CYPRESS®
Embedded in Tomorrow™

AN218191

WICED™ Quick Start Guide for BT CYW20706

Associated Part Family: CYW2070x
WICED™ Studio 4
To get the latest version of this application note, please visit www.cypress.com/AN218191

This document describes how to use WICED Studio to develop Bluetooth applications for CYW20706 devices.

Contents

1 Introduction

This document provides detailed instructions to set up the Cypress Wireless Internet Connectivity for Embedded Devices (WICED; pronounced "wick-ed") Studio Development System for Bluetooth Basic Rate (BR) and Low Energy (LE) devices based on the CYW20706 device.

WICED Studio supports application development using a WICED evaluation board (CYW92070xV3_EVAL). The development system is compatible with Windows, Mac OS X, and Linux operating systems. This document describes the software components included in the WICED Studio Development System and provides instructions for compiling WICED sample applications using the WICED Studio Integrated Development Environment (IDE).

Note: The CYW92070xV3_EVAL board features the CYW2070x device, which can be used as either a CYW20706 (fully embedded version) or as a CYW20707 (standalone BT controller version). The CYW20706 contains the embedded BT stack and supports running user-developed applications using WICED Studio 4. The CYW20707 operates as a standard BT controller and communicates with an external MCU via the standard HCI UART, but does not support user-developed applications. This document applies to **WICED Studio 4 and WICED Bluetooth CYW20706 modules.** See [2] for information on using the CYW20707 device as a standard HCI BT Controller.

Instructions in this document must be completed before the WICED evaluation board can be used with WICED Studio.

2 IoT Resources and Technical Support

Cypress provides a wealth of data at http://www.cypress.com/internet-things-iot to help you to select the right IoT device for your design, and quickly and effectively integrate the device into your design. Cypress provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates. Customers can acquire technical documentation and software from the Cypress Support Community website (http://community.cypress.com/).

3 WICED Studio Development System Overview

WICED Studio comprises a evaluation board, a Software Development Kit (SDK), and the Eclipse Integrated Development Environment (IDE).

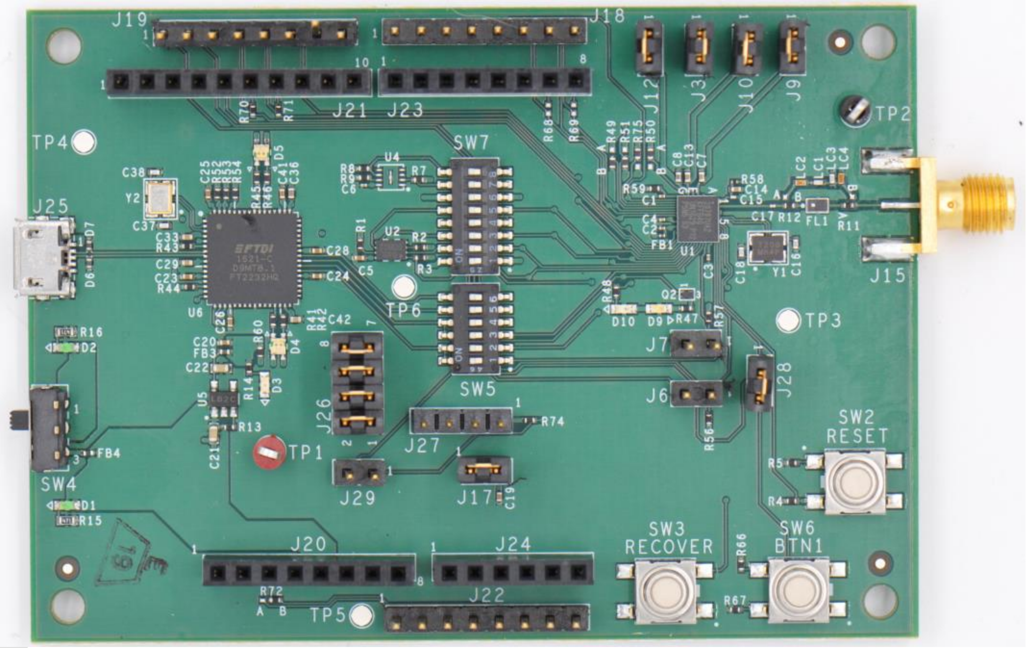
3.1 WICED Evaluation Board

The Cypress WICED evaluation board CYW92070xV3_EVAL incorporates a Cypress CYW20706 device and additional circuitry to enable application programming, debugging, and evaluation.

The CYW92070xV3_EVAL board can be used for feature evaluation, debugging, and developing Bluetooth applications for designs based on the CYW20706 device.

Figure 1 shows the front of the CYW92070xV3_EVAL board.

Figure 1. CYW92070xV3_EVAL WICED Evaluation Board



3.2 Software Development Kit

WICED Studio includes the following:

- The Bluetooth Basic Rate and Low Energy software stack, including a number of sample applications that demonstrate the use of API
- A generic profile-level API called the WICED BT API
- Drivers to access on-chip peripherals including UART, SPI, I^{2}C, ADC, PWM, etc.
- Reference applications for the devices with profiles defined by the Bluetooth SIG
- WICED BT API documentation
- Utilities to support development, testing, and mass production on Windows, Mac OS X, and Linux environments

3.3 Directory Structure

WICED Studio may support multiple different types of WICED modules depending on installed components. Table 1 provides an overview of the directory structure of WICED Studio for the CYW20706 device.

