AT Command Version 2.2 Application Note

This document provides information for controlling Ameba through external UART.

Table of Contents

1 Introduction 4
1.1 Compile guide 4
1.2 Setup guide 4
1.3 Command description 5
1.4 AT command list 5
2 Common Command 7
2.1 AT – Test AT command ready 7
2.2 ATS? – List all AT command 7
2.3 ATSR – Restart module 8
2.4 ATSV – Query version info 8
2.5 ATSP – Set power saving mode 9
2.6 ATSE – Set UART echo mode and debug mode 10
2.7 ATSY – Factory Reset 11
2.8 ATSU – UART configuration 12
2.9 ATSW – Start Webserver 13
2.10 ATSO – OTA upgrade 14
2.11 ATSC – Choose Activated Image 14
2.12 ATSG – GPIO control 15
3 Wifi command 16
3.1 ATPW – Set wifi mode 16
3.2 ATPN – Connect to AP 17
3.3 ATWD - Disconnect from AP 18
3.4 ATWS - Scan AP 18
3.5 ATPA - Set AP mode 19
3.6 ATW? - Wifi information 20
3.7 ATPH - Set DHCP mode 20
3.8 ATPE - Set static IP for STA 21
3.9 ATPF - Set DHCP rule and gateway 22

3.10 ATPG - Set Auto connect 23
3.11 ATPM - Set MAC address 24
3.12 ATWQ - Start simple config 24
4 TCP/IP command 25
4.1 Compile guide 25
4.2 ATP0 – Get LWIP errno 25
4.3 ATPS – Create TCP/UDP Server 26
4.4 ATPC – Create TCP/UDP Client 28
4.5 ATPD – Close TCP or UDP connection 30
4.6 ATPT – Send data 32
4.7 ATPR – Receive data 34
4.8 ATPK – Set auto receive data mode 36
4.9 ATPI – Check network connection status 37
4.10 ATPP – PING Command 38
4.11 ATPU – Set transparent transmission mode 40
4.12 ATPL – Save translink and enable autolink 42

1 Introduction

1.1 Compile guide

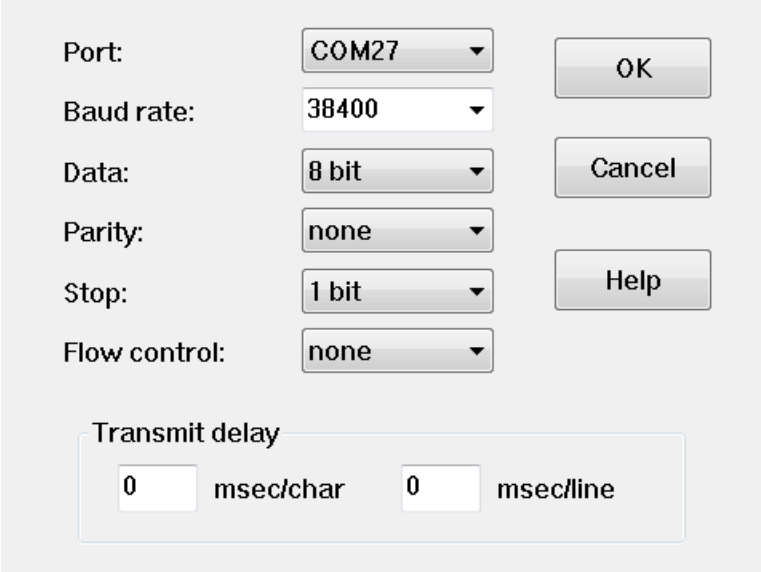
To use this version AT command, please configure **CONFIG_EXAMPLE_UART_ATCMD** to 1 in platform_opts.h.

1.2 Setup guide

To use this version AT command, it should use UART2 but not default log uart.

