WINC Driver Demonstrations

MPLAB Harmony Integrated Software Framework

WINC Driver Demonstrations

Contents

Introduction

These applications demonstrate how to use the MPLAB Harmony WINC Driver. This section describes the hardware requirements and procedures to build and execute the demonstration projects on Microchip development tools. Six demonstrations are provided:

- ap_scan– In this demonstration, the WINC Driver will scan for available BSSs and display the list

- power_save_mode_example– In this demonstration, the WINC's power save feature is demonstrated.

- wifi_winc_tcp_client – In this demonstration, the WINC Driver will connect to a BSS and begin a TCP client socket to a remote server service.

- wifi_provision_via_ble– In this demonstration, the WINC3400 will advertise over BLE. BLE connection is made to provision a WiFi connection.

- wifi_provisioning_via_softap – In this demonstration, the WINC Driver will advertise a SoftAP. The SoftAP connection is leveraged to provision a WiFi connection.

- wifi_provisioning_via_wps – In this demonstration, the WINC Driver will move into WPS mode – PIN or PBC. By activating the WPS mode in the AP concurrently, a WiFi connection is established.

- wifi_tcp_server_in_softap– In this demonstration, the WINC Driver will advertise a SoftAP. The SoftAP connection is used to demonstrate a TCP client-server operation.

- wifi_winc_sta_bypass– In this demonstration, the creation of a TCP/IP demo on the SAM E54 Xplained Pro board with the ATWINC1500/3400 XPRO board through Bypass/Ethernet mode is explained.

In case a custom application is created from scratch through the MHC, WINC 1500/3400(from Wireless v3.3.0) requires the following dependent repository versions to generate the plibs and supporting source code. These are also specified in the package.xml file in the root directory.

- "csp" version="v3.8.0"

- "core" version="v3.8.0"

- "net" version="v3.6.2_release"

- "bsp" version="v3.8.0"

All applications follow a simple flow in sequence to achieve the desired objective. For example, consider the Connect sequence shown below.
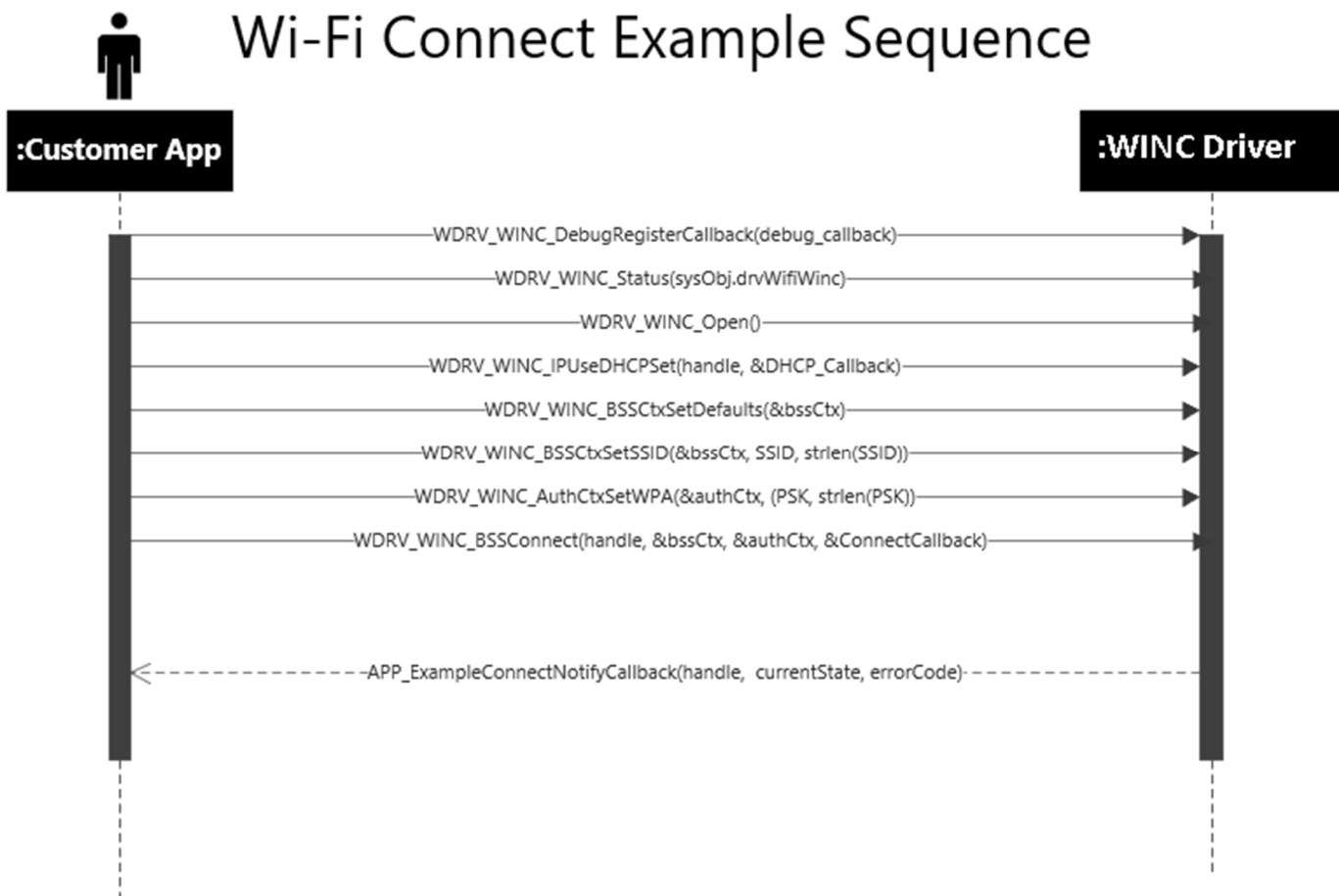
To know more about the MPLAB Harmony WINC Driver, configuring the WINC Driver and the APIs provided by the WINC Driver, refer to the WINC Driver Library documentation.

