

# **AN1284: RS9116W Throughput Application Note**

Version 1.3

October 21, 2021

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>3</b>
<b>2</b>	<b>Hardware and Software Setup .....</b>	<b>4</b>
<b>3</b>	<b>Setup Diagram.....</b>	<b>5</b>
<b>4</b>	<b>Setup Description .....</b>	<b>6</b>
<b>5</b>	<b>Execution Steps .....</b>	<b>7</b>
5.1	Throughput Application Using SPI Interface .....	7
5.2	User Datagram Protocol (UDP).....	8
5.2.1	UDP Tx.....	8
5.2.2	UDP Rx .....	9
5.3	Transmission Control Protocol (TCP) .....	11
5.3.1	TCP Tx.....	11
5.3.2	TCP Rx.....	13
5.4	Wireshark I/O graphs .....	15
5.4.1	UDP Tx.....	15
5.4.2	UDP Rx.....	15
5.4.3	TCP Tx.....	16
5.4.4	TCP Rx .....	16
<b>6</b>	<b>Summary.....</b>	<b>17</b>
<b>7</b>	<b>References and Related Documentation.....</b>	<b>18</b>
<b>8</b>	<b>Troubleshooting.....</b>	<b>19</b>
<b>9</b>	<b>Revision History.....</b>	<b>20</b>

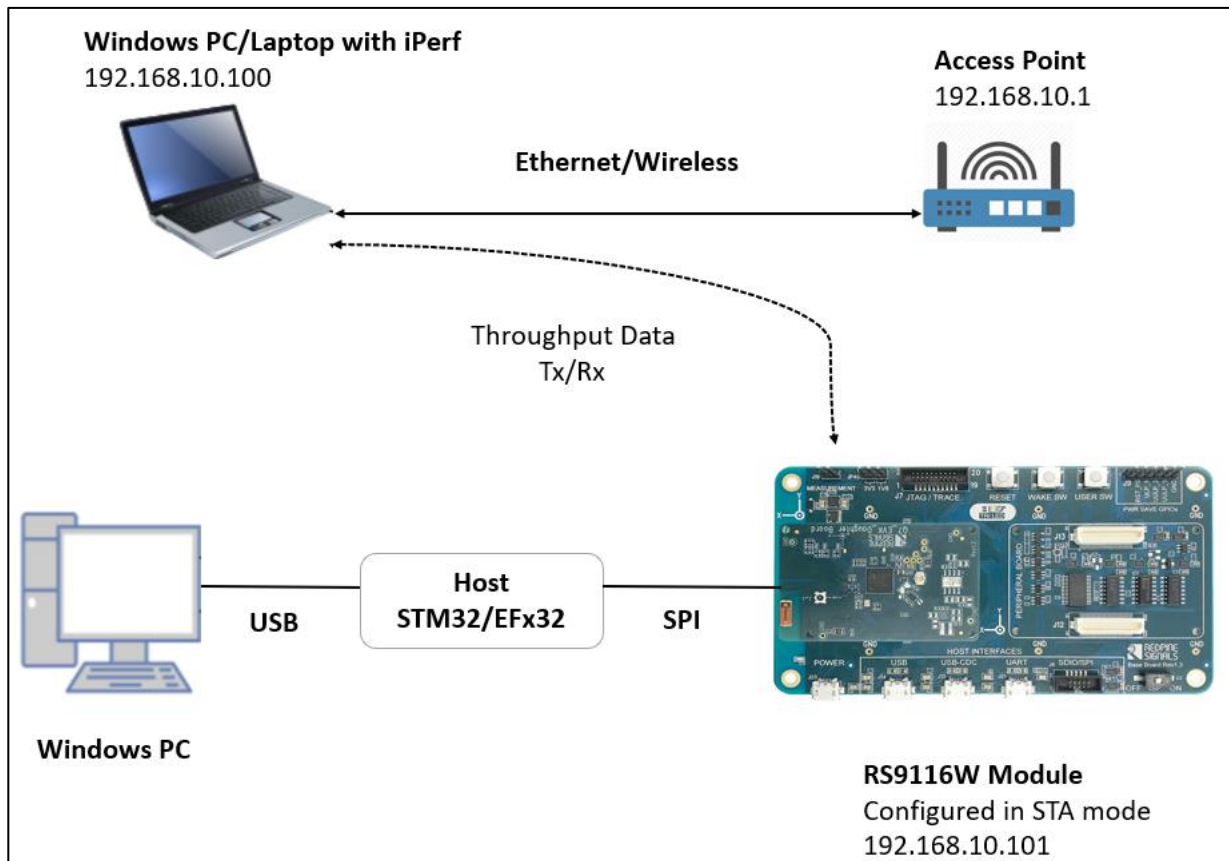
## 1 Introduction

In wireless communications, the term 'throughput' is defined as the number of data units transferred within a specified amount of time over a communication channel. In general, throughput is measured in bits/s or bps i.e., number of bits transferred in one second. This Application Note describes how data throughput performance of the RS9116 WiSeConnect module can be measured over the network.

## 2 Hardware and Software Setup

- Silicon Labs RS9116X-SB-EVK1/RS9116X-DB-EVK1
- Download the latest RS9116 WiSeConnect release package from the website <https://www.silabs.com/development-tools/wireless/wi-fi/rs9116x-sb-evk-development-kit> (under Software & Tools tab) and load the latest firmware accordingly
- Access point (AP)
- iPerf application  
**Note:** iPerf version 2.0.9 is used to verify this example. Download link: <https://iperf.fr/iperf-download.php#windows>
- Tera Term software, download link: <https://osdn.net/projects/ttssh2/releases/>
- Ethernet Cable

### 3 Setup Diagram



## 4 Setup Description

Many applications need to transfer a burst of data quickly over their Wi-Fi link before returning to their sleep or wait state. The User would therefore like to measure the throughput that can be sustained in their device. This document helps to measure the TCP and UDP transmit (Tx) and receive (Rx) throughput of RS9116 using SAPIs on the host MCU. Here the RS9116 is configured in Station mode and would connect to an Access Point. The RS9116 WiSeConnect module would then connect to an iPerf Server or Client running on a PC connected to the same Access Point.

To get the maximum possible throughput:

- Configure the RS9116 WiSeConnect module in '384k memory configuration mode' and 'enable aggregation' via opermode command.
- Configure the RS9116 WiSeConnect module SoC clock to 160MHz
- Enable the TCP Window division factor
- For more info on above configurations, refer to <https://docs.silabs.com/rs9116-wisecconnect/latest/wifibt-wc-sapi-reference/opermode>
- Enable the PLL mode in feature frame

All the above parameters are by default set in `rsi_wlan_config.h` file which you can find at `\examples\featured\wlan_throughput` in the release package.

