# User Manual

## DA16200 AT GUI Tool

### UM-WI-004

# Contents

# Figures



## Terms and Definitions

| | |
|---|---|
| GUI | Graphical User Interface |
| UART | Universal Asynchronous Receiver Transmitter |
| SSID | Service Set IDentifier |
| DHCP | Dynamic Host Configuration Protocol |
| AP | Access Point |
| USB | Universal Serial Bus |
| MFC | Microsoft Foundation Class |

## References

[1] DA16200, Datasheet, Dialog Semiconductor

[2] DA16200, SDK Programmer Guide, User Manual, Dialog Semiconductor

[3] DA16200, EVK User Guide, Dialog Semiconductor

[4] DA16200, AT Command User Guide, Dialog Semiconductor

# 1 Introduction

The DA16200 GUI tool (Dialog_IoTWiFi_GUI_yyyymmdd.EXE) lets users control the DA16200 EVK in a GUI environment. There are two modes of operation: Certification mode and Network mode.

## 1.1 Certification Mode

This mode is for Wi-Fi RF test (Tx power, Rx sensitivity, etc.) also known as “TEST Mode”.

## 1.2 Network Mode

With Network mode, the Station or AP mode of the DA16200 is tested.

- **AP mode**: Soft-AP test with configurable parameters like SSID, security, DHCP Server, etc
- **Station mode**: STA test to search and connect to an Access Point and check/test the STA function

After the network mode is set, the user can test the TCP/UDP or the MQTT protocol and manage the TLS certificates.

- **Data Transfer**: message exchange through the TCP Server/Client, and UDP session
- **MQTT Client**: message exchange through the MQTT protocol
- **TLS Setting**: management of TLS certificate that is set (e.g. Root CA, Client Certificate, Client Private Key)

## 1.3 OTP Mode

This mode is for **power calibration** and **temperature calibration with OTP memory location.**

# 2 How to Connect

## 2.1 USB to Serial Driver

The DA16200 evaluation board supports both serial port (UART) and USB port (default USB). The user connects with a micro-USB cable and then two COM ports will be detected automatically.

- If USB port is used, install the FT232 Driver for windows
- In most cases, it will be installed automatically

The FTDI driver for the FT2232 FTDI chip used on the Pro-DK motherboard is available for download at the following link: http://www.ftdichip.com/Drivers/CDM/CDM21224_Setup.zip

When the PC and the EVB board are connect, two ports will be detected.

- See the EVK User Guide [3] to find information about the UART1 port. GUI tool uses UART1 port.

  The other port (UART0) is for debug console connection. This port is used for firmware updates.
- To update with a new DA16200 firmware, see the DA16200 EVK User Guide [3].

The RF certification program was developed by MFC. Therefore, this program may require installation of the MFC library package.

- The Visual Studio Redistributable file is available for download at:

  https://www.microsoft.com/en-us/download/details.aspx?id=48145
- Select the Windows OS type (32-bit or 64-bit), and then do download and install.

## 2.2 Setup Serial Terminal Utility

Users can control DA16200 with a serial terminal tool (such as Teraterm or SecureCRT). The following configurations are required to connect to the console of DA16200:

- Port: Com port number on Windows system
- Baud rate: 115,200 bps
- Data bits: 8 bits
- Parity: None
- Stop bits: 1 bit
- Flow control: None

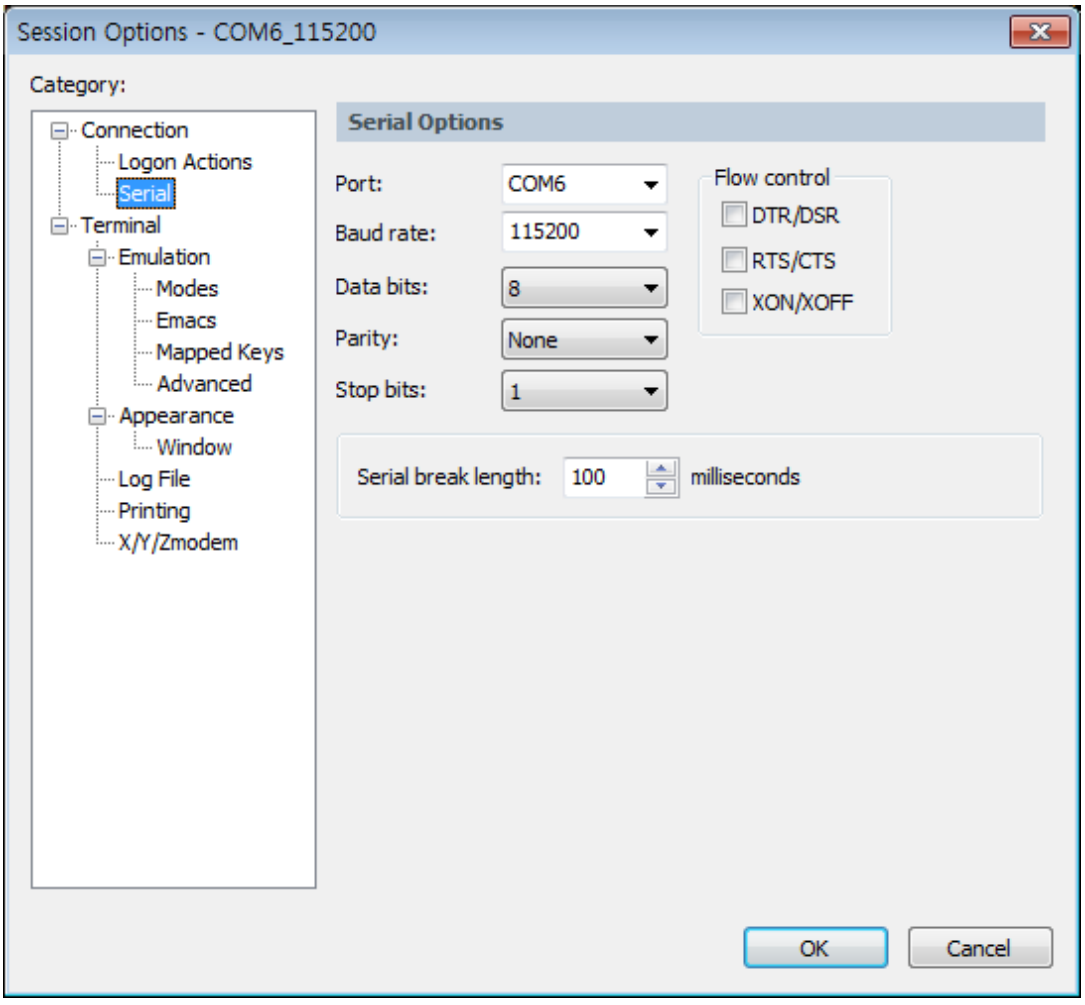



**Figure 1: Terminal Configuration**

When the DA16200 board is connected, the Windows operating system will detect two COM ports. One (UART0) is for console command and the other (UART1) is for AT command. Normally, the higher number COM port is for AT commands and the lower number COM port is for the console.

# 3 How to Run GUI for DA16200 Board

## 3.1 Board Setup

Some source files need to be modified in the DA16200 SDK to use the GUI function.

Enable AT Command function as shown in the example below.

```
[\src\customer\config_generic_sdk.h]

// AT-CMD features
    #define   __SUPPORT_ATCMD__                    // Support AT-CMD
```

```
[\src\customer\sys_common_features.h]

       #define__SUPPORT_NET_CMD__          // Network feature commands
       #define__SUPPORT_LMAC_RF_CMD__      // lmac/rf feature commands
       #define__SUPPORT_PERI_CMD__         // peri. feature commands
```

DA16200 AT GUI program is a single executable file.

| NOTE |
| --- |
| If there is an error with the message "*a DLL file (e.g. vcruntime140.dll or mfc1400u.dll) is missing*", then install Microsoft Visual Studio redistributable package (https://www.microsoft.com/en-us/download/details.aspx?id=48145) or copy the .dll file into the Windows system folder (C:\Windows\System32 or C:\Windows\SysWOW64). |

The program setup sequence is:

1. Connect the DA16200 Development Kit to the host PC as shown in Figure 2.

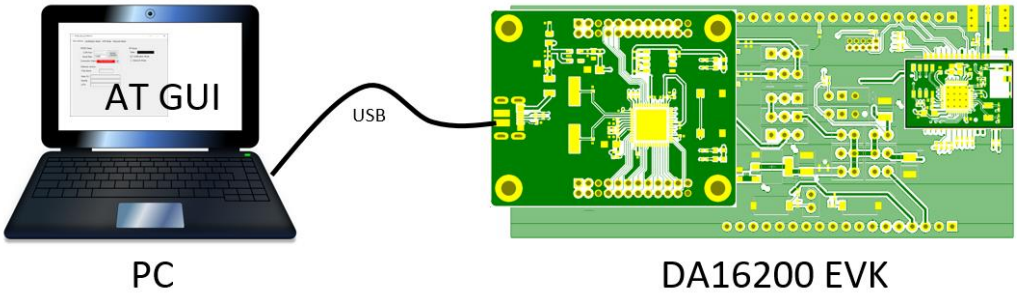



**Figure 2: Development Kit Connections**

| NOTE |
| --- |
| See the EVK User Guide [3] to know which port to use. |

2. Start the AT GUI program.

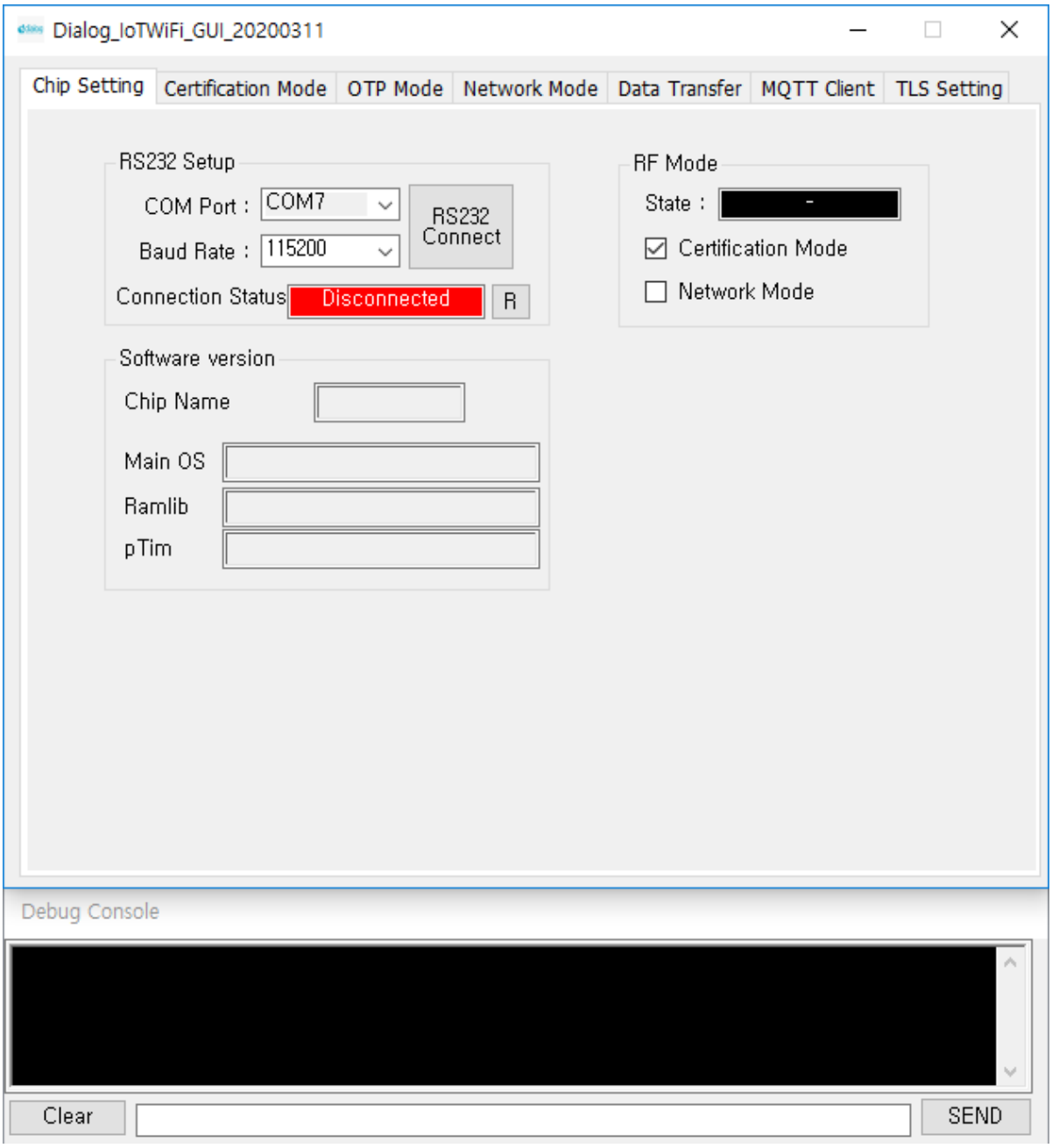



**Figure 3: AT GUI**

## 3.2 Board Connection

1. When the AT GUI program runs, the connection status is red (not connected). If the program does not detect any COM port, click the **R** button to refresh the COM Port. See Figure 4.

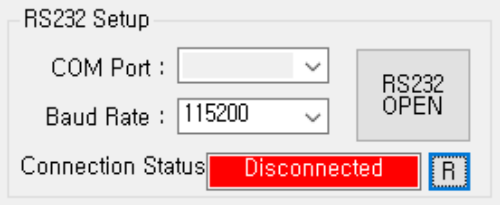



**Figure 4: RS232 Setup - Disconnected**

2. Select a COM port and click the RS232 **OPEN** button and wait for a few seconds. See Figure 5 and Figure 6.
   - The connection status will soon change to a green color and the Connection Status field shows “Connected” in a green color. This means that communication is OK.

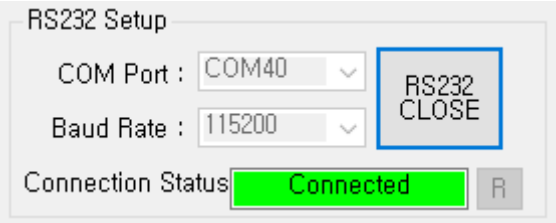



**Figure 5: RS232 Setup - Connected**

   - In the Debug Console window (in the black box), the message “`Echo on`” is shown.

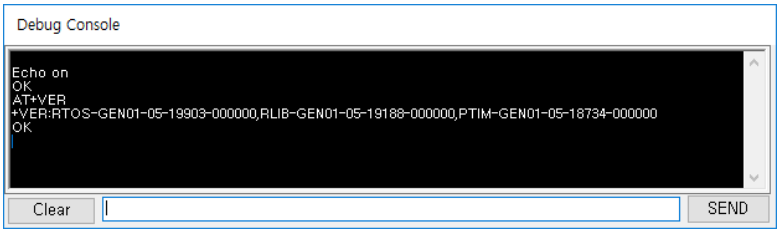



**Figure 6: Debug Console - Connected**

## 3.3 RF Certification Mode

To enable certification mode:

1. Open the **Chip Setting** tab. See Figure 7.
2. In the **RF Mode** area, select the **Certification Mode** check box.

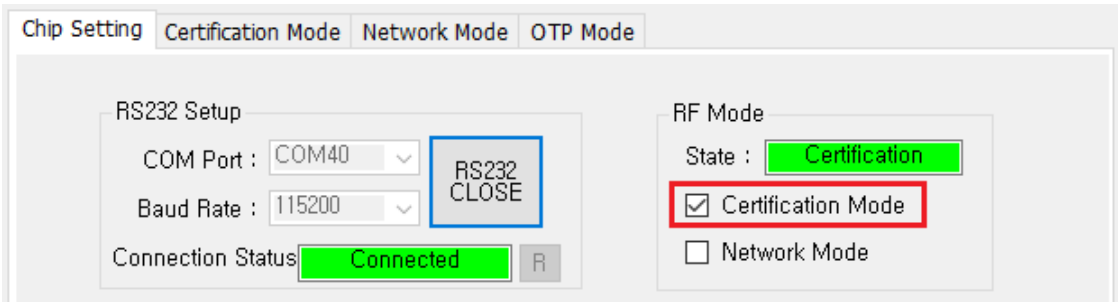



**Figure 7: Certification Mode Configuration**

- Alternatively, on the **Certification Mode** tab, select the **Certification Mode** checkbox. See Figure 8.