Microchip-Direct references/order codes:

- SAMD21: ATSAMD21-XPRO (homepage: https://www.microchip.com/Developmenttools/ProductDetails/ATSAMe54-XPRO)
- SAME54: ATSAME54-XPRO (homepage: https://www.microchip.com/Developmenttools/ProductDetails/ATSAMD21-XPRO)
- WINC1500: ATWINC1500-XPRO (homepage: https://www.microchip.com/Developmenttools/ProductDetails/ATWINC1500-XPRO)
- WINC3400: ATWINC3400-XPRO (homepage: https://www.microchip.com/Developmenttools/ProductDetails/ATWINC3400-XPRO)

MPAB X IDE Project Configurations

This table lists and describes the supported configurations of the demonstrations, which are located within firmware\src\config.

Project Configuration Name	BSP Used	Description
sam_d21_xpro_winc1500.X	SamD21	This demonstration runs on the SAMD21host and WINC1500 Daughter Board.
sam_d21_xpro_winc3400.X	SAMD21	This demonstration runs on the SAMD21host and WINC3400 Daughter Board.
sam_e54_xpro_winc1500.X	SAME54	This demonstration runs on the SAME54host and WINC1500 Daughter Board.

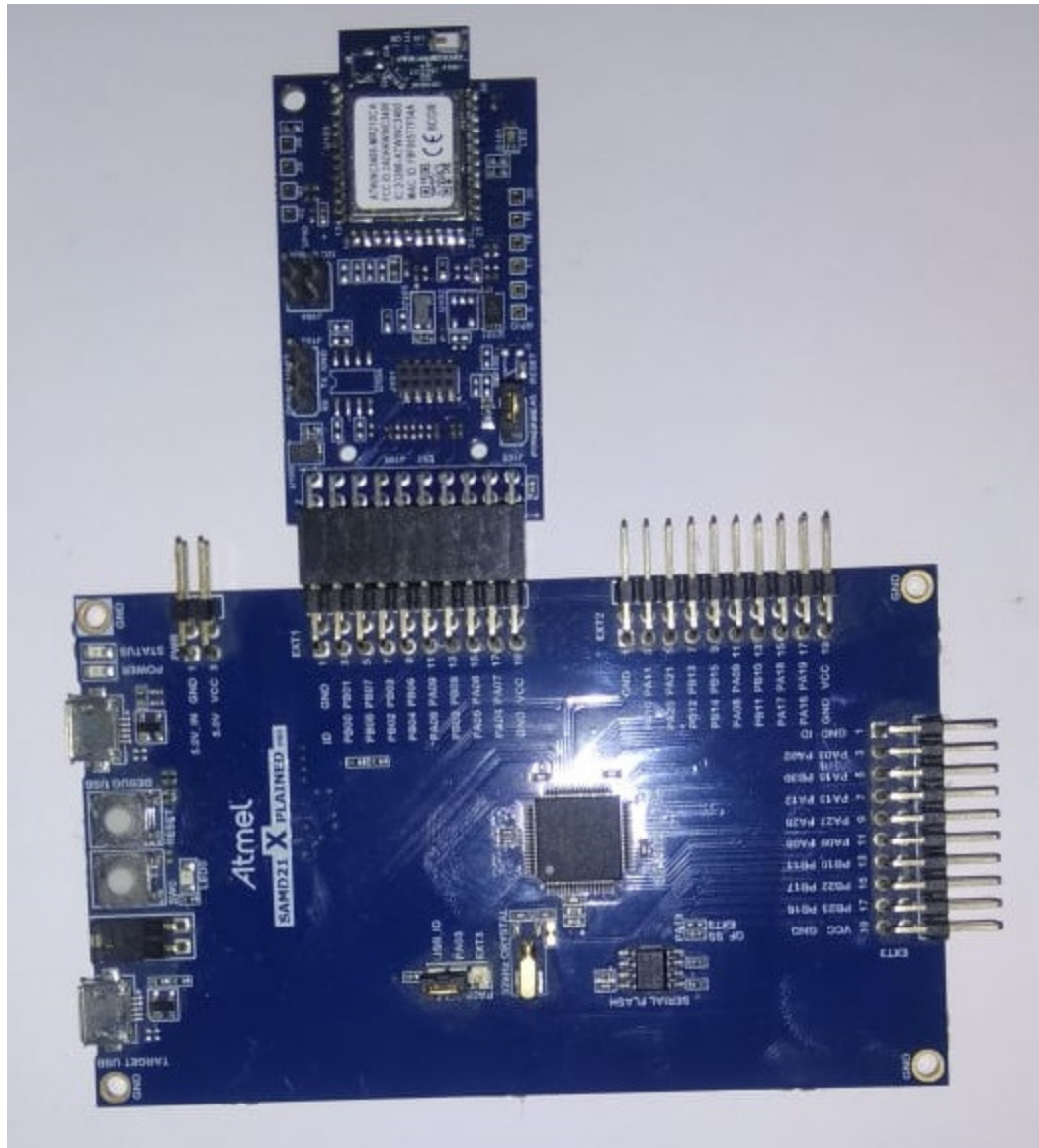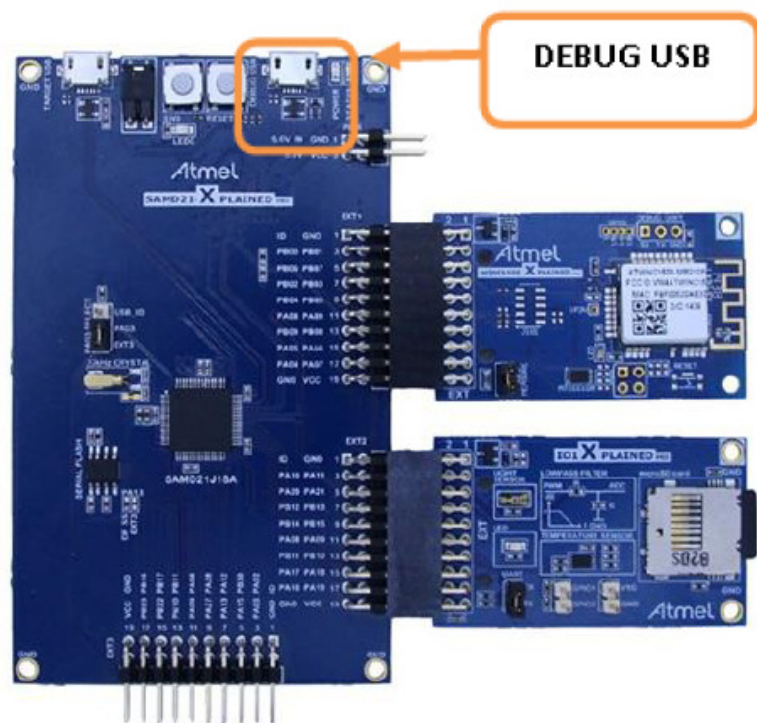| sam_e54_xpro_winc3400.X	SAME54	This demonstration runs on the SAME54host and WINC3400 Daughter Board.

Configuring the Hardware

The debug UART and USB ports can be used to read the output logs on a terminal software.

