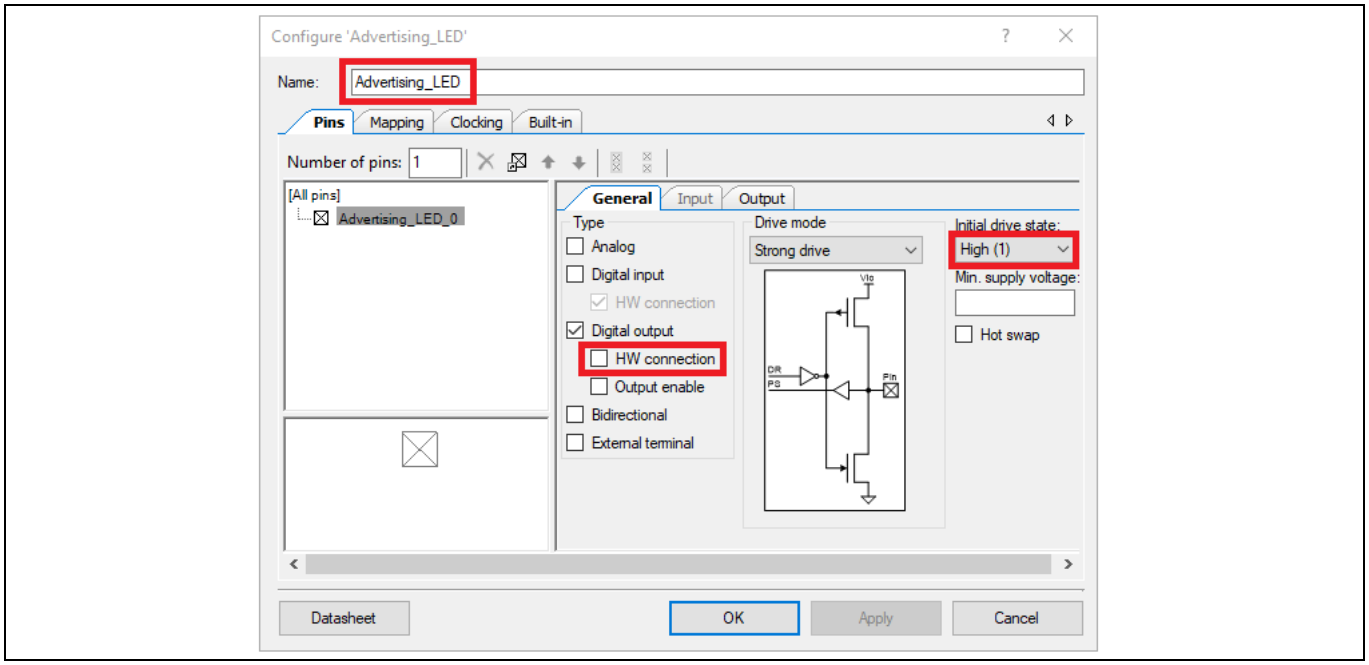


Figure 37　宣伝 LED のコンフィギュレーション

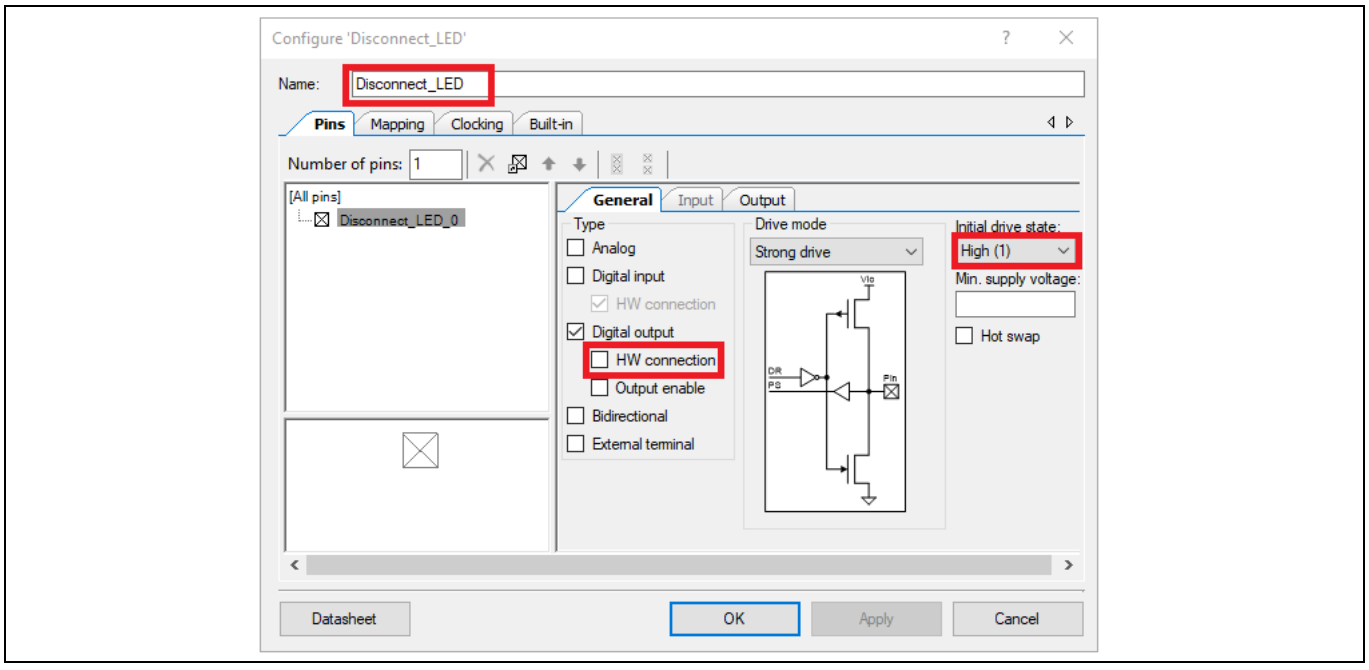


Figure 38　切断 LED のコンフィギュレーション

13. 回路図設定がを完了した後、設計は **Figure 39** のようになります。

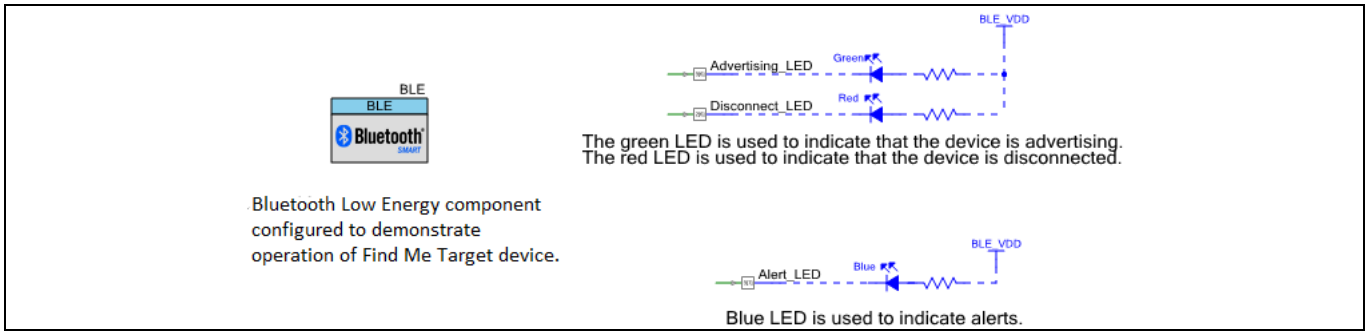


Figure 39　回路図のコンフィギュレーション

Note: ***Figure 39*** *に示した青色の点線、LED 記号、抵抗記号は便宜上にのみ表示されるチップ外部の PSoC™ Creator コンポーネントであり、設計の機能には不要です。チップ外部のコンポーネントを設計にするために、PSoC™ Creator のチップ外部コンポーネント カタログから必要なチップ外部コンポーネントをプロジェクト回路図ページにドラッグ & ドロップします。*

14. **Workspace Explorer** のデザイン全体のリソース ファイル (*BLE_FindMe.cydwr*)で、**Pins** セクションをダブルクリックしてください。このセクションを使用して、出力 (Advertising_LED、Disconnect_LED、および Alert_LED) のデバイスピンを選択できます。**Figure 40** に、Advertising_LED、Disconnect LED、Alert_LED ピンを Bluetooth® LE Pioneer Kit 上のそれぞれ緑色 LED, 赤色 LED, 青色 LED に接続するピンコンフィギュレーションを示します。

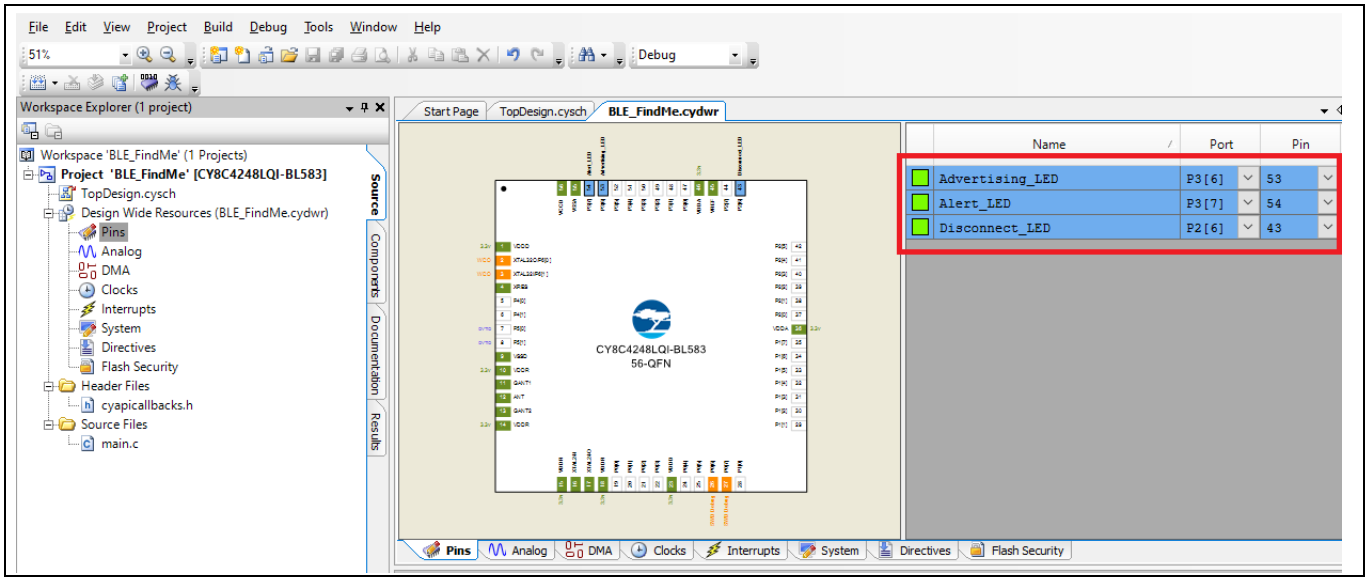


Figure 40　ピンの選択

自分の基板、または LED を備えていない開発キットを使用する場合、適切なピンを選択してください。**Figure 21** に示すように、選択したピンに外部 LED を接続できます。

15. 同様に、Design Wide Resources (BLE_FindMe.cydwr) ファイルの下の **Clocks** セクションで、**Edit Clock** ボタンをクリックしてください。クロックを設定するためのウィンドウが開きます。**Figure 41** に示すように、**High Frequency Clocks** タブに移動し、IMO を変更して内部メインオシレータ (IMO) を 12MHz に設定してください。**OK** をクリックしてください。

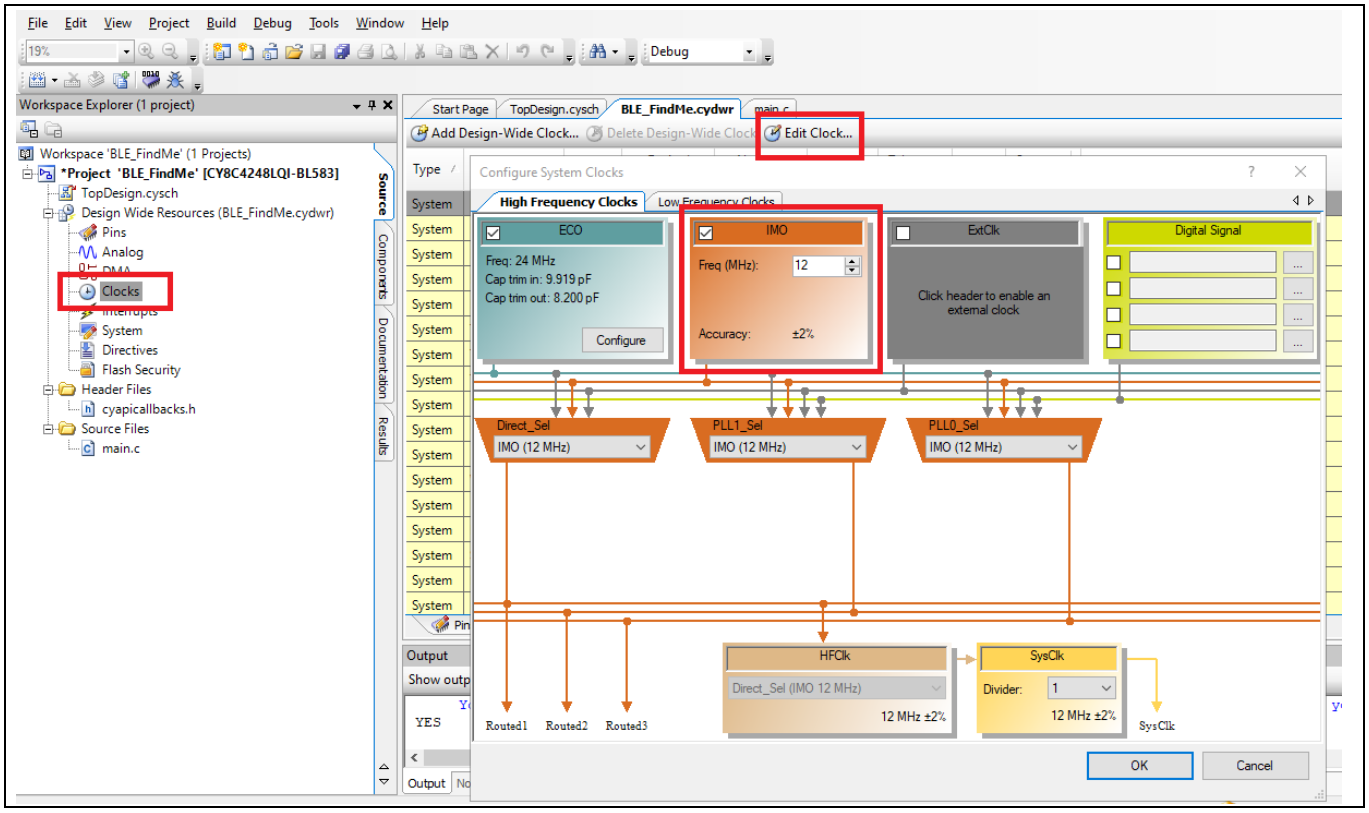


Figure 41　クロックのコンフィギュレーション

16. **Build** メニューから **Generate Application** を選択します。**Figure 42** に示すように、**Workspace Explorer** ウィンドウには、PSoC™ Creator によって自動的に生成された Bluetooth® LE とデジタル出力ピン コンポーネントのソースコード ファイルが表示されています。次の節に進む前に、生成された main.c ファイルの内容を消去します。

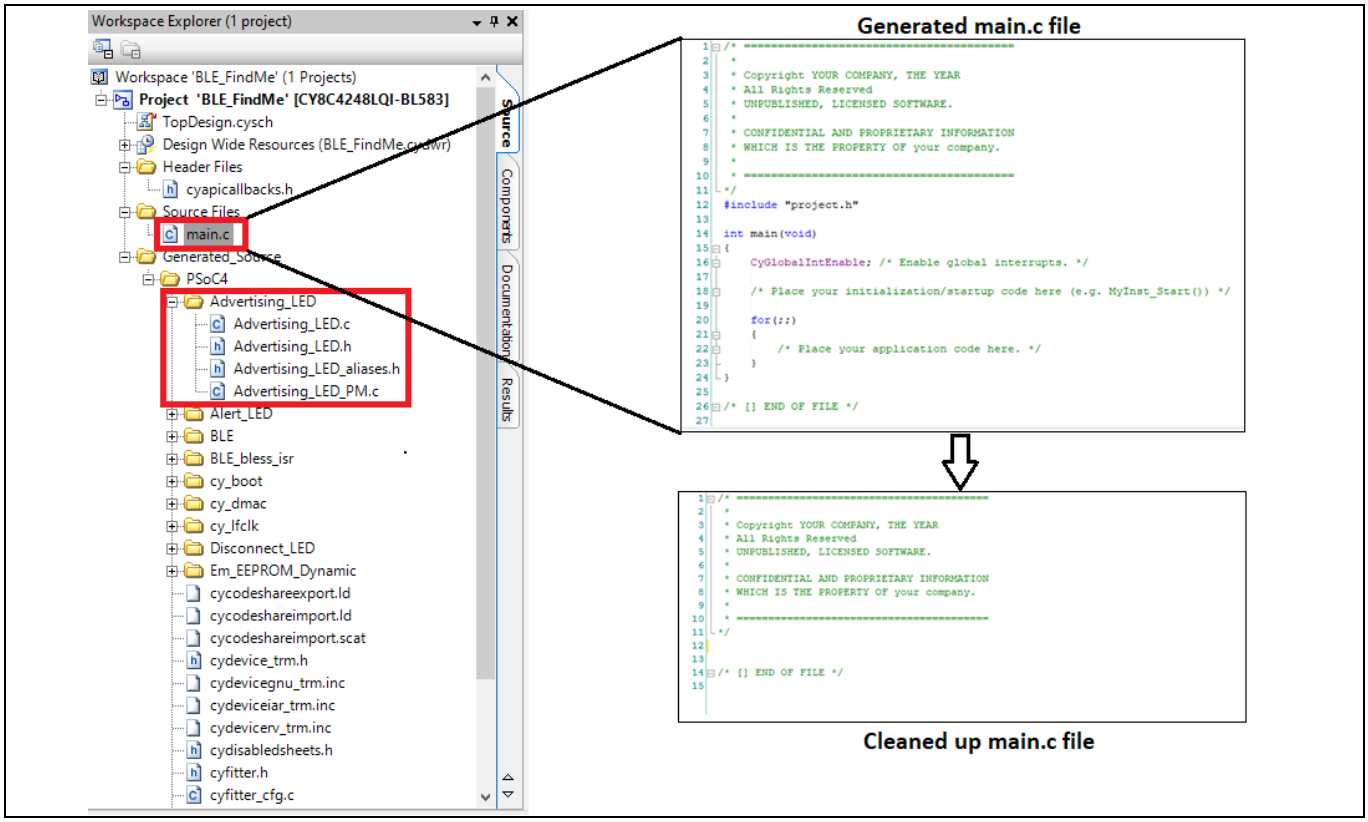


Figure 42　生成されたソース ファイル

7.4　第 2 段階: ファームウェアを書く

PSoC™ Creator で Bluetooth® LE 標準プロファイルのアプリケーションを設計するには、以下の 4 つの主なファームウェア ブロックが必要です。

- システム初期化
- Bluetooth® LE スタック イベント ハンドラ
- Bluetooth® LE サービス固有イベント ハンドラ
- メイン ループおよび低消費電力実装

以下の節で、「**第 1 段階: 設計を作成**」で設定した設計の各ブロックを説明します。本節で提供されたコード スニペットは、**Figure 42** に示した main.c ファイルに入ってあります。

7.4.1　システム初期化

PSoC™ 4 with AIROC™ Bluetooth® LE デバイスがリセットされると、ファームウェアはまずシステムの初期化を実行します。このプロセスでは、グローバル割り込みおよび設計で使用するコンポーネントを有効にします。システムが初期化された後、ファームウェアは Bluetooth® LE コンポーネントを初期化し、その結果、Bluetooth® LE サブシステム全体が内部的に初期化されます。

Bluetooth® LE コンポーネントの初期化プロセスの一部として、保留中のイベントを通知するために Bluetooth® LE スタックによって呼び出されるイベント ハンドラ関数を登録する必要があります。**Code Listing 2** に示す Bluetooth® LE スタック イベント ハンドラは Bluetooth® LE の初期化の一部として登録されます。Bluetooth® LE コンポーネントの初期化が正常に完了すると、ファームウェアは Immediate Alert Service (IAS) 固有イベント用のイベント ハンドラを登録し、制御権をメイン ループに渡します。**Figure 43** と **Code Listing 1** に、システム初期化のフロー図とファームウェア ソースコードを示します。