Table 1. WICED Studio Directory Structure

WICED Studio Directory	Directory Contents
Doc	API, reference documentation, and schematics
Drivers	USB drivers for the evaluation board
wiced_tools	Tools including download tool, and other utilities and scripts
43xxx_Wi-Fi\tools\ARM_GNU	Toolchain including compiler, linker, libraries, and headers
20706-A2_Bluetooth\apps	Sample applications
20706-A2_Bluetooth\build	Output files of built applications
20706-A2_Bluetooth\include	WICED API function prototypes and definitions
20706-A2_Bluetooth\libraries	Sources for various WICED interface libraries
20706-A2_Bluetooth\platforms	Configuration files and information for supported hardware platforms
20706-A2_Bluetooth\tools	Common utilities used by the IDE build processes
20706-A2_Bluetooth\WICED	WICED core components

3.4 Hardware and Software Requirement

- WICED Studio runs on 32- and 64-bit versions of Microsoft Windows, Mac OS X, and Linux
- WICED Studio is distributed as a bundle with the Eclipse IDE as executable installers for Windows, Mac OS X, and Linux
- The development computer requires a single USB port to connect to the WICED evaluation board

3.5 Development Process

To prepare and run an application, perform the following high-level steps:

1. Download and install WICED Studio 4 (see Install WICED Studio)
2. Connect the WICED Evaluation board (see Connect the WICED Evaluation Board)
3. Create and load an application (see Build and Load a Sample Application)

4 Setting up WICED Studio

Download WICED Studio 4 from the Cypress WICED Products website or Cypress Customer Support Portal.

4.1 Install WICED Studio

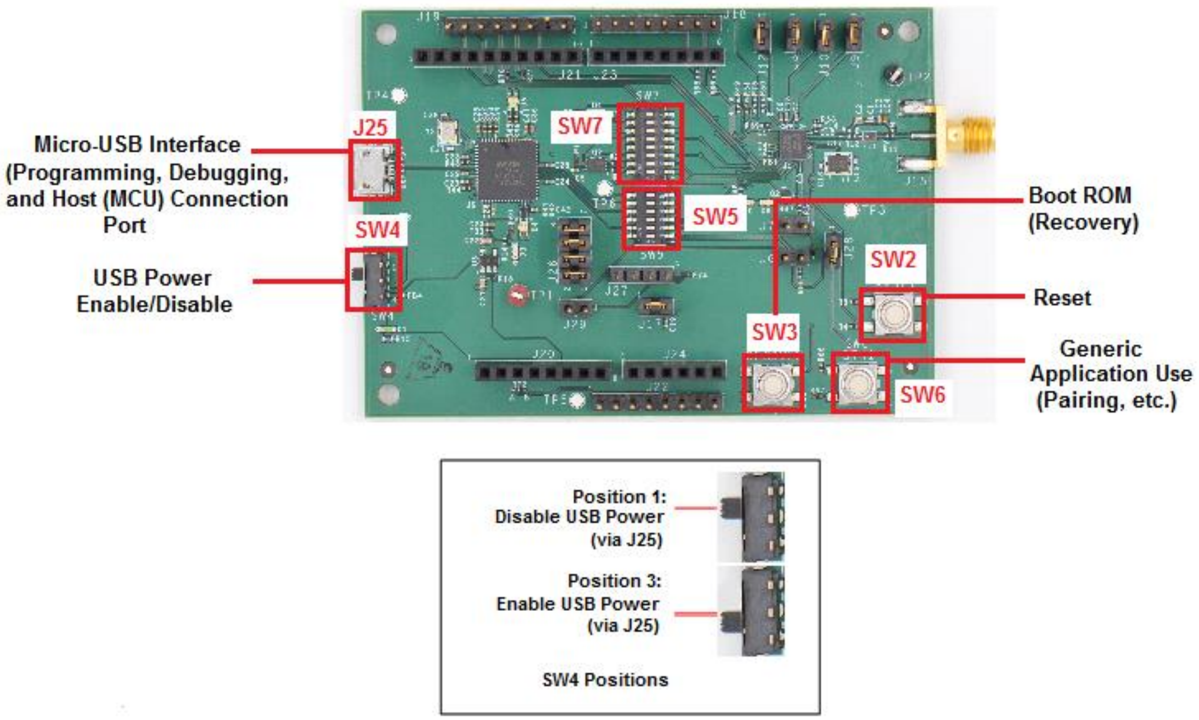
The WICED Studio distribution is provided as a self-installing executable files. Double-click the *WICED-Studio-4.0.x.x-IDE-Installer.exe* file. Follow the prompts to choose any available customizations, for example destination folders for the Eclipse IDE and the WICED Studio SDK files.

After installation has completed, start WICED Studio IDE by using the WICED Studio desktop icon.

4.2 Connect the WICED Evaluation Board

Figure 2 shows the CYW92070xV3_EVAL WICED evaluation board. The figure shows call-outs to the ports, switches, and switch positions relevant to this document.