UART	UART2 TX	UART2 RX	UART2 RTS	UART2 CTS
Pin name	GPIOA_4	GPIOA_0	GPIOA_2	GPIOA_1

Teraterm console settings as below:



1.3 Command description

1. Comma (,), quotation marks (“ ”), square brackets ([ ]) and backslash (\) are used as delimiter in this version AT command, so if they are needed, use escape character “\”. For example, if need to input “[”, it should be “\[” instead.
2. Every normal command should end with “\n”, except data command (refer to section 4.6).

1.4 AT command list

Description	AT Command
Common command	
Test AT command ready	AT
Print all AT command	ATS?
Restart module	ATSR
Query version info	ATSV
Set power saving mode	ATSP
Set AT commands echo mode	ATSE
Factory Reset	ATSY
UART configuration	ATSU
Start web server	ATSW
OTA upgrade	ATSO
Choose activated image	ATSC
GPIO control	ATSG
Wifi command	
Set wifi mode	ATPW
Connect to AP (STA mode)	ATPN
Disconnect from AP	ATWD

Scan AP	ATWS
Set AP mode	ATPA
Wifi information	ATW?
Set DHCP mode	ATPH
Set static IP for STA	ATPE
Set static IP for AP, and DHCP rule	ATPF
Set Auto connect	ATPG
Set MAC address	ATPM
Start simple config	ATWQ
TCPIP command	
Check network connection status	ATPI
Get errno	ATP0
TCP/UDP Server	ATPS
TCP/UDP Client	ATPC
Close TCP or UDP connection	ATPD
Send packet	ATPT
Receive packet	ATPR
Enable auto receive data mode	ATPK
Ping	ATPP
Set transparent transmission mode	ATPU
Save translink and enable autolink	ATPL

2 Common Command

2.1 AT – Test AT command ready

AT	
Description	This command is used to test system boot successfully
Response	[AT] OK

2.2 ATS? – List all AT command

ATS?	
Description	This command will list all usable AT command
Response	[ATS?] <command list> [ATS?] OK [ATS?] ERROR: <error_no>
Error Number	1: get command list fail

2.3 ATSR – Restart module

ATSR	
Description	This command is used to restart the module
Response	[ATSR] OK

2.4 ATSV – Query version info

ATSV	
Description	This command is used to query module AT version as well as SDK version
Response	[ATSV] OK:<at-version>,<sdk-version>(<compile_time>) [ATSV] ERROR:<error_no>
Error Number	1: get version info fail

2.5 ATSP – Set power saving mode

ATSP=<mode>		
Description	This command is used to set module power saving mode	
Response	[ATSP] OK: <OS wakelock status (0/1)> [ATSP] ERROR:<error_no>	
Parameter	<mode>	a : acquire OS wakelock (OS sleep forbidden) r : release OS wakelock (OS sleep permission) ? : get OS wakelock status
Error Number	1: command format error 2: command parameter error	
Note	1. This module maybe not sleep immediately because other modules (Wi-Fi, SDIO, LOG_UART) may hold the wakelock, only when all module release its wakelock, the OS begin to sleep 2. when OS is sleeping, an extra gpio interrupt pin should be parallel to RX to wake up OS	

2.6 ATSE – Set UART echo mode and debug mode

ATSE=<echo>[,< debug mask >,< debug level >]		
Description	This command is used to enable/disable UART echo and set debug mask and level	
Response	[ATSE] OK	
Parameter	<echo>	0 : disable echo 1 : enable echo (default)
	[<debug mask>]	BIT 6: OS (default on) BIT 7: LWIP (default on) BIT 8: COMMON (default on)
	[<debug level>]	0: OFF 1: ALWAYS 2: ERROR (default) 3: WARNING 4: INFO
Error Number	1, 2: parameter number error 3: echo should be ‘0’ or ‘1’ only	
Note	Disable echo and debug message # ATSE=0,0x0,0x0	

2.7 ATSY – Factory Reset

ATSY	
Description	This command is used to clean flash data, module will restore to factory setting
Response	[ATSY] OK [ATSY] ERROR:<error_no>
Error Number	1: restore default data fail 2: restore default image fail
Note	System will reboot

