WICED Studio

WICED - BLE WiFi Introducer

Document Number. 002-19364 Rev. *C

WICED, WiFi Introducer, BLE, DCT

Cypress Semiconductor

198 Champion Court

San Jose, CA 95134-1709

www.cypress.com

Contents

About This Document

The BLE WiFi Introducer app shows an example interfacing the BLE and Wi-Fi components on a combo chip. It demonstrates GATT database initialization, DCT configuration, processing read/write requests from a BLE client, and sending data to the client. The BLE WiFi Introducer has two components, an app running on a WICED® device, and an app running on an iOS device.

Acronyms and Abbreviations

In most cases, acronyms and abbreviations are defined on first use.

For a comprehensive list of acronyms and other terms used in Cypress documents, go to www.cypress.com/glossary.

IoT Resources and Technical Support

Cypress provides a wealth of data at 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 (community.cypress.com/).

WICED App Functionality

- Reads Wi-Fi network configuration from DCT and attempts to connect to the network.
- Upon failure to join the network, BLE GATT server is started to advertise network connection characteristic values
- When the iOS app triggers the BLE GATT Characteristic write event, network credential values are received. The WICED device will attempt to connect to the network and send the result via a notification message to the iOS device.
- Pressing the user button on the WICED board wipes the Wi-Fi network credentials from the DCT and the system is rebooted.

iOS App Functionality

- Scans for a WICED-based device and reads advertising data.
- Connects to the WICED device and allows the user to enter network credentials for the currently connected network.
- Writes the network credentials to the corresponding BLE GATT Characteristic values.
- Indicates the success or failure state of the WICED device network connection and allows for re-entering credentials upon connection failure.

Terminology

DCT: Device Configuration Table – Persistent store of system information and application data

BLE: Bluetooth Low Energy

GATT: Generic Attribute BLE Profile

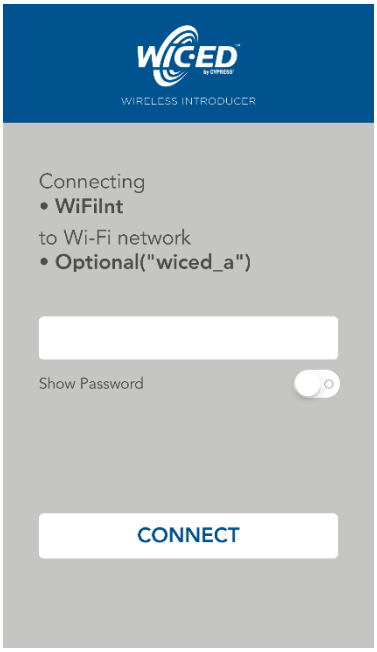
Step-By-Step Usage Guide

1. Use an OS X system to install the iOS Demo App (WiFiIntroducer) onto an iOS device. The files are located at *<WICED_SDK>/apps/demo/ble_Wi-Fi_introducer/peerapps/iOS*.
2. Compile and download the ble_wifi_introducer app onto a WICED device using the target `demo.ble_wifi_introducer-<platform> download run`" command.
3. Open a terminal to view the UART output. The app will try to connect to an access point (AP) with the default credentials and fail, then the BLE GATT server will start advertising. You should see an output similar to the following:

```
Starting WICED v3.7.0
Platform BCM94343W_AVN initialised
Started ThreadX v5.6
Initialising NetX_Duo v5.7_sp2
Creating Packet pools
WWD SDIO interface initialised
WLAN MAC Address : B0:38:29:3A:42:BE
WLAN Firmware    : wl0: May  5 2016 03:06:34 version 7.45.45.13 (r635786) FWID 01-3c7fac4b
WiFi Introducer Sensor Start
Joining : YOUR_AP_SSID
Failed to join : YOUR_AP_SSID
Joining : YOUR_AP_SSID
Failed to join : YOUR_AP_SSID
Joining : YOUR_AP_SSID
Failed to join : YOUR_AP_SSID
00:00:23.052248 GKI_create_task func=0x801bbad  id=1  name=BTU  stack=0x0  stackSize=6144
00:00:23.060248 GKI_create_task func=0x801d0c5  id=0  name=HCISU  stack=0x0  stackSize=4096

 wifi_introducer_bt_management_callback: 15
[WiFi Introducer] Local Identity Keys Request Event

 wifi_introducer_bt_management_callback: 0
apollo_config_management_callback:wiced_bt_dev_write_local_addr result = 0x0
> wifi_introducer_gatt_server_init

 wiced_bt_gatt_register: 0
wiced_bt_gatt_db_init 0
wiced_bt_ble_set_advertisement_data 0

 wifi_introducer_bt_management_callback: 17
Advertisement State Change: 3
wiced_bt_start_advertisements 0

 wifi_introducer_bt_management_callback: 14
[WiFi Introducer] Local Identity Keys Update type:1

 wifi_introducer_bt_management_callback: 14
[WiFi Introducer] Local Identity Keys Update type:3
```