Figure 2. CYW92070xV3_EVAL Evaluation Board Settings


J25
Micro-USB Interface (Programming, Debugging, and Host (MCU) Connection Port
SW4
USB Power Enable/Disable
SW7
SW5
Boot ROM (Recovery)
SW2
Reset
SW3
SW6
Generic Application Use (Pairing, etc.)
Position 1: Disable USB Power (via J25)
Position 3: Enable USB Power (via J25)
SW4 Positions

The Micro-USB connector (J25) supports UART connections and provides +5 V power to the board if SW4 is in position 3.

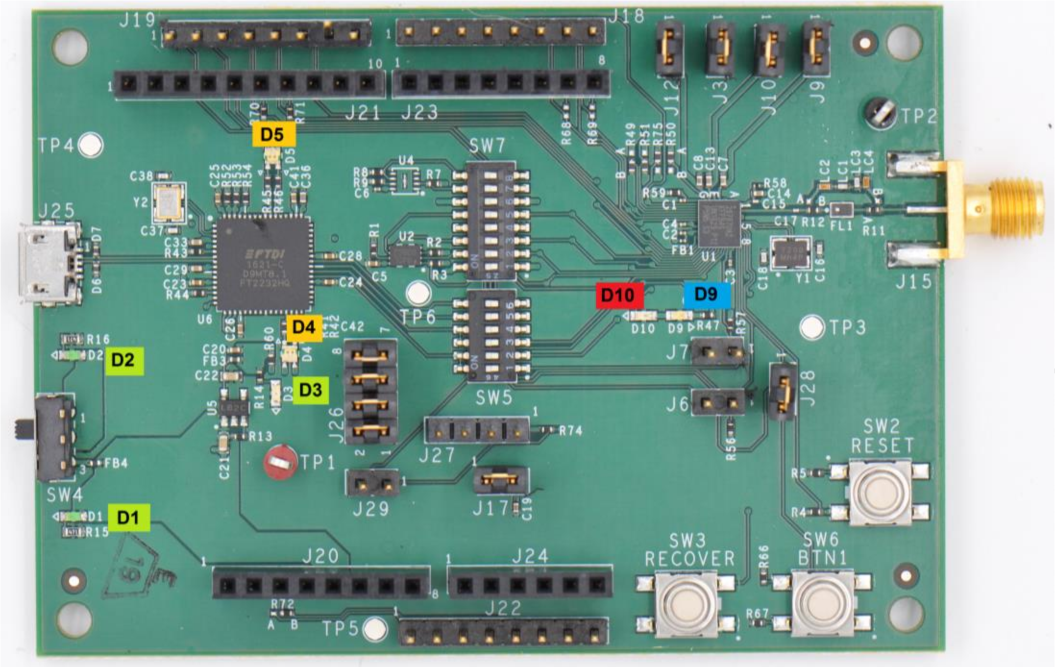
Perform the following steps before verifying the driver installation:

1. Set the six DIP switches of the **SW5** mini-switch to the OFF position so that the peripheral UART is selected. See [2] for complete DIP switch setting information.
2. Set the **SW7** switches 1-6 to the ON position and **SW7** switches **7** and **8** to the OFF position to enable on-board serial flash memory. See [2] for complete DIP switch setting information.
3. Set **SW4** to position 3 (see Figure 2).
4. Connect J25 of the WICED evaluation board to the development PC with a USB cable. The USB UART driver will load automatically.

The LEDs called out in Figure 3 serve the following purposes:

- D1 (green) indicates that SW4 is in position 3 (that is, USB power is enabled).
- D2 (green) indicates that an external PC is connected to the Micro-USB interface (J25).
- D3 (green) indicates that 3.3 V power is ON.
- D4 (variable color, but orange during a download) indicates HCI UART activity.
- D5 (variable color) indicates peripheral UART activity.
- D9 (blue) and D10 (red) are generic LEDs controlled by GPIOs.

Figure 3. LED Indicators



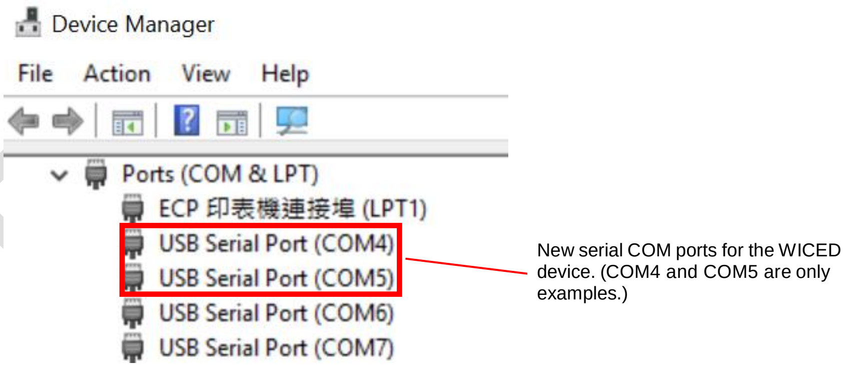
4.3 Verify Driver Installation

To verify that the driver installation is complete:

1. On a Windows system, open **Device Manager** (right-click **My Computer**, select **Properties**, and then select **Device Manager**).
2. In the **Device Manager** window, verify that two new USB serial COM ports are listed under **Ports (COM & LPT)**.