2.8 ATSU – UART configuration

ATSU=<baudrate>,<databits>,<stopbits>,<parity>,<flowcontrol>,<configmode>		
Description	This command is used to setup uart mode	
Response	[ATSU] OK [ATSU] ERROR:<error_code>	
Parameter	<baudrate>	2400, 4800, 9600, 19200, 38400(default), 57600, 115200, 921600, 1152000
	<databits>	5: 5 bit data 6: 6 bit data 7: 7 bit data 8: 8 bit data (default)
	<stopbits>	1: 1 bit stop (default) 2: 2 bit stop
	<parity>	0: None parity (default) 1: Odd parity 2: Even parity
	<flowcontrol>	0: disable flowcontrol (default) 1: enable RTS and CTS
	<configmode>	0: set the current configuration and will not save to flash 1: save configuration to flash and take effect immediately 2: save configuration to flash and take effect after reboot
Error number	1: command format error 2: command parameter error	
Note		

2.9 ATSW – Start Webserver

ATSW=<mode>		
Description	This command is used to start and stop webserver, which is disabled in default FW.	
Response	[ATSW] OK [ATSW] ERROR:<error_code>	
Parameter	<mode>	c : create webserver s : stop webserver
Error number	1: command format error 2: command parameter error	
Note	Module should be configured as AP mode using command ATPA	

2.10 ATSO – OTA upgrade

ATSO=<ip>,<port>		
Description	This command is used to upgrade firmware	
Response	[ATSO] OK [ATSO] ERROR:<error_code>	
Parameter	<ip>	Download server ip address
	<port>	Download server port number
Error number	1: command format error 2: command parameter error	
Note	1: download server should run first 2: module should connect to the same network as download server	

2.11 ATSC – Choose Activated Image

ATSC=<image ID>		
Description	This command is used to choose the activated image	
Response	[ATSC] OK [ATSC] ERROR:<error_code>	
Parameter	<image ID>	0: default image 1: OTA upgrade image
Error number	1: command format error 2: command parameter error	
Note	System will reboot	

2.12 ATSG – GPIO control

ATSG=<R/W>,<PORT>[,<DATA>,<DIR>,<PULL>]		
Description	This command is used to control gpio pin	
Response	[ATSG] OK:<val> //val is the value read from gpio or write to gpio [ATSG] ERROR:<error_code>	
Parameter	<R/W>	"R": read gpio "W": write gpio
	<PORT>	Px_x, ex: PC_4
	[<DATA>]	0 or 1 when write gpio
	[<DIR>]	Pin direction: 0: PIN_INPUT 1: PIN_OUTPUT
	[<PULL>]	Pin mode: 0: PullNone/PullDefault 1: PullUp 2: PullDown 3: OpenDrain
Error number	1: command format error 2: command parameter error 3: invalid pin name	
Note		

3 Wifi command

3.1 ATPW – Set wifi mode

ATPW=<mode>		
Description	This command is used to set wifi mode, when executing ATPN and ATPA command must check mode first	
Response	[ATPW] OK [ATPW] ERROR:<error_no>	
Parameter	<mode>	1 : Station mode (default) 2 : AP mode 3 : Concurrent mode
Error Number	1: command format error 2: command parameter error	
Note	Concurrent mode must do ATPA first then ATPN	

3.2 ATPN – Connect to AP

ATPN=<ssid>,<pwd>[,<key_id>,<bssid>]		
Description	This command is used to connect to AP for station	
Response	[ATPN] OK [ATPN] ERROR:<error_code>	
Parameter	<ssid>	This parameter can't be empty Format: "ssid" Must add prefix '\' for special character(',' , '\' , '"', '[', ']')
	<pwd>	1. WPA/WPA2 : length is 8~64 2. WEP : length is 5 or 13
	[<key_id>]	For WEP security, must be 0~3. If not set, it will use id 0 as default
	[<bssid>]	Format : 6 bytes hex number e.g. 112233445566
Error number	1: command format error 2: command parameter error 3: wifi initial error 4: connect to AP failed 5: wifi mode error 6: get ap security type failed 7: dhcp timeout, use static ip 192.168.1.80	
Note	1. Execute ATPW first, must be STA or Concurrent mode. 2. If no password, remain the parameter <pwd> NULL e.g. ATPN="SSID" or ATPN="SSID",,,112233445566	