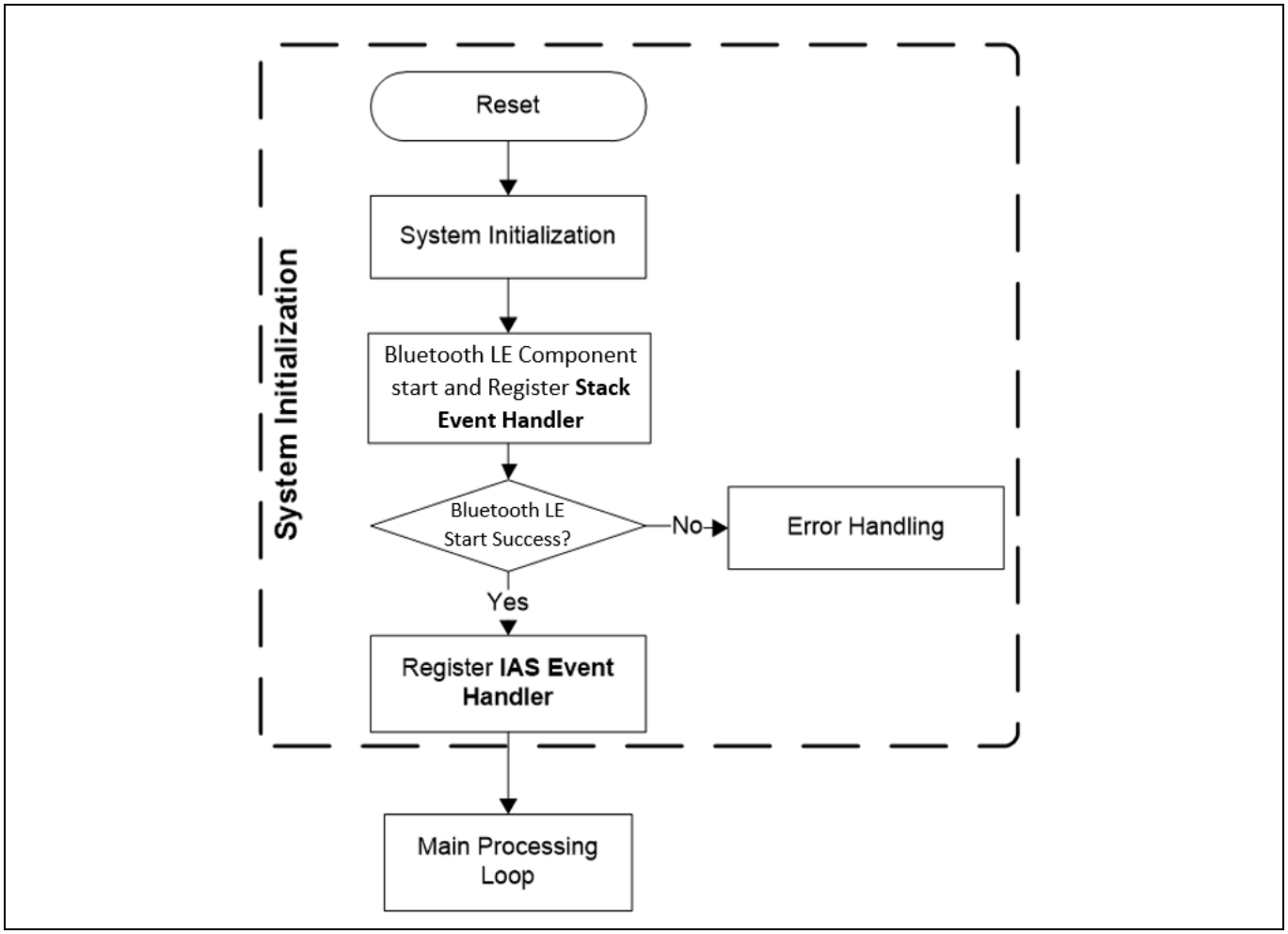


Figure 43　システム初期化のフロー図

Code Listing 1　　システム初期化のファームウェア

```
#include <project.h>

#define LED_ON                  (0u)
#define LED_OFF                 (1u)

#define NO_ALERT                (0u)
#define MILD_ALERT              (1u)
#define HIGH_ALERT              (2u)

#define LED_TOGGLE_TIMEOUT      (100u)

void StackEventHandler(uint32 event, void *eventParam);
```

Code Listing 1　システム初期化のファームウェア

```
void IasEventHandler(uint32 event, void *eventParam);

uint8 alertLevel;

int main()
{
  CYBLE_API_RESULT_T apiResult;

  CyGlobalIntEnable;

  apiResult = CyBle_Start(StackEventHandler);

  if(apiResult != CYBLE_ERROR_OK)
  {
    /* BLE stack initialization failed, check your configuration */
    CYASSERT(0);
  }

  CyBle_IasRegisterAttrCallback(IasEventHandler);

  /* Place the main application loop here */
}
```

7.4.2　Bluetooth® LE スタック イベント ハンドラ

Bluetooth® LE コンポーネント内の Bluetooth® LE スタックはイベントを生成し、ユーザーによって登録された Bluetooth® LE スタック イベント ハンドラを介して Bluetooth® LE インターフェースの状態とデータをアプリケーション ファームウェアに提供します。イベント ハンドラは、デバイス接続やスタック オンなどいくつかのスタックからの基本的なイベントを処理し、それに応じてスタックを設定して Bluetooth® LE 接続を確立および維持します。ご作成の Find Me Target アプリケーションでは、Bluetooth® LE スタック イベント ハンドラは **Table 1** で説明するすべてのイベントを処理する必要があります。**Figure 44** と **Code Listing 2** に、Bluetooth® LE スタック イベント処理のフロー図とファームウェアを示します。

Table 1　Bluetooth® LE スタック イベント

Bluetooth® LE スタック イベント名	イベントの説明	イベント ハンドラの応答処理
CYBLE_EVT_STACK_ON	Bluetooth® LE スタックの初期化が正常に完了した	宣伝を開始し、宣伝状態を LED に反映
CYBLE_EVT_GAP_DEVICE_DISCONNECTED	ピア デバイスとの Bluetooth® LE 接続が切断される	宣伝を再開し、宣伝状態を LED に反映
CYBLE_EVT_GAP_DEVICE_CONNECTED	ピア デバイスとの Bluetooth® LE 接続が確立される	Bluetooth® LE 接続状態を LED に更新
CYBLE_EVT_GAPP_ADVERTISEMENT_START_STOP	Bluetooth® LE スタックの宣伝開始／停止イベント	宣伝がタイムアウトした場合、デバイスをストップ モードに設定

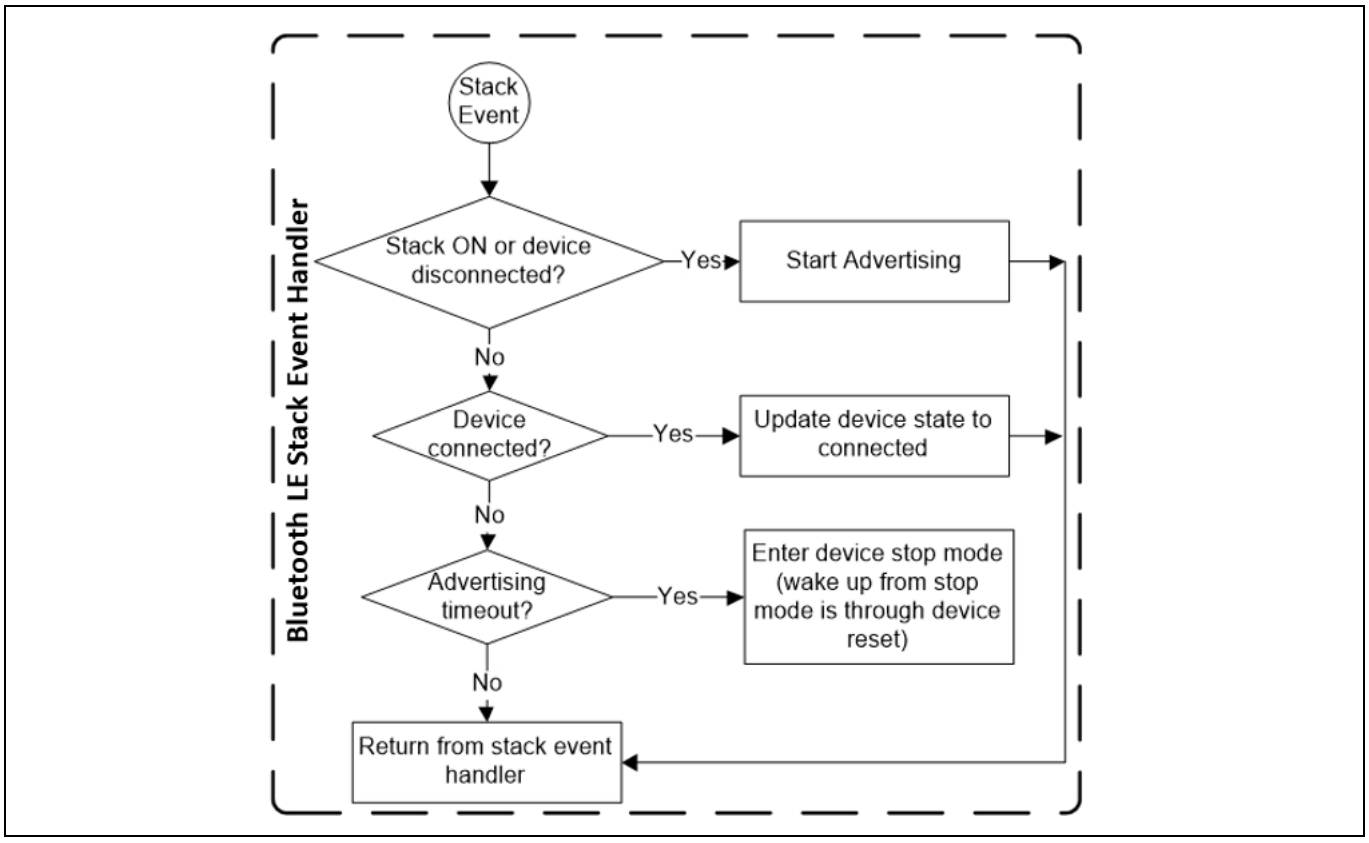


Figure 44　Bluetooth® LE スタック イベント ハンドラのフロー図

Code Listing 2　Bluetooth® LE スタック イベント ハンドラのファームウェア

```
void StackEventHandler(uint32 event, void *eventParam)
{
    switch(event)
    {
```

Code Listing 2　Bluetooth® LE スタック イベント ハンドラのファームウェア

```
        /* Mandatory events to be handled by Find Me Target design */
        case CYBLE_EVT_STACK_ON:
        case CYBLE_EVT_GAP_DEVICE_DISCONNECTED:
            /* Start BLE advertisement for 30 seconds and update link
             * status on LEDs */
            CyBle_GappStartAdvertisement(CYBLE_ADVERTISING_FAST);
            Advertising_LED_Write(LED_ON);
            alertLevel = NO_ALERT;
        break;

        case CYBLE_EVT_GAP_DEVICE_CONNECTED:
            /* BLE link is established */
            Advertising_LED_Write(LED_OFF);
            Disconnect_LED_Write(LED_OFF);
        break;

        case CYBLE_EVT_GAPP_ADVERTISEMENT_START_STOP:
            if(CyBle_GetState() == CYBLE_STATE_DISCONNECTED)
            {
                /* Advertisement event timed out, go to low power
                 * mode (Stop mode) and wait for device reset
                 * event to wake up the device again */
                Advertising_LED_Write(LED_OFF);
                Disconnect_LED_Write(LED_ON);
                CySysPmSetWakeupPolarity(CY_PM_STOP_WAKEUP_ACTIVE_HIGH);
                CySysPmStop();

                /* Code execution will not reach here */
            }
        break;

        default:
        break;
    }
}
```

7.4.3 Bluetooth® LE サービス固有イベント ハンドラ

Bluetooth® LE コンポーネントはまた、設計でサポートされる各サービスに対応するイベントを生成します。ご作成中の Find Me Target アプリケーションでは、Bluetooth® LE コンポーネントはアラート レベルのキャラクタリスティックが新しい値に更新されるかをアプリケーションに通知する IAS イベントを生成します。**Figure 45** と **Code Listing 3** に、Bluetooth® LE IAS イベント処理のフロー図とファームウェアを示します。

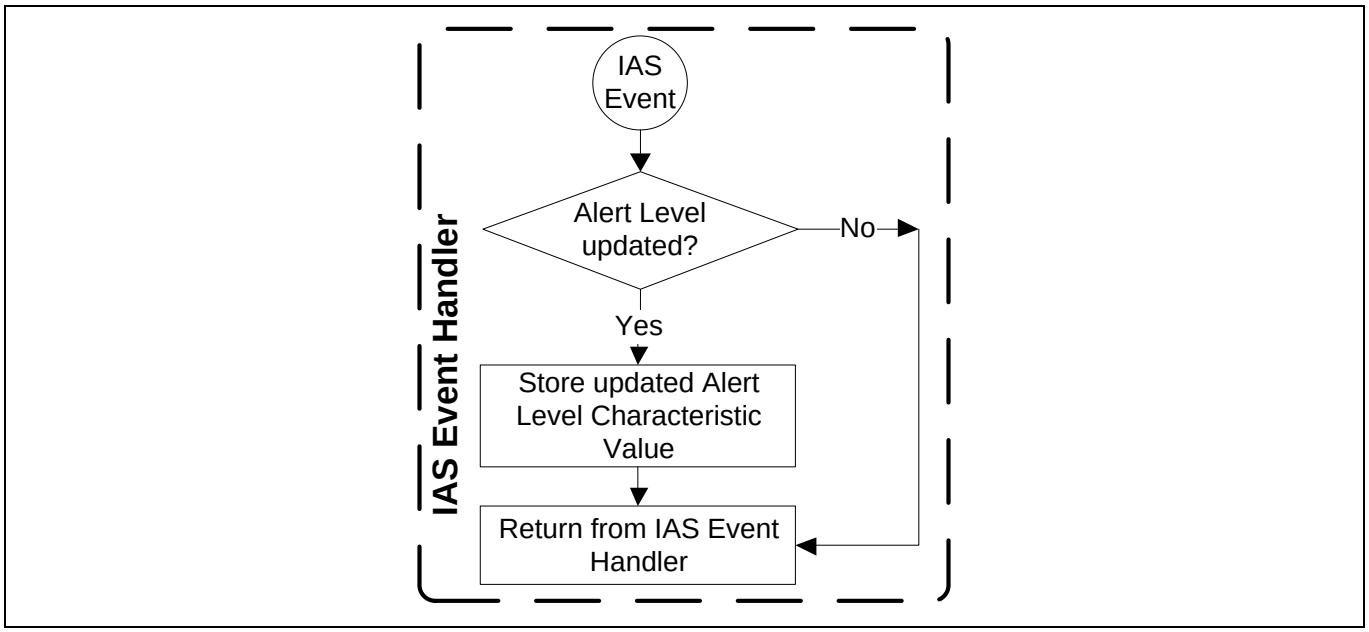


Figure 45 Bluetooth® LE IAS イベント ハンドラのフロー図

Code Listing 3 Bluetooth® LE IAS イベント ハンドラのファームウェア

```
void IasEventHandler(uint32 event, void *eventParam)
{
    /* Alert Level Characteristic write event */
    if(event == CYBLE_EVT_IASS_WRITE_CHAR_CMD)
    {
        /* Read the updated Alert Level value from the GATT database */
        CyBle_IassGetCharacteristicValue(CYBLE_IAS_ALERT_LEVEL,
            sizeof(alertLevel), &alertLevel);
    }
}
```

7.4.4 メインループおよび低消費電力実装

設計のメイン ループ ファームウェアは、Bluetooth® LE スタック処理イベントを定期的に処理し、IAS のアラート レベルのキャラクタリスティック値に応じて青色 LED を更新し、**Bluetooth® LE サブシステム (BLESS)** ブロックと PSoC™ 4 with AIROC™ Bluetooth® LE デバイスを低消費電力モードに設定します。**Figure 46** と **Code Listing 4** に、メイン ループのフロー図とファームウェアを示します。IasEventHandler 関数コールバックが登録された後、**Code Listing 4** 内のコード ブロックを **Code Listing 1** に示されるメイン関数内に入れる必要があります。

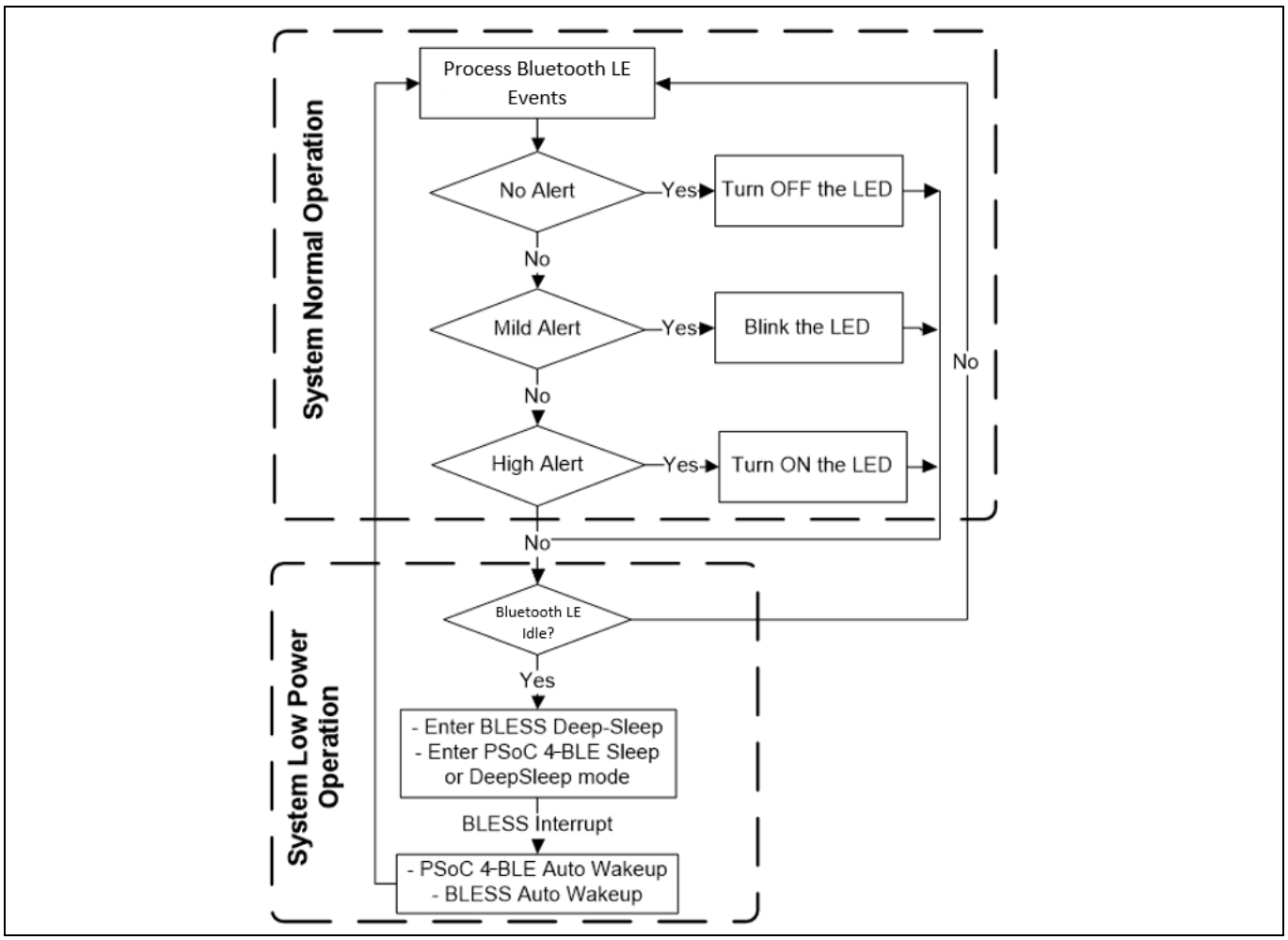


Figure 46 ファームウェア メイン ループのフロー図

Code Listing 4 メイン ループのファームウェア

```
/* Place the main application loop here */


for(;;)
{
    static uint8 toggleTimeout = 0;
    CYBLE_BLESS_STATE_T blessState;
```

Code Listing 4　メイン ループのファームウェア

```
uint8 intrStatus;

/* Single API call to service all the BLE stack events. Must be
 * called at least once in a BLE connection interval */
                                    CyBle_ProcessEvents();
/* Update Alert Level value on the blue LED */
switch(alertLevel)
{
    case NO_ALERT:
    Alert_LED_Write(LED_OFF);
    break;

    case MILD_ALERT:
    toggleTimeout++;
    if(toggleTimeout == LED_TOGGLE_TIMEOUT)
    {
        /* Toggle alert LED after timeout */
        Alert_LED_Write(Alert_LED_Read() ^ 0x01);
        toggleTimeout = 0;
    }
    break;

    case HIGH_ALERT:
    Alert_LED_Write(LED_ON);
    break;
}

/* Configure BLESS in Deep-Sleep mode */
CyBle_EnterLPM(CYBLE_BLESS_DEEPSLEEP);

/* Prevent interrupts while entering system low power modes */
intrStatus = CyEnterCriticalSection();

/* Get the current state of BLESS block */
blessState = CyBle_GetBleSsState();
```

Code Listing 4　メイン ループのファームウェア

```
    /* If BLESS is in Deep-Sleep mode or the XTAL oscillator is turning on,
     * then PSoC™ 4 BLE can enter Deep-Sleep mode (1.3uA current consumption) */
    if(blessState == CYBLE_BLESS_STATE_ECO_ON ||
        blessState == CYBLE_BLESS_STATE_DEEPSLEEP)
    {
        CySysPmDeepSleep();
    }
    else if(blessState != CYBLE_BLESS_STATE_EVENT_CLOSE)
    {
        /* If BLESS is active, then configure PSoC™ 4 BLE system in
         * Sleep mode (~1.6mA current consumption) */
        CySysPmSleep();
    }
    else
    {
        /* Keep trying to enter either Sleep or Deep-Sleep mode */
    }
    CyExitCriticalSection(intrStatus);

    /* BLE link layer timing interrupt will wake up the system from Sleep
     * and Deep-Sleep modes */
}
```

7.5　第 3 段階: デバイスをプログラム

本節は、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスをプログラムする方法を説明します。プログラマ内蔵の開発キット(Bluetooth® LE Pioneer Kit など) を使用する場合、USB ケーブルでキット基板をコンピュータに接続します。その他のキットについては、キット ガイドを参照ください。自分のハードウェアで開発する場合、インフィニオン **CY8CKIT-002 MiniProg3** などのハードウェア デバッガが必要です。

Note:　*本設計のソース プロジェクトは、PSoC™ Creator 3.3 SP1 またはそれ以降のバージョンで見つかります(**File > Code Example** の下で **Filter by** を **Find Me > BLE_FindMe** に選択します。)*

1. **Figure 47** に示すように、PSoC™ Creator で **Debug** > **Select Debug Target** を選択します。

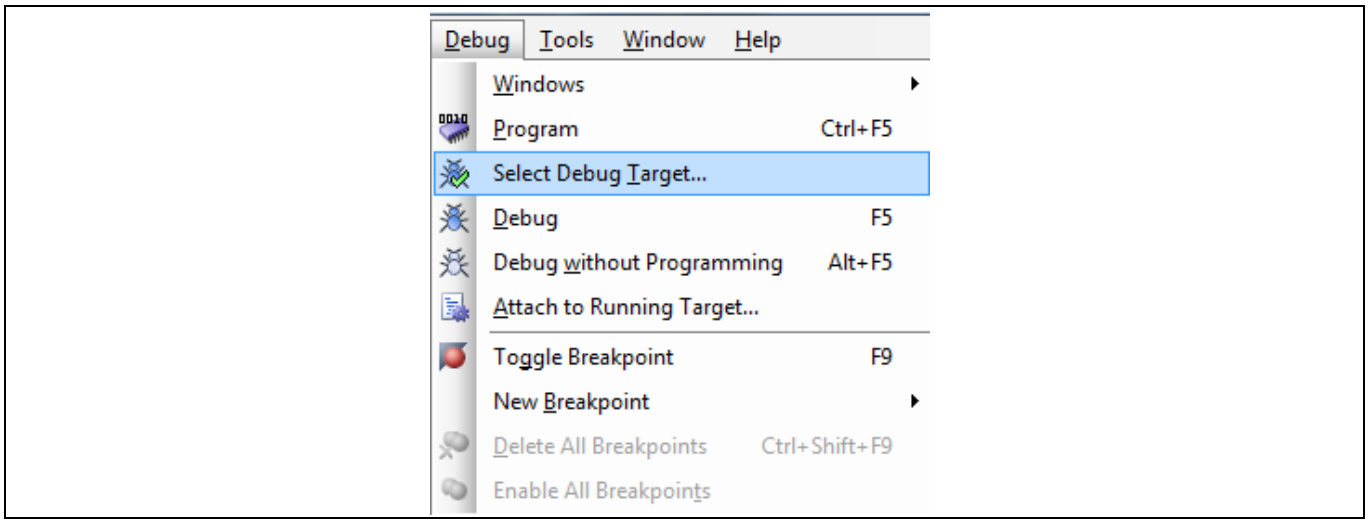


Figure 47　デバッグ ターゲットの選択

2. **Figure 48** に示すように、**Select Debug Target** ダイアログ ボックスで **Port Acquire** をクリックして **Connect** をクリックします。**OK** をクリックしてダイアログ ボックスを閉じます。