   Note: In Figure 4, Device Manager identifies the new (WICED evaluation board) USB serial COM ports as COM4 and COM5. Assigned port numbers vary among systems.

Figure 4. Device Manager COM Ports



Ports (COM & LPT)
- ECP 印表機連接埠 (LPT1)
- USB Serial Port (COM4)
- USB Serial Port (COM5)
- USB Serial Port (COM6)
- USB Serial Port (COM7)

New serial COM ports for the WICED device. (COM4 and COM5 are only examples.)

Note: If an error occurs during driver installation, download new drivers from Windows Update. Verify you have an Internet connection, disconnect and then reconnect the board and wait for the drivers to automatically install. If the new WICED evaluation board serial COM ports do not appear in Device Manager after reinstalling drivers via Windows Update, then the drivers can be manually installed from the Drivers folder of the WICED Studio installation. If the error persists, check all switch settings (see Connect the WICED Evaluation Board) on the board and/or replace the USB cable.

5 Using the WICED Studio IDE

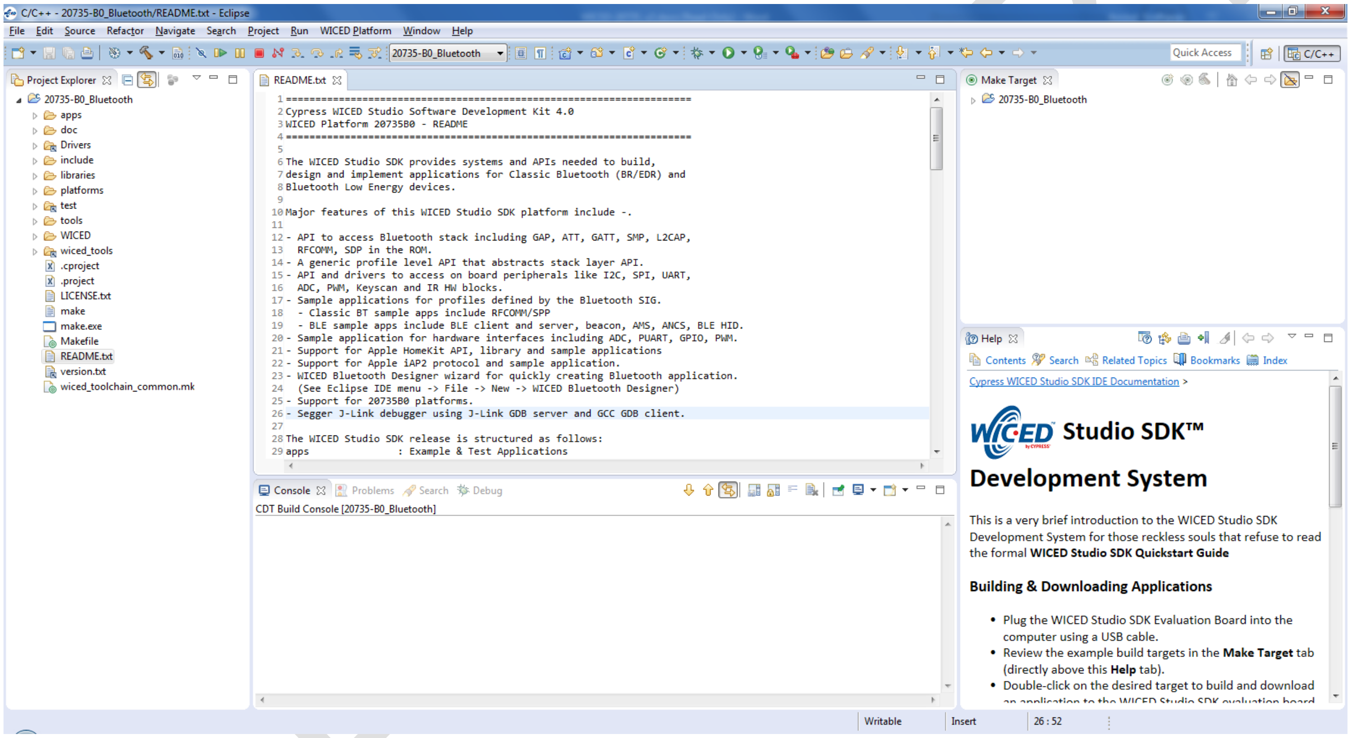
This section describes how to:

- Use the WICED Studio IDE to create application build targets for the WICED evaluation board
- Download applications to the board
- Verify that the application running on the board is working correctly using a Windows 8.x PC with Bluetooth capability.

5.1 Build and Load a Sample Application

Start the IDE by selecting **START** > **All Programs** > **Cypress** > **WICED-Studio** or double-click the **WICED-Studio** icon on desktop. The WICED Studio IDE looks similar to the screenshot shown in Figure 5.

