

Objective

This example project demonstrates how to configure PSoC 4 BLE & PProC BLE devices in low power mode during device startup. BLE device startup time is dominated by 32kHz WCO and 24MHz ECO clock startup time and this example project shows how to configure the BLE device in DeepSleep mode (1.2uA current consumption) during clock startup instead of the default active mode (>5mA current consumption).

Overview

PSoC 4 BLE and PProC BLE devices require two clock sources for designing a low power BLE system:

1. 32kHz Watch Crystal Oscillator (WCO) for accurately maintaining the interval between BLE advertisement or connection state packets when the system is in low power mode.
2. 24MHz External Crystal Oscillator (ECO) for BLE RF packet transmission and reception

The maximum startup time specification for WCO is 500ms and that for the ECO is around 800us.

Typically, these two clocks are enabled as part of the BLE device startup or boot code. The startup code waits in active mode for the maximum startup time required for WCO and ECO (total time of >501ms) in active mode and thereby consuming unnecessary higher current while waiting for the clocks to startup. This behavior is not desirable for small battery operated devices or devices that often power up/down the BLE device.

The obvious solution to this problem is to configure the BLE device in lowest possible power mode during the clock startup period and wakeup the device after the clock startup period expires. This lowest possible power mode in PSoC 4 BLE and PProC BLE device is the **DeepSleep** mode and the wakeup source that can transition the device from DeepSleep to Active mode after the clock startup period expires is the **Watchdog Timer**. See PSoC 4 BLE device [technical reference manual](#) for more details on BLE device power modes, ECO & WCO startup time and watchdog timers.

This example project demonstrates how to start WCO & ECO clocks in application code (= bypass WCO & ECO start in device startup code) and use the BLE device's DeepSleep mode and watchdog timer to keep the power consumption under check during the WCO and ECO startup time.

After the device starts up in low power mode, an **AltBeacon** is instantiated in BLE Component that continuously advertises in non-connectable mode with the payload as per the AltBeacon specification. The advertisement interval for AltBeacon is set at 100ms in this example project.

Requirements

Design Tool: [PSoC Creator 3.1 SP2](#) with built-in GCC 4.8.4

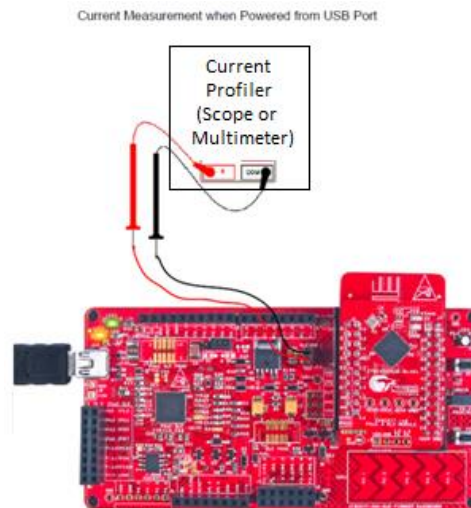
Associated Devices: All PSoC 4 BLE & PProC BLE devices

Required Hardware: [CY8CKIT-042-BLE Bluetooth® Low Energy \(BLE\) Pioneer Kit](#)

Hardware Setup

The BLE Pioneer Kit has all the necessary hardware required for this example. [Figure 1](#) shows the hardware setup for this example. USB connection is required for programming the BLE Pioneer Kit and also to display user messages on host PC's serial port terminal program. Use a current probe or equivalent for profiling the device startup mode current consumption. Refer to [BLE Low Power](#) application note for current measurement and profiling details.