To achieve the maximum throughput values, the packet lengths for different networking protocols like TCP, UDP and, SSL are by default set to maximum in the `rsi_throughput_app.c` file which you can find at `\examples\featured\wlan_throughput` in the release package. Throughput values would differ if the below packet lengths are reduced.

- `TCP_BUFF_SIZE` 1460
- `UDP_BUFF_SIZE` 1470
- `SSL_BUFF_SIZE` 1370

For SPI Interface, in addition to the above setting, it is recommended to configure the MCU at the best possible SPI Clock, and also to use DMA for SPI communication. The example in this Application Note is run with a SPI clock of 40 MHz with STM32 as host MCU.

## 5 Execution Steps

### 5.1 Throughput Application Using SPI Interface

Note:

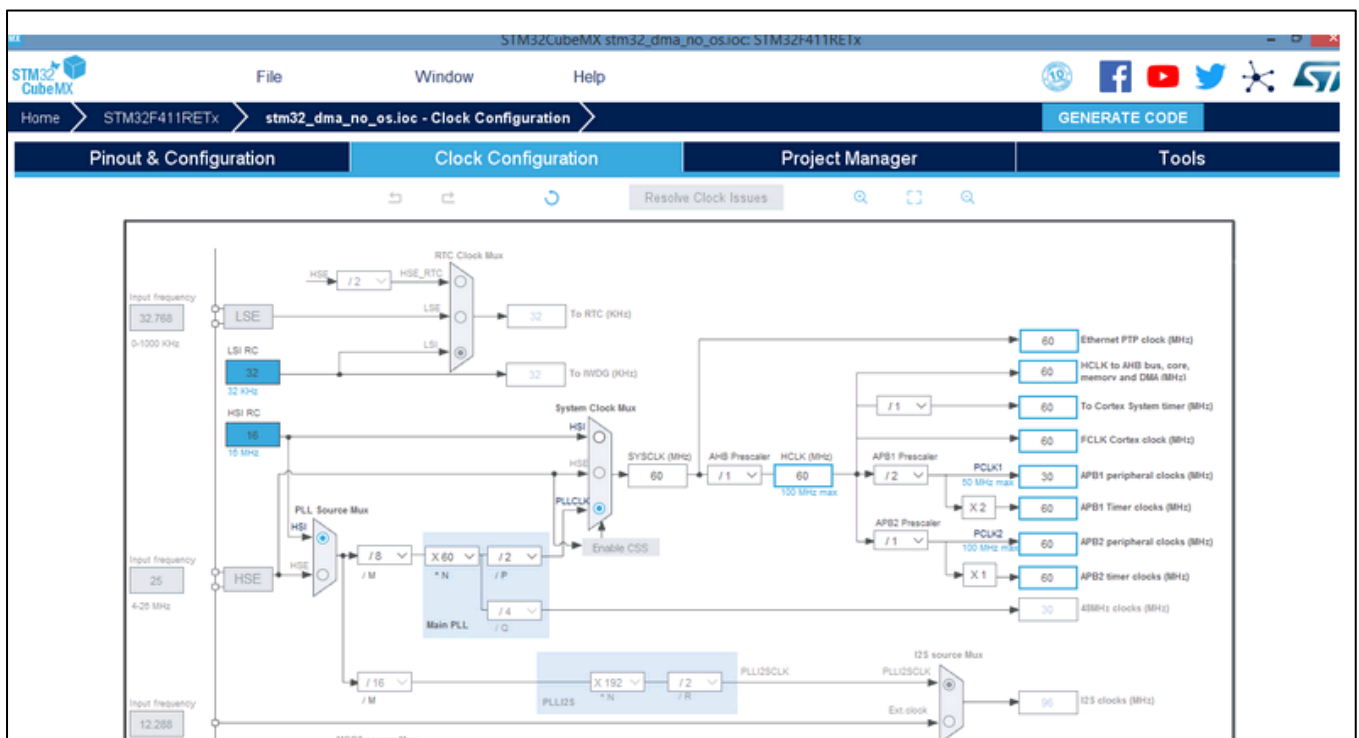
A reference project for STM32 and EFX32 using the SPI interface is available in the release package.

Refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-getting-started-with-stm32/> for connection b/w STM32 and RS9116 WiSeConnect Module

Refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-getting-started-with-efx32/> for connection b/w EFX32 and RS9116 WiSeConnect Module

For example, refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-featured-examples/wlan-throughput-readme>.

1. To achieve the best throughput using STM32, operate the MCU at the best possible SPI peripheral clock. This can be configured in project settings.
2. For STM MCUs, the CUBEMX tool is used to generate the project with the required settings.
3. The image below describes the best possible SPI Clock Configurations for STM32



4. If the MCU has the provision to configure the hardware chip select, it can be enabled to get better throughput results.

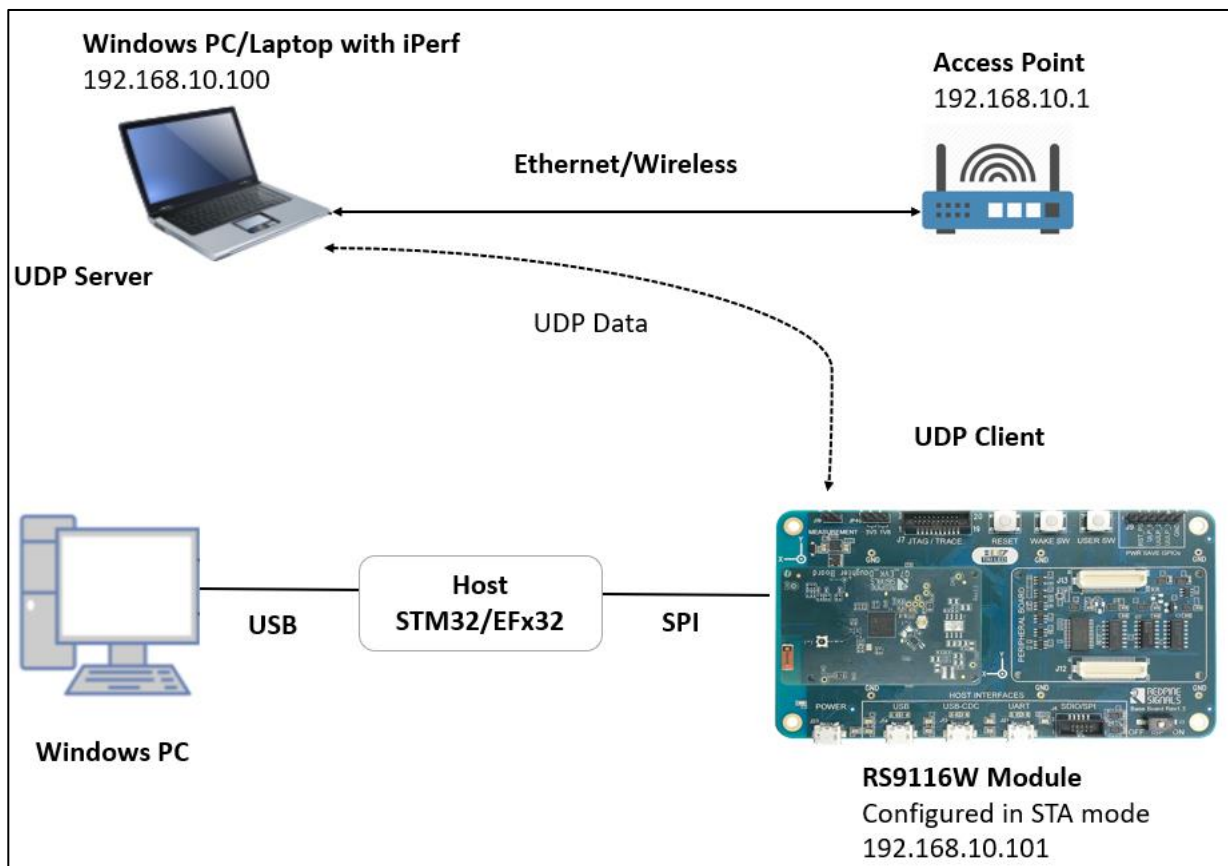
Note: Using the above SPI clock configuration, we can achieve throughputs of up to 22 Mbps.

