WF5000
AT command
User Guide

I&C TECHNOLOGY

www.inctech.co.kr

Version 6.2

2019. 03. 14

I&C Technology Co., Ltd.,

I&C Building, Pangyo-ro255 24, Bundang-gu, Seongnam-si, Gyeonggi-do, 463-400, South KoreaI&C Technology Co., Ltd. reserves the right to make changes to the product(s) or specifications to improve performance, reliability, or manufacturability. Information furnished is believed to be accurate and reliable, but I&C Technology Co., Ltd. shall not be responsible for any errors that may appear in this document. I&C Technology Co., Ltd. makes no commitment to update or keep current the information contained in this document.

However, no responsibility is assumed for its use; or any infringement of patents or other rights of third parties, which may result from its use. No liability is assumed as a result of their use or application. No rights under any patent accompany the sale of any such product(s) or information.

I&C Technology Co., Ltd. products are not designed or intended for use in Life Support Systems. A Life Support System is a product or system intended to support or sustain life, which if it fails, can be reasonably expected to result in significant personal injury or death. If Buyer or any of its direct or indirect customers applies any product purchased or licensed from I&C Technology Co., Ltd. to any such unauthorized use, Buyer shall indemnify and hold I&C Technology Co., Ltd., its affiliates and their respective suppliers, harmless against all claims, costs, damages and expenses arising directly or indirectly, out of any such unintended or unauthorized use, even if such claims alleges that I&C Technology Co., Ltd. or any other person or entity was negligent in designing or manufacturing the product.

Specifications are subject to change without notice.

	Comment	Date	Author	Approver
1.00	Initial release	2012-08-17	tjjeong	
1.10	Modified RESET & APSTART command	2014-04-29	tjjeong	
1.20	Added Mode & Sleep command	2014-06-19	tjjeong	
1.30	Added TCP Connection Timeout Event	2014-06-25	tjjeong	
1.40	Added Network Status & Auto Connect command	2014-07-03	tjjeong	
1.50	Modified TCP multi-connections	2014-07-07	tjjeong	
1.60	Added Application Protocol Program for Standalone & ADHOC mode, EAP settings	2014-09-02	thkim	
1.61	Added Application Protocol Program for Standalone & HTTP client	2014-09-12	evergreen	
1.62	Added Web UI screenshots for Enterprise security	2014-10-07	evergreen	
1.63	Added Firmware Upgrade command	2014-10-29	tjjeong	
1.64	Added Firmware Upgrade using Web UI	2014-10-29	evergreen	
1.65	Added Binary UART mode, Update Auto Connection mode	2014-10-31	thkim	
1.66	Modified and Added Mode & Sleep command	2014-11-06	jryu	
1.67	Changed Sleep command to PS command	2014-11-10	jryu	
1.68	Added UPNP AT command	2014-11-12	evergreen	
1.69	Added MIB command, LPD command	2014-11-24	smkang	
2.0	Added DNS query command	2014-11-26	tjjeong	
2.1	Added APLEASEIP command	2014-12-04	jryu	
2.2	Added Factory Reset command and Update MIB command	2014-12-05	smkang	
2.3	Added Data Mode command	2014-12-05	tjjeong	
2.4	Added DDNS command	2014-12-08	thkim	
2.5	Modified Auconmode command description	2014-12-08	thkim	
2.6	Modified Firmware Upgrade command	2014-12-17	tjjeong	
2.7	Added SETMIB command	2014-12-19	smkang	
2.8	Added TCP Connection Reset Event	2014-12-24	tjjeong	
2.9	Added HTTP POST request command	2014-12-29	evergreen	
3.0	Added Remote IP Settings	2015-01-29	smkang	
3.01	Modified http close command	2015-02-28	evergreen	
3.1	Added TCP SSL	2015-03-13	smkang	
3.2	Modify AUCONMODE – To minimize AUCONMODE functions	2015-03-20	smkang	
3.3	Added Modified HTTPC/UPNP indication	2015-03-25	evergreen	
3.4	Added Web Server	2015-03-26	jhkim	
3.5	Added OTA command	2015-05-12	evergreen	
3.6	Modified UPNP_ADDPORTMAP command	2015-06-10	evergreen	
3.7	Modified OTA command	2015-06-23	evergreen	
3.8	Added Set Antenna command	2015-06-29	tjjeong	
3.9	Modify P2P commands and events	2015-07-02	smkang	
4.0	Added MIB contents	2015-09-23	smkang	
4.1	Added TCP SSL server	2015-10-08	smkang	
4.2	Modified OTA command (added FTP OTA)	2015-10-22	thkim	
4.3	Added SNTP client	2015-10-22	thkim	
4.4	Added UDAP contents	2015-11-02	smkang	
4.5	Modified HTTP POST command	2015-11-06	evergreen	
4.6	Added WDS Contents	2015-11-19	shlee	
4.7	Modified Binary Protocol section	2015-11-30	thkim	
4.8	Added FTPC SET/GET, SNTP SET command	2015-12-17	thkim	
4.9	Added DHCP SERVER START/STOP command	2015-12-30	thkim	
5.0	Modify SMODE mode option and Update TCP SSL server	2016-01-06	smkang	
5.1	Added 5Ghz Wireless Mode (A, AN , ABGN)	2016-03-15	yjkim	
5.2	Modified SNTP GET/SET command	2016-03-15	thkim	
5.3	Added TX Gain Reduce command	2016-03-18	shlee	
5.4	Added Remote Connection check command (NW_CONN)	2016-03-23	cylee	
5.5	Added INITSCAN event	2016-04-19	cylee	
5.6	Added HTTPHEADER command	2016-04-22	evergreen	
5.7	Added EAP LEAP, FAST contents	2016-05-31	cylee	
5.8	Modified Country Code section	2016-06-27	thkim	
5.9	Added TCP SSL Socket Descriptor	2016-07-19	cylee	
6.0	Added *ICT*STA_ASSOCIATED event message	2017-04-03	thkim	
6.1	Added SCONN delimeter parameter	2017-10-24	cylee	
6.2	Added DATA_INTERVAL command	2019-03-14	cylee	

Table of Contents

Table of Figures

1 Introduction

1.1 Purpose

The purpose of this document is to describe the architecture and operations of standalone Wi-Fi Modem used to manage existing serial applications via wireless LAN and to provide TCP/IP accessibility to them in an easy, quickly and cost-efficient way.

1.2 Scope

Writing information in this document is as follows.

1.3 Definition, acronyms, and abbreviations

1.4 References

Please refer to below the rule of detail technique

- Data Transmission Systems and Equipment - Serial Asynchronous Automatic Dialing and Control - Extended Command Syntax, TIA/EIA-615, 1993

1.5 Overview

The standalone Wi-Fi Modem that is existing Serial Application to used for data communications via wireless LAN is a serial to Wi-Fi device. It is also called as a Serial Wireless Adaptor or Serial Wi-Fi Router, which can be set by using a simple AT command set without the need of device drivers for a specific operating system or a host PC, and complex software programming, and Standalone Wi-Fi Modem can set Wi-Fi configuration or TCP/IP configuration of the Modem. The Standalone Wi-Fi Modem needs no additional operating system but minimum AT Commands to establish Wireless Internet connection, which reduces overheads of development time, certification and test time. So most software stacks such as RTOS, TCP/IP Protocol Stack, Wi-Fi Protocol Stack, and WPA supplicant are embedded in the Modem.
The existing serial communication applications which has distance limitation in data communications will be solved with Serial Wireless Gateway function, which can be applied to various applications as follows.

- Building/Factory Automation
- Smart Energy Management
- Industrial Sensor Network
- Medical Applications
- Remote Control and Monitoring Systems

- Security Systems
- Transportation

Such applications can be implemented on 8/16/32 bit Micro-controllers with firmware by the minimum AT commands. It allows any customer who has no experience in Wi-Fi Protocols or TCP/IP stacks can develop proper applications without difficulty.

2 AT Command Set

This chapter describes AT Command Set and their functions on the Standalone Wi-Fi Modem to control the Modem. In general, an AT command is made of an ASCII string, and there are Commands, Responses, and Event Information primitives between the Host System and a WF5000 Modem. Hereinafter, < > is an essential parameter, [ ] is an optional parameter. Every AT Command sent from the Host System to a Modem starts with “AT” string, and last string with <0x0D>, which means a carriage return (CR). A Response or Event Information of an AT Command sent from a Modem to the Host System ends with <0x0D><0x0A> of last string showing a Carriage Return (CR) and a Line Feed (LF). When entering an AT COMMAND on a terminal through UART, the COMMAND is not echoed on the screen by default. So if you want to check the entered command, use “ATE1” command to turn on echo. “ATE0” command is to turn off echo.
Every Command, Response, and Event Information has a String, “*ICT*”, attached after the string, “AT”, to distinguish it from other AT command sets used in other communications Modem. The separator between parameters, use a space (“ ”). That is, a command (Host System->Modem) starts with a text string, “AT*ICT*”, while a response or an event (Modem->Host System) starts with a text string, “*ICT*”.
To synchronize the data transmissions, the Host System shall send the next command after receiving a response to the command transmitted. For reference, an event received from the Modem is asynchronous data.