Figure 1: Kit Setup



PSoC Creator Schematic

Figure 2. PSoC Creator Schematic

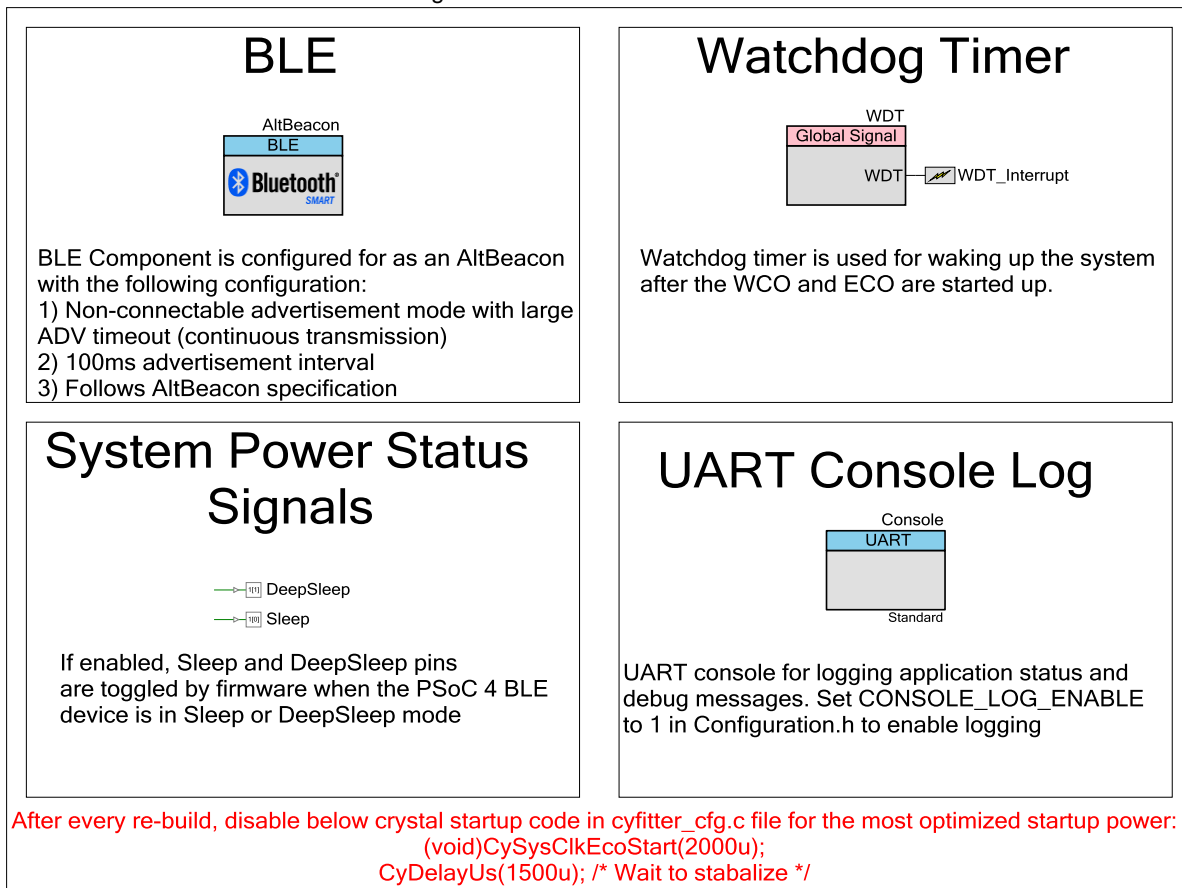
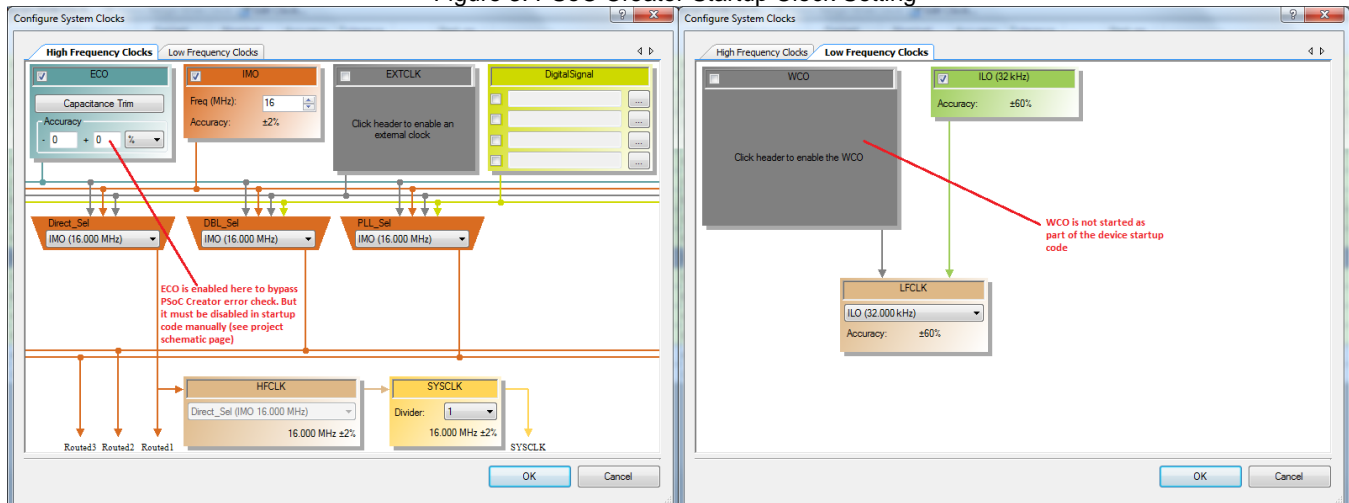