## 5.2 User Datagram Protocol (UDP)

UDP is a simple, connectionless Internet protocol wherein error-checking and recovery services are not required. It is a lightweight protocol with no support for handshaking, ordering, error recovery etc., and is usually used in protocols that stream music or video. UDP is a preferred protocol where reliability can be traded-off for low end-to-end delay experienced by the users. With UDP, there is no overhead for opening a connection, maintaining a connection, or terminating a connection. Data is continuously sent to the recipient, whether they receive it or not.

### 5.2.1 UDP Tx

1. To measure UDP Tx throughput, the RS9116 module is configured as a UDP client. It connects to an Access Point (whose IP address is A.B.C.E - say 192.168.10.1 in this case) and receives an IP address A.B.C.D (say 192.168.10.101 in this case).
2. Then, open a UDP server port using iPerf at remote computer with IP address A.B.C.F (say 192.168.10.100 in this case).



3. Use the following command to start UDP server on remote computer. You can see throughput in the command prompt window, where iPerf is running.

To measure UDP Tx throughput, configure the module as a UDP client. Open UDP server at the remote port.

```
iperf.exe -s -u -p <SERVER_PORT> -i 1
```

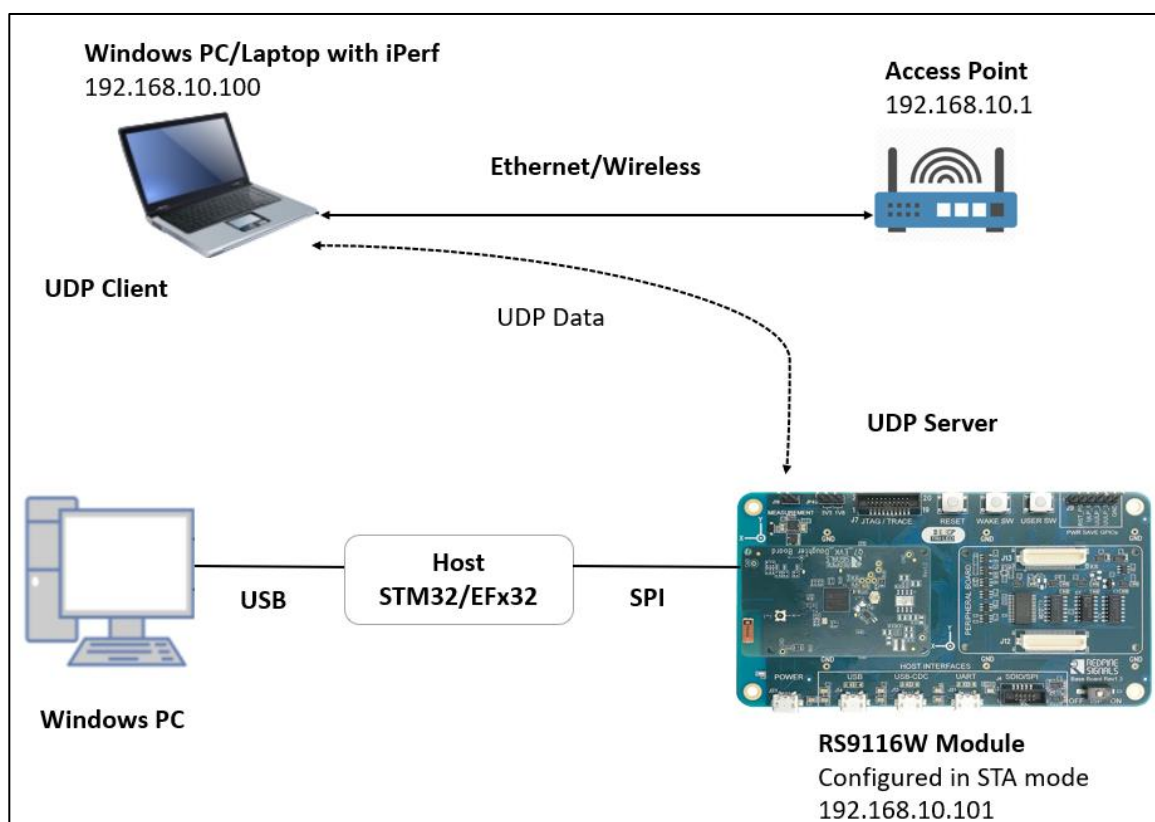


```
C:\Windows\System32\cmd.exe - iperf.exe -s -u -p 5001 -i 1

C:\iperf-2.0.9-win64>iperf.exe -s -u -p 5001 -i 1
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 3] local 192.168.1.100 port 5001 connected with 192.168.1.102 port 30000
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/TOTAL  Datagrams
[ 3] 0.0- 1.0 sec      2.74 MBytes  23.0 Mbits/sec   0.551 ms  64093/66051 (97%)
[ 3] 0.00-1.00 sec    1957 datagrams received out-of-order
[ 3] 1.0- 2.0 sec      2.11 MBytes  17.7 Mbits/sec   0.353 ms    0/    0 (nan%)
[ 3] 1.00-2.00 sec    1508 datagrams received out-of-order
[ 3] 2.0- 3.0 sec      2.74 MBytes  23.0 Mbits/sec   0.523 ms    0/    0 (nan%)
[ 3] 2.00-3.00 sec    1953 datagrams received out-of-order
[ 3] 3.0- 4.0 sec      2.74 MBytes  23.0 Mbits/sec   0.657 ms    0/    0 (nan%)
[ 3] 3.00-4.00 sec    1953 datagrams received out-of-order
[ 3] 4.0- 5.0 sec      2.74 MBytes  23.0 Mbits/sec   0.453 ms    0/    0 (nan%)
[ 3] 4.00-5.00 sec    1954 datagrams received out-of-order
[ 3] 5.0- 6.0 sec      2.74 MBytes  23.0 Mbits/sec   0.411 ms    0/    0 (nan%)
[ 3] 5.00-6.00 sec    1954 datagrams received out-of-order
[ 3] 6.0- 7.0 sec      2.31 MBytes  19.3 Mbits/sec   1.132 ms    0/    0 (nan%)
[ 3] 6.00-7.00 sec    1645 datagrams received out-of-order
[ 3] 7.0- 8.0 sec      2.77 MBytes  23.2 Mbits/sec   0.468 ms    0/    0 (nan%)
[ 3] 7.00-8.00 sec    1976 datagrams received out-of-order
[ 3] 8.0- 9.0 sec      2.45 MBytes  20.5 Mbits/sec   0.418 ms    0/    0 (nan%)
[ 3] 8.00-9.00 sec    1745 datagrams received out-of-order
[ 3] 9.0-10.0 sec      2.66 MBytes  22.3 Mbits/sec   0.332 ms    0/    0 (nan%)
[ 3] 9.00-10.00 sec   1896 datagrams received out-of-order
[ 3] 10.0-11.0 sec     2.73 MBytes  22.9 Mbits/sec   0.961 ms    0/    0 (nan%)
```