WINC3400 with SAMD21 shown below

DEBUG UART

DEBUG USB
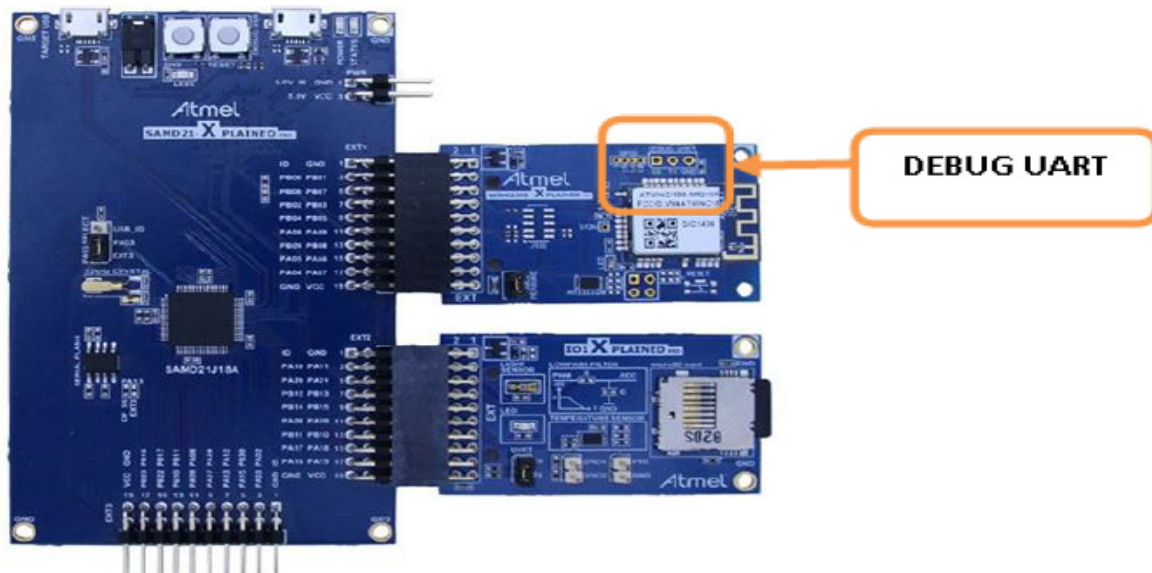
WINC1500 with SAME54 shown below
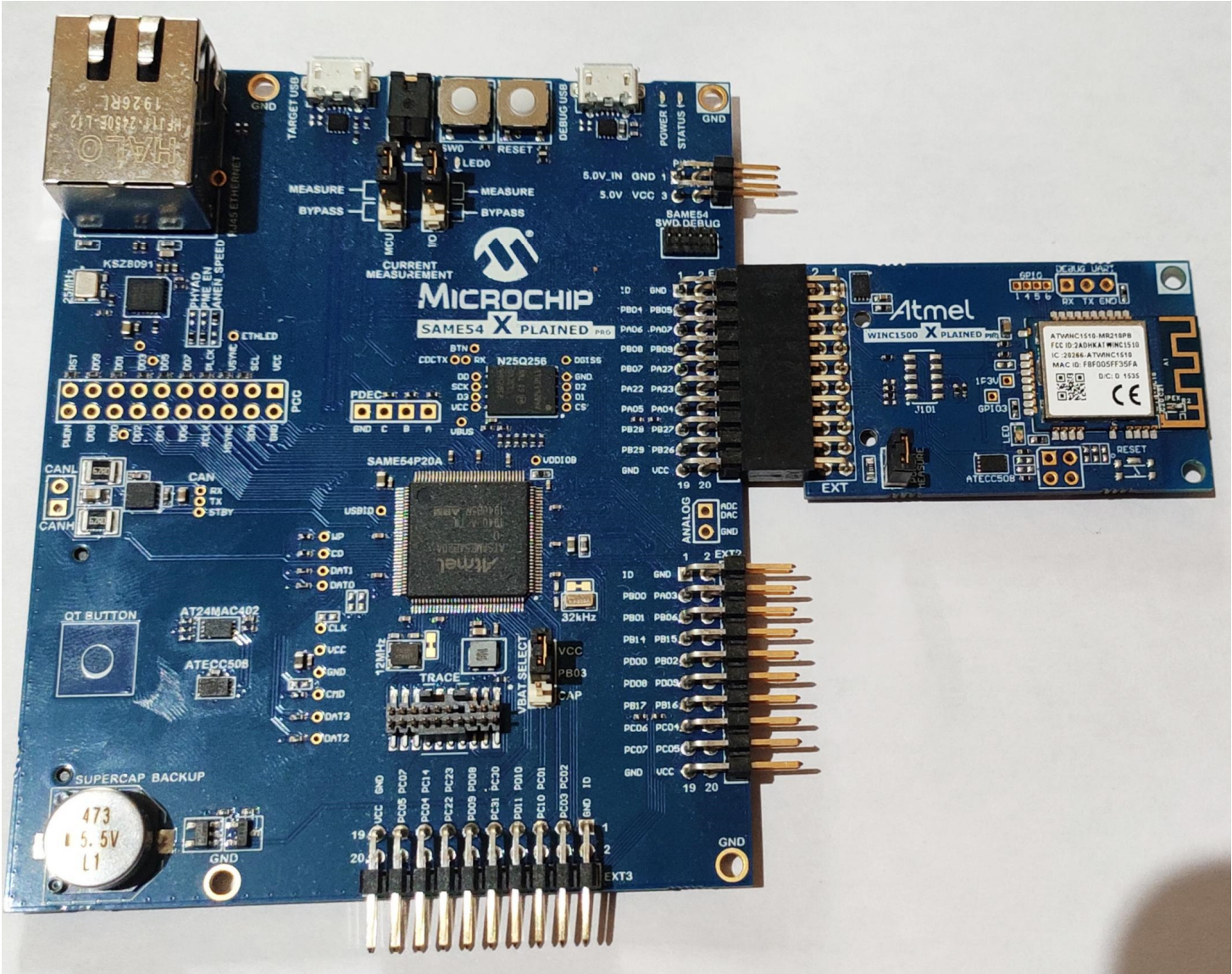
ap_scan

In this demonstration, the WINC Driver will scan for available BSSs and display the list.

