

WizFi360

Application – Throughput

Version 1.0

WIZnet Co.,Ltd

Copyright© 2019

History

Ver	Date	Description
1.0	Aug.2019	Initial version

Contents

1.	Test environment	4
2.	Using Serial command	5
3.	The result of UART Throughput	7
Appendix 1		8

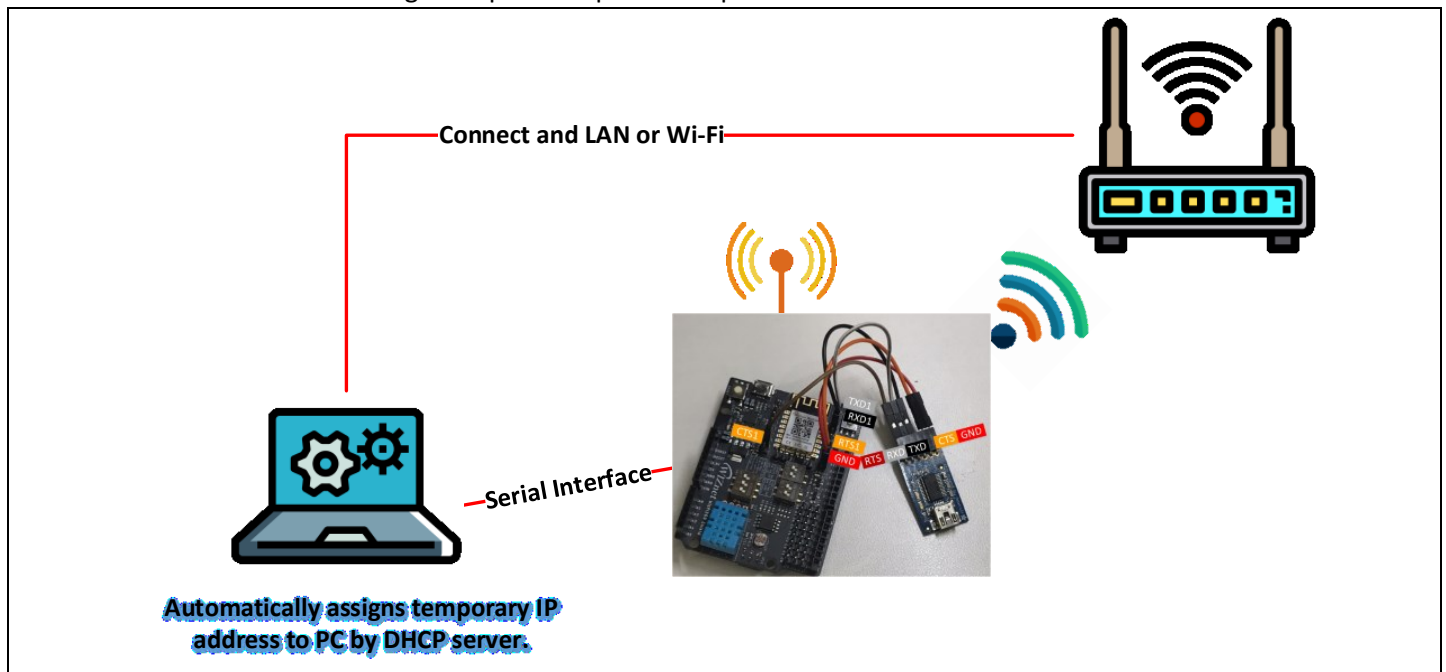
1. Test environment

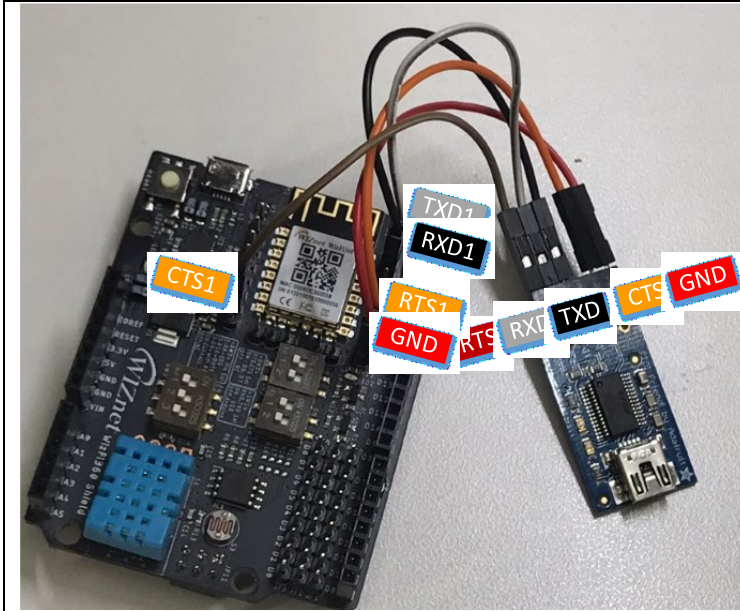
To UART throughput test, it controls using CTS / RTS is required.

- WizFi360 EVB or WizFi360io
- PC
- Serial Tool
 - o YAT Serial Tool(Data Mode)
 - o Python(Command Mode)
- 1Mbyte data file
- WiFi Router(exclude when it use in softAP mode)

When data mode uses, it sets RTS/CTS in flow control the using the YAT Serial Tool and it sets DTR as Data Read signal.

When command mode uses, it sets the AT+CIPSENDLBUF=2048 as maximum length of the data to be transmitted and it sends data of 2048 length. Repeat the previous operation.





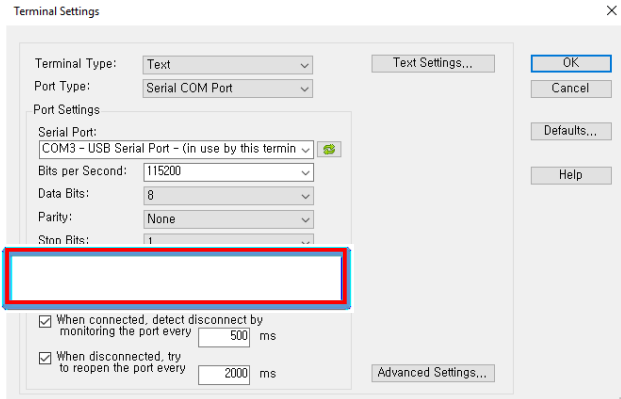
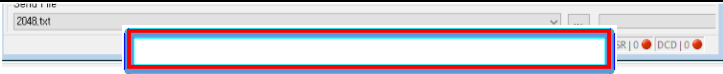
회로도 추가

2. Using Serial command

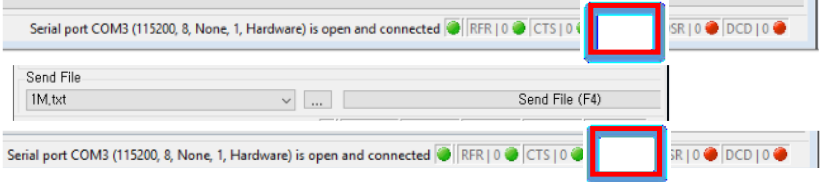
- Station Mode

AT command	Terminal