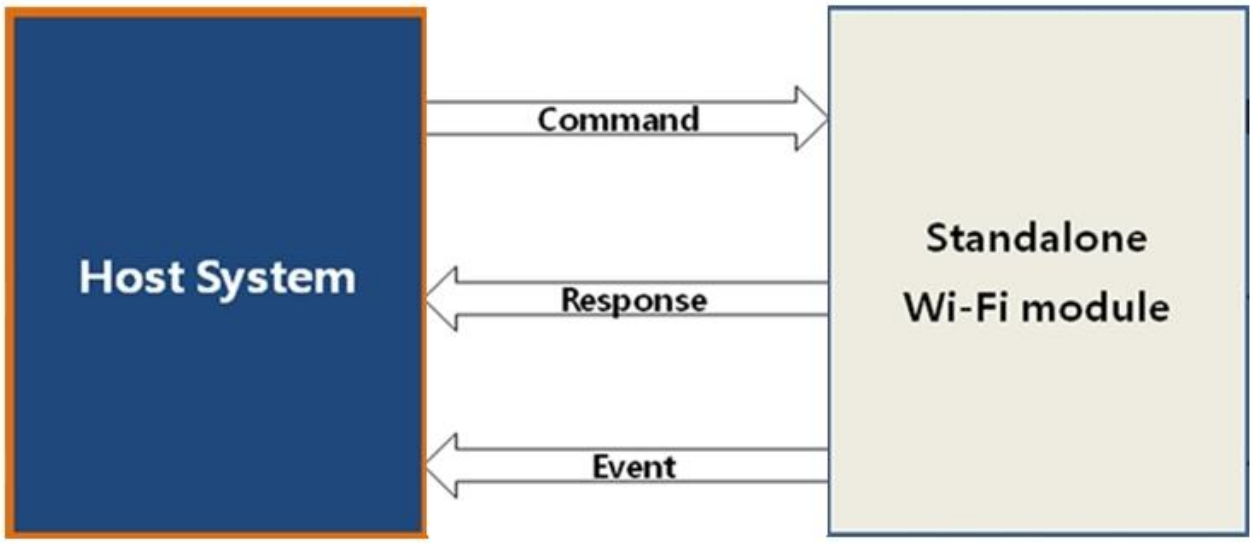


Figure 1. AT Command Primitive

The categories of the supported AT Command Set in the Standalone Wi-Fi Modem are as follows.

- Basic Command
- Wi-Fi Configuration
- TCP/IP Configuration & Stream Data
- Special Command
- Event Information

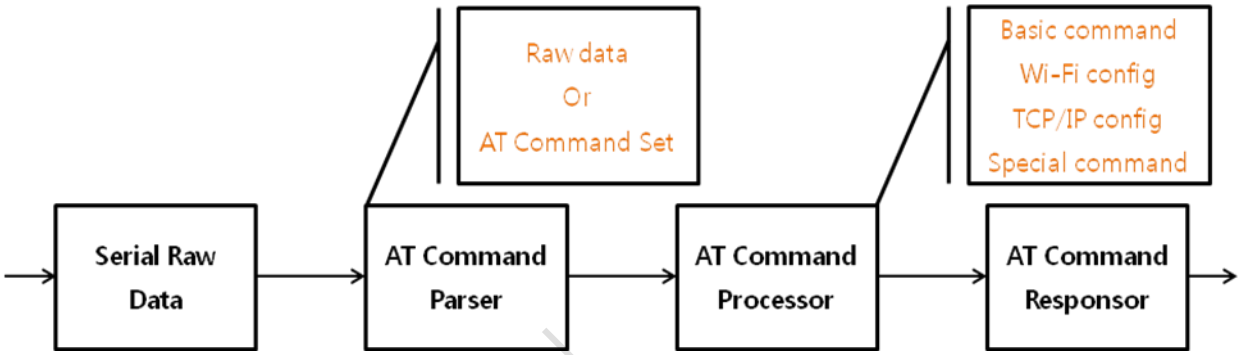


Figure 2. AT Command Implementation

There are some suggestions when sending TCP/IP stream data (stream_data field of AT*ICT*SEND command) and datagram data (datagram_data field of AT*ICT*SENDTO command) from the Host System. Default interface between the Host System and a Modem is AT command, the end of every AT command sent from the Host System to the Modem can be checked with a Carriage Return (0x0d). However, there may be control characters such as Carriage Return (0x0d) or Backspace (0x08) in actual data such as TCP/IP stream data and datagram data. Thus, it is necessary to process the data by escaping characters or byte stuffing as follows at the Host System to make them recognize as user data, not control characters by the Modem. In addition, The Modem are also able to un-escape characters.

Byte stuffing of actual data shall be used only on command (AT*ICT*SEND or AT*ICT*SENDTO) sent from the Host System to the Modem, and commands or events shall not be used sent from the Modem to the Host System, and the Host System needs to check the size parameter to confirm whether they are actual data.

Carriage Return : 0x0d, Back Space : 0x08, Escape Character : 0x1b, Escape Mask : 0x20

- Host System : **0x0d** => **0x1b 0x0d**^**0x20** => **0x1b** 0x2d
- Wi-Fi Modem : **0x1b** 0x2d => ~~**0x1b**~~ 0x2d^**0x20** => **0x0d**

- Host System : **0x08** => **0x1b 0x08**^**0x20** => **0x1b** 0x28
- Wi-Fi Modem : **0x1b** 0x28 => ~~**0x1b**~~ 0x28^**0x20** => **0x08**

- Host System : **0x1b** => **0x1b 0x1b**^**0x20** => **0x1b** 0x3b
- Wi-Fi Modem : **0x1b** 0x3b => ~~**0x1b**~~ 0x3b^**0x20** => **0x1b**

2.1 Basic Command

2.1.1 AT*ICT*SWVER

Description	Retrieve the firmware version
Usage	AT*ICT*SWVER=?
Parameters	?
Response	*ICT*SWVER:OK <SW Version> or *ICT*SWVER:ERROR SW Version: Major.Minor
Example	Host System -> Modem : AT*ICT*SWVER=?<0x0D> Modem -> Host System : *ICT*SWVER:OK **20.00**<0x0D><0x0A>

2.1.2 AT*ICT*MAC

Description	Retrieve the MAC address of the Modem device
Usage	AT*ICT*MAC=?
Parameters	?
Response	*ICT*MAC:OK <MAC address> or *ICT*MAC:ERROR MAC address: aa:bb:cc:dd:ee:ff
Example	Host System -> Modem : AT*ICT*MAC=?<0x0D> Modem -> Host System : *ICT*MAC:OK **84:72:07:01:02:03**<0x0D><0x0A>

2.1.3 AT*ICT*RESET

Description	Reset the Modem device
Usage	AT*ICT*RESET=<mode>
Parameters	0 - Internal IRAM Start 1 - External Serial Flash Start (Recommend)
Response	*ICT*RESET:OK or *ICT*RESET:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*RESET=1<0x0D>

	Modem -> Host System : *ICT*RESET:OK<0x0D><0x0A>

2.1.4 AT*ICT*FWUPGRADE

Description	Enter Firmware upgrade mode and Ready to receive a Firmware file with the XMODEM protocol
Usage	AT*ICT*FWUPGRADE=<bank> Refer to Chap 5. Firmware Upgrade
Parameters	Bank: 1 - Bank 1 block 2 - Bank 2 block
Response	*ICT*FWUPGRADE:OK or *ICT*FWUPGRADE:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*FWUPGRADE=2<0x0D> Modem -> Host System : *ICT*FWUPGRADE:OK<0x0D><0x0A>

2.1.5 AT*ICT*FACRESET

Description	Factory reset NV memory in the Modem device
Usage	AT*ICT*FACRESET=<mode>
Parameters	0 - NV Item Initialization 1 – NV Item Initialization and Operate as Soft AP mode after reboot
Response	*ICT*FACRESET:OK or *ICT*FACRESET:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*FACRESET=0<0x0D> Modem -> Host System : *ICT*FACRESET:OK<0x0D><0x0A> Host System -> Modem : AT*ICT*FACRESET=1<0x0D> Modem -> Host System : *ICT*FACRESET:OK<0x0D><0x0A>

2.1.6 AT*ICT*EVTDEL

Description	Delete the event message from the Modem to Host System
Usage	1) AT*ICT*EVTDEL=? or 2) AT*ICT*EVTDEL=<option>
Parameters	1) ?

	or 2) <option> 0 - event message on 1 - event message off
Response	*ICT*EVTDEL:OK or *ICT*EVTDEL:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*EVTDEL=0<0x0D> Modem -> Host System : *ICT*EVTDEL:OK<0x0D><0x0A>

2.2 Wi-Fi Configuration

2.2.1 AT*ICT*MODE

Description	Get or Set the wireless operating mode
Usage	1) AT*ICT*MODE=? or 2) AT*ICT*MODE=<mode>
Parameters	1) ? or 2) Wireless mode : 0 - G only 1 - B+G 2 - B+G+N (default) 3 – B only 4 – N only 5 – A only (5Ghz Support or Dual Mode) 6 – A+N (5Ghz Support or Dual Mode) 7 – A+B+G+N (5Ghz Support or Dual Mode)
Response	*ICT*MODE:OK [mode] or *ICT*MODE:ERROR [Error Code]
Example	1) Get the wireless mode Host System -> Modem : AT*ICT*MODE=?<0x0D> Modem -> Host System : *ICT*MODE:OK 2<0x0D><0x0A> 2) Set the wireless mode Host System -> Modem : AT*ICT*MODE=2<0x0D> Modem -> Host System : *ICT*MODE:OK<0x0D><0x0A>

2.2.2 AT*ICT*SMODE

Description	Set operating mode(SoftAP or Infrastructure) and activate mode
Usage	1) AT*ICT*SMODE=<mode> {SSID}
Parameters	mode : 0 – No SSID, Default Soft AP (Predefined SSID, channel, security is applied) ✓ Set default AP mode and active default AP 1 –Set SSID, Default Soft AP (User defined SSID, Predefined channel, security is applied) ✓ Set AP mode and active AP 99 –No SSID, Default Infrastructure Predefined SSID, security is applied)

	✓ Set Infrastructure mode and connect automatically predefined AP
Response	*ICT*MODE:OK [mode] or *ICT*MODE:ERROR [Error Code]
Example	Set the operating mode Host System -> Modem : AT*ICT*SMODE=0<0x0D> Modem -> Host System : *ICT*MODE:OK<0x0D><0x0A> Host System -> Modem : AT*ICT*SMODE=1 TEST_AP<0x0D> Modem -> Host System : *ICT*MODE:OK<0x0D><0x0A> Host System -> Modem : AT*ICT*SMODE=99<0x0D> Modem -> Host System : *ICT*MODE:OK<0x0D><0x0A>

2.2.3 AT*ICT*AUCONMODE

Description	Get or Set the auto connection mode
Usage	1) AT*ICT*AUCONMODE=? or 2) AT*ICT*AUCONMODE=0 <0~1> ~~or~~ ~~3) AT*ICT*AUCONMODE=1 <0~3>~~ ~~or~~ ~~4) AT*ICT*AUCONMODE=2~~