4. On the iOS device, connect to the Wi-Fi access point to which the WICED device will also join. Open the WiFi Introducer application; you should be greeted with the following splash screen:

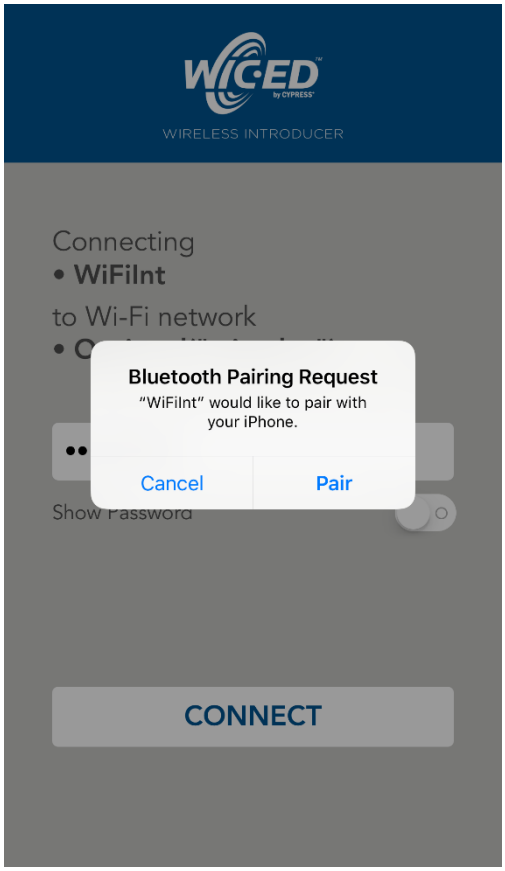
5. The app will scan until a nearby WICED device is detected. Bring the WICED device close to the iOS device and the following screen will be brought up:



Note: If the iOS app does not detect the WICED device, kill and reopen the iOS application.

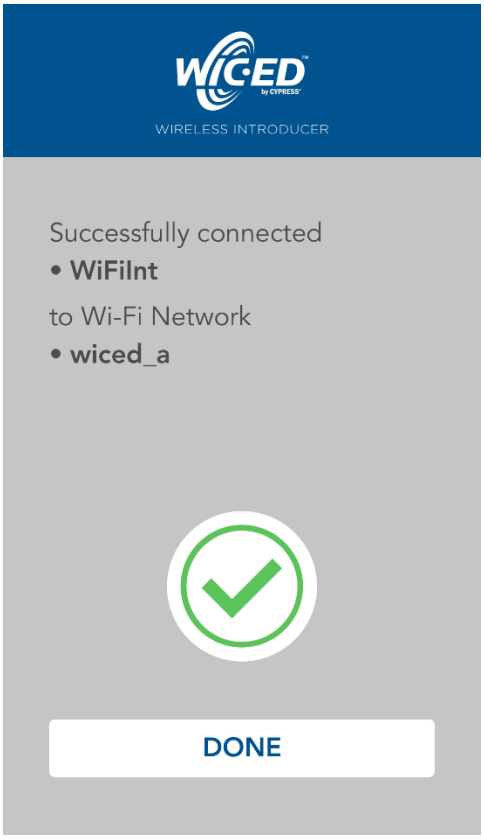
Note: The application uses iOS APIs to fetch the currently connected AP. From iOS12, the APIs used to access WiFi interface have been disabled for Apple developers, by default. To access these APIs, you must enroll in the Apple Developer program (ADP). To use these WiFi APIs in XCode, go to **Project Settings → Select your App Target**. Go to the **Capabilities** tab and enable the **Access WiFi Information** switch.

