PSoC 4 BLE – BLE RTC

Objective

This example project demonstrates how to create a Real Time Clock (RTC) using configurable watchdog timers of PSoC 4 BLE device. The initial time for RTC is set by reading the time from a BLE time server (iOS device for example) using BLE Current Time Service (CTS).

Overview

PSoC 4 BLE and PRoC BLE devices have a watchdog timer that is driven by low frequency (LF) clock of the device. The source of the LF clock can either be:

1. Inaccurate internal Low Speed Oscillator (ILO)
2. Accurate Watch Crystal Oscillator (WCO)

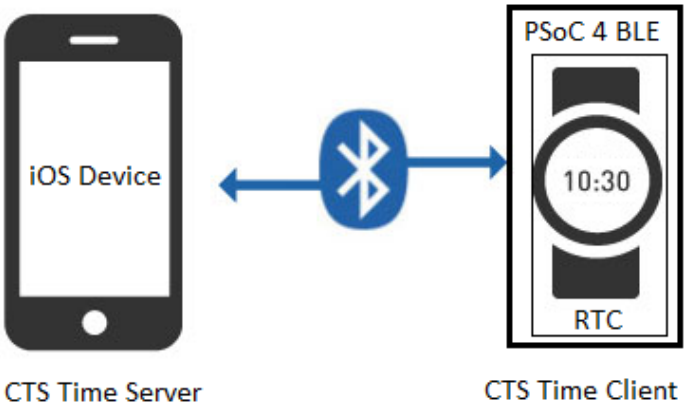
WCO is required for low power mode of operation of the BLE interface and most of the BLE end applications will use a WCO in their design. The accuracy of the WCO in a typical BLE design is <70ppm across PVT when using a 10ppm accurate external crystal.

This highly accurate WCO can be used in conjunction with watchdog times of PSoC 4 BLE device to provide a 1 second pulse interrupt that can drive a firmware RTC module. This example project demonstrates how to instantiate an RTC in PSoC 4 BLE device using a combination of WCO, watchdog counter and firmware RTC components. See PSoC 4 BLE device technical reference manual for more details on WCO and watchdog timers.

BLE also provides Current Time Service (CTS) that lets a GATT time client to get the current timing information from a GATT time server. The BLE CTS server is in-built in iOS and all the iOS devices allow a BLE GATT time client to connect to them and extract the timing information. This example project utilizes this feature of iOS devices and instantiates a CTS GATT client that initially fetches the current time value on establishing a BLE connection with an iOS device and the RTC is initialized with this timing information.

The functional block diagram of this example is as shown in Figure 1.