Parameters

Parameter	Value	Means
? (get)		get auto connection mode/type settings
0 (mode)	0	Auto connection deactivate (default)
	1	Auto connection activate and save currently settings to NV memory
	~~2~~	~~SoftAP mode activate (ssid: inc_ap, channel: 1, OPEN)~~
~~1 (type)~~	~~0~~	~~Not save currently settings to NV memory~~
	~~1~~	~~1) Save currently settings to NV memory when Successfully connected to an AP 2) If the connection fails, then retry connection to an AP by the latest connection information~~
	~~2~~	~~1) Save currently settings to NV memory when Successfully connected to an AP 2) If the connection fails, then retry connection to an AP by existing successful connection information in NV memory~~
	~~3~~	~~Save currently settings to NV memory~~

	2 (save) | | save auto connection mode/type settings
Response	*ICT* AUCONMODE:OK [mode] [type] or *ICT* AUCONMODE:ERROR [Error Code]
Example	1) Get the auto connection mode/type Host System -> Modem : AT*ICT*AUCONMODE=?<0x0D> Modem -> Host System : *ICT*AUCONMODE:OK **1 1**<0x0D><0x0A> 2) Set the auto connection activate Host System -> Modem : AT*ICT*AUCONMODE=**0** 1<0x0D> Modem -> Host System : *ICT*AUCONMODE:OK<0x0D><0x0A> 3) Set the auto connection deactivate Host System -> Modem : AT*ICT*AUCONMODE=**0** 0<0x0D> Modem -> Host System : *ICT*AUCONMODE:OK<0x0D><0x0A> 4) Set the auto connection type 1 Host System -> Modem : AT*ICT*AUCONMODE=1 1<0x0D> Modem -> Host System : *ICT*AUCONMODE:OK<0x0D><0x0A> 5) Set the auto connection type 2 Host System -> Modem : AT*ICT*AUCONMODE=1 2<0x0D> Modem -> Host System : *ICT*AUCONMODE:OK<0x0D><0x0A> 6) save the auto connection type Host System -> Modem : AT*ICT*AUCONMODE=2<0x0D> Modem -> Host System : *ICT*AUCONMODE:OK<0x0D><0x0A>

2.2.4 AT*ICT*UARTPROTO(Optional)

Description	Change UART protocol mode from AT command mode to Binary mode
Usage	AT*ICT*UARTPROTO
Parameters	None
Response	*ICT* UARTPROTO:OK
Example	Host System -> Modem : AT*ICT*UARTPROTO<0x0D> Modem -> Host System : *ICT*UARTPROTO:OK<0x0D><0x0A>

2.2.5 AT*ICT*HWPS (Optional)

Description	Get or Set the power save mode The Default is auto mode. If there is no traffic during 5 seconds, the Modem enters power save mode.
Usage	1) AT*ICT*HWPS=? or 2) AT*ICT*HWPS=<mode> Refer to Chap7.2 Hardware Power Save on Standalone mode with UART Interface
Parameters	1) ? or 2) Power save mode : 0 - Auto (default), Automatically power save is on /off by traffic conditions 1 – Always power save is off 2 – Always power save is on
Response	*ICT*HWPS:OK or *ICT* HWPS:ERROR [Error Code]
Example	1) Get the power save mode Host System -> Modem : AT*ICT*HWPS=?<0x0D> Modem -> Host System : *ICT*HWPS:OK **0**<0x0D><0x0A> 2) Set the power save mode Host System -> Modem : AT*ICT*HWPS=0<0x0D> Modem -> Host System : *ICT*HWPS:OK<0x0D><0x0A>

2.2.6 AT*ICT*SCAN

Description	Scan in all the channels
Usage	AT*ICT*SCAN
Parameters	None
Response	*ICT*SCAN:OK or *ICT*SCAN:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*SCAN<0x0D> Modem -> Host System : *ICT*SCAN:OK<0x0D><0x0A>

2.2.7 AT*ICT*WEP

Description	Configure the WEP key
Usage	AT*ICT*WEP=<key_index> <key>
Parameters	key_index: 1~4 key: 10 or 26 hexadecimal digits corresponding to a 40-bit or 104-bit key or 5 or 10 Ascii characters corresponding to a 40-bit or 104-bit key Don't use space between key value when input wep key.
Response	*ICT*WEP:OK or *ICT*WEP:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*WEP=**1** **1234567890**<0x0D> Modem -> Host System : *ICT*WEP:OK<0x0D><0x0A>

2.2.8 AT*ICT*PSK

Description	Configure the PSK(Pre Shared Key) that is used for creating Wi-Fi Protected Access
Usage	AT*ICT*PSK=<passphrase>
Parameters	Passphrase: ASCII passphrase must be between 8 and 64 characters If there is space between the passphrase parameters, must use a quotation (“ ”).
Response	*ICT*PSK:OK or *ICT*PSK:ERROR [Error Code]
Example	Normal case Host System -> Modem : AT*ICT*PSK=12345678<0x0D> Modem -> Host System : *ICT*PSK:OK<0x0D><0x0A> or Space between passphrase parameters Host System -> Modem : AT*ICT*PSK=**“**1234 5678**”**<0x0D> Modem -> Host System : *ICT*PSK:OK<0x0D><0x0A>

2.2.9 AT*ICT*CRYPTO

Description	Configure the encryption type and the AKM(Authentication Key Management)
Usage	AT*ICT*CRYPTO=<key_mgmt> <pairwise_cipher> <group_cipher>
Parameters	key_mgmt: 0 - OPEN 1 - WEP 2 - WPA_PSK 3 - WPA2_PSK pairwise cipher:

	0 - TKIP 1 - CCMP group cipher: 0 - TKIP 1 - CCMP 2 - WEP
Response	*ICT*CRYPTO:OK or *ICT*CRYPTO:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*CRYPTO=**1 0 0**<0x0D> Modem -> Host System : *ICT*CRYPTO:OK<0x0D><0x0A>

2.2.10 AT*ICT*ASSOCIATE

Description	Connect to an AP specified with ssid parameter
Usage	AT*ICT*ASSOCIATE=<ssid> [channel] Refer to Chap3. Example Sequence for AT Commands
Parameters	ssid: 32 characters maximum If there is space between the SSID string parameters, must use a quotation (“ ”). channel: 1~14
Response	*ICT*ASSOCIATE:OK or *ICT*ASSOCIATE:ERROR [Error Code]
Example	1) Normal case Host System -> Modem : AT*ICT*ASSOCIATE=iptime_lab<0x0D> Modem -> Host System : *ICT*ASSOCIATE:OK<0x0D><0x0A> 1) Space between SSID case Host System -> Modem : AT*ICT*ASSOCIATE=“ inctech ap”<0x0D> Modem -> Host System : *ICT*ASSOCIATE:OK<0x0D><0x0A>

2.2.11 AT*ICT*DISASSOCIATE

Description	Disconnect from an AP
Usage	AT*ICT*DISASSOCIATE
Parameters	None
Response	*ICT*DISASSOCIATE:OK or

	*ICT*DISASSOCIATE:ERROR
Example	Host System -> Modem : AT*ICT*DISASSOCIATE<0x0D> Modem -> Host System : *ICT*DISASSOCIATE:OK<0x0D><0x0A>

2.2.12 AT*ICT*APSTART

Description	Set the configuration information for SoftAP mode and Operate as SoftAP mode
Usage	AT*ICT*APSTART=<essid> <channel> [key_mgmt] [pairwise_cipher] [group_cipher] [passphrase] Refer to Chap 3. Example Sequence for AT Commands
Parameters	Essid: 32 characters maximum If there is space between the SSID string parameters, must use a quotation (“ ”). Channel: 1~14 Key_mgmt: 0 - OPEN **1 - WEP (Not Supported)** 2 - WPA_PSK 3 - WPA2_PSK Pairwise_cipher: 0 - TKIP 1 - CCMP Group_cipher: 0 - TKIP 1 - CCMP Passphrase: ASCII passphrase must be between 8 and 64 characters. If there is space between the passphrase parameters, must use a quotation (“ ”).
Response	*ICT*APSTART:OK or *ICT*APSTART:ERROR [Error Code]
Example	1) Open Host System -> Modem : AT*ICT*APSTART=**testap 6**<0x0D> Modem -> Host System : *ICT*APSTART:OK<0x0D><0x0A> 2) WPA2PSK with CCMP Host System -> Modem : AT*ICT*APSTART=**testap 11 3 1 1 12345678**<0x0D> Modem -> Host System : *ICT*APSTART:OK<0x0D><0x0A>

	3) Space between passphrase parameters Host System -> Modem : AT*ICT*APSTART=**testap 11 3 1 1 "1234 5678"**<0x0D> Modem -> Host System : *ICT*APSTART:OK<0x0D><0x0A> or Space between SSID and passphrase parameters Host System -> Modem : AT*ICT*APSTART=**"test standalone" 11 3 1 1 "1234 5678"**<0x0D> Modem -> Host System : *ICT*APSTART:OK<0x0D><0x0A>

2.2.13 AT*ICT*APSTOP

Description	Terminate SoftAP operation
Usage	AT*ICT*APSTOP
Parameters	None
Response	*ICT*APSTOP:OK or *ICT*APSTOP:ERROR
Example	Host System -> Modem : AT*ICT*APSTOP<0x0D> Modem -> Host System : *ICT*APSTOP:OK<0x0D><0x0A>

2.2.14 AT*ICT*ADSTART (Optional)

Description	Set the configuration information for AdHoc mode and Operate as AdHoc mode
Usage	AT*ICT*ADSTART=<essid> [channel] [key_mgmt] [passphrase] Refer to Chap 3. Example Sequence for AT Commands
Parameters	Essid: 32 characters maximum Channel: 0~14 (Use 0 channel for connect to another IBSS STA and use 1~14 channel for establishing IBSS Both parameters used in WEP encryption mode) Key_mgmt: 0 - OPEN 1 - WEP 2 - **WPA_PSK(Not Supported)** 3 - **WPA2_PSK(Not Supported)** Passphrase: 10 or 26 hexadecimal digits corresponding to a 40-bit or 104-bit key or 5 or 10 Ascii characters corresponding to a 40-bit or 104-bit key.