## 5.2.2 UDP Rx

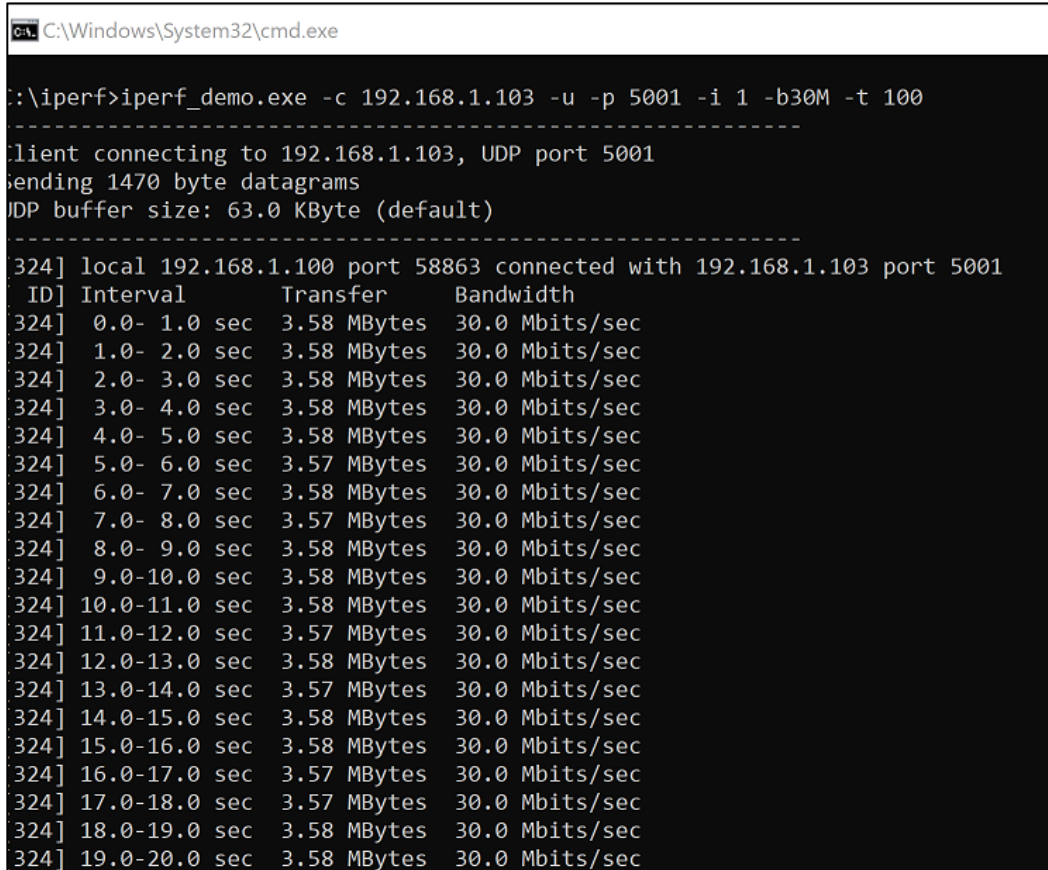
1. To measure UDP Rx throughput, RS9116 is configured as a UDP Server. It connects to an Access Point (whose IP address is A.B.C.E - say 192.168.10.1 in this case) and gets an IP A.B.C.D (say 192.168.10.101 in this case).
2. Then, open a UDP client port using iPerf at remote computer with IP address A.B.C.F (say 192.168.10.100 in this case).



3. Module port number is configured by default as 5001 in application. To know the module's IP address, see the prints on Tera Term serial terminal. Refer to the image below.
4. Use the following command to start measuring the throughput on UDP Client port. You can see throughput in the command prompt window where iPerf is running.

To measure UDP Rx throughput, configure the module as a UDP server. Open UDP client at the remote port.

```
iperf.exe -c <Module_IP> -u -p <Module_Port> -i 1 -b <Bandwidth>
```



```
C:\Windows\System32\cmd.exe

C:\iperf>iperf_demo.exe -c 192.168.1.103 -u -p 5001 -i 1 -b30M -t 100
-----
Client connecting to 192.168.1.103, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 63.0 KByte (default)
-----
[ 324] local 192.168.1.100 port 58863 connected with 192.168.1.103 port 5001
[ 324] ID Interval Transfer Bandwidth
[ 324] 0.0- 1.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 1.0- 2.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 2.0- 3.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 3.0- 4.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 4.0- 5.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 5.0- 6.0 sec 3.57 MBytes 30.0 Mbits/sec
[ 324] 6.0- 7.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 7.0- 8.0 sec 3.57 MBytes 30.0 Mbits/sec
[ 324] 8.0- 9.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 9.0-10.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 10.0-11.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 11.0-12.0 sec 3.57 MBytes 30.0 Mbits/sec
[ 324] 12.0-13.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 13.0-14.0 sec 3.57 MBytes 30.0 Mbits/sec
[ 324] 14.0-15.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 15.0-16.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 16.0-17.0 sec 3.57 MBytes 30.0 Mbits/sec
[ 324] 17.0-18.0 sec 3.57 MBytes 30.0 Mbits/sec
[ 324] 18.0-19.0 sec 3.58 MBytes 30.0 Mbits/sec
[ 324] 19.0-20.0 sec 3.58 MBytes 30.0 Mbits/sec
```

```

COM33 - Tera Term VT
File Edit Setup Control Window Help

Wireless Initialization Success
Send Feature Frame Success
WLAN AP Scan Success
WLAN AP Connect Success
IP Config Success
RSI_STA IP ADDR: 192.168.1.103
UDP_RX TEST
Socket Create Success
Bind Success
UDP RX start
UDP RX completed
throughput in mbps=23.520000
Time taken in sec: 5

