



MiWi™

MiWi™ Quick Start Guide

Introduction

This document describes prototyping, implementing, testing, and deploying wireless sensor networks using MiWi Protocol based on the Microchip software platform. This document also details how to get started with the MiWi by installing the development environment, setting up hardware, and programming devices with the reference applications.

Table of Contents

Introduction.....	1
1. Overview.....	4
1.1. MiWi v6.3 Release Content.....	4
1.2. Supported Hardware Platforms and IDEs.....	5
2. MiWi Documentation.....	6
3. Development Environment Setup.....	7
3.1. IDE Installation.....	7
3.2. Stack Configurations.....	7
3.3. Building Applications in Atmel Studio.....	8
3.4. Building Applications in IAR Embedded Workbench.....	11
3.5. Installing the Bootloader PC Tool.....	11
3.6. Installing the WiDBG Tool.....	12
4. Hardware Environment Setup.....	13
4.1. Supported Platforms and Boards.....	13
4.2. Additional Boards.....	15
4.3. Connections.....	15
5. Simple_Example_P2P Reference Application.....	17
6. Chat_Demo P2P Reference Application.....	23
7. Simple_Example_Star Reference Application.....	27
8. WSN Demo Mesh Reference Application.....	32
8.1. Launching the Demo.....	32
8.2. Network Startup.....	33
8.3. WSNMonitor.....	33
8.4. Identifying Nodes.....	35
8.5. Node Timeouts.....	35
8.6. Sensor Data Visualization.....	36
9. OTAU in WSN Demo Mesh Application.....	38
9.1. Software Prerequisites.....	38
9.2. OTAU Client.....	38
9.3. OTAU Server.....	40
10. Atmel WiDBG for OTAU.....	42
10.1. Starting the Session.....	42
10.2. Main Window.....	44
10.3. Upgrading the Node.....	50
11. Document Revision History.....	53

The Microchip Website.....	54
Product Change Notification Service.....	54
Customer Support.....	54
Microchip Devices Code Protection Feature.....	54
Legal Notice.....	55
Trademarks.....	55
Quality Management System.....	56
Worldwide Sales and Service.....	57

1. Overview

MiWi Development Environment (MiWi DE) was developed by Microchip to support a wide range of wireless applications. The backbone of MiWi DE is MiMAC and MiApp interfaces, which links the support of multiple RF transceivers and wireless communication protocols together as a well-defined, simple but robust Microchip proprietary wireless development environment.

Within MiWi DE, application developers can switch between RF transceivers and wireless protocols with little or no modification in the application layer. Such migration capability in MiWi DE reduces the firmware development risk to a very minimum level. MiWi DE is defined by three layers:

- Application layer
- Protocol layer
- RF transceiver layer

The three layers are linked together by MiMAC and MiApp interfaces. The Application layer uses MiApp interfaces to communicate to the protocol layer. In the protocol layer, there are implementations of MiWi P2P, MiWi Star, and MiWi Mesh wireless communication protocols. The drivers for Microchip RF transceivers (AT86RF233, AT86RF212B) are called by protocol layers via MiMAC interfaces. Configuration files are also presented in each layer.

Figure 1-1. MiWi™



The MiWi stack offers a significantly smaller footprint relative to the open standard based ZigBee® compliant protocol stack. This enables operation in microcontrollers with smaller memory/lower cost.

- MiWi P2P/Star – Simple Peer-to-Peer or Star network requiring minimal code size.
- MiWi Mesh – A True Routing Mesh network topology. This can be used for a department store HVAC, alarm application or a large smart home application.

1.1 MiWi v6.3 Release Content

MiWi is released as part of ASF. The main items provided as part of the MiWi release are as follows:

- Implementation of MiWi P2P and Star protocol in the form of sources and API header files. The same source is used for all MiWi P2P and Star applications.
- Implementation of MiWi Mesh protocol in the form of libraries and API header files. The same library is used for all MiWi Mesh applications.

Source code and IDE projects for reference applications:

- Simple Example P2P – MiWi P2P reference application
- Chat Demo P2P – MiWi P2P reference application
- Simple Example Star – MiWi Star reference application
- WSN Demo – MiWi Mesh reference application

Documentation:

- Quick Start Guide
- Migration Guide

- Release Notes
- Software Design Guide

1.2 Supported Hardware Platforms and IDEs

The following table lists the supported hardware platforms and IDEs for the MiWi protocol.

Table 1-1. Supported Hardware Platforms and IDEs

Microcontroller	RF Transceiver	Supported Evaluation Kit	Supported IDEs
SAMR21G18A (SIP)	RF233 (in SIP)	SAMR21 ZLLEK SAMR21 XPRO	Atmel Studio v7.0 IAR Embedded Workbench® for ARM 7.4
SAMR30G18A (SIP)	RF212B (in SIP)	SAMR30 XPRO SAMR30M XPRO	Atmel Studio v7.0

2. MiWi Documentation

This chapter provides the list of documentation available for MiWi. It is intended to help the user find required information during application evaluation and development.

ASF Documentation

- [ASF Documentation](#)
- [ASF Getting Started](#)
- [ASF Wizard](#)

MiWi Documentation/Tools on Website

- [MiWi™ P2P and Star Protocol Application Note](#)
- [AN1284 - Microchip Wireless \(MiWi™\) Application Programming Interface - MiApp](#)
- [AN1283 - Microchip Wireless \(MiWi™\) Media Access Control Interface - MiMAC](#)
- [MiWi™ Protocol Sniffer](#)

MiWi™ Documentation in ASF Package

Title	Description
Quick Start Guide	This document which helps to quickly start with MiWi™
Migration Guide	Lists the migration guidelines to use the current version of MiWi™
Release Notes	Provides information on release features and enhancements
Software Design Guide	Describes the MiWi™ applications implemented on the MiWi™ protocol

3. Development Environment Setup

This chapter provides instructions on how to set up the MiWi software package and supported IDEs. It also describes the structure of the MiWi package, and includes references to hardware setup of the supported platforms.

3.1 IDE Installation

3.1.1 Atmel Studio

The Atmel Studio can be used to develop and debug applications for AVR- and ARM-based platforms. Atmel Studio is equipped with the GCC compiler and does not require any additional external tools to compile and debug MiWi applications.

Perform the following steps to install the Atmel Studio.

1. Download and install the latest Atmel Studio version, if not already installed on your PC.
2. Add path to the folder containing the ARMGCC compiler to the Path Windows environment variable. The compiler is located in the `\Atmel\Studio\7.0\toolchain\arm\arm-gnu-toolchain\arm-none-eabi\bin` directory of the Atmel Studio installation directory.
This step is necessary for command line compilation (with makefiles).

3.1.2 IAR Embedded Workbench

The IAR Embedded Workbench for ARM can be used to develop and debug applications on ARM-based platforms. The IAR IDEs support editing of application source code, compiling source files, linking object modules with libraries and application debugging.

Perform the following steps to install the IAR Embedded Workbench.

1. Download and install IAR Embedded Workbench for ARM, if not already installed on your PC.
2. Add path to the folder containing the IAR RAM compiler to the Path Windows environment variable. The compiler is located in the `\IAR Systems\Embedded Workbench 7.4\arm\bin` directory of the IAR installation directory.
This step is necessary for command line compilation (with makefiles).

3.2 Stack Configurations

MiWi uses the configuration files to regulate the behavior of the stack.

The following are the locations of header files to configure the behavior of P2P application:

- `thirdparty\wireless\miwi\apps\simple_example_p2p\miwi_config.h`
- `thirdparty\wireless\miwi\apps\simple_example_p2p\miwi_config_p2p.h`

The following are the locations of header files to configure the behavior of Star application:

- `thirdparty\wireless\miwi\apps\simple_example_star\miwi_config.h`
- `thirdparty\wireless\miwi\apps\simple_example_star\miwi_config_p2p.h`

The following are the locations of header files to configure the behavior of Mesh application:

- `thirdparty\wireless\miwi\apps\wsn_demo\miwi_config.h`
- `thirdparty\wireless\miwi\apps\wsn_demo\miwi_config_mesh.h`

3.3 Building Applications in Atmel Studio

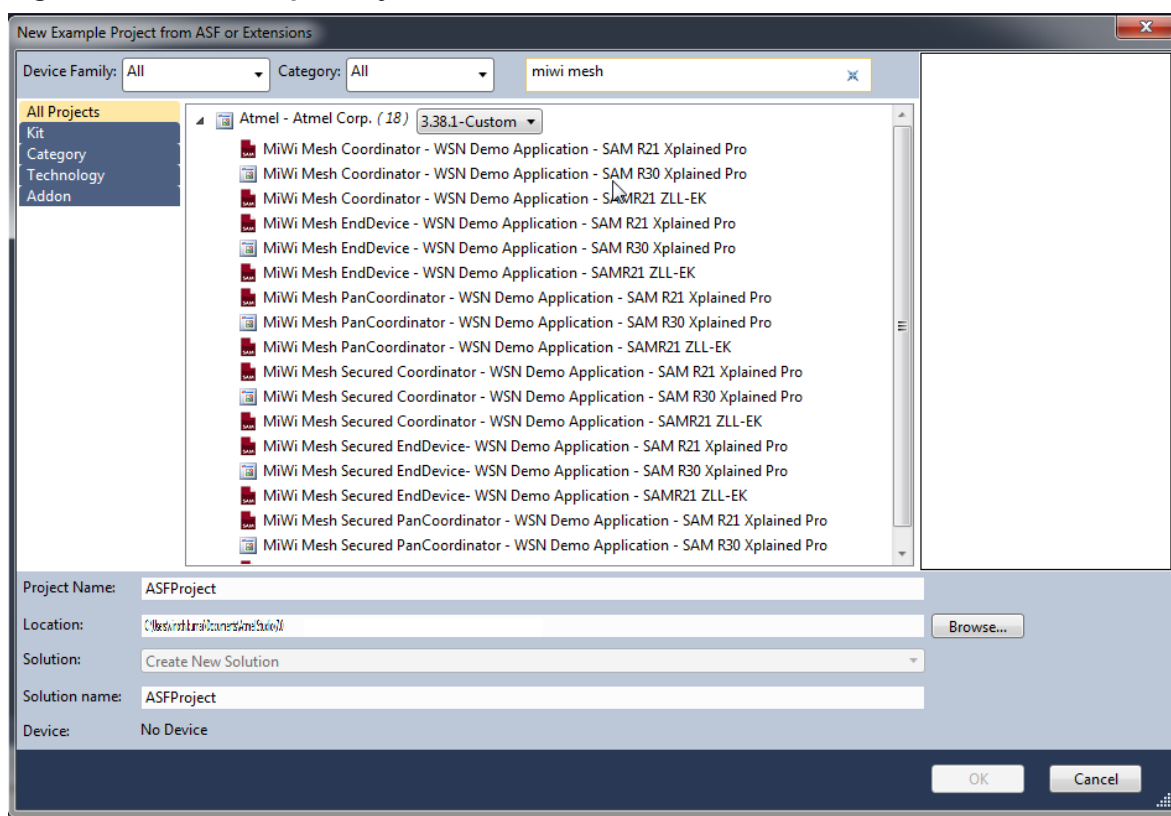
Atmel Studio can be used to develop and build MiWi applications. Reference applications include Atmel Studio project files located in the \as5_arm subdirectory of the application root directory. These projects depend on the configurations provided by the external low-level makefiles.

3.3.1 Opening a Project from ASF

Perform the following steps to open a project from ASF.

1. Open Atmel Studio.
2. Choose *File>New>Example Project...* in Atmel Studio.
3. In the New Example Project from ASF or Extensions window search for "MiWi Mesh" to list the available projects in MiWi Mesh as shown in the following screenshot.

Figure 3-1. New Example Project from ASF or Extensions Window⁽¹⁾



Note:

(1) Use the latest version of ASF

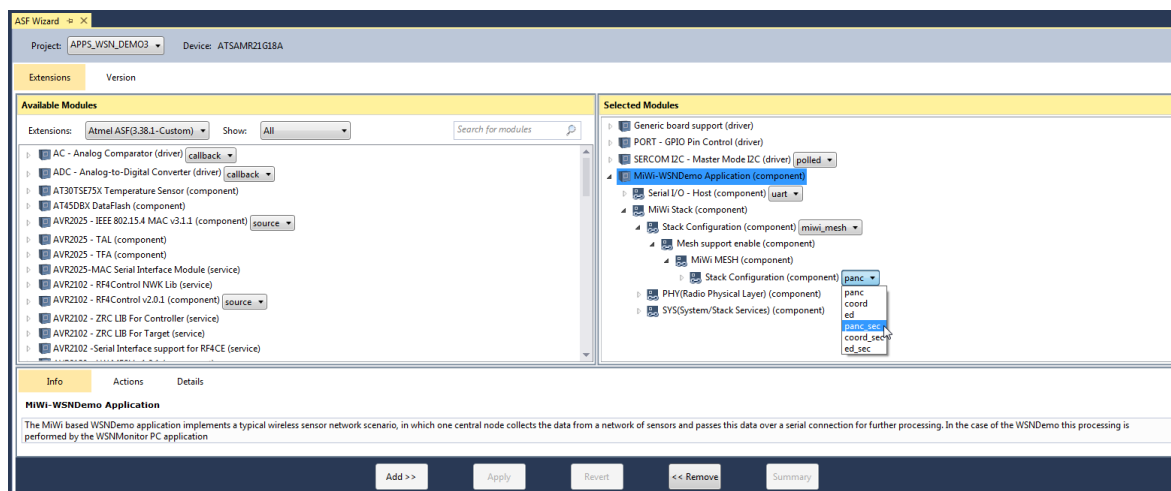
4. Select **MiWi Mesh Coordinator - WSN Demo Application** project.
5. Click **OK** and accept the license agreement by checking the "I accept the license agreement" checkbox.
6. Click **Finish**.
The MiWi Mesh Coordinator - WSN Demo Application project is created and loaded in Atmel Studio.

3.3.2 Switching/Changing the Configuration using ASF Wizard

If the user is using the MiWi Mesh Coordinator and needs to switch to the MiWi Mesh PANC Security, then perform the following steps.

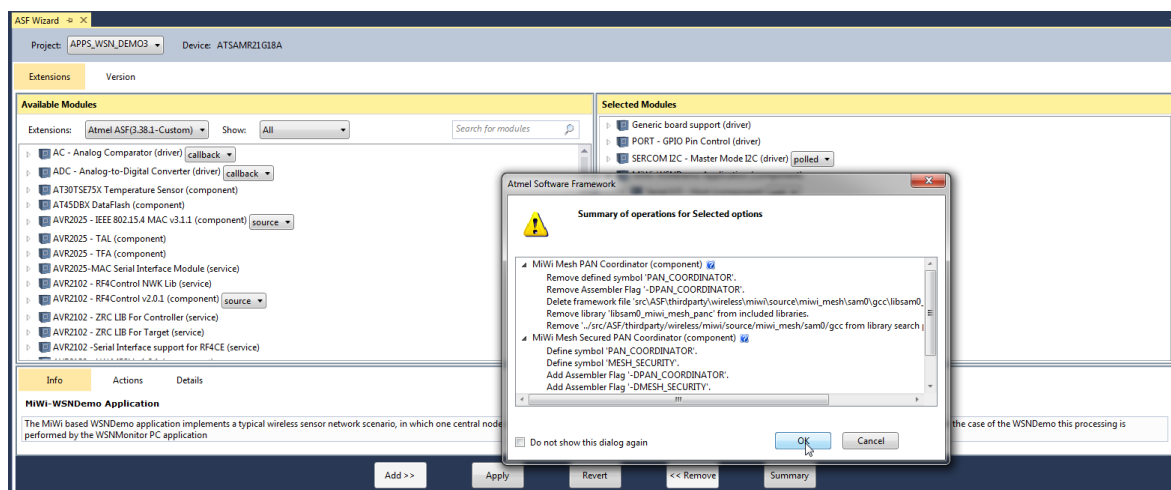
1. Choose **ASF>ASF Wizard** to open the ASF Wizard window.
2. In the Selected Modules pane, expand MiWi-WSNDemo Application (component) and choose **panc_sec** from the drop-down list as shown in the following screenshot.