Response	*ICT*ADSTART:OK or *ICT*ADSTART:ERROR [Error Code]
Example	1) Open Host System -> Modem : AT*ICT*ADSTART=**adhoc_test**<0x0D> or AT*ICT*ADSTART=**adhoc_test 11**<0x0D> Modem -> Host System : *ICT*ADSTART:OK<0x0D><0x0A> 2) WEP (Establish IBSS or Connect to any other IBSS) When connecting to any other IBSS case does not matter if you enter the channel number. Host System -> Modem : AT*ICT*ADSTART=**adhoc_test 11 1 1234567890**<0x0D> Modem -> Host System : *ICT*ADSTART:OK<0x0D><0x0A>

2.2.15 AT*ICT*ADSTOP (Optional)

Description	Terminate AdHoc operation or Disassociate AdHoc connection
Usage	AT*ICT*ADSTOP
Parameters	None
Response	*ICT*ADSTOP:OK or *ICT*ADSTOP:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*ADSTOP<0x0D> Modem -> Host System : *ICT*ADSTOP:OK<0x0D><0x0A>

2.2.16 AT*ICT*SCONN

Description	Connect to an AP simply without setting security
Usage	AT*ICT*SCONN=<essid> {BSSID} {wep key index} [passphrase] Refer to Chap 3. Example Sequence for AT Commands
Parameters	[essid]: 32 characters maximum If there is a space between the SSID string parameters, must use a double quotation (“ ”). “: ASCII 0x22 If there is a double quotation between the SSID string parameters, must use a slash (/). /: ASCII 0x2F [BSSID] AP’s MAC Address, It is used that you want to connect a specified AP [wep key index] wep key index value’s range is 1 to 4

	It is only used when you setup wep. If the value is not set or omitted, the default value is 1 [Passphrase] WEP - 10 or 26 hexadecimal digits corresponding to a 40-bit or 104-bit key or 5 or 10 Ascii characters corresponding to a 40-bit or 104-bit key WPA or WPA2 - ASCII passphrase must be between 8 and 64 characters. If there is a space between the passphrase parameters, must use a double quotation (“ ”). “: ASCII 0x22 If there is a double quotation between the passphrase parameters, must use a slash (/). /: ASCII 0x2F
Response	*ICT* SCONN:OK or *ICT* SCONN:ERROR [Error Code]
Example	1) Open Host System -> Modem : AT*ICT*SCONN=Standalone<0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 2) WEP Host System -> Modem : AT*ICT*SCONN=Standalone 1234567890<0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 2) WEP (using wep key index) Host System -> Modem : AT*ICT*SCONN=Standalone *2* 1234567890<0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 3) WPA/WPA2 Host System -> Modem : AT*ICT*SCONN=Standalone 12345678<0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 3) WPA/WPA2 (space between passphrase case) Host System -> Modem : AT*ICT*SCONN=Standalone “1234 5678”<0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 4) WPA/WPA2 (double quotation between SSID case) Host System -> Modem : AT*ICT*SCONN=”Stand /”alone/”” “1234 5678” <0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 5) Specified BSSID Host System -> Modem : AT*ICT*SCONN=Standalone 00:01:02:03:04:05:06<0x0D> Host System -> Modem : AT*ICT*SCONN=Standalone 00:01:02:03:04:05:06 12345678<0x0D>

2.2.17 AT*ICT*EAPSET (Optional)

Description	Used to configure the EAP parameters for connecting to an Enterprise security AP
Usage	AT*ICT*EAPSET=<EAP SET NUM> <EAP Parameters>
Parameters	EAP SET NUM: EAP SET NUM must be set between 0 and 4 value. 0: Outer method – EAP authentication method. Valid values are tls, ttls, peap, leap, fast. 1: User identity – This is user identity used to generate the TLS, PEAP, TTLS, LEAP, FAST certificate. This is present in the user configuration file in the radius sever. 2: Anonymous user identity – Anonymous user identity can be used in TTLS, FAST. 3: User password – This is Private key password used to generate the TLS certificate. This should be same as the password in the user configuration file in the Radius Server for that User Identity. 4: EAP password – This is EAP password used to generate TTLS, PEAP, LEAP, FAST certificate.
Response	*ICT*EAPSET:OK or *ICT*EAPSET:ERROR [Error Code]
Example	1) TLS Host System -> Modem : AT*ICT*EAPSET=**0 tls 1 "wifi-user" 3 "12345678"** <0x0D> Modem -> Host System : *ICT*EAPSET:OK<0x0D><0x0A> 2) PEAP Host System -> Modem : AT*ICT*EAPSET=**0 peap 1 "wifi-user" 4 "wifi#11"** <0x0D> Modem -> Host System : *ICT*EAPSET:OK<0x0D><0x0A> 3) TTLS(when use user id) Host System -> Modem : AT*ICT*EAPSET=**0 ttls 1 "wifi-user" 4 "wifi#11"** <0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 4) TTLS(when use anonymous user id) Host System -> Modem : AT*ICT*EAPSET=**0 ttls 2 "anonymous id" 4 "wifi#11"** <0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A>

SET NUM	Parameters	Description
0	tls, peap, ttls, leap, fast	Setting eap outer method type
1	Maximum of 64bytes ASCII	Setting user identity
2	Maximum of 64bytes ASCII	Setting anonymous user identity
3	Maximum of 128bytes ASCII	Setting user password
4	Maximum of 128bytes ASCII	Setting EAP key password

	5) LEAP Host System -> Modem : AT*ICT*EAPSET=**0 leap 1 "wifi-user" 4 "wifi#11"** <0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A> 5) FAST Host System -> Modem : AT*ICT*EAPSET=**0 fast 1 "wifi-user" 4 "wifi#11" 2 "anonymous id"**<0x0D> Modem -> Host System : *ICT*SCONN:OK<0x0D><0x0A>

2.2.18 AT*ICT*EAPCERT (Optional)

Description	Used to configure the EAP Certification parameters for connecting to an Enterprise security AP
Usage	AT*ICT*EAPCERT=<CERT TYPE> <EAP CERT Parameters>
Parameters	EAP CERT SET NUM: EAP CERT SET NUM must be set between 0 and 4 value. 0: Inner method (phase 2) – Inner method used in TTLS, PEAP. The value can be set to MSCHAPv2 in all cases, including TLS, where it will not be used. 1: Peap label (phase1) – This is phase1 value used to generate PEAP certificate. The value can be set to peaplabel=0 in PEAP 2: CA certificate file– This is Certificate Authority (or Root Certificate) pem file used to certify the TLS, PEAP, TTLS. 3: Client certificate file– This is Client certificate pem file used to cetify the TLS certification. 4: Private key file – Private key pem file for TLS certification.
Response	*ICT*EAPSET:OK or *ICT*EAPSET:ERROR [Error Code]
Example	1) TLS Host System -> Modem : AT*ICT*EAPCERT=**2 cacert.pem 3 clcert.pem 4 cert_key.pem** <0x0D> Modem -> Host System : *ICT*EAPCERT:OK<0x0D><0x0A>

SET NUM	Parameters	Description
0	MSCHAPv2	Setting EAP inner method type
1	0	Setting phase1 value in PEAP
2	Maximum of 16bytes ASCII	CA Certificate File name
3	Maximum of 16bytes ASCII	TLS Client Certificate File name
4	Maximum of 16bytes ASCII	TLS Private key File name
5	Maximum of 16bytes ASCII	FAST PAC File name

	2) PEAP Host System -> Modem : AT*ICT*EAPCERT=**0** **MSCHAPv2** **1** **0** **2** **cacert.pem** <0x0D> Modem -> Host System : *ICT*EAPCERT:OK<0x0D><0x0A> 3) TTLS Host System -> Modem : AT*ICT*EAPCERT=**0** **MSCHAPv2** **2** **cacert.pem** <0x0D> Modem -> Host System : *ICT*EAPCERT:OK<0x0D><0x0A> 3) FAST Host System -> Modem : AT*ICT*EAPCERT=**5** **pac.key** <0x0D> Modem -> Host System : *ICT*EAPCERT:OK<0x0D><0x0A>

2.2.19 AT*ICT*ANTVER

Description	Get current WiFi Antenna type
Usage	AT*ICT*ANTVER=?
Parameters	?: Return value means 0: Chip Antenna 1: u.FL Antenna
Response	*ICT*ANTVER:OK [type] or *ICT*ANTVER:ERROR [Error Code]
Example	Host System ->Modem : AT*ICT*ANTVER=? <0x0D> Modem -> Host System : *ICT*ANTVER:OK 0<0x0D><0x0A>

2.2.20 AT*ICT*SETANT

Description	Set WiFi Antenna type
Usage	AT*ICT*SETANT=<type>
Parameters	0: Chip Antenna 1: u.FL Antenna
Response	*ICT*SETANT:OK or *ICT*SETANT:ERROR [Error Code]
Example	Host System ->Modem : AT*ICT*SETANT=1 <0x0D> Modem -> Host System : *ICT*SETANT:OK<0x0D><0x0A>

2.2.21 AT*ICT*MIB

Description	Get current MIB
Usage	AT*ICT*MIB=[mib index]
Parameters	MIB index

Index num	Name	Return Values
0	SSID	SSID value (string)
1	Wi-Fi Channel	Channel value (1 ~ 252)
2	Wi-Fi Network Mode	0 : Infrastructure 1 : Adhoc 2: SoftAP 3:P2P
3	Encryption Protocol Type	0 : OPEN 1: WEP 2: WPA 3: WPA2 4: WAPI 5: WPA Enterprise 6: WPA2 Enterprise
4	Pairwise Cipher Suite	0 : TKIP 1 : CCMP 2 : WEP 3: Reserved 4 : NONE 5 : IGTK 6 : WAPI 7 : BIP
5	Group Cipher Suite	0 : TKIP 1 : CCMP 2 : WEP 3: Reserved 4 : NONE 5 : IGTK 6 : WAPI 7 : BIP
6	WEP key	WEP key value (string)
7	WPA Passphrase	Passphrase (string)
8	WEP key index	WEP key index (1 ~ 4)
9	EAP type	0 : None 1 : TTLS 2 : TLS