```

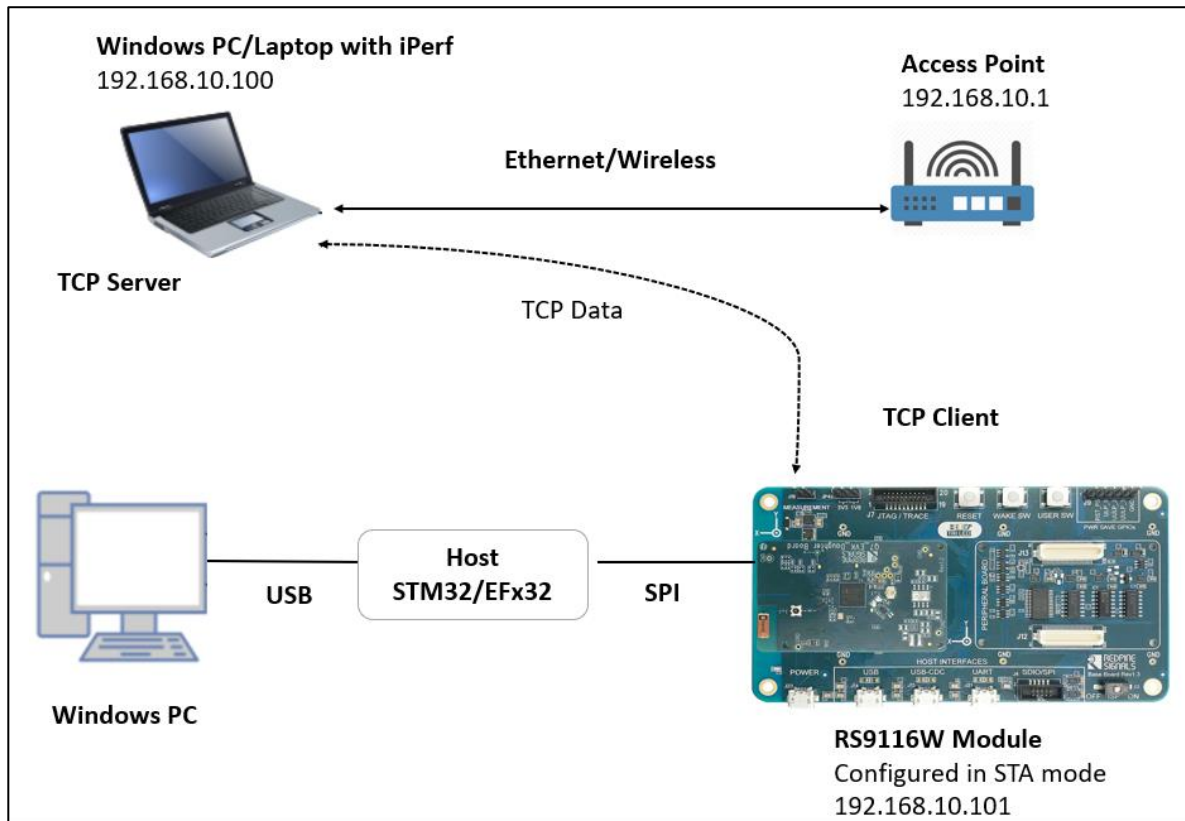
### 5.3 Transmission Control Protocol (TCP)

TCP is connection-oriented, meaning once a connection has been established, data can be transmitted in two directions. TCP has built-in systems to check for errors and to guarantee data will be delivered in the order it was sent, making it the perfect protocol for transferring information like still images, data files, and web pages.

TCP throughput is the rate at which the data is successfully delivered over a TCP connection. It is an important metric to measure the quality of a network connection. TCP protocol integrates a mechanism that checks that all packets are correctly delivered. This mechanism is called acknowledgment, it consists of having the receiver transmit a specific packet or flag to the sender to confirm the proper reception of a packet.

#### 5.3.1 TCP Tx

1. By default, in the application throughput type is configured as TCP Tx



2. To measure TCP Tx throughput, the RS9116 module is configured as a Wi-Fi station running a TCP Client and connects to an Access Point (whose IP address is A.B.C.E - say 192.168.10.1 in this case) and RS9116 gets an IP A.B.C.D (say 192.168.10.101 in this case).
3. Then, open TCP server port using iPerf at remote computer with IP address A.B.C.F (say 192.168.10.100 in this case).
4. Use the following iPerf commands on same/remote computer. Here we are using different PC to run iPerf server.

To measure TCP Tx throughput, configure the module as a TCP client. Open the TCP server at the remote port.