Figure 1: PSoC 4 BLE RTC Example Block Diagram



Requirements

Design Tool: PSoC Creator 3.1 SP1 with built-in GCC 4.8.4

Associated Devices: All PSoC 4 BLE & PRoC BLE devices

Required Hardware: CY8CKIT-042-BLE Bluetooth® Low Energy (BLE) Pioneer Kit

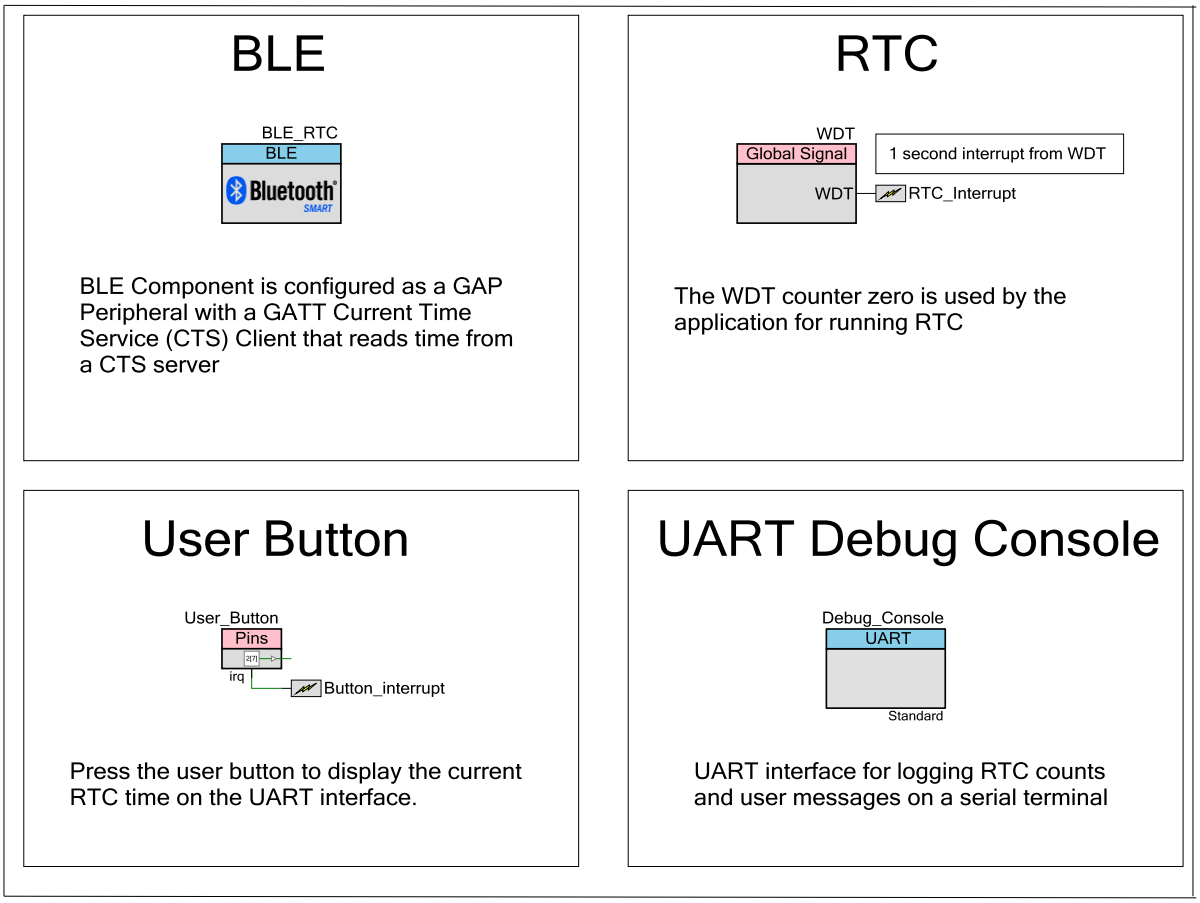
Hardware Setup

The BLE Pioneer Kit has all the necessary hardware required for this example. Figure 2 shows the hardware setup for this example. USB connection is required for programming the BLE Pioneer Kit and also to display user messages and RTC timing information on host PC's serial port terminal program.

