FTP Client Application

User guide

Version 0.2

May 2016

Redpine Signals, Inc.
2107 N. First Street, #540
San Jose, CA 95131.
Tel: (408) 748-3385
Fax: (408) 705-2019
Email: info@redpinesignals.com
Website: www.redpinesignals.com

About this Document

This document describes the process of bringing up the RS9113 based module as a FTP client application and using FTP client application user can upload, download, delete, rename, move and copy files on a server.

Disclaimer:

The information in this document pertains to information related to Redpine Signals, Inc. products. This information is provided as a service to our customers, and may be used for information purposes only. Redpine assumes no liabilities or responsibilities for errors or omissions in this document. This document may be changed at any time at Redpine's sole discretion without any prior notice to anyone. Redpine is not committed to updating this document in the future.

Table of Contents

1 Introduction 4
1.1 Protocol Overview 4
1.2 Application Overview 4
1.2.1 Overview 4
1.2.2 Sequence of Events 4
1.3 Application Setup 5
1.3.1 SPI based Setup Requirements 5
1.3.2 UART/USB-CDC based Setup Requirements 5
2 Configuration and Execution of the Application 7
2.1 Installation of FTP server 7
2.2 Initializing the Application 10
2.2.1 SPI Interface 10
2.2.2 UART/USB-CDC Interface 10
2.3 Configuring the Application 10
2.4 Executing the Application 12

Table of Figures

Figure 1: Simple FTP connection between FTP Server and FTP client 4
Figure 2: FTP Client demo set up 6

Table of Tables

No table of figures entries found.

1 Introduction

This project is applicable to all the WiSeConnect variants like WiSeConnect Plus, WiSeMCU and WyzBee. The term WiSeConnect refers to its appropriate variant.

1.1 Protocol Overview

File Transfer Protocol (FTP) is a protocol through which internet users can upload files from their computers to a website or download files from a website to their PCs.

FTP is a client-server protocol that relies on two TCP communications channels between client and server and a command channel for controlling the conversation (command port) and a data channel for transmitting file content(data port). The standard port number used by FTP servers is 21 and is used only for sending commands. Clients initiate conversations with servers by requesting to download a file. Using FTP, a client can upload, download, delete, rename, move and copy files on a server. A user typically needs to log on to the FTP server, although some servers make some or all of their content available without login, which is also known as anonymous FTP.

FTP sessions work in passive or active modes. In active mode, after a client initiates a session via a command channel request, the server initiates a data connection back to the client and begins transferring data. In passive mode, the server instead uses the command channel to send the client the information it needs to open a data channel.

Note: WiSeConnect device supports only Active mode of FTP sessions

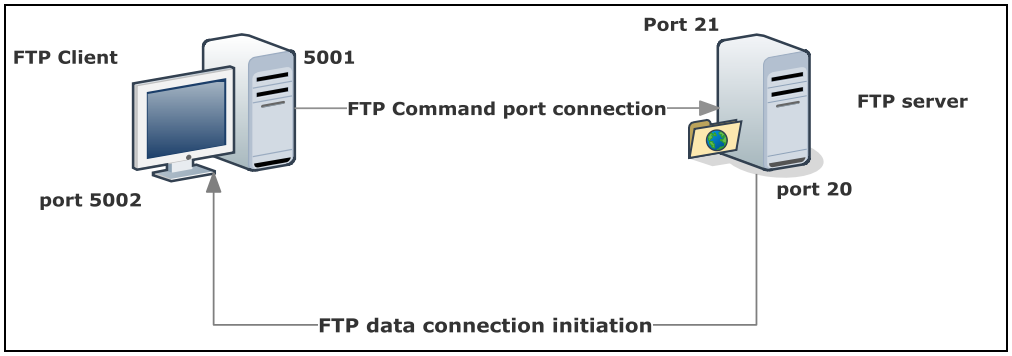


Figure 1: Simple FTP connection between FTP Server and FTP client

1.2 Application Overview

1.2.1 Overview

This application demonstrates how to connect to FTP server opened on remote peer using FTP client and how to read file from FTP server and how to write file on to the FTP server.

In this application, WiSeConnect device connects to Access Point and establishes FTP client connection with FTP server opened on remote peer. After successful connection, Application reads the data from **"read.txt"** file present in FTP server and writes back same data read from the file **"read.txt"** to the FTP server by creating file **"write.txt"**.