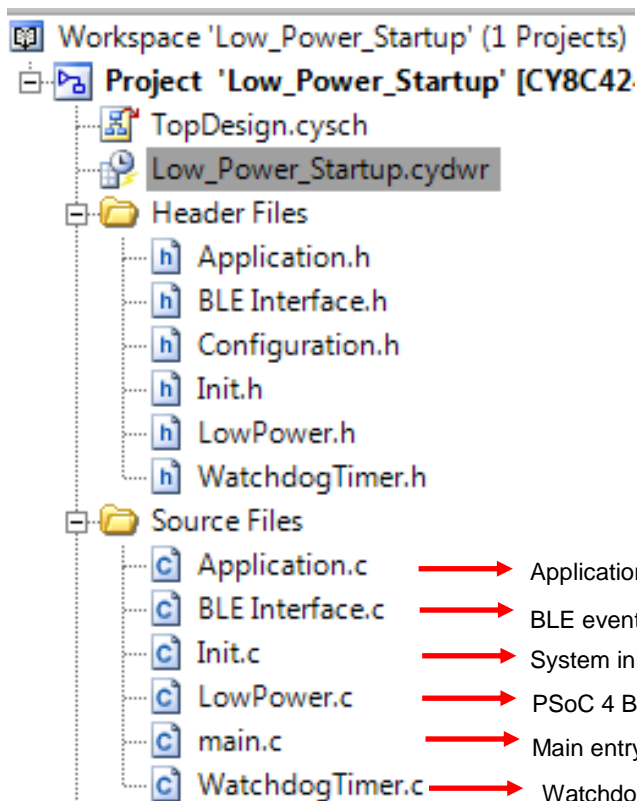
Figure 3. PSoC Creator Startup Clock Setting



- Enable ECO in Creator clocks tab only to prevent Creator build error. Comment out the ECO startup code in cyfitter_cfg.c file to achieve lowest device startup average current.
- Disable WCO and set ILO as the LFCLK source during device startup. See Figure 2 and Figure 3 for details.

Firmware

Figure 4. Low Power Startup Example Project Source Files



- Application.c → Application layer code. This is almost zero for the AltBeacon
- BLE Interface.c → BLE event handler and component initialization code
- Init.c → System initialization source code. Also contains the WCO/ECO start code
- LowPower.c → PSoC 4 BLE device low power mode configuration source code
- main.c → Main entry point source code for the example project
- WatchdogTimer.c → Watchdog timer source code for waking up device after WCO/ECO starts

The firmware source files for this example project and a short description of each of the source files is shown in [Figure 4](#). The firmware is simple and all the low power WCO/ECO startup firmware at the application layer is within the **#if LOW_POWER_STARTUP_ENABLE** block. The firmware configuration flags are located in Configuration.h file and a short description of each of the configuration flags are shown in [Figure 5](#).

Figure 5. Low Power Startup Firmware Configuration Flags

```

/*****
 * Configuration flags
 *****/
#define DEBUG_ENABLE (1u) /* Enables GPIO toggling on different power mode (Sleep, DeepSleep)
 * transitions. Poor man's power profiler */

#define CONSOLE_LOG_ENABLE (1u) /* Enable UART console logging */

#define LOW_POWER_STARTUP_ENABLE (1u) /* Enable this to configure the device in DeepSleep mode when WCO & ECO are
 * starting up */