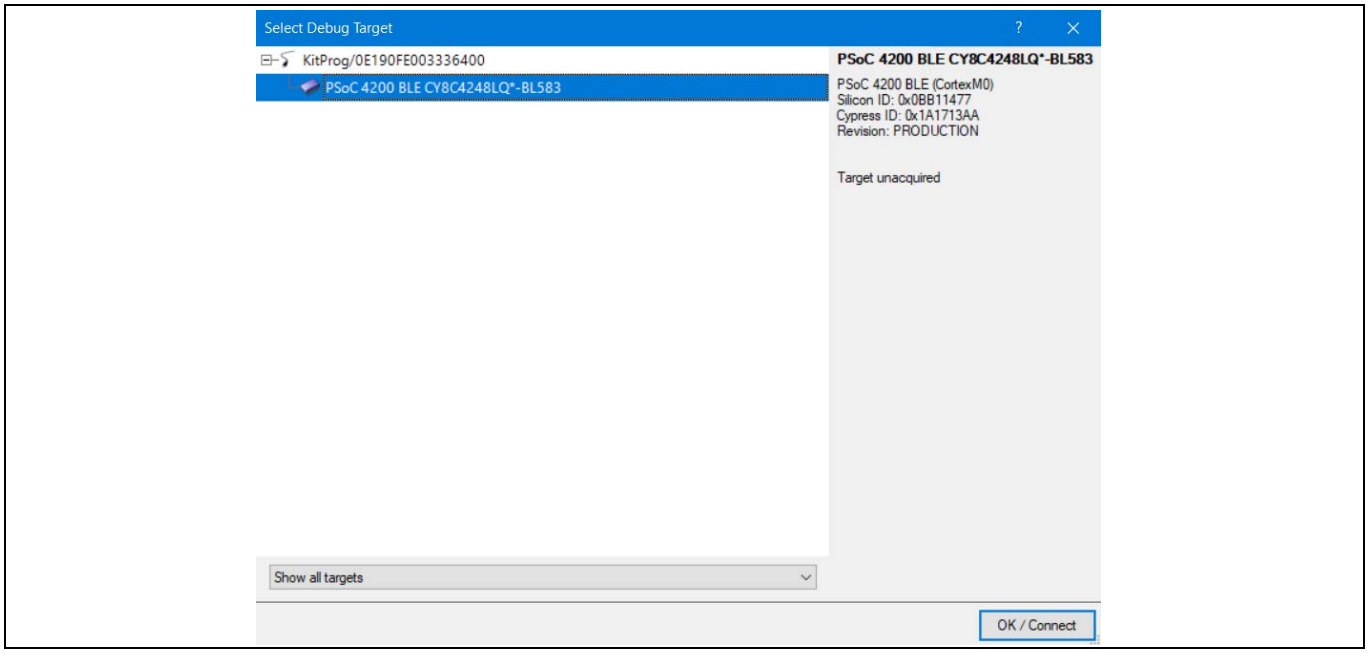


Figure 48　デバイスの接続

自分のハードウェアを使用する場合、プログラミング用ハードウェアの **Select Debug Target** ウィンドウの **Port Setting** コンフィギュレーションがセットアップの通りに設定されたことをご確認ください。

3. **Figure 49** に示すように、PSoC™ Creator で **Debug**> **Program** を選択してプロジェクトをデバイスにプログラムします。

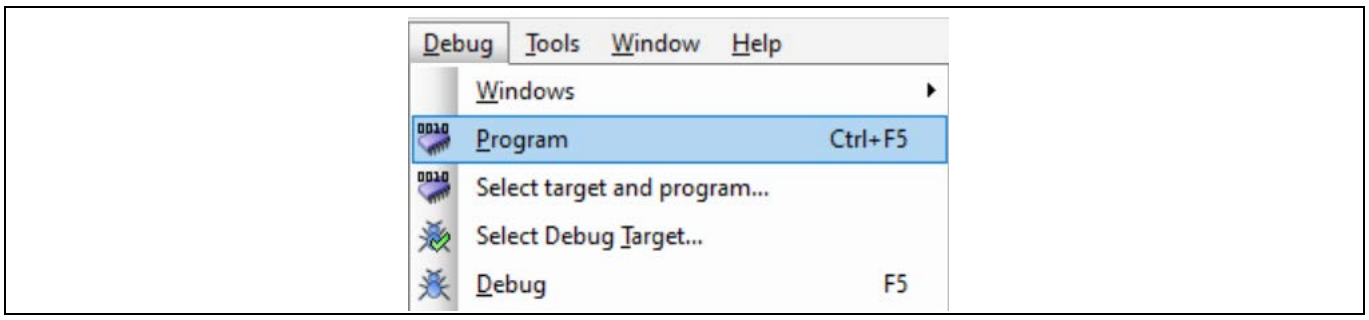


Figure 49 デバイスのプログラミング

4. **Figure 50** に示すように、PSoC™ Creator ステータス バー (画面の左下隅) でプログラミングの状態を確認できます。

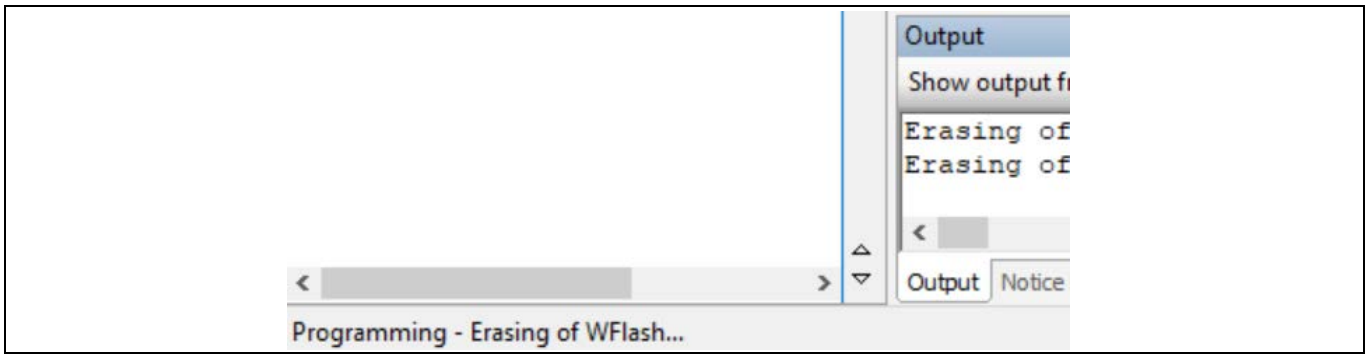


Figure 50 プログラミングの状態

7.6 第 4 段階: 設計をテスト

本節で、CySmart モバイル アプリと PC ツールを用いて Bluetooth® LE 設計をテストする方法を示します。**Figure 19** に、Bluetooth® LE Pioneer Kit を用いた設計テストのセットアップを示します。

このツールの詳細については以下の付録を参照ください。

- **CySmart ホスト エミュレーション ツール**
- **CySmart モバイル アプリ**

7.6.1 CySmart モバイル アプリ

1. ご使用の iOS または Android モバイル機器の Bluetooth 機能をオンにします。
2. CySmart アプリを起動します。
3. Bluetooth® LE Pioneer Kit 上のリセット スイッチを押してご開発の設計からの Bluetooth® LE 宣伝を開始します。
4. CySmart アプリのホーム画面を引き下ろし、Bluetooth® LE ペリフェラルのスキャンを開始します。これで、ご使用のデバイスは CySmart アプリのホーム画面に表示されます。ご使用のデバイスを選択して Bluetooth® LE 接続を確立します。
5. カルーセル画面で「Find Me」プロファイルを選択します。

6. Find Me **Profile** 画面でアラート レベルの 1 つを選択します。ご覧のように、デバイス上の LED 状態が選択に応じて変化します。

Figure 51 と **Figure 52** に、CySmart モバイル アプリの設定手順を示します。

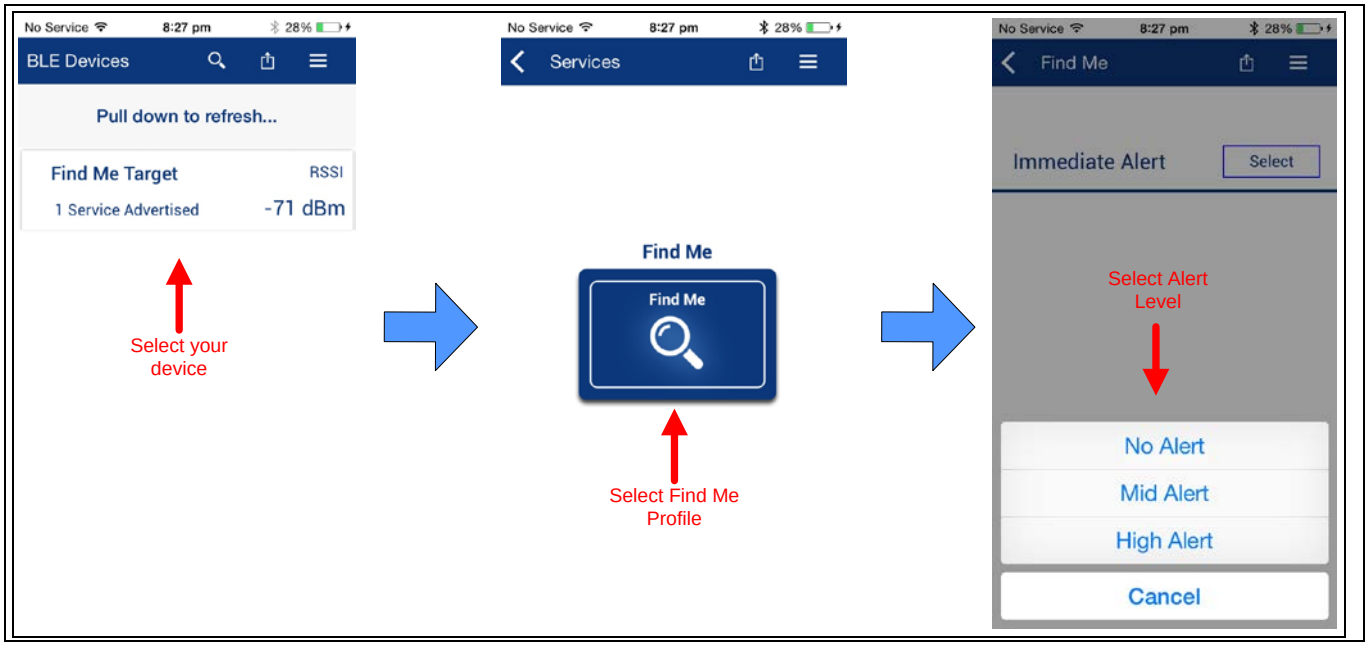


Figure 51　iOS 向け CySmart アプリでのテスト

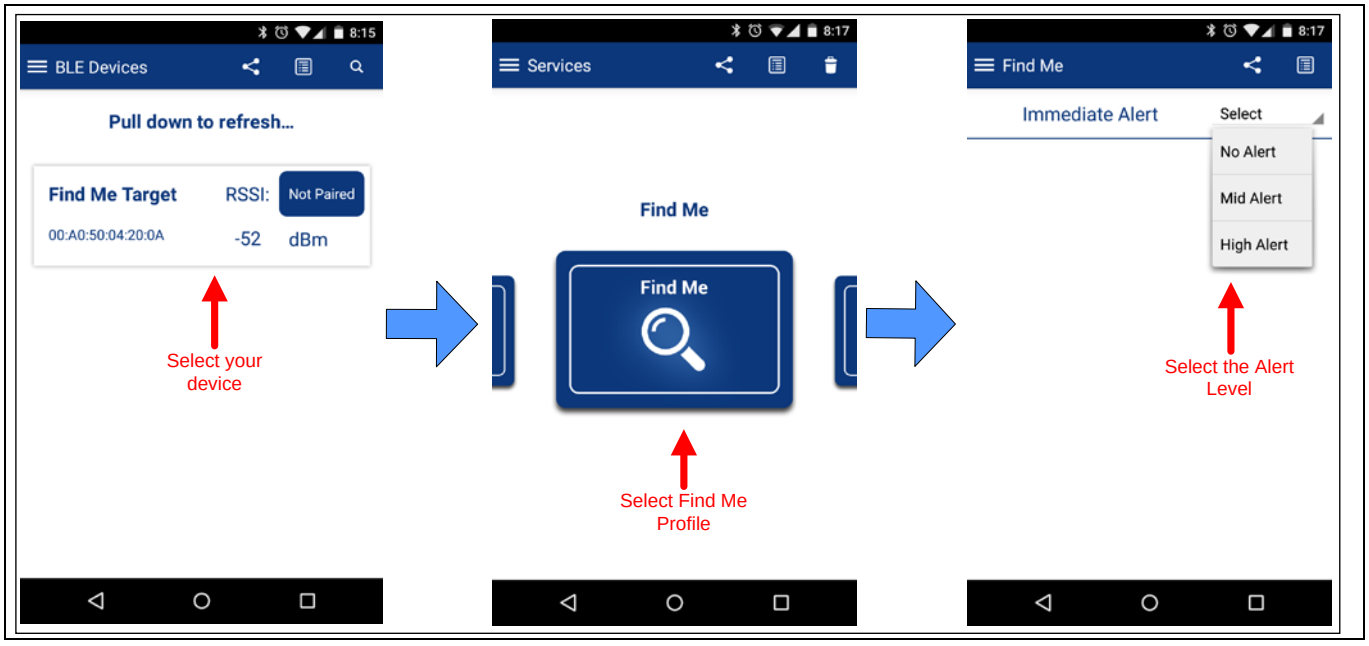


Figure 52　Android 向け CySmart アプリでのテスト

7.6.2 CySmart ホスト エミュレーション ツール

CySmart モバイル アプリと同様に、CySmart ホスト エミュレーション ツールを使用してご開発の設計との Bluetooth® LE 接続を確立し、Bluetooth® LE キャラクタリスティックに対して読み書き処理を実行できます。

1. Bluetooth® LE ドングルをご使用の Windows マシンに接続します。ドライバーのインストールが完了するまで待ちます。
2. CySmart ホスト エミュレーション ツールを起動します。ツールは自動的に Bluetooth® LE ドングルを検出します。Bluetooth® LE ドングルが **Select BLE Dongle Target** ポップアップ ウィンドウに表示されない場合、**Refresh** をクリックします。**Figure 53** に示すように **Connect** をクリックします。

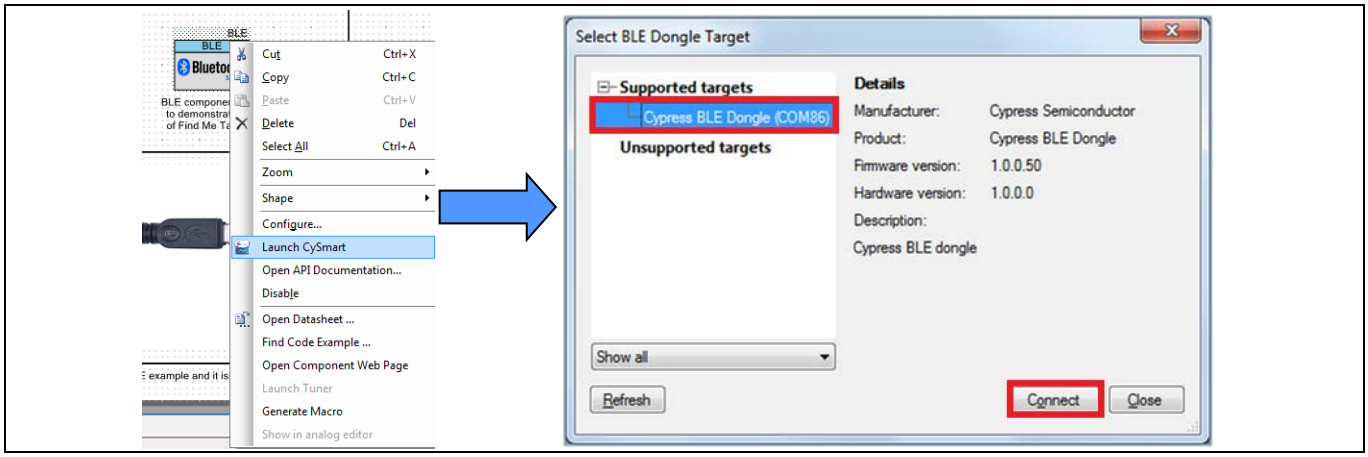


Figure 53 CySmart での Bluetooth® LE ドングル選択

3. **Figure 54** に示すように、**Configure Master Settings** を選択して初期設定値を復元します。

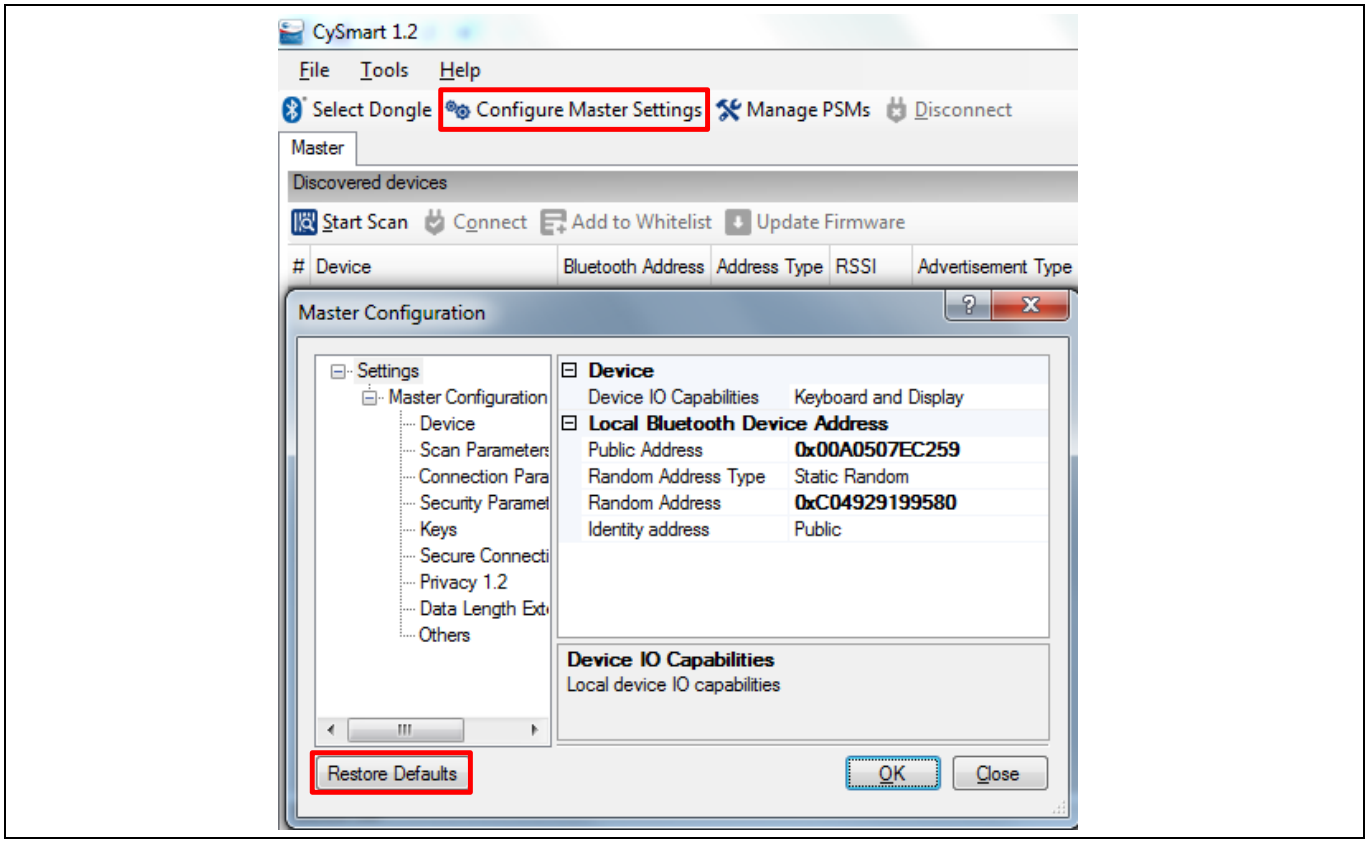


Figure 54　CySmart のマスター設定のコンフィギュレーション

4. Bluetooth® LE Pioneer Kit 上のリセット スイッチを押してご開発の設計からの Bluetooth® LE 宣伝を開始します。
5. CySmart ホスト エミュレーション ツールで **Start Scan** をクリックします。**Figure 55** に示すように、ご使用のデバイスが **Discovered devices** リストに表示されます。

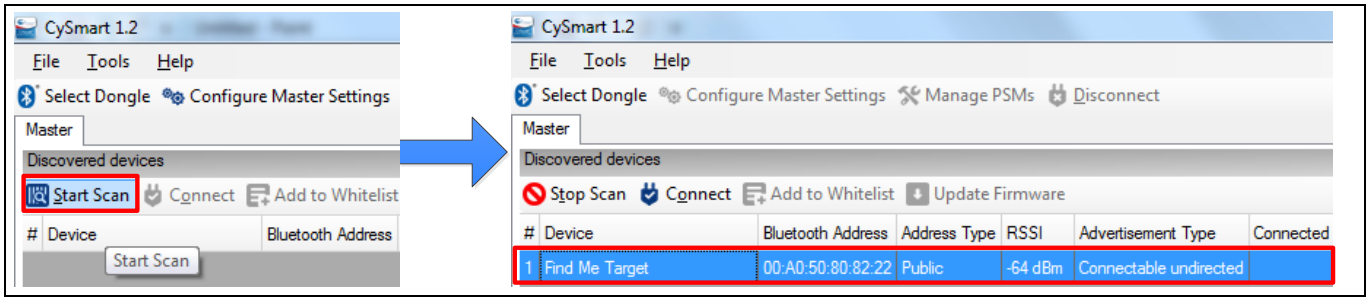


Figure 55　CySmart でのデバイス検出

6. **Figure 56** に示すように、ご使用のデバイスを選択し、**Connect** をクリックして CySmart ホスト エミュレーション ツールとの Bluetooth® LE 接続を確立します。