1.2.2 Sequence of Events

This Application explains user how to:

- Connect to Access Point
- Establish FTP connection with FTP server opened on remote peer.
- Read data from “read.txt” file present in FTP server
- Write the data to “write.txt” file which is read from the “read.txt” .

1.3 Application Setup

The WiSeConnect in its many variants supports SPI and UART interfaces. Depending on the interface used, the required set up is as below:

1.3.1 SPI based Setup Requirements

- Windows PC1 with CooCox IDE
- Spansion (MB9BF568NBGL) micro controller

Note: If user does not have Spansion (MB9BF568NBGL) host platform, please go through the SPI-Porting guide **\sapis\docs\RS9113-WiSeConnect-SAPI-Porting-Guide-vx.x.pdf** for SAPIs porting to that particular platform.

- WiSeConnect device
- A Mobile device as a Wi-Fi station (This example uses a windows Laptop)
- Windows PC2 with FTP server installed in it

Note1: FTP Server demo application can be downloaded from the given below link:
https://filezilla-project.org/download.php?type=server

Note2: FTP client functionality verified with FileZilla server version 0.9.51. So, recommended version of FileZilla software is “FileZilla_Server-0_9_51.exe”

1.3.2 UART/USB-CDC based Setup Requirements

- Windows PC1 with Dev-C++ IDE
- WiSeConnect device
- Windows PC2 with FTP server installed in it

Note1: FTP Server demo application can be downloaded from the given below link:
https://filezilla-project.org/download.php?type=server

Note2: FTP client functionality verified with FileZilla server version 0.9.51. So, recommended version of FileZilla software is “FileZilla_Server-0_9_51.exe”