AT AT+CWMODE_CUR=1 AT+CWDHCP_CUR=1,1 AT+CWLAP AT+CWJAP_CUR="wizms1","maker0701" AT+CIPSTA_CUR?	<pre> AT<CR><LF> <CR><LF> OK<CR><LF> AT+CWMODE_CUR=1<CR><LF> <CR><LF> OK<CR><LF> AT+CWDHCP_CUR=1,1<CR><LF> <CR><LF> OK<CR><LF> AT+CWLAP<CR><LF> +CWLAP: (4,"DIR-815_Wiznet",-59,"",1)<CR><LF> +CWLAP: (0,"ESP_574935",-71,"",1)<CR><LF> +CWLAP: (3,"WIZnet_irina",-46,"",1)<CR><LF> +CWLAP: (3,"Matthew2.4",-63,"",2)<CR><LF> +CWLAP: (3,"rena",-46,"",3)<CR><LF> +CWLAP: (0,"iptime",-67,"",4)<CR><LF> +CWLAP: (3,"Dap",-63,"",5)<CR><LF> +CWLAP: (0,"ESP_577CC7",-67,"",6)<CR><LF> +CWLAP: (3,"wizms1",-63,"",6)<CR><LF> +CWLAP: (0,"Wizfi360",-69,"",6)<CR><LF> +CWLAP: (4,"DLINK-IPv6",-55,"",10)<CR><LF> +CWLAP: (0,"iptime",-59,"",11)<CR><LF> +CWLAP: (3,"WIZnet Scott",-51,"",11)<CR><LF> +CWLAP: (0,"WizFi360_A1B2D1",-69,"",11)<CR><LF> +CWLAP: (3,"Teddy_AP",-57,"",13)<CR><LF> <CR><LF> OK<CR><LF> AT+CWJAP_CUR="wizms1","maker0701"<CR><LF> WIFI_DISCONNECT<CR><LF> WIFI_CONNECTED<CR><LF> WIFI_GOT_IP<CR><LF> <CR><LF> OK<CR><LF> AT+CIPSTA_CUR?<CR><LF> +CIPSTA_CUR:ip:"192.168.1.120"<CR><LF> +CIPSTA_CUR:gateway:"192.168.1.1"<CR><LF> +CIPSTA_CUR:netmask:"255.255.255.0"<CR><LF> <CR><LF> OK<CR><LF> </pre>