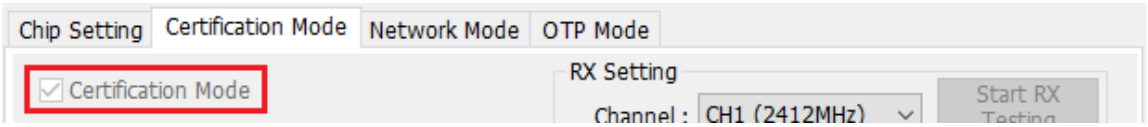



**Figure 8: Certification Mode Tab**

## 3.4 TX Test Mode

1. Open the **Certification Mode** tab. See Figure 9.

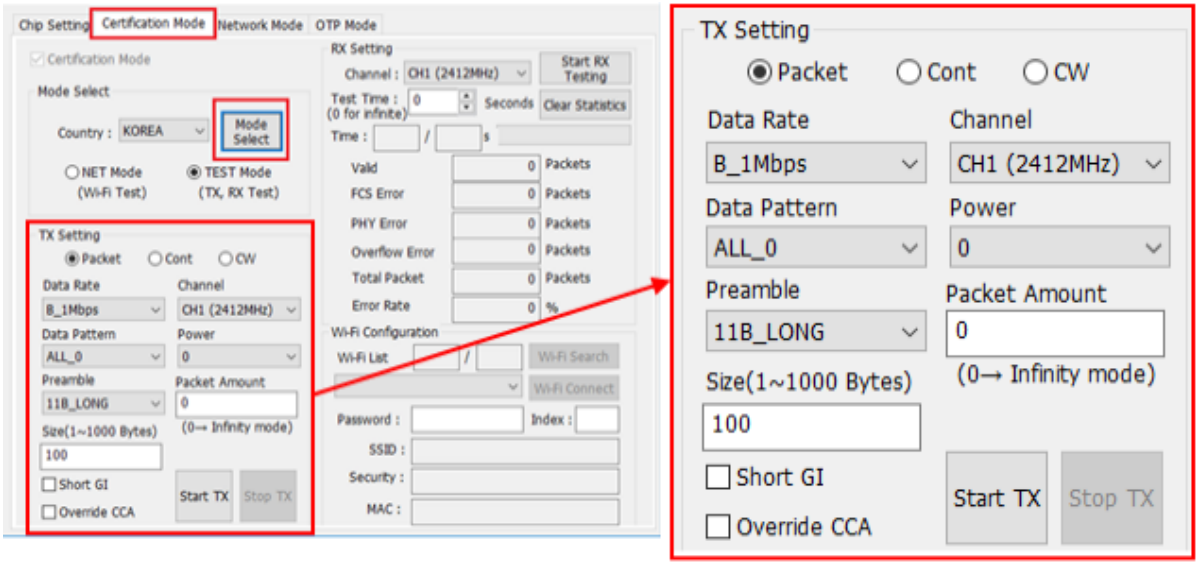



**Figure 9: Tx Test Mode Configuration**

2. Click the **Mode Select** button (Figure 9). A confirmation message is shown as in Figure 10. This is a normal state.

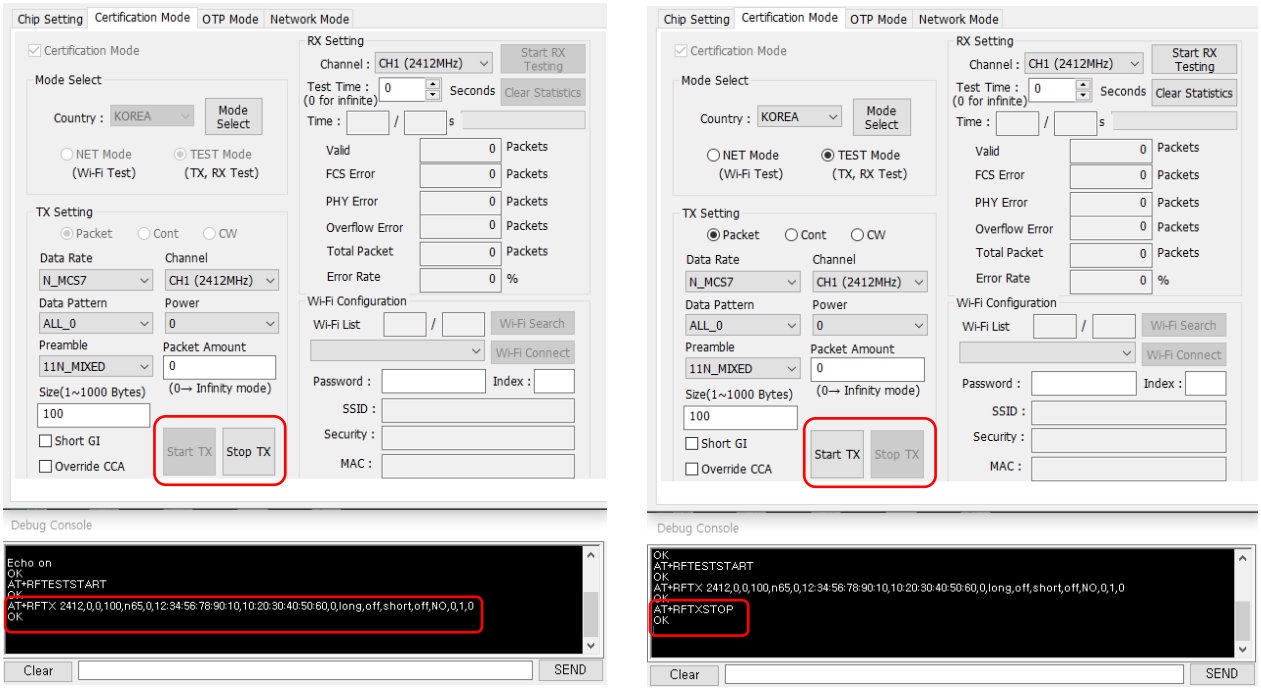
```
AT+TMRFNOINIT=1
OK
AT+RESTART
OK

+INIT:DONE

Echo on
OK
AT+RFTESTSTART
OK
```

**Figure 10: Debug Console - TX Mode**

3. Select the **Data Rate**, **Channel**, and **Power** for the purpose of the test.
   - **Packet mode**: this is the normal test mode with packet generation mode. Offers the possibility to adjust duty of RF Burst at time domain
   - **Cont mode**: Continuous TX out mode. This mode is for TX power test etc. In this mode, TX packet is generated continuously over 95% duty cycle
   - **CW mode**: Only single sinewave tone out mode. This mode is for freq err check
   - **Data Rate**: Choose modulation type to test
   - **Power**: Select or tune the power level. (“0” step is Maximum). The difference between power steps is about 0.8~1 dB/1step
   - **Size**: You can adjust the duty rate with this number. However, the size is not linear as the number, so to set the exact number you need equipment like a spectrum analyzer to check the value. (Equipment setting is set to zero span setting or burst mode setting)

To do TX packet generation:

- To start and stop TX packet generation, use the **Start TX** button and the **Stop TX** button. If you want to make changes for another condition, click **Stop TX** before a new test is started.




Start TX Stop TX

**Figure 11: Start and Stop TX**

For example: there is a test setting with 802.11n MCS7, channel 1, 100 bytes packet and power grade 0.

- When the **Start TX** button is clicked, messages as shown in Figure 12 will be shown.

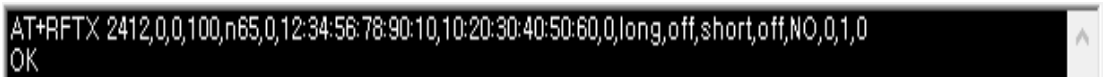



**Figure 12: Debug Console - Start TX**

- When you click the **Stop TX** button, messages as shown in Figure 13 will be shown.

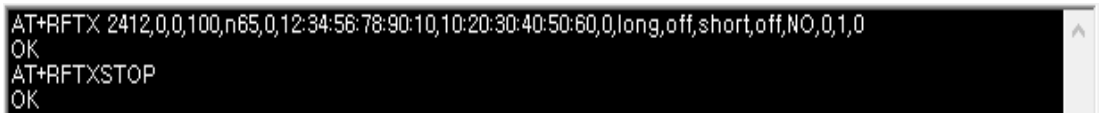



**Figure 13: Debug Console - Stop TX**

## 3.5 RX Test Mode

The settings are made in the **RX Setting** area. See Figure 14.

- **Channel**: Support CH1 ~ CH13
- **Test Time**: Maximum 3600 s (Duration is 1 second fixed)

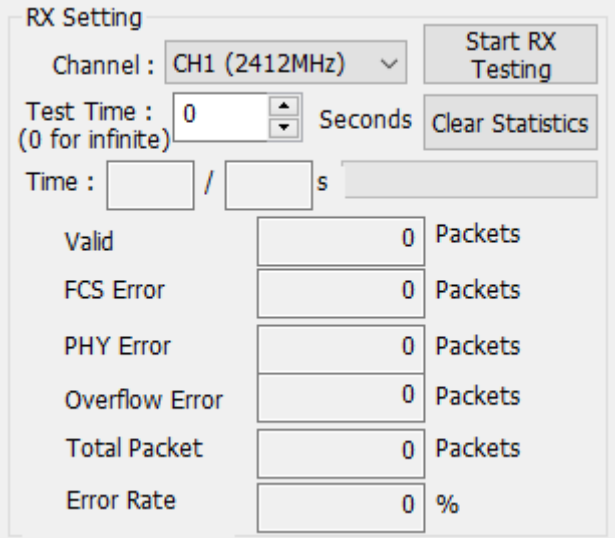



**Figure 14: Rx Configuration**

- RX Packet Rate
  - FCS + PHY + Overflow packet / Total packet = Error rate

# 4 Network Mode

To enable network mode:

1. Open the **Chip Setting** tab.
2. Select the **Network Mode** checkbox. See Figure 15.

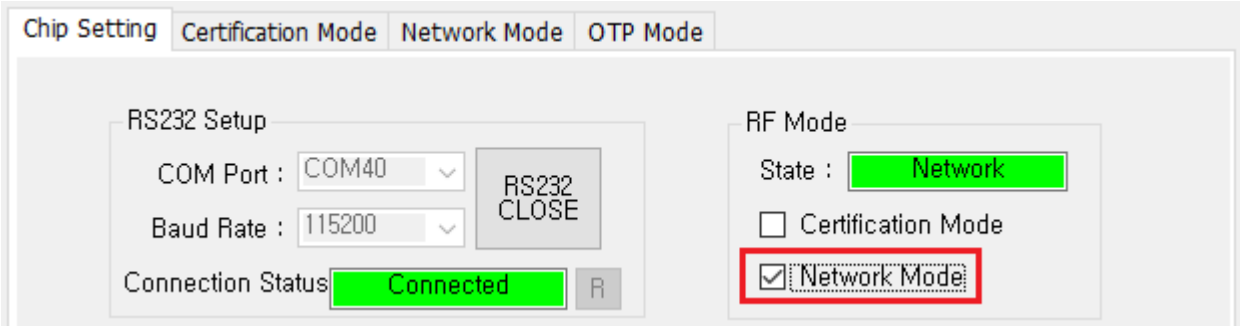



**Figure 15: Network Mode Configuration**

- Alternatively, open the **Network Mode** tab and select the **Network Mode** checkbox. See Figure 16.

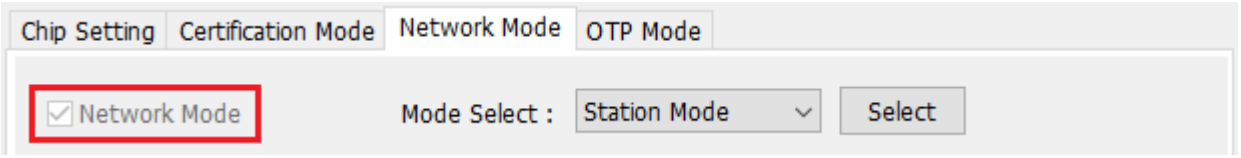



**Figure 16: Network Mode Tab**

## 4.1 Station Mode

1. In the **Mode Select:** field, select **Station Mode** and click the **Select** button. See Figure 16.
   - The network mode is changed. Next, DA16200 reboots and the station mode setup window opens. See Figure 17.