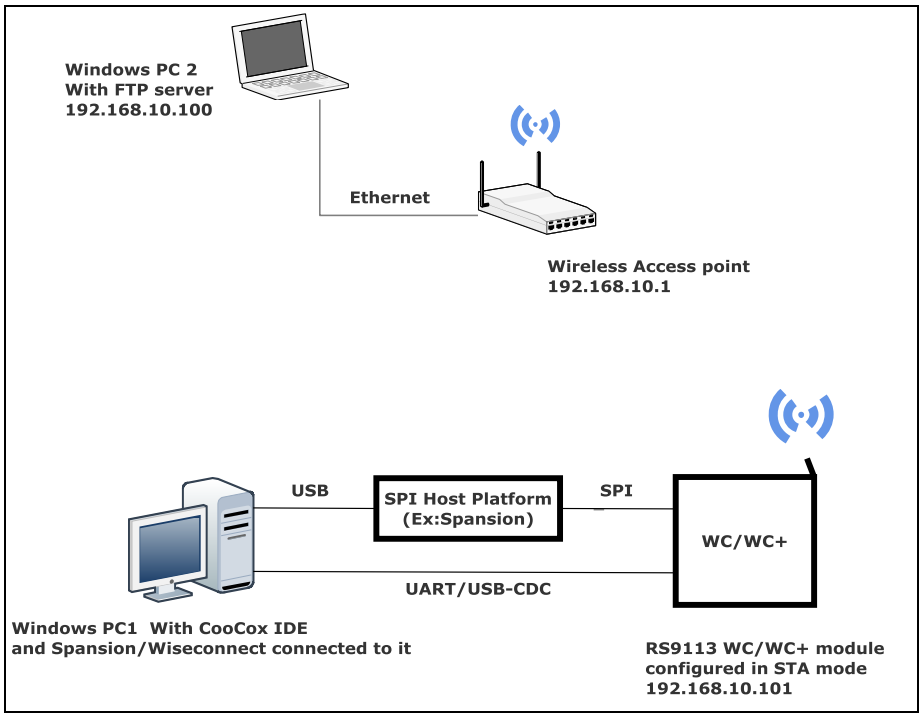


Figure 2: FTP Client demo set up

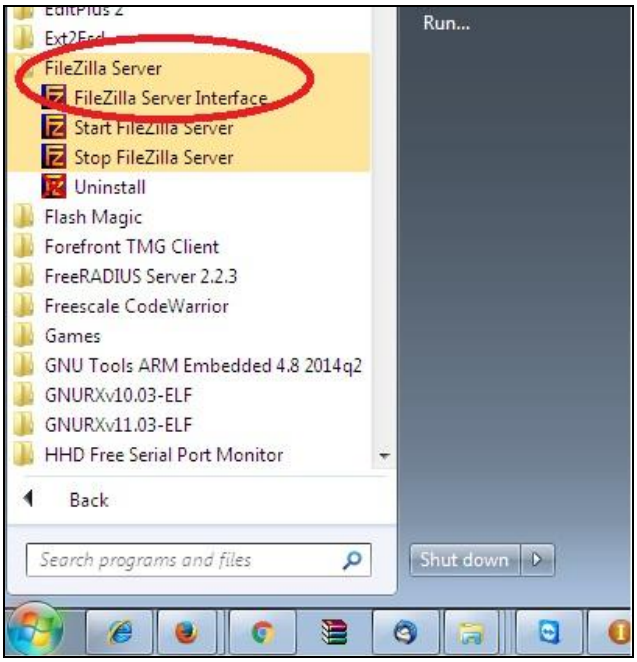
2 Configuration and Execution of the Application

The example application is available in the Release at **{Release $}/host/sapis/examples.** These examples will have to be initialized, configured and executed to test the application. The initialization varies based on the interface but configuration and execution are the common.

2.1 Installation of FTP server

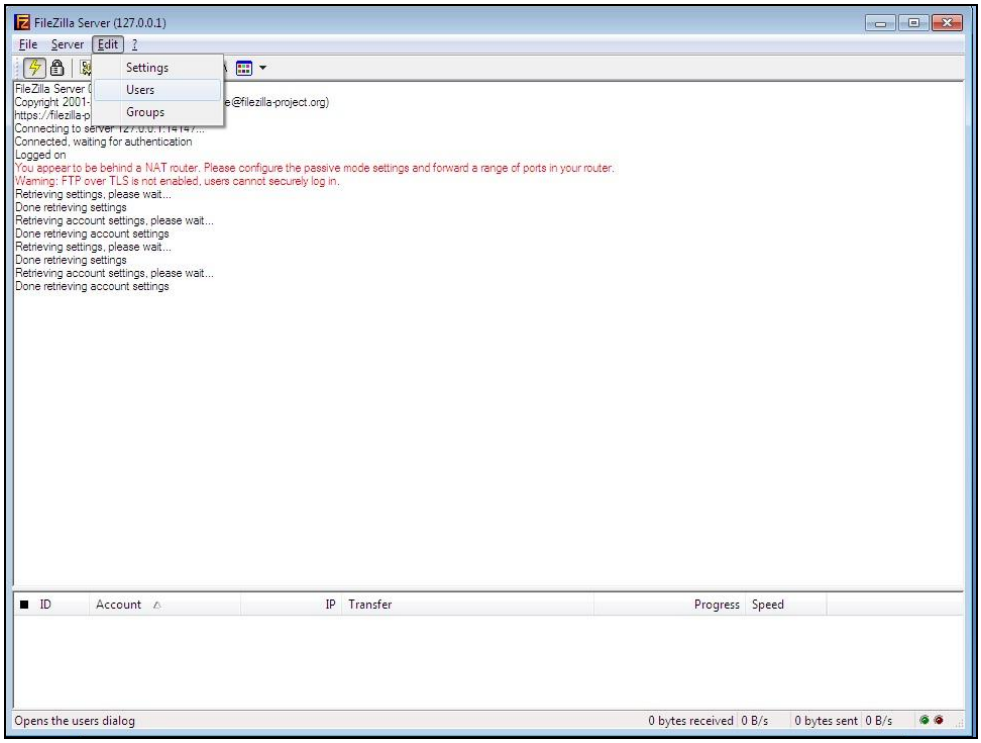
1. Download the FileZilla FTP server from below link,
   https://filezilla-project.org/download.php?type=server
2. Install downloaded FileZilla FTP server in Windows PC2 which is connected to AP through LAN.
3. Configure and run FTP server. Please refer the given below images for configuring and running FileZilla FTP server,

