BT800 WLAN Coexistence Schemes and LED Indication
Application Note v1.0

Laird’s BT800 USB HCI module supports WLAN coexistence schemes (Unity-3, Unity-3e, and Unity+). Supporting this feature are three coexistence pins (WLAN_ACTIVE, BT_PRIORITY, and BT_ACTIVE). This Application note illustrates how to configure WLAN coexistence schemes by modifying a few keys in the Bluetooth Persistent Store. Additionally, the application note covers how to reconfigure these coexistence pins to trigger LED indications for different Bluetooth states.

REQUIREMENTS

- BT800 Development board
- Windows PC

Note: Windows 7 is used in this guide. Windows 8 and XP should also be compatible.

- CSR BlueSuite
- CSR USB driver or CSR USB-SPI adaptor

Notes:
1. CSR BlueSuite is made available to customers under a Laird NDA. Customers should contact LT-wirelessinfo@lairdtech.com for obtaining the NDA. After BlueSuite is installed, PSTools can be found under the CSR folder.
2. CSR USB driver is provided along with the CSR BlueSuite. The CSR USB-SPI adaptor is available at: http://parts.digikey.com/1/parts/1406287-converter-usbspi-dev-sys-1808-1a.html

ENABLING WI-FI / BLUETOOTH COEXISTENCE

The following sections illustrate how to enable and configure Wi-Fi / Bluetooth coexistence schemes on the BT800 module. If you aren’t using the CSR driver, skip Installing CSR USB driver and move to CSR USB-SPI adaptor.

Installing CSR USB driver

When the BT800 development board is first plugged in the PC USB port, Windows installs the driver automatically. It is recognized as the “Generic USB Adapter” in the Windows device manager.

![Generic Bluetooth Adapter in Device Manager](image)

Figure 1: Generic Bluetooth Adapter in Device Manager

Complete the following steps to install the CSR USB driver:

1. Right-click on Generic Bluetooth Adapter, then click Update Driver Software.
2. Select **Driver** tab and click **Update Driver**.

3. Click **Browse my computer for driver software.**
4. Click **Let me pick from a list of device drivers on my computer.**
5. Click **Have Disk.**

6. Navigate to where the CSR driver is located on your computer, and select *CSRBlueCoreUSB.inf.* Proceed through the windows until software installation is complete.
Figure 5: CSR USB Driver installation

The BT800 development board is now recognized as “CSR BlueCore Bluetooth” in Windows device manager. It can be found by expanding “Universal Serial Bus controllers”. 
Figure 6: CSR BlueCore Bluetooth in Device Manager

You can skip to Open PStools, if you are not planning to use the SPI adaptor.

**CSR USB-SPI adaptor**

A RJ45 cable comes with the adaptor. Cut the RJ45 cable in halves. Plug in the RJ45 jack to the adaptor and connect the opened end to the development board.

![CSR USB SPI adaptor](image)

**Figure 7: CSR USB SPI adaptor**

Table 1 details RJ45 pins and their corresponding SPI signals.

**Table 1: RJ45 to SPI wiring**

<table>
<thead>
<tr>
<th>Signal</th>
<th>RJ45 Connector Pin</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPI_CS8</td>
<td>1</td>
</tr>
<tr>
<td>SPI_MOSI</td>
<td>5</td>
</tr>
<tr>
<td>SPI_CLK</td>
<td>7</td>
</tr>
<tr>
<td>SPI_MISO</td>
<td>3</td>
</tr>
<tr>
<td>GND</td>
<td>8</td>
</tr>
</tbody>
</table>
Open PStools

After invoking PStools, customers must select either USB transport or SPI transport to access the PSkeys.

On the BT800 DVK board, there is a 10-pin header for SPI and WLAN Coexistence, and SPI/PCM switch. Make sure to set the SPI/PCM switch to the SPI position before plugging in the development board to your PC if SPI interface will be used for opening PStools.

In PStools, select either USB or SPI transport as shown in Figure 10 and Figure 11.

![Figure 8: Wire numbering in RJ45 jack](image)

![Figure 9: BT800 DVK Board, SPI/PCM switch set to SPI](image)

![Figure 10: Selecting USB transport](image)
BT800 WLAN Coexistence Schemes and LED Indication
Application Note

Figure 11: Selecting SPI transport

PStools first reads all the PSkeys from the module. This process takes about 10 seconds, and then the screen in Figure 12 appears with “Bluetooth Address” highlighted on the list, and MAC address shown on the right. The Bluetooth address of your module will be different. The address is unique. Do not attempt to change it, or you might put the module in nonoperational condition.