Figure 5. WICED Studio IDE



```
==============================================================
Cypress WICED Studio Software Development Kit 4.0
WICED Platform 20735B0 - README
==============================================================

The WICED Studio SDK provides systems and APIs needed to build,
design and implement applications for Classic Bluetooth (BR/EDR) and
Bluetooth Low Energy devices.

Major features of this WICED Studio SDK platform include -.

- API to access Bluetooth stack including GAP, ATT, GATT, SMP, L2CAP,
  RFCOMM, SDP in the ROM.
- A generic profile level API that abstracts stack layer API.
- API and drivers to access on board peripherals like I2C, SPI, UART,
  ADC, PWM, Keyscan and IR HW blocks.
- Sample applications for profiles defined by the Bluetooth SIG.
  - Classic BT sample apps include RFCOMM/SPP
  - BLE sample apps include BLE client and server, beacon, AMS, ANCS, BLE HID.
- Sample application for hardware interfaces including ADC, PUART, GPIO, PWM.
- Support for Apple HomeKit API, library and sample applications
- Support for Apple iAP2 protocol and sample application.
- WICED Bluetooth Designer wizard for quickly creating Bluetooth application.
  (See Eclipse IDE menu -> File -> New -> WICED Bluetooth Designer)
- Support for 20735B0 platforms.
- Segger J-Link debugger using J-Link GDB server and GCC GDB client.

The WICED Studio SDK release is structured as follows:
apps                : Example & Test Applications
```

CDT Build Console [20735-B0_Bluetooth]

Cypress WICED Studio SDK IDE Documentation >

WICED Studio SDK™

Development System

This is a very brief introduction to the WICED Studio SDK Development System for those reckless souls that refuse to read the formal **WICED Studio SDK Quickstart Guide**

Building & Downloading Applications

- Plug the WICED Studio SDK Evaluation Board into the computer using a USB cable.
- Review the example build targets in the **Make Target** tab (directly above this **Help** tab).
- Double-click on the desired target to build and download an application to the WICED Studio SDK evaluation board

The **Help** pane in the lower-right corner of the IDE (see Figure 5) describes how to build and download the sample applications shown in the **Make Target** pane, which is located above the Help pane, and describes how to create new applications and targets based on the samples. The Make Target pane contains a build targets that is preconfigured for a sample application that run on CYW92070xV3_EVAL evaluation boards.

The example below shows how to build and run the sample application, which can then connect the WICED evaluation board to a Windows 8.x host PC. A sample host application is provided as part of WICED Studio under peerapps in the application folder.

1. Connect the evaluation board to the PC.
2. Verify that the UART port is present after WICED evaluation board is connected to the PC (see Verify Driver Installation)
3. Double-click the **hello_sensor-BCM920706_P49 download** target start the build. The IDE console pane (bottom center of the IDE window) will display the build and download progress.

The build output looks similar to the following:

```
22:24:45 **** Build of configuration Release for project 20706-A2_Bluetooth ****

"C:\\Users\\arvinds\\Documents\\WICED\\WICED-Studio-4.0\\20706-A2_Bluetooth\\make.exe"
hello_sensor-BCM920706_P49 download
Compiling spar_setup.c
Compiling hello_sensor.c
Compiling wiced_bt_cfg.c
Compiling lib_installer.c
Linking target ELF
OK, made elf.
..\..\..\43xxx_Wi-Fi\tools\ARM_GNU\bin\Win32\arm-none-eabi-objdump: section '.aon'
mentioned in a -j option, but not found in any input file
Call to hello_sensor_spar_crt_setup @ 0021844d
OK, made C:/Users/arvinds/Documents/WICED/WICED-Studio-4.0/20706-A2_Bluetooth/Wiced-
BT/spar/../../build/hello_sensor-BCM920706_P49-rom-ram-Wiced-release/A_20703A2-
hello_sensor-rom-ram-spar.cgs. MD5 sum is:
806c5323f8b67625baf701f62d71e7c9 *../../build/hello_sensor-BCM920706_P49-rom-ram-Wiced-
release/A_20703A2-hello_sensor-rom-ram-spar.cgs

--------------------------------------------------------------------------------
Patch code starts at              0x00270400 (RAM address)
Patch code ends at                0x00271C2C (RAM address)
Patch RW/ZI size                  2140 bytes
Application starts at             0x002173FC (RAM address)
Application ends at               0x00218449 (RAM address)

Patch code size                         6188 bytes
Application RAM footprint               4173 bytes
                                      ------
Total RAM footprint                     6313 bytes (6.2kiB)
--------------------------------------------------------------------------------


Converting CGS to HEX...
Conversion complete

Creating OTA images...
Conversion complete
OTA image footprint in NV is 21935 bytes

Detecting device...
Device found

Downloading application...
Download complete

Application running

09:11:40 Build Finished (took 1m:46s.379ms)
```