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\ap_scan\firmware`

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir>\apps\ap_scan\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

```
WINC: Initializing...
Chip ID 1503a0
DriverVerInfo: 0x13301361


WINC1500 Firmware Data:
Firmware Ver: 19.6.1 SVN Rev 16761
Firmware Built at May 23 2018 Time 14:39:16
Firmware Min Driver Ver: 19.3.0
Driver Ver: 19.6.1
Driver Built at Jan 01 2019 Time 12:00:00



=========================
WINC BSS Scan Example
=========================
Scan complete, 4 BSS(s) found
BSS found: RSSI: -72 ExampleAP1
BSS found: RSSI: -54 ExampleAP2
BSS found: RSSI: -63 ExampleAP3
BSS found: RSSI: -63 ExampleAP4
```

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.
2. Open the project configuration when prompted.
3. Make necessary changes through the Project graph.
4. Select 'Generate Code' to update the MPLAB Harmony configuration.
5. Build and run the demonstration.

power_save_mode_example

In this demonstration, the WINC's power save feature is demonstrated. The Power save mode is demonstrated depending on the value configured in example_conf.h.

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\power_save_mode_example`.

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir>\apps\power_save_mode_example\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

In example_conf.h, configure the desired settings or power save.

#define MAIN_PS_SLEEP_MODE   WDRV_WINC_PS_MODE_MANUAL

/* WDRV_WINC_PS_MODE_MANUAL / WDRV_WINC_PS_MODE_AUTO_LOW_POWER / WDRV_WINC_PS_MODE_OFF */

/** Request sleep time for PowerSave manual mode */

#define MAIN_REQUEST_SLEEP_TIME   5000

```
WINC: Initializing...
Chip ID 1503a0
DriverVerInfo: 0x13301361

WINC1500 Firmware Data:
Firmware Ver: 19.6.1 SVN Rev 16761
Firmware Built at May 23 2018 Time 14:39:16
Firmware Min Driver Ver: 19.3.0
Driver Ver: 19.6.1
Driver Built at Jan 01 2019 Time 12:00:00



========================
WINC Power Save Mode Example
========================

WDRV_WINC_PS_MODE_MANUAL
.
.
.
.
.
```

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.
2. Open the project configuration when prompted.
3. Make necessary changes through the Project graph.
4. Select 'Generate Code' to update the MPLAB Harmony configuration.
5. Build and run the demonstration.

wifi_winc_tcp_client

In this demonstration, the WINC Driver will connect to a BSS and begin a TCP client socket to a remote server service.

This demonstration can be configured in the following ways.

- The BSS to connect to.

- The TCP server address and port to connect to.

- The receive buffer size.

- The message which should be sent as the first data once connected to the server.

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\tcp_client`

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir> \apps\tcp_client\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

```
WINC: Initializing...
Chip ID 1503a0
DriverVerInfo: 0x13301361

WINC1500 Firmware Data:
Firmware Ver: 19.6.1 SVN Rev 16761
Firmware Built at May 23 2018 Time 14:39:16
Firmware Min Driver Ver: 19.3.0
Driver Ver: 19.6.1
Driver Built at Jan 01 2019 Time 12:00:00



==========================
WINC TCP Client Example
==========================
Connected
IP address is 192.168.0.51
Socket 0 session ID = 1
Socket 0 connect success
Socket 0 send completed
Socket receive, size = 30 bytes [TCP message from WIN...]
```

The flow of control and data follows a similar approach as the connect example shown at the beginning of this document. Below sequence shows the use of a tcp socket by the customer application to send/receive data between the application and the WINC 1500/3400.

TCP Socket Example Sequence

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.
2. Open the project configuration when prompted.
3. Make necessary changes through the Project graph.
4. Select 'Generate Code' to update the MPLAB Harmony configuration.
5. Build and run the demonstration.

wifi_provision_via_ble

In this demonstration, the WINC3400 will advertise over BLE. BLE connection is made to provision a WiFi connection.

1. Build and run the demonstration.
2. Use Microchip Bluetooth App (MBD App) to connect and share details for provisioning over Bluetooth.

Building the Application

To build this project, you must open the WINC3400 winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>`

```
\apps\wifi_provision_via_ble.
```

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
`<Configuration>_winc1500/3400.X`	`<install-dir> \apps\wifi_provision_via_ble\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

```
WINC: Initializing...
Chip ID 1503a0
DriverVerInfo: 0x13301361


WINC3400 Firmware Data:
Firmware Ver: 1.2.2
Firmware Built at May 23 2018 Time 14:39:16
Firmware Min Driver Ver: 19.3.0
Driver Ver: 1.2.2
Driver Built at Jan 01 2019 Time 12:00:00



==============================
WINC3400 BLE WiFi Provisioning Example
==============================
BLE provisioning started
Scanning for BSSs…..
Scan complete, 5 BSS(s) found
BSS found: RSSI: -72 ExampleAP1
BSS found: RSSI: -54 ExampleAP2
BSS found: RSSI: -63 ExampleAP3
BSS found: RSSI: -63 ExampleAP4
Provisioning data received
Connecting to: ExampleAP1
```

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.
2. Open the project configuration when prompted.
3. Make necessary changes through the Project graph.
4. Select 'Generate Code' to update the MPLAB Harmony configuration.
5. Build and run the demonstration.

wifi_provisioning_via_softap

In this demonstration, the WINC Driver will advertise a SoftAP. The SoftAP connection is leveraged to provision a WiFi connection.