Figure 3-2. ASF Wizard



3. Click **Summary** to understand the component changes.
4. Click **Apply** to apply the changes to the project.

Figure 3-3. ASF Wizard-Summary of Operations for Selected Option

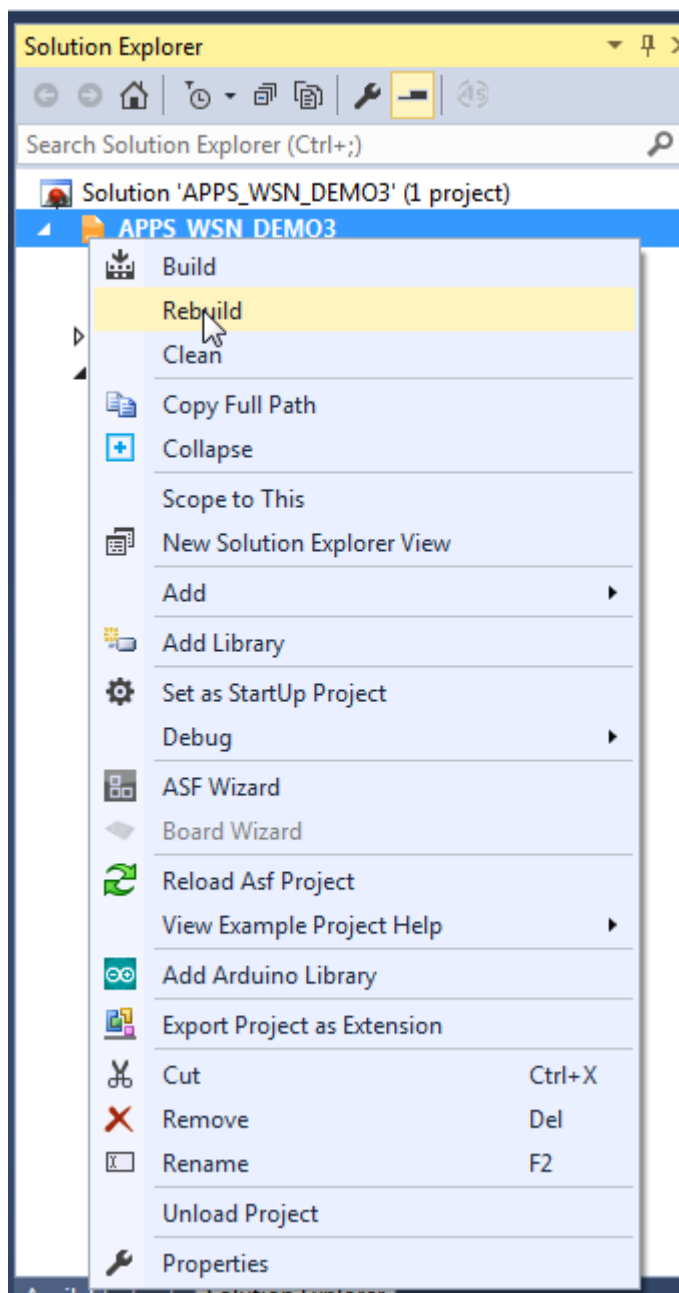


3.3.3 Building the Application using IDE

Perform the following steps to build the application using IDE.

1. Open an appropriate `.ats1n` project file from the `<appName>\..\as5_arm` directory with Atmel Studio. The Solution Explorer tab provides access to the application source files and stack components that compile together with the application.

Figure 3-4. Example Structure of Atmel Studio Application Project



2. From the main menu, choose **Build>Rebuild All**.
When the build process is complete, some of the .hex, .srec, .bin, and .elf image files are generated based on the selected platform configuration. Use the .hex file for programming the devices via EDBG. The .elf file is used for debugging.

3.3.4 Building the Application using Command Line Interface

Compile the application by running the make utility command to build the application.

```
make clean all
```

It is possible to run the make utility command from Atmel Studio by selecting **Tools>Command Prompt**. This ensures that the make utility provided with Atmel Studio is used. Otherwise, the path to the folder

containing the make utility can be added to the path environment variable. In this case, run the make utility in the command line from the application's root directory.

3.4 Building Applications in IAR Embedded Workbench

IAR Embedded Workbench can be used to develop and build MiWi applications. All reference applications include IAR project files located in the `\iar` sub-directory of the application root directory.

A part of stack components and drivers are compiled with the application. The source files for these components are included in the IAR projects, therefore they are an effective part of the application.

For compilation from the command line with the IAR compiler, `make` files are used.

3.4.1 Building the Application using IDE in IAR Embedded Workbench

Perform the following steps to build the application using IDE.

1. Open the `.eww` file in the `iar_projects` sub-directory of the appropriate application directory with IAR Embedded Workbench. For example, for WSN Demo, use the `APP_WSN_Demo.eww` file from the `apps\WSN_Demo\..\iar` sub directory.
2. From the main menu, choose **Build>Rebuild All**.
By default, the `.a90` file (for WSN Demo, `APP_WSN_Demo.a90`) is generated in the `\iar\Debug\exe` sub directory (for WSN Demo, in the `apps\WSN_Demo\iar\Debug\exe` directory) with format as specified in Linker Output Options of the IAR project.

3.4.2 Building the Application using CLI in IAR Embedded Workbench

Compile the application by running the make utility command to build the application.

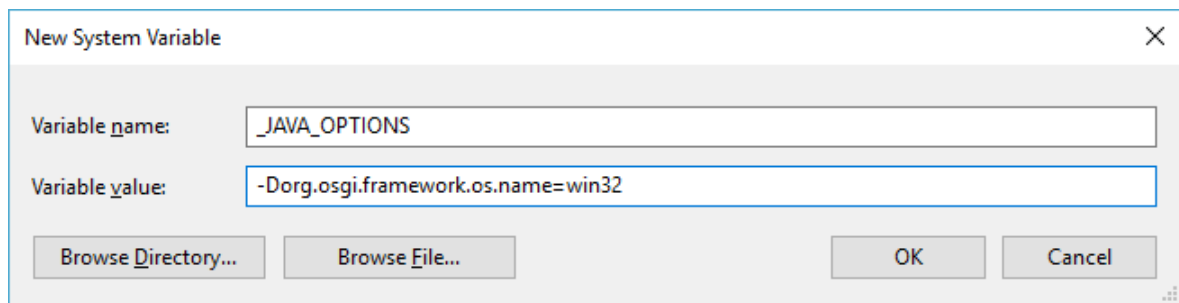
```
make clean all
```

Some of the `.hex`, `.srec`, `.bin`, and `.elf` image files are generated depending on the selected platform configuration.

3.5 Installing the Bootloader PC Tool

Run the bootloader PC tool installer with the following instructions.

1. Navigate to the Bootloader PC Tool, available in the `<ProjectDir>\thirdparty\wireless\miwi\services\otau\tools` directory.
2. Double click the `Bootloader_PC_Tool_Setup.exe` file.
3. Click **Next** in all the pop-up windows.
4. Change the installation path if required, and click **Next**.
5. Click **Done** to finish the installation.
For a Windows 10 PC, add the following in System Environment Variables.

Figure 3-5. Adding System Environment Variable

3.6 Installing the WiDBG Tool

Run the WiDBGSetup with the following instructions.

1. Navigate to the WiDBGSetup, available in the <ProjectDir>\thirdparty\wireless\miwi\services\otau\tools directory.
2. Double click the WiDBGSetup.msi file.
3. Accept the terms in the License Agreement and click **Install**.
4. Click **Finish** to complete the installation.

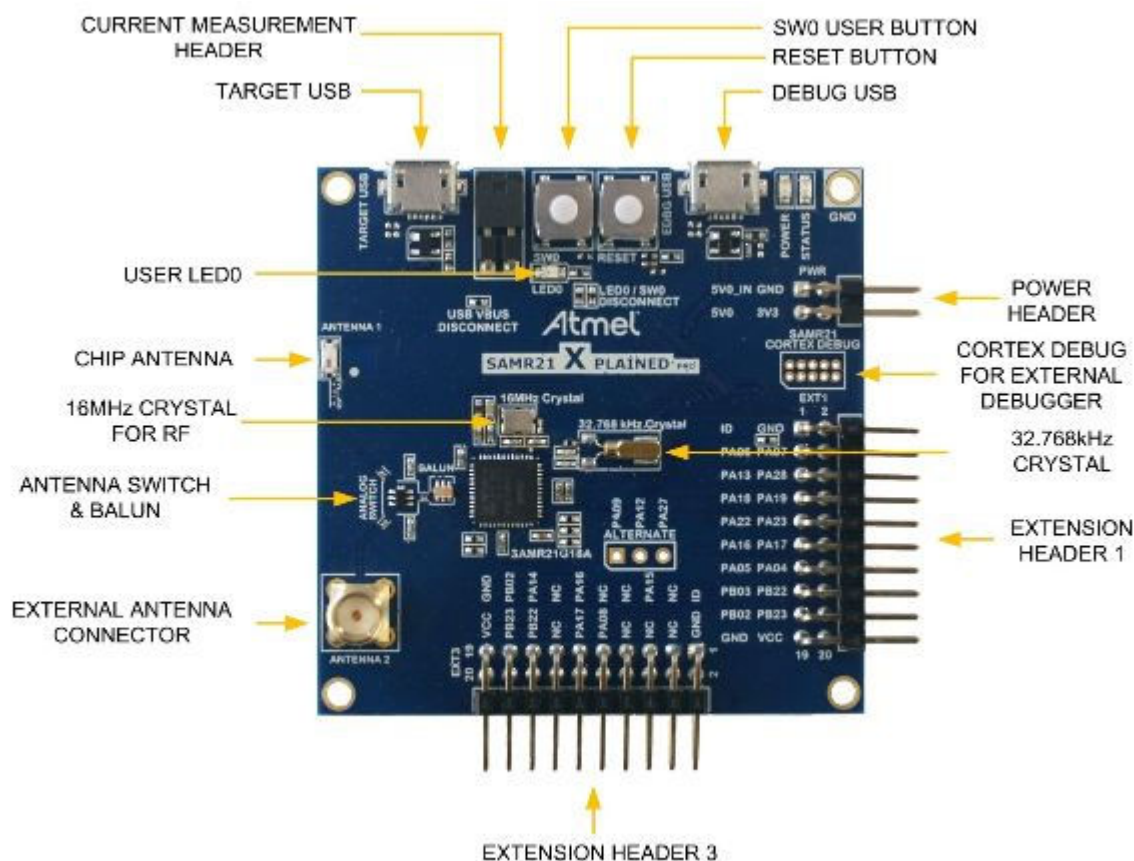
4. Hardware Environment Setup

4.1 Supported Platforms and Boards

The following boards are used/supported in the 6.1 release.

1. SAM R21 Xplained Pro

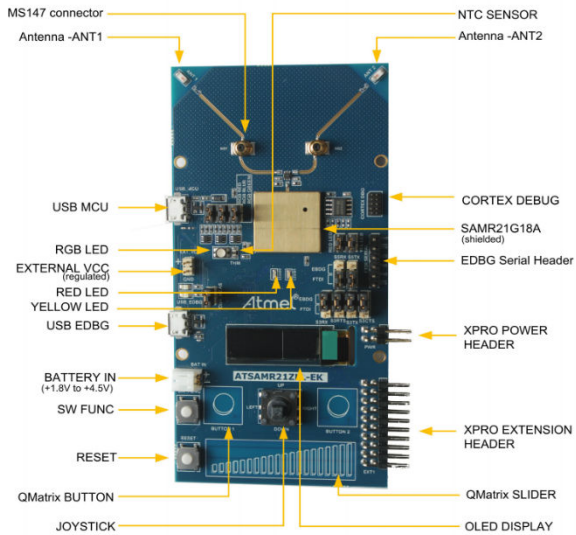
Figure 4-1. SAM R21 Xplained Pro



For more details, see [SAMR21 Xplained Pro User Guide](#).

2. SAM R21 ZLLEK

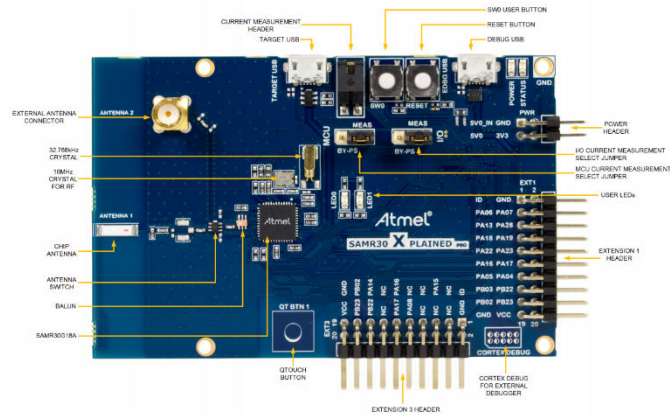
Figure 4-2. SAM R21 ZLL-EK



For more details, see [SAM R21 ZLL-EK User Guide](#).

3. SAM R30 Xplained Pro

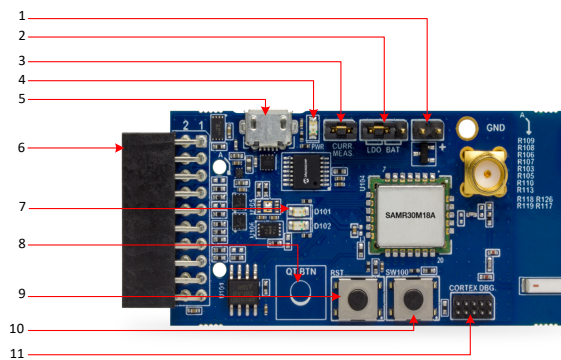
Figure 4-3. SAM R30 Xplained Pro



For more details, see [SAM R30 Xplained Pro User Guide](#).

4. SAMR30M Xplained Pro

Figure 4-4. SAMR30M Xplained Pro



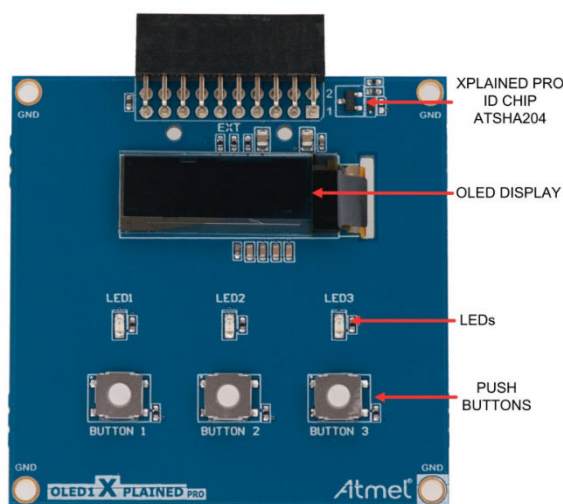
1. Battery header (J102)

2. Battery/LDO Selection header (J103)
3. Current measurement header (J104)
4. Power LED
5. USB header
6. XPRO Extension header
7. User LEDs
8. QTouch® button
9. Reset button
10. User button
11. Cortex DBG (Debug) header

4.2 Additional Boards

For Simple_Example_P2P and Simple_Example_Star applications, OLED1 Xplained PRO can be used as an additional board to show the full-featured demo on SAM R21 XPRO and SAM R30 XPRO. This board must be connected as Extension board 1.