Figure 12: PStools opened successfully

Selecting WLAN Coexistence Scheme

You must be very cautious when changing PSkey values. Unsupported WLAN coexistence schemes and GPIOs can render the development board non-operational. Only GPIO 0, GPIO 1, and GPIO 2 can be used for coexistence schemes. Four PSkeys are required to change the coexistence scheme and its associated pins:

- Coex Module: Coexistence Scheme
- Coex Signal: BT_ACTIVE
- Coex Signal: BT_STATUS
- Coex Signal: WLAN_DENY

BT800 is shipped with Unity-3 enabled as default.
The BT800 supports Unity-3, Unity-3e and Unity-Pro coexistence schemes (Figure 17).
To select a different supported scheme other than the default scheme, follow below steps.

1. After PSTools is first opened, click **View** and then **Sort by Ordinal**.

2. On the list, scroll down to **2480 Coex Module: Coexistence Scheme** and select a different scheme.

**Note:** Only select Unit-3, Unity-3e or Unity-Pro. Otherwise, you may break the module.
3. Click Set on the lower right of the window to write the selection to the module.

4. 3 GPIOs are required for coexistence scheme. Only GPIO 0, GPIO 1 and GPIO 2 can be assigned. First parameter configures the GPIO number (0000 for GPIO 0, 0001 for GPIO 1, 0002 for GPIO 2). Second parameter indicates the active logic level (0000 for disable, 0001 for active high, 0002 for active low). DO NOT use GPIO numbers other than these 3 GPIOs.

5. Scroll down to 2483 Coex Signal: BT_ACTIVE to select GPIO for BT_ACTIVE output.
6. Click **Set** on the lower right of the window to write the selection to the module.

   ![Figure 23: Commit BT_ACTIVE value](image)

7. Scroll down to **2484 Coex Signal: BT_STATUS** to select GPIO for BT_STATUS output.

   ![Figure 24: Select BT_STATUS value](image)

8. Click **Set** on the lower right of the window to write the selection to the module.

   ![Figure 25: Commit BT_STATUS value](image)

9. Scroll down to **2485 Coex Signal: WLAN_DENY** to select GPIO for WLAN_DENY input.

   ![Figure 26: Select WLAN_DENY value](image)

10. Click **Set** on the lower right of the window to write the selection to the module.
The following table illustrates supported configurations for coexistence schemes on the BT800.

<table>
<thead>
<tr>
<th>Coexistence Scheme</th>
<th>Value: Coex Scheme</th>
<th>Value: BT ACTIVE First</th>
<th>Second</th>
<th>Value: BT STATUS First</th>
<th>Second</th>
<th>Value: WLAN DENY First</th>
<th>Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unity-3</td>
<td>0x0003</td>
<td>0002</td>
<td>0001</td>
<td>0002</td>
<td>0001</td>
<td>0002</td>
<td>0001</td>
</tr>
<tr>
<td>Unity-3e</td>
<td>0x0007</td>
<td>0001</td>
<td>0001</td>
<td>0001</td>
<td>0001</td>
<td>0001</td>
<td>0001</td>
</tr>
<tr>
<td>Unity-Pro</td>
<td>0x000B</td>
<td>0000</td>
<td>0001</td>
<td>0000</td>
<td>0001</td>
<td>0000</td>
<td>0001</td>
</tr>
</tbody>
</table>

CONFIGURING GPIO PINS FOR LED INDICATION

If Wi-Fi / Bluetooth coexistence is not required, the reserved GPIO pins can be used for LED indication for different Bluetooth states. Five PSkeys are required to enable LED indication, which is disabled by default. They are as follows:

- Coex Module: Coexistence Scheme
- TX and RX PIO Control
- VM disable
- User configuration data 1
- User configuration data 15

The following subsections cover the PSkey settings for enabling LED indication.

**Note:** You will be asked to click “Set” every time you move to a new PSkey.

**PSkey settings for LED Indication**

**Coex Module: Coexistence Scheme**

This PSkey must be set to Disabled as shown in Figure 28.

**TX and RX PIO Control**

This PSkey must be set to PIO does not drive tx+rx, internal amp ramps power, as shown in Figure 29.
Figure 29: TX and RX PIO Control PSkey setting

**VM Disable**

This PSkey must be set to *False* as shown in Figure 30.

Figure 30: VM Disable PSkey setting

**User Configuration Data 1**

This PSkey must be set to *0001* as shown in Figure 31.

Figure 31: User Configuration Data 1 PSkey setting

**User Configuration Data 15**

This PSkey defines the blinking pattern for each of 6 radio states. For each state, there are 4 Usages. Therefore there are 24 parameters for this PSkey. Radio states are enumerated in Table 3, and LED state values are listed in Table 4.