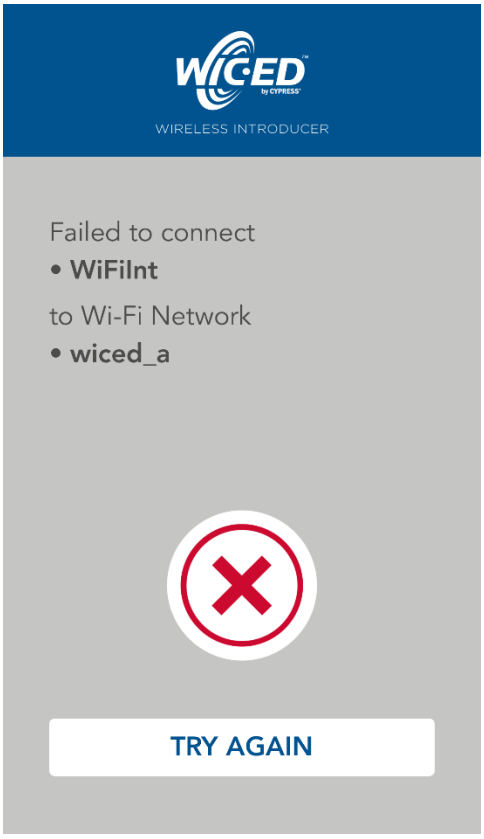
6. Enter the Wi-Fi AP passphrase and press connect. You will be prompted to pair the device. Select **Pair**; a secure LE connection will be made between the iOS device and the WICED device using the ECDH algorithm to generate and exchange secure keys.



Note: If the device has been previously paired, it may be necessary to remove the pairing before the app will run correctly. Do this by going to **Settings** > **Bluetooth**. Under **My Devices**, click the "**i**" icon next to WiFiInt and select **Forget This Device**.

7. The Wi-Fi network credentials will be sent to the WICED device and the connection result will be pushed back to the iOS device. If the connection fails, press **Try Again** to re-input the passphrase. Upon successful connection, the network credentials will be saved to the DCT.






```
 wifi_introducer_gatt_server_write_request_handler:wifi_introducer_char_nw_passphrase_value value: 123456789
SSID configured
Infra 94:10:3E:D2:CC:64 OFF 150.0   11   WPA2 AES PSK         wiced_a                                    PROBE
Joining : wiced_a


 wifi_introducer_bt_management_callback: 21
[WiFi Introducer] Unhandled Bluetooth Stack Callback event :33
wifi_introducer_gatt_server_request_handler. conn 2, type 1
read_hndlr conn_id:2 hdl:16 offset:0 len:7

 wifi_introducer_bt_management_callback: 21
[WiFi Introducer] Unhandled Bluetooth Stack Callback event :33
Failed to join : wiced_a
Joining : wiced_a
Failed to join : wiced_a
Joining : wiced_a
Failed to join : wiced_a
Join result 1007:
Join Failed!!!

 wifi_introducer_send_message: Client's Characteristic configuration:1
wifi_introducer_char_notify_value = 0
connection_down  conn_id:2 reason:19
```

```
 wifi_introducer_gatt_server_write_request_handler:wifi_introducer_char_nw_passphrase_value value: 12345678
SSID configured

 wifi_introducer_bt_management_callback: 21
[WiFi Introducer] Unhandled Bluetooth Stack Callback event :33
Infra 94:10:3E:D2:CC:64 OFF 150.0   11   WPA2 AES PSK         wiced_a                                    BEACON
Joining : wiced_a
connection_down  conn_id:2 reason:8

 wifi_introducer_bt_management_callback: 17
Advertisement State Change: 4
wiced_bt_start_advertisements 0

 wifi_introducer_bt_management_callback: 21
[WiFi Introducer] Unhandled Bluetooth Stack Callback event :33
Successfully joined : wiced_a
Obtaining IPv4 address via DHCP
DHCP CLIENT hostname WICED IP
IPv4 network ready IP: 192.168.2.7
Setting IPv6 link-local address
IPv6 network ready IP: FE80:0000:0000:0000:02A0:50FF:FE28:781C
Join Succeeded!!!

 wifi_introducer_send_message: Client's Characteristic configuration:1
wifi_introducer_char_notify_value = 1
```

8. When rebooting the WICED device after a successful Wi-Fi credential transfer, the DCT will have valid network credentials. If the network is successfully joined, the BLE GATT Server will not be started. You can reset the network settings in the DCT by pressing the user button on the WICED board. The device will reboot and the BLE GATT Server will start advertising again.

Document Revision History

Document Title: WICED - BLE WiFi Introducer

Document Number: 002-19364

Revision	ECN	Issue Date	Description of Change
**	–	10/13/2016	Initial release
*A	5686732	04/27/2017	Converted to Cypress template format
*B	6371162	10/31/2018	Updated for ModusToolbox
*C	6511739	03/15/2019	Updated instructions for iOS 12 and later

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 | PSoC 6 MCU

Cypress Developer Community

Community | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

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