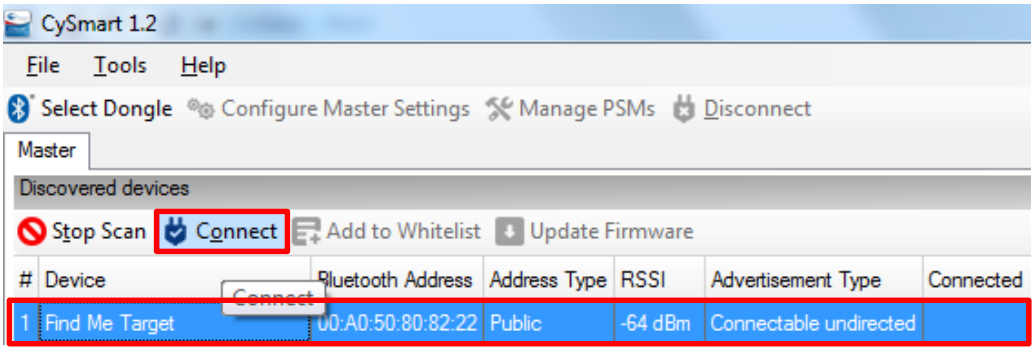


Figure 56 CySmart でのデバイス接続

7. 接続が確立された後、**Figure 57** に示すように、CySmart ホスト エミュレーション ツールでご開発の設計のすべてのアトリビュートを検出します。

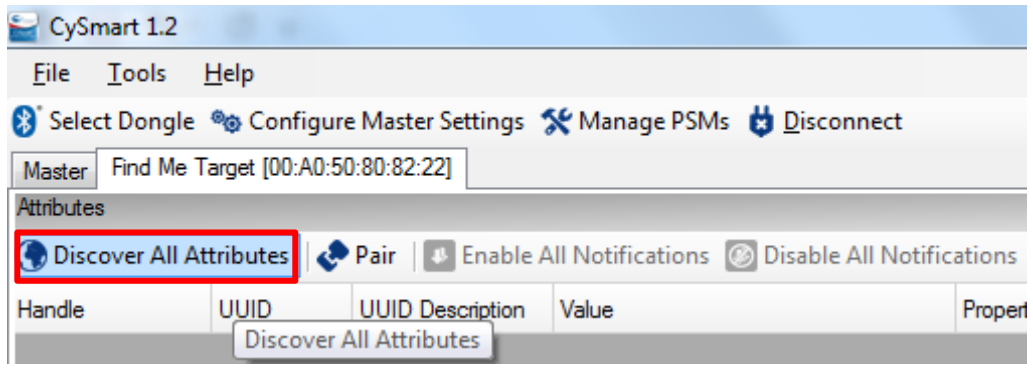


Figure 57 CySmart でのアトリビュート検出

8. **Figure 58** に示すように、Immediate Alert Service の下の Alert Level キャラクタリスティックに値 0, 1, 2 のいずれかを書き込みます。ご覧のように、ご使用のデバイス上の LED 状態が Alert Level キャラクタリスティックの設定に応じて変化します。

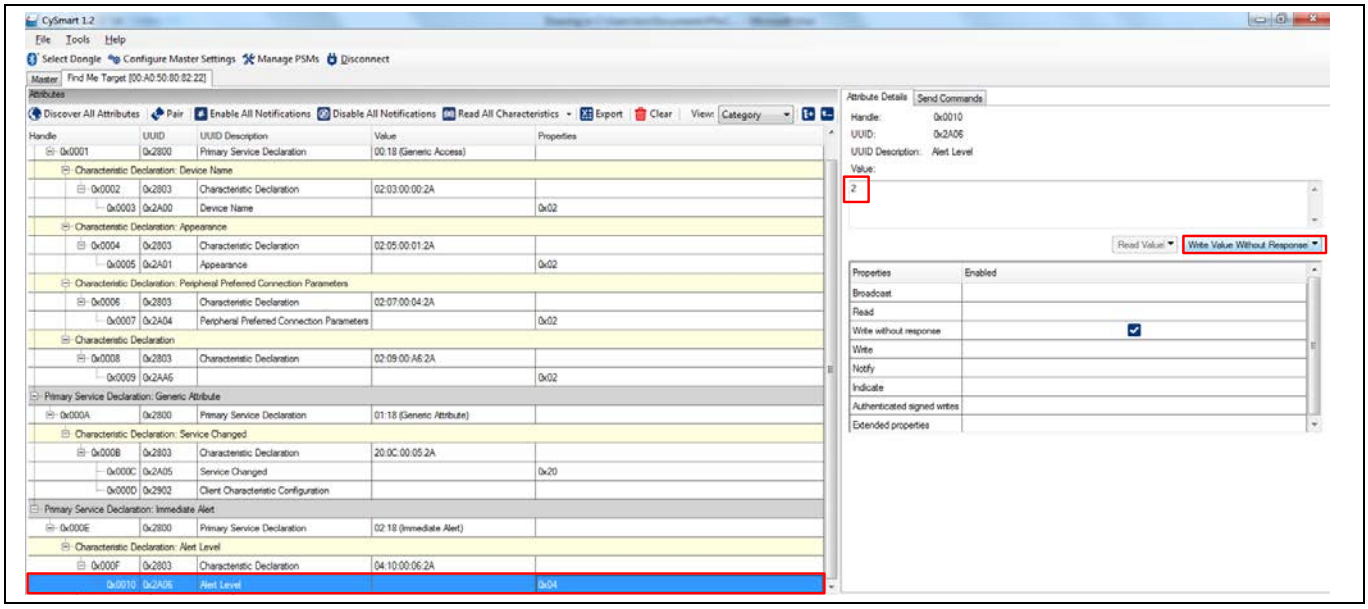


Figure 58　CySmart ホスト エミュレーション ツールでのテスト

8 まとめ

本アプリケーション ノートは Bluetooth® LE プロトコルと PSoC™ 4 with AIROC™ Bluetooth® LE デバイスのアーキテクチャおよび開発ツールの基礎について説明しました。PSoC™ 4 with AIROC™ Bluetooth® LE は、Bluetooth® LE 無線ブロック, 設定可能なアナログとデジタルのペリフェラル機能, メモリ, および Arm® Cortex®-M0 マイクロコントローラーを単一のチップに集積した真のプログラマブル組込みシステムオンチップです。統合された機能および低消費電力モードにより、PSoC™ 4 with AIROC™ Bluetooth® LE はバッテリ駆動ウェアラブル機器、健康関連製品の Bluetooth® LE アプリケーションに最適になります。

本アプリケーション ノートはまた、PSoC™ 4 with AIROC™ Bluetooth® LE を迅速に深く理解できるように総合的なリソース収集へのアクセスも提供しました。

9 付録 A: Bluetooth® LE デバイス ファミリの比較

Table 2 はインフィニオンの Bluetooth® LE デバイス ファミリの特長と機能をまとめます。

Table 2 Bluetooth® LE デバイス ファミリ

機能	デバイス ファミリ				
	CY8C41x7-BLxxx	**CY8C42x7-BLxxx**	**CYBL10xxx*** **	**CY8C41x8-BL**	**CY8C42x8-BL**
Bluetooth® LE サブシステム	Bluetooth 4.1 準拠のプロトコル スタックに対応している Bluetooth® LE 無線と Link Layer ハードウェア ブロック	Bluetooth 4.1 準拠のプロトコル スタックに対応している Bluetooth® LE 無線と Link Layer ハードウェア ブロック	Bluetooth 4.2 準拠のプロトコル スタックに対応している Bluetooth® LE 無線と Link Layer ハードウェア ブロック	Bluetooth 4.2 準拠のプロトコル スタックに対応している Bluetooth® LE 無線と Link Layer ハードウェア ブロック **	Bluetooth 4.2 準拠のプロトコル スタックに対応している Bluetooth® LE 無線と Link Layer ハードウェア ブロック**
Bluetooth 4.2 機能	LE セキュア接続	LE セキュア接続	LE セキュア接続、Link Layer プライバシーと Link Layer データ長拡張	LE セキュア接続、Link Layer プライバシーと Link Layer データ長拡張 **	LE セキュア接続、Link Layer プライバシーと Link Layer データ長拡張**
CPU	24 MHz Arm® Cortex®-M0 CPU、シングルサイクルの乗算に対応	48 MHz Arm® Cortex®-M0 CPU、シングル サイクルの乗算に対応	48 MHz Arm® Cortex®-M0 CPU、シングル サイクルの乗算に対応	24 MHz Arm® Cortex®-M0 CPU、シングルサイクルの乗算に対応	48 MHz Arm® Cortex®-M0 CPU、シングル サイクルの乗算に対応
フラッシュ メモリ	128 KB	128 KB	256 KB	256 KB	256 KB
SRAM	16 KB	16 KB	32 KB	32 KB	32 KB
GPIO	最大 36	最大 36	最大 36	最大 36	最大 36
CapSense™	最大 35 個のセンサー	最大 35 個のセンサー	最大 35 個のセンサー	最大 35 個のセンサー	最大 35 個のセンサー
CapSense™ ジェスチャー	一部のデバイスのみ	一部のデバイスのみ	一部のデバイスのみ	一部のデバイスのみ	一部のデバイスのみ
ADC	シーケンサを備えた 12 ビット、806 ksps SAR ADC	シーケンサを備えた 12 ビット、1 Msps SAR ADC	シーケンサを備えた 12 ビット、1 Msps SAR ADC	シーケンサを備えた 12 ビット、806 ksps SAR ADC	シーケンサを備えた 12 ビット、1 Msps SAR ADC
オペアンプ	ディープスリープ モードでアクティブな 2 個のプログラマブルなオペアンプ	ディープスリープ モードでアクティブな 4 個のプログラマブルなオペアンプ	なし	ディープスリープ モードでアクティブな 2 個のプログラマブルなオペアンプ	ディープスリープ モードでアクティブな 4 個のプログラマブルなオペアンプ

付録 A: Bluetooth® LE デバイス ファミリの比較

機能	デバイス ファミリ				
	CY8C41x7-BLxxx	**CY8C42x7-BLxxx**	**CYBL10xxx*** **	**CY8C41x8-BL**	**CY8C42x8-BL**
コンパレータ	ウェイクアップ機能を備えた 2 個の低消費電力コンパレータ	ウェイクアップ機能を備えた 2 個の低消費電力コンパレータ	なし	ウェイクアップ機能を備えた 2 個の低消費電力コンパレータ	ウェイクアップ機能を備えた 2 個の低消費電力コンパレータ
電流 DAC	7 ビット (1 個)、8 ビット (1 個)	7 ビット (1 個)、8 ビット (1 個)	なし	7 ビット (1 個)、8 ビット (1 個)	7 ビット (1 個)、8 ビット (1 個)
電源範囲	1.9V～5.5V	1.9V～5.5V	1.9V～5.5V	1.9V～5.5V	1.9V～5.5V
低消費電力モード	ディープ スリープ モード: 1.3 µA の消費電流 ハイバネート モード: 150 nA の消費電流 ストップ モード: 60 nA の消費電流	ディープ スリープ モード: 1.3 µA の消費電流 ハイバネート モード: 150 nA の消費電流 ストップ モード: 60 nA の消費電流	ディープ スリープ モード: 1.3 µA の消費電流 ハイバネート モード: 150 nA の消費電流 ストップ モード: 60 nA の消費電流	ディープ スリープ モード: 1.3 µA の消費電流 ハイバネート モード: 150 nA の消費電流 ストップ モード: 60 nA の消費電流	ディープ スリープ モード: 1.3 µA の消費電流 ハイバネート モード: 150 nA の消費電流 ストップ モード: 60 nA の消費電流
セグメント LCD 駆動	一部のデバイスでの 4 COM、32 セグメントの LCD 駆動	一部のデバイスでの 4 COM、32 セグメントの LCD 駆動	一部のデバイスでの 4 COM、32 セグメントの LCD 駆動	一部のデバイスでの 4 COM、32 セグメントの LCD 駆動	一部のデバイスでの 4 COM、32 セグメントの LCD 駆動
シリアル通信	I^2C／SPI／UART にプログラム可能な 2 個の独立したシリアル通信ブロック (SCB)	I^2C／SPI／UART にプログラム可能な 2 個の独立したシリアル通信ブロック (SCB)	I^2C／SPI／UART にプログラム可能な 2 個の独立したシリアル通信ブロック (SCB)	I^2C／SPI／UART にプログラム可能な 2 個の独立したシリアル通信ブロック (SCB)	I^2C／SPI／UART にプログラム可能な 2 個の独立したシリアル通信ブロック (SCB)
タイマー／カウンター／パルス幅変調器 (TCPWM)	4	4	4	4	4

付録 A: Bluetooth® LE デバイス ファミリの比較

機能	デバイス ファミリ				
	CY8C41x7-BLxxx	**CY8C42x7-BLxxx**	**CYBL10xxx*** **	**CY8C41x8-BL**	**CY8C42x8-BL**
ユニバーサル デジタル ブロック (UDB)	なし	4 個、各々は 8 つのマクロセルおよび 1 つのデータパスを備える。追加のデジタル ペリフェラル (タイマー、カウンター、PWM) または通信インターフェース (UART、SPI) を合成するのに使用可能	なし	なし	4 個、各々は 8 つのマクロセルおよび 1 つのデータパスを備える。追加のデジタル ペリフェラル (タイマー、カウンター、PWM) または通信インターフェース (UART、SPI) を合成するのに使用可能
追加のデジタル ペリフェラル (I^2S、PWM)	なし	あり (一部のデバイスでの UDB ベースのデジタル ペリフェラル)	あり (一部のデバイスでの固定機能ブロック)	なし	あり (一部のデバイスでの UDB ベースのデジタル ペリフェラル)
クロック	IMO: 3 MHz～24 MHz ILO: 32 kHz ECO: 24 MHz WCO: 32 kHz	IMO: 3 MHz～48 MHz ILO: 32 kHz ECO: 24 MHz WCO: 32 kHz	IMO: 3 MHz～48 MHz ILO: 32 kHz ECO: 24 MHz WCO: 32 kHz	IMO: 3 MHz～24 MHz ILO: 32 kHz ECO: 24 MHz WCO: 32 kHz	IMO: 3 MHz～48 MHz ILO: 32 kHz ECO: 24 MHz WCO: 32 kHz
電源電圧監視	パワーオン リセット (POR) 電圧低下検出 (BOD) 低電圧検出 (LVD)	POR BOD LVD	POR BOD LVD	パワーオン リセット (POR) 電圧低下検出 (BOD) 低電圧検出 (LVD)	POR BOD LVD
パッケージ	56-QFN (7.0 × 7.0 × 0.6mm) および 68-WLCSP (3.52 × 3.91 × 0.55mm)	56-QFN (7.0 × 7.0 × 0.6mm) および 68-WLCSP (3.52 × 3.91 × 0.55mm)	56-QFN* (7.0 × 7.0 × 0.6mm) および 76-WLCSP (4.04 × 3.87 × 0.55mm)	56-QFN* (7.0 × 7.0 × 0.6mm) および 76-WLCSP (4.04 × 3.87 × 0.55mm)	56-QFN* (7.0 × 7.0 × 0.6mm) および 76-WLCSP (4.04 × 3.87 × 0.55mm)
DMA	なし	なし	最大 8 本のチャネル	最大 8 本のチャネル**	最大 8 本のチャネル**

* = CY8C41x8-BL と CY8C42x8-BL (256 K フラッシュ) QFN パッケージはそれぞれ CY8C41x7-BL と CY8C42x7-BL (128 K フラッシュ) QFN パッケージとピン ツー ピンの互換性があります。

** = CY8C41x8-BL と CY8C42x8-BL ファミリは–BL4xx と–BL5xx の 2 つのサブファミリがあります。–BL4xx シリーズは Bluetooth 4.2 Link Layer プライバシー、Link Layer データ長拡張、DMA エンジンに対応していませんが、–BL5xx シリーズはそれらの機能に対応しています。

付録 A: Bluetooth® LE デバイス ファミリの比較

*** = CY8CBL10xxx PRoC デバイス ファミリは複数のサブファミリがあり、ここで説明する機能はスーパーセットのデバイスの機能です。詳細については、「**AN94020 - PRoC™ BLE 入門**」を参照ください。

10 付録 B : インフィニオンの専門用語

本節では、インフィニオンの PSoC™ファミリのデバイスで操作する際によく出てくる用語について説明します。

コンポーネント コンフィギュレーション ツール: PSoC™ Creator のシンプルな GUI で、各コンポーネントに組み込まれています。コンポーネント パラメーターをカスタマイズするために使用され、コンポーネントを右クリックすることでアクセスされます。

コンポーネント: PSoC™ Creator ソフトウェア上にアイコンとして表示する無償の組み込み IC です。これらのコンポーネントは複数の IC とシステム インターフェースを、本質的にメイン システム バスを介して MCU に接続される 1 個の PSoC™コンポーネントに統合するのに使用されます。例えば、Bluetooth® LE コンポーネントで Bluetooth Smart 製品をほんの数分で作成できます。同様に、プログラム可能なアナログ コンポーネントをセンサー用に使用できます。

MiniProg3: 開発用のプログラミング ハードウェアであり、内蔵のプログラマがサポートされていないカスタム基板または PSoC™開発キットの上に PSoC™デバイスをプログラムするために使用されます。

PSoC™: 32 ビット Arm® Cortex®-M0 のような CPU、プログラマブル アナログとデジタル ブロックを内蔵するプログラム可能な組み込み設計プラットフォームです。タッチ センシングなどの信頼性が高く、使いやすいソリューションにより組込みシステムの設計を加速させ、低消費電力の設計を可能にします。

PSoC™ 4 with AIROC™ Bluetooth® LE: Bluetooth 4.2 仕様に準拠したロイヤリティフリーの Bluetooth® LE プロトコル スタックに対応している Bluetooth® LE 無線ブロックを統合した PSoC™ 4 IC です。

PSoC™ Creator: 開発を行うお客様の PC にインストールされた PSoC™ 3, PSoC™ 4, および PSoC™ 5LP の統合設計環境 (IDE) ソフトウェアであり、PSoC™システムのハードウェアとファームウェアを同時に設計するか、またはハードウェア設計後に一般的な IDE へエクスポートすることを可能にします。

PSoC™ Programmer: PSoC™デバイスをプログラムするための柔軟性のある統合プログラム アプリケーションです。PSoC™ Programmer は PSoC™ Creator と統合され、PSoC™ 3, PSoC™ 4, ProC, および PSoC™ 5LP 設計をプログラムします。

11 付録 C: インフィニオン Bluetooth® LE 開発ツール

PSoC™ 4 with AIROC™ Bluetooth® LE Pioneer Kit

Figure 59 に示す Bluetooth® LEPioneer Kit は、PSoC™ 4 with AIROC™ Bluetooth® LE のデバイス ファミリに対応しているインフィニオンの Bluetooth® LE 開発キットです。このキットは Bluetooth® LE Pioneer ベースボードに接続した取り外し可能な PSoC™ 4 with AIROC™ Bluetooth® LE (および PRoC-BLE) モジュールから成ります。このキットはコイン電池から、または USB インターフェースを介して電源供給を受けます。

Bluetooth® LE Pioneer ベースボードと Bluetooth® LE モジュールを組み合わせて使用すれば、標準 Arduino Uno コネクタ準拠のシールドと併用する低消費電力のバッテリ式 Bluetooth® LE デバイス、または CapSense™ユーザー インターフェースなどの基板搭載 PSoC™ 4 with AIROC™ Bluetooth® LE デバイス機能を開発できます。

Bluetooth® LE Pioneer ベースボードはオンボード プログラムとデバッグ インターフェースを搭載しているので、PSoC™ Creator IDE を介した Bluetooth® LE 設計のデバッグ処理は手早く簡単になります。Bluetooth® LE Pioneer Kit は Digilent®などの第三者ベンダーからの Pmod™ドーター カードとのインターフェースでの接続をサポートする追加のヘッダーを備えています。このキットは、Bluetooth® LE リンク マスターとして機能し、CySmart ホスト エミュレーション ツールと連動して、非 Bluetooth® LE Windows PC で Bluetooth® LE ホスト エミュレーション プラットフォームを実現する Bluetooth® LE ドングルも内蔵しています。

このキットには、ユーザーが独自の Bluetooth® LE アプリケーションを開発できるように、Bluetooth® LE サンプル プロジェクトと資料一式が含まれています。キットの最新版, キット デザイン, サンプル プロジェクトやドキュメント ファイルを入手するには、**Bluetooth® LE Pioneer Kit** ウェブページにアクセスしてください。