Figure 2: Kit Setup

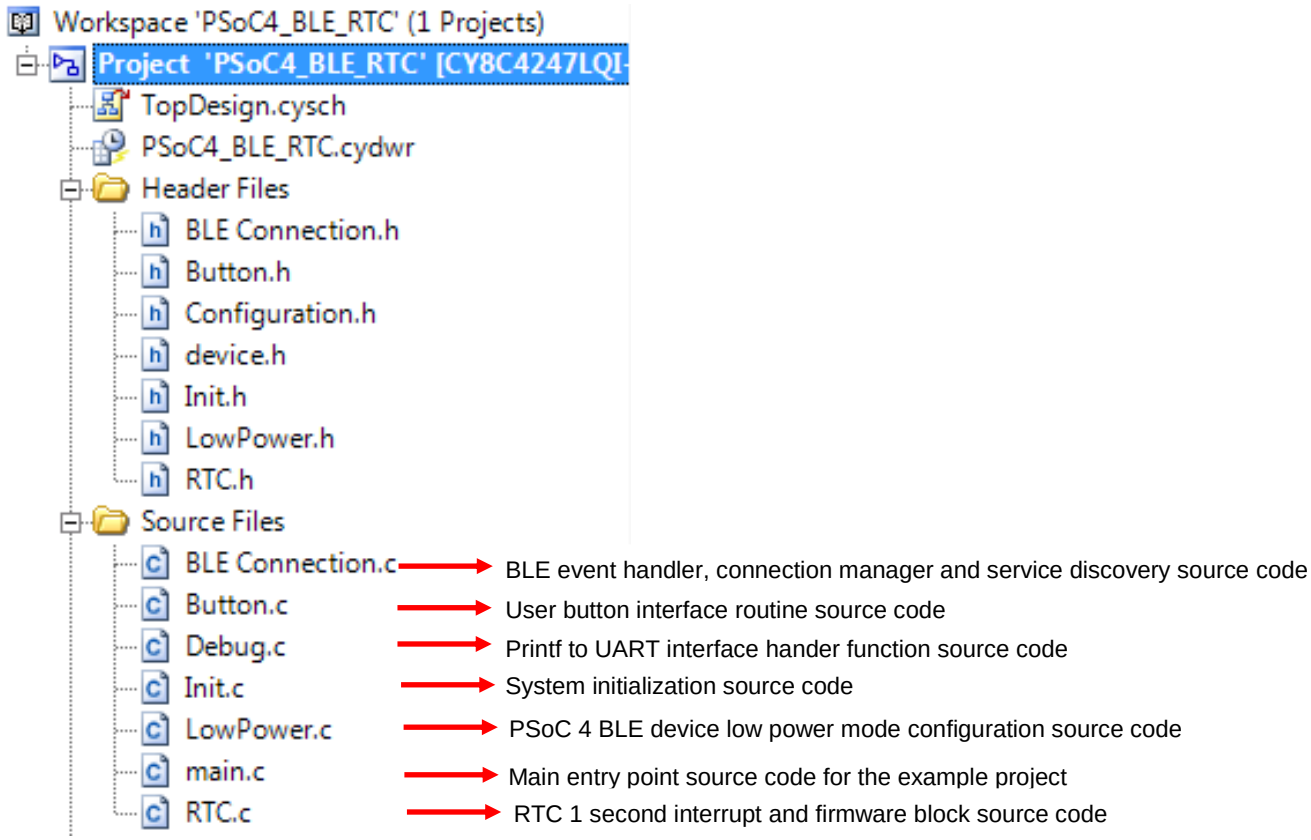
PSoC Creator Schematic

Figure 3. PSoC Creator Schematic



Firmware

Figure 4. BLE RTC Example Project Source Files



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 modular and provides multiple configuration flags to test and debug different RTC modes of operation. All 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. BLE RTC Firmware Configuration Flags

```
/***************************************
*    Project configuration flags
***************************************/
#define BLE_GATT_CLIENT_ENABLE                  (1u) /* Enables the GATT client to discover and fetch time from the
                                                      * peer device's GATT current timer server */

#define RTC_ENABLE                              (1u) /* Enable RTC operation in firmware */

#define DISCONNECT_BLE_AFTER_TIME_SYNC          (1u) /* After the RTC is synced with BLE time server's time,
                                                      * disconnect the link to reduce system power consumption */

#define RESTART_ADV_ON_DISCONNECTION            (0u) /* After the BLE interface is disconnected, restart the
                                                      * advertisement */

#define CONSOLE_LOG_ENABLED                     (0u) /* Enable UART console logging */

#if CONSOLE_LOG_ENABLED
#define DISPLAY_ON_BUTTON_PRESS                 (1u) /* When UART logging is enabled, on the press of user button
                                                      * display the RTC time on the UART interface */
#endif
```