1. Build and run the demonstration.
2. Use Android-Provision-AP_v0.8 to connect and share details for provisioning over WiFi.

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\wifi_provisioning_via_softap`

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir>\apps\wifi_provisioning_via_softap\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

```
WINC: Initializing...

Chip ID 1503a0

DriverVerInfo: 0x13301361


WINC3400 Firmware Data:

Firmware Ver: 1.2.2

Firmware Built at May 23 2018 Time 14:39:16

Firmware Min Driver Ver: 19.3.0

Driver Ver: 1.2.2

Driver Built at Jan 01 2019 Time 12:00:00


===========================================

WINC WiFi Provisioning Soft AP Example

===========================================

 *

AP started, you can connect to WINC1500_PROVISION_AP

On the android device, connect to WINC1500_PROVISION_AP then run setting app

Bind on socket 0 successful, server_socket = 0

Listen on socket 0 successful

AP Mode: Station connected

AP Mode: Station IP address is 192.168.1.100

Connection from  192.168.1.100:47600

Disable to AP.

AP Mode: Station disconnected

STA mode: Station connected

STA Mode: Station IP address is 192.168.0.103
```

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.

2. Open the project configuration when prompted.

3. Make necessary changes through the Project graph.

4. Select 'Generate Code' to update the MPLAB Harmony configuration.

5. Build and run the demonstration.

wifi_provisioning_via_wps

In this demonstration, the WINC Driver will move into WPS mode – PIN or PBC. By activating the WPS mode in the AP concurrently, a WiFi connection is established.

1. Configure WPS Pin or PBC configuration in the example_conf.h.

2. Build and run the demonstration.

3. Use WPS PBC/PIN in AP to establish connect over WPS

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\wifi_provisioning_via_wps`.

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir>\apps\wifi_provisioning_via_wps\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

```
WINC: Initializing...
Chip ID 1503a0
DriverVerInfo: 0x13301361

WINC1500 Firmware Data:
Firmware Ver: 19.6.1 SVN Rev 16761
Firmware Built at May 23 2018 Time 14:39:16
Firmware Min Driver Ver: 19.3.0
Driver Ver: 19.6.1
Driver Built at Jan 01 2019 Time 12:00:00


==================================
Wi-Fi Provisioning via WPS Example
==================================
WPS Push Button Test
SW0 button pressed
Device is connecting using WPS Push Button option
[APP_ExampleWPSPushButtonDiscoveryCallback] In
SSID : xxxxx, authtyp : x pw : xxxxxxxx
Connect AP, SSID = xxxxx
```

```
STA mode: Station connected

STA Mode: Station IP address is xxx.xxx.xxx.xxx
```

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.

2. Open the project configuration when prompted.

3. Make necessary changes through the Project graph.

4. Select 'Generate Code' to update the MPLAB Harmony configuration.

5. Build and run the demonstration.

wifi_tcp_server_in_softap

In this demonstration, the WINC Driver will advertise a SoftAP. The SoftAP connection is leveraged to provision a WiFi connection.

1. Build and run the demonstration.

2. Connect your personal computer to the network named defined by macro "WLAN_SSID" in example_conf.h file

3. For creating a TCP connection, you can use any open source programs (e.g. packet sender or Tera Term).

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\wifi_tcp_server_in_softap`

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir> \apps\wifi_tcp_server_in_softap\firmware`

Running the Demonstration

Once the demonstration application has been compiled successfully for the intended configuration, program the firmware into the target device.

Upon execution, the application will transmit the following via the UART connection.

```
WINC: Initializing...
Chip ID 1503a0
DriverVerInfo: 0x13301361

WINC1500 Firmware Data:
Firmware Ver: 19.6.1 SVN Rev 16761
Firmware Built at May 23 2018 Time 14:39:16
Firmware Min Driver Ver: 19.3.0
Driver Ver: 19.6.1
Driver Built at Jan 01 2019 Time 12:00:00


==========================================
```

```
WINC WiFi TCP Server Soft AP Example

==========================================


AP started, you can connect to WINC1500_SOFT_AP

On the connected device, start a TCP client connection to 192.168.1.1 on port 6666


AP Mode: Station connected

AP Mode: Station IP address is 192.168.1.100

Bind on socket 0 successful, server_socket = 0

Listen on socket 0 successful

Connection from  192.168.1.100:55686

Receive on socket 1 successful

Client sent 13 bytes

Client sent Hello Server!

Sending a test message to client

Socket 1 send completed

TCP Server Test Complete!

Closing sockets
```

Configuring the Demonstration

If required, the example's configuration can be changed in the MHC such as Pin, Interrupt, debug configuration etc. and the above steps repeated to observe the changes in the application behaviour.

To configure these aspects of the demonstration.