3.3 ATWD - Disconnect from AP

ATWD	
Description	This command is used to disconnect with AP for station
Response	[ATWD] OK [ATWD] ERROR:<error_code>
Error number	3: operation failed 4: disconnect timeout

3.4 ATWS - Scan AP

ATWS	
Description	This command is used to scan AP in the air
Response	AP : <num>,<ssid>,<chl>,<sec>,<rssi>,<bssid> [ATWS] OK [ATWS] ERROR:<error_no>
Note	The information of AP in order are number, SSID, channel, security mode, strength of signal, BSSID

3.5 ATPA - Set AP mode

ATPA=<ssid>,<pwd>,<chl>,<hidden>[,<max_conn>]		
Description	This command is used to config AP mode	
Response	[ATPA] OK [ATPA] ERROR:<error_no>	
Parameter	<ssid>	This parameter can't be empty Format: "ssid" Must add prefix '\' for special character(',' , '\' , '"', '[', ']')
	<pwd>	WPA/WPA2 : length is 8~64
	<chl>	Channel : 1~11
	<hidden>	0 : Not hidden SSID 1 : hidden SSID
	[<max_conn>]	Max number of STAs, should be [1,3], default is 3
Error number	1: command format error 2: command parameter error 3: wifi initial error 4: start AP failed 5: wifi mode error	
Note	1. Execute ATPW first, must be AP or Concurrent mode 2. If no password, remain the parameter NULL. e.g. ATPA="SSID",,11,0	

3.6 ATW? - Wifi information

ATW?	
Description	This command is used to list wifi information
Response	<mode>,<SSID>,<chl>,<sec>[,<key_id>],<pwd>,<mac>,<ip>,<gw> CLIENT : <num>,<mac> [ATW?] OK
Note	1. The information in order are wifi mode, SSID, channel, security mode, (key id for WEP), password, device mac, device IP, gateway. 2. In AP mode, show extra client information, number and the BSSID of client

3.7 ATPH - Set DHCP mode

ATPH=<mode>,<enable>		
Description	This command is used to set DHCP function for both mode	
Response	[ATPH] OK [ATPH] ERROR:<error_no>	
Parameter	<mode>	1 : AP mode 2 : STA mode
	<enable>	1 : DHCP 2 : Static IP
Error number	1: command format error 2: command parameter error	
Note	1. Default is DHCP for both mode 2. Use ATPE to set static IP for station 3. Use ATPF to set DHCP rule for AP	

3.8 ATPE - Set static IP for STA

ATPE=<ip>[,<gateway>,<mask>]		
Description	This command is used to set static IP for station	
Response	[ATPE] OK [ATPE] ERROR:<error_no>	
Parameter	<ip>	Static station IP, e.g. 192.168.1.2
	[<gateway>]	[optional] set gateway IP
	[<mask>]	[optional] set mask IP
Error number	1: command format error 2: command parameter error	
Note	1. Default static IP of station is 192.168.1.80 2. Effective in static IP mode for station. (ATPH=2,2)	
Example	# ATPE=192.168.1.150 //Set static IP for station to 192.168.1.150 # ATPH=2,2 //Make static IP effective # ATPN=iot_newifi,abcdef1234 //Connect to iot_newifi # ATW? //query wifi information *STA,iot_newifi,11,AES,abcdef1234,ec:f0:0e:4e:75:0b,192.168.99.150,192.168.99.1* *[ATW?] OK*	