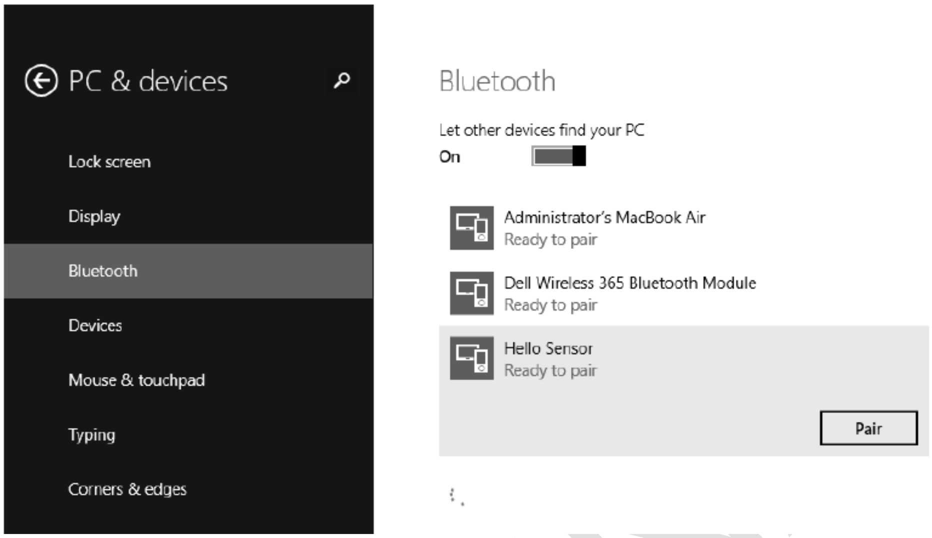
Note: The warning ‘`section '.aon' mentioned in a -j option, but not found in any input file`’ above is not critical; this is only an indication that the application did not use any data in retention RAM.

Note: If the download fails, see Appendix D: Recovering a Corrupted Board for instructions on how to recover a possibly corrupted serial flash.

4. Because the sample target includes the ***download*** option, the tool will download the firmware to evaluation board automatically when the build is complete.
5. After finishing the download, on the Windows 8.x PC, open the **Settings** charm (move the mouse to the lower or the upper right corner of the screen, then up or down, and click **Settings.**

6. Click **Change PC Settings** and select **Bluetooth** tab (see Figure 6).
7. Click **Add a device** and wait while the PC searches for devices in range.
8. Select **Hello Sensor** device, click **Pair**, and wait for the device connection to complete.

Figure 6. Windows Bluetooth Pairing Screen



Let other devices find your PC
On

Administrator's MacBook Air
Ready to pair

Dell Wireless 365 Bluetooth Module
Ready to pair

Hello Sensor
Ready to pair

5.2 The Hello Client PC Application

The Hello Client (hello_client) PC application is provided with WICED Studio to complement the Hello Sensor (hello_sensor) tag application. It is located here:

<WICED-Studio>\20706-A2_Bluetooth\Apps\hello_sensor\peerapps\Windows\HelloClient\Release

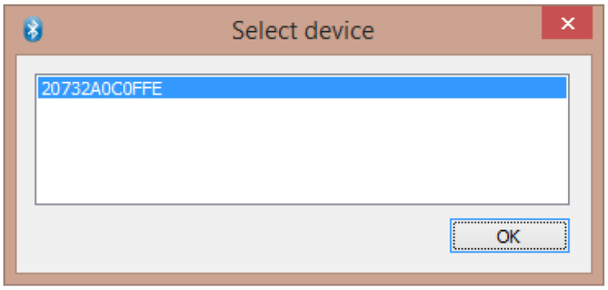
The application is provided as full source code, along with an executable binary that runs on Windows 32-bit and 64-bit machines.

1. Run *HelloClient.exe*. A Hello Client Select device window similar to that shown in Figure 7 is displayed. The window shows a list of Bluetooth device addresses for Hello Sensors paired to the PC.
2. The Bluetooth device address programmed in the tag is located in:

   <WICED-Studio>\20702-A2_Bluetooth\platforms\BCM920706_P49\20706_SFLASH.btp

   To change the Bluetooth device address, open the file and modify the **DLConfigBD_ADDRBase** variable (or you may set **BT_DEVICE_ADDRESS=xxxxxxxxxxxx** where xxxxxxxxxxxx are the 12 hex nibbles of the device address).
3. Select the correct device (if it is not already selected) and click **OK** to initiate a connection to the tag. The connection process may take 5–10 seconds.

Figure 7. Hello Client Select Device Window



20732A0C0FFE

5.3 Testing the Hello Sensor Application

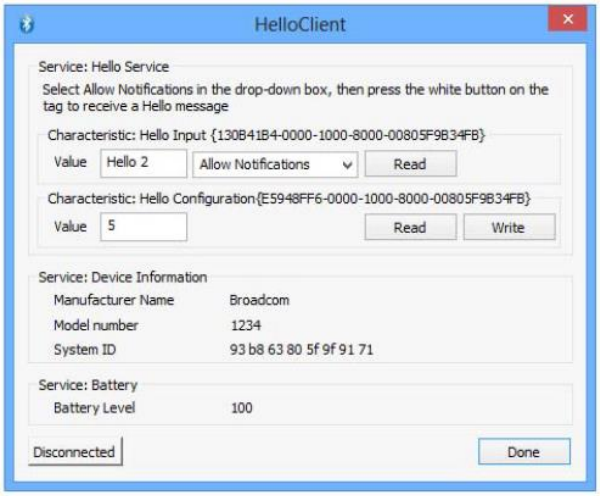
The Hello Sensor application provides paired devices the following information:

- A Hello Service (a proprietary service) with two proprietary characteristics:
  - The value of the Hello Input read-only characteristic may be retrieved using one of the following methods:
    - Manually, by using a mouse to click the Read button on the HelloClient PC application.
    - Automatically, by pressing the **Application** button **SW6** (see Figure 3) on the WICED evaluation board (the **Allow Notifications** drop-down must be selected to allow automatic notifications).
  - The Hello Configuration read-write characteristic is used to configure the number of times a LED on the tag blinks (see Figure 3) when the **Application** button is pressed.
- A Device Information Service that provides information including:
  - Manufacturer Name
  - Model Number
  - System ID
- A Battery Service that provides a battery-level indication

To test the application with a WICED evaluation board, follow the instructions provided for each of the Hello Service characteristics (see Figure 8).