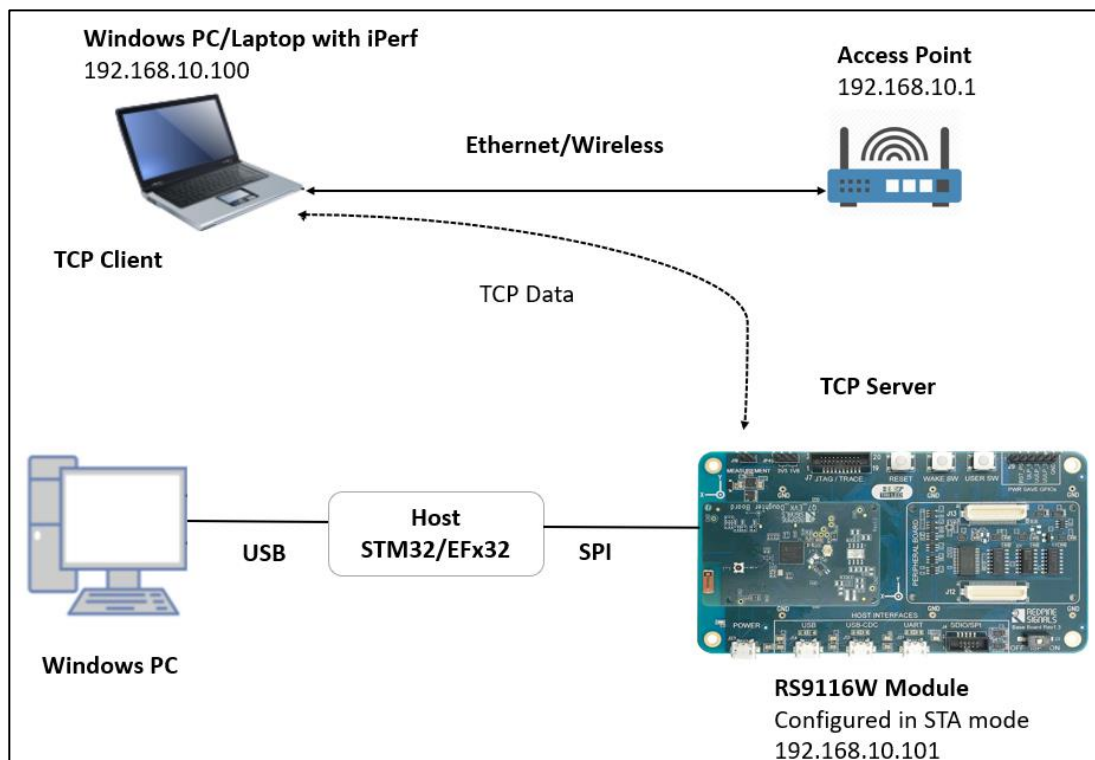
```
iperf.exe -s -p <SERVER_PORT> -i 1
```

```
C:\Windows\System32\cmd.exe - iperf.exe -s -p 5001 -i 1
Microsoft Windows [Version 10.0.19042.1083]
(c) Microsoft Corporation. All rights reserved.

C:\iperf-2.0.9-win64>iperf.exe -s -p 5001 -i 1
-----
Server listening on TCP port 5001
TCP window size: 208 KByte (default)
-----
[ 4] local 192.168.1.100 port 5001 connected with 192.168.1.102 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 4] 0.0- 1.0 sec  2.65 MBytes 22.2 Mbits/sec
[ 4] 1.0- 2.0 sec  2.64 MBytes 22.2 Mbits/sec
[ 4] 2.0- 3.0 sec  2.63 MBytes 22.1 Mbits/sec
[ 4] 3.0- 4.0 sec  2.65 MBytes 22.2 Mbits/sec
[ 4] 4.0- 5.0 sec  2.67 MBytes 22.4 Mbits/sec
[ 4] 5.0- 6.0 sec  2.64 MBytes 22.1 Mbits/sec
[ 4] 6.0- 7.0 sec  2.61 MBytes 21.9 Mbits/sec
[ 4] 7.0- 8.0 sec  2.60 MBytes 21.8 Mbits/sec
[ 4] 8.0- 9.0 sec  2.56 MBytes 21.5 Mbits/sec
[ 4] 9.0-10.0 sec  2.58 MBytes 21.6 Mbits/sec
[ 4] 10.0-11.0 sec 2.59 MBytes 21.7 Mbits/sec
[ 4] 11.0-12.0 sec 2.53 MBytes 21.2 Mbits/sec
[ 4] 12.0-13.0 sec 2.51 MBytes 21.0 Mbits/sec
[ 4] 13.0-14.0 sec 2.56 MBytes 21.5 Mbits/sec
[ 4] 14.0-15.0 sec 2.54 MBytes 21.3 Mbits/sec
[ 4] 15.0-16.0 sec 2.52 MBytes 21.1 Mbits/sec
[ 4] 16.0-17.0 sec 2.62 MBytes 22.0 Mbits/sec
[ 4] 17.0-18.0 sec 2.44 MBytes 20.5 Mbits/sec
[ 4] 18.0-19.0 sec 2.56 MBytes 21.5 Mbits/sec
[ 4] 19.0-20.0 sec 2.57 MBytes 21.5 Mbits/sec
[ 4] 20.0-21.0 sec 2.52 MBytes 21.1 Mbits/sec
[ 4] 21.0-22.0 sec 2.57 MBytes 21.5 Mbits/sec
```

### 5.3.2 TCP Rx

1. To measure TCP Rx throughput, RS9116 module is configured as a TCP Server, connects to an Access Point (whose IP address is A.B.C.E - say 192.168.10.1 in this case) and gets an IP A.B.C.D (say 192.168.10.101 in this case).
2. Open a TCP client port using iPerf at same/remote computer. Here we are using different PC to run iPerf with IP address A.B.C.F (say 192.168.10.100 in this case).



3. Module port number is configured by default as 5001 in application. To know the module (RS9116's) IP address, you can see the prints on Tera Term serial terminal, refer to the image below.



4. Use the following command to start measuring the throughput on TCP Client port.

To measure TCP Rx throughput, configure the module as a TCP server. Open TCP client at the remote port.

```
iperf.exe -c <Module_IP> -p <module_PORT> -i 1 -t 100
```

```
Microsoft Windows [Version 10.0.19042.1083]
(c) Microsoft Corporation. All rights reserved.

C:\iperf-2.0.9-win64>iperf.exe -c 192.168.1.102 -p 5001 -i 1
-----
Client connecting to 192.168.1.102, TCP port 5001
TCP window size: 208 KByte (default)
-----
[ 3] local 192.168.1.100 port 56349 connected with 192.168.1.102 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 1.0 sec  2.75 MBytes 23.1 Mbits/sec
[ 3] 1.0- 2.0 sec  2.25 MBytes 18.9 Mbits/sec
[ 3] 2.0- 3.0 sec  2.50 MBytes 21.0 Mbits/sec
[ 3] 3.0- 4.0 sec  2.38 MBytes 19.9 Mbits/sec
[ 3] 4.0- 5.0 sec  2.38 MBytes 19.9 Mbits/sec
[ 3] 5.0- 6.0 sec  2.50 MBytes 21.0 Mbits/sec
[ 3] 6.0- 7.0 sec  2.50 MBytes 21.0 Mbits/sec
[ 3] 7.0- 8.0 sec  2.50 MBytes 21.0 Mbits/sec
[ 3] 8.0- 9.0 sec  2.50 MBytes 21.0 Mbits/sec
[ 3] 9.0-10.0 sec  2.38 MBytes 19.9 Mbits/sec
[ 3] 0.0-10.1 sec 24.8 MBytes 20.6 Mbits/sec

C:\iperf-2.0.9-win64>
```

```
VT COM33 - Tera Term VT
File Edit Setup Control Window Help

WLAN AP Connect Success
IP Config Success
RSI_STA IP ADDR: 192.168.1.103
TCP RX Test
Socket Create Success
Set Socket Option Success
Bind Success
Listen Success
Socket Accept Success
TCP RX started
TCP RX completed
throughput in mbps=19.405228
Time taken in sec: 10
```