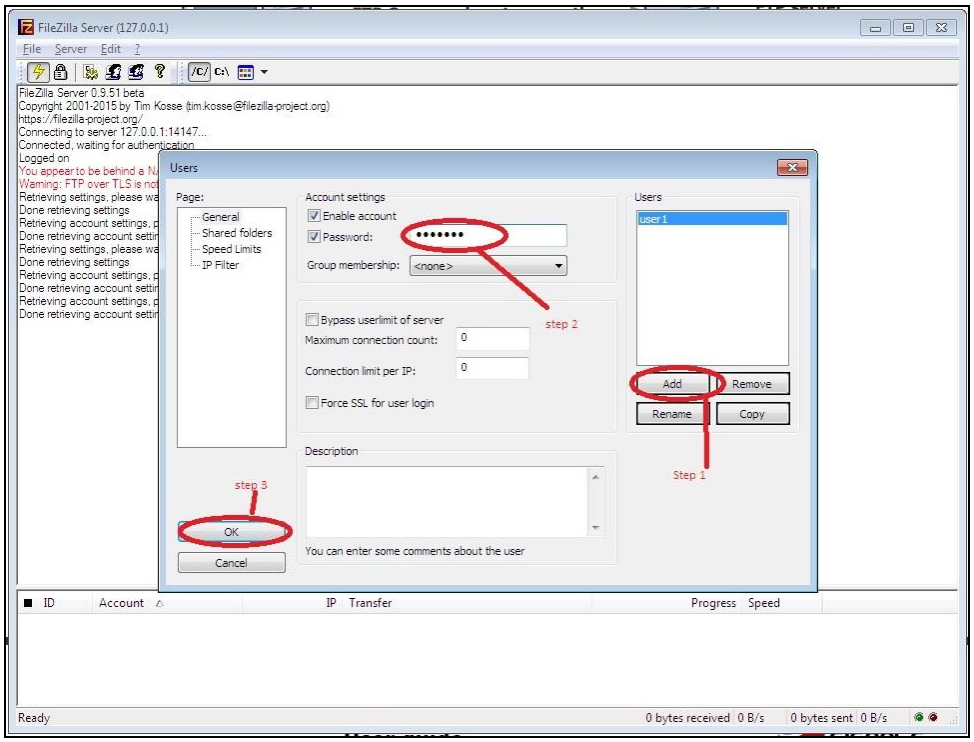
After installation, open FileZilla Server interface.

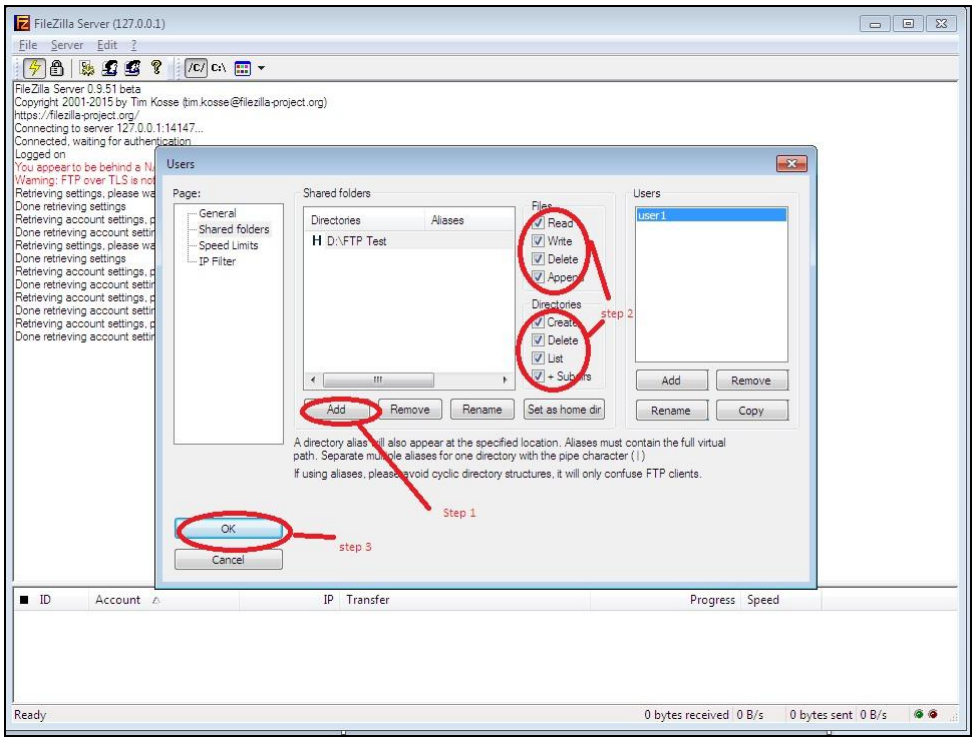


After opening FileZilla server interface, connect to server admin interface.