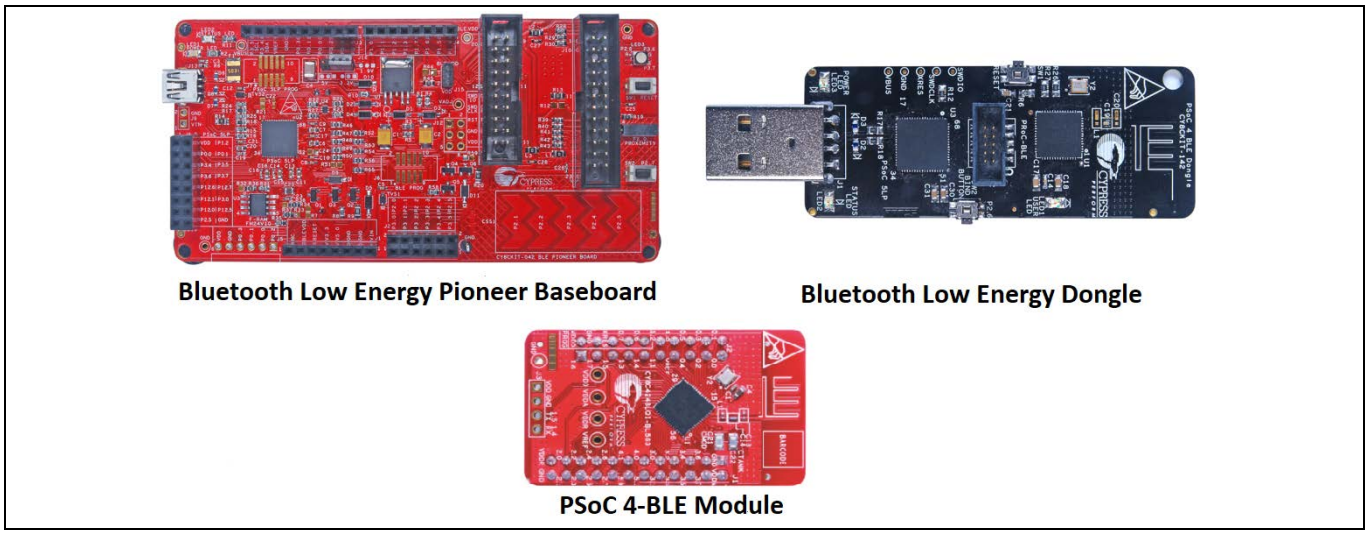


Figure 59 Bluetooth® LE Pioneer Kit

11.1 CySmart ホスト エミュレーション ツール

CySmart ホスト エミュレーション ツールは Bluetooth® LE Pioneer Kit の Bluetooth® LE ドングルを使用して Bluetooth® LE セントラル デバイスをエミュレートするための Windows アプリケーションです (**Figure 19** を参照ください)。このアプリケーションは Bluetooth® LE Pioneer Kit の一部としてインストールされ、Bluetooth® LE コンポーネントの右クリック オプションから起動できます。これは、ペリフェラルの Bluetooth® LE サービス, キャラクタリスティック, およびアトリビュートを調べ、構成することで、GATT または L2CAP 接続指向のチャネルを介して、ご使用の PSoC™ 4 with AIROC™ Bluetooth® LE ペリフェラル実装をテストためのプラットフォームを提供します。

CySmart ホスト エミュレーション ツールを使用して実現できる動作は以下が含まれますが、これらに限定されません。

- Bluetooth® LE ペリフェラルをスキャンして、接続できるデバイスを検出
- サービスとキャラクタリスティックなどの、接続したペリフェラル デバイスの Bluetooth® LE アトリビュートを検出
- キャラクタリスティック バリューとディスクリプタに基づいて読み書きを行う
- 接続したペリフェラル デバイスからのキャラクタリスティック通知および指示を受ける
- Bluetooth® LE セキュリティ マネージャーを使用して、接続したペリフェラル デバイスとの結び (ボンディング) を確立
- ペリフェラル デバイスとの Bluetooth® LE L2CAP 接続指向のセッションを設立し、Bluetooth 4.2 仕様に従ってデータ交換を行う
- インフィニオンの Bluetooth® LE ペリフェラル デバイスの Over-the-air (OTA；無線) ファームウェア アップグレードを行う

Figure 60 と **Figure 61** に、CySmart ホスト エミュレーション ツールのユーザー インターフェースを示します。このツールのセットアップおよび使用方法の詳細については、**Help** メニュー内の CySmart ユーザー ガイドを参照ください。

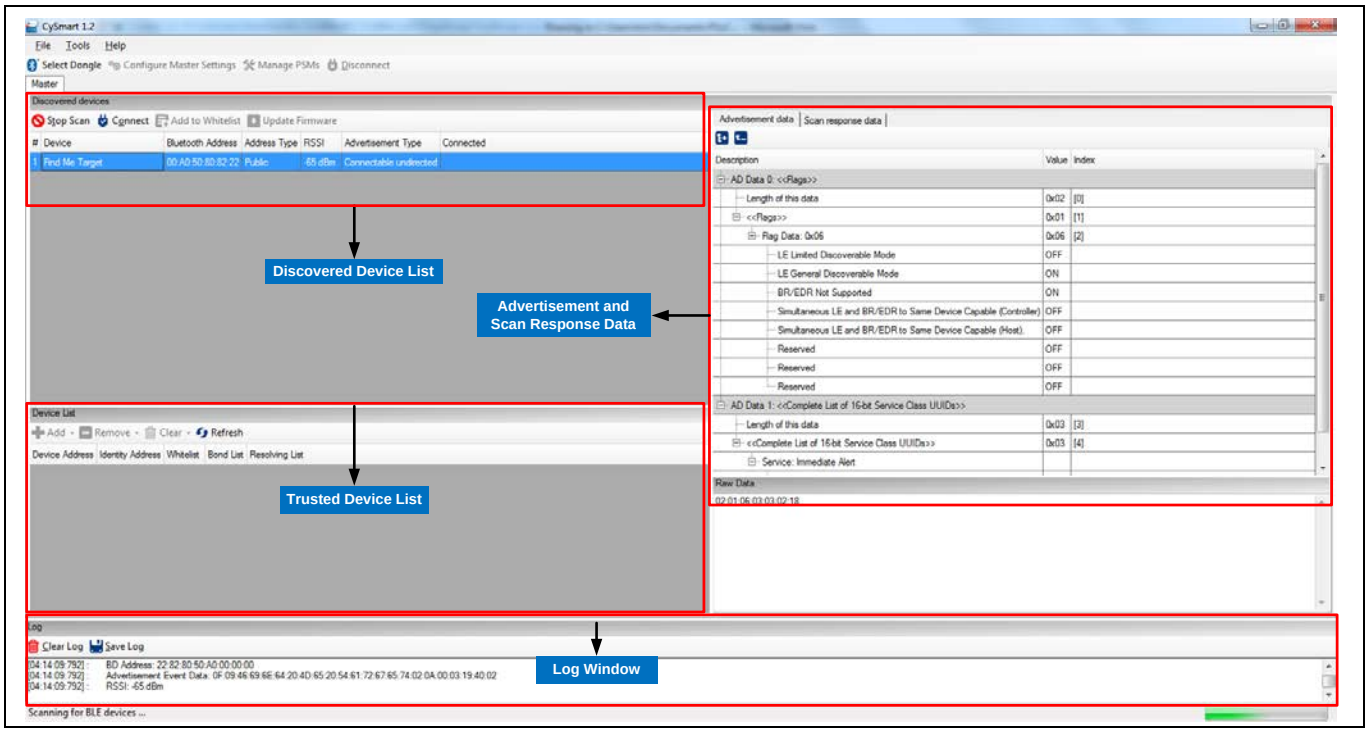


Figure 60 CySmart ホスト エミュレーション ツールのマスター デバイスのタブ

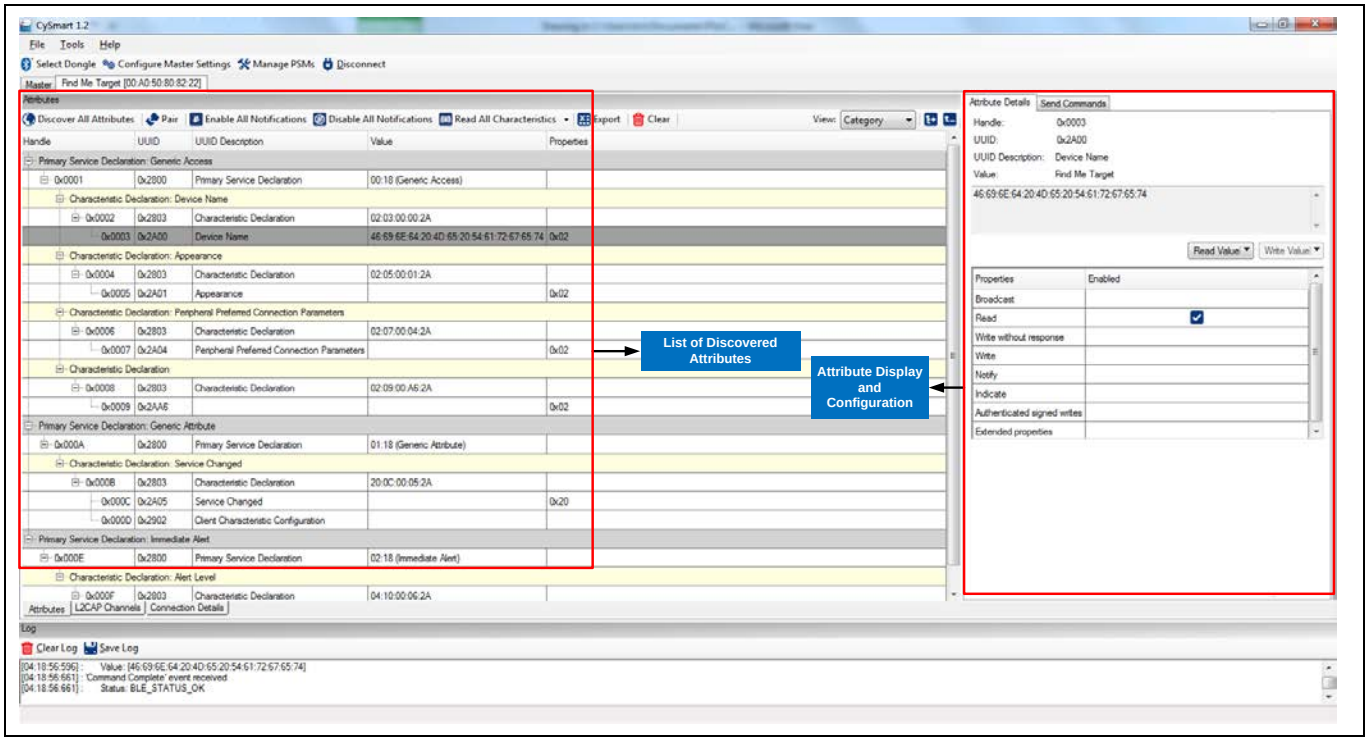


Figure 61　CySmart ホスト エミュレーション ツールのペリフェラル デバイス アトリビュートのタブ

11.2　CySmart モバイル アプリ

PC ツールの他に、それぞれのアプリケーション ストアから iOS または Android 向けの CySmart モバイル アプリをダウンロードすることもできます。このアプリは、Bluetooth® LE 対応のスマートフォンを、ペリフェラル デバイスをスキャンしてそれに接続できるセントラル デバイスとして設定できるよう、それぞれの iOS コアの Bluetooth フレームワークと Android 向けの Bluetooth® LE 用組み込みプラットフォーム フレームワークを使用しています。

モバイル アプリは、直観的な GUI を使用して SIG 採用の Bluetooth® LE 標準プロファイルをサポートし、基本的な Bluetooth® LE サービスおよびキャラクタリスティックの詳細を簡略化します。Bluetooth® LE 標準プロファイルに加えて、アプリはインフィニオンの LED と CapSense™実証例を使用してカスタム プロファイルの実装をデモします。**Figure 62** と **Figure 63** に、CySmart アプリにおける心拍数プロファイルのユーザー インターフェースの一連のスクリーンショットを示します。Bluetooth® LE Pioneer Kit のプロジェクト例を使用してアプリケーションを実行する方法については、Bluetooth® LE Pioneer Kit ガイドを参照ください。

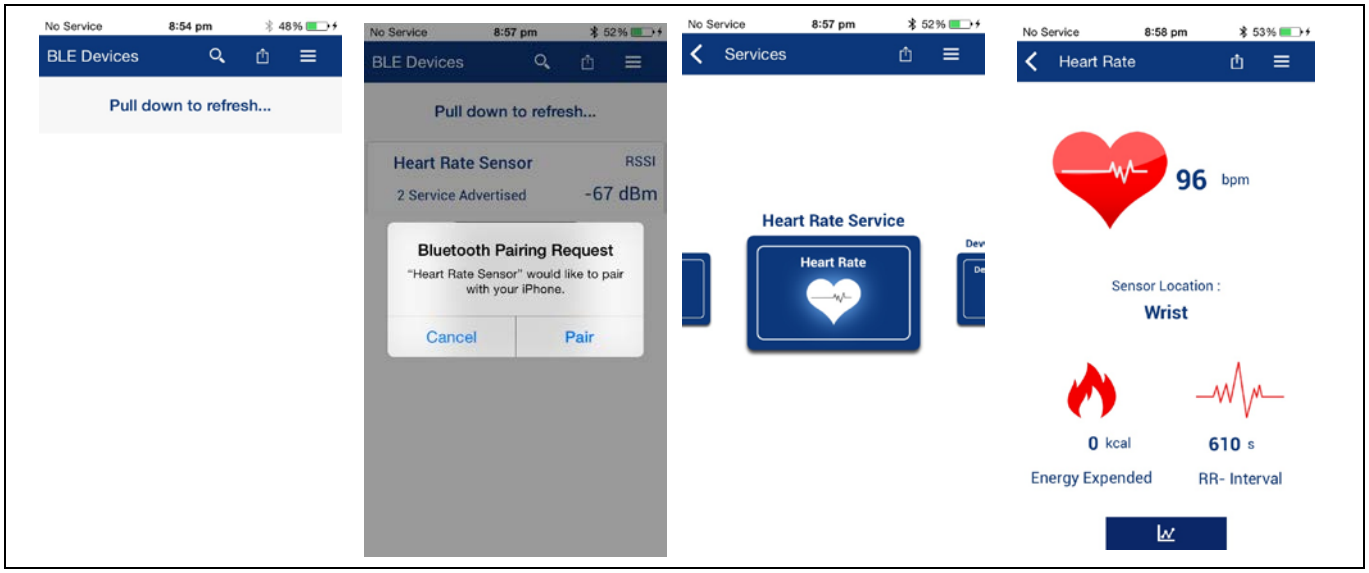


Figure 62　iOS 向け CySmart アプリでの心拍数プロファイルの例

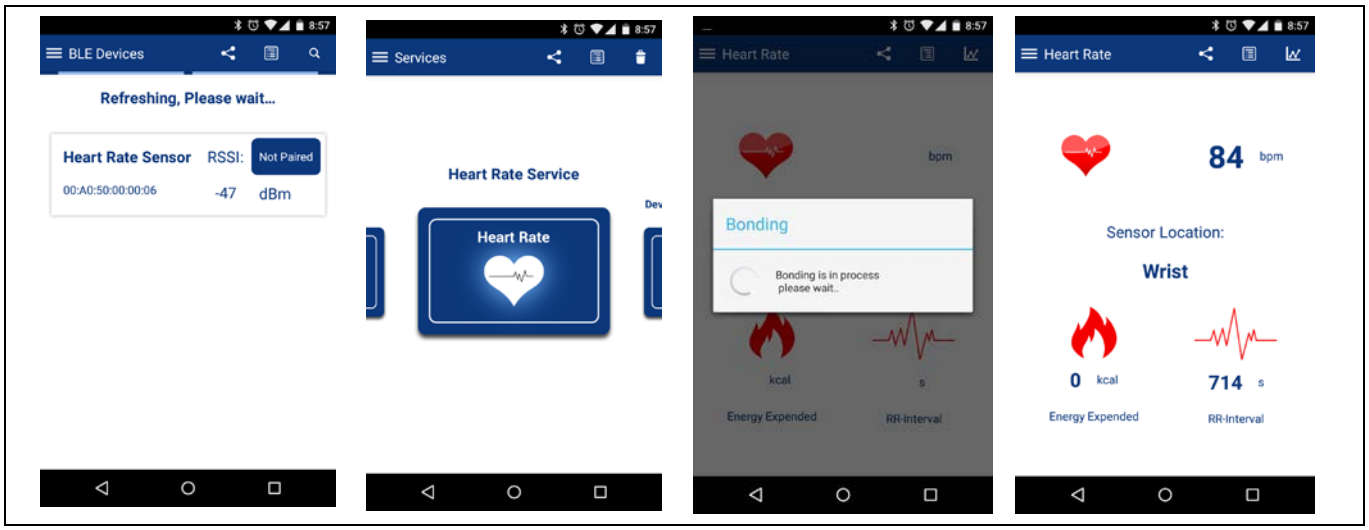


Figure 63　Android 向け CySmart アプリでの心拍数プロファイルの例

12 付録 D: PSoC™ 4 with AIROC™ Bluetooth® LE デバイス

12.1 Bluetooth Low Energy サブシステム (BLESS)

Bluetooth® LE サブシステムは、Physical Layer (PHY) エンジンおよび AES-128 セキュリティ エンジンを組み込んだ Link Layer エンジンを含みます。

Physical Layer は、Bluetooth 4.2 仕様に準拠しているデジタル PHY と RF トランシーバーをから成ります。そのトランシーバーは、2.4 GHz ISM バンドを介して 1 Mbps で GFSK パケットを送受信します。ベースバンド コントローラーは、マスターとスレーブ両方のモードに対応している複合ハードウェア／ファームウェア実装です。HCI とリンク制御などの主なプロトコル要素はファームウェアで実装されます。一方、暗号化や CRC, データ ホワイトニング, アクセス コード相関などのタイム クリティカルな機能はハードウェアで実装されます。

BLESS は Bluetooth 4.2 に準拠して、すべての Bluetooth 4.0 仕様の機能と、低デューティ サイクルの宣伝、LE ping、L2CAP 接続指向のチャネル、Link Layer プライバシー、Link Layer データ長拡張、LE セキュア接続などの Bluetooth 4.2 仕様の追加機能に対応しています。BLESS ブロックはまた、正確な RF 周波数を生成するための外部水晶発振器 (ECO) と、Bluetooth® LE 接続での連続接続間隔の間の時間を守るための時計用水晶発振器 (WCO) を含んでいます。BLESS ブロックの ECO と WCO は PSoC™ 4 with AIROC™ Bluetooth® LE システムの高周波数と低周波数のクロック ソースとして使用されます。「クロック供給システム」を参照ください。

BLESS は次の 5 つの機能的電力モードに対応しています: ディープスリープ、スリープ、アイドル、送信、受信。

Note: *本節で説明する電力モードは BLESS ブロックの固有のモードです。PSoC™ 4 with AIROC™ Bluetooth® LE システムの電力モードについては、「低リーク電力モード」を参照ください。*

12.1.1 ディープスリープ モード

ディープスリープ モードは、BLESS によって対応されている最小消費電力モードです。このモードでは、無線ブロックはオフになっています。パケットの送信と受信が完了した後、宣伝中または接続間隔の間にこのモードに移行して、最小消費電力を実現します。このモードで、消費電力を節約するために ECO をオフにできます。WCO は、低周波数のクロックであり、Bluetooth® LE Link Layer タイミング リファレンス ロジックを維持するためにオンにします。CPU はこの状態の開始と終了を制御します。

12.1.2 スリープ モード

スリープ モードでは、無線ブロックはオフになっています。ブロックはすべての設定を維持します。ECO と WCO がオンですが、コア BLESS ロジックへのクロックはオフです。CPU はこの状態の開始と終了を制御します。

12.1.3 アイドル モード

アイドル モードは送信と受信状態の準備状態です。このモードでは、無線ブロックはオフですが、CPU がプロトコル ステートマシンを起動するために Link Layer ロジックの Link Layer クロックは有効にされています。

12.1.4 送信モード

送信モードはアクティブ機能的モードです。BLESS 内の全ブロックは電源供給されています。Link Layer クロックは、Link Layer と RF-PHY 間のロジックを形成するために有効にされています。このモードで、RF-PHY は Link Layer から 1 Mbps のシリアル データを受信し、2.4 GHz GFSK 変調のデータをアンテナ ポートに送信します。Bluetooth® LE はアイドル モードから送信モードに入ります。

12.1.5 受信モード

このモードは、BLESS が受信状態に入り、Bluetooth® LE 固有のレシーバー動作を実行できるようにします。RF-PHY は復調後に、RF アナログ ブロックから受信した 1 Mbps のデータを変換して、それを Link Layer コントローラーに転送します。BLESS 電力モードと動作サブブロックのまとめは **Table 3** に示しています。

Table 3 BLESS 電力モード

BLESS 電力モード	ECO	WCO	RF Tx	RF Rx	BLESS コア
ディープスリープ	オフ	オン	オフ	オフ	オフ
スリープ	オン	オン	オフ	オフ	オフ
アイドル	オン	オン	オフ	オフ	オン
送信	オン	オン	オン	オフ	オン
受信	オン	オン	オフ	オン	オン

12.2 Arm® Cortex®-M0, メモリ, DMA

PSoC™ 4 with AIROC™ Bluetooth® LE は 32 ビット Arm® Cortex®-M0 CPU を内蔵しており、最大 48MHz の周波数で動作可能で、43 DMIPS 性能を実現します。CPU はシングル サイクルの 32 ビット乗算をサポートします。

PSoC™ 4 with AIROC™ Bluetooth® LE は 16 KB／32 KB の SRAM および 128 KB／256 KB のフラッシュ メモリを内蔵しており、ほとんどの Bluetooth® LE アプリケーションに対応できます。フラッシュ メモリはリードアクセラレータを備えています。デバイスはまた、Bluetooth® LE デバイス アドレスや暗号化キーなどのユーザー固有のデータを格納する 512 バイトの監視フラッシュ領域を提供します。

PSoC™ 4XX8-BL デバイス ファミリは 8 本のチャネルを備えた DMA エンジンを搭載しています。このエンジンはペリフェラル間、メモリ間、またはペリフェラルとメモリ間で 32 ビット転送を実行できます。

12.3 プログラマブルなデジタル ペリフェラル

PSoC™ 4 with AIROC™ Bluetooth® LE は、プログラマブルなシリアル通信ブロック (SCB)、タイマー／カウンター／パルス幅変調器 (TCPWM)、ユニバーサル デジタル ブロック (UDB) と呼ばれるプログラマブルなロジック アレイを含むデジタル ペリフェラルの豊富なセットを提供します。

12.3.1 プログラマブルな SCB

PSoC™ 4 with AIROC™ Bluetooth® LE は、実行時に SCB は I^2C、SPI または UART にプログラム可能な、独立した SCB を内蔵しています。SCB は以下の機能をサポートします。

- Motorola®、Texas Instruments®、National Semiconductor®のプロトコルに対応している標準的な SPI マスターと SPI スレーブ機能
- スマートカード リーダー、ローカル相互接続ネットワーク (LIN)、および赤外線データ協会 (IrDA) プロトコルに対応している標準的な UART 機能

- 標準的な I²C マスターとスレーブ機能
- CPU 介入無しで動作可能な SPI および EZI²C モード
- SPI および I²C プロトコル用の低消費電力 (ディープ スリープ) 動作モード (外部クロックを使用)

詳細については、**PSoC™ 4 SCB コンポーネント データシート**を参照ください。

12.3.2 プログラマブルな TCPWM

PSoC™ 4 with AIROC™ Bluetooth® LE は 4 個のプログラマブルな 16 ビット TCPWM ブロックを持っています。各 TCPWM は、16 ビット タイマー, カウンター, PWM, または直交デコーダを実装できます。TCPWM は、相補出力および選択可能なスタート, リロード, ストップ, カウント, およびキャプチャ イベントの信号を提供します。PWM モードは中央揃え, エッジ, および疑似ランダム動作をサポートします。