1. Ensure the project is set as the main project and open the MPLAB Harmony Configurator.
2. Open the project configuration when prompted.
3. Make necessary changes through the Project graph.
4. Select 'Generate Code' to update the MPLAB Harmony configuration.
5. Build and run the demonstration.

wifi_winc_sta_bypass

In this demonstration, the creation of a TCP/IP demo on the SAM E54 Xplained Pro board with the ATWINC1500/3400 XPRO board through Bypass/Ethernet mode is explained.

Building the Application

To build this project, you must open the winc.X project in MPLAB X IDE, and then select the desired configuration.

The following tables list and describe the project and supported configurations. The parent folder for these files is `<install-dir>\apps\wifi_winc_sta_bypass`

MPAB X IDE Project

This table lists the name and location of the MPLAB X IDE project folder for the demonstration.

Project Name	Location
<Configuration>_winc1500/3400.X	`<install-dir> \apps\wifi_winc_sta_bypass\firmware`

Programming the Demonstration

Refer to the hardware user-guide to see the different programming options supported.

The SAME54 Xplained board can be programmed using the on-board debugger **EDBG**.

1. Connect a micro-USB cable from computer to the **DEBUG USB** port on the SAM E54 Xplained Ultra board.
2. Program the application by clicking on the **Make and Program Device Main Project**.
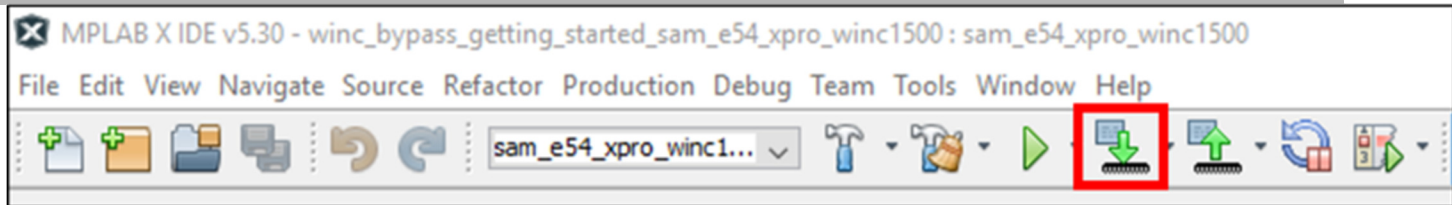
3. Make sure that the programming is successful.

Test the Application

The testing of the application is done using different console commands.

There will be a virtual COM port enumerated when a micro-USB cable from the computer is connected to the **DEBUG USB** port on SAM E54 Xplained Ultra board.