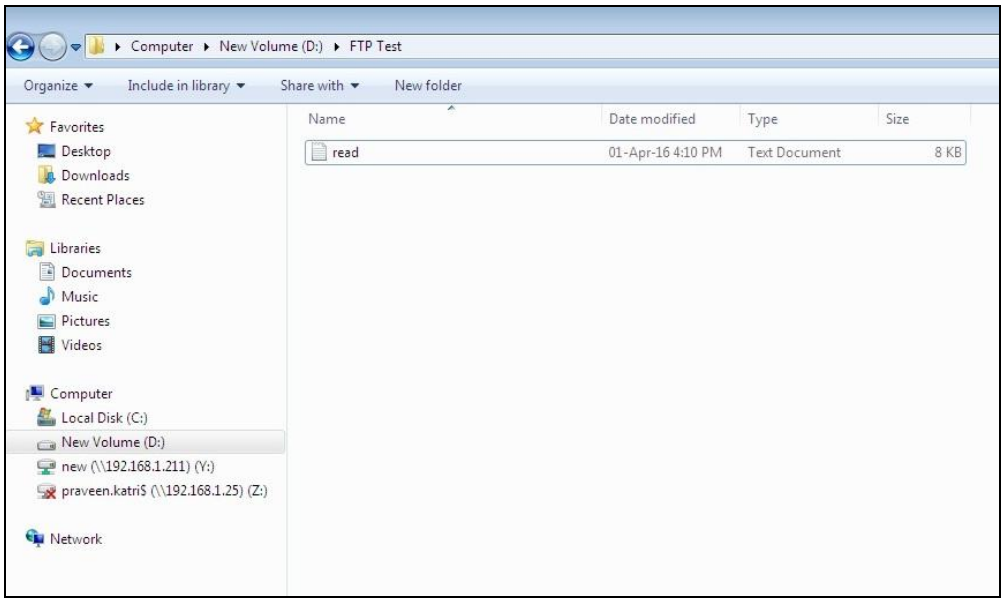
After connection with server, configure the user settings.







4. Place “read.txt” file in FTP directory (Ex: “D:\FTP Test\read.txt”).



2.2 Initializing the Application

2.2.1 SPI Interface

If User using SPI interface, Please refer the document ***sapis/platforms/spansion_MB9BF568NBGL/RS9113-WiSeConnect_SAPIS_Spansion_Project_User_guide.pdf*** for opening the ***ftp_client*** example in CooCox IDE.

2.2.2 UART/USB-CDC Interface

If User using UART interface, Please refer the document ***sapis/platforms/ windows_uart/RS9113-WiSeConnect_SAPIS_Windows_Project_UserGuide.pdf*** for opening the ***ftp_client*** example in Dev-C++ IDE

2.3 Configuring the Application

1. Open ***sapis/examples/wlan/ftp_client/ rsi_ftp_client.c*** file and update/modify the following macros,

SSID refers to the name of the Access point.

```
#define SSID                    "<ap name>"
```

SECURITY_TYPE refers to the type of security. In this application STA supports Open, WPA-PSK, WPA2-PSK securities.

Valid configuration is:

RSI_OPEN - For OPEN security mode

RSI_WPA - For WPA security mode

RSI_WPA2 - For WPA2 security mode

```
#define SECURITY_TYPE          RSI_OPEN
```

PSK refers to the secret key if the Access point configured in WPA-PSK/WPA2-PSK security modes.

```
#define PSK                    "<psk>"
```

FTP_SERVER_PORT port refers remote FTP server port number.

By default FileZilla Server runs on port number 21.

```
#define FTP_SERVER_PORT       21
```

SERVER_IP_ADDRESS refers remote peer IP address to connect with TCP server socket.

IP address should be in long format and in little endian byte order.

Example: To configure "192.168.0.100" as remote IP address, update the macro **SERVER_IP_ADDRESS** as **0x6400A8C0**.

```
#define SERVER_IP_ADDRESS       0x6400A8C0
```

FTP Server login username

```
#define FTP_SERVER_LOGIN_USERNAME    "username"
```

FTP Server login password

```
#define FTP_SERVER_LOGIN_PASSWORD    "password"
```

File to read which is on FTP server

Create a file on FTP server with the file name along with the path with data given below

```
#define FTP_FILE_TO_READ              "read.txt"
```