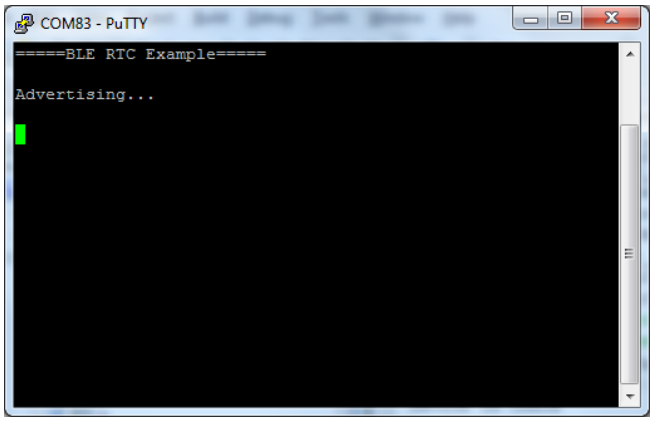
Test Setup

- Enable required firmware configuration flags in Configuration.h file
- Build this example project in PSoC Creator 3.1 SP1 or later
- Program the hex file generated onto BLE-Pioneer kit Baseboard.

Test Procedure

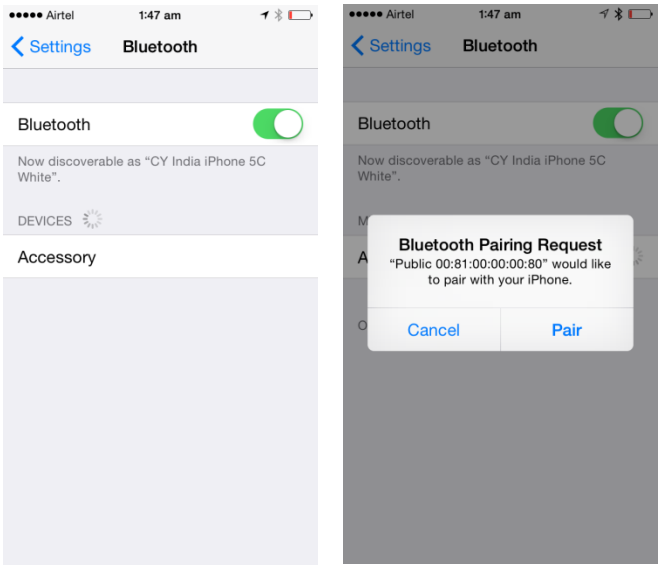
- Connect the BLE-Pioneer Kit Baseboard to Windows host machine using the USB cable
- Open a terminal emulator such as Putty or Tera Term for the BLE-Pioneer Kit. The COM settings are: Baud rate – 115200 bps, Data bits – 8, Stop bits – 1, Parity – None.
- Reset the device (by pressing the Reset switch SW1 on BLE-Pioneer Kit Baseboard) to see the output shown in Figure 6 on the terminal. The device is advertising at this point and waiting to be connected with an iOS device.