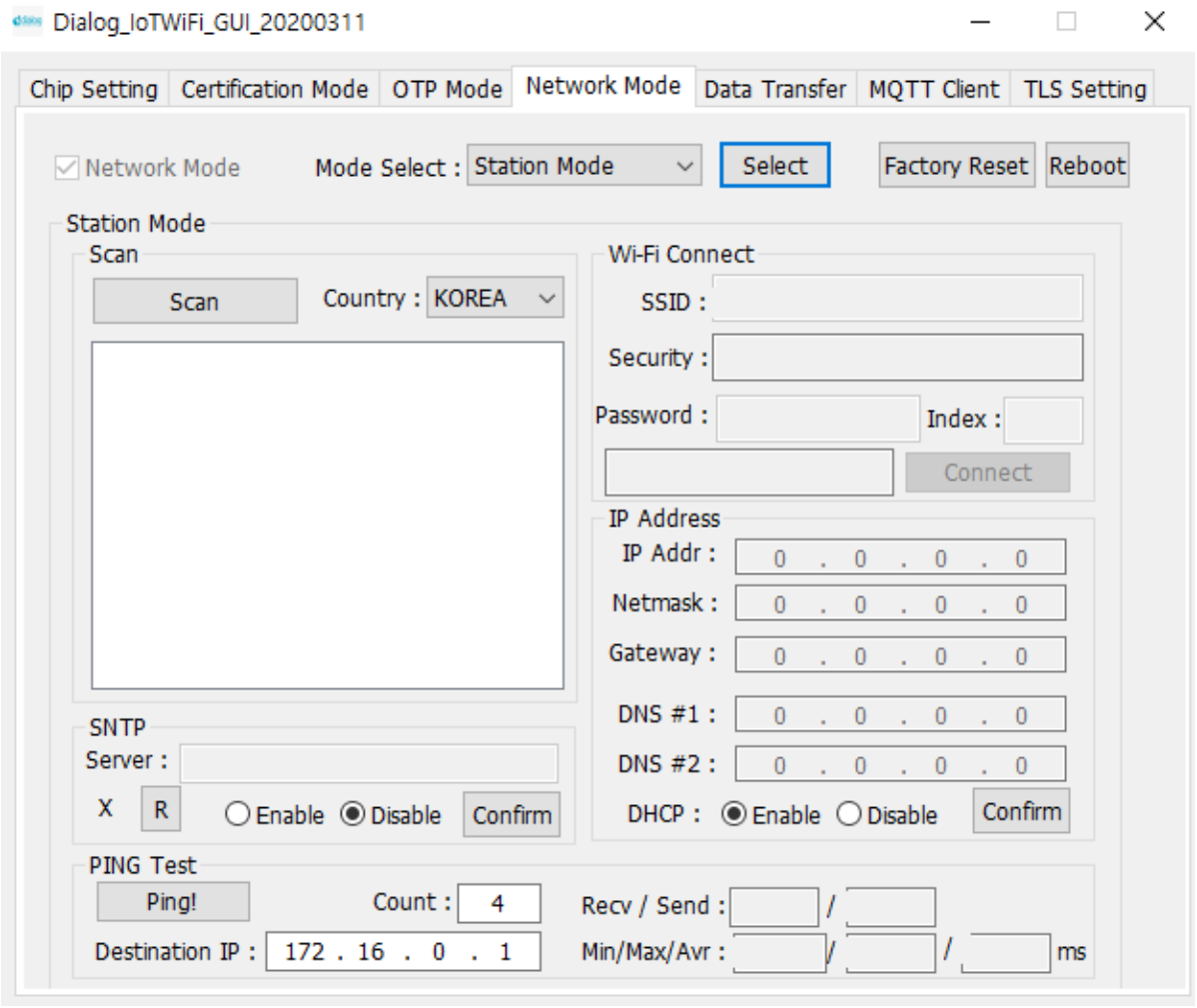



**Figure 17: Setup Window - Station Mode**

2. Click the **Scan** button to scan APs. See Figure 18.

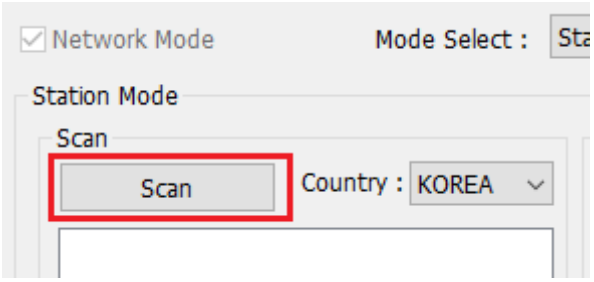



**Figure 18: Station Mode - SCAN**

3. When scanning is finished, choose one AP in the list. See Figure 19.

4. Click the **Connect** button.
5. If required by the security mode of the AP, fill in a password or key index.

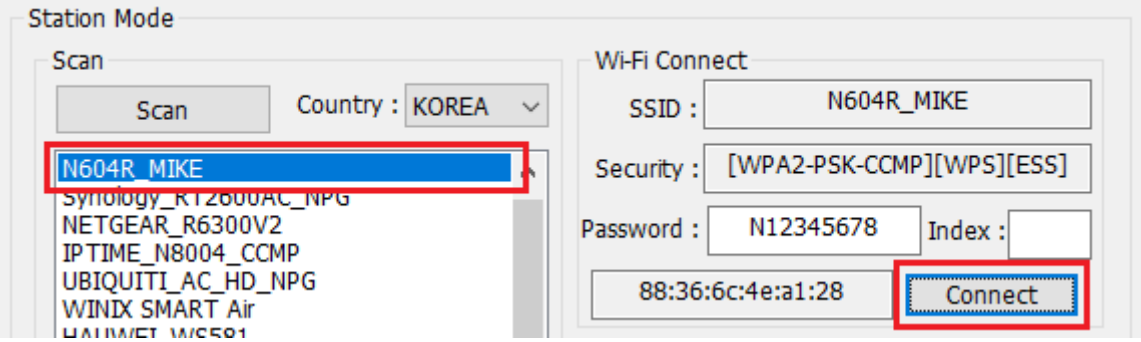



**Figure 19: Station Mode - Choose AP and Connect**

```
AT+WFJAP=N604R_MIKE,4,2,N12345678
OK

+WFJAP:1,N604R_MIKE,192,168,0,9
AT+NWIP=?
+NWIP:0,192,168,0,9,255,255,255,0,192,168,0,1
OK
```

**Figure 20: Debug Console - Connect to AP**

## 4.2 AP Mode

1. In the **Mode Select** field, select **AP Mode** and click the **Select** button. See Figure 16.
   - The network mode changes, DA16200 reboots and the AP Mode setup window opens. See Figure 21.

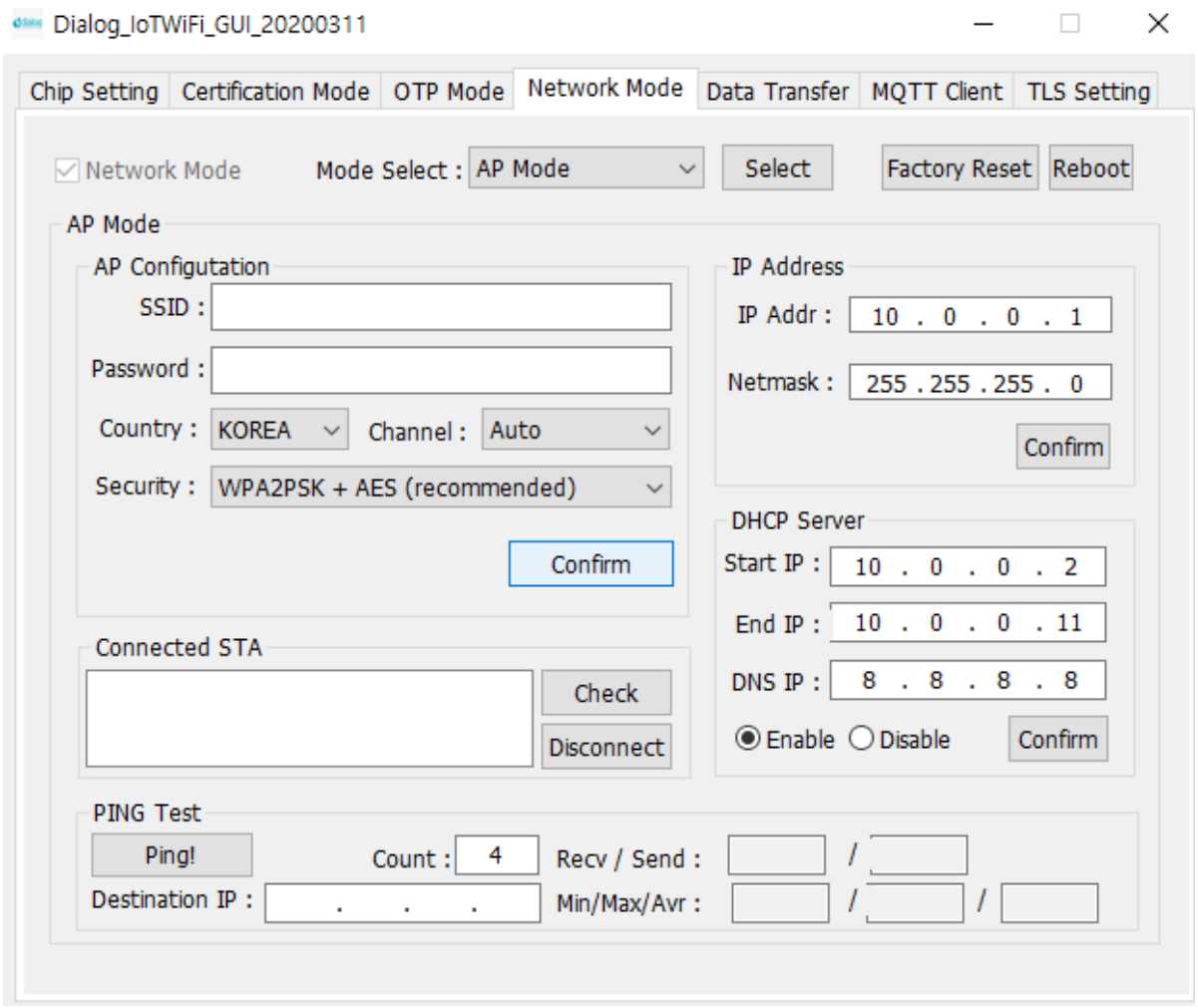



**Figure 21: Setup Window - AP Mode**

2. Set the fields SSID, Password, Country, Channel, and Security mode. See Figure 22.

3. Click the **Confirm** button.

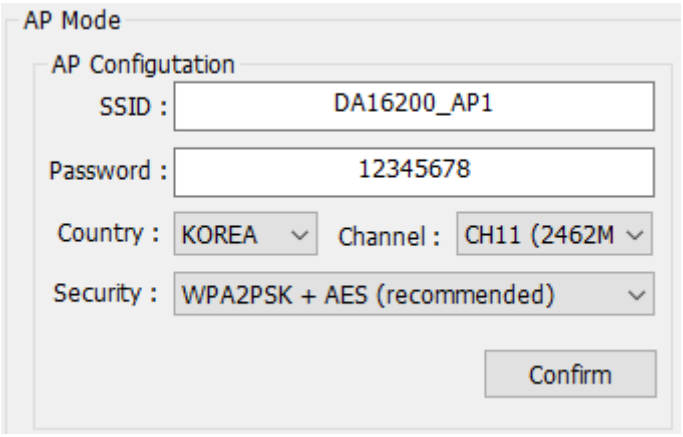



**Figure 22: AP Configuration**

- The Debug Console window messages appears as shown in Figure 23.