(Or)

```
#define FTP_FILE_TO_READ           "D:\FTP Test\read.txt"
```

FILE_CONTENT_LENGTH refers content length of the read file from FTP server (Ex: configure **FILE_CONTENT_LENGTH** >= Sizeof ("read.txt"))

```
#define FILE_CONTENT_LENGTH    10000
```

File name to create on FTP server and write the same content which is read from "read.txt"

```
#define FTP_FILE_TO_WRITE             "write.txt"
```

(Or)

```
#define FTP_FILE_TO_WRITE      "D:\FTP Test\write.txt"
```

To configure IP address

DHCP_MODE refers whether IP address configured through DHCP or STATIC

```
#define DHCP_MODE    1
```

Note: If user wants to configure STA IP address through DHCP then set **DHCP_MODE** to 1 and skip configuring the following **DEVICE_IP, GATEWAY** and **NETMASK** macros.

(Or)

If user wants to configure STA IP address through STATIC then set DHCP_MODE macro to "0" and configure following DEVICE_IP, GATEWAY and NETMASK macros.

IP address to be configured to the device in STA mode, it should be in long format and in little endian byte order.

Example: To configure "192.168.10.10" as IP address, update the macro **DEVICE_IP** as **0x0A0AA8C0**.

```
#define  DEVICE_IP          0X0A0AA8C0
```

IP address of the gateway should also be in long format and in little endian byte order

Example: To configure "192.168.10.1" as Gateway, update the macro GATEWAY as **0x010AA8C0**

```
#define  GATEWAY            0x010AA8C0
```

IP address of the network mask should also be in long format and in little endian byte order

Example: To configure "255.255.255.0" as network mask, update the macro **NETMASK** as **0x00FFFFFF**

```
#define   NETMASK            0x00FFFFFF
```

The following parameters need to be configure if OS is used.

WLAN task priority is given and this should be of low priority

```
#define RSI_WLAN_TASK_PRIORITY      1
```

Driver task priority is given and this should be of highest priority

```
#define RSI_DRIVER_TASK_PRIORITY    1
```

WLAN Task stack size is configured by this macro

```
#define RSI_WLAN_TASK_STACK_SIZE    500
```

Driver Task stack size is configured by this macro

```
#define RSI_DRIVER_TASK_STACK_SIZE  500
```

2. Open ***sapis/include/rsi_wlan_config.h*** file and update/modify following macros :

```
#define CONCURRENT_MODE             RSI_DISABLE
#define RSI_FEATURE_BIT_MAP         FEAT_SECURITY_OPEN
#define RSI_TCP_IP_BYPASS           RSI_DISABLE
#define RSI_TCP_IP_FEATURE_BIT_MAP (TCP_IP_FEAT_DHCPV4_CLIENT
                                    | TCP_IP_FEAT_FTP_CLIENT)
#define  RSI_CUSTOM_FEATURE_BIT_MAP  0
#define  RSI_BAND                    RSI_BAND_2P4GHZ
```

2.4 Executing the Application

1. Configure the Access point in OPEN/WPA-PSK/WPA2-PSK mode to connect WiSeConnect device in STA mode.
2. Run FTP server on Windows PC2 and place "read.txt" file in the FTP directory

3. **SPI Interface**

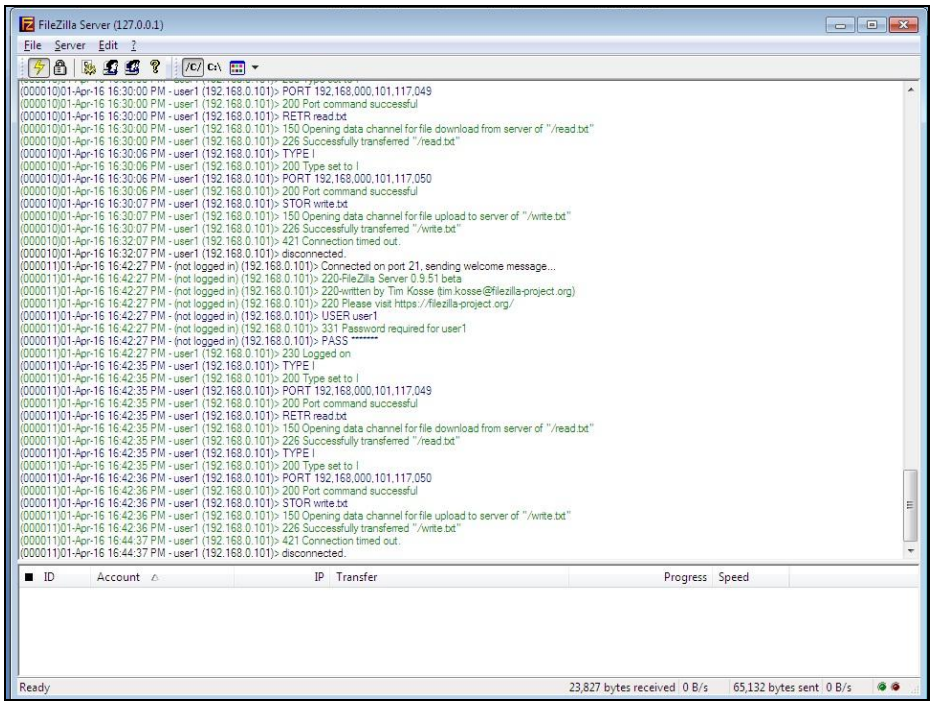
If User using SPI interface, Please refer the document ***sapis/platforms/spansion_MB9BF568NBGL/RS9113-WiSeConnect_SAPIS_Spansion_Project_User_guide.pdf*** for executing the ***ftp_client*** example in CooCox IDE.