詳細については、**PSoC™ 4 TCPWM コンポーネント データシート**を参照ください。

12.3.3 ユニバーサル デジタル ブロック

Figure 64 に示すように、UDB は CPLD および FPGA ブロックと同様の機能を提供するプログラマブルなロジック ブロックです。UDB により、タイマーやカウンター, PWM, 疑似ランダム シーケンス (PRS), CRC, シフト レジスタ, SPI, UART, I²S, カスタム組み合わせおよびシーケンシャル ロジック回路などのさまざまなデジタル機能を作成できます。

各 UDB は、それぞれ 12 つの入力と 8 つのプロダクト タームを備えた 2 個のプログラマブル ロジック デバイス (PLD) を持っています。PLD はレジスタードまたは組み合わせ積和ロジックを形成できます。さらに、「データパス」と知られている 8 ビットのシングル サイクル算術ロジック ユニット (ALU) は各 UDB に備えられています。データパスにより、タイマーやカウンター、PWM、CRC などの機能を効果的に実施できます。

また UDB は、通信と制御用にペリフェラルとポートからの信号を UDB に、また UDB を介して送信することを可能にするスイッチト デジタル信号相互接続 (DSI) ファブリックを提供しています。

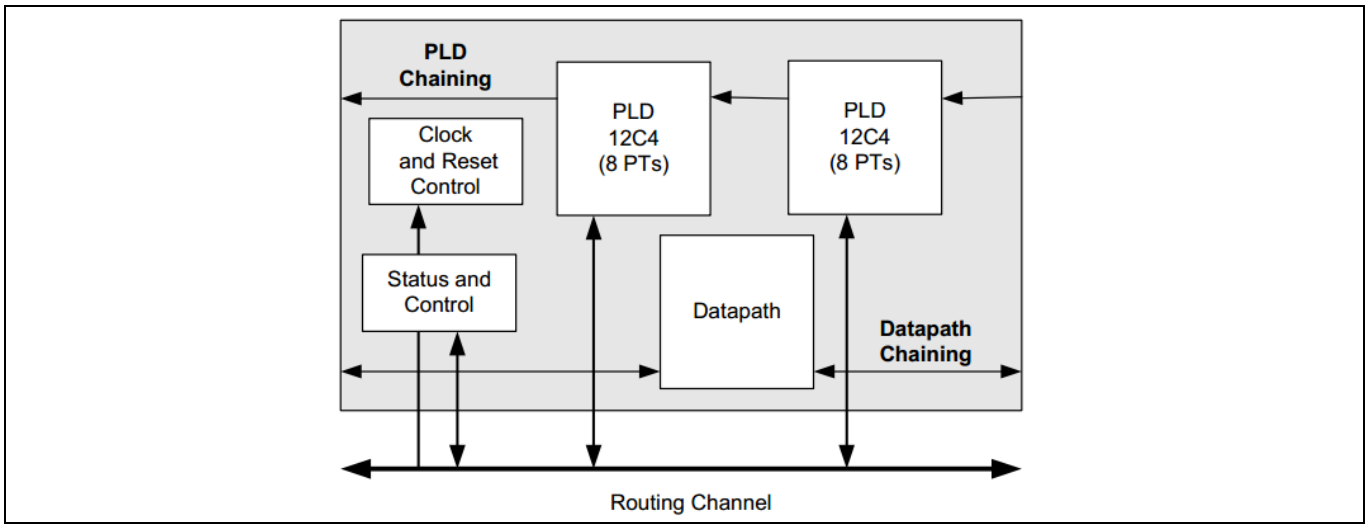


Figure 64 ユニバーサル デジタル ブロック図

ハードウェア記述言語 (HDL) に精通していなくても UDB を使用できます。PSoC™ Creator というインフィニオン PSoC™ 4 with AIROC™ Bluetooth® LE 用の開発ツールは、回路図から必要な機能を生成できます。上級ユーザーは必要に応じて、Verilog を使って UDB でカスタム ロジックを実装できます。

詳細については、次のアプリケーション ノートを参照ください。

- **AN62510 – Implementing State Machines with PSoC™3, PSoC™4, and PSoC™5LP**
- **AN82156 – PSoC™3, PSoC™4, and PSoC™5LP – Designing PSoC™ Creator Components with UDB Datapaths**
- **AN82250 – PSoC™3, PSoC™4, and PSoC™5LP –Implementing Programmable Logic Designs with Verilog**

12.3.4 アプリケーション

Bluetooth® LE アプリケーションのプログラマブルなデジタル ペリフェラルの使用は **Table 4** に示しています。

Table 4 プログラマブルなデジタル ペリフェラルのアプリケーション

アプリケーション	デジタル ペリフェラル
センサー ハブ	I^2C (デジタル センサー インターフェース)、PWM (アクチュエータ)、I^2S (音声入力)
健康関連製品	PWM (ユーザー インターフェース)、カウンター (波形ピーク測定)、SPI (外部メモリ インターフェース)
産業機器	UART (modbus)、カウンター (イベント カウント)
ホーム オートメーション	PWM (ガレージ ドア制御)、PWM (照明管理)

12.4 プログラマブルなアナログ

PSoC™ 4 with AIROC™ Bluetooth® LE は、クラス最高のアナログ統合を提供します。アナログ システムは連続時間ブロック ミニ (CTBm) ブロック, 12 ビット高速 SAR ADC, 低消費電力コンパレータ, 静電容量タッチセンシング (CapSense™), およびセグメント LCD 直接ドライブを含みます。

12.4.1 連続時間ブロック ミニ (CTBm)

PSoC™ 4 with AIROC™ Bluetooth® LE は 2 個の CTBm ブロックを内蔵しています。各ブロックは 2 個のプログラマブルなオペアンプとスイッチ マトリックスで構成されます。それぞれのオペアンプを個別にコンパレータ, 電圧フォロワー, または外部フィードバック付きオペアンプに設定できます。必要に応じて、デバイスのディープスリープ モードであっても CTBm ブロックが機能するように設定できます。

詳細については、次のコンポーネント データシートを参照ください。

- **PSoC™ 4 オペアンプ**
- **PSoC™ 4 電圧コンパレータ**

12.4.2 ハードウェア シーケンサを備えた SAR ADC

PSoC™ 4 with AIROC™ Bluetooth® LE は、プログラマブルな分解能とシングル エンド／差動入力 オプションに対応している入力チャネルを備えた 12 ビットの 1 Msps 逐次比較レジスタ (SAR) ADC を内蔵しています。GPIO 数は実装可能な ADC 入力 チャネル数を制限します。

SAR ADC は、CPU の介入無しに最大 8 本のチャネルで自動スキャンを実行できるハードウェア シーケンサを備えています。SAR ADC はまた、これらの 8 本のチャネルで出力データの累積や平均化などの前処理動作をサポートします。

ファームウェアやタイマー, ピン, UDB などのさまざまな手段でスキャンをトリガーでき、追加の設計の柔軟性を与えます。

詳細については、**PSoC™ 4 SAR ADC コンポーネント データシート**を参照ください。

12.4.3 低消費電力コンパレータ

PSoC™ 4 with AIROC™ Bluetooth® LE デバイスは、**ストップ** モードを除きすべてのシステム電力モードで動作可能な低消費電力コンパレータを備えています。パワー センシティブな設計では、デバイスが低消費電力モードに入った時、低消費電力コンパレータを使用してアナログ入力を監視し、システムを復帰させられる割り込みを生成できます。

詳細については、**PSoC™ 4 低消費電力コンパレータのコンポーネント データシート**を参照ください。

12.4.4 静電容量タッチ センシング (CapSense™)

静電容量タッチ センサーは、人体の静電容量を使ってセンサー上またはセンサーの近くの指の存在を検出します。静電容量センサーは審美的に優れており、使い易く、長寿命を持っています。

PSoC™ 4 with AIROC™ Bluetooth® LE の CapSense™機能は、前例のない信号対ノイズ比、クラス最高の耐水性、および多種多様なセンサー (ボタン, スライダー, トラックパッド, 近接センサーなど) を提供します。

インフィニオンが提供するソフトウェア コンポーネントを使用すれば、静電容量センシングの設計が簡単になります。このコンポーネントは、SmartSense™というハードウェア自動チューニング機能をサポートし、トラックパッドと近接センサーのジェスチャー認識ライブラリを提供します。

静電容量センシングが使用されない場合、CapSense™ブロックに備えられた 7 ビットと 8 ビットの 2 個の電流 DAC (IDAC) は一般用途に使用できます。CapSense™ブロックのコンパレータは汎用使用可能です。

詳細については、**PSoC™ 4 CapSense™設計ガイド**を参照ください。

12.4.5 セグメント LCD 直接ドライバー

血糖値計, マルチメータ, 血圧計などのほとんどの低消費電力のポータブルなハンドヘルド機器は、セグメント LCD を使って情報を表示します。通常、セグメント LCD はマイクロコントローラーと通信するために外部ドライバーを必要とします。PSoC™ 4 with AIROC™ Bluetooth® LE はセグメント LCD ガラスを直接駆動できる低消費電力 LCD ドライバーを内蔵しています。

PSoC™ 4 with AIROC™ Bluetooth® LE は最大 4 コモンと 32 セグメント電極で LCD 駆動できます。セグメント LCD ドライバーは、ディープスリープ モードで最大 7 µA のシステム消費電流でスタティック ディスプレイを保持できます。

詳細については、「**AN87391 – PSoC™ 4 Segment LCD Direct Drive**」を参照ください。

12.4.6 アプリケーション

Table 5 に、異なる Bluetooth® LE アプリケーションにおけるプログラマブルなアナログ ペリフェラルの使用を示します。

Table 5 プログラマブルなアナログ ペリフェラルのアプリケーション

アプリケーション	アナログ ペリフェラル
健康関連製品	オペアンプ: TIA (心拍測定) オペアンプ: フォロワー (アナログ リファレンス バッファ) ADC (サンプリング心拍信号) CapSense™ (リスト バンド上のユーザー インターフェース) セグメント LCD (リスト バンド上の表示)
センサー ハブ	ADC (アナログ センサー インターフェース)

付録 D: PSoC™ 4 with AIROC™ Bluetooth® LE デバイス

アプリケーション	アナログ ペリフェラル
	アナログ マルチプレクサ (多センサー入力) オペアンプ (信号アンプ) セグメント LCD (UI)
産業機器	ADC: 差動モード (温度測定) IDAC (温度センサーの駆動) 低消費電力コンパレータ (閾値検出時のウェイクアップ) オペアンプ: PGA (4 mA～20 mA の電流ループ システム)
ホーム オートメーション	オペアンプ: PGA (モーション センサー、光センサー) コンパレータ (ドア センサー) ADC とオペアンプ フィルター (煙探知器)

12.5 システム全体のリソース

本節は、PSoC™ 4 with AIROC™ Bluetooth® LE で使用可能なすべてのペリフェラル システム全体のリソースを説明します。

12.5.1 低リーク電力モード

PSoC™ 4 with AIROC™ Bluetooth® LE は以下の電力モードを提供します。これらのモードは PSoC™ 4 with AIROC™ Bluetooth® LE デバイスの電力モードであり、「**Bluetooth Low Energy サブシステム (BLESS)**」節で記述された電力モードとは異なることに注意してください。

- **アクティブ モード:** これは主要な動作モードです。このモードでは、すべてのペリフェラルが有効です。
- **スリープ モード:** このモードでは、CPU がスリープ状態になり、SRAM が保持状態になり、すべてのペリフェラルが有効です。割り込みが発生すると、CPU がウェイクアップして、システムがアクティブ モードに戻ります。
- **ディープスリープ モード:** このモードでは、高周波数クロック (IMO) およびすべての高速ペリフェラルが無効です。オプションとして、低周波数クロック (32 kHz ILO と WCO) および低速ペリフェラルは使用可能です。低速度、非同期または低消費電力アナログ ペリフェラルの割り込みにより復帰が可能です。このモードでは、消費電流は 1.3 µA です。
- **ハイバネート モード:** この電力モードは、SRAM および低消費電力コンパレータや GPIO によって生成された割り込みによるウェイクアップ機能を保持しながら、150 nA のベストインクラスの消費電流を実現します。
- **ストップ モード:** この電力モードでは GPIO 状態が保持されます。固定の WAKEUP ピンからのウェイクアップは可能です。このモードでは、消費電流はわずか 60 nA です。

バッテリ駆動 Bluetooth® LE システムで、スリープ、ディープスリープ、ハイバネートおよびストップ モードの組み合わせを使用し、より長いバッテリ寿命があるベストインクラスのシステム消費電力を実現できます。

Table 6 に、PSoC™ 4 with AIROC™ Bluetooth® LE システムの電力モードと BLESS 電力モードの依存性を示します。すべての依存性は簡単な API で処理されます。例については、「**メインループおよび低消費電力**」節を参照ください。

心拍数監視などの標準的な Bluetooth® LE アプリケーションで、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスは、心拍数を測定する際にアクティブ モードであり、Bluetooth® LE 無線ブロックがパケットを送受信する際にスリープ モードであり、連続した Bluetooth® LE 接続間隔の間にディープスリープ モードであり、Bluetooth® LE の宣伝タイムアウトの間にハイバネートまたはストップ モードです。

Table 6 PSoC™ 4 with AIROC™ Bluetooth® LE 電力モード

BLESS モード	PSoC™ 4 with AIROC™ Bluetooth® LE システム電力モード				
	アクティブ	スリープ	ディープスリープ	ハイバネート	ストップ
送信	✓	✓	×	×	×
受信	✓	✓	×	×	×
アイドル	✓	✓	×	×	×
スリープ	✓	✓	×	×	×
ディープスリープ	✓	✓	✓	×	×
電源投入	×	×	×	✓	✓

12.5.2 電源と電圧の監視

PSoC™ 4 with AIROC™ Bluetooth® LE は 1.9 V～5.5 V 電源電圧で動作できます。デバイスの様々な電力モードをサポートするために多数の内部レギュレータがあります。PSoC™ 4 with AIROC™ Bluetooth® LE は POR, BOD, LVD の 3 種類の電圧監視機能があります。

12.5.3 クロック供給システム

PSoC™ 4 with AIROC™ Bluetooth® LE は以下のクロック ソースを備えています。

- **内部主発振器 (IMO)**: IMO は PSoC™ 4 with AIROC™ Bluetooth® LE の内部クロック供給の主なソースです。CPU およびすべての高速ペリフェラルは IMO または外部水晶発振器 (ECO) で動作できます。PSoC™ 4 with AIROC™ Bluetooth® LE は IMO または ECO で動作する多数のペリフェラル クロック分周器を備えています。これらの分周器は高速ペリフェラル用のクロックを生成します。IMO は、3 MHz～48 MHz の範囲にわたり 1 MHz インクリメントで±2%の精度でクロックを生成できます。
- **内部低速発振器 (ILO):** ILO は 32 kHz の超低消費電力発振器であり、BLESS を除き主にディープスリープ モードで動作する低速ペリフェラル用にクロックを生成します (**WCO** を参照ください)。
- **外部水晶発振器 (ECO):** 水晶負荷容量が調整可能な外部水晶発振器は、精度の高い 24 MHz クロックを生成するために使用されます。これは主に RF クロックを生成する Bluetooth® LE サブシステムにクロックを供給するために使用されます。高精度 ECO クロックはまた、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスの高周波数クロック (HFCLK) のクロック ソースとして使用できます。
- **時計用水晶発振器 (WCO):** 32kHz の WCO は (ILO と共に) LFCLK のクロック源として使用されます。WCO は、Bluetooth® LE 宣伝および接続イベントの時間間隔を正確に維持するために使用されます。ILO と同様、WCO はハイバネートとストップ モードを除きすべてのモードで使用可能です。

Figure 65 に、PSoC™ 4 with AIROC™ Bluetooth® LE デバイスのクロック供給アーキテクチャを示しています。

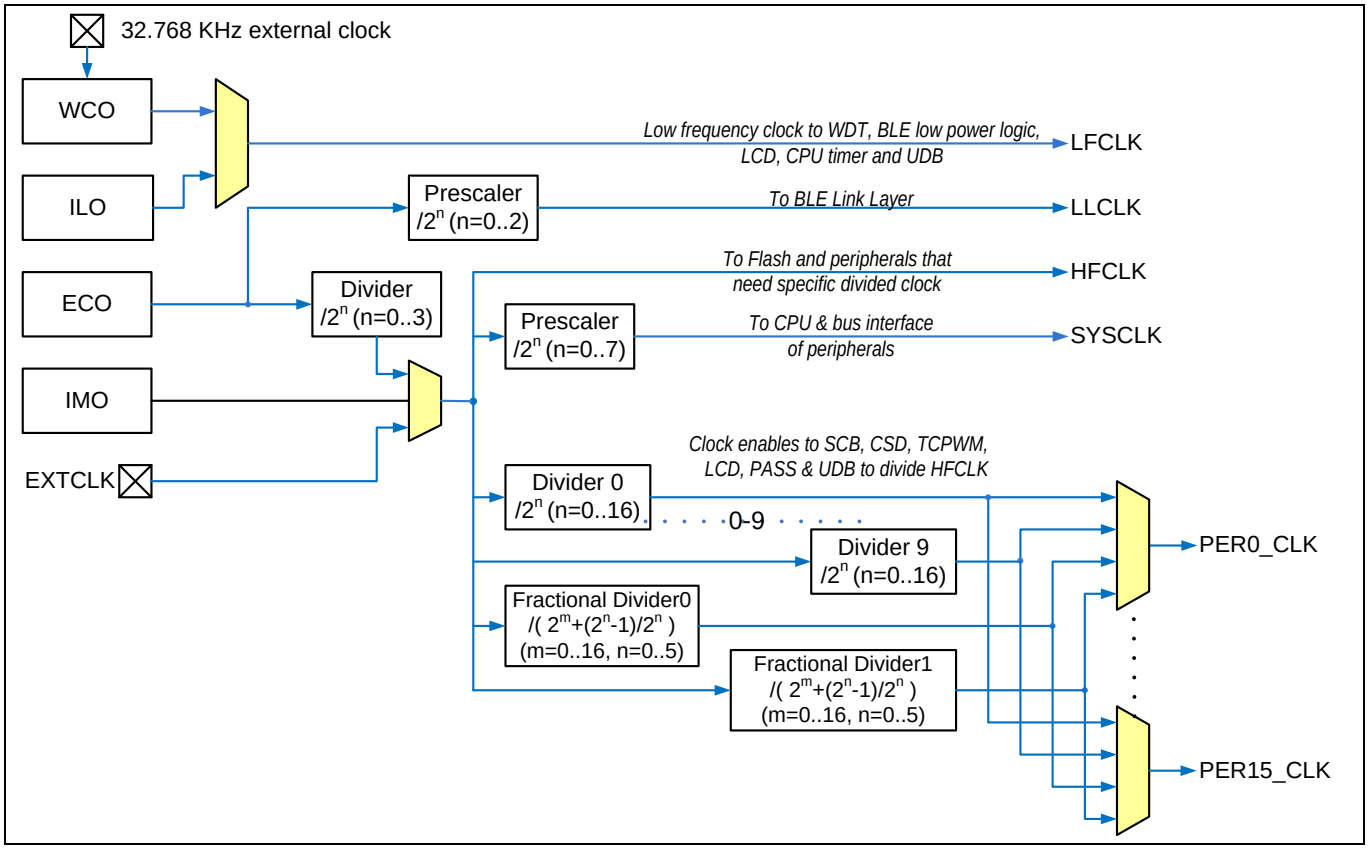


Figure 65 PSoC™ 4 with AIROC™ Bluetooth® LE クロック供給システム

12.5.4 デバイス セキュリティ

PSoC™ 4 with AIROC™ Bluetooth® LE はフラッシュ メモリを不正アクセスあるいはコピーから保護するオプションをいくつか提供します。フラッシュの各行は保護ビットが 1 つあります。これらのビットはフラッシュの監視行に格納されています。

12.6 プログラマブルな GPIO

I/O システムは CPU やペリフェラルと外部間のインターフェースを提供します。PSoC™ 4 with AIROC™ Bluetooth® LE は最大 36 本のプログラマブルな GPIO ピンを備えています。CapSense™, LCD, アナログ, またはデジタル信号用に GPIO を設定できます。PSoC™ 4 with AIROC™ Bluetooth® LE の GPIO は多くの駆動モード、駆動強度およびスルーレートに対応しています。

PSoC™ 4 with AIROC™ Bluetooth® LE は内部信号を GPIO に接続する多数のオプションを与えるインテリジェントな配線システムを提供します。この柔軟な配線により、回路設計と基板レイアウトが簡単になります。

13 付録 E: Bluetooth® LE プロトコル

13.1 概要

Bluetooth® LE (Bluetooth Smart とも呼ばれている) は、2.4 GHz ISM 帯で動作する低消費電力の無線規格として Bluetooth SIG により策定されました。**Figure 66** に Bluetooth® LE プロトコル スタックを示します。

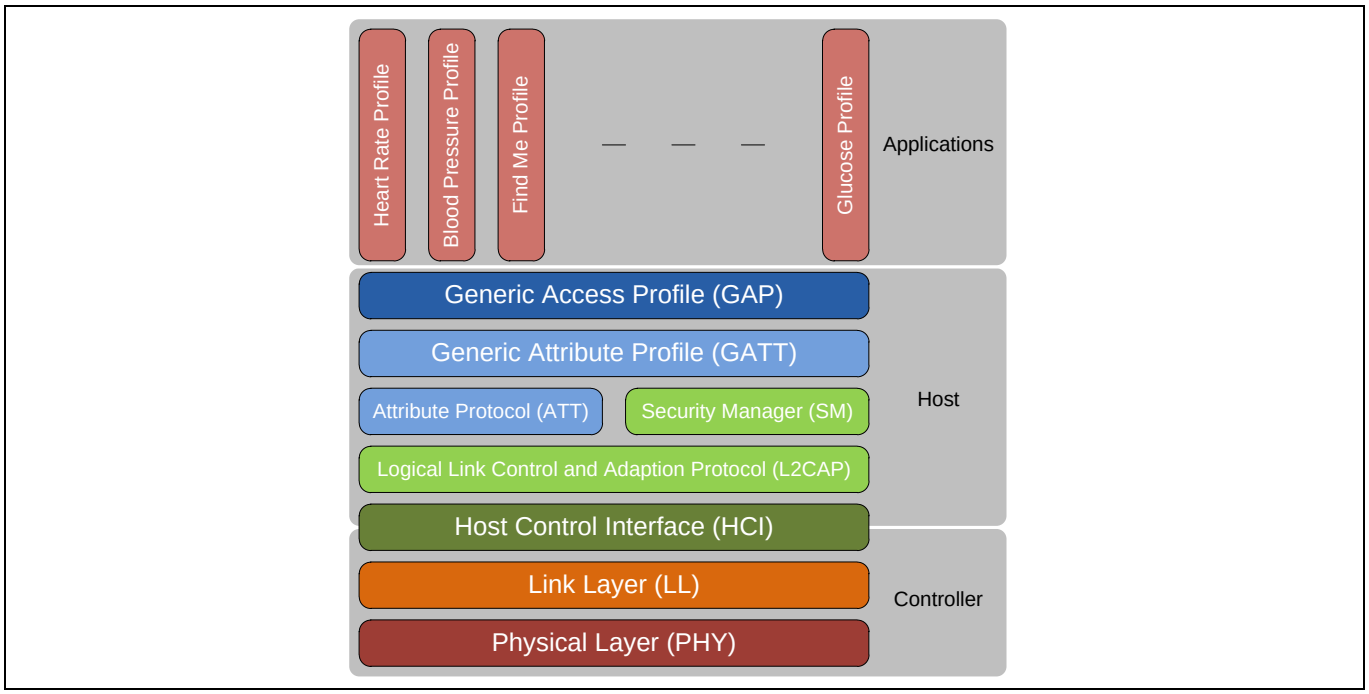


Figure 66 Bluetooth® LE アーキテクチャ

Bluetooth® LE スタックは次の 3 グループに分けられています。

- **コントローラー:** 物理デバイスで、パケットを暗号化して無線信号として送信します。信号の受信時、コントローラーは無線信号を復号してパケットを復元します。
- **ホスト:** セキュリティ マネージャー、アトリビュート プロトコルなどの様々なプロトコルとプロファイルを含むソフトウェア スタックであり、2 つ以上のデバイスがどのように互いに通信するかを管理します。
- **アプリケーション:** ソフトウェア スタックとコントローラーを使用して特定の機能を実装するユースケースです。