- UART CTS/RTS Setting

AT command	Terminal
AT+CWUART_CUR = 115200,8,1,0,1	<pre>AT+UART_CUR=115200,8,1,0,1<CR><LF> <CR><LF> OK<CR><LF></pre>
Terminal Setting	
<ol style="list-style-type: none"> Pressing Ctrl+Shift+S and Open the Terminal Settings window You have to change the Hardware(RFR/CTS) in Flow Control 	
<ol style="list-style-type: none"> If you can see under the terminal window that the CTS/DTR is green 	

- TCP Client /Data mode

AT command	Terminal
<pre>AT+CIPSTART="TCP","192.168.100.27",5001 AT+CIPMODE=1 AT+CIPSEND</pre>	<pre>AT+CIPSTART="TCP","192.168.100.27",5001<CR><LF> CONNECT<CR><LF> <CR><LF> OK<CR><LF> AT+CIPMODE=1<CR><LF> <CR><LF> OK<CR><LF> AT+CIPSEND<CR><LF> <CR><LF> ></pre>
Terminal Setting	
<ol style="list-style-type: none"> When DTR is red, it sends the 1M.txt If you click the DTR, it changes the DTR is green and it is sending the data through Serial 	

- TCP Client / Command mode

AT command	Terminal
------------	----------

AT+CIPSTART="TCP","192.168.100.27",5001 AT+CIPMODE=0 AT+CIPSEND BUF=2048 Send the 1Mbyte.txt	AT+CIPSTART="TCP","192.168.100.27",5001<CR><LF> CONNECT<CR><LF> <CR><LF> OK<CR><LF> AT+CIPMODE=0<CR><LF> AT+CIPMODE=0<CR><LF> <CR><LF> OK<CR><LF> AT+CIPSEND BUF=2048<CR><LF> AT+CIPSEND BUF=2048<CR><LF> 1,0<CR><LF> <CR><LF> OK<CR><LF> >
---	---

3. The result of UART Throughput

PC sends the 1Mbyte through serial of WizFi360(UART1) and WizFi360 send the data to TCP Server.

Baud rate	Data mode		Command mode	
	Time	Speed(bit/s)	Time	Speed(bit/s)
115200	123s	66K		
921600	16.3s	502K		
1000000	14.9s	550K		
1250000	12.7s	645K		
1500000	10.5s	780K		
2000000	9.7s	845K		

We measured the time from the start of data transfer to the end of data transfer using the wireshark tool, see Appendix 1.