**Figure 23: Debug Console - AP Configuration**

4. In the **DHCP Server** area, make the required settings for the IP addresses and click the **Confirm** button. See Figure 24.
   - When a client is connected, the MAC address will be shown as in Figure 25.

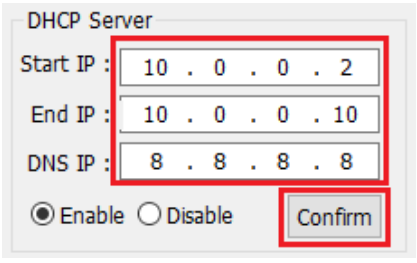



**Figure 24: DHCP Configuration**


**Figure 25: Debug Console - DHCP Configuration**

## 4.3 Data Transfer (TCP/UDP)

The DA16200 GUI tool provides data transfer functions with TCP/UDP. Before messages are sent with this tool, connect the DA16200 to an access point as shown in Section 4.1. Then click the **Data Transfer** tab. If DA16200 succeeds to connect to an AP, the IP address will be filled in.
See Figure 26.

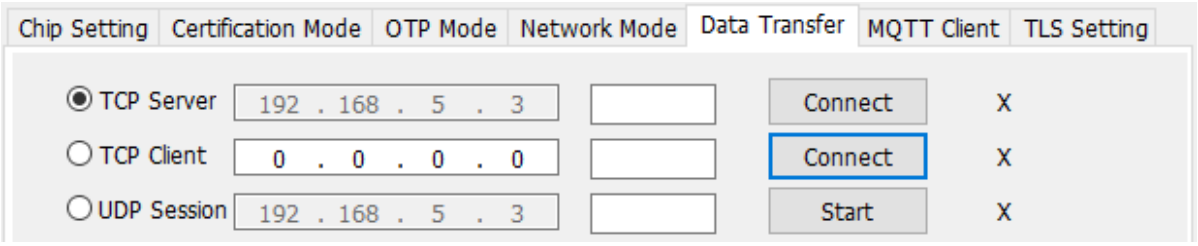



**Figure 26: Data Transfer Tab**

The DA16200 provides three kinds of sessions: TCP server, TCP client, and UDP session. The user can use these three simultaneously.

### 4.3.1 TCP Server

To use TCP server, fill in a port number and click the **Connect** button on the **TCP Server** line. When a server is opened, the **X** will change to **O**. See Figure 27.

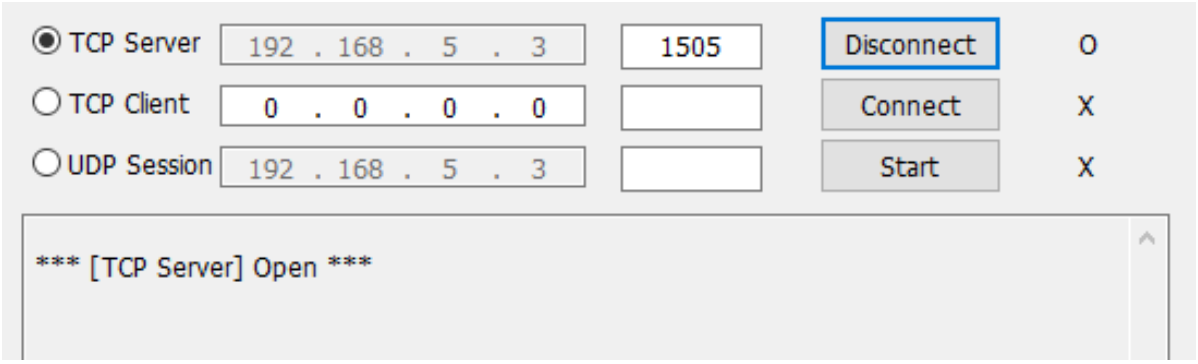



**Figure 27: TCP Server Open**

The user can connect to the DA16200 TCP server with tools for data exchange like the IO Ninja. When a client connects successfully, its information will be shown as in Figure 28. The DA16200 TCP server can accept up to eight client sessions.

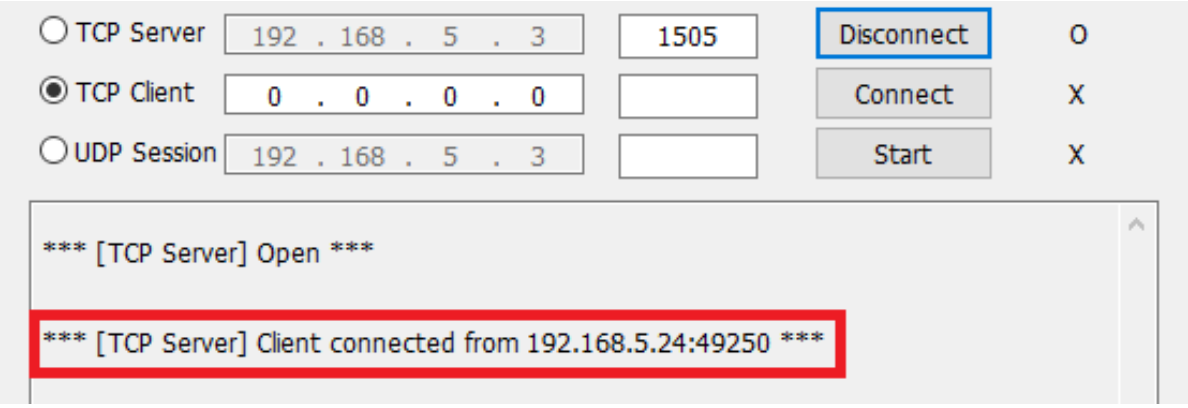



**Figure 28: TCP Server Connection with a Client**

### 4.3.2 TCP Client

To connect to a TCP server, fill in the IP address and port number of the server and click the **Connect** button on the **TCP Client** line. When the DA16200 TCP client succeeds to connect, the **X** will change to **O**. See Figure 29.

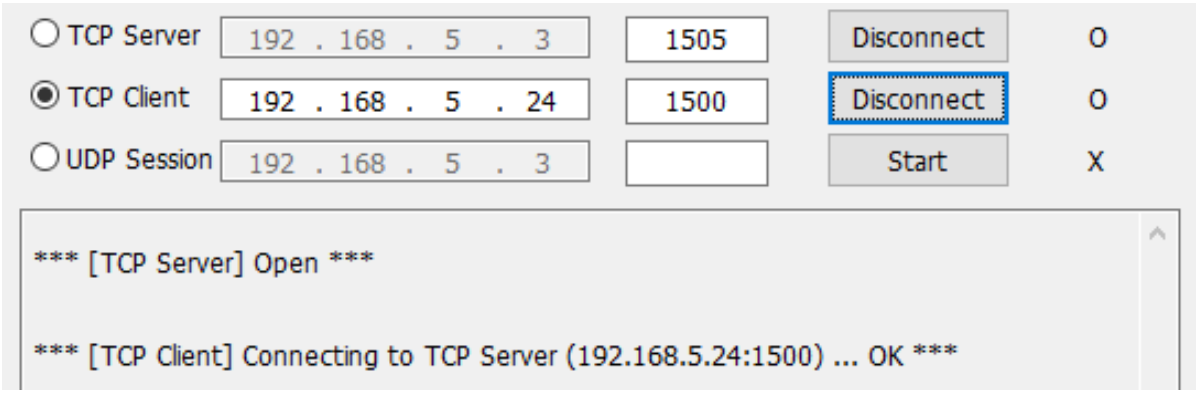



**Figure 29: TCP Client Connection to a Server**

### 4.3.3 UDP Session

To open a UDP session, fill in a port number and click the **Start** button. When the DA16200 TCP client succeeds to connect, the **X** will change to **O**. See Figure 30.

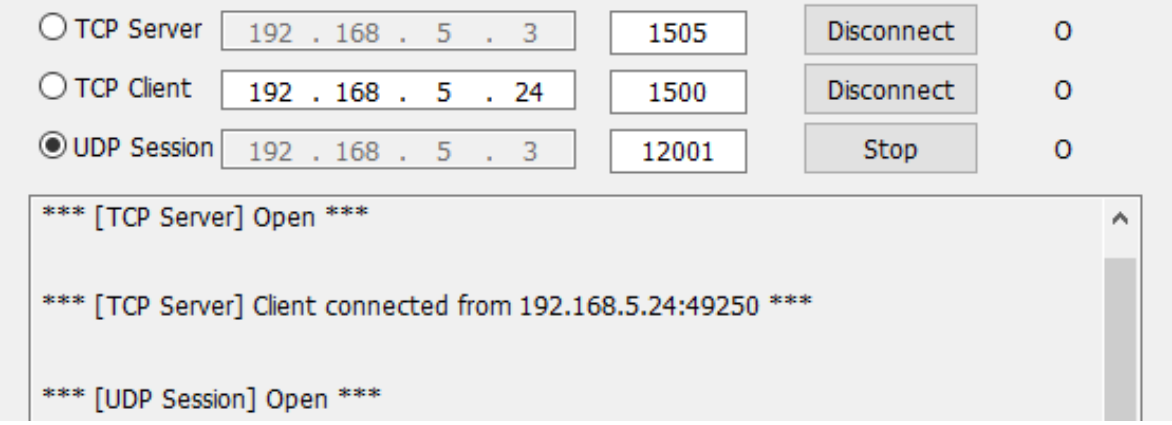



**Figure 30: UDP Session Open**

### 4.3.4 Data Exchange

To send a message to a peer, first select a session. Each session is slightly different in method.