			3 : PEAP 4 : LEAP 5 : FAST
	10	EAP identity	EAP identity (string)
	11	EAP password	EAP password (string)
	12	Server Certification Name	File name (string)
	13	Client Certification Name	File name (string)
	14	Key Certification Name	File name (string)
	15	Serial Number	Device Serial Number (string)
	16	Device Code	Device Code (0 ~ 65535)
	17	Device Type	Device Type (0 ~ 65535)
	18	Device Name	Device Name (string) – 16 bytes
	19	Model Name	Model Name
	20	Manufacture	Manufacture
	21~ 31	Reserved	
	32	Version	Firmware Version (string)
	33	MAC Address	MAC Address (ex:12:34:56:78:90)
	34	BSSID	BSSID (ex:00:12:34:56:78:90)
	35	Wi-Fi Frequency	Frequency value (2412 ~ 5825)
	36	Baud Rate	Baud rate
	37	Wi-Fi Connection Status	0 : Ready 1 : Connected 2 : IP Allocated
	38 ~ 63	Reserved	
	64	RTS threshold	RTS threshold value (1 ~ 2347)
	65	CTS threshold	CTS threshold value (1 ~ 2347)
	66	Fragmentation threshold	Fragmentation threshold value (1 ~ 65535)
	67	Beacon Interval	Beacon Interval (50~ 500)
	68 ~ 95	Reserved	
	96	IP Address	IP Address value (ex:192.168.0.10)
	97	Subnet Mask	Subnet Mask value (ex:255.255.255.0)
	98	Gateway Address	Gateway Address value (ex:192.168.0.255)
	99	DNS Address	DSN Address value (ex:164.124.101.2)

	100	IP Type	0 : Static IP 1 : DHCP
	101	DHCP Server lease IP	10:14
	102	Service Port	Service Port List (ex:23:80:63)
	103	UDAP Port	Reserved (49152 ~ 65535) (ex: 49152: 49152)
	104	Remote Network Status (Data mode only)	0 : Not connected 1: Connected
	105	Remote IP (Data mode only)	Remote IP (192.168.0.1)
	106	Remote Port (Data mode only)	Remote Port (9000)
	107 ~ 111	Reserved	
	112	RSSI	RSSI value (-100 ~ 0)
Response	*ICT*MIB:OK [return value] or *ICT*MIB:OK null → There is no assigned value. or *ICT*MIB:ERROR [Error Code]		
Example	Host System ->Modem : AT*ICT*MIB=0 <0x0D> Modem -> Host System : *ICT*MIB:OK SSID_sample<0x0D><0x0A> Host System ->Modem : AT*ICT*MIB=1 <0x0D> Modem -> Host System : *ICT*MIB:OK 11<0x0D><0x0A>		

2.2.22 AT*ICT*SETMIB

Description	Set MIB values
Usage	AT*ICT*SETMIB=[mib index] {Parameters}
Parameters	MIB index

Index num	Name	Parameters
0 ~14	Reserved	
15	Serial Number Max size of serial number is 32 bytes	at*ict*setmib=15 [serial number] ex) at*ict*setmib=15 12345678901234567890
16	Device Code (0 ~ 65535)	at*ict*setmib=16 [device code] ex) at*ict*setmib=16 10

	17	Device Type (0 ~ 65535)	at*ict*setmib=17 [device type] ex) at*ict*setmib=17 10
	18	Device Name Max size of device name is 16 bytes	at*ict*setmib=18 [device name] ex) at*ict*setmib=18 printer
	19	Model Name Max size of device name is 32 bytes	at*ict*setmib=19 [model name] ex) at*ict*setmib=19 fw5000
	20	Manufacture Max size of device name is 16 bytes	at*ict*setmib=20 [manufacture] ex) at*ict*setmib=20 inctechnology
	21 ~ 35	Reserved	
	36	Baud Rate To set baud rate to be used in the AT command. This command applies need to be rebooted	at*ict*setmib=36 [baudrate] [baudrate] 110 ~ 9216000 ex) at*ict*setmib = 36 115200 – Set baudrate at*ict*reset=0 – Set Reboot
	37 ~ 101	Reserved	
	102	Service Port This service Port is maintained only one. If you set the other Service Port, the existing setting will be lost	at*ict*setmib=102 [socke_type] [port] [socket_type] 0: NONE (Disable Socket) 4: TCP Server 8: UDP Server [port] 1024~ 65535 ex) [Set TCP server] at*ict*setmib = 102 4 9001 [Set UDP server] at*ict*setmib = 102 8 49152 [Set disable Server]

			at*ict*setmib = 102 0 0
	105	Remote IP (Data mode only)	at*ict*setmib=105 [remote ip] ex) at*ict*setmib=105 192.168.0.1
	106	Remote Port (Data mode only)	at*ict*setmib=106 [remote port] ex) at*ict*setmib=106 9000
	107 ~ 254	Reserved	
Response	*ICT*MIB:OK or *ICT*MIB:ERROR [Error Code]		
Example	Host System ->Modem : AT*ICT*SETMIB=102 4 9001 <0x0D> Modem -> Host System : *ICT*MIB:OK <0x0D><0x0A> Host System ->Modem : AT*ICT*SETMIB=102 4 0 <0x0D> Modem -> Host System : *ICT*MIB:OK null<0x0D><0x0A>		

2.2.23 AT*ICT*COUNTRY

Description	Get / Set Country Code – Country Codes set allowed Wi-Fi channels for All countries
Usage	AT*ICT*COUNTRY=[? / Country Code]
Parameters	?: Return Country Code [Country Code]

Country Code	Country Name	Wi-Fi Channel
AT	Austria	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
AU	Australia	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 165
BE	Belgium	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140

	BR	Brazil	2.4 GHz : 1 ~ 11 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 165
	CA	Canada	2.4 GHz : 1 ~ 11 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 165
	CH	Switzerland and Liechtenstein	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	CN	China	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 149, 153, 157, 161, 165
	CY	Cyprus	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	CZ	Czech Republic	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	DE	Germany	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	DK	Denmark	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	EE	Estonia	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	ES	Spain	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128,

		132, 136, 140
FI	Finland	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
FR	France	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
GB	United Kingdom	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 140
GR	Greece	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
HK	Hong Kong	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 165
HU	Hungary	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
ID	Indonesia	2.4 GHz : 1 ~ 13 5 GHz : 149, 153, 157, 161, 165
IE	Ireland	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
IL	Israel	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64
~~IO~~	~~Israel OUTDOOR~~	~~2.4 GHz : 5 ~ 13~~ ~~5 GHz : 36, 40, 44, 48, 52, 56, 60, 64~~
IN	India	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140

	IS	Iceland	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	IT	Italy	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	JP	Japan	2.4 GHz : 1 ~ 14 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	KR	Republic of Korea	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 149, 153, 157, 161, 165
	LT	Lithuania	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	LU	Luxembourg	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	LV	Latvia	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	RU	RUSSIAN FEDERATION	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 132, 136, 140, 149, 153, 157, 161
	MY	Malaysia	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	NL	Netherlands	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
	NO	Norway	2.4 GHz : 1 ~ 13

		5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
NZ	New Zealand	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 165
PH	Philippines	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
PL	Poland	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
PT	Portugal	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
SE	Sweden	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
SG	Singapore	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 132, 136, 140, 149, 153, 157, 161
SI	Slovenia	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
SK	Slovak Republic	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
TH	Thailand	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
TW	Taiwan	2.4 GHz : 1 ~ 13

			5 GHz : 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161
	US	United states of America	2.4 GHz : 1 ~ 11 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 149, 153, 157, 161, 165
	~~USE~~ UE	~~United states of America~~	~~2.4 GHz : 1 ~ 11~~ ~~5 GHz : 36, 40, 44, 48, 52, 56, 60, 64~~
	~~USL~~ UL	~~United states of America Low~~	~~2.4 GHz : 1 ~ 11~~ ~~5 GHz : 36, 40, 44, 48, 52, 56, 60, 64~~
	ZA	South Africa	2.4 GHz : 1 ~ 13 5 GHz : 36, 40, 44, 48, 52, 56, 60, 64, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140
Response	*ICT*COUNTRY:OK [Country Code] or *ICT*COUNTRY:OK		
Example	Host System ->Modem : AT*ICT*COUNTRY=? <0x0D> Modem -> Host System : *ICT*COUNTRY:OK KR<0x0D><0x0A> or Host System ->Modem : AT*ICT*COUNTRY=KR <0x0D> Modem -> Host System : *ICT*COUNTRY:OK<0x0D><0x0A>		

2.2.24 AT*ICT*WDS

Description	WDS(Wireless Distribution System) mode Set
Usage	AT*ICT*WDS=command {value / mac address}
Parameters	[command] ?: Return WDS Status 0 : Set WDS enable – value [0 : disable, 1 : enable] 1: Add WDS peer MAC address - You can add up to two WDS address 2 : Delete WDS peer MAC address
Response	[Command - ?] *ICT*WDS:OK {WDS enable} {WDS peer address1} {WDS peer address2} WDS enable – 0 : WDS disable, 1: WDS enable WDS peer address : WDS peer MAC address

	● 00:00:00:00:00:00 or FF:FF:FF:FF:FF:FF WDS peer address is not available [Others] *ICT*WDS:OK or *ICT*WDS:ERROR [Error Code]
Example	[Get WDS info] Host System ->Modem : AT*ICT*WDS=? <0x0D> Modem -> Host System : *ICT*WDS:OK 1 01:02:03:04:05:06<0x0D><0x0A> [WDS enable] Host System ->Modem : AT*ICT*WDS=0 1 <0x0D> Modem -> Host System : *ICT*WDS:OK<0x0D><0x0A> [WDS disable] Host System ->Modem : AT*ICT*WDS=0 0 <0x0D> Modem -> Host System : *ICT*WDS:OK<0x0D><0x0A> [Add WDS address] Host System ->Modem : AT*ICT*WDS=1 01:02:03:04:05:06 <0x0D> Modem -> Host System : *ICT*WDS:OK<0x0D><0x0A> Host System ->Modem : AT*ICT*WDS=1 01:02:03:04:05:07 <0x0D> Modem -> Host System : *ICT*WDS:OK<0x0D><0x0A> [Delete WDS address] Host System ->Modem : AT*ICT*WDS=2 01:02:03:04:05:07 <0x0D> Modem -> Host System : *ICT*WDS:OK<0x0D><0x0A>