- Hello Input Characteristic
  - Select **Allow Notifications** in the combo box.
  - Press the **Application** button **SW6** on the WICED board (see Figure 2). The Hello X message is displayed in the Value field. Each time the button is pressed, the message number increments.
- Hello Configuration Characteristic
  - Change the value for the Hello Configuration to '5' and then click **Write**.
  - Press the **Application** button **SW6** on the WICED board (see Figure 2). The LED on the tag blinks five times.

Figure 8. Hello Client Services Window



HelloClient

Service: Hello Service

Select Allow Notifications in the drop-down box, then press the white button on the tag to receive a Hello message

Characteristic: Hello Input {130B41B4-0000-1000-8000-00805F9B34FB}

Value: Hello 2 — Allow Notifications

Characteristic: Hello Configuration{E5948FF6-0000-1000-8000-00805F9B34FB}

Value: 5

Service: Device Information

Manufacturer Name: Broadcom

Model number: 1234

System ID: 93 b8 63 80 5f 9f 91 71

Service: Battery

Battery Level: 100

Disconnected

5.4 Viewing Traces

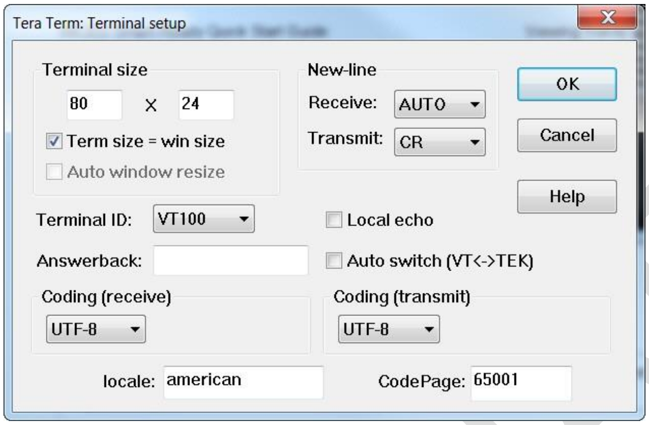
1. Compile and download an application to the WICED evaluation board as described in Build and Load a Sample Application.
2. Wait for the "Build Finished" message to display in the console window.
3. Use a terminal emulation program (see View Traces Using a Terminal Emulation Program).

5.4.1 View Traces Using a Terminal Emulation Program

To view traces with a terminal emulation program (such as Tera Term):

1. Start the terminal emulation program.
2. Set the **Terminal ID** to **VT100** and New-Line **Receive** to **AUTO**.

Figure 9. Tera Term Settings



Tera Term: Terminal setup

Terminal size: 80 X 24

☑ Term size = win size

☐ Auto window resize

New-line

Receive: AUTO

Transmit: CR

Terminal ID: VT100

☐ Local echo

Answerback:

☐ Auto switch (VT<->TEK)

Coding (receive): UTF-8

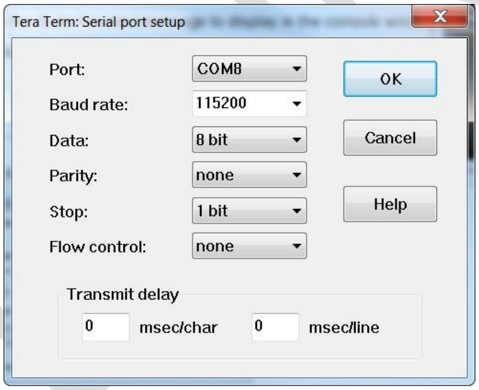
Coding (transmit): UTF-8

locale: american

CodePage: 65001

3. In the terminal emulator, initiate a connection with the following serial port settings:

Figure 10. Serial Port Settings



Tera Term: Serial port setup

Port: COM8

Baud rate: 115200

Data: 8 bit

Parity: none

Stop: 1 bit

Flow control: none

Transmit delay

0 msec/char 0 msec/line

4. Press the **Reset** button (see Figure 2) on the WICED evaluation board to view the application start-up messages.

5.5 What's NEXT?

Now that you have a basic understanding of how to compile and download a WICED Studio application, we recommend building and running the example applications provided in the WICED Studio applications directory.

The header of the main source file of every application provides additional information on the features demonstrated by the application and the usage model.

"IDE Hints & Tips" contains hints and tips about navigating the WICED Studio code base.

We hope you enjoy using the WICED Studio Development System!

-- The WICED Development Team

6 References

[1] Bluetooth SoC for Embedded Wireless Devices: CYW92070xV3_EVAL Schematic

[2] AN216535 - CYW92070xV3_EVAL Evaluation Board Hardware User Manual

Appendix A: Eclipse IDE Hints & Tips

A.1 Hints

1. The **Help** tab (and any other tab) may be click-dragged to any window pane to customize the IDE layout.
2. To revert to the C/C++ perspective (rather than the Debug perspective for example), click the C/C++ icon in the top-right corner of the window.