下記の節は、標準的な心拍数監視サービスと電池サービスを例として取り上げて Bluetooth® LE スタックの複数の層の概要について説明します。Bluetooth® LE アーキテクチャの詳細説明については、**Bluetooth Developer** のサイトに掲載されている Bluetooth 4.2 仕様または学習ビデオをご覧ください。

13.2 Physical Layer (PHY)

Physical Layer は、2.4 GHz ISM 帯のガウス型周波数偏移変調 (GFSK) という変調方式を利用して 1Mbps でデジタル データを送受信します。Bluetooth® LE の Physical Layer は ISM 帯を 2 MHz ごと 40 本の RF チャネルに分けており、その内データ チャネルは 37 本で、宣伝チャネルは 3 本です。

13.3 Link Layer (LL)

Link Layer は、(認識およびフロー制御に基づいたアーキテクチャを利用して) 信頼性のある物理リンクを成立するための重要な手順、および堅牢で低消費電力の Bluetooth® LE プロトコルを実現する機能を実装します。Link Layer の機能の一部は以下の通りです。

- 宣伝、スキャン、接続の作成および維持: 物理リンクを確立
- 24 ビット CRC および 128 ビット AES 暗号化: データ交換時の堅牢性と安定さを確保
- 高速接続の確立および低デューティ サイクルの宣伝: 低消費電力動作を実現
- Adaptive Frequency Hopping (AFH): パケット送信用の通信チャネルを変更し他のデバイスからの干渉を軽減

Link Layer では、次の 2 つのロールが定義されます。

- **マスター:** スマートフォンは Link Layer をマスター コンフィギュレーションに設定する例です。
- **スレーブ:** 心拍数監視機器は Link Layer をスレーブ コンフィギュレーションに設定する例です。

PSoC／PRoC-BLE デバイスは両方のコンフィギュレーションで動作できます。

Link Layer スレーブは自体の存在を他の Link Layer マスターに宣伝するものです。Link Layer マスターは宣伝パケットを受信し、アプリケーションの要求に応じてスレーブとの接続を決定します (**Figure 67** をご覧ください)。この心拍数監視アプリケーションの実装例では、心拍数監視機器はスレーブとして動作し、データをマスターとしてのスマートフォンに送信します。その後、スマートフォンのアプリケーションはスマートフォン上に読み出し結果を表示します。

PSoC／PRoC-BLE デバイスは、宣伝、CRC や AES 暗号化などタイム クリティカルでプロセッサに集中する Link Layer の機能をハードウェアで実装します。宣伝状態への移行や暗号化の開始などの Link Layer の制御動作はファームウェアで実装されます。

Figure 67 に、Bluetooth® LE の Link Layer パケットの構造および各々のフィールドのサイズを示します。Link Layer パケットは上層のすべてのデータをペイロード フィールドに格納します。物理リンク上の通信を一意に識別し同じ RF チャネルで動作する近隣 Bluetooth® LE デバイスからのパケットを無視するための 4 バイト アクセス アドレスを用意しています。24 ビット CRC によりデータの堅牢性を確保できます。

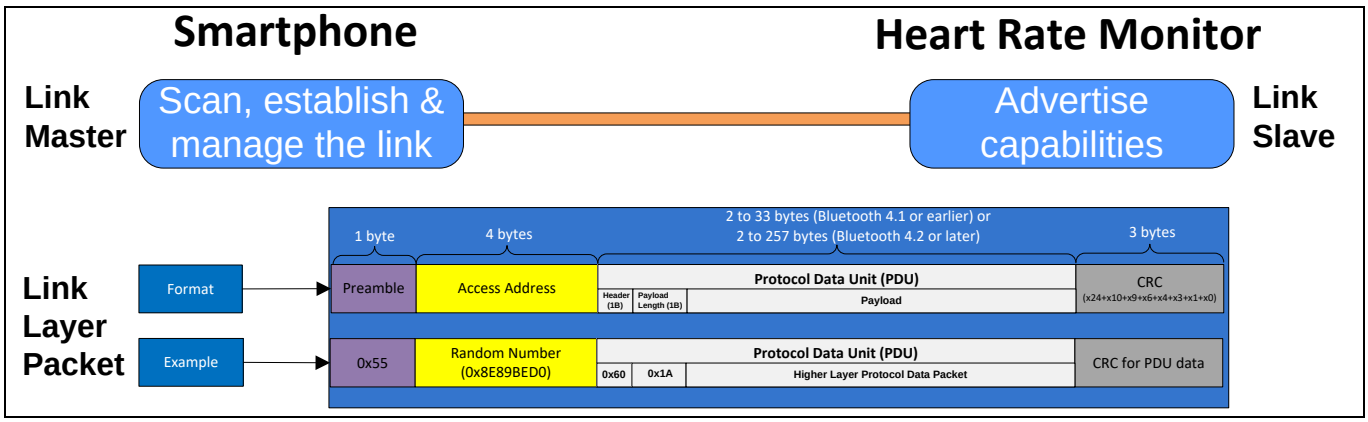


Figure 67 Bluetooth® LE Link Layer プロトコル

13.4 Host Control Interface (HCI)

HCI はホストとコントローラー間の標準として定義されたインターフェースです。これによりホストとコントローラーがコマンド、データ、イベントなどの情報を USB や UART のような異なる物理的転送方式で相互交換できます。コントローラーとホストは異なるデバイスの場合にのみ HCI に物理的転送方法を整備する必要があります。

PSoC／PRoC-BLE デバイスでは、HCI が単なるファームウェア プロトコル層で、メッセージとイベントをコントローラーとホスト間で転送するだけです。

13.5 Logical Link Control and Adaptation Protocol (L2CAP)

L2CAP はプロトコルの多重化、分割、再構成サービスを上層のプロトコルに提供します。分割サービスでは、上層から受信したパケットを Link Layer が送信できる多くのより小さいパケットに分割します。再構成サービスでは、Link Layer から受信した分割済みより小さいパケットを組み合わせて有意味のパケットを構成します。L2CAP 層は、**Figure 68** のように、**Attribute Protocol (ATT)**、**Security Manager (SM)**、および L2CAP 自体の制御用に 3 つのプロトコル チャネル ID を提供します。Bluetooth 4.2 仕様により、これらのプロトコル チャネルの上部にある L2CAP 接続指向チャネルから直接データ チャネルを作成できます。

PSoC／PRoC-BLE では、L2CAP とその上部にある各層がファームウェアで実装されます。

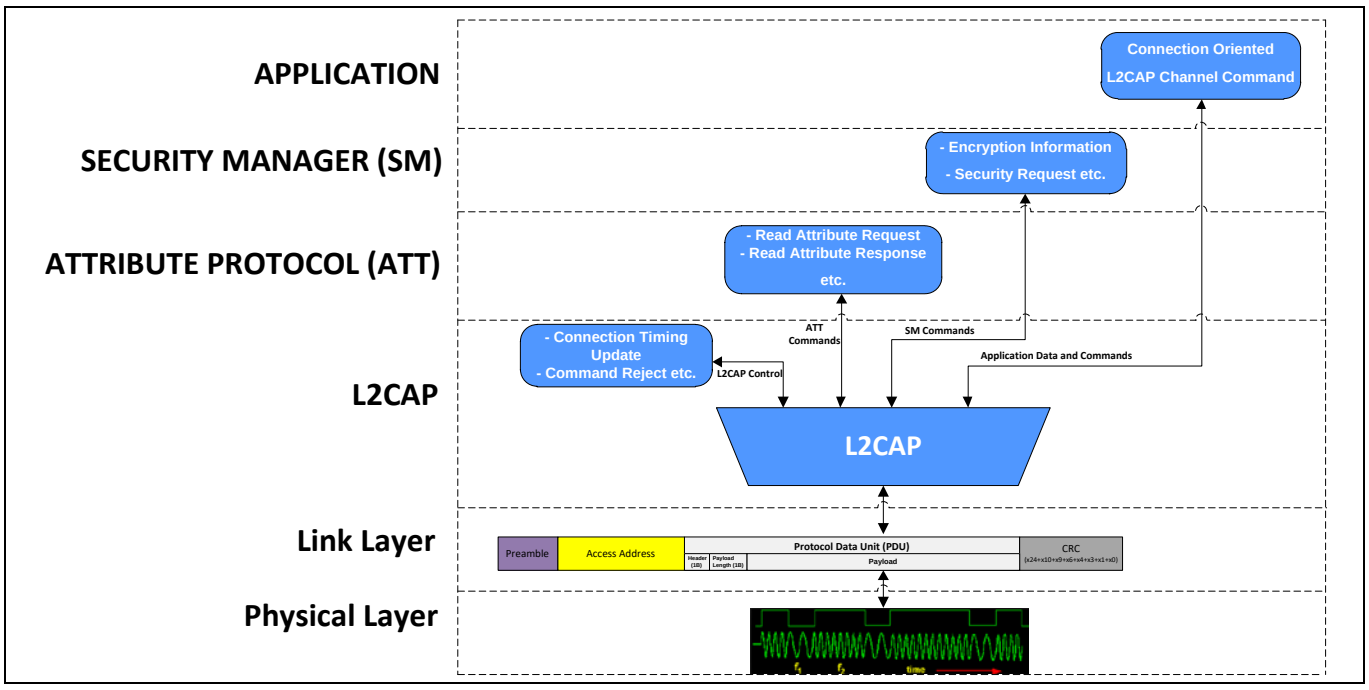


Figure 68 Bluetooth® LE L2CAP 層

13.6 Security Manager (SM)

SM 層はペアリング、暗号化、およびキーの配布の方法を定義します。

- **ペアリング**はセキュリティ機能を有効にするプロセスです。このプロセスでは、2 つのデバイスが認証され、リンクが暗号化されてから、暗号化のキーが交換されます。これにより、RF チャネル上でスパイにより探られることなく、Bluetooth® LE インターフェースを介してデータが安全に交換できるようになります。

- **ボンディング**は、ペアリング プロセスで交換されたキーと識別情報が保存されるプロセスです。デバイスが一度ボンディングされたら、再接続する際に再度ペアリング プロセスを経過する必要がありません。

Bluetooth® LE はデータ暗号化に 128 ビット **AES** 方式を用いています。

13.7 Attribute Protocol (ATT)

Bluetooth® LE の GATT 仕様には、ATT と GATT 層を理解するために把握する必要がある 2 つのロールがあります。

- **GATT サーバー:** データや情報を格納します。GATT クライアントから要求を受信しデータで返答します。例えば、心拍数モニターの GATT サーバーは心拍数の情報を格納します。Bluetooth® LE 型 HID キーボードの GATT サーバーはユーザーがキーを押す情報を格納します。
- **GATT クライアント:** GATT サーバーからデータを要求／受信します。例えば、スマートフォンは心拍数モニターの GATT サーバーから心拍数情報を受信する GATT クライアントです。ノートパソコンは Bluetooth® LE キーボードからキー押下情報を受信する GATT クライアントです。

ATT は Bluetooth® LE 通信の基盤となっています。このプロトコルにより、GATT クライアントが GATT サーバーでデータやアトリビュートを検索してアクセスすることが可能になります。GATT クライアントと GATT サーバーのアーキテクチャの詳細については、「**Generic Attribute Profile (GATT)**」を参照ください。

アトリビュートは次の内容を含む ATT／GATT 層の基本的なデータ コンテナです。

- **アトリビュート ハンドル:** アトリビュートをアドレス指定してそれにアクセスする 16 ビット アドレスです。
- **アトリビュート タイプ:** アトリビュートに格納されるデータの種類を指定します。Bluetooth SIG が定義した 16 ビット UUID で表示されます。

  例えば、心拍数サービスの 16 ビット UUID は 0x180D で、デバイス名アトリビュートの UUID は 0x2A00 です。SIG が指定した 16 ビット UUID の一覧については、Bluetooth の**ホームページ**へアクセスして参照ください。

- **アトリビュート バリュー:** アトリビュートに格納される実際の値です。
- **アトリビュート パーミッション:** アトリビュート アクセス、認証、および権限の要件を指定します。これは上層の仕様によって設定され、アトリビュート プロトコルを通してアクセスできません。

Figure 69 にデバイス名アトリビュートの構造を例として取り上げます。

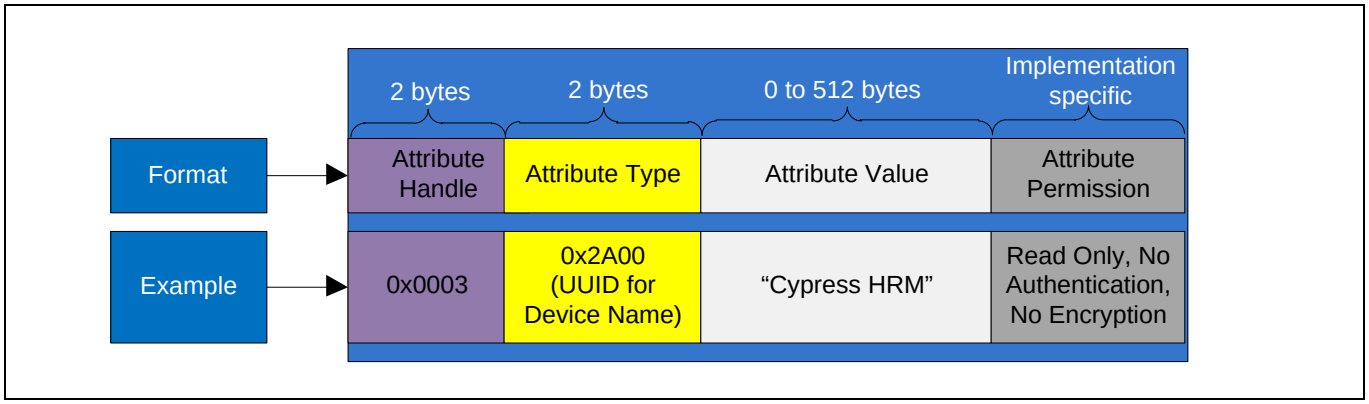


Figure 69　アトリビュート フォーマットの例

13.7.1 アトリビュート階層

アトリビュートはデータを表示する ATT／GATT での構成要素です。アトリビュートは次の 2 グループに大きく分けられ、データ階層とデータ抽象化を提供します。

- **キャラクタリスティック:** システム情報や有意味のデータを表すアトリビュートのコレクションです。キャラクタリスティックは次のアトリビュートからなります。
  - キャラクタリスティック宣言アトリビュート: キャラクタリスティックの開始を定義します。
  - キャラクタリスティック バリュー アトリビュート: 実際のデータを格納します。
  - キャラクタリスティック ディスクリプタ アトリビュート: これらは任意のアトリビュートであり、キャラクタリスティック値の追加情報を提供します。

「電池残量」は電池サービス (BAS) のキャラクタリスティックの一例です。電池残量をパーセント値で表示するのはキャラクタリスティック ディスクリプタの一例です。

Figure 70 に、例として電池残量のキャラクタリスティックの構造を示します。

  - キャラクタリスティックの最初の部分はそのキャラクタリスティックの宣言 (キャラクタリスティックの開始) であり、**Figure 70** の例に「Battery Level Characteristic」(電池残量キャラクタリスティック) で示されています。
  - 次は実際のキャラクタリスティック値 (実際のデータ) です。電池残量キャラクタリスティックの場合は現時点の電池残量です。電池残量は「65」、「90」など全測定範囲に対する割合 (パーセント) で表示されます。
  - キャラクタリスティック ディスクリプタはキャラクタリスティックの値の意味を解明するのに必要な追加情報を提供します。例えば、電池残量のキャラクタリスティック表示形式ディスクリプタは電池レベルがパーセントで表示されることを示します。よって、「90」を読み出すと、GATT クライアントは 90mV や 90mAh ではなく 90%と理解します。同様に、(**Figure 70** に表示されない) 有効範囲のキャラクタリスティック ディスクリプタは電池レベルの範囲が 0～100 パーセントであることを示します。
  - クライアント キャラクタリスティック コンフィギュレーション ディスクリプタ (CCCD) はもう 1 つの一般的なキャラクタリスティック ディスクリプタであり、GATT クライアントが GATT サーバー上のキャラクタリスティックの動作を設定することを可能にします。GATT クライアントが 0x01 の値をキャラクタリスティックの CCCD に書き込むと、(後節で説明するように) GATT サーバーからの通知が非同期に送信されます。電池レベル キャラクタリスティックの場合は、電池残量の CCCD に 0x01 の値を書き込むと、電池サービスが自体の電池残量を定期的または電池残量の変化時に通知することが可能になります。

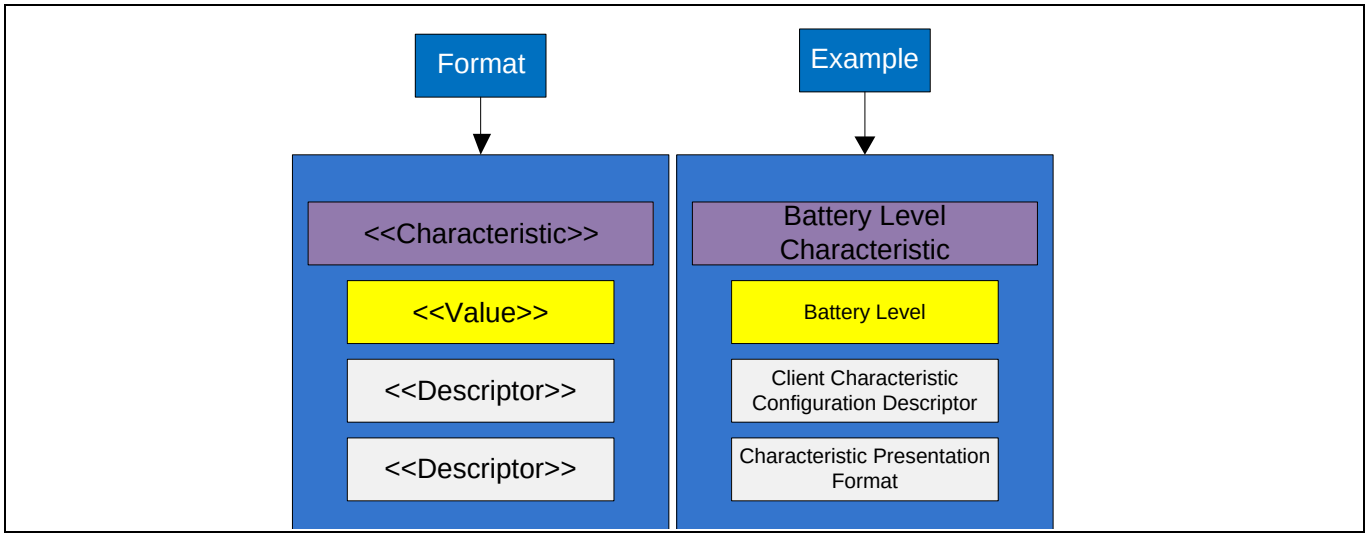


Figure 70　キャラクタリスティックのフォーマットと例

- **サービス:** GATT サーバーが実行する機能を定義するアトリビュートの一種です。サービスはキャラクタリスティックのコレクションであり、他のサービスを含むこともあります。サービスは、関連のあるデータのグループを設立し、データ階層を提供するのに使用されます。心拍数サービス (**HRS**) の例は、**Figure 71** を参照ください。

サービスは一次サービスと二次サービスの 2 種類に分けられています。一次サービスはデバイスの主な機能を提供し、二次サービスは追加機能を提供します。例えば、心拍数監視機器の場合には、HRS は一次サービスで、BAS は二次サービスです。

サービスは GATT サーバーにある他のサービスも含むことがあります。含まれているすべてのサービスは新しいサービスの一部になります。

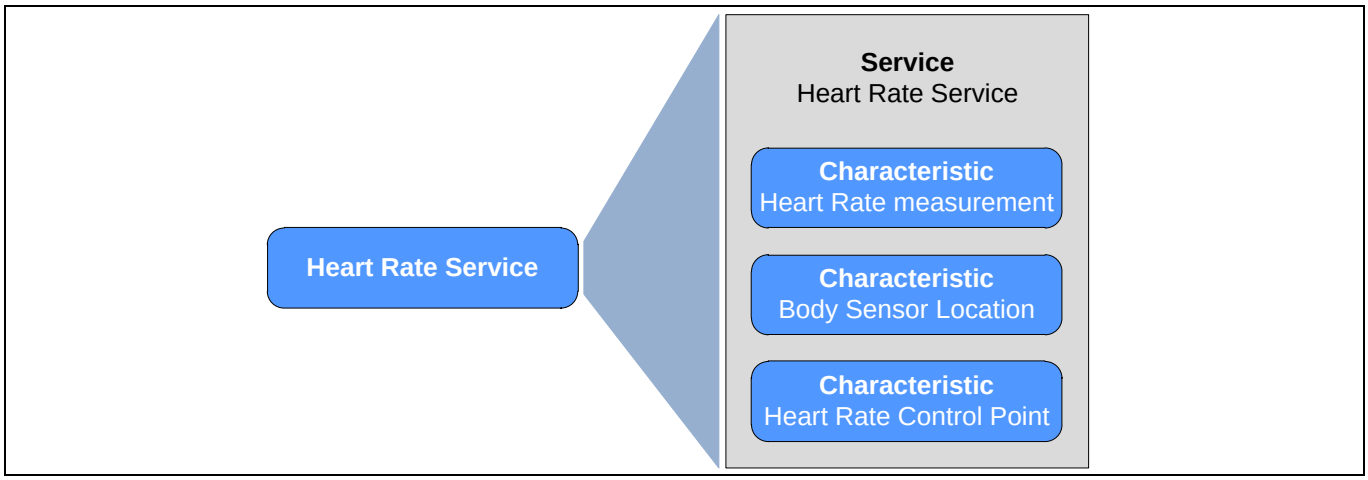


Figure 71　Bluetooth® LE 心拍数サービスの例

Bluetooth® LE における「プロファイル」という言葉は、特定のエンド アプリケーションを共に実現するサービスとその動作のコレクションです。心拍数プロファイル (**HRP**) は Bluetooth® LE プロファイルの一例で、心拍数監視機器の作成に必要なすべてのサービスを定義します。詳細については、「**Generic Access Profile (GAP)**」節を参照ください。