2.2.25 AT*ICT*TXGAIN

Description	TX Gain Power Reduce
Usage	AT*ICT*TXGAIN={value}
Parameters	[command] Value = Reduce Power (0.25dB per 1)
Response	*ICT*TXGAIN:OK or *ICT*TXGAIN:ERROR [Error Code]
Example	[Reduce Power] ex) 1dB -> 4 Host System ->Modem : AT*ICT*TXGAIN=4<0x0D> Modem -> Host System : *ICT*TXGAIN:OK<0x0D><0x0A>

2.3 Wi-Fi Direct

2.3.1 AT*ICT*P2P_CONFIG(Optional)

Description	The default behavior is to run a single full scan in the beginning and then scan only social channels
Usage	AT*ICT*P2P_CONFIG= <type> <method> <#PIN>
Parameters	<command> ? : Configure Status <type> 0 : Passive – Automatically process P2P negotiation by WF5000. 1: Active – Host directly controlled P2P negotiation using P2P AT command <Method> 0 : PBC 1 : PIN 2 : BOTH <PIN#> PIN number : 8 digits, default PIN number is 00000000
Response	[Configure Status] Host System -> Modem : AT*ICT*P2P_CONFIG=?<0x0D> *ICT* P2P_CONFIG:OK [type] [method] [pin] [Others] *ICT* P2P_CONFIG:OK Or *ICT* P2P_CONFIG:ERROR [Error Code]
Example	[Get] Host System -> Modem : AT*ICT*P2P_CONFIG=?<0x0D> Modem -> Host System : *ICT*P2P_CONFIG:OK 0 2 12345678<0x0D><0x0A> [Set] Host System -> Modem : AT*ICT*P2P_CONFIG=1 2 12345678<0x0D> Modem -> Host System : *ICT*P2P_CONFIG:OK<0x0A> Or Host System -> Modem : AT*ICT*P2P_CONFIG=1 0<0x0D> Modem -> Host System : *ICT*P2P_CONFIG:OK<0x0A>

2.3.2 AT*ICT*P2P_FIND (Optional)

Description	The default behavior is to run a single full scan in the beginning and then scan only social channels
Usage	AT*ICT*P2P_FIND
Parameters	None

Response	*ICT* P2P_FIND:OK or *ICT* P2P_FIND:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*P2P_FIND<0x0D> Modem -> Host System : *ICT*P2P_FIND:OK<0x0D><0x0A>

2.3.3 AT*ICT*P2P_CONNECT (Optional)

Description	Start P2P group formation with a discovered P2P peer. "pbc" string starts pushbutton method, PIN# means that a pre-selected PIN can be used (e.g., 123456780).
Usage	AT*ICT*P2P_CONNECT=<peer device address> <pbc | PIN#>
Parameters	Peer device address: [device address] Method: [pbc or PIN number], PIN number is 8 digits
Response	*ICT* P2P_CONNECT:OK or *ICT* P2P_CONNECT:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*P2P_CONNECT=**02:01:02:03:04:05 pbc**<0x0D> Modem -> Host System : *ICT*P2P_CONNECT:OK<0x0D><0x0A> or Host System -> Modem : AT*ICT*P2P_CONNECT=**02:01:02:03:04:05 12345678**<0x0D> Modem -> Host System : *ICT*P2P_CONNECT:OK<0x0D><0x0A>

2.3.4 AT*ICT*P2P_CANCEL (Optional)

Description	Cancel an ongoing P2P group formation related operation.
Usage	AT*ICT*P2P_CANCEL
Parameters	None
Response	*ICT* P2P_CANCEL:OK or *ICT* P2P_CANCEL:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*P2P_CANCEL<0x0D> Modem -> Host System : *ICT*P2P_CANCEL:OK<0x0D><0x0A>

2.3.5 AT*ICT*P2P_REJECT (Optional)

Description	Reject an ongoing P2P group formation related operation.
Usage	AT*ICT*P2P_REJECT [peer address]
Parameters	Peer address
Response	*ICT* P2P_REJECT:OK

	or *ICT* P2P_REJECT:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*P2P_REJECT =00:01:02:03:04:05<0x0D> Modem -> Host System : *ICT*P2P_REJECT:OK<0x0D><0x0A>

2.3.6 AT*ICT*WPS_PBC

Description	WPS (Wi-Fi Protected Setup) PBC is connection system by push button method. Push-Button-Method, in which the user has to push a button, either an actual or virtual one, on both the access point and the new wireless client device.
Usage	AT*ICT*WPS_PBC=[any | bssid] Refer to Chap 3. Example Sequence for AT Commands
Parameters	any : Connect to any other AP that was supported WPS PBC method. bssid : Connect to the AP with a specific bssid that was supported WPS PBC method
Response	*ICT* WPS_PBC:OK or *ICT* WPS_PBC:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*WPS_PBC<0x0D> Modem -> Host System : *ICT*WPS_PBC:OK<0x0D><0x0A>

2.3.7 AT*ICT*WPS_PIN

Description	WPS (Wi-Fi Protected Setup) PIN is connection system by pin input method. PIN Method, in which a personal identification number (PIN) has to be read from either a sticker or the display on the new wireless device (AP or other devices). This PIN must then be entered at the represent pin number of the network, usually the access point of the network. Time input period was set default as two minutes. Alternately, a PIN on the Access Point may be entered into the new device.
Usage	AT*ICT*WPS_PIN=[pin] Refer to Chap 3. Example Sequence for AT Commands
Parameters	pin : 8 number digits
Response	*ICT* WPS_PIN:OK or *ICT* WPS_PIN:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*WPS_PIN<0x0D> Modem -> Host System : *ICT*WPS_PIN:OK<0x0D><0x0A>

2.3.8 AT*ICT*WPS_CANCEL

Description	WPS CANCEL used for stopping WPS PBC/PIN connection
Usage	AT*ICT*WPS_CANCEL
Parameters	None
Response	*ICT*WPS_CANCEL:OK or *ICT* WPS_CANCEL:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*WPS_CANCEL<0x0D> Modem -> Host System : *ICT*WPS_CANCEL:OK<0x0D><0x0A>

2.4 TCP/UDP Data Communication

2.4.1 Data Communication in AT command mode

AT command is used as a basic communication interface between Host system and WF5000. It is called as AT command mode or Command mode.
WF5000 could parse and process AT commands, such as Wi-Fi management, TC/UDP socket, and Internet Protocol, which are received from Host system on this mode.
Additionally, if it is necessary to open multi-sockets, Host system could send data frames to WF5000 using below AT commands on the mode.
In this case, user data should be sent after byte stuffing processing on Host system to WF5000.
AT*ICT*SEND / AT*ICT*SENDTO / AT*ICT*RECV / AT*ICT*RECVFROM.
If data communication is only working with single socket, Data mode would be used.

The Host System must use every TCP/IP command by using Socket Descriptor received as a response to "AT*ICT*SOCKET" command which is the socket creation command.
A response to the "AT*ICT*CONNECT" command is "*ICT*CONNECT:OK", which shows that the Modem received the command normally, it means that there is no error grammatically, not that the TCP connection is actually completed at a remote location.
When the TCP connection with the remote location is completed, "*ICT*CONNECTED:<socket descriptor>" event is received from the Modem asynchronously. If the TCP connection is released at the remote location by "AT*ICT*CLOSE" command, then "*ICT*CLOSED:<socket descriptor>" event will be received from the Modem. To run a TCP server on the Host System, it must be to use commands such as "AT*ICT*SOCKET, AT*ICT*BIND, and AT*ICT*LISTEN". Once the relevant port is listened, it is possible to allow TCP Clients for remote access. It can generate up to 6 sockets, and also generate and operate up to 3 TCP connections and up to 3 UDP connections.

2.4.2 Data communication in Data mode

Host system could directly send TCP/UDP data on Data mode instead of using AT commands on Command mode.
Transparent data transmission is provided between UART interface of Host system and that of WF5000.
It could be simply used for data communication without byte stuffing processing used on Command mode.
On the other hands, single socket have to be only open to use the data communication method of Data mode.
It is not possible to determine the destination of the actual data that you want to receive and send, if you use a multi-socket into data mode.

2.4.2.1 Switch from Command Mode to Data Mode

AT*ICT*DATA_SOCKET command could be used to enter the Data mode.
The command should be rejected if the other socket (TCP or UDP) had been previously open.

If the socket (TCP or UDP) is normally open, WF5000 should enter immediately into Data mode after transmitting *ICT*DATAMODE event to Host system.

2.4.2.2 Switch from Data Mode to Command Mode

Please type the "+++" character sequence to switch from the Data mode to AT command mode.
If WF5000 receives the "+++" character sequence from Host system, it changes its mode to AT command mode from Data mode after closing the socket which is used for data communication.
Also, if WF5000 is disconnected from the peer device communicating on Data mode, it is automatically changed to AT command mode.
To prevent the "+++" escape sequence from being misinterpreted as data, it should comply to following sequence.
No characters entered for 500ms after entering "+++" characters.
"+++" characters entered with no characters in between.

2.4.3 AT*ICT*DATA_SOCKET

Description	Open a socket with Data mode
Usage	AT*ICT*DATA_SOCKET=<socket_type> <remote_ip> <remote_port> <local_port>
Parameters	Socket_type : 1 - TCP client, 2 - UDP client, 4 - TCP server, 8 - UDP server Remote_ip : valid when operating as TCP/UDP client Remote_port : valid when operating as TCP/UDP client Local_port : valid when operating as TCP/UDP server
Response	*ICT*DATA_SOCKET:OK or *ICT*DATA_SOCKET:ERROR <error code>
Example	1) Create TCP client Host System -> Modem : AT*ICT*DATA_SOCKET=**1 192.168.0.2 60000 0**<0x0D> Modem -> Host System : *ICT*DATA_SOCKET:OK **0**<0x0D><0x0A> 2) Create TCP server Host System -> Modem : AT*ICT*DATA_SOCKET=**4 0 0 50000**<0x0D> Modem -> Host System : *ICT*DATA_SOCKET:OK **0**<0x0D><0x0A>

2.4.4 AT*ICT*DATA_INTERVAL

Description	Set a data mode transmit interval
Usage	AT*ICT*DATA_INTERVAL=<interval_ms>
Parameters	interval_ms: 10ms ~ 1000ms.