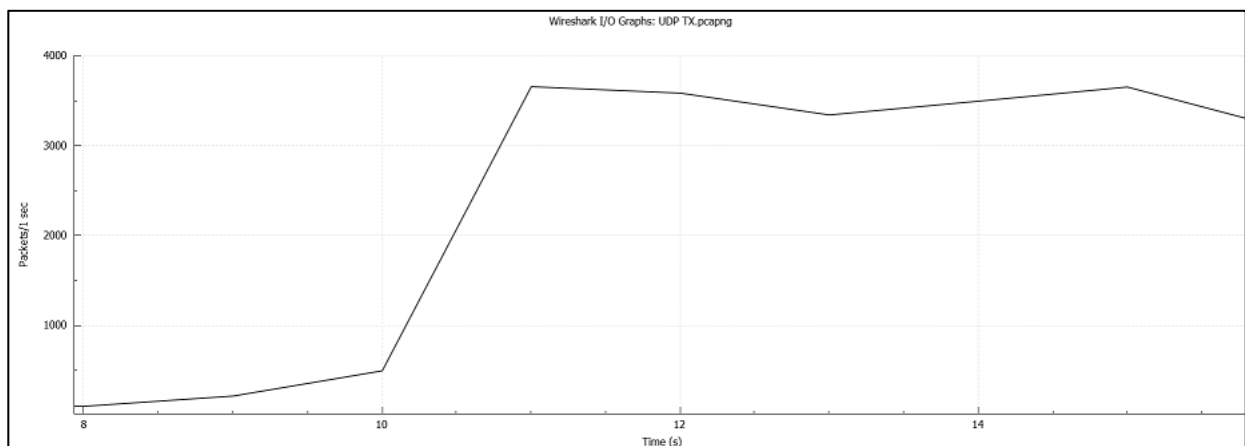
#### Note:

All the throughput numbers have been measured in an RF enclosure by running iPerf at the remote peer. Throughput numbers might vary depending on the environment and the wireless traffic on air. For example, in AP settings in 802.11 mode, 11bg throughputs are less when compared to 11n. It would also differ based on the host interface speeds, host processor capabilities (CPU frequency, RAM, etc.), wireless medium, physical obstacles, distance, etc. Make sure the AP and Server Running in PC are connected over ETHERNET Cable as it is generally defined as the maximum bandwidth without any packet loss. The SPI clock configurations may be changed depending on the MCU. For configurations, refer to the datasheet of the respective MCU.

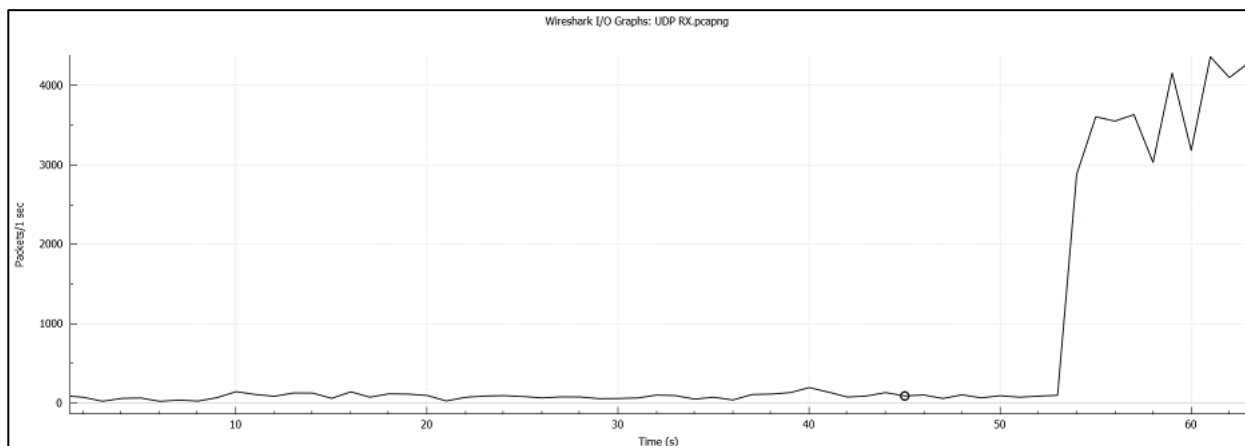
The above throughput measurement captures are with bare metal application. This application has been designed to work with FreeRTOS and bare metal configurations. By default, the application project files are configured with FreeRTOS enabled. To check with the bare metal, please go through the readme section - "Selecting Bare Metal" provided in the application, refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-featured-examples/wlan-throughput-readme#selecting-bare-metal>