1. Open a terminal application (eg: TeraTerm) and configure the baud rate for **115200**.
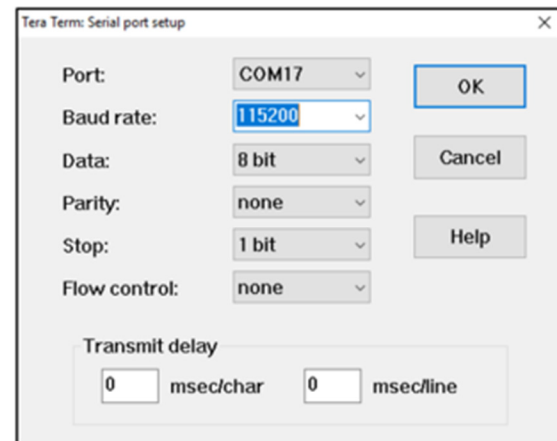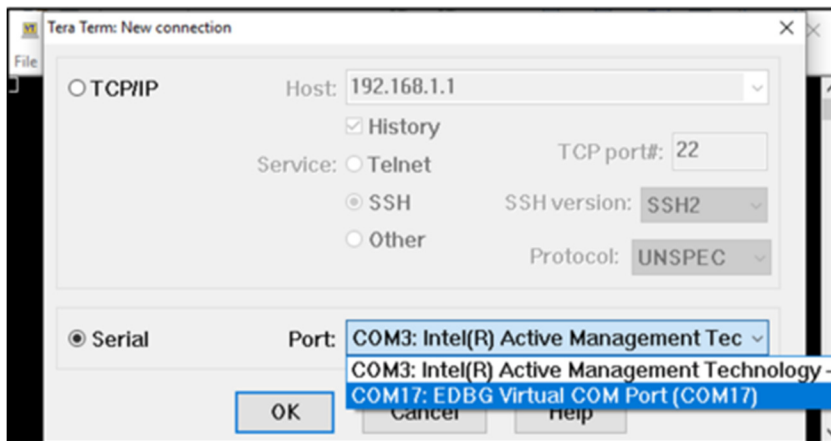
2. Press the **Reset** button on the SAM E54 Xplained Ultra board.
3. The following messages will be visible on the terminal window.
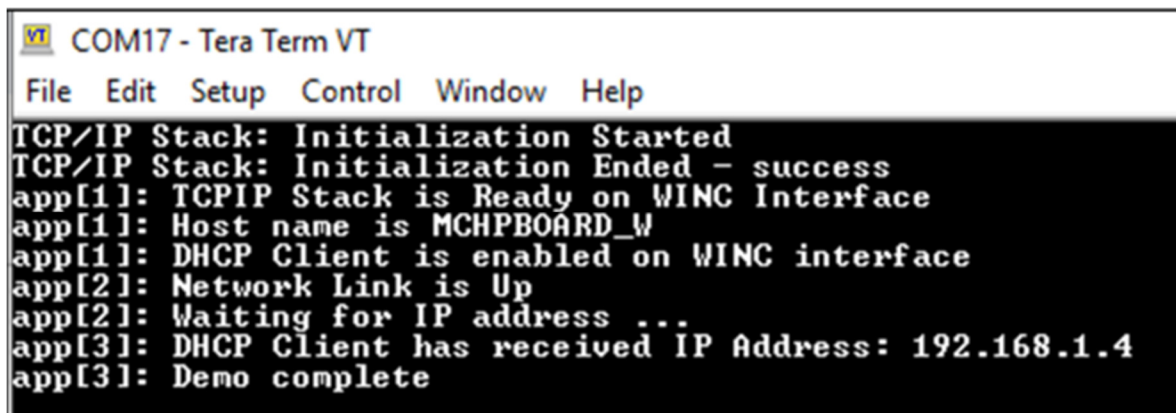
4. Type **ping <PC's IP address>** in the terminal window and press **Enter** to see responses.

```
COM17 - Tera Term VT
File  Edit  Setup  Control  Window  Help
TCP/IP Stack: Initialization Started
TCP/IP Stack: Initialization Ended - success
app[1]: TCPIP Stack is Ready on WINC Interface
app[1]: Host name is MCHPBOARD_W
app[1]: DHCP Client is enabled on WINC interface
app[2]: Network Link is Up
app[2]: Waiting for IP address ...
app[3]: DHCP Client has received IP Address: 192.168.1.4
app[3]: Demo complete

>ping 192.168.1.3
>Ping: reply[1] from 192.168.1.3: time = 12ms
Ping: reply[2] from 192.168.1.3: time = 5ms
Ping: reply[3] from 192.168.1.3: time = 5ms