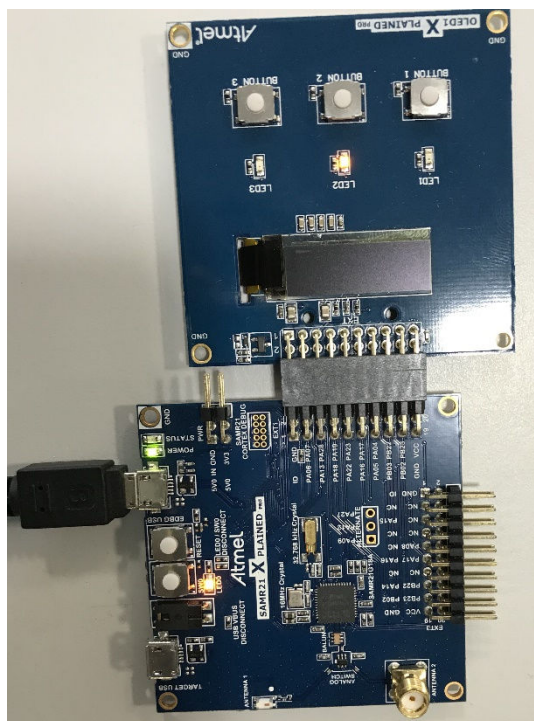
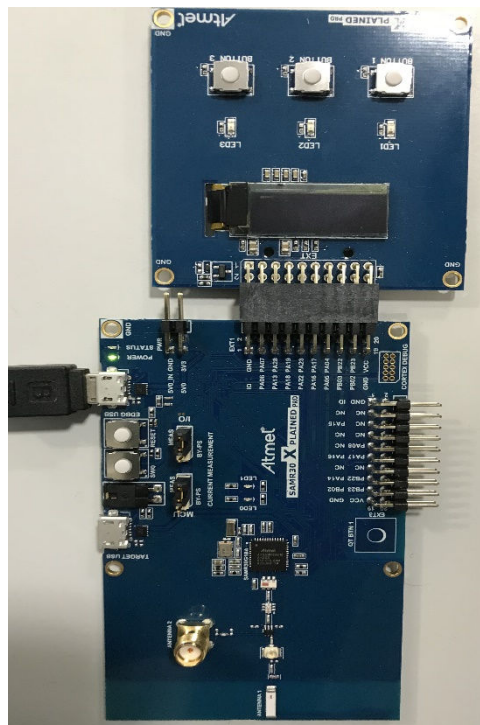
Figure 4-5. OLED1 Xplained Pro



For more details, see [OLED1 Xplained Pro User Guide](#).

4.3 Connections

Connect the Micro USB cable to the 'Debug USB/EDBG USB' port of the board and connect the other end to the PC.

Figure 4-6. Connection of SAM R21 XPRO and OLED1 XPRO**Figure 4-7. Connection of SAM R30 XPRO and OLED1 XPRO**

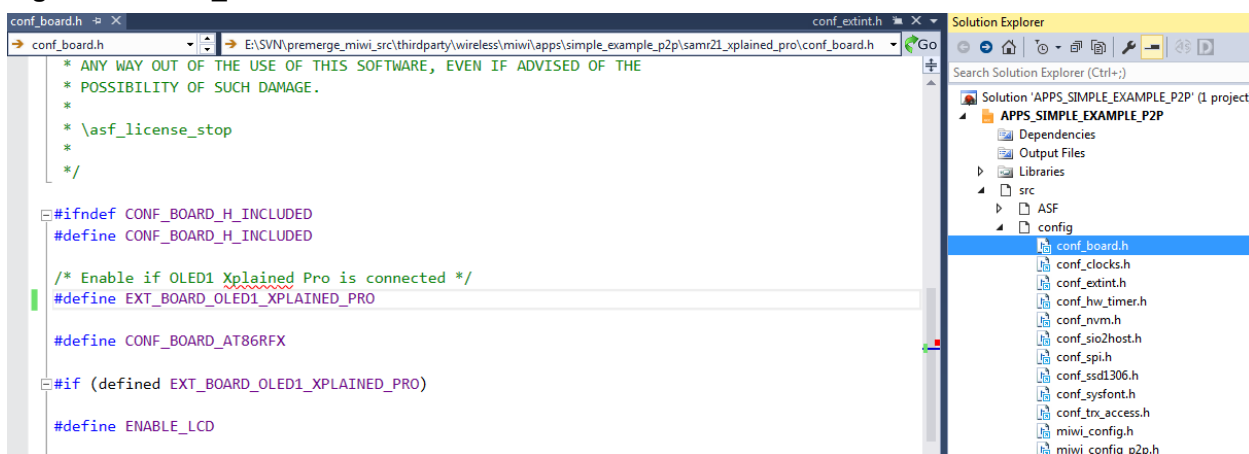
5. Simple_Example_P2P Reference Application

The simple example application code focuses on the simplicity of the MiWi DE protocol stack application programming interfaces. It provides wireless communication between two devices with less than 30 lines of C code to run the stack in the application layer for both devices. In this application, the following features of MiWi DE protocol stack are demonstrated:

- Establish a connection automatically between two devices
- Broadcast a packet
- Unicast a packet
- Apply security to the transmitted packet

To run full-featured Simple_Example_P2P application on SAMR21 XPRO or SAMR30 XPRO, the user needs to connect OLED1 Xplained PRO. When using OLED1 Xplained PRO along with SAMR21 XPRO or SAMR30 XPRO, the user needs to enable the macro `#define EXT_BOARD_OLED1_XPLAINED_PRO` in `conf_board.h` file.

Figure 5-1. conf_board.h File



To run the simple example application, perform the following instructions.

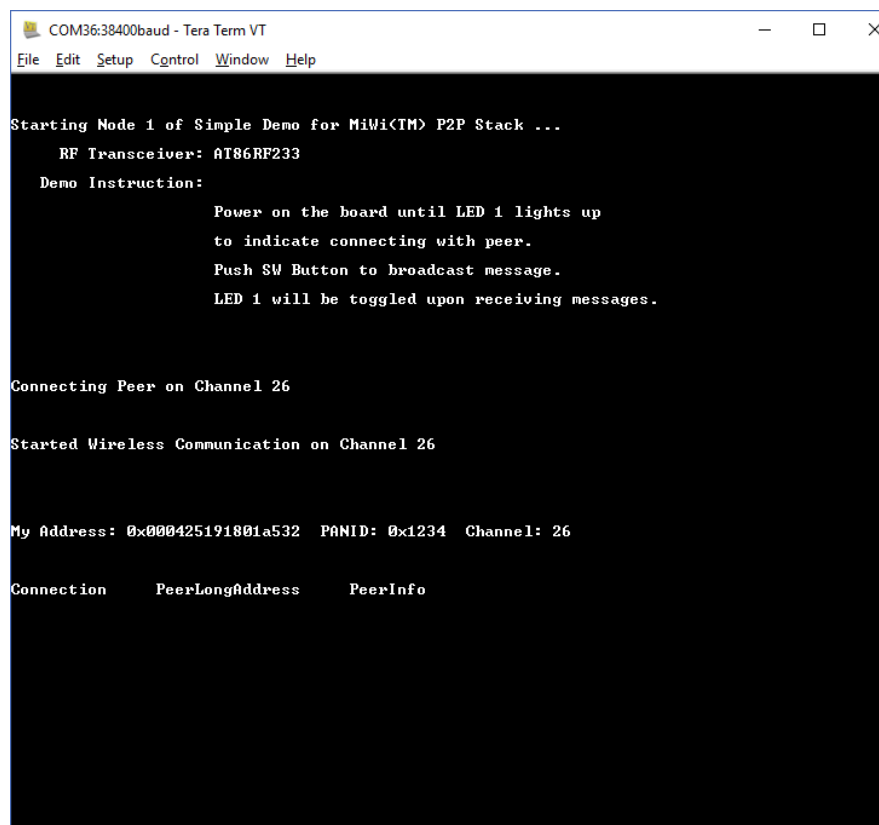
1. Program node 1 and node 2 with Simple_Example_P2P application firmware.
2. Power on node 1 and node 2, respectively.
3. Wait for few seconds, until the first LED on both nodes are powered-up. This indicates that a connection is established automatically.
 - For the details of connection establishment, refer to section “Variations for Handshaking” in Application Note AN1204 Microchip MiWi™ P2P Wireless Protocol if MiWi P2P protocol is used. If MiWi protocol is used, refer to section “MAC Function Description” in IEEE® 802.15.4 specification.
 - If the demo is running on the SAMR21 ZLLEK (or) SAMR21 XPRO with the ATOLED1-XPRO (or) SAMR30 XPRO with the ATOLED1-XPRO, critical information is shown on the LCD of the demo board. It first shows the demo name, RF transceiver and node number, then connecting information and channel information is displayed before the demo instruction.

Table 5-1. LCD Display

Text on LCD Display	Description
Simple P2P Demo on SAMR21 Node	After powering ON the boards, this splash screen message is displayed on the LCD screen.
Started Wireless Communication on Channel 26	The device did not find any network to join, started new network on channel 26.
Connecting Peer on Channel 26	Connecting to a network if found.
Connected Peer on Channel 26	Connected to a network. This log does not show for the first device since it starts the network when it is unable to find network to join.
SAM R21 ZLL EK • SW FUNC : Broadcast • BUTTOB1 : Unicast SAM R21 XPRO and OLED1XPRO (or) SAM R30 XPRO and OLED1 XPRO • SW : Broadcast • BUTTOB1 : Unicast	Demo instruction is displayed.

- If a hyper terminal is opened to monitor firmware output, the user can see the peer device information printed out from both the nodes.

Figure 5-2. Monitor Firmware Output 1



```
COM36:38400baud - Tera Term VT
File Edit Setup Control Window Help

Starting Node 1 of Simple Demo for MiWi(TM) P2P Stack ...
  RF Transceiver: AT86RF233
  Demo Instruction:
    Power on the board until LED 1 lights up
    to indicate connecting with peer.
    Push SW Button to broadcast message.
    LED 1 will be toggled upon receiving messages.

Connecting Peer on Channel 26

Started Wireless Communication on Channel 26

My Address: 0x000425191801a532  PANID: 0x1234  Channel: 26

Connection    PeerLongAddress    PeerInfo
```

Figure 5-3. Monitor Firmware Output 2

```

COM7:38400baud - Tera Term VT
File Edit Setup Control Window Help

Starting Node 1 of Simple Demo for MiWi(TM) P2P Stack ...
RF Transceiver: AT86RF233
Demo Instruction:
    Power on the board until LED 1 lights up
    to indicate connecting with peer.
    Push SW Button to broadcast message.
    Press Joystick CENTER Button to unicast encrypted message.
    LED 1 will be toggled upon receiving messages.

Connecting Peer on Channel 26

Connected Peer on Channel 26

My Address: 0x000425191801b742 PANID: 0x1234 Channel: 26

Connection    PeerLongAddress    PeerInfo
00            00042519180186bc    09

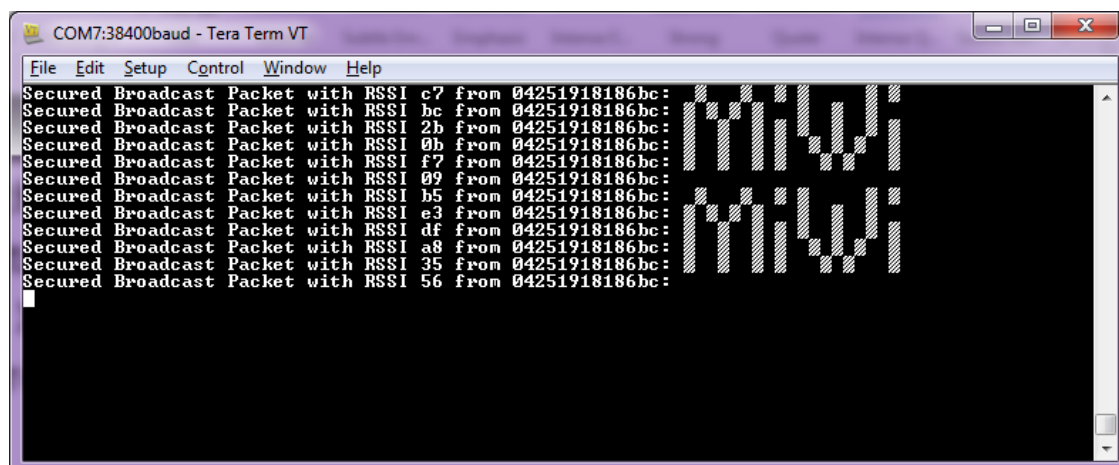
```

4. Press SW/SW FUNC on one node to toggle the second LED on the other node.
 - This shows how a broadcast packet is transmitted.
 - If the demo is running on SAM R21 ZLLEK (or) SAM R21 XPRO with ATOLED1-XPRO (or) SAMR30 XPRO with ATOLED1-XPRO, the total number of transmitted and received messages will be shown on the LCD.

Text on LCD Display	Description
TX Messages:0 RX Messages: 2	Total number of transmitted and received messages

- If a hyper terminal is used, on the receiving end (the device that has LED2 toggled) the user can see the printout of the broadcast packet source address, signal strength, and the packet payload. The packet payload is the one line of bitmap of MiWi. Pressing the SW button at an interval of two seconds between each press on one end displays the complete bitmap of MiWi.

Figure 5-4. Bit Map of MiWi™



5. For unicast setup:

- For SAMR21ZLLEK,

1. Pressing JOYSTICK center button on one node:

Text on LCD Display	Description
UP : 00-42b701 DOWN: Change node	Pressing JOYSTICK center button on one node

2. Pressing Joystick DOWN button selects the next device for unicast if available.

3. Pressing Joystick UP button sends unicast to the selected device and toggles the second LED on the other node.

- For SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO,

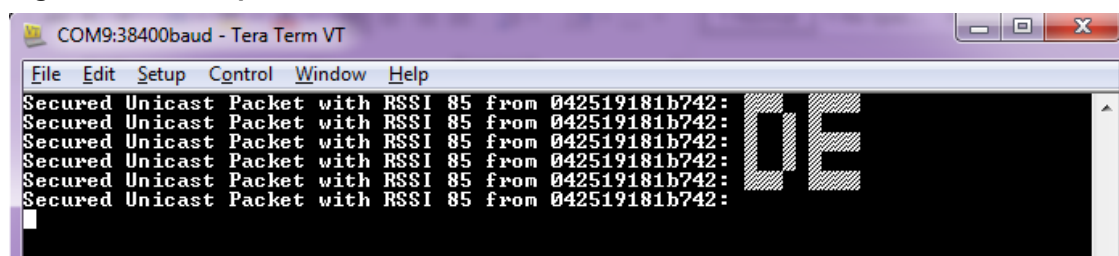
1. Pressing BUTTON1 center on one node:

Text on LCD Display	Description
SW0 : Unicast BUTTON: Next Node	Pressing BUTTON1 center button on one node

2. Pressing BUTTON1 selects the next device for unicast if available.

3. Pressing SW0 button sends unicast to the selected device.

- This shows how an encrypted unicast packet is transmitted and decrypted by the radio after it is received. For more details of how MiWi P2P handles encryption, refer to section "Security Features" in the *AN1204 Microchip MiWi P2P Wireless Protocol Application Note*.
- If the demo runs on SAM R21 ZLLEK (or) SAM R21 XPRO with ATOLED1-XPRO (or) SAMR30 XPRO with ATOLED1-XPRO, the total number of transmitted and received messages are shown on the LCD.
- If hyper terminal is used, on the receiving end (the device that has LED2 toggled), the user can see the printout of the secured unicast packet source address, signal strength, and the packet payload. The packet payload must be decrypted by the receiving device. The packet payload is the one line of bit map of "DE". Sending unicasts continuously at an interval of two seconds between each unicast/button press on one end displays the complete bit map of "DE".

Figure 5-5. Bit Map of DE

6. By default, Network Freezer is enabled in the application. The Network Freezer feature is used to store critical network information into the non-volatile memory. When the device reboots or power is rebooted, it restores the persistent data from the non-volatile memory and continues to operate in the same network with the existing network information.
- If the power is rebooted on a node, the following description may be used to restore the persistent data if available in the node using the Network Freezer.

- After Step 1, the following message is shown on the LCD. The user must press the SW button within 5 seconds to boot the application from stored network parameters, or the user can press and hold the SW button immediately after power reboot till the LCD shows the following message.

Text on LCD Display
SW: Use Nwk Freezer Press in 5 sec

- If the SW button is pressed within the timeout, the application restores the data from Network Freezer and the following message is shown on LCD.

Text on LCD Display
Restoring Network !!

6. Chat_Demo P2P Reference Application

The chat demoP2P application code focuses on the simplicity of the MiWi DE protocol stack application programming interfaces. It provides wireless communication between two devices with less than 30 lines of C code to run the stack in the application layer for both devices. In this application, the following features of MiWi DE protocol stack are demonstrated:

- Establish a connection automatically between two devices.
- Unicast a packet.
- Apply security to the transmitted packet.