A.2 Shortcuts

A useful cheat-sheet outlining shortcuts for the WICED Studio IDE (Eclipse) is included online at:

http://www.cheat-sheets.org/saved-copy/eclipseCDT8.0-cheatsheet.pdf

Particularly useful keystrokes are listed below:

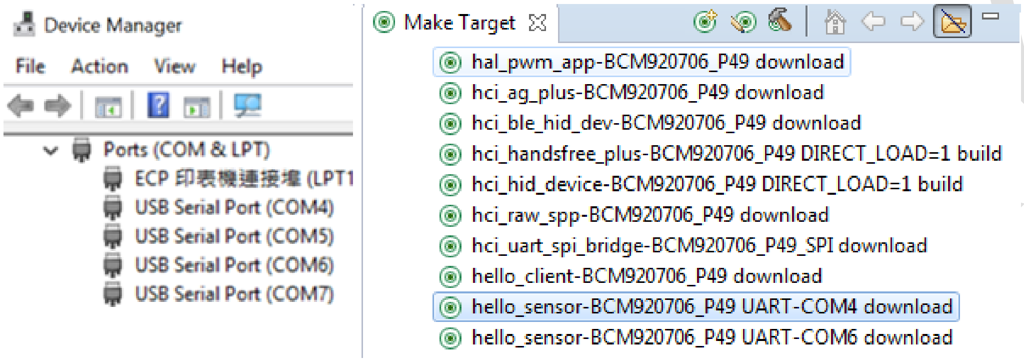
- General search: to search the WICED-Studio tree for a variable:
    1. **Click** the root WICED-Studio folder in the Project Explorer pane.
    2. Press **CTRL-H** (on Windows).
    3. In the File Search tab, enter the variable name (regular expressions work too).
    4. Click **Search**.
- Search for a *C* source element (variable, function, enum, etc.).
    1. Open a *C* source file, for example: <20706-A2_Bluetooth>/apps/hello_sensor/hello_sensor.c.
    2. Press **CTRL-SHIFT-T**.
    3. Start typing an element, for example, `BTM_BLE_ADVERT_`.
    4. Suggestions appear in the pop-up window.
- Press **ALT-Left** (arrow) and **ALT-Right** (arrow) to navigate between open files.

Appendix B: Multiple WICED Evaluation Boards

Multiple boards can be programmed from a single computer to run the same or different applications. To use the feature, edit the make target for the required application to add the `UART=COMx` parameters.

Figure 11 shows two WICED evaluation boards connected to a PC and appropriate targets to build and download the hello sensor application.

Figure 11. Configuration for Two WICED Evaluation Boards



Ports (COM & LPT)
ECP 印表機連接埠 (LPT1
USB Serial Port (COM4)
USB Serial Port (COM5)
USB Serial Port (COM6)
USB Serial Port (COM7)

hal_pwm_app-BCM920706_P49 download
hci_ag_plus-BCM920706_P49 download
hci_ble_hid_dev-BCM920706_P49 download
hci_handsfree_plus-BCM920706_P49 DIRECT_LOAD=1 build
hci_hid_device-BCM920706_P49 DIRECT_LOAD=1 build
hci_raw_spp-BCM920706_P49 download
hci_uart_spi_bridge-BCM920706_P49_SPI download
hello_client-BCM920706_P49 download
hello_sensor-BCM920706_P49 UART-COM4 download
hello_sensor-BCM920706_P49 UART-COM6 download

Appendix C: Connecting to Linux Platforms

An additional step is required when connecting a WICED board to a computer running Linux. On common Linux distributions, the serial UART ports (usually `/dev/ttySx or /dev/ttyUSBx devices`) belong to the root user and to the dialout group. Standard users are not allowed to access these devices.

An easy way to allow the current user access to Linux's serial ports is by adding the user to the dialout group. This can be done using the following command:

```
$sudo usermod -a -G dialout $USER
```

Note: For this command to take effect, the user must log out and then log back in.

Appendix D: Recovering a Corrupted Board

The following steps describe how to recover a WICED evaluation board if the serial flash has been corrupted.

Note: See Figure 2 for port and switch references associated with the recovery steps provided in this section.

First, verify the following:

1. J25 of the WICED evaluation board is connected to the development PC.
2. The USB-UART driver is correctly installed (see Verify Driver Installation).
3. The SWx mini-switches are in their correct positions

After these verifications, use the following method to recover a corrupt serial flash:

1. Press and hold the Recovery button (SW3).
2. Press the Reset button (SW2).
3. Release SW3.

Document History

Document Title: AN218191 – WICED™ Quick Start Guide for BT CYW20706

Document Number: 002-18191

Revision	Submission Date	Description of Change
*B	01/12/17	Updates to template and alignments with other documents
*A	11/14/16	Updates for 20706/20707 distinctions and CYW92070xV3_EVAL board
**	10/18/16	Initial release (Preliminary)

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

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

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

CYPRESS
Embedded in Tomorrow™

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

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.