3.9 ATPF - Set DHCP rule and gateway

ATPF=<start_ip>,<end_ip>,<gateway>		
Description	This command is used to set DHCP rule and gateway for AP	
Response	[ATPF] OK [ATPF] ERROR:<error_no>	
Parameter	<start_ip>	Set the start IP for client
	<end_ip>	Set the end IP for client
	<gateway>	set gateway IP
Error number	1: command format error 2: command parameter error	
Note	1. Default gateway IP is 192.168.43.1 2. For DHCP mode, config the DHCP rule of AP. (ATPH=1,1) 3. For static IP mode, config the IP of AP. (ATPH=1,2)	
Example	# ATPF=192.168.99.100,192.168.99.102,192.168.99.1 //Set static IP for AP to 192.168.99.1(also used as gateway) # ATPH=1,1 //Make DHCP server effective # ATPW=2 //Configure device to AP mode # ATPA=iot_test,abcdef1234,1,0 // Start Soft AP “iot_test” # ATW? //query wifi information *AP,iot_test,1,AES,abcdef1234,ec:f0:0e:4e:75:0b,192.168.99.1,192.168.99.1* *[ATW?] OK*	

3.10 ATPG - Set Auto connect

ATPG=<enable>		
Description	This command is used to set the auto connection when device booting	
Response	[ATPG] OK [ATPG] ERROR:<error_no>	
Parameter	<enable>	0 : disable auto connect 1 : enable auto connect
Error number	1: command format error 2: command parameter error	
Note	Default is disable	
Example	# ATPN=iot_newifi,abcdef1234 //connect to “iot_newifi”, device will store this information into flash # ATPG=1 //enable auto connect, this will be store in flash >>reboot device >>device will read connection information from flash and auto connect to “iot_newifi”	

3.11 ATPM - Set MAC address

ATPM=<mac>		
Description	This command is used to set the mac address of device	
Response	[ATPM] OK [ATPM] ERROR:<error_no>	
Parameter	<mac>	Format : 6 bytes hex number e.g. 112233445566
Error number	1: command format error 2: command parameter error	
Note	Must restart system for effecting new MAC	

3.12 ATWQ - Start simple config

ATWQ	
Description	This command is used to start simple config
Response	[ATWQ] OK [ATWQ] ERROR:<error_no>
Error number	1: cannot get station information 2: cannot parse the station info 3: cannot scan the target channel 4: fail to connect to target AP 5: fail to get IP address from target AP 6: fail to create UDP socket to send info to controller

4 TCP/IP command

4.1 Compile guide

To enable transport TCP/IP command, please configure **CONFIG_TRANSPORT** to 1 in platform_opts.h.

4.2 ATP0 – Get LWIP errno

AT	
Description	This command is used to get errno in LwIP
Response	[ATP0] OK:<errno> [ATP0] ERROR (errno isn't enabled in FW)

4.3 ATPS – Create TCP/UDP Server

ATPS =<mode>,<Local Port>		
Description	This command is used to create TCP/UDP Server.	
Response	[ATPS] OK [ATPS] con_id=x (x=[1,9], con_id 0 is reserved) Under TCP mode, if a client connects, there will be response as below: [ATPS] A client connected to server[<server_id>] con_id:<x>,seed,tcp,address:xxx.xxx.xxx.xxx,port:<x>,socket:<x> (response format refer to section 4.8 ATPI) [ATPS] ERROR:<error_no>	
Parameter	<Mode>	0 : TCP mode 1 : UDP mode
	<Local Port>	1~65535
Error Number	1: parameter number error 2: local port should be 1~65535 3: create con_id error 4: create server task error 5: create socket error 6: set socket option error 7: bind error 8: listen error 9: tcp server already exists error 10: accept error 11: create con_id for seed error 12: udp server already exists error 13: server can't start under TT(transparent transmission) mode	
Note	This command will assign a con_id to this TCP/UDP Server	

Example	//create a TCP server on PORT 5001
	# ATPS=0,5001
	[ATPS] OK
	[ATPS] con_id=1
	
	//when a client connects to TCP server[con_id=1]
	[ATPS] A client connected to server[1]
	con_id:2,seed,tcp,address:192.168.99.185,port:64068,socket:1
	