	If set not supported range, it works with default value. (typically 200ms)
Response	*ICT*DATA_INTERVAL:OK or *ICT*DATA_ INTERVAL:ERROR <error code>
Example	Host System -> Modem : AT*ICT*DATA_INTERVAL=?<0x0D> Modem -> Host System : *ICT*DATA_INTERVAL:OK **200**<0x0D><0x0A> Host System -> Modem : AT*ICT*DATA_INTERVAL=**100**<0x0D> Modem -> Host System : *ICT*DATA_INTERVAL:OK<0x0D><0x0A>

2.4.5 AT*ICT*SOCKET

Description	Open a socket
Usage	AT*ICT*SOCKET=<socket_type> Refer to Chap 3. Example Sequence for AT Commands
Parameters	1 - TCP socket type 2 - UDP socket type 3 - reserved 4 – TCP SSL Client socket type
Response	*ICT*SOCKET:OK <socket descriptor> or *ICT*SOCKET:ERROR <error code> Valid Socket descriptor: 0~5 (max 6 socket descriptor, TCP : 3/UDP : 3) 6 (TCP SSL Client socket descriptor)
Example	Host System -> Modem : AT*ICT*SOCKET=1<0x0D> Modem -> Host System : *ICT*SOCKET:OK **0**<0x0D><0x0A>

2.4.6 AT*ICT*CLOSE

Description	Close a socket
Usage	AT*ICT*CLOSE=<socket_descriptor> Refer to Chap 3. Example Sequence for AT Commands
Parameters	Socket descriptor to be returned from AT*ICT*SOCKET command
Response	*ICT*CLOSE:OK or

	*ICT*CLOSE:ERROR <error code>
Example	Host System -> Modem : AT*ICT*CLOSE=0<0x0D> Modem -> Host System : *ICT*CLOSE:OK<0x0D><0x0A>

2.4.7 AT*ICT*CONNECT

Description	Try to connect to remote peer with the remote IP address and the specified port
Usage	AT*ICT*CONNECT=<socket_descriptor> <ip_addr> <rport> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor ip address: ex) 192.168.100.1 remote port: ex) 50000
Response	*ICT*CONNECT:OK or *ICT*CONNECT:ERROR <error code>
Example	Host System -> Modem : AT*ICT*CONNECT=**0** **192.168.100.1** **50000**<0x0D> Modem -> Host System : *ICT*CONNECT:OK<0x0D><0x0A>

2.4.8 AT*ICT*DISCONNECT

Description	Try to disconnect remote client with the remote IP address and the specified port when operating TCP server.
Usage	AT*ICT*DISCONNECT=<socket_descriptor> <ip_addr> <rport>
Parameters	socket descriptor: server socket descriptor ip address: ex) 192.168.100.1 remote port: ex) 50000
Response	*ICT*DISCONNECT:OK or *ICT*DISCONNECT:ERROR <error code>
Example	Host System -> Modem : AT*ICT*DISCONNECT=**0** **192.168.100.10** **7986**<0x0D> Modem -> Host System : *ICT*DISCONNECT:OK<0x0D><0x0A>

2.4.9 AT*ICT*SEND

Description	Transmit a tcp stream data to a socket
Usage	AT*ICT*SEND=<socket_descriptor> <ip_addr> <rport> <size> <stream_data> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor ip address remote port size : stream data size - 1460 bytes maximum stream data : payload with escaping character
Response	*ICT*SEND:OK or *ICT*SEND:ERROR <error code>
Example	1) There is TCP SERVER operating Host System -> Modem : AT*ICT*SEND=**0 192.168.0.64 9100 5 Hello**<0x0D> Modem -> Host System : *ICT*SEND:OK<0x0D><0x0A> 2) IP addr, rport set to 0, when TCP client operating Host System -> Modem : AT*ICT*SEND=**0 0 0 5 Hello**<0x0D> () Modem -> Host System : *ICT*SEND:OK<0x0D><0x0A>

2.4.10 AT*ICT*SENDTO

Description	Transmit a udp datagram data to a socket on the remote IP address and the specified port
Usage	AT*ICT*SENDTO=<socket_descriptor> <ip_addr> <rport> <size> <datragram_data> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor ip address remote port size datagram data : payload with escaping character
Response	*ICT*SENDTO:OK or *ICT*SENDTO:ERROR <error code>
Example	Host System -> Modem : AT*ICT*SENDTO=**0 192.168.0.127 1000 5 Hello**<0x0D> Modem -> Host System : *ICT*SENDTO:OK<0x0D><0x0A>

2.4.11 AT*ICT*BIND

Description	Bind the socket to the specified port
Usage	AT*ICT*BIND=<socket_descriptor> <lport>

	Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor local port
Response	*ICT*BIND:OK or *ICT*BIND:ERROR <error code>
Example	Host System -> Modem : AT*ICT*BIND=**0** **5000**<0x0D> Modem -> Host System : *ICT*BIND:OK<0x0D><0x0A>

2.4.12 AT*ICT*LISTEN

Description	Open a listening tcp socket on the local IP address and the specified port
Usage	AT*ICT*LISTEN=<socket_desccriptor> Refer to Chap 3. Example Sequence for AT Commands
Parameters	Socket descriptor
Response	*ICT*LISTEN:OK or *ICT*LISTEN:ERROR <error code>
Example	Host System -> Modem : AT*ICT*LISTEN=0<0x0D> Modem -> Host System : *ICT*LISTEN:OK<0x0D><0x0A>

2.4.13 AT*ICT*LSTATUS

Description	Retrieve active connection established with a listening TCP socket when working as TCP server
Usage	AT*ICT*LSTATUS=<socket_desccriptor>
Parameters	Socket descriptor
Response	*ICT* LSTATUS:OK <socket> [remote ipaddr] [remote port] [remote ipaddr] [remote port] ... or *ICT* LSTATUS:ERROR <error code>
Example	1) There is no connection Host System -> Modem : AT*ICT*LSTATUS=0<0x0D> Modem -> Host System : *ICT* LSTATUS:OK **0**<0x0D><0x0A> 2) There are three active connections. Host System -> Modem : AT*ICT*LSTATUS=0<0x0D> Modem -> Host System : *ICT* LSTATUS:OK **0** **192.168.123.111** **1691** **192.168.123.130** **9100** **192.168.123.142** **5500**<0x0D><0x0A>

2.4.14 AT*ICT*NWSTATUS

Description	Get current Wireless and Network status
Usage	AT*ICT* NWSTATUS =?
Parameters	?
Response	*ICT* NWSTATUS:OK <mac_addr> <network_type> <channel> <rssi> <ssid> <bssid> <security_type> <dhcp_mode> <ip> <subnet> <gateway> <dns> or *ICT* NWSTATUS:ERROR <error code> network_type : 1 - AdHoc 2 - Station 3 - SoftAP rssi : Absolute value of the RSSI, valid when operating as Station mode ssid : Null (default) security_type : 0 - OPEN 1 - WEP 2 - WPA_PSK 3 - WPA2_PSK dhcp_mode : valid when operating as Station mode 0 - Static 1- DHCP enabled
Example	1) Nothing Host System -> Modem : AT*ICT*NWSTATUS=?<0x0D> Modem -> Host System : *ICT* NWSTATUS:OK **84:72:07:12:34:56 2 6 -0 Null FF:FF:FF:FF:FF:FF 3 1 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0**<0x0D><0x0A> 2) Operating as Station mode Host System -> Modem : AT*ICT*NWSTATUS=?<0x0D> Modem -> Host System : *ICT* NWSTATUS:OK **84:72:07:12:34:56 2 6 -51 Standalone 00:08:9F:42:44:E0 3 1 192.168.0.68 255.255.255.0 192.168.0.1 164.124.101.2**<0x0D><0x0A> 3) Operating as SoftAP mode Host System -> Modem : AT*ICT*NWSTATUS=?<0x0D> Modem -> Host System : *ICT* NWSTATUS:OK **84:72:07:12:34:56 3 11 -0 TestAP 84:72:07:12:34:56 3 1 192.168.55.1 255.255.255.0 192.168.55.1 0.0.0.0**<0x0D><0x0A>

2.4.15 AT*ICT*IPCONFIG

Description	Configure the static IP parameters or Retrieve the IP parameters to operate as Station or AdHoc mode
Usage	1) AT*ICT*IPCONFIG=? or 2) AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway]
Parameters	1) ? or 2) dhcp mode : 0 - static 1 - DHCP enabled (default) ip / subnet / gateway : valid when the dhcp mode is static.
Response	*ICT*IPCONFIG:OK [dhcp mode] [ip] [subnet] [gateway] [dns] or *ICT*IPCONFIG:ERROR <error code>
Example	1) Retrieve IP information Host System -> Modem : AT*ICT*IPCONFIG=?<0x0D> Modem -> Host System : *ICT*IPCONFIG:OK **1 192.168.100.100 255.255.255.0 192.168.100.1 164.124.101.2**<0x0D><0x0A> 2) Configure static mode Host System->Modem : AT*ICT*IPCONFIG=**0 192.168.0.1 255.255.255.0 192.168.0.1**<0x0D> Modem->Host System : *ICT*IPCONFIG:OK<0x0D><0x0A> 3) Configure DHCP mode Host System->Modem : AT*ICT*IPCONFIG=1<0x0D> Modem->Host System : *ICT*IPCONFIG:OK<0x0D><0x0A>