- **TCP Server**: Select the **TCP Server** checkbox→ Select the destination IP in the drop-down list → Type a message → Click the **Send** button (see Figure 31)

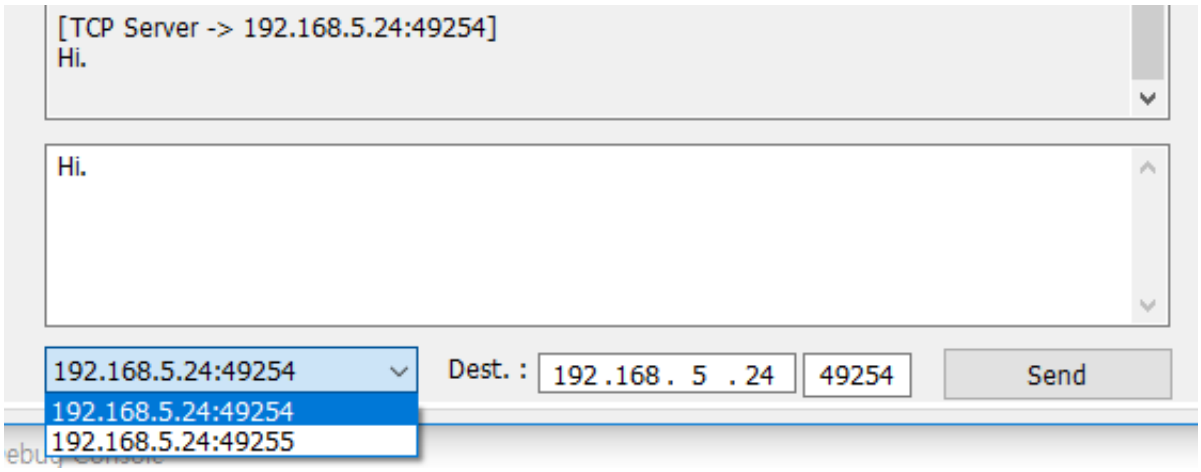



**Figure 31: TCP Server Data Transfer**

- **TCP Client**: Select the **TCP Client** checkbox→ Type a message → Click the **Send** button (see Figure 32)

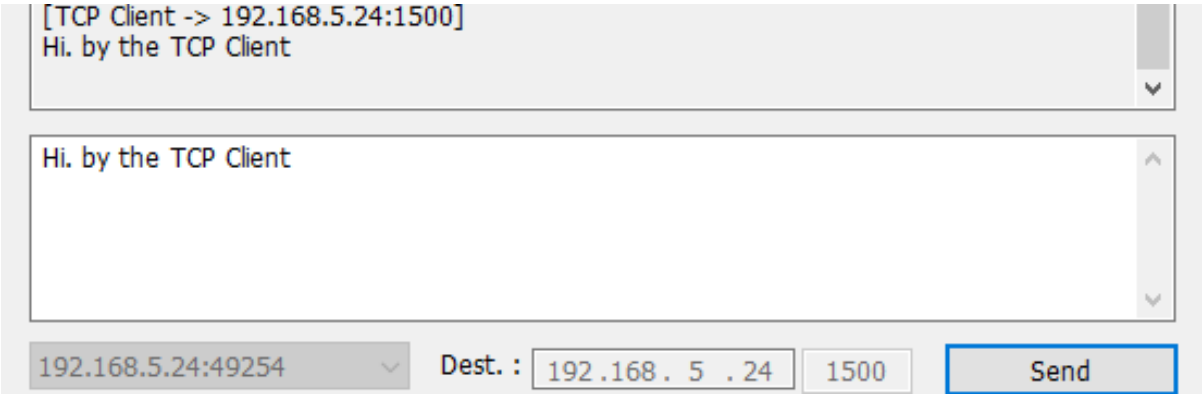



**Figure 32: TCP Client Data Transfer**

- **UDP Session**: Select the **UDP Session** checkbox→ Enter the destination IP and port number → Type a message → Click the **Send** button (see Figure 33)

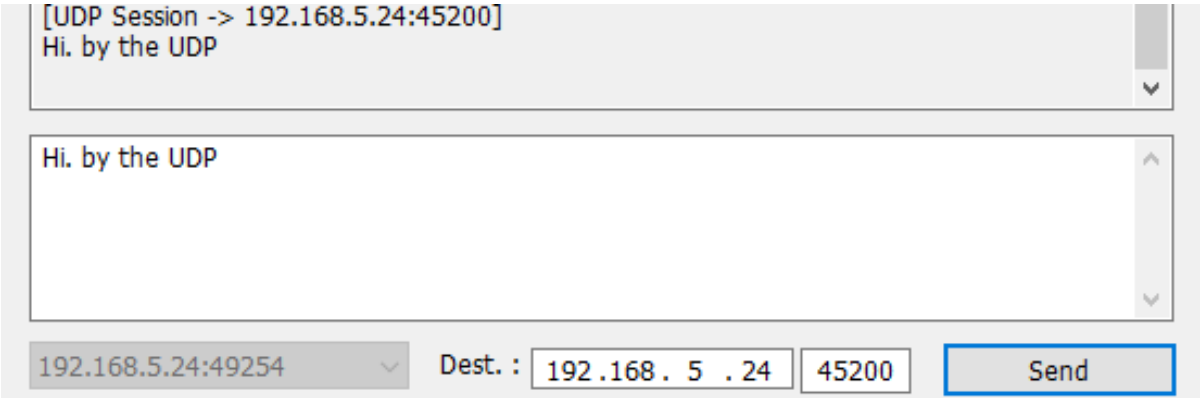



**Figure 33: UDP Session Data Transfer**

When a session receives a message from a peer, the message is shown in the message window. See Figure 34.


**Figure 34: TCP/UDP Data Reception**

### 4.3.5 Data Transfer on DPM Mode

The DA16200 GUI provides TCP/UDP sockets operation in DPM (DA16200 power-save) mode. See Figure 35.

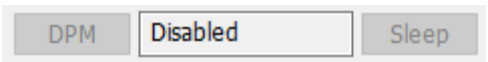



**Figure 35: DPM Mode Setting**

- **DPM Button**: DPM mode Start / End
- **DPM_STATUS**: DPM state (Disabled / DPM init… / DPM Sleep / DPM wake-up)
- **Sleep Button**: DA16200 returns to DPM sleep state (only available DPM wake-up state)

To initiate TCP/UDP sockets in DPM mode:

1. Open sockets user wants to.
2. Click the **DPM** button to start DPM mode.
3. After all sockets are ready in DPM mode, the **DPM_STATUS** will change to "DPM sleep".

When DA16200 in DPM sleep state receives a TCP or UDP messages from a peer, it is shown in the message window like in Figure 34.

To send a message in DPM Sleep status:

1. Wake up DA16200 with the switch. The **DPM_STATUS** will change to "DPM wake-up".
2. Select the session to send, type a message, and click the **Send** button.
3. After sending messages, click the **Sleep** button. The **DPM_STATUS** will change to "DPM sleep".

To exit DPM mode:

1. Wake up DA16200 with the switch. The **DPM_STATUS** will change to "DPM Wake-up".
2. Click the **Sleep** button. The **DPM_STATUS** will change to "Disabled".

## 4.4 MQTT Client

The DA16200 GUI tool provides data transfer functions with MQTT protocol. Before messages are sent with this tool, connect the DA16200 to an access point as shown in Section 4.1. Click the **MQTT Client** tab.

### 4.4.1 Configuration

Before a connection is made to an MQTT broker, set the required fields for the connection information. See Figure 36.

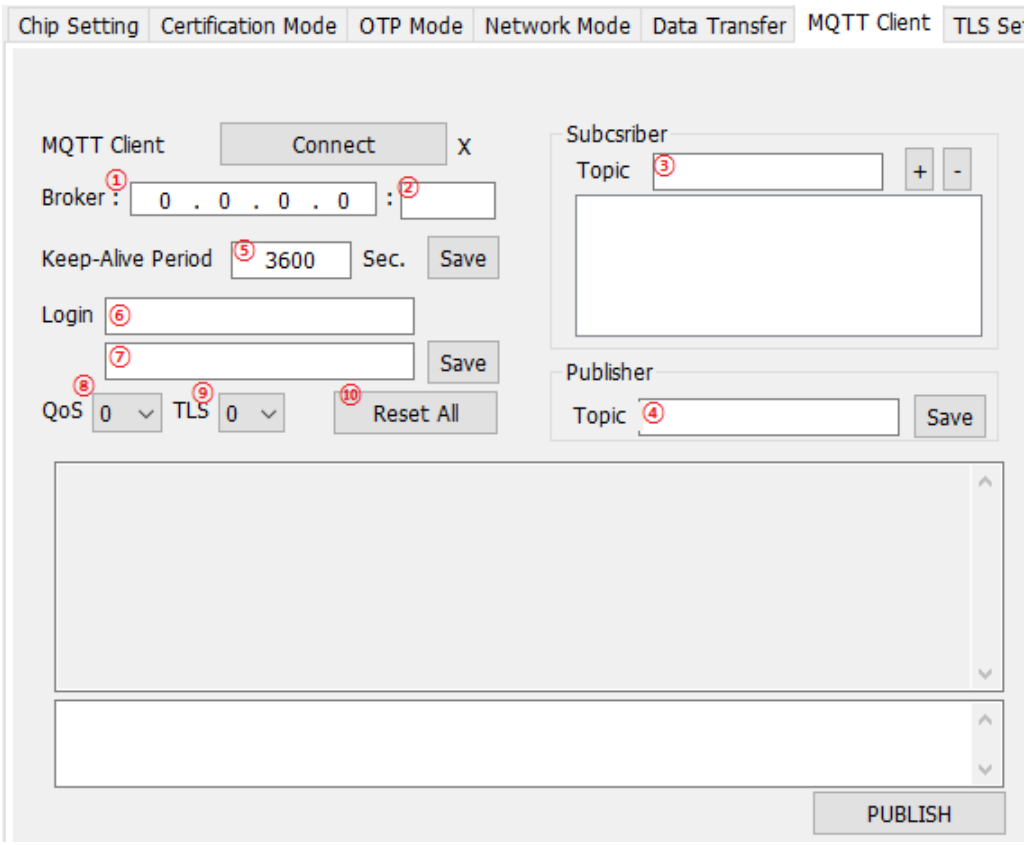



**Figure 36: MQTT Client Tab**

① Broker IP address

② Broker port number

③ Subscriber topics (up to 4)

④ Publisher topic

⑤ Sending PINGREQ cycle (second)

⑥ Login ID

⑦ Login password

⑧ MQTT QoS (0~2)

⑨ TLS use (0 or 1)

⑩ Reset all configuration to default

When the DA16200 MQTT client succeeds to connect, the letter **X** will change to **O**. Moreover, the user can send a PUBLISH message to the broker or receive a message.