Figure 6. BLE RTC Firmware Startup



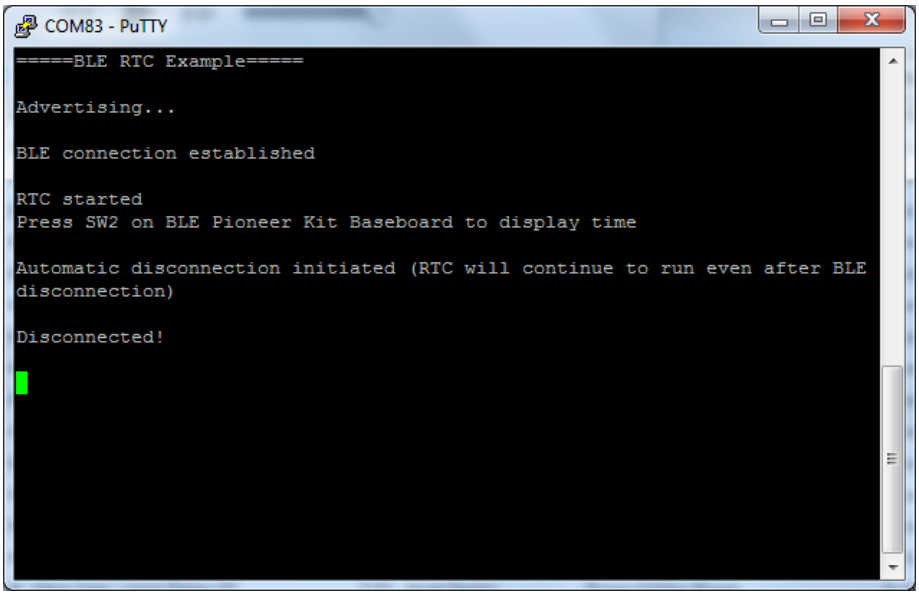
- For this example, we are using an iPhone 5C. On the iPhone, navigate to the phone Settings, enable Bluetooth. The BLE Pioneer Kit is listed as one of the devices nearby. Touch it to connect. See Figure 7.

Figure 7. iPhone Settings and RTC Device Listing



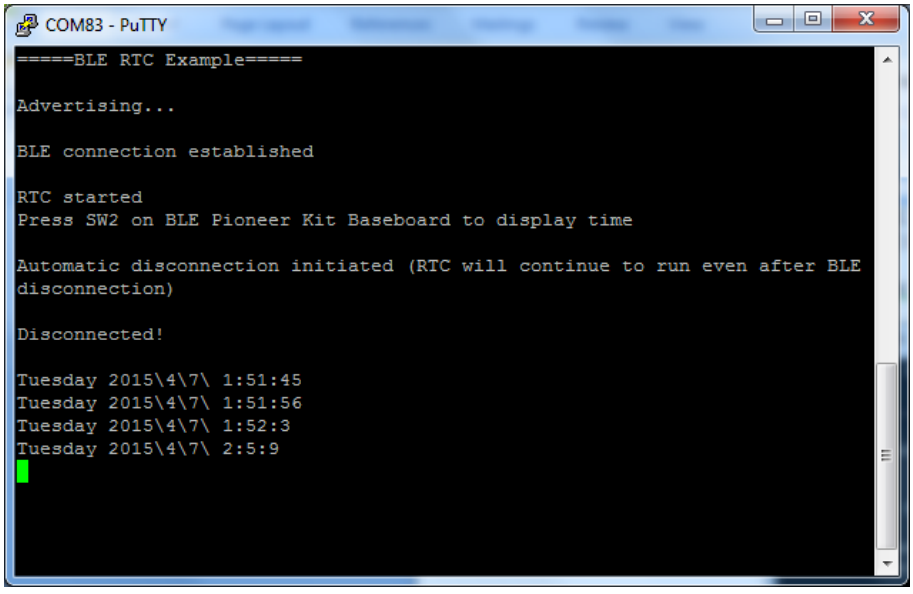
- Once connected, RTC on PSoC 4 BLE device is started with the time synced from the peer iOS device (iPhone in this case) and disconnects the BLE link automatically as shown in Figure 8.

Figure 8. PSoC 4 BLE Connection, RTC Time Sync & Disconnection



- Now press SW2 on BLE-Pioneer Kit Baseboard to display the current time of the RTC as shown in Figure 9.

Figure 9. PSoC 4 BLE RTC Time Display



RTC Accuracy & Power Consumption

The accuracy of RTC is dependent on the WCO clock variation resulting from inherent crystal frequency stability, PSoC 4 BLE WCO oscillator block specification and WCO load capacitance tuning. On a bench level testing using BLE-Pioneer Kit, the accuracy of the RTC was found to be 58ppm (5 seconds offset per day). When the BLE interface is completely shut down and only RTC is active in the design, then PSoC 4 BLE device consumes an average current of <1.8uA.

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.