	//create a UDP server on PORT 5002
	# ATPS=1,5002
	[ATPS] OK
	[ATPS] con_id=3
	
	//query connection information
	# ATPI
	con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0
	con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1
	con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2
	[ATPI] OK

4.4 ATPC – Create TCP/UDP Client

ATPC =<mode>,< Remote Addr>,< Remote Port>[,<Local Port>]		
Description	This command is used to create TCP/UDP Client.	
Response	[ATPC] OK [ATPC] con_id=x (x=[1,9], con_id 0 is reserved) [ATPC] ERROR:<error_no>	
Parameter	<Mode>	0 : TCP mode 1 : UDP mode
	<Remote Addr>	xxx.xxx.xxx.xxx Or “www.xxx.com”
	< Remote Port>	1~65535
	[<Local Port>]	Local port to bind, only valid for UDP
Error Number	1: parameter number error 2: remote IP format or host unfound error 3: remote port should be 1~65535 error 4: create con_id error (none available) 5: create client task error 6: inet_ntoa_r remote address error 7: create socket error 8: hang node error for tcp client 9: connect error for tcp client 10: hang node error for udp client 11: local port should be 1~65535	

	12: bind local port error 13: connection already exists for TT(transparent transmission) mode
Note	This command will assign a con_id to this TCP/UDP Client

Example	//Create a TCP client and connect to TCP server IP 192.168.99.185 on server's port 5001
	# ATPC=0,192.168.99.101,5001
	[ATPC] OK
	[ATPC] con_id=4
	
	//Create a UDP client targeting to server "www.google.com" on server's port 8080
	# ATPC=1,"www.google.com",8080
	[ATPC] OK
	[ATPC] con_id=5
	
	//query connection information
	# ATPI
	con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0
	con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1
	con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2
	con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3
	con_id:5,client,udp,address:64.233.189.104,port:8080,socket:4
	[ATPI] OK

4.5 ATPD – Close TCP or UDP connection

ATPD=<con_id>		
Description	This command is used to close TCP/UDP connection	
Response	[ATPD] OK [ATPD] ERROR:<error_no>	
Parameter	< con_id >	con_id=[1,9] for certain connection con_id=0 to close all connections
Error Number	1: command format error 2: command parameter error 3: no con_id is found	
Note	Use the ATPI command to show the connection id	

Example	//query connection information
	# ATPI
	con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0
	con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1
	con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2
	con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3
	con_id:5,client,udp,address:64.233.189.104,port:8080,socket:4
	[ATPI] OK
	
	//close con_id 5 (udp client)
	# ATPD=5
	[ATPD] OK
	
	//query connection information
	# ATPI
	con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0
	con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1
	con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2
	con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3
	[ATPI] OK	

	//close con_id 1 (TCP server), and its seed(con_id=2) will be also closed
	# ATPD=1
	[ATPD] OK
	
	//query connection information
	# ATPI
	con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2
	con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3
	[ATPI] OK
	
	//close all connections
	# ATPD=0
	[ATPD] OK
	
	//query connection information
	# ATPI
	[ATPI] OK

4.6 ATPT – Send data

ATPT=<data_size>,<con_id>[,<dst_ip>,<dst_port>]:<data>		
Description	This command is used to send data to a specific connection	
Response	[ATPT] OK,<con_id> [ATPT] ERROR:<error_no>	
	<data_size>	Data length
	<con_id>	(1~9, con_id 0 is reserved)
	[<dst_ip>]	[optional]xxx.xxx.xxx.xxx (only need for udp server mode)
	[<dst_port>]	[optional]1~65535 (only need for udp server mode)
	<data>	Payload data
Error Number	1: parameter number error 2: <Buffer Size> exceeds ATPT send buffer size 3: con_id is not found 4: <UDP Client IP> or <UDP Client Port> error for udp server case 5: sendto() error for udp server 6: sendto() error for udp client 7: TCP server should send data to the seed 8: write error for tcp client/server	
Note	1. Use the ATPI command to show the connection status 2. The ATPT command can't send data via TCP server created at localhost. 3. After delimiter ":", any input will count	
Example	//query connection information # ATPI *con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0* *con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1* *con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2* *con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3* *con_id:5,client,udp,address:64.233.189.104,port:8080,socket:4* *[ATPI] OK*	