To run the chat demo application, do the following.

1. Program node 1 and node 2 with proper firmware.
2. Power on node 1 and node 2 respectively. A hyper terminal must be opened to monitor firmware output.
3. Wait for few seconds, until the first LED on both nodes are powered-up.
4. The user can see the peer device information printed from both the nodes.

Figure 6-1. Node 1

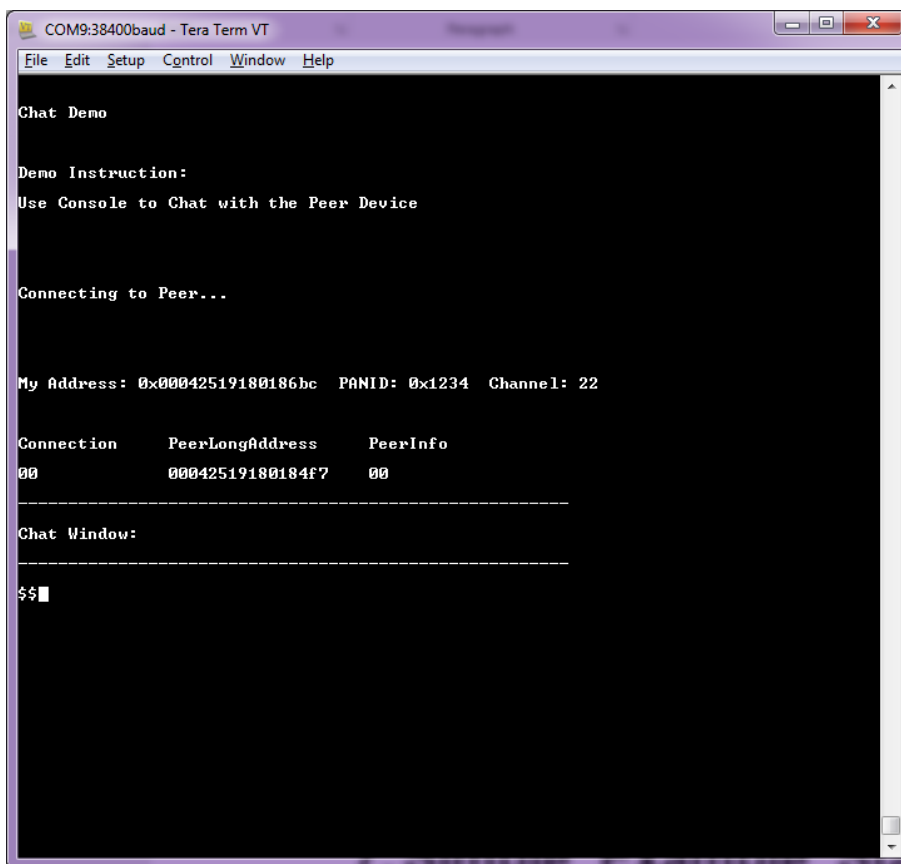
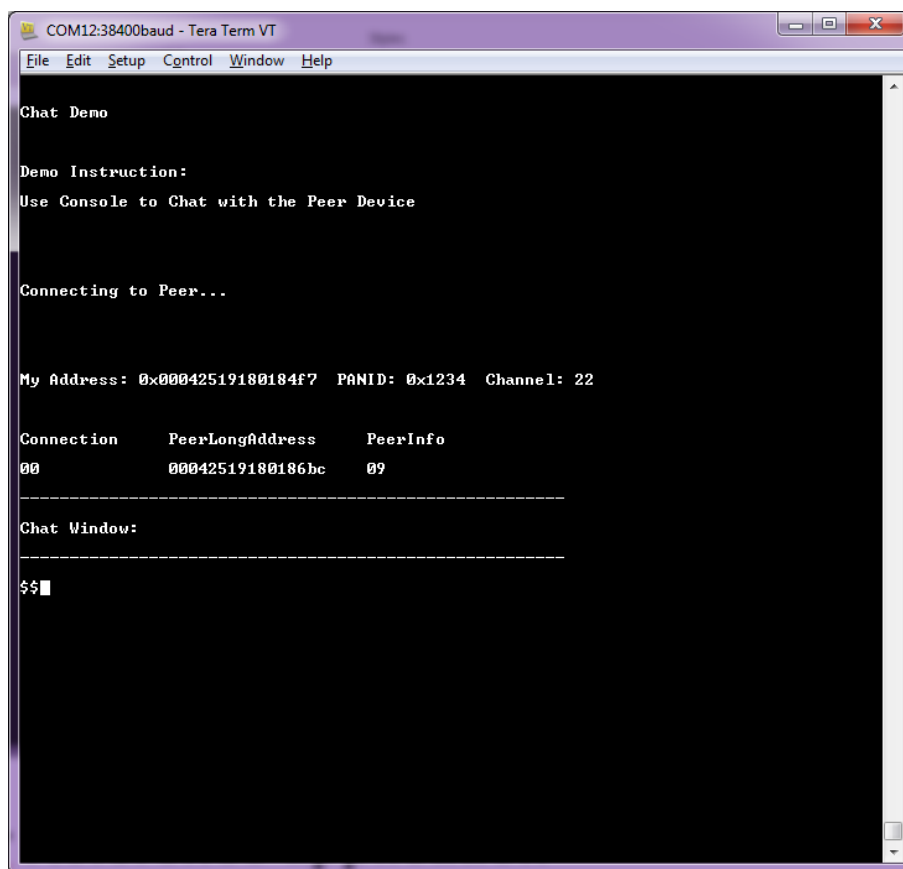
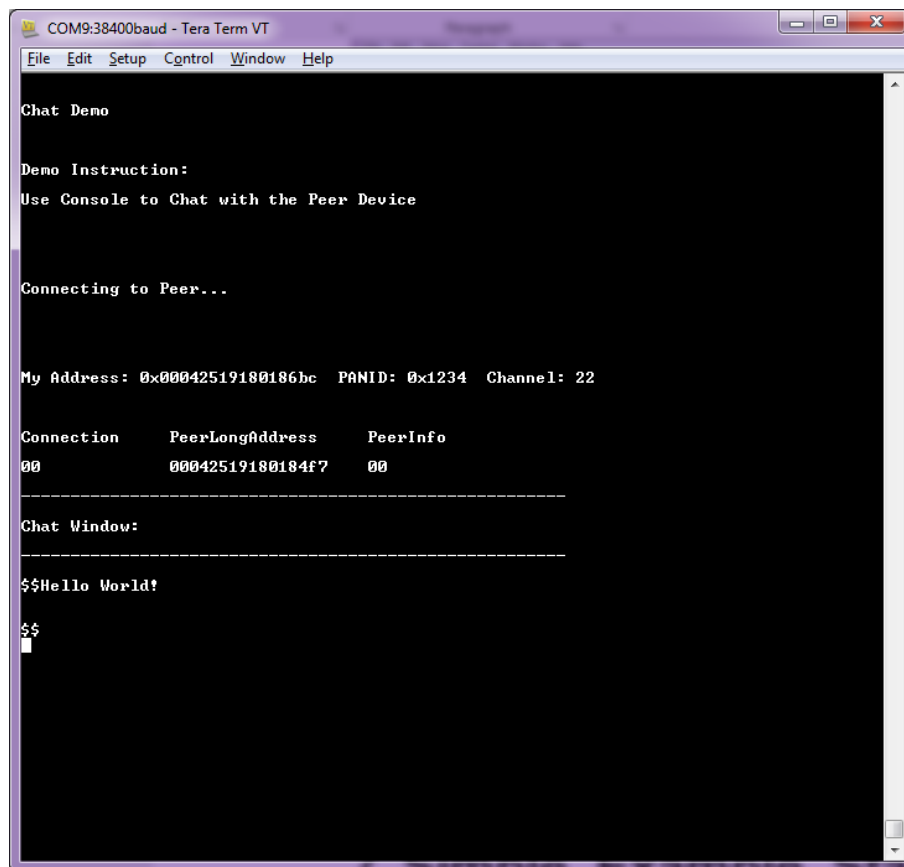


Figure 6-2. Node 2



5. When the devices are connected, start typing and press Enter. For example, type 'Hello World!' on node 1 and press Enter. The user can see the entered text on node 2 as follows.

Figure 6-3. Node 1



COM9:38400baud - Tera Term VT

File Edit Setup Control Window Help

Chat Demo

Demo Instruction:
Use Console to Chat with the Peer Device

Connecting to Peer...

My Address: 0x00042519180186bc PANID: 0x1234 Channel: 22

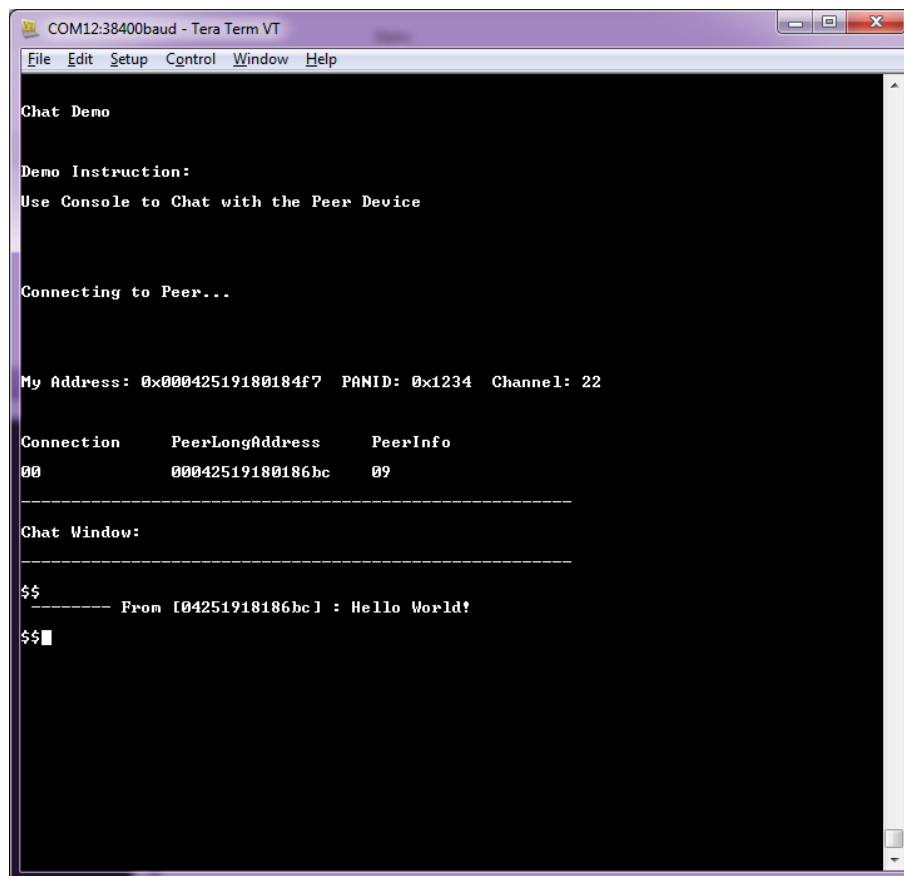
Connection	PeerLongAddress	PeerInfo
00	00042519180184f7	00

Chat Window:

\$\$Hello World!

\$\$

Figure 6-4. Node 2



The screenshot shows a Tera Term VT window titled "COM12:38400baud - Tera Term VT". The window contains the following text:

```
Chat Demo

Demo Instruction:
Use Console to Chat with the Peer Device

Connecting to Peer...

My Address: 0x00042519180184f7 PANID: 0x1234 Channel: 22

Connection      PeerLongAddress      PeerInfo
00              00042519180186bc      09
-----

Chat Window:
-----

$$
----- From [04251918186bc] : Hello World!
$$
```

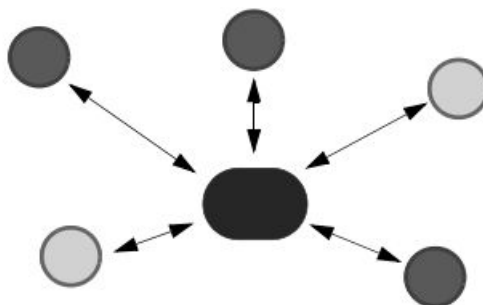
7. Simple_Example_Star Reference Application

The simple example application code focuses on the simplicity of the MiWi DE protocol stack application programming interfaces. It provides wireless communication between two devices with less than 30 lines of C code to run the stack in the application layer for both devices. In this application, the following features of MiWi DE protocol stack are demonstrated:

- Establish a connection automatically between PAN coordinator and end device.
- Broadcast a packet.
- Unicast a packet from one end device to another end device through PAN coordinator.
- Apply security to the transmitted packet.

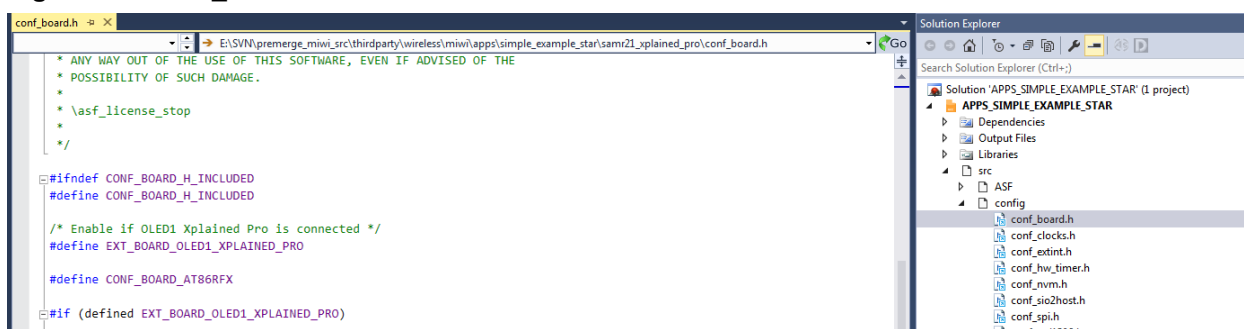
By default, the Network Freezer is enabled in the application. The Network Freezer feature is used to store critical network information into the nonvolatile memory. Star network is supported on three different Hardware Development Boards and works as described in the following configurations.

Figure 7-1. Star Network Configuration



To run full-featured Simple_Example_Star application on the SAMR21 XPRO or SAMR30 XPRO, the user needs to connect the OLED1 Xplained PRO. When the OLED1 Xplained PRO is used along with the SAMR21 XPRO or SAMR30 XPRO, then the user needs to enable the macro `#define EXT_BOARD_OLED1_XPLAINED_PRO` in `conf_board.h` file.

Figure 7-2. conf_board.h File



1. Perform the following.

Table 7-1. LCD Display

Text on LCD Display	Description
Simple STAR on SAMR21 Node	After powering ON the boards, this splash screen message is displayed on the LCD screen.

.....continued

Text on LCD Display	Description
Started Wireless Communication on Channel 25	The device did not find any network to join, started new network on channel 25.
Connecting Peer on Channel 25	The LCD screen displays the operating channel.
Connected Peer on Channel 26	Connected to a network. This log does not show for first device since it starts network when it is unable to find network to join.
SAM R21 ZLL EK • PC : SW FUNC to Broadcast SAM R21 XPRO and OLED1XPRO (or) SAM R30 XPRO and OLED1 XPRO • PC : SW to Broadcast	If no network is found, the node creates its own network and acts as a MiWi™ PAN coordinator.
Connected Peer on Channel 25	If any other MiWi™ PANCO node is found in the vicinity, then it gets connected to the PAN coordinator as End Node.

Hyper terminal shows the following on successful connection.