Figure 72 に、本節の上記に定義されたアトリビュート、キャラクタリスティック、サービス、およびプロファイルを使用するデータ階層を示します。

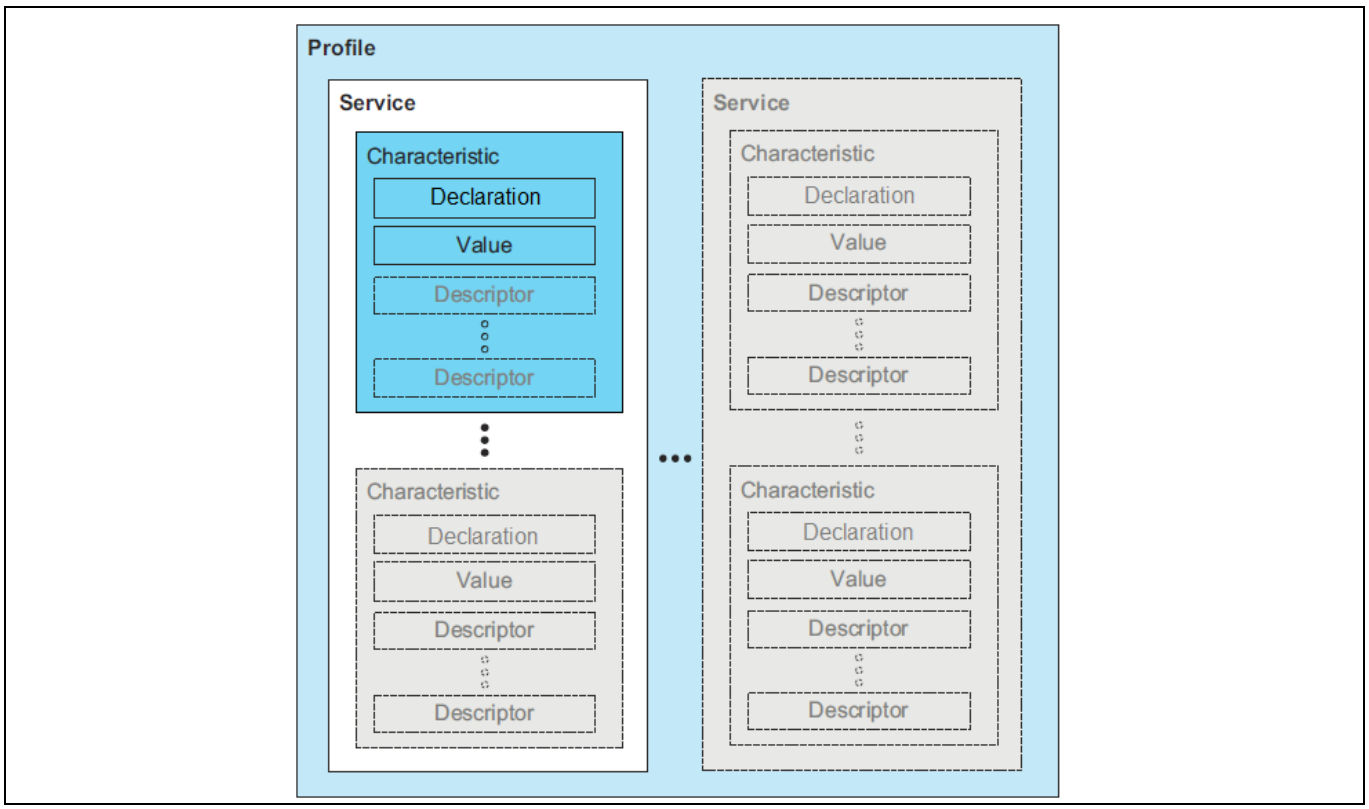


Figure 72 Bluetooth® LE データ階層*

*Bluetooth SIG により提供された無料イメージ

13.7.2 アトリビュート動作

前節で定義したアトリビュートは、次の 5 つの基本的方法でアクセスされます。

- **読み出し要求:** GATT クライアントはアトリビュート バリューの読み出しのためにこの要求を GATT サーバーに送信します。各要求ごとに GATT サーバーは GATT クライアントに応答を送信します。この読み出し要求の一例は、スマートフォンが心拍数監視機器の電池残量キャラクタリスティックを読み出すことです (**Figure 70** を参照ください)。
- **書き込み要求:** GATT クライアントはアトリビュート バリューの書き込みのためにこの要求を GATT サーバーに送信します。GATT サーバーはその値が書き込まれたか否かを GATT クライアントに返答します。書き込み要求の一例は、スマートフォンが 0x01 の値を電池残量キャラクタリスティックの CCCD に書き込んで通知を有効にすることです。
- **書き込みコマンド:** GATT クライアントはアトリビュート バリューの書き込みのためにこのコマンドを GATT サーバーに送信します。GATT サーバーはこのコマンドに対して何の返答も送信しません。例えば、Bluetooth® LE Immediate Alert Service (**IAS**) は、IAS ロケーター (スマートフォンなど) から IAS ターゲット デバイス (Bluetooth® LE キーフォブなど) でのアラート (LED をオンにする、ブザーを鳴らす、振動モーターを駆動することなど) をトリガーするために書き込みコマンドを使用します。
- **通知:** GATT サーバーはアトリビュートの新しい値を通知するためにこの通知を GATT クライアントに送信します。GATT クライアントはこの通知に対して何の確認も送信しません。例えば、心拍数監視機器はその CCCD に 0x01 が書き込まれると心拍数測定通知をスマートフォンに送信します。

- **指示:** GATT サーバーはこの種類のメッセージを送信します。GATT クライアントはそれに対して常に確認を行います。例えば、Bluetooth® LE 健康体温計サービス (**HTS**) は測定した体温の値を、スマートフォンのような健康体温計コレクターに確実に送信するためにこれらの指示を使用します。

13.8 Generic Attribute Profile (GATT)

GATT はアトリビュートの検出と使用方法を定義します。GATT は次の 2 つのロールのいずれかで動作します。

- **GATT クライアント:** データを要求するデバイス (例えば、スマートフォン) です。
- **GATT サーバー:** データを提供するデバイス (例えば、心拍数モニター) です。

Figure 73 に、心拍数監視機器を例として GATT 層内のクライアントとサーバーのアーキテクチャを示します。心拍数監視機器は多くのサービス (HRS、BAS、およびデバイス情報サービス) を提供し、それぞれのサービスは **Figure 70** に示すようにキャラクタリスティック値とディスクリプタを含む 1 つ以上のキャラクタリスティックを持っています。

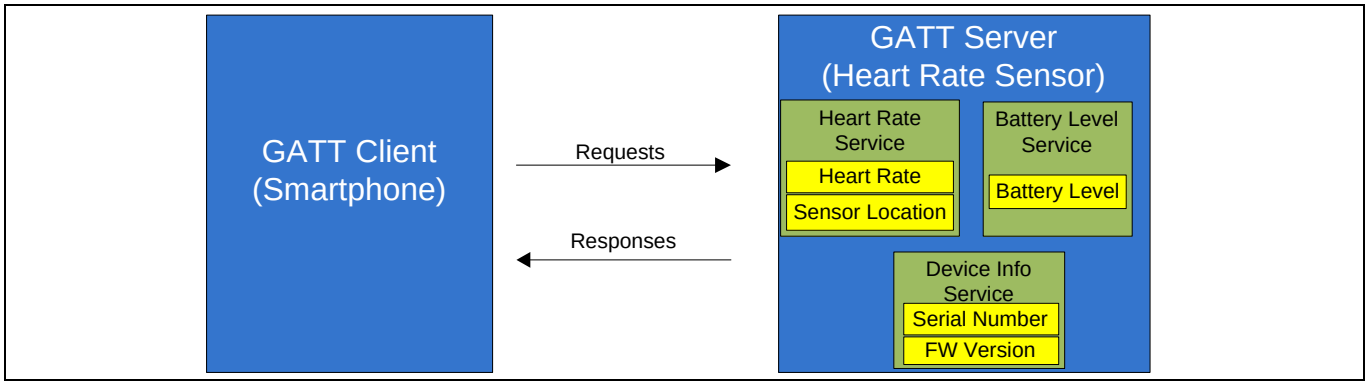


Figure 73 GATT クライアントとサーバーのアーキテクチャ

Bluetooth® LE 接続が Link Layer レベルで確立された後、GATT クライアントは (最初に接続された Bluetooth® LE デバイスについて何も分からず) 「サービス発見」というプロセスを開始します。サービス発見の一部として、GATT サーバーでのすべてのサービス, キャラクタリスティック, アトリビュートの一覧を得るために GATT クライアントは GATT サーバーに多くの要求を送信します。サービス発見が完了した後、GATT クライアントは、前述したアトリビュート動作を使用して GATT サーバーが提供する情報の読み出しや修正に必要な情報を得ました。

13.9 Generic Access Profile (GAP)

GAP 層はデバイス アドレス、デバイス名などのデバイス固有の情報、および発見／接続／ボンディング方法を提供します。このプロファイルは、デバイスの発見／接続方法、使用可能なサービス一覧、およびサービスの使用方法を定義します。**Figure 75** に心拍数プロファイルの例を示します。

GAP 層は次の 4 つのロールのいずれかで動作します。

- **ペリフェラル:** デバイスを GAP セントラルに接続することを可能にする宣伝のロールです。セントラルとの接続が確立された後、デバイスはスレーブとして動作します。例えば、遠隔装置に測定された心拍数を報告する心拍数センサーは GAP ペリフェラルとして動作します。
- **セントラル:** 宣伝をスキャンしてペリフェラルとの接続を開始する GAP ロールです。この GAP ロールはペリフェラルとの接続が確立した後、マスターとして動作します。例えば、ペリフェラル (心拍数センサー) から心拍数計測データを受信したスマートフォンは GAP セントラルとして動作します。

付録 E: Bluetooth® LE プロトコル

- **ブロードキャスター:** データを放送するのに使用される宣伝ロールです。Bluetooth® LE 接続を確立できず、データ交換にも関与しません (要求／応答動作がありません)。このロールは、聞き手が有無かを問わずデータを連続して送信するラジオ放送局と同様の役割を果たしています。これは一方向データ通信です。何の応答も求めなく情報を連続して放送するビーコンは GAP ブロードキャスターの典型的な例です。
- **オブザーバー:** 宣伝をスキャンするが宣伝デバイスとの接続を確立しない聞き取りロールです。これはブロードキャスター ロールの逆のロールです。情報を連続して聞き取るが、情報源と通信できないラジオ受信機と同様に動作します。ビーコンからの信号を連続的に聞き取るスマートフォン アプリケーションは GAP オブザーバーの典型的な例として取り上げられます。

Figure 74 に、インフィニオンの Bluetooth® LE Pioneer Kit がペリフェラルで、スマートフォンがセントラル デバイスである一般的な Bluetooth® LE システムを示します。また、Bluetooth® LE プロトコルの層間の相互作用、およびセントラルとペリフェラル デバイス上に該当するロールも示しています。

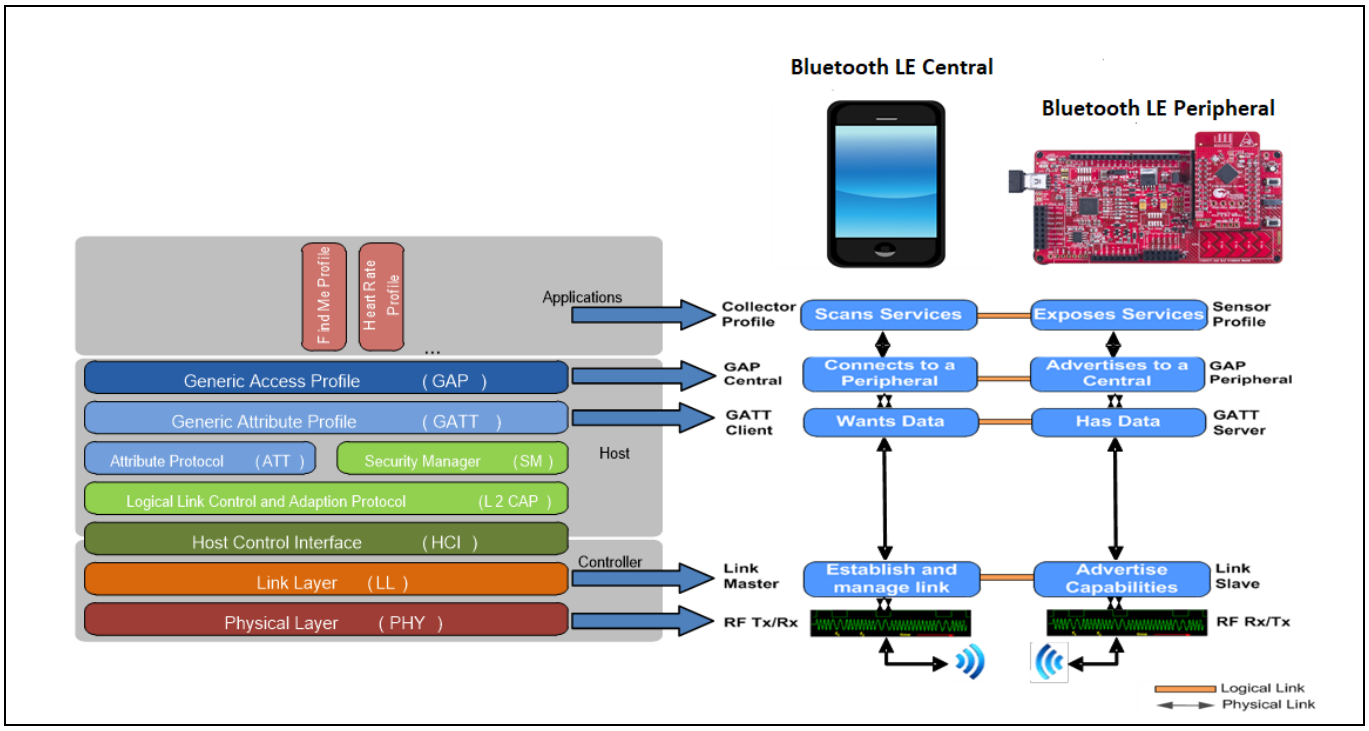


Figure 74 Bluetooth® LE システムの設計

Figure 75 に、心拍数アプリケーションを備えたスマートフォンがセントラルとして動作し、心拍数センサーがペリフェラルとして動作する例を示します。心拍数監視機器は心拍数センサー プロファイルを実装し、データを受信するスマートフォンは心拍数コレクター プロファイルを実装します。

この例では、心拍数センサー プロファイルは次の 2 つの標準サービスを実装します。1 つ目は心拍数サービスで、3 つのキャラクタリスティック (心拍数計測キャラクタリスティック, 身体センサー位置キャラクタリスティック, および心拍数制御点キャラクタリスティック) からなります。第 2 のサービスはデバイス情報サービスです。Link Layer では、心拍数計測デバイスがスレーブで、スマートフォンがマスターです。心拍数のサービスとプロファイルの詳細については、Bluetooth Development Portal サイトへアクセスして参照ください。

付録 E: Bluetooth® LE プロトコル

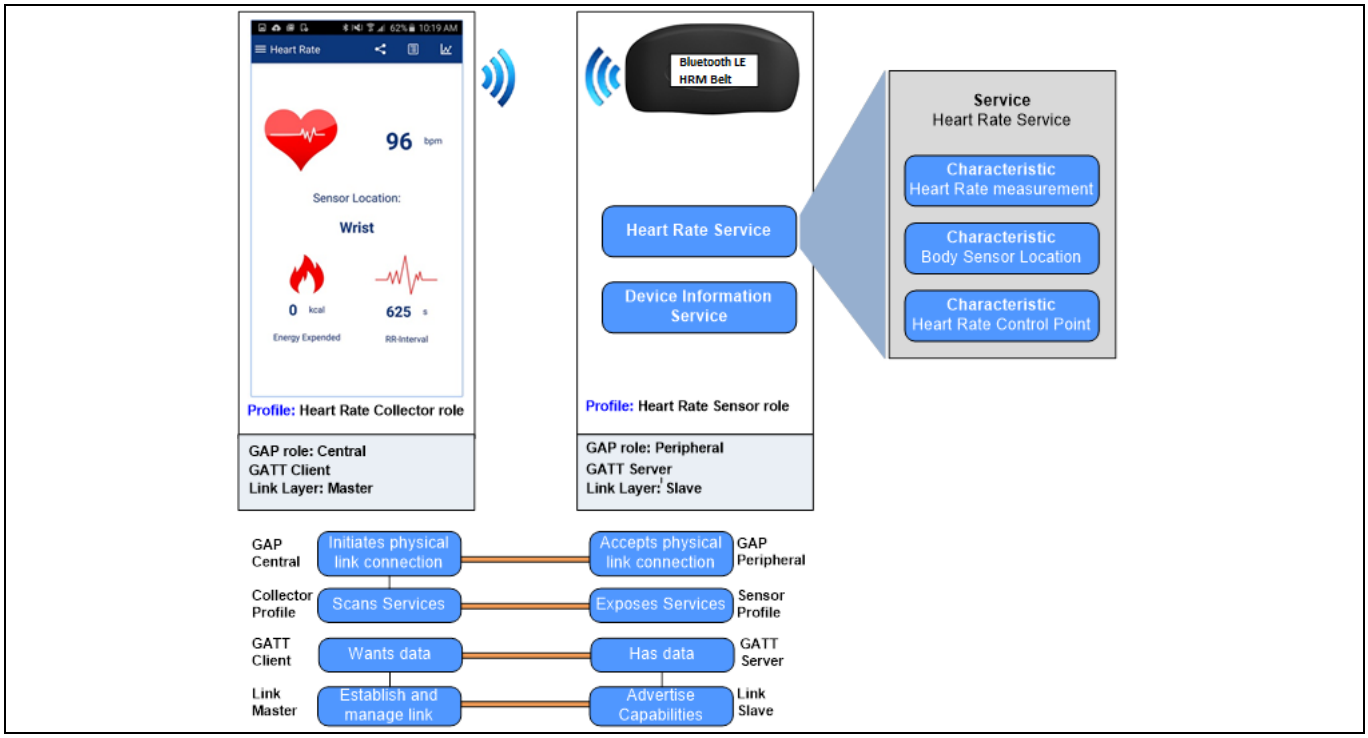


Figure 75 Bluetooth® LE 心拍数監視システム

関連アプリケーション ノート

[1] **AN94020 - Getting Started with PRoC Bluetooth® Low Energy**

[2] **AN96841 - Getting started with EZ-Bluetotth® LE Creator modules**

[3] **AN91184 - PSoC™ 4 BLE - Designing Bluetooth® Low Energy Applications**

[4] **AN91162 - Creating a Bluetooth® Low Energy Custom Profile**

[5] **AN92584 - Designing for Low Power and Estimating Battery Life for Bluetooth® Low Energy Applications**

[6] **AN95089 - PSoC™ 4/PRoC Bluetooth® Low Energy Crystal Oscillator Selection and Tuning Techniques**

[7] **AN91445 - AN91445-アンテナ設計および RF レイアウトのガイドライン**

[8] **AN97060 - PSoC™ 4 BLE and PRoC Bluetooth® LE - Over-The-Air (OTA) Device Firmware Upgrade (DFU) Guide**

改訂履歴

Document version	Date of release	Description of changes
**	2014-12-24	これは英語版 001-91267 Rev. *F を翻訳した日本語版 002-10056 Rev. **です。
*A	2017-04-28	更新されたロゴと著作権。
*B	2019-01-15	これは英語版 001-91267 Rev. *H を翻訳した日本語版 002-10056 Rev. *B です。
*C	2021-07-07	これは英語版 001-91267 Rev. *I を翻訳した日本語版 Rev. *C です。

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-07-07
Published by
Infineon Technologies AG
81726 Munich, Germany

© 2021 Infineon Technologies AG. All Rights Reserved.

Do you have a question about this document?
Go to www.cypress.com/support

Document reference
002-10056 Rev. *C

重要事項

本文書に記載された情報は、いかなる場合も、条件または特性の保証とみなされるものではありません（「品質の保証」）。本文に記された一切の事例、手引き、もしくは一般的価値、および／または本製品の用途に関する一切の情報に関し、インフィニオンテクノロジーズ（以下、「インフィニオン」）はここに、第三者の知的所有権の不侵害の保証を含むがこれに限らず、あらゆる種類の一切の保証および責任を否定いたします。

さらに、本文書に記載された一切の情報は、お客様の用途におけるお客様の製品およびインフィニオン製品の一切の使用に関し、本文書に記載された義務ならびに一切の関連する法的要件、規範、および基準をお客様が遵守することを条件としています。

本文書に含まれるデータは、技術的訓練を受けた従業員のみを対象としています。本製品の対象用途への適合性、およびこれら用途に関連して本文書に記載された製品情報の完全性についての評価は、お客様の技術部門の責任にて実施してください。

本製品、技術、納品条件、および価格についての詳しい情報は、インフィニオンの最寄りの営業所までお問い合わせください（**www.infineon.com**）。

警告事項

技術的要件に伴い、製品には危険物質が含まれる可能性があります。当該種別の詳細については、インフィニオンの最寄りの営業所までお問い合わせください。

インフィニオンの正式代表者が署名した書面を通じ、インフィニオンによる明示の承認が存在する場合を除き、インフィニオンの製品は、当該製品の障害またはその使用に関する一切の結果が、合理的に人的傷害を招く恐れのある一切の用途に使用することはできないこと予めご了承ください。