```

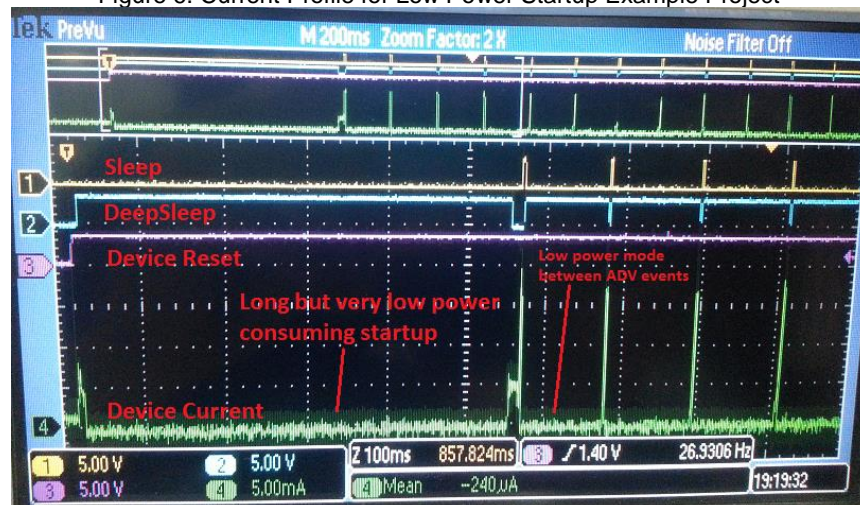
Test Setup

- Enable required firmware configuration flags in Configuration.h file
- Build this example project in PSoC Creator 3.1 SP2 or later
- Program the hex file generated onto BLE-Pioneer Kit Baseboard.

Test Procedure

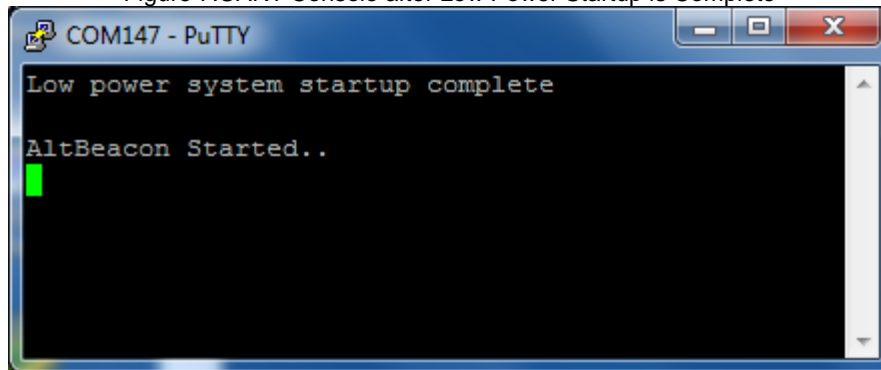
- Connect the BLE-Pioneer Kit Baseboard to Windows host machine using the USB cable
- Open a terminal emulator such as Putty or Tera Term for the BLE-Pioneer Kit to see the user messages. The COM settings are: Baud rate – 115200 bps, Data bits – 8, Stop bits – 1, Parity – None.
- Measure the current consumed by the device during power up by using a current probe. See [Figure 1](#).
- Monitor the Sleep (P1.0), DeepSleep (P1.1) and Reset signals to see the device state during ECO and WCO startup. The measured current profile should look similar to [Figure 6](#).

Figure 6. Current Profile for Low Power Startup Example Project



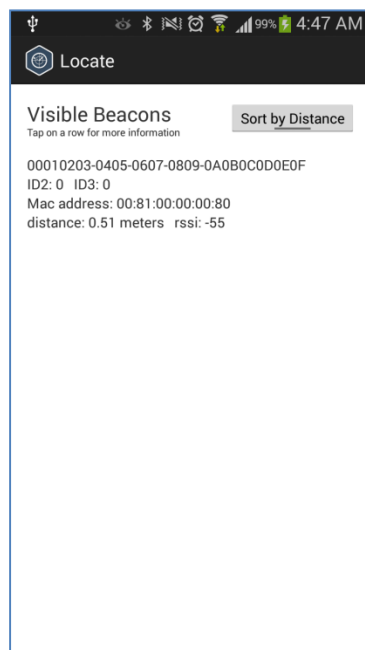
- After the device starts up in low power mode, it starts an AltBeacon and continuously advertises a beacon as per the AltBeacon specification. The UART console should look similar to [Figure 7](#) after starting the AltBeacon

Figure 7. UART Console after Low Power Startup is Complete



- To verify working of AltBeacon, use [Locate Beacon](#) or similar app in listen mode on your mobile device as shown in [Figure 8](#).

Figure 8. AltBeacon on Locate Beacon App



Related Documents

Table 1 lists all relevant application notes, code examples, knowledge base articles, device datasheets, and Component datasheets.

Table 1. Related Documents

Document	Title	Comment
AN91267	Getting Started with PSoC 4 BLE	Provides an introduction to PSoC 4 BLE device that integrates a Bluetooth Low Energy radio system along with programmable analog and digital resources.
PSoC 4 BLE TRM	PSoC 4 BLE device TRM	A detailed description of all the features and internal architecture of PSoC 4 BLE device.
AN92584	Designing for Low Power and Estimating Battery Life for BLE Applications	A detailed guide to designing low power BLE applications.