## 5.4 Wireshark I/O graphs

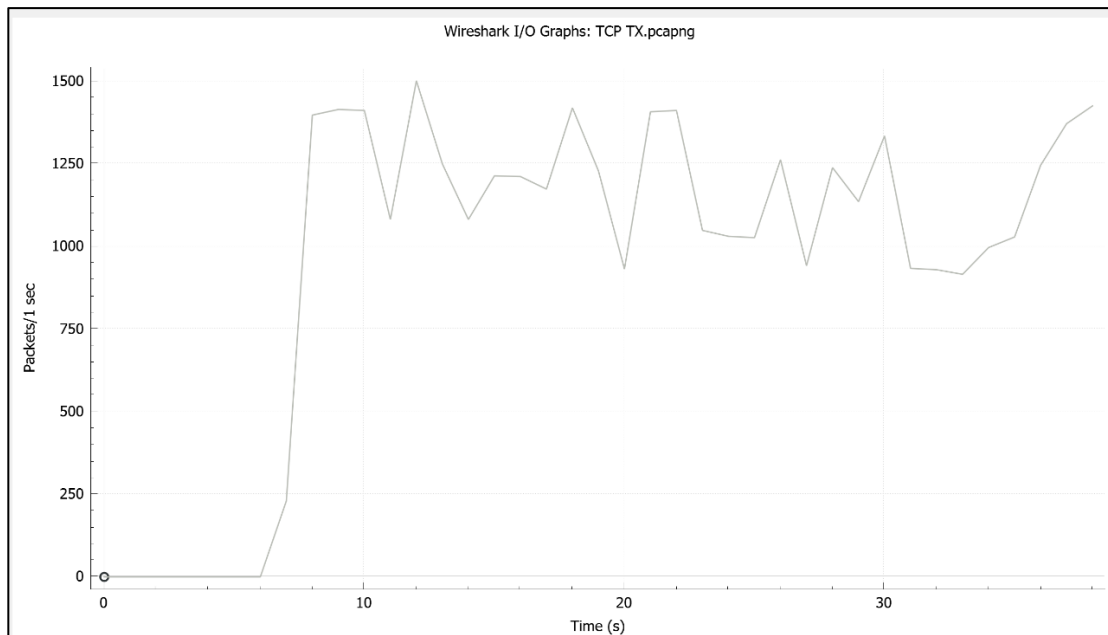
### 5.4.1 UDP Tx



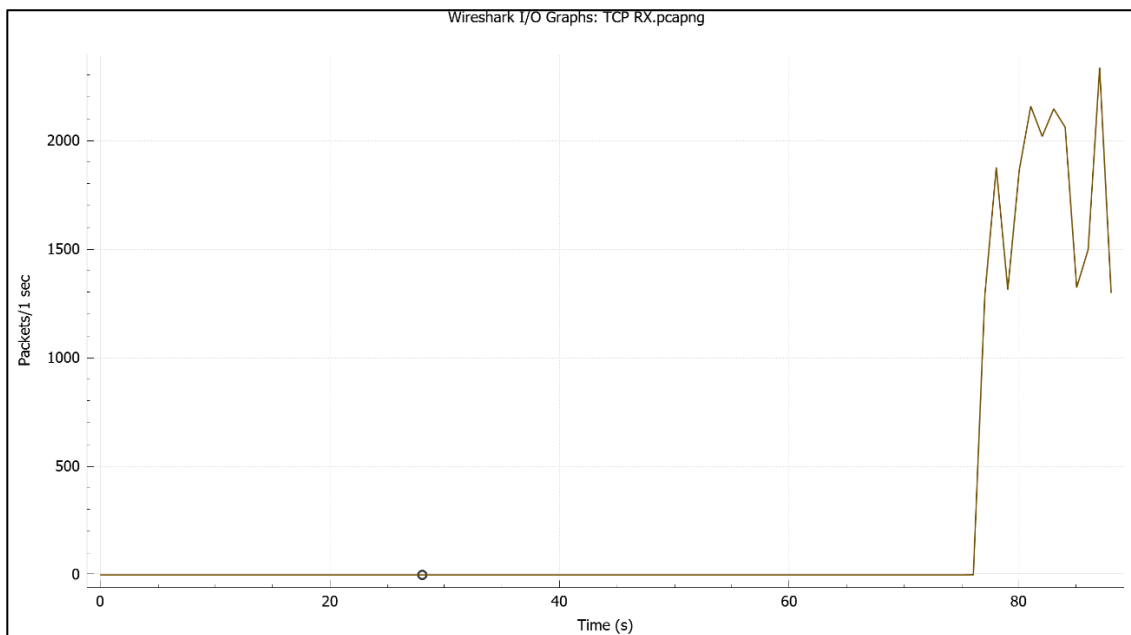
### 5.4.2 UDP Rx



### 5.4.3 TCP Tx



### 5.4.4 TCP Rx





## 6 Summary

By following the above procedure, we can configure the RS9116W module in station mode and can measure the maximum throughput by running iPerf at remote PC. Throughput applications are recommended to be run while there is minimal traffic.

## 7 References and Related Documentation

Refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-overview/> for all RS9116W documents

Refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-example-applications-overview/> for all the examples available in Latest SDK.

Refer to <https://docs.silabs.com/rs9116-wiseconnect/latest/wifibt-wc-release-notes/> for latest Release Notes to see the enhancements done for Throughput application.

## 8 Troubleshooting

1. Ensure that the AP and Server running in the PC are connected over an Ethernet cable.
2. Ensure that IP addresses are assigned to both client and server before running the iPerf test.
3. Ensure that the station is connected to the AP.
4. For running iPerf, the server should first begin listening, and then the client has to send the data.

## 9 Revision History.

Revision No.	Version No	Date	Changes
1	1.1	May, 2020	Initial version
2	1.2	Oct, 2020	Changes in the Application folder paths Updated the links and document names
3	v1.3	Aug, 2021	1. Updated the website links 2. Removed information regarding USB

# Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



**IoT Portfolio**  
[www.silabs.com/iot](http://www.silabs.com/iot)



**SW/HW**  
[www.silabs.com/simplicity](http://www.silabs.com/simplicity)



**Quality**  
[www.silabs.com/quality](http://www.silabs.com/quality)



**Support & Community**  
[www.silabs.com/community](http://www.silabs.com/community)

## Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and “Typical” parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice to the product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Without prior notification, Silicon Labs may update product firmware during the manufacturing process for security or reliability reasons. Such changes will not alter the specifications or the performance of the product. Silicon Labs shall have no liability for the consequences of use of the information supplied in this document. This document does not imply or expressly grant any license to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any FDA Class III devices, applications for which FDA premarket approval is required or Life Support Systems without the specific written consent of Silicon Labs. A “Life Support System” is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons. Silicon Labs disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of a Silicon Labs product in such unauthorized applications.

**Note: This content may contain offensive terminology that is now obsolete. Silicon Labs is replacing these terms with inclusive language wherever possible. For more information, visit [www.silabs.com/about-us/inclusive-lexicon-project](http://www.silabs.com/about-us/inclusive-lexicon-project)**

## Trademark Information

Silicon Laboratories Inc.<sup>®</sup>, Silicon Laboratories<sup>®</sup>, Silicon Labs<sup>®</sup>, SiLabs<sup>®</sup> and the Silicon Labs logo<sup>®</sup>, Bluegiga<sup>®</sup>, Bluegiga Logo<sup>®</sup>, EFM<sup>®</sup>, EFM32<sup>®</sup>, EFR, Ember<sup>®</sup>, Energy Micro, Energy Micro logo and combinations thereof, “the world’s most energy friendly microcontrollers”, Redpine Signals<sup>®</sup>, WiSeConnect, n-Link, ThreadArch<sup>®</sup>, EZLink<sup>®</sup>, EZRadio<sup>®</sup>, EZRadioPRO<sup>®</sup>, Gecko<sup>®</sup>, Gecko OS, Gecko OS Studio, Precision32<sup>®</sup>, Simplicity Studio<sup>®</sup>, Telegesis, the Telegesis Logo<sup>®</sup>, USBXpress<sup>®</sup>, Zentri, the Zentri logo and Zentri DMS, Z-Wave<sup>®</sup>, and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. Wi-Fi is a registered trademark of the Wi-Fi Alliance. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.  
400 West Cesar Chavez  
Austin, TX 78701  
USA

[www.silabs.com](http://www.silabs.com)