Figure 7-3. Node Starting the Network (PANC)

```

COM36:38400baud - Tera Term VT
File Edit Setup Control Window Help

Starting Node 1 of Simple Demo for MiWi(TM) STAR Stack ...
RF Transceiver: AT86RF233
Demo Instruction:
    Power on the board until LED 1 lights up
    to indicate connecting with peer.
    Press SW Button to broadcast message.
    LED 1 will be toggled upon receiving messages.

Connecting Peer on Channel 25

Started Wireless Communication on Channel 25

My Address: 0x000425191801a532 PANID: 0x1234 Channel: 25

Connection PeerLongAddress PeerInfo

```

Figure 7-4. Node Joining the Network (PANC)

```

COM9:38400baud - Tera Term VT
File Edit Setup Control Window Help

Starting Node 1 of Simple Demo for MiWi(TM) STAR Stack ...
RF Transceiver: AT86RF233
Demo Instruction:
    Power on the board until LED 1 lights up
    to indicate connecting with peer.
    Press SW Button to broadcast message.
    LED 1 will be toggled upon receiving messages.

Connecting Peer on Channel 25
Connected Peer on Channel 25

My Address: 0x00042519180186bc PANID: 0x1234 Channel: 25

Connection PeerLongAddress PeerInfo
00          000425191801b742 01

```

Note: For every 15 seconds, PAN CO broadcasts the connection table to all the end nodes and, the end nodes send the link status back to PAN CO. After the PAN coordinator has established a network, power on a second node and follow the instructions in Step 5 to join the PAN Coordinator. This process may be repeated to add any number of Nodes to the network.

2. After getting connected to PAN Coordinator, the LCD displays options to unicast a message to either the PAN coordinator or to another node in the network.

2.1. For SAMR21ZLLEK:

Text on LCD Display	Description
Press JoyStick Center to Unicast	Pressing joystick center button on one node
UP:00-42b701 me DOWN: Change node	Pressing joystick UP button sends the unicast message

2.2. For SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO:

Text on LCD Display	Description
SW0 : Unicast BUTTON: Next Node	Pressing BUTTON1 displays the next node to choose
SW0:02-bc8601-me BUTTON1: Change node	Pressing SW0 button sends unicast to the selected device.

3. Pressing joystick DOWN push button on the node on the SAMR21ZLLEK (or) pressing BUTTON 1 on the SAMR21XPRO+OLED1XPRO or SAMR30XPRO+OLED1XPRO, displays the address of the next node in the unicast address selection list.

The LCD screen displays the three bytes of the short address followed by “me” keyword indicating its MAC address in the list, or MAC address of the next node in the unicast address selection list. Depending on the location within the list, the LCD screen shows one of the following texts:

Text on LCD Display	Description
UP:00-42b701 me DOWN: Change node	SAM R21 ZLL EK
SW0:02-bc8601-me BUTTON1: Change node	SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO

4. If joystick UP push button on the SAMR21ZLLEK
(or)

SW0 button on the SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO is pressed at the end node, a unicast message is sent to,

- the PAN Coordinator when indicated as, "xx-xxxxxx-me"
(or)
- the destination node when indicated as, "xx-xxxxxx".

After a successful transmission, the TX value is incremented at the source end node. The RX value is incremented at the destination node (PAN CO) and three bytes of source MAC addresses are displayed.

Text on LCD Display	Description
TX Messages: 2 RX Messages: 4	In the nodes, LCD displays the number of transmitted and received messages.
Data Packet from Address:f78401	After one second, the LCD display shows this message.

5. If Joystick DOWN push button on the SAMR21ZLLEK or BUTTON 1 on the SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO is pressed at the end node, the LCD displays the next node available in the connection table.

UP: 01-XXXXXX

DOWN: Change Node

Note: At the end nodes, UP push button on the SAMR21ZLLEK or SW0 button on the SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO is used to unicast message to the selected node. DOWN push button on the SAMR21ZLLEK or BUTTON 1 on the SAMR21XPRO and OLED1XPRO (or) SAMR30XPRO and OLED1XPRO is used to change or select the other destination node of the unicast message.

6. Press SW button at PAN CO to broadcast message to all the end nodes in the network. The LCD displays the incremented TX values.

TX: xx , RX: yy

Message Count

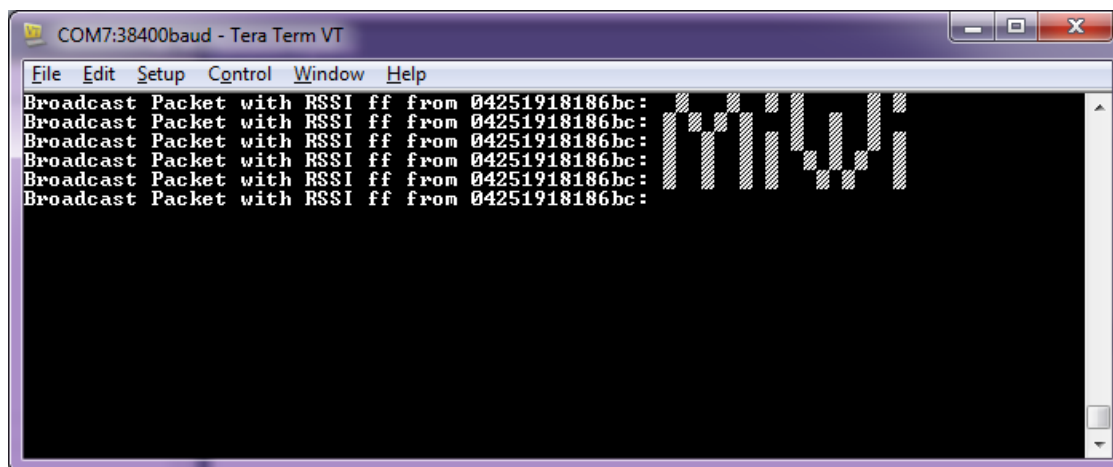
Note: When the destination node receives the message from source (end nodes or PAN CO), the RX value is incremented. The respective nodes display the three bytes of source MAC address from which they received the message.

7. After few seconds, the display reverts to following messages on the LCD display.

Text on LCD Display	
PC : SW FUNC to Broadcast	At PAN CO
Press Joystick Center to Unicast	At source or destination end nodes

Hyper terminal shows the following upon many broadcasts from the PAN coordinator.

Figure 7-5. Hyper Terminal Output



8. When the device reboots or power is rebooted, it tries to restore the existing data from the nonvolatile memory and continues to operate in the same network with existing network information.

After a reboot, perform the following steps using the Network Freezer to restore the existing data in the node:

- After Step 1, the LCD displays (SW: Use Nwk Freezer Press in 5 sec) message.
Press the SW button within 5 seconds to boot the application from stored network parameters, or the user can press and hold the SW button immediately after the power reboot till the LCD displays the following message.

Text on LCD Display
SW: Use Nwk Freezer Press in 5 sec

- If the SW button is pressed within the timeout, the application restores the data from the Network Freezer and the LCD displays the following message.

Text on LCD Display
Restoring Network !!

8. WSN Demo Mesh Reference Application

The WSNDemo application is based on the MiWi API. This application demonstrates the network and radio frequency performance of the hardware components. This application consists of the embedded firmware, which supports functions for PAN coordinator, coordinator, end device, GUI visualization application, and WSNMonitor, which runs on a PC. In the WSNDemo, the nodes communicate based on a proprietary messaging protocol.

The package includes the WSNMonitor PC application in binary format, and the WSNDemo embedded application is available in binary format and source code. The source code for the WSNDemo application can be modified and extended to develop WSN applications for a variety of application scenarios.

The end devices, coordinators, and PAN coordinator devices emulate the sensor data reading for light and temperature sensors, and forward collected data to the WSNMonitor application for visualization. The end devices follow a duty cycle to transmit data to the coordinator. Using the serial connection, the PAN coordinator transmits the received packets, along with its own sensor data (or emulated sensor data), to the WSNMonitor application. Those transmitted values are displayed on the WSNMonitor panes as temperature, light, and battery level measurements.

The WSNMonitor also visualizes network topology by drawing a tree of nodes that have joined the network. For each of the nodes, parameters like node address, node sensor information, and link quality data are displayed.

- RSSI indicates a link's current condition and is measured in dBm. The RSSI resolution is 3 dBm.
- LQI is a numeric parameter defined within the 0 to 255 range, is used to measure the link quality. Larger values mean to have a better link, while values close to zero indicate a poor connection.

The Network Freezer feature is used to save or store critical network information into the non-volatile memory. By default, Network Freezer is enabled in WSN Demo application.

When the device reboots, it restores the existing data from the non-volatile memory and continues to operate in the same network with existing network information.

By default, the sleep feature is enabled in the end device application. Both the MCU and transceiver are in sleep when the device is idle.

8.1 Launching the Demo

Perform the following steps to launch the demo application.

1. Assemble the devices.
2. Program the devices with firmware images. One node must be programmed as PAN coordinator, and the others as coordinators or end devices.
3. Connect the PAN coordinator node to the PC using the serial interface.
4. Run the WSNMonitor.
Use the following setting for the serial connection of the WSNMonitor:
 - BAUD RATE – 38400
 - PARITY – None
 - DATA BITS – 8
 - STOP BITS – 1
 - FLOW CONTROL – Off (On for the XPRO board)
5. Observe the coordinator node in the WSNMonitor.

6. Power on the other nodes and observe them displayed in the WSNMonitor.
7. Select any coordinator node and click on the bulb icon next to it, and observe the device blink its LEDs.
The SW button on the board is used to erase all the existing items in the memory and reset the device as a factory new device.

8.2 Network Startup

The PAN coordinator organizes the wireless network automatically. Upon starting the network, every node informs the network of its role. When the PAN coordinator is powered on, it switches to an active state even though no child node is present. This behavior is normal. It indicates that the PAN coordinator is ready and the child nodes can join the network with the coordinator's PAN ID. By default, the coordinator uses PAN ID 0x1234, which is recognized by all the coordinators. The PAN ID can be modified by the user through the application's configuration file.

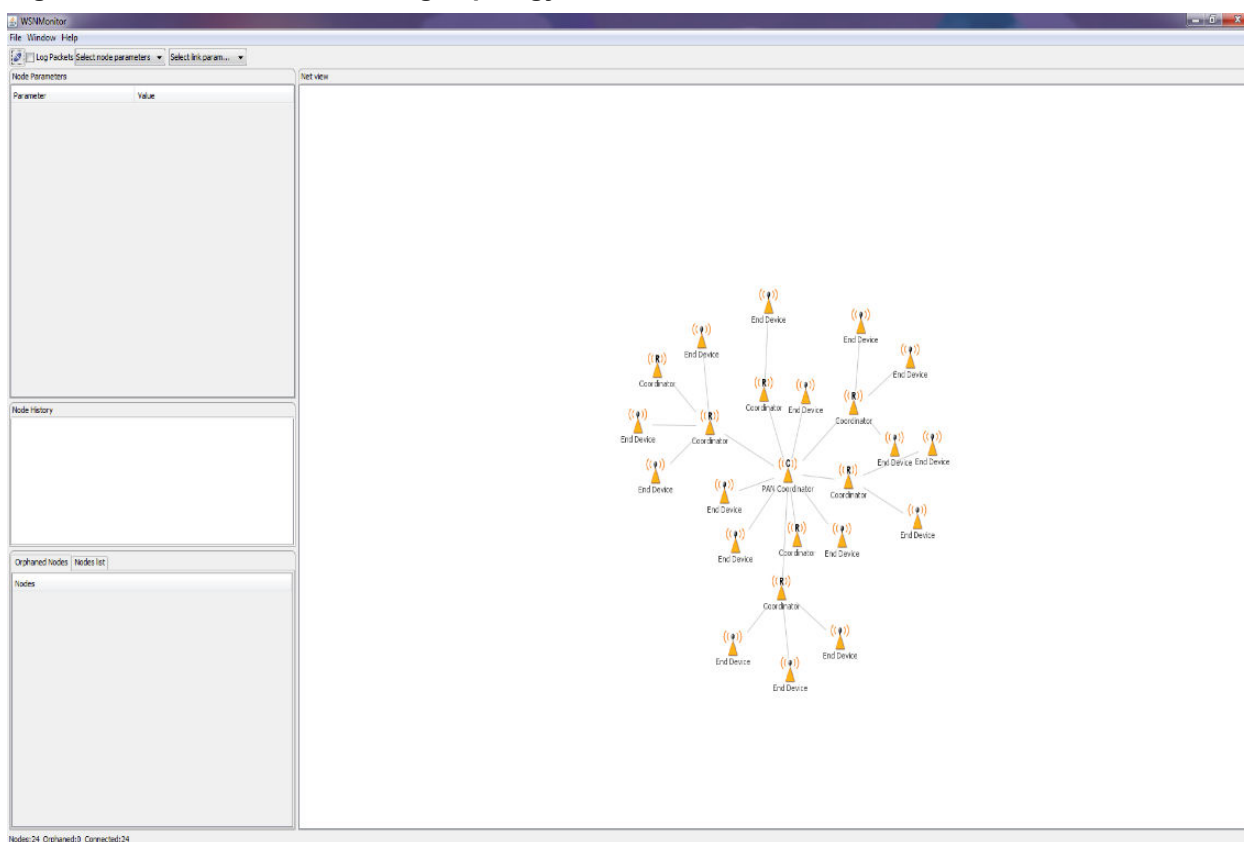
If the PAN coordinator is absent or has not been turned on, the coordinators and end devices remain in the Network Search mode. In this mode, the coordinators scan the channels specified in the channel mask in search of a network. By default, the channel mask in the application provided with the release contains a single channel. On rare occasions, if the frequency corresponding to the radio channel is busy, the coordinator node may stay in the network search mode. If this happens, it may become necessary to change the application's channel mask to select another channel by changing the application's configuration file and recompiling the application.

Network health can be monitored through the WSNMonitor application.

8.3 WSNMonitor

The WSNMonitor is a PC counterpart to the WSNDemo embedded application. It can be used to display MiWi mesh network topology and other information about a wireless sensor network. A typical WSNMonitor screen is shown in the following figure. It contains topology, sensor data, node data panes, and application toolbars.

The following is a sample topology of MiWi Mesh protocol.

Figure 8-1. WSNMonitor Showing Topology of MiWi™ Mesh Protocol

The topology pane displays the network topology in real time. This helps the user to monitor:

1. the formation of the network,
2. dynamic changes while nodes join,
3. nodes sending data across, or,
4. when nodes leave the network.

The network topology is constructed on the basis of next-hop information for each of the nodes. Each link is also tipped with RSSI and LQI values. Each of the nodes has an icon, with the node's address or name below and sensor readings to the right of the icon, if required by settings.

The sensor data pane displays data coming from onboard sensors of the selected node. It is presented in graph and table format. Other parameters for each node are available in a table format. The node data pane includes a sensor selection combo-box, which is used to switch between sensor types.

By default, in the topology pane, nodes are labeled with their short addresses. Double click to assign another title to any desired node. Press 'Cancel' to set back the node's title to the short address.

The screenshot displays the WSNMonitor application interface, which is divided into several functional areas:

- Menu bar:** Located at the top, containing 'File', 'Window', and 'Help' menus.
- Log Packets:** A button in the top toolbar for logging network packets.
- Select node parameters / Select link param...:** Dropdown menus for configuring data display.
- Node Parameters:** A panel on the left showing details for the selected node (0x1C02), including MAC address, software version, channel, and LQI.
- Serial port settings:** A sub-panel within Node Parameters for configuring the communication port.
- Node History:** A table showing recent node activity, including time and value.
- Orphaned Nodes / Nodes list:** A panel at the bottom left showing a list of connected nodes and their addresses.
- Net view:** The central area displaying a network topology diagram with nodes represented by icons and their addresses.
- Topology pane:** A label pointing to the network diagram area.
- Selected node:** A callout pointing to the node 0x1C02, which is highlighted with a green dashed box.
- Click to make the node identify itself by flashing LED:** A callout pointing to a light bulb icon next to the selected node.

The network topology shows a central Coordinator node (0x1C02) connected to several Router nodes (R) and End Nodes (E). The nodes are distributed across the network, with some having specific addresses like 0x29:ZB_230_0 and 0x30:ZB_230_1.