	//send data to TCP client(Seed) (con_id 2)
	# ATPT=14,2:Hello Realtek!
	[ATPT] OK,2
	
	//send data to UDP Server via UDP client(con_id 5)
	# ATPT=14,5:Hello Realtek!
	[ATPT] OK,5
	
	//send data to TCP Server via TCP client(con_id 4)
	# ATPT=14,4:Hello Realtek!
	[ATPT] OK,4
	
	//send data to UDP client(ip: 192.168.99.185, port:55339) via UDP Server(con_id 3)
	# ATPT=14,3,192.168.99.185,55339:Hello Realtek!
	[ATPT] OK,3

4.7 ATPR – Receive data

ATPR =<con_id>,<Buffer Size>		
Description	This command is used to receive data from a specific connection id, and FW can also be configured to auto receive mode which means any packet received on any connection will return to host automatically(refer to section 4.8 command ATPK)	
Response	[ATPR] OK,<data size>,<con_id>[,<dst_ip>,<dst_port>]:<data> [ATPR] ERROR:<error_no>	
Parameter	<con_id>	(1~9, con_id 0 is reserved)
	<Buffer Size>	Data length
Error Number	1: command format error 2: <Buffer Size> error (should be 1 ~ MAX_BUFFER(default 1600)) 3: <con_id> is not found 4: recvfrom() error for udp server 5: recvfrom() error for udp client/seed 6: TCP server should receive from seed 7: connection lost 8: read() error for tcp con_id	
Note	1.Use the ATPR command to receive data from the specific connection id 2. The ATPR command can't receive data via TCP server created at localhost. 3. [,<dst_ip>,<dst_port>] will append only if receive data via UDP server created at localhost	
Example	//query connection information # ATPI *con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0* *con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1* *con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2* *con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3* *con_id:5,client,udp,address:64.233.189.104,port:8080,socket:4* *[ATPI] OK*	

	//receive data “12345678” via TCP seed (con_id 2)
	# ATPR=2,1500
	[ATPR] OK,8,2:12345678
	
	//receive data “12345678” via UDP server(con_id 3)
	# ATPR=3,1500
	[ATPR] OK,8,3,192.168.99.185,52795:12345678
	
	//receive data “12345678” via TCP client(con_id 4)
	# ATPR=4,1500
	[ATPR] OK,8,4:12345678

4.8 ATPK – Set auto receive data mode

ATPK=<enable>		
Description	This command is used to set auto receive data mode	
Response	[ATPK] OK [ATPK] ERROR:<error_no>	
Parameter	<enable>	0 : disable auto receive data mode (default) 1 : enable auto receive data mode
Error Number	1: command parameter error 2: start auto receive task fail	
Note	Once the auto receive mode is enabled, any packet received on any connection will return to host automatically in the same format as ATPR (refer to section 4.7, response of command ATPR) in normal transmission mode. But if under transparent transmission mode, received data will return to host without any information in the head. Normal mode: *[ATPR] OK,8,3,192.168.99.185,52795:12345678* TT(transparent transmission) mode: *12345678*	

4.9 ATPI – Check network connection status

ATPI	
Description	This command is used to print network connection status
Response	con_id :<con_id >,<server/seed(TCP client)/client>,\ <tcp/udp>,address:<IP ADDRESS>,port:<PORT>,socket:<socket id> … [ATPI] OK
Error Number	
Example	# ATPI *con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0* *con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1* *con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2* *con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3* *con_id:5,client,udp,address:64.233.189.104,port:8080,socket:4* *[ATPI] OK*