### 4.4.2 Data Exchange

To exchange a message with the broker, connect the MQTT Client.

To send a PUBLISH, type the message and click the **PUBLISH** button. See Figure 37.

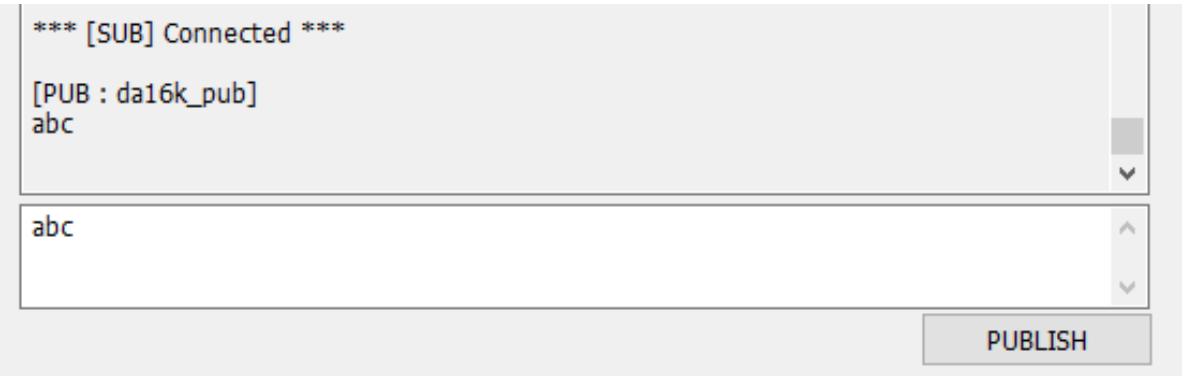



**Figure 37: MQTT PUBLISH Transfer**

When the client receives a message from the broker, the message is displayed in the message window. See Figure 38.

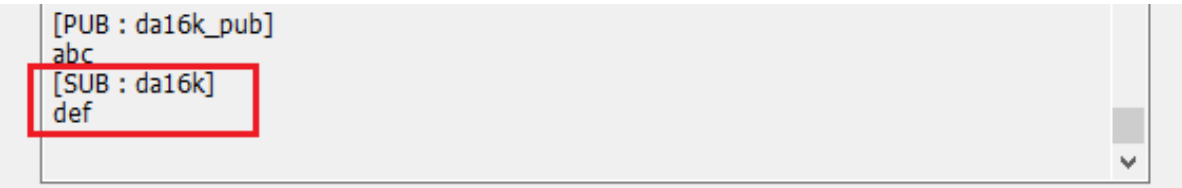



**Figure 38: MQTT PUBLISH Reception**

### 4.4.3 MQTT on DPM Mode

The DA16200 GUI provides MQTT client operation in DPM mode. There is the DPM mode setting as shown in Figure 35 in MQTT Client tab.

To initiate MQTT client in DPM mode:

1. After setting up, connect to a MQTT Broker.
2. Click the **DPM** button to start DPM mode.
3. After the MQTT client session is ready in DPM mode, the **DPM_STATUS** will change to "DPM sleep".

When DA16200 in DPM sleep state receives a PUBLISH from the broker, it is shown in the message window like Figure 38.

To send a message in DPM Sleep status:

1. Wake up DA16200 with the switch. The **DPM_STATUS** will change to "DPM wake-up".
2. Type a message and click the **PUBLISH** button.
3. After sending messages, click the **Sleep** button. The DPM_STATUS will change to "DPM sleep".

And to exit DPM mode:

1. Wake up DA16200 with the switch. The **DPM_STATUS** will change to "DPM Wake-up".
2. Click the **Sleep** button. The **DPM_STATUS** will change to "Disabled".

## 4.5 TLS Setting

The user can store a TLS certificate that is set with this tool for MQTT, HTTPs, CoAPs, and so on. There are three kinds of items: Root CA, Client Certificate, and Private Key.

DA16200 can only process Privacy Enhanced Mail type(.pem). To enter the certificates directly, input the text and click the **Upload** button with the TLS item selection. If this is done successfully, `<Content exists>` will appear in the window. See Figure 39.

If the user has TLS certificate files with PEM type, click **File Search** and select the file. The text will be loaded in the window.

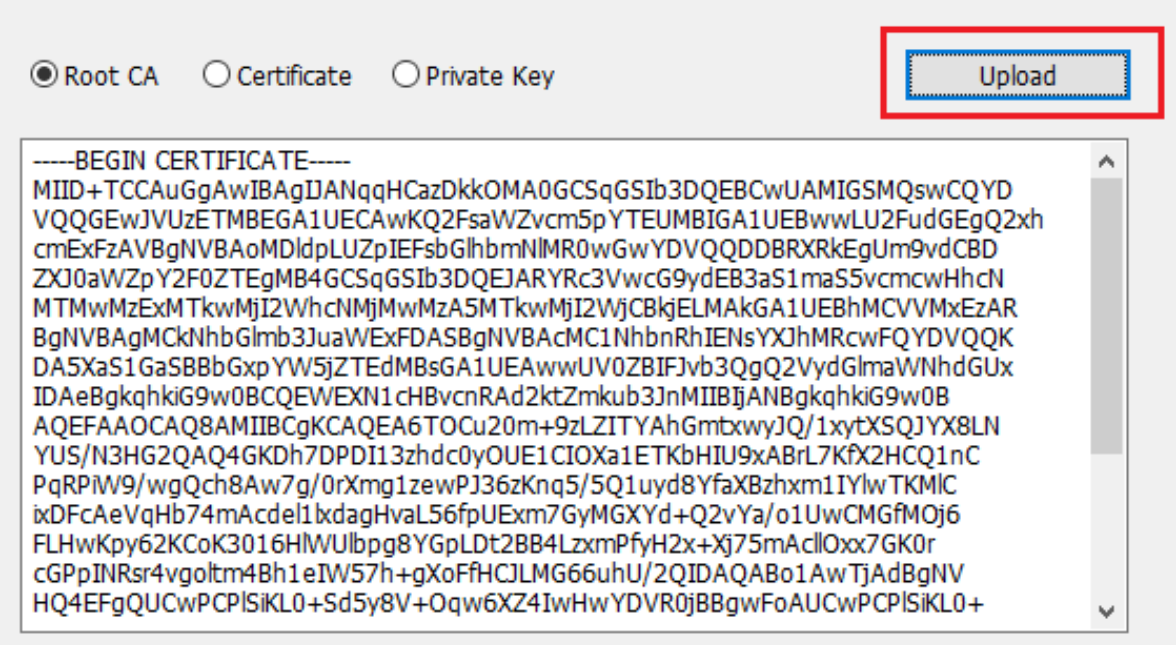



**Figure 39: TLS Setting Tab**

# 5 OTP Mode Setup

To enable OTP mode:

1. Open the **OTP Mode** tab.
2. Click the **OPT Enable** checkbox
   - The current OTP data is as shown in Figure 40.

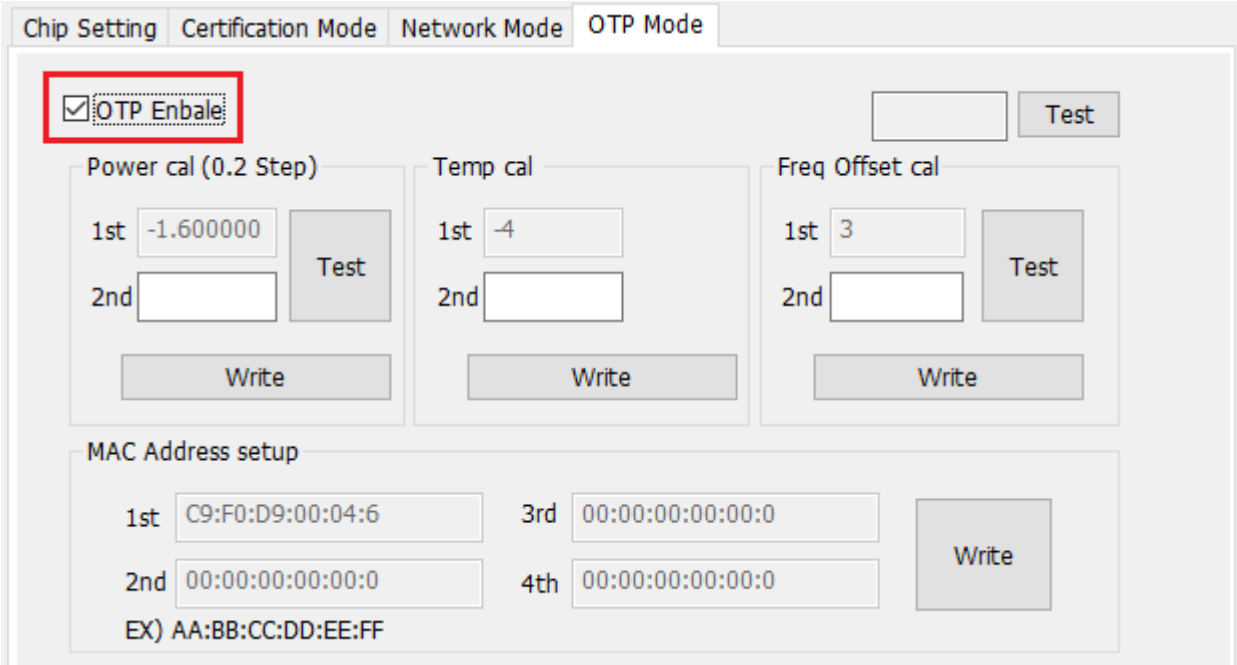



**Figure 40: OTP Mode**

**Power calibration** and **temperature calibration** has been completed for the chipset by Dialog and it may not need to be done by the customer.

**Freq Offset cal:** the range of frequency offset calibration is 0x0 ~ 0x7F.

**MAC Address setup**

DA16200 has 4 slots to store MAC addresses. The 1st slot is written by Dialog. The user can use 3 slots to write their own MAC address. The MAC address written in the OTP must be an even number. it is used for WLAN0 MAC address (for STA interface), and the next number is automatically used for WLAN1 MAC address (for Soft-AP interface).