Table 3: PSkey 15 and radio states

<table>
<thead>
<tr>
<th>PSkey</th>
<th>Word Position</th>
<th>Radio State</th>
<th>State Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSKEY_USR15</td>
<td>0-3</td>
<td>Idle</td>
<td>Radio has no active connections, is not discoverable, and is not attempting to establish connections</td>
</tr>
<tr>
<td></td>
<td>4-7</td>
<td>Inquiry</td>
<td>Radio is performing an inquiry operation</td>
</tr>
<tr>
<td></td>
<td>8-11</td>
<td>Page</td>
<td>Radio is performing a Page operation</td>
</tr>
<tr>
<td></td>
<td>12-15</td>
<td>Scan</td>
<td>Radio is performing a scan operation</td>
</tr>
<tr>
<td></td>
<td>16-19</td>
<td>Connected</td>
<td>Radio is connected to another device</td>
</tr>
<tr>
<td></td>
<td>20-23</td>
<td>Data Flowing</td>
<td>Radio is transmitting Data to a device</td>
</tr>
</tbody>
</table>
Table 4: Radio State values and descriptions

<table>
<thead>
<tr>
<th>Radio State Word Position</th>
<th>Usage</th>
<th>Units</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Pulse On Duration. Sets how long the LED should be lit per pulse</td>
<td>128ms</td>
<td>0x0000</td>
<td>Set to 0x0000 when using the Pulse Off Duration word to set the status indicator to continuously inactive while in this state.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0x0001 – 0x000E</td>
<td>Number of 128ms intervals the status indicator should be active for each pulse while in this state.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0x000F – 0xFFFF</td>
<td>Indicates the status indicator should be continuously active in this state.</td>
</tr>
<tr>
<td>1</td>
<td>Pulse Off Duration</td>
<td>128ms</td>
<td>0x0000</td>
<td>Set to 0x0000 when using the Pulse On Duration word to set the status indicator to continuously inactive while in this state.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0x0001 – 0x000E</td>
<td>Number of 128ms intervals the status indicator should be active for each pulse while in this radio state.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0x000F – 0xFFFF</td>
<td>Indicates the status indicator should be continuously active while in this state.</td>
</tr>
<tr>
<td>2</td>
<td>Pulse Count (1)</td>
<td>1 Pulse</td>
<td>0x0000 – 0x000F</td>
<td>Indicates the number of pulses that shall occur for each cycle while in this state.</td>
</tr>
<tr>
<td>3</td>
<td>Cycle Repeat Delay</td>
<td>1s</td>
<td>0x0000 – 0x000F</td>
<td>Indicates the time delay that shall occur before each pulse sequence is restarted while in this radio state.</td>
</tr>
</tbody>
</table>

(1) If the status indicator is set to be either continuously active or continuously inactive, then the value held in the Pulse Count word (i.e. Radio State Word Position 2) is ignored.

Figure 32: Setting LED status values (User Configuration Data 15) PSkey

Table 5 shows examples of LED flash patterns that may be set via the User Configuration Data 15 PSkey.

Table 5: LED Flash pattern examples

<table>
<thead>
<tr>
<th>Slow Flash 1.2s</th>
<th>Fast Flash</th>
<th>Slow Flash</th>
<th>Off &lt;reserved&gt;</th>
<th>Fast Pulse</th>
<th>Fast Flash</th>
</tr>
</thead>
<tbody>
<tr>
<td>000A 000A</td>
<td>0001 0001</td>
<td>0004 0004</td>
<td>0000 0000</td>
<td>0001 0001</td>
<td>0001 0001</td>
</tr>
<tr>
<td>0001 0001</td>
<td>0004 0004</td>
<td>0000 0000</td>
<td>0000 0000</td>
<td>0001 0001</td>
<td>0004 0001</td>
</tr>
</tbody>
</table>

Steady on

<table>
<thead>
<tr>
<th>Fast Flash</th>
<th>Slow Flash</th>
<th>Off &lt;reserved&gt;</th>
<th>Fast Pulse</th>
<th>Double Fast Flash</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001 0001</td>
<td>0004 0004</td>
<td>0000 0000</td>
<td>0001 0001</td>
<td>0001 0001</td>
</tr>
<tr>
<td>0000 0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
If USB transport is used to access the PSkeys, the device must be reverted to the Generic Bluetooth Adaptor driver before it can function as a normal Bluetooth device in Windows. To do so, complete the following steps:

1. In the Device Manager, right click **CSR BlueCore Bluetooth**, and select **Update Driver Software**.

   ![Figure 33: Update Driver Software dialog](image)

2. Click **Browse my computer for driver software**.

   ![Figure 34: Browse for driver software](image)

3. Click **Let me pick from a list of device drivers on my computer**.
4. Select *Generic Bluetooth Adapter* and click **Next**. Windows will reinstall the driver.

5. The BT800 is now recognized as “Generic Bluetooth Adapter.”
6. Expand “Bluetooth Radios” to confirm.

![Figure 38: Generic Bluetooth Adapter in Windows Device Manager](image)

Make new settings effective

If SPI transport is used, you must re-insert the development board to the PC to make new settings effective.

**Revision History**

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Description</th>
<th>Approved By</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>5 Aug 2013</td>
<td>Initial Release</td>
<td>Jonathan Kaye</td>
</tr>
</tbody>
</table>