Ping: reply[4] from 192.168.1.3: time = 5ms
Ping: done. Sent 4 requests, received 4 replies.

>
```

5. You can also ping this board from your PC using cmd prompt as shown below.

```
Command Prompt

C:\Users>ping 192.168.1.4

Pinging 192.168.1.4 with 32 bytes of data:
Reply from 192.168.1.4: bytes=32 time=14ms TTL=128
Reply from 192.168.1.4: bytes=32 time=5ms TTL=128
Reply from 192.168.1.4: bytes=32 time=5ms TTL=128
Reply from 192.168.1.4: bytes=32 time=6ms TTL=128

Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 14ms, Average = 7ms

C:\Users>
```

6. Currently, WINC driver is configured not to display any logs. You can change this setting in MHC by going to **DRIVER LAYER** and modifying the WINC configuration as shown below.
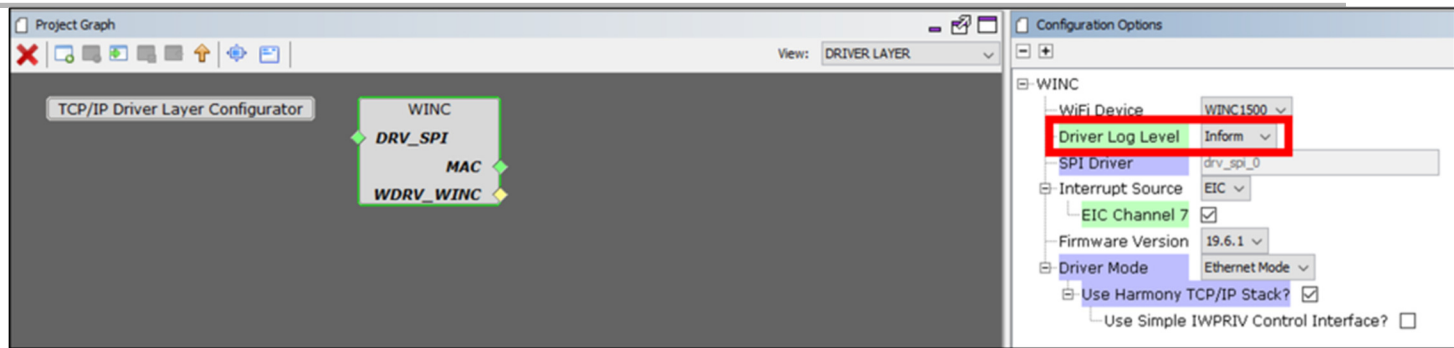
7. After re-generating and re-programming with logging turned on, you will see following output in tera term console.

You have successfully created and tested your first WINC1500 Bypass mode Application.