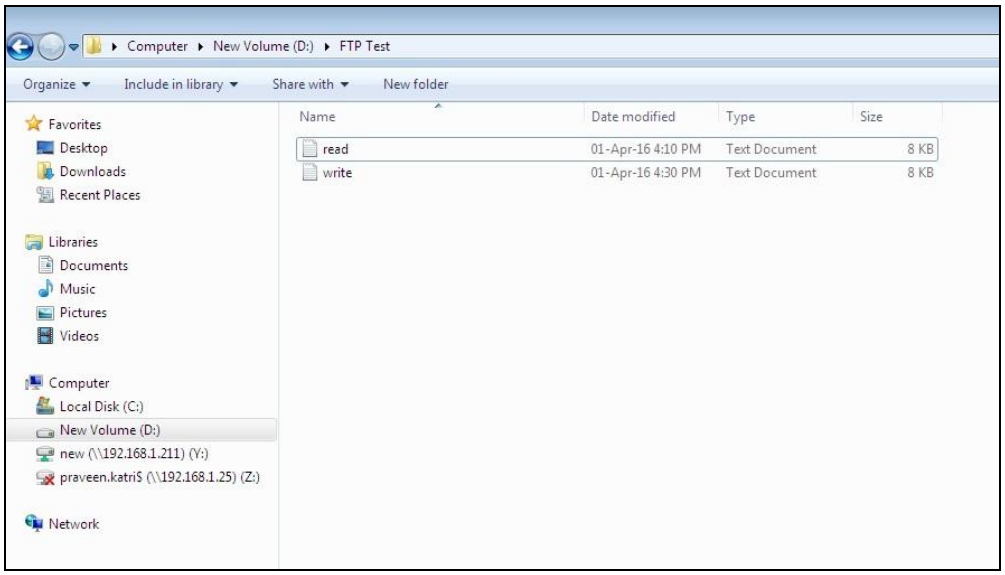
4. **UART/USB-CDC Interface**

If User using UART interface, Please refer the document ***sapis/platforms/windows_uart/RS9113-WiSeConnect_SAPIS_Windows_Project_UserGuide.pdf*** for executing the ***ftp_client*** example in Dev-C++ IDE

5. After the program gets executed, WiSeConnect Device will be connected to same Access point having the configuration same as that of in the application and get IP
6. After successful connection with the Access Point, WiSeConnect device connects to FTP server and reads the file content of given file ("**read.txt**") and creates a file name "**write.txt**" in FTP directory and writes the same content which is read from "**read.txt**". After successful file write WiSeConnect device disconnects from FTP server.

Please refer the given below images for message exchanges with FTP server and for read and write files,




Computer ▸ New Volume (D:) ▸ FTP Test
Organize
Include in library
Share with
New folder
Favorites
Desktop
Downloads
Recent Places
Libraries
Documents
Music
Pictures
Videos
Computer
Local Disk (C:)
New Volume (D:)
new (\\192.168.1.211) (Y:)
praveen.katri$ (\\192.168.1.25) (Z:)
Network
Name
Date modified
Type
Size
read
01-Apr-16 4:10 PM
Text Document
8 KB
write
01-Apr-16 4:30 PM
Text Document
8 KB