4.10 ATPP – PING Command

ATPP=<xxxx.xxxx.xxxx.xxxx>,[y/loop] Or ATPP=<con_id>, [y/loop]		
Description	This command is used to PING a specific connection id, or PING a specific network address	
Response	[ATPP] OK [ATPP] ERROR:<error_no>	
Parameter case 1	<Remote IP>	xxx.xxx.xxx.xxx
	[y/loop]	No assign: Only five ping requests will be sent. Loop: loop, no count Count: loop with count
Parameter case 2	<con_id>	1~NUM_NS(default 10)
	[y/loop]	No assign: Only five ping requests will be sent. Loop: loop, no count Count: loop with count
Error Number	1: command format error 2: con_id is not found	
Note	Use the ATPR command to receive data from the specific connection id	
Example	//parameter case 1 # ATPP=192.168.1.1 // Only five ping requests will be sent # ATPP=192.168.1.1,loop // loop, no count # ATPP=192.168.1.1,10 // loop 10 times	

	//parameter case 2 # ATPI *con_id:1,server,tcp,address:192.168.99.143,port:5001,socket:0* *con_id:2,seed,tcp,address:192.168.99.185,port:64089,socket:1* *con_id:3,server,udp,address:192.168.99.143,port:5002,socket:2* *con_id:4,client,tcp,address:192.168.99.185,port:5001,socket:3* *con_id:5,client,udp,address:64.233.189.104,port:8080,socket:4* *[ATPI] OK* # ATPP=2 //Ping TCP client(con_id 4) # ATPP=5 //Ping UDP server via UDP client(con_id 2) # ATPP=4 //Ping TCP server via TCP client(con_id 3)

4.11 ATPU – Set transparent transmission mode

ATPU=<enable>		
Description	This command is used to set transparent transmission(TT) mode	
Response	[ATPU] OK [ATPU] ERROR:<error_no>	
Parameter	<enable>	1 : enable TT mode (only “1” is valid by now)
Error Number	1: command parameter error 2: no connection found when try to enter TT mode 3: cannot enter TT mode if it’s server connection 4: more than one connection when try to enter TT mode 5: start TT task failed	
Note	Once the TT mode is enabled, only one TCP/UDP client connection can be created.	

Example	//For TT(transparent transmission) mode
	# ATPD=0 //close all connectiosn
	[ATPD] OK
	# ATPC=0,192.168.99.101,5001 //create TCP client, single connection
	[ATPC] OK
	[ATPC] con_id=1
	# ATPU=1 //enter TT mode
	//20ms interval between sending packets
	//auto recv mode is also enabled
	[ATPU] OK
	> *//enter data transmission mode, any input is treated as data to send,*
	//besides the uart echo is turned off, which means any input character
	//won’t have uart echo
	Hello Realtek! //first packet
	(wait for 20ms)
	Hello Realtek! //second packet
	(wait for 20ms)
	---- //input four hyphens(“-”) to return to command mode	

	# //return to command mode now, auto recv is disabled, uart echo is turned on

4.12 ATPL – Save translink and enable autolink

ATPL=<enable>		
Description	This command is used to save connection information to flash and enable auto connect while booting up	
Response	[ATPL] OK [ATPL] ERROR:<error_no>	
Parameter	<enable>	0 : erase translink info in flash and disable autolink 1 : save translink and enable autolink
Error Number	1: command parameter error 2: parameter number error 3: no connection found	
Note	Device will auto establish connection by using the information stored in flash, and enter data transparent transmission mode.	
Example	# ATPD=0 //close all connectiosn *[ATPD] OK* # ATPN=iot_test,12345678 //connect to AP *[ATPN] OK* # ATPG=1 //enable auto connect, this will be store in flash *[ATPG] OK* # ATPC=0,192.168.99.101,5001 //create TCP client, single connection *[ATPC] OK* *[ATPC] con_id=1* # ATPL=1 //save information into flash *[ATPL] OK* # ATSR //reboot device *[ATSR] OK* *AT COMMAND READY* *> // start data transmission from here, 20ms between packets* ---- //input four hyphens("-") to return to command mode *# //return to command mode*	