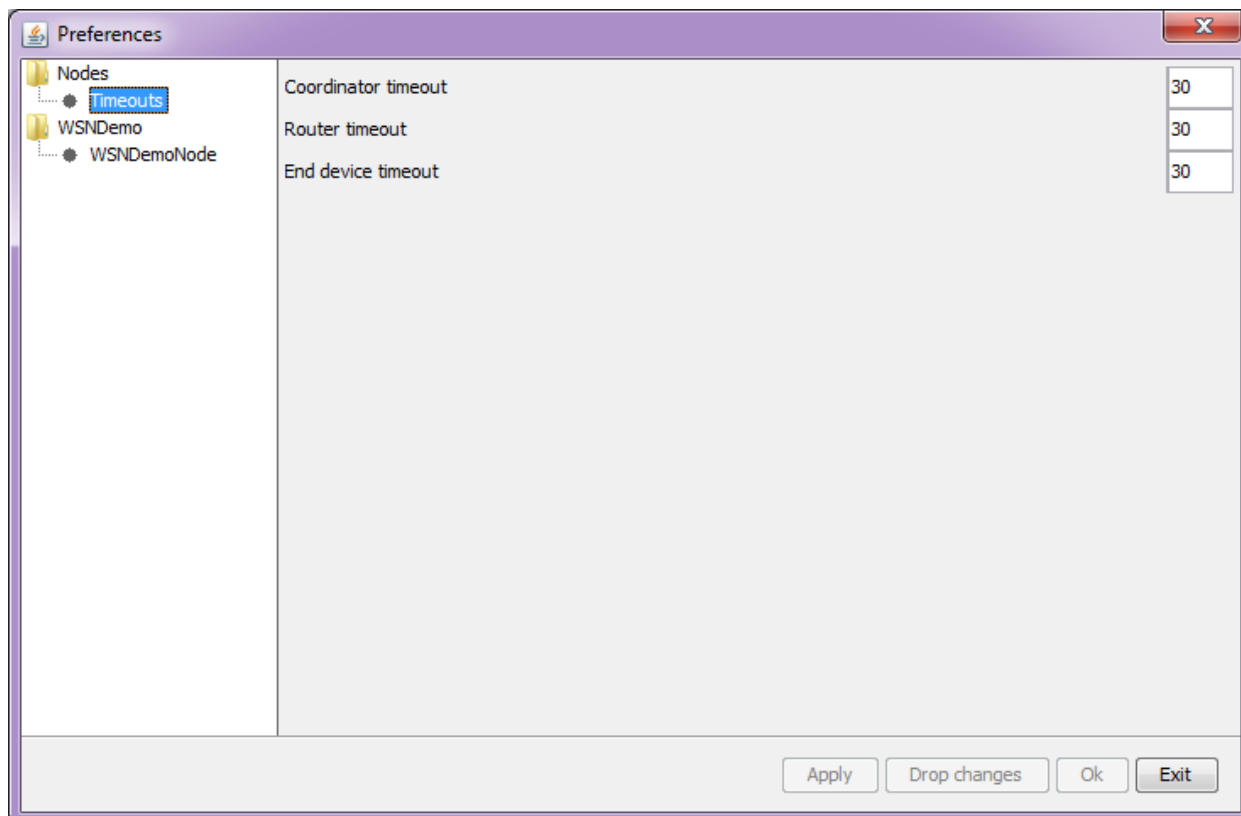
When the user clicks a node in the topology pane, a button to identify the node appears under the node's icon. When the user clicks this button, WSNMonitor sends a command which is delivered to the PAN Coordinator through a serial connection and wirelessly to the target node. After receiving the command, the target node blinks with its LED for several seconds.

Node Timeouts

Note: This value does not correspond to the frequency with which data are transmitted by each type of the device.

To get smooth topology visualization, it is recommended to set timeouts at 20 seconds for the PAN coordinator and the coordinator, and 30 seconds for an end device. Assuming a default application configuration, these timeouts cover three periods between sending a packet. Therefore, at least three packets must be lost before a node is removed from the WSNMonitor topology pane.

Figure 8-3. WSNMonitor Preferences Menu



In WSNMonitor, the PAN coordinator refers to the coordinator, the coordinator refers to the router, and the end device refers to the end device.

8.6 Sensor Data Visualization

Each board sends temperature, light, or battery sensor values (or emulated values) to the PAN coordinator, which in turn sends it to the PC. The WSNMonitor displays the values from onboard sensors next to a node icon inside the topology pane. A corresponding option can be selected in the node or link parameters from the quick settings toolbar.

The user can select any node in the topology pane to monitor the node's activity and the node data in one of the following three different forms:

- Text
- Table
- Chart

The onboard sensor data is displayed next to each node in the topology pane. These values are tipped with arrows indicating whether the value increased or decreased in relation to the previous sample. A given node is selected when it is clicked and a dashed frame is visible around it.

The sensor data pane displays the same values. This helps the user to observe how the values change over a period of time. The sensor data pane includes a sensor selection combo-box. Use the button on the sensor control toolbar to display the desired types of sensor data.

9. OTAU in WSN Demo Mesh Application

9.1 Software Prerequisites

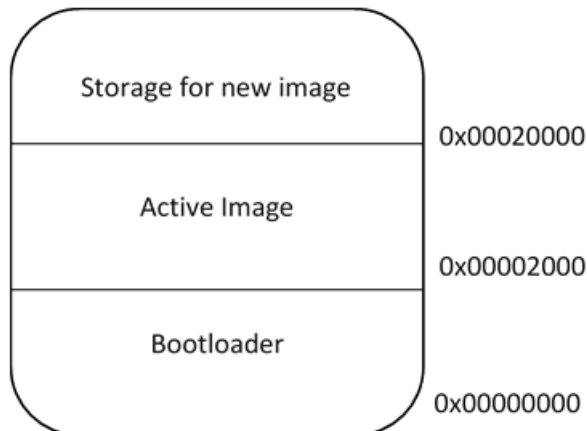
1. The Bootloader PC Tool is available in the <ProjectDir>\thirdparty\wireless\miwi\services\otau\tools directory. For more details on the installation procedure, see [3.5 Installing the Bootloader PC Tool](#).
2. WiDBGSetup.msi is available in the <ProjectDir>\thirdparty\wireless\miwi\services\otau\tools directory. For more details on the installation procedure, see [3.6 Installing the WiDBG Tool](#).
3. Bootloader files for SAMR21 and SAMR30 are available at <ProjectDir>\thirdparty\wireless\miwi\services\otau\tools.

9.2 OTAU Client

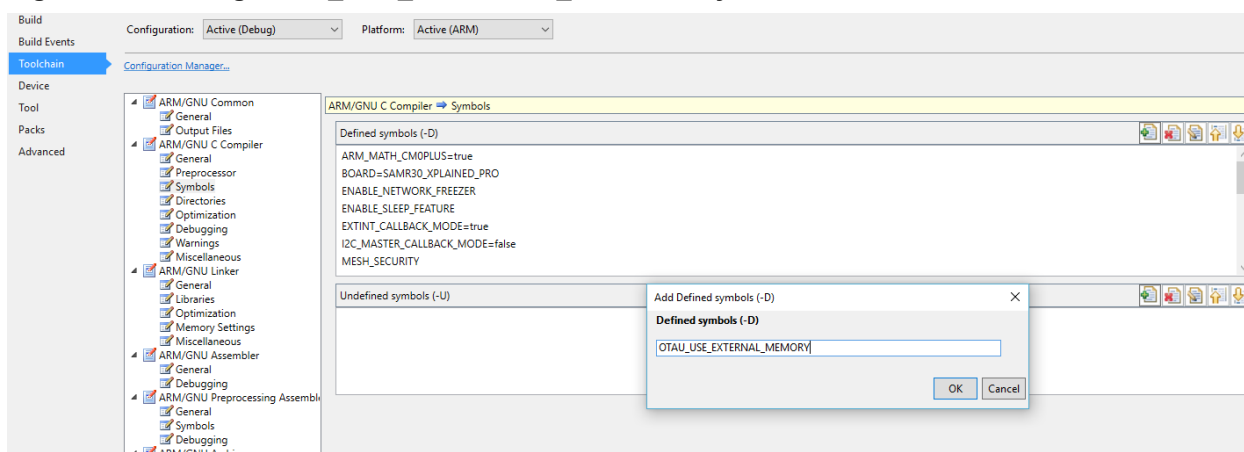
9.2.1 Memory Layout of Client

To upgrade the application from WiDBG UI, the serial bootloader in the AVR2054 is modified to support internal Flash swap functionality. The last page (of size 64 bytes) of Flash is used to store the information for the bootloader to perform a Flash swap. The following figure displays the memory layout of the client if the image is stored in the internal Flash.

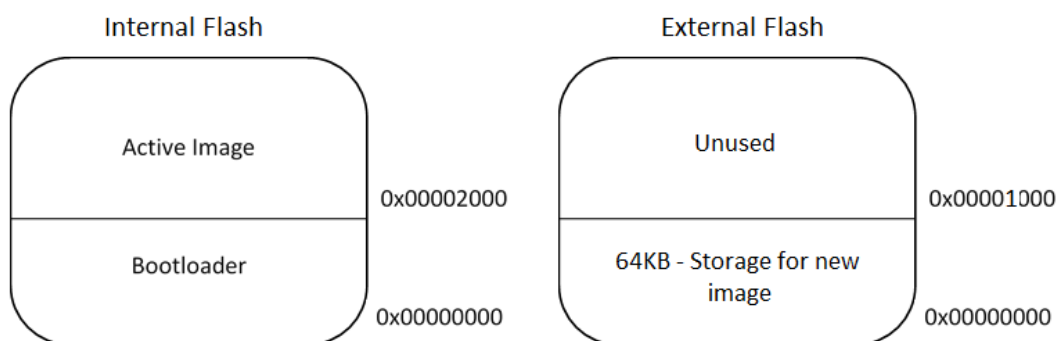
Figure 9-1. Internal Flash



The user has an option to use external Flash for storing the new image. The SAMR30 Module Xplained Pro board has AT25DFX041B external Flash. To use the external memory for storing the image, add symbol `OTAU_USE_EXTERNAL_MEMORY` in the project configuration as shown in the following screenshot.

Figure 9-2. Adding OTAU_USE_EXTERNAL_MEMORY Symbol

The memory layout is shown in the following figure.

Figure 9-3. Memory Layout

9.2.2 Programming Client Firmware

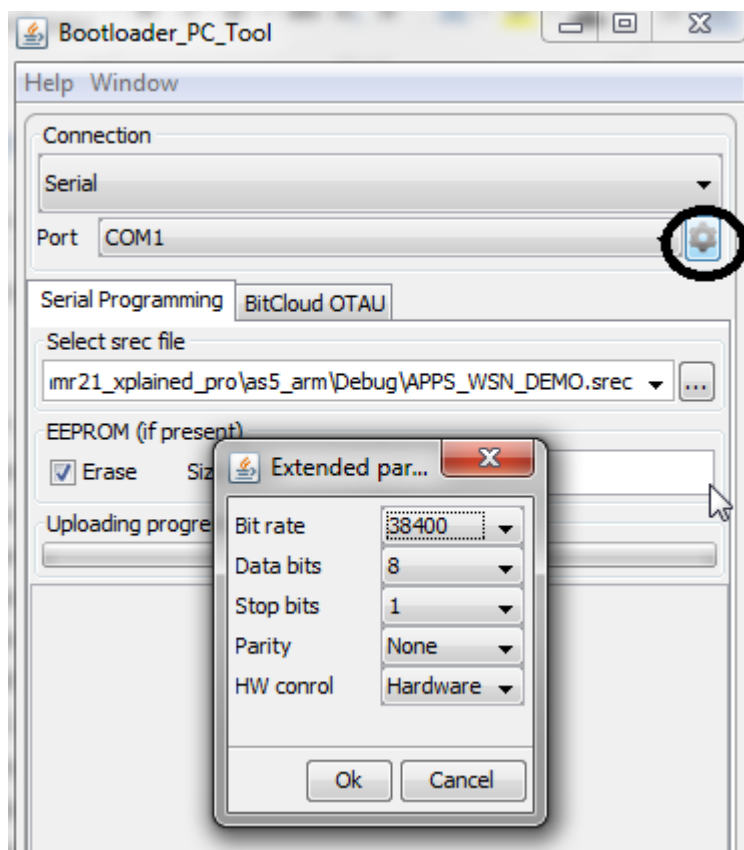
Perform the following steps to program the client firmware.

1. Select a WSN Demo application with required device type. For more details, see [3. Development Environment Setup](#).
2. Open `<ProjectDir>\thirdparty\wireless\miwi\services\pds\src\wl\linkerscripts\samr21\gcc\samr21g18a_flash.ld`.
3. Modify the memory region of ROM from 0x00000000 to 0x00002000 as:

```
/* Memory Spaces Definitions */
MEMORY
{
  rom      (rx)   : ORIGIN = 0x00002000, LENGTH = 0x00040000
  ram      (rwx)  : ORIGIN = 0x20000000, LENGTH = 0x00008000
}
```

4. Rebuild the project.
5. Program the client node with bootloader elf file available in `<ProjectDir>\thirdparty\wireless\miwi\services\otau\tools`.
6. Open the Bootloader PC Tool and configure the serial settings as shown in the following figure.

Figure 9-4. Bootloader PC Tool Serial Settings

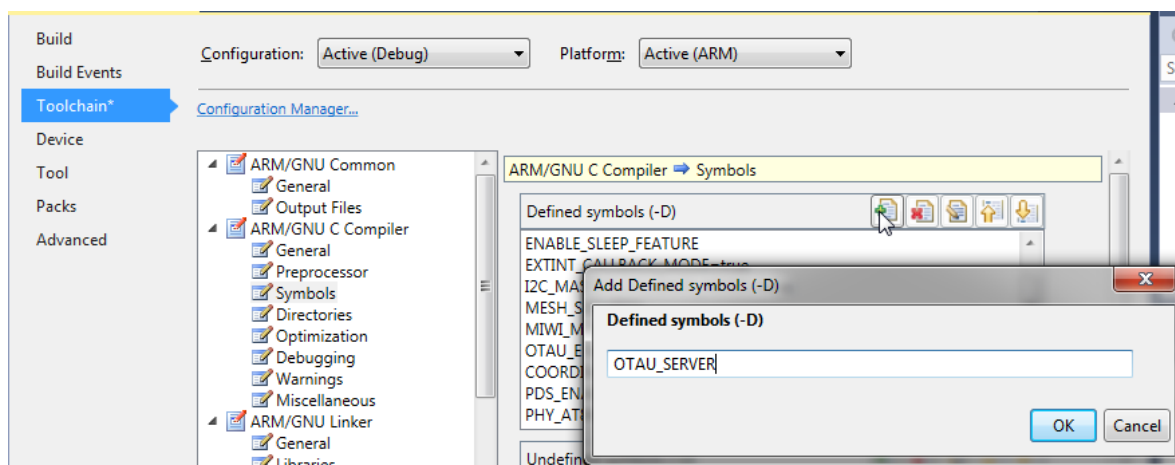


7. Select the `APP_WSN_DEMO.srec` file from the project directory which is generated at Step 4, and click **Upload** button.
8. To start upload, reset the client node.

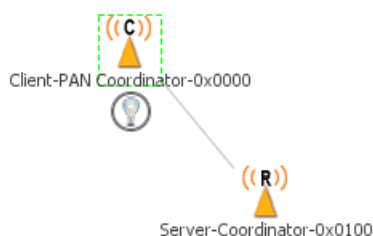
9.3 OTAU Server

Note: The PAN coordinator cannot be used as OTAU Server because the PAN coordinator uses the serial interface to communicate to the WSNMonitor. However, the PAN coordinator can be configured as OTAU Client.

1. Configure the WSN Demo application as coordinator (see [3. Development Environment Setup](#)).
2. Add the OTAU_SERVER in symbol as shown in the following screenshot.
3. Click OK.

Figure 9-5. Add Defined Symbols Dialog Box

4. Compile and program the OTAU server node. When the node joins the network with the available coordinator, the WSNMonitor is updated as shown in the following figure.

Figure 9-6. OTAU Server Node in WSNMonitor

10. Atmel WiDBG for OTAU

When the installation of WiDBGSetup.msi is completed, a shortcut is created on the desktop. Double click and open the WiDBG.