Appendix 1

Baud rate	Data mode	Command mode
115200	123s : 66Kbit/s <pre> 3823 122.860907 192.168.100.27 192.168.100.28 TCP 54 5001 → 52161 [ACK] Seq=1023825 Win=65535 Len=0 3824 122.865190 192.168.100.28 192.168.100.27 TCP 490 52161 → 5001 [PSH, ACK] Seq=1023825 Ack=1 Min=6144 Len=436 [TCP segment of 3825 122.906620 192.168.100.27 192.168.100.28 TCP 54 5001 → 52161 [ACK] Seq=1023461 Win=65099 Len=0 3826 122.912979 192.168.100.28 192.168.100.27 TCP 594 52161 → 5001 [PSH, ACK] Seq=1023461 Ack=1 Min=6144 Len=540 [TCP segment 3827 122.958838 192.168.100.27 192.168.100.28 TCP 54 5001 → 52161 [ACK] Seq=1024001 Win=64559 Len=0 </pre>	
921600	16.3s :502Kbit/s <pre> 2547 16.217822 192.168.100.28 192.168.100.27 TCP 8970 52165 → 5001 [ACK] Seq=1022161 Ack=1 Min=6144 Len=3204 [TCP segment of 2548 16.217860 192.168.100.27 192.168.100.28 TCP 54 5001 → 52165 [ACK] Seq=1023185 Win=65535 Len=0 2549 16.317118 192.168.100.28 192.168.100.27 TCP 870 52165 → 5001 [PSH, ACK] Seq=1023185 Ack=1 Min=6144 Len=816 [TCP segment 2550 16.357729 192.168.100.27 192.168.100.28 TCP 54 5001 → 52165 [ACK] Seq=1024001 Win=64719 Len=0 </pre>	
1000000	14.9s : 550Kbit/s <pre> 3868 14.774321 192.168.100.28 192.168.100.27 TCP 490 58128 → 5001 [PSH, ACK] Seq=1023125 Ack=1 Min=6144 Len=436 [TCP 3869 14.815213 192.168.100.27 192.168.100.28 TCP 54 5001 → 58128 [ACK] Seq=1023561 Win=65099 Len=0 3870 14.833495 192.168.100.28 192.168.100.27 TCP 494 58128 → 5001 [PSH, ACK] Seq=1023561 Ack=1 Min=6144 Len=440 [TCP 3871 14.859281 192.168.100.27 192.168.100.28 TCP 54 5001 → 58128 [ACK] Seq=1024001 Win=64659 Len=0 </pre>	
1250000	12.7s : 645Kbit/s <pre> 2863 12.592480 192.168.100.28 192.168.100.27 TCP 490 58136 → 5001 [PSH, ACK] Seq=1023185 Ack=1 Min=6144 Len=436 [T 2864 12.631883 192.168.100.27 192.168.100.28 TCP 54 5001 → 58136 [ACK] Seq=1023621 Win=65099 Len=0 2865 12.633959 192.168.100.28 192.168.100.27 TCP 434 58136 → 5001 [PSH, ACK] Seq=1023621 Ack=1 Min=6144 Len=380 [T 2866 12.674856 192.168.100.27 192.168.100.28 TCP 54 5001 → 58136 [ACK] Seq=1024001 Win=64719 Len=0 </pre>	
1500000	10.5s : 780Kbit/s <pre> 2242 10.389973 192.168.100.28 192.168.100.27 TCP 490 65021 → 5001 [PSH, ACK] Seq=1023074 Ack=1 Min=6144 Len=436 [TC 2244 10.430942 192.168.100.27 192.168.100.28 TCP 54 5001 → 65021 [ACK] Seq=1023518 Win=65099 Len=0 2245 10.445897 192.168.100.28 192.168.100.27 TCP 545 65021 → 5001 [PSH, ACK] Seq=1023518 Ack=1 Min=6144 Len=491 [TC 2247 10.486915 192.168.100.27 192.168.100.28 TCP 54 5001 → 65021 [ACK] Seq=1024001 Win=64608 Len=0 </pre>	
2000000	9.7s : 845Kbit/s <pre> 6316 9.646387 192.168.100.28 192.168.100.27 TCP 490 65031 → 5001 [PSH, ACK] Seq=1023245 Ack=1 Min=6144 Len=436 6317 9.686546 192.168.100.27 192.168.100.28 TCP 54 5001 → 65031 [ACK] Seq=1023681 Win=65099 Len=0 6318 9.698489 192.168.100.28 192.168.100.27 TCP 374 65031 → 5001 [PSH, ACK] Seq=1023681 Ack=1 Min=6144 Len=320 6319 9.731538 192.168.100.27 192.168.100.28 TCP 54 5001 → 65031 [ACK] Seq=1024001 Win=64779 Len=0 </pre>	