To write the MAC address, give a MAC address in the first empty slot and click the **Write** button, then the new address substitutes for the previous address.

# 6 How to Build GUI for DA16200 Board

If necessary, the user can get the source code of the GUI. The GUI source code structure is as shown in Figure 41. Folder **MFCApplication2** contains all the source files (.cpp) and header files(.h) for the GUI. The package total size is about 160 MB. The compile environment used in this document is Visual Studio 2017 professional.

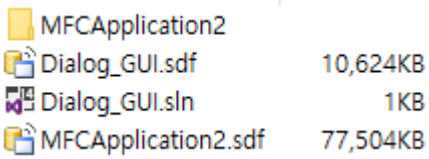



**Figure 41: GUI Source Codes Structure**

Execute the file Dialog_GUI.sln. The GUI project is loaded as shown in Figure 42.

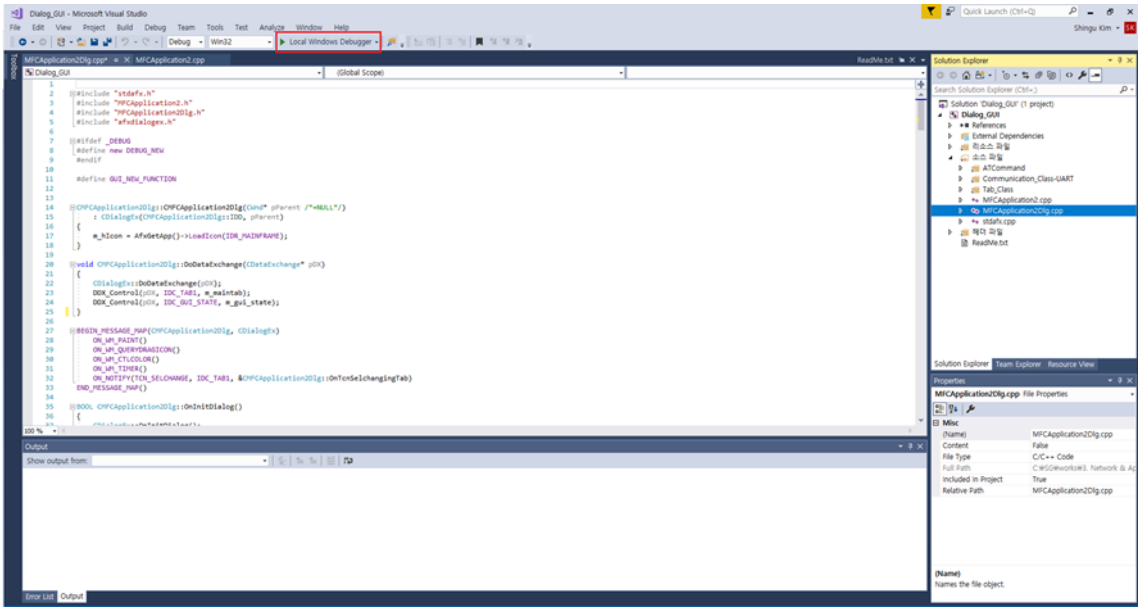


**Figure 42: Dialog GUI Project**

Click the **Local Windows Debugger** in the menu bar. The GUI is built and executed as shown in Figure 43.

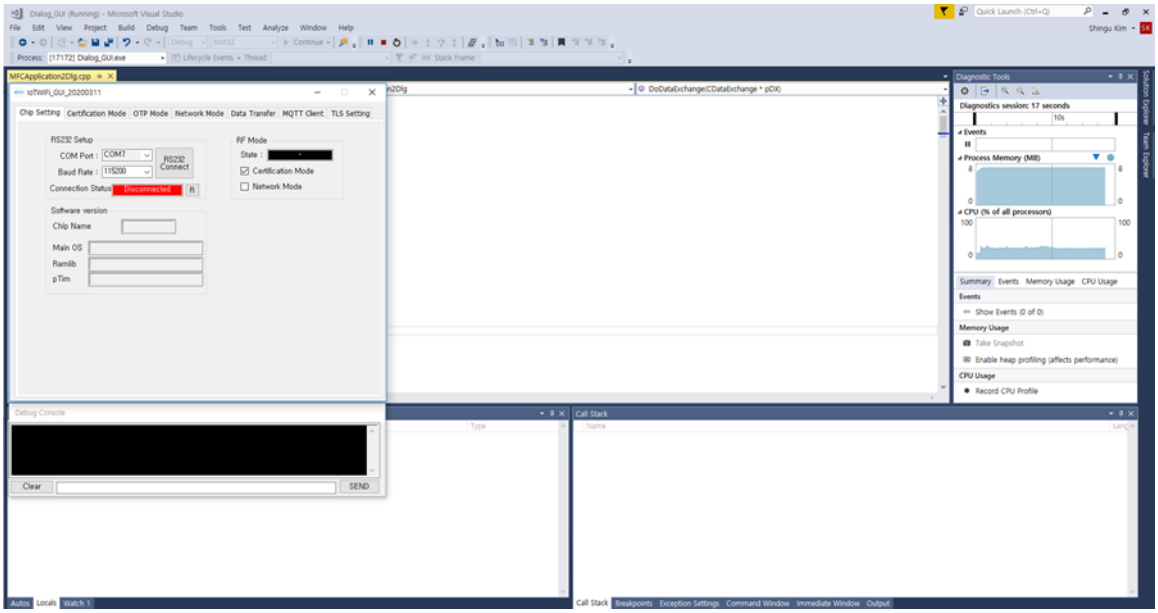


**Figure 43: Local Windows Debugger Execution**

After the GUI is built, the user can see the newly created Debug folder as shown in Figure 44. Then, the user can execute the GUI modified without compile tools. In general, the GUI file name in the Debug folder is Dialog_IoTWiFi_GUI.EXE.

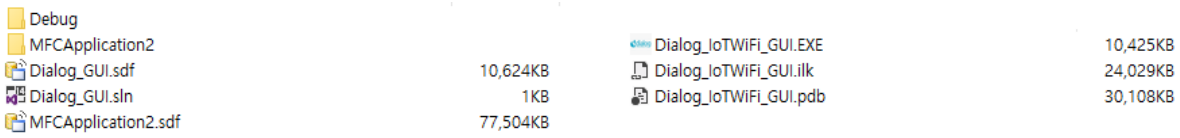



**Figure 44: GUI Files**

To modify the source files(.cpp) or header files(.h), the user can open them from the **Solution Explorer** tab as shown in Figure 45.

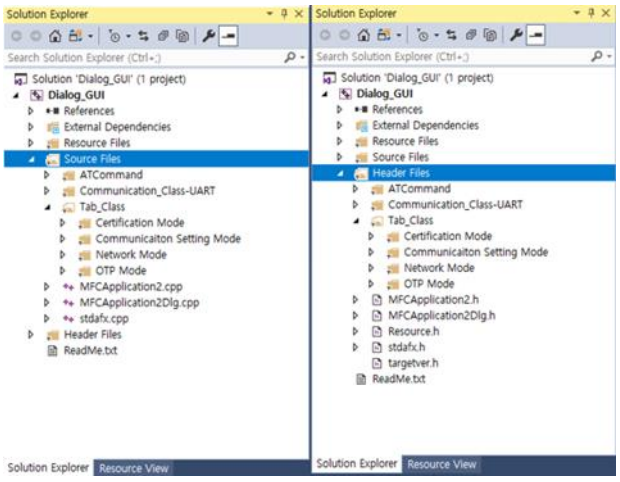



**Figure 45: Solution Explorer Tab**

In the **Resource View** tab, the user can manage resources like the GUI windows and the icon as shown in Figure 46.

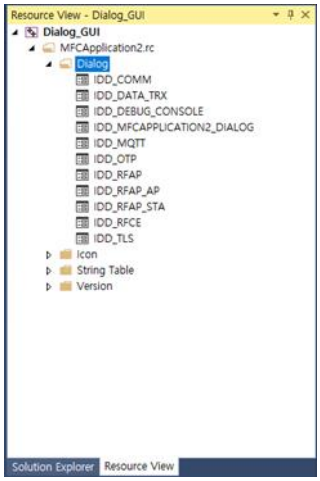



**Figure 46: Resource View Tab**

# Revision History

| Revision | Date | Description |
|---|---|---|
| 1.4 | 21-Aug-2020 | 4.3.5 Data Transfer on DPM Mode Modification<br>4.4.3 MQTT on DPM Mode Modification<br>Figure 36: MQTT Client Tab Change<br>Figure 45: Solution Explorer Tab |
| 1.3 | 06-Apr-2020 | TCP/UDP, MQTT, and TLS added, GUI Build added |
| 1.2 | 31-Oct-2019 | Finalized for publication |
| 1.1 | 15-Oct-2019 | Editorial review |
| 1.0 | 03-Jul-2019 | Preliminary DRAFT Release |

## Status Definitions

| Status | Definition |
|---|---|
| DRAFT | The content of this document is under review and subject to formal approval, which may result in modifications or additions. |
| APPROVED or unmarked | The content of this document has been approved for publication. |

## Disclaimer



# Contacting Dialog Semiconductor

United Kingdom (Headquarters)
*Dialog Semiconductor (UK) LTD*
Phone: +44 1793 757700

Germany
*Dialog Semiconductor GmbH*
Phone: +49 7021 805-0

The Netherlands
*Dialog Semiconductor B.V.*
Phone: +31 73 640 8822

Email:
enquiry@diasemi.com

North America
*Dialog Semiconductor Inc.*
Phone: +1 408 845 8500

Japan
*Dialog Semiconductor K. K.*
Phone: +81 3 5769 5100

Taiwan
*Dialog Semiconductor Taiwan*
Phone: +886 281 786 222

Web site:
www.dialog-semiconductor.com

Hong Kong
*Dialog Semiconductor Hong Kong*
Phone: +852 2607 4271

Korea
*Dialog Semiconductor Korea*
Phone: +82 2 3469 8200

China (Shenzhen)
*Dialog Semiconductor China*
Phone: +86 755 2981 3669

China (Shanghai)
*Dialog Semiconductor China*
Phone: +86 21 5424 9058