Figure 10-1. Opening WiDBG Tool

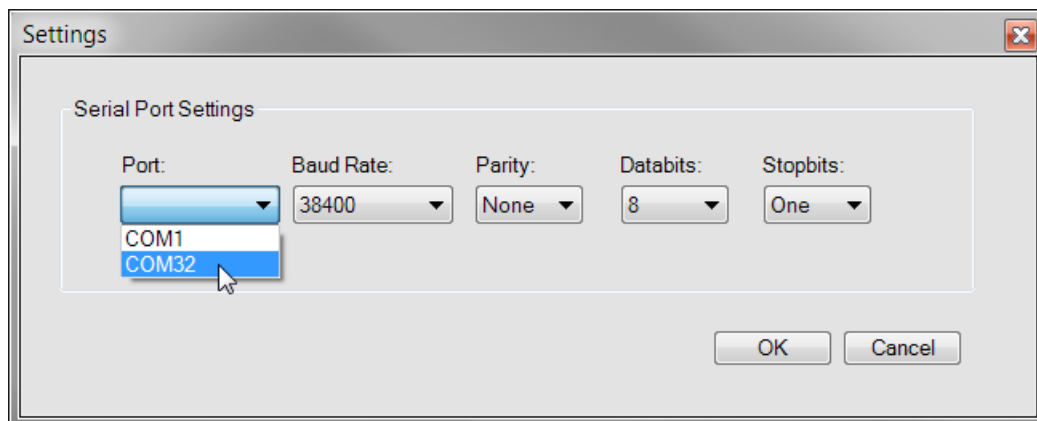


10.1 Starting the Session

This section explains how to start the session using the WiDBG tool. Ensure that the server node is connected to the Host Machine (PC) via USB.

1. Open the WiDBG application.
2. Choose Settings>Connection.
3. In the Settings dialog box, select the desired COM port and click **OK** to save the settings.

Figure 10-2. Serial Port Settings



4. To connect to the OTAU server, perform the following:
 - 4.1. Choose Session>Connect>Serial Port. When successfully connected, the connection success status message is displayed.
 - 4.2. Select the desired channel number from the drop-down list.
5. Click **Start**  to start the debugging session.

10.1.1 Notification Settings

When the session is started, all the OTAU clients start to notify after configuring the notification interval. This section explains how to configure the notification intervals of the clients.

Depending on the number of client nodes, the user can adjust the interval; therefore, the traffic on the channel can be minimized for effective debugging during the session. By default, all clients notify the server every five seconds. This can be changed if the number of nodes is higher.

1. Click the **Refresh Notification** icon to open the Notify Interval Settings window. The properties are set as follows:

Figure 10-3. Refresh Notification

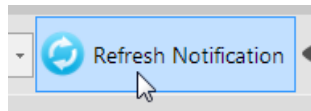
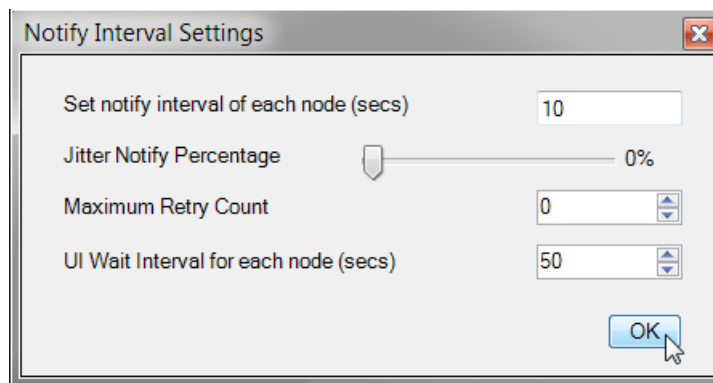


Figure 10-4. Notify Interval Settings for the Client



– **Notify Interval** – Client Notify Interval Setting

This is the interval assigned to each client node of the network. When it receives the value from the tool it starts sending the notification to the user according to the specified interval.



Tip: For example: If 5 clients are connected to the server and the time is set to 10 seconds, then each client will send a notification to WiDBG on the 10th, 20th ... and so on after it receives the value from tool until we change or stop the session.

– **Jitter Notify Percentage** – Jitter Notify Percentage Setting

This is the percentage value assigned to each client to randomize the time to avoid the collision when the other client is notifying at the same time. By default, randomization is not applied to the clients.



Tip: Increase the percentage if the network is large and decrease the percentage if the network is small.

- **Maximum Retry Count** – Retry Count Setting
If the notification is not recognized by the tool due to heavy traffic, it forces the client to retry the notification for specified count value.
- **UI Wait Interval** – UI Wait Interval Setting
The user must set the minimum wait time for the tool to decide a node whether it is active, using **UI Wait Interval** settings from **Preferences** window.



Tip: The user must set the timing accordingly at UI Wait Interval based on Notify Interval Value.

By default, the timing is configured as:

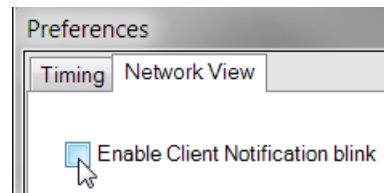
- Notify Interval for Client = 5 seconds
- UI Wait Interval for WiDBG = 15 seconds

Therefore, the WiDBG waits for 15 seconds which is three times the client notify interval setting to decide is a node as ACTIVE.



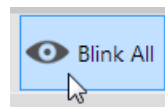
Tip: Users can see the client's notification by enabling the Enable Client Notification blink checkbox from Settings>Preferences>Network View Settings. When the Enable Client Notification blink checkbox is enabled, the node blinks as GREEN color for a moment when it is responding to the tool.

Figure 10-5. Network View



2. Identify all the clients using Identify All button.
Click the **Blink All** icon to blink all the nodes connected to the network. This blinks by toggling its board LED.

Figure 10-6. Blink All



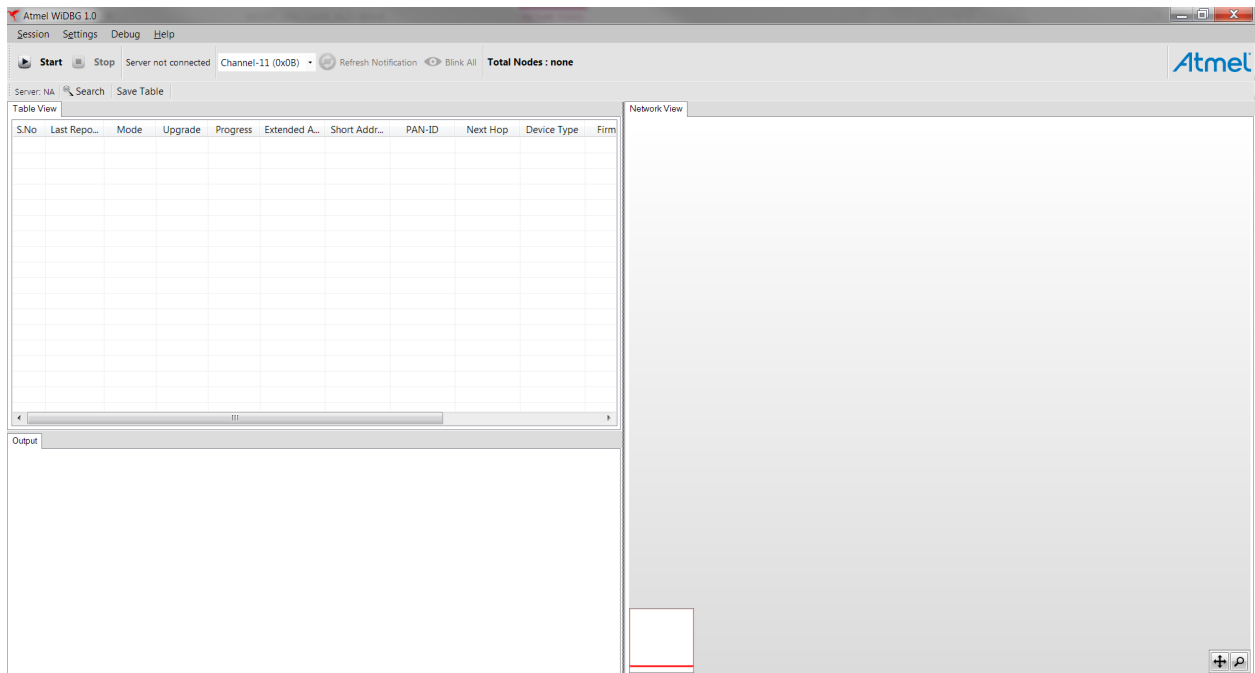
10.2 Main Window

The Main window summarizes nearly all the information of the connected network. When the session is started, this window provides information on the stability of the system at any time to the user. This window provides the following information:

1. Table View
2. Network Topology View
3. Output View

Each view is designed to provide useful information about connected clients. The network view and table view are interconnected to each other. It gives clarity to the user to perform analyzing tasks on each node. Also, both the views are lively in nature and reflect the status of nodes immediately with the time stamp.

Figure 10-7. Main Window



10.2.1 Table View

The Table View lists all the nodes that are connected to the network. It has useful information about the node. Each node occupies a row with several columns, as shown in the following screenshot.

Figure 10-8. Table View

S.No	Last Repo...	Mode	Upgrade	Progress	Extended Address	Short Address	Device Type	Parent	LQI	Supported ...	Firmware	Firmware v...	Board	Board versi...
1	17:08:53	APP	Upgrade		0x1122334455667788	0x1					LWMesh	1.2.1	SAMR21XPro	1.0.0
2	17:11:40	APP	Upgrade		0x4250000005002	0x5002	Router	0x1	228	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0
3	17:11:35	APP	Upgrade		0x4250000005032	0x5032	Router	0x5019	236	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0
4	17:11:36	APP	Upgrade		0x4250000005027	0x5027	Router	0x1	220	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0
5	17:12:17	APP	Upgrade		0x4250000005015	0x5015	Router	0x4	244	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0
6	17:12:24	APP	Upgrade		0x4250000005011	0x5011	Router	0x5027	204	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0
7	17:12:36	APP	Upgrade		0x4250000005016	0x5016	Router	0x501C	172	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0
8	17:12:41	APP	Upgrade		0x6	0x6	Router	0x5026	244	0x3	LWMesh	1.2.1	RFR2XPro	1.0.0
9	17:12:50	APP	Upgrade		0x4	0x4	Router	0x6	240	0x3	LWMesh	1.2.1	RFR2XPro	1.0.0
10	17:12:59	APP	Upgrade		0x3	0x3	Router	0x5030	252	0x3	LWMesh	1.2.1	RFR2XPro	1.0.0
11	17:13:03	APP	Upgrade		0x4250000005018	0x5018	Router	0x1	208	0x3	LWMesh	1.2.1	SAMR21XPro	1.0.0

Each row is updated when the client sends data to the WiDBG and the last reported time is updated. Each node supports two operational modes called PHY mode, and APP mode. By default, all the nodes operate on the APP mode. The user can switch to the PHY mode when the node is not responding to the tool.

The user can upgrade the node using the upgrade button provided and it displays the live status in the progress bar, as shown in the following figure. The user can upgrade multiple devices in parallel.

Figure 10-9. Upgrade Button

Upgrade	Upgrade Progress	Extended Address
Upgrade		0x112233445566...
Upgrade		0x42500000001EB
Upgrade		0x42500000001EF

The other columns explain detailed information about the node.

When any node goes to IDLE or Not-Responding state, the Short-Address column turns to red, indicating its state.

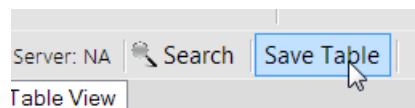
Figure 10-10. Idle State Indication at Table

Extended Address	Short Address	Device Type
3445566...	0x0	
000001EB	0x1EB	Router
000001EF	0x1EF	Router
000001F1	0x1F1	Router
000001EC	0x1EC	Router
000001EE	0x1EE	Router
000001F3	0x1F3	Router
000001ED	0x1ED	Router

10.2.2 Saving the Table View

The content of the Table View can be saved as CSV file by using the **Save Table** icon.

Figure 10-11. Save Table Icon

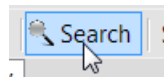


10.2.3 Searching the Nodes

To locate the node on a Table and Topology View, the user can search using the node's addresses.

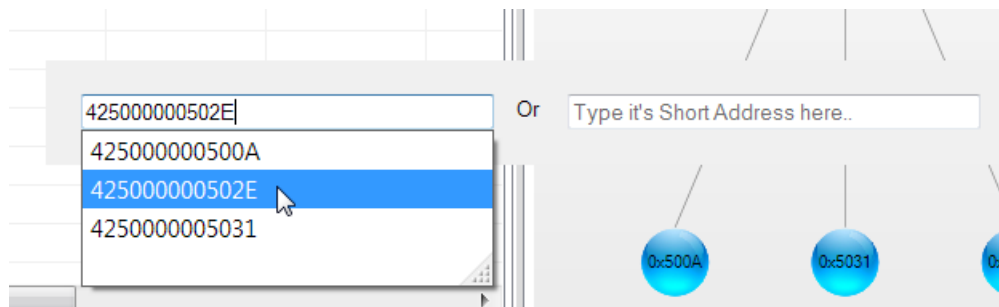
Click **Search**, which is available above the Table View.

Figure 10-12. Search Icon



Enter the address of the node to be located.

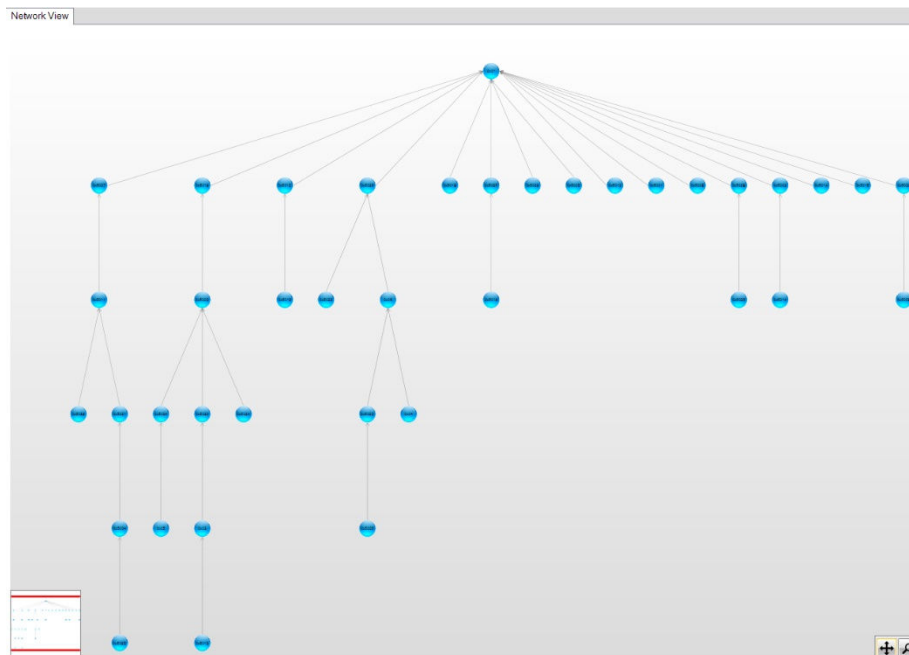
Figure 10-13. Searching the Node



10.2.4 Network Topology View

The network view is an advanced diagrammatic representation provided to the user to view the topology of the network.

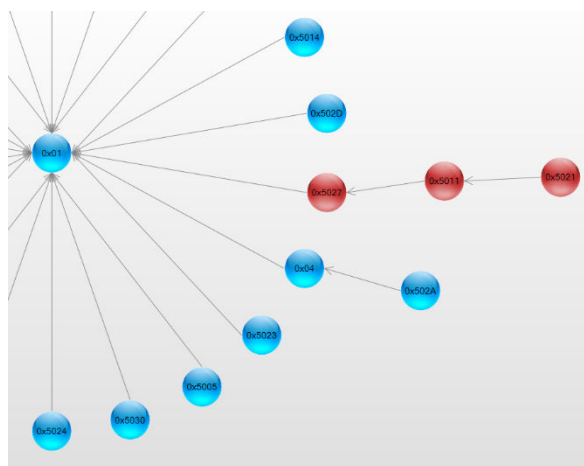
Figure 10-14. Network Topology View



When the session starts, the network view becomes active and starts to populate the view. Each node joins with its parent and forms the network. It shows the hopping of the network immediately when it happens in the network. Also, when a node on the table is clicked, its corresponding node in this view turns orange and the user can zoom in/out or pan the diagram, when the network becomes huge.

Similar to the Table View, this view also actively reflects the node status as red or blue based on the node state.

Figure 10-15. Idle State Indication at Network View

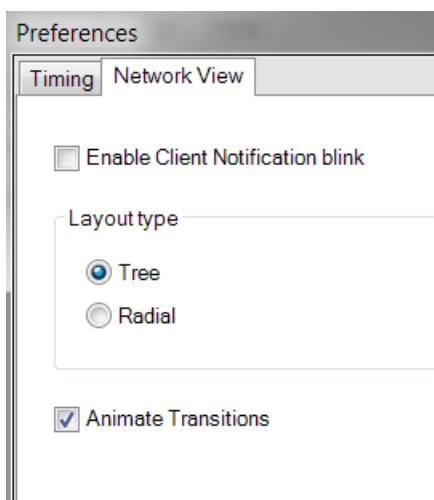


10.2.4.1 Layout Type

The Topology View can be configured as two different layout types.

To set the layout type, navigate to Settings>Preferences>Network View.

Figure 10-16. Network View

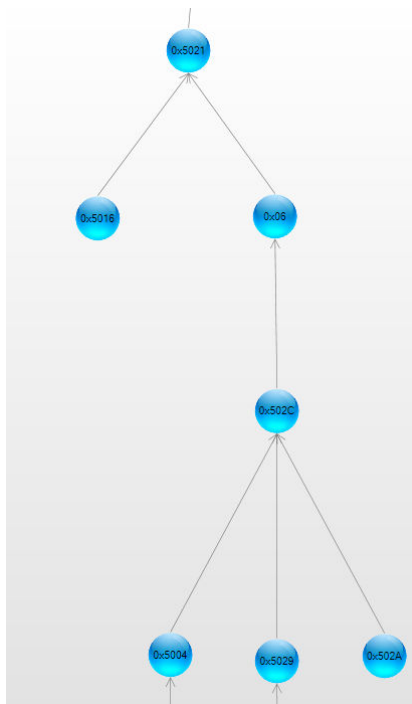


Layout Type Setting

- **Tree**

It is the default layout view when the tool runs for the first time. It populates the view in a tree-like structure, as shown in the following figure.

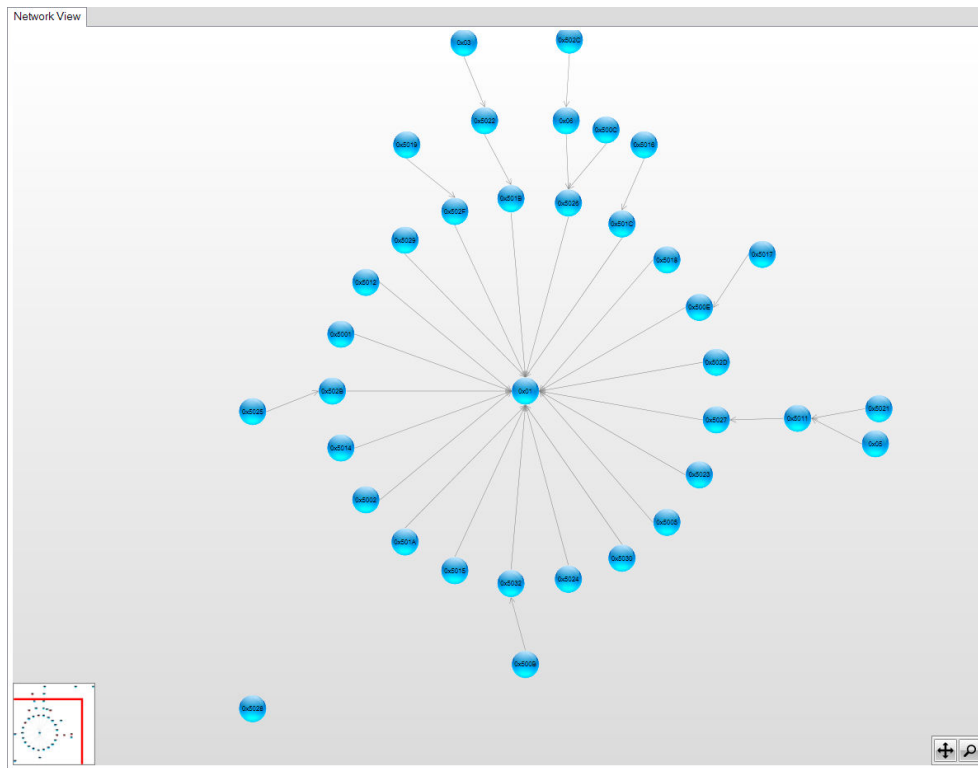
Figure 10-17. Tree Layout View



- **Radial**

It forms the nodes in a circular fashion, as shown in the following figure.

Figure 10-18. Radial Layout View



10.2.5 Output View

The Output View is used to understand all the activities happening in the tool. It logs all the information that the user requested and provides its status or progress. All the success or failure messages can be seen using this view.

Figure 10-19. Output View

```

Output

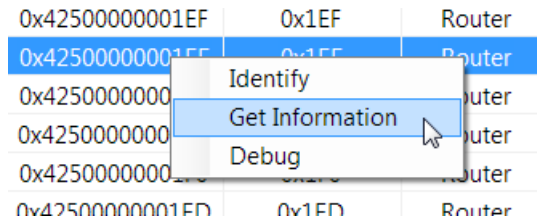
20:36:08 : WiDBG has started collecting data..
20:36:08 : Getting server information...
20:36:08 : Server : 0x1122334455667788 has responded.
20:36:08 : Waiting for the client nodes from Channel-21 (0x15)
20:36:08 : Please wait...
20:36:10 : Node : 0x42500000001F1 has joined the network
20:36:11 : Node : 0x42500000001F2 has joined the network
20:36:11 : Node : 0x42500000001F4 has joined the network
20:36:11 : Node : 0x42500000001EF has joined the network
20:36:11 : Node : 0x42500000001EE has joined the network
20:36:12 : Node : 0x42500000001F3 has joined the network
20:36:12 : Node : 0x42500000001EB has joined the network
20:36:12 : Node : 0x42500000001F0 has joined the network
20:36:12 : Node : 0x42500000001ED has joined the network
20:36:12 : Node : 0x42500000001EC has joined the network
  
```

10.3 Upgrading the Node

A node can be upgraded using the Table View. To upgrade a node, perform the following steps.

- Right-click on the row and click **Get Information** as shown in the following screenshot. This provides the tool information about the firmware version which is currently running on the node.

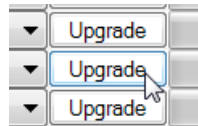
Figure 10-20. Get the Firmware and Board Information



The server node asks the client node about its version details and populates it on the respective columns.

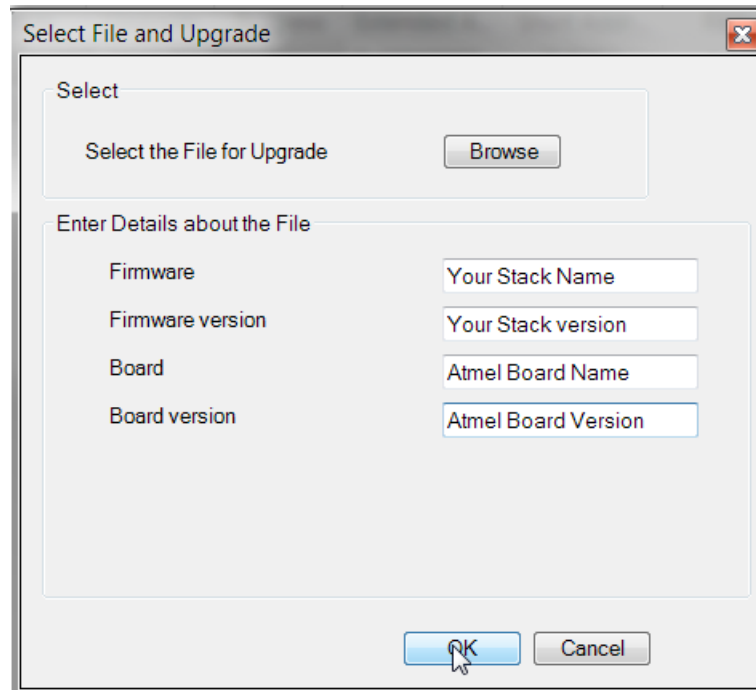
- Click the **Upgrade** button of the corresponding node.

Figure 10-21. Upgrading the Respective Node



- Click the **Browse** button and select the HEX file from the list.
Details about the HEX file can be sent to the client using the following configuration. The same can be modified in `otauClientInfoIndication_t clientInfo` in `client_notify.c` of the client project to see the upgraded image information after upgrading.

Figure 10-22. Selecting the HEX File and Entering the Firmware Details



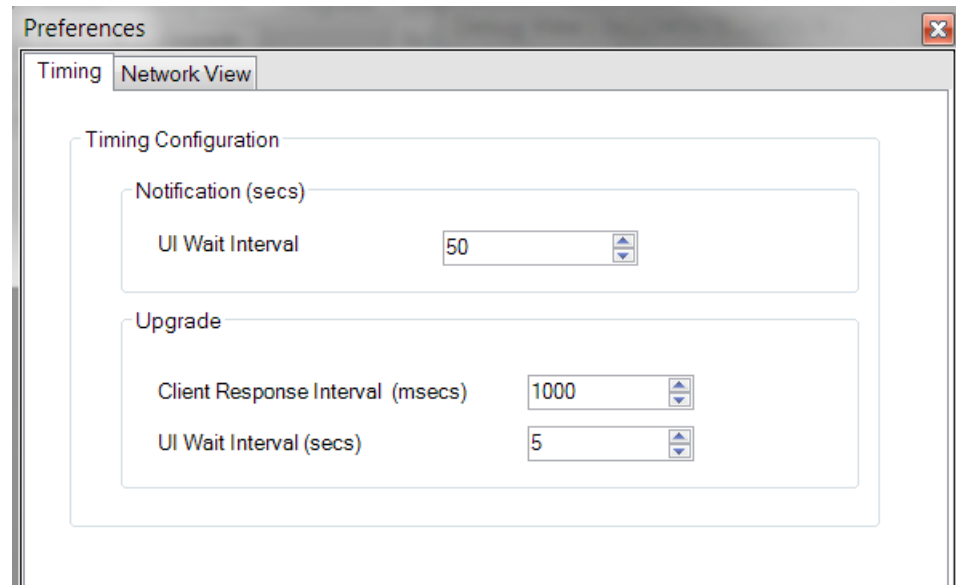
Note: The details about the HEX file on each field must not exceed the length of `client_information` sequence of the `user_config.xml` file.

- Click **OK**. The upgrade begins and progress bar starts to increase when the client receives the hex file data.

The upgrade data is sent to the client for every 1000 milliseconds by default.

To modify this interval, open the **Preferences** window from **Settings** menu and change the **Client Response Interval** under the Timing tab.

Figure 10-23. Setting the Client Response Interval to Upgrade



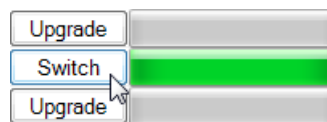
- If no response is received from the client, the tool stops the upgrade process after a predefined time period. This can be configured using the **UI Wait Interval** value in the **Upgrade** section of the **Preferences** window.

Figure 10-24. Setting the UI Wait Interval to Upgrade



- When the full image is sent to the client, it is ready to switch the image as per the new image file. The user can switch to the new image as required.

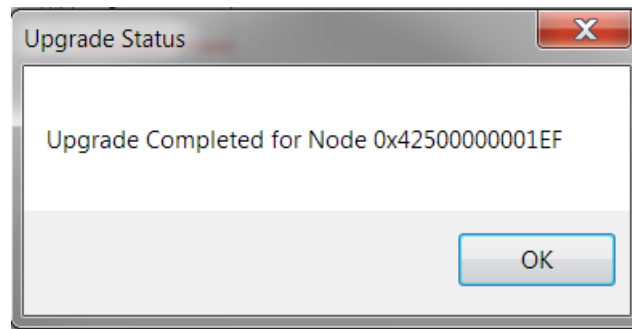
Figure 10-25. Switching to a New Image



Click **Switch** to write into its Flash memory by the client and restarts as per the new image file selected during this process.

When the image is successfully switched, a pop-up notification about the upgrade status is displayed as shown in the following screenshot.

Figure 10-26. Upgrade Completed Status Window



11. Document Revision History

Revision	Date	Section	Description
B	06/2019	1.1 MiWi v6.3 Release Content	Updated the title from 6.2 to 6.3.
		• Table 5-1 • Figure 5-2	Updated
		• Table 7-1 • Figure 7-3	Updated
		8.4 Identifying Nodes	Added a new paragraph.
A	02/2019	Document	Initial Revision

The Microchip Website

Microchip provides online support via our website at <http://www.microchip.com/>. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to <http://www.microchip.com/pcn> and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the web site at: <http://www.microchip.com/support>

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTrackr, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, FlashTec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, Vite, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BlueSky, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestlC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2019, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-4594-4

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit <http://www.microchip.com/quality>.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: http://www.microchip.com/support Web Address: http://www.microchip.com	Australia - Sydney Tel: 61-2-9868-6733 China - Beijing Tel: 86-10-8569-7000 China - Chengdu Tel: 86-28-8665-5511 China - Chongqing Tel: 86-23-8980-9588 China - Dongguan Tel: 86-769-8702-9880 China - Guangzhou Tel: 86-20-8755-8029 China - Hangzhou Tel: 86-571-8792-8115 China - Hong Kong SAR Tel: 852-2943-5100 China - Nanjing Tel: 86-25-8473-2460 China - Qingdao Tel: 86-532-8502-7355 China - Shanghai Tel: 86-21-3326-8000 China - Shenyang Tel: 86-24-2334-2829 China - Shenzhen Tel: 86-755-8864-2200 China - Suzhou Tel: 86-186-6233-1526 China - Wuhan Tel: 86-27-5980-5300 China - Xian Tel: 86-29-8833-7252 China - Xiamen Tel: 86-592-2388138 China - Zhuhai Tel: 86-756-3210040	India - Bangalore Tel: 91-80-3090-4444 India - New Delhi Tel: 91-11-4160-8631 India - Pune Tel: 91-20-4121-0141 Japan - Osaka Tel: 81-6-6152-7160 Japan - Tokyo Tel: 81-3-6880-3770 Korea - Daegu Tel: 82-53-744-4301 Korea - Seoul Tel: 82-2-554-7200 Malaysia - Kuala Lumpur Tel: 60-3-7651-7906 Malaysia - Penang Tel: 60-4-227-8870 Philippines - Manila Tel: 63-2-634-9065 Singapore Tel: 65-6334-8870 Taiwan - Hsin Chu Tel: 886-3-577-8366 Taiwan - Kaohsiung Tel: 886-7-213-7830 Taiwan - Taipei Tel: 886-2-2508-8600 Thailand - Bangkok Tel: 66-2-694-1351 Vietnam - Ho Chi Minh Tel: 84-28-5448-2100	Austria - Wels Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 Denmark - Copenhagen Tel: 45-4450-2828 Fax: 45-4485-2829 Finland - Espoo Tel: 358-9-4520-820 France - Paris Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 Germany - Garching Tel: 49-8931-9700 Germany - Haan Tel: 49-2129-3766400 Germany - Heilbronn Tel: 49-7131-72400 Germany - Karlsruhe Tel: 49-721-625370 Germany - Munich Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 Germany - Rosenheim Tel: 49-8031-354-560 Israel - Ra'anana Tel: 972-9-744-7705 Italy - Milan Tel: 39-0331-742611 Fax: 39-0331-466781 Italy - Padova Tel: 39-049-7625286 Netherlands - Drunen Tel: 31-416-690399 Fax: 31-416-690340 Norway - Trondheim Tel: 47-72884388 Poland - Warsaw Tel: 48-22-3325737 Romania - Bucharest Tel: 40-21-407-87-50 Spain - Madrid Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 Sweden - Gothenberg Tel: 46-31-704-60-40 Sweden - Stockholm Tel: 46-8-5090-4654 UK - Wokingham Tel: 44-118-921-5800 Fax: 44-118-921-5820