2.4.16 AT*ICT*APNSET

Description	Configure or Retrieve the SoftAP network parameters to operate as SoftAP mode
Usage	1) AT*ICT*APNSET=? or 2) AT*ICT*APNSET=<ip> <subnet> <gateway>
Parameters	1) ? or 2) ip / subnet / gateway Default :

	ip - 192.168.0.1 subnet - 255.255.255.0 gateway - 192.168.0.1
Response	*ICT*APNSET:OK [ip] [subnet] [gateway] or *ICT*APNSET:ERROR <error code>
Example	1) Retrieve Network information Host System -> Modem : AT*ICT*APNSET=?<0x0D> Modem -> Host System : *ICT*APNSET:OK **192.168.0.1** **255.255.255.0** **192.168.0.1**<0x0D><0x0A> 2) Configure Network parameters Host System->Modem : AT*ICT*APNSET=**192.168.55.1** **255.255.255.0** **192.168.55.1**<0x0D> Modem->Host System : *ICT*APNSET:OK<0x0D><0x0A>

2.4.17 AT*ICT*APLEASEIP

Description	Configure or Retrieve the range of IP address allocated to STAs on SoftAP mode.
Usage	1) AT*ICT*APLEASEIP=? or 2) AT*ICT*APLEASEIP=< lease_ip_min><lease_ip_max>
Parameters	1) ? or 2) ip / subnet / gateway Default : lease_ip_min - 10 lease_ip_max - 254
Response	*ICT*APNSET:OK [lease_ip_min][lease_ip_max] or *ICT*APNSET:ERROR <error code>
Example	1) Retrieve Network information Host System -> Modem : AT*ICT*APLEASEIP=?<0x0D> Modem -> Host System : *ICT*APLEASEIP:OK **10** **254**<0x0D><0x0A> 2) Configure Network parameters Host System->Modem : AT*ICT*APLEASEIP=**10** **254**<0x0D> Modem->Host System : *ICT*APLEASEIP:OK<0x0D><0x0A>

2.4.18 AT*ICT*DNSQUERY

Description	Request DNS query
Usage	AT*ICT*DNSQUERY=<host name>

Parameters	Host name to query
Response	*ICT*DNSQUERY:OK or *ICT*DNSQUERY:ERROR <error code>
Example	Host System -> Modem : AT*ICT*DNSQUERY=www.inctech.co.kr<0x0D> Modem -> Host System : *ICT*DNSQUERY:OK<0x0D><0x0A> Modem -> Host System : *ICT*DNSRESPONSE:210.107.64.131<0x0D><0x0A>

2.4.19 AT*ICT*HTTPGET

Description	Request HTTP GET message to HTTP server
Usage	AT*ICT*HTTPGET=<ip/domain>:<port><uri>
Parameters	ip/domain – domain name or IP address of HTTP server port – port number of HTTP server uri – URI
Response	*ICT*HTTPGET:OK or *ICT*HTTPGET:ERROR
Example	Host System -> Modem : AT*ICT*HTTPGET=**192.168.0.1/index.shtml**<0x0D> Modem -> Host System : *ICT*HTTPGET:OK<0x0D><0x0A> or Host System -> Modem : AT*ICT*HTTPGET=**webserver.net:8008/login/login.php?id=58394**<0x0D> Modem -> Host System : *ICT*HTTPGET:OK<0x0D><0x0A>

2.4.20 AT*ICT*HTTPSET

Description	Set HTTP Message Content-Type.
Usage	AT*ICT*HTTPSET=<option> <value>
Parameters	option - default 0 value - 0 (URLENCODED) - 1 (OCTETSTREAM) - 2 (JSON)
Response	*ICT*HTTPSET:OK or *ICT*HTTPSET:ERROR
Example	Host System -> Modem : AT*ICT*HTTPSET=**0 0**<0x0D> Modem -> Host System : *ICT*HTTPSET:OK<0x0D><0x0A>

2.4.21 AT*ICT*HTTPPOST

Description	Request HTTP POST message to HTTP server
Usage	AT*ICT*HTTPPOST=<ip/domain>:<port><uri> (length) (octet-stream)
Parameters	ip/domain – domain name or IP address of HTTP server port – port number of HTTP server uri – URI length - length of octet-stream octet-stream - body of post message
Response	*ICT*HTTPPOST:OK or *ICT*HTTPPOST:ERROR
Example	Host System -> Modem : AT*ICT*HTTPPOST=**192.168.0.1/index.shtml**<0x0D> Modem -> Host System : *ICT*HTTPPOST:OK<0x0D><0x0A> or Host System -> Modem : AT*ICT*HTTPPOST=**webserver.net:8008/login/login.php?id=58394**<0x0D> Modem -> Host System : *ICT*HTTPPOST:OK<0x0D><0x0A> or Host System -> Modem : AT*ICT*HTTPPOST=**webserver.net:8008/post.php 19 {"param1","value1"}**<0x0D> Modem -> Host System : *ICT*HTTPPOST:OK<0x0D><0x0A>

2.4.22 AT*ICT*HTTPSTOP

Description	Close http session
Usage	AT*ICT*HTTPSTOP=<type>
Parameters	Type 0 – HTTP session 1 – HTTPS session
Response	*ICT*HTTPSTOP:OK → HTTP session close is success or *ICT*HTTPSTOP:ERROR → HTTP session close is failed ✓ No HTTP session socket ✓ Receiving HTTP data
Example	Host System ->Modem : AT*ICT*HTTPSTOP=1<0x0D> Modem -> Host System : *ICT*HTTPSTOP:OK<0x0D><0x0A>

2.4.23 AT*ICT*HTTPHEADER

Description	Add HTTP header when sending HTTP GET/POST message

Usage	AT*ICT*HTTPHEADER=<length> <header>
Parameters	length - length of header (stuffed length) header - octet-stream of header (stuffed stream)
Response	*ICT*HTTPHEADER:OK → HTTP header customizing was success. or *ICT*HTTPHEADER:ERROR [Error Code] → HTTP header customizing was failed. > Error Code - 7 : Out of Memory - 8 : Invalid Parameter
Example	Host System ->Modem : AT*ICT*HTTPHEADER=11 PARAM:VALUE<0x0D> Modem -> Host System : *ICT*HTTPHEADER:OK<0x0D><0x0A>

2.4.24 AT*ICT*PING

Description	send ping request
Usage	AT*ICT*PING=<Repeat count> <Target IP> <data size>
Parameters	Repeat count : repeat number for ping request Target IP : peer ip address Data size : data size 1460 bytes maximum
Response	*ICT*PING:OK or *ICT*PING:ERROR [Error Code]
Example	Host System ->Modem : AT*ICT*PING=10 192.168.1.1 1460 <0x0D> Modem -> Host System : *ICT*PING:OK 0<0x0D><0x0A>

2.4.25 AT*ICT*UPNP_EXTIP

Description	Get external IP address of AP using UPNP
Usage	AT*ICT*UPNP_EXTIP Refer to Chap 3.4.5 UPNP_EXTIP
Parameters	None
Response	*ICT*UPNP_EXTIP:OK or *ICT*UPNP_EXTIP:ERROR
Example	Host System -> Modem : AT*ICT*UPNP_EXTIP<0x0D> Modem -> Host System : *ICT*UPNP_EXTIP:OK<0x0D><0x0A>

2.4.26 AT*ICT*UPNP_ADDPORTMAP

Description	Add portmapping in AP using UPNP
Usage	AT*ICT*UPNP_ADDPORTMAP=<ip> <port_int> <port_ext> <protocol> [description] Refer to Chap 3.4.6 UPNP_ADDPORTMAP
Parameters	ip – IP address of client port_int – internal port number port_ext – external port number protocol: 1 - TCP 2 - UDP description – portmapping desciption (optional)
Response	*ICT*UPNP_ADDPORTMAP:OK or *ICT*UPNP_ADDPORTMAP:ERROR
Example	Host System -> Modem : AT*ICT*UPNP_ADDPORTMAP=192.168.0.100 12345 54321 1 INC_54321<0x0D> Modem -> Host System : *ICT*UPNP_ADDPORTMAP:OK<0x0D><0x0A>

2.4.27 AT*ICT*UPNP_DELPORTMAP

Description	Delete portmapping in AP using UPNP
Usage	AT*ICT*UPNP_DELPORTMAP=<port_ext> <protocol> Refer to Chap 3.4.7 UPNP_DELPORTMAP
Parameters	port_ext – external port number protocol – TCP/UDP
Response	*ICT*UPNP_DELPORTMAP:OK or *ICT*UPNP_DELPORTMAP:ERROR
Example	Host System -> Modem : AT*ICT*UPNP_DELPORTMAP=54321 1<0x0D> Modem -> Host System : *ICT*UPNP_DELPORTMAP:OK<0x0D><0x0A>

2.4.28 AT*ICT*LPD_START (Optional)

Description	Start LPD service
Usage	1) AT*ICT*LPD_START or

	2) AT*ICT*LPD_START=<queue_name> If the queue name is not assigned, default queue name is "lpd0"
Parameters	None
Response	1) *ICT*LPD_START:OK or 2) *ICT*LPD_START:ERROR [Error Code]
Example	1) Host System -> Modem : AT*ICT*LPD_START<0x0D> Modem -> Host System : *ICT*LPD_START:OK<0x0D><0x0A> 2) Host System -> Modem : AT*ICT*LPD_START=lpd1<0x0D> Modem -> Host System : *ICT*LPD_START:OK<0x0D><0x0A>

2.4.29 AT*ICT*LPD_STOP (Optional)

Description	Stop LPD service
Usage	AT*ICT*LPD_STOP
Parameters	None
Response	*ICT*LPD_STOP:OK or *ICT*LPD_STOP:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*LPD_STOP<0x0D> Modem -> Host System : *ICT*LPD_STOP:OK<0x0D><0x0A>

2.4.30 AT*ICT*DDNS_GETIP

Description	Get external public IP address of AP using UPNP
Usage	AT*ICT*DDNS_GETIP=?
Parameters	?
Response	*ICT*DDNS_GETIP:OK or *ICT*DDNS_GETIP:ERROR
Example	Host System -> Modem : AT*ICT*DDNS_GETIP<0x0D> Modem -> Host System : *ICT*DDNS_GETIP:OK<0x0D><0x0A>

2.4.31 AT*ICT*DDNS_UPDATE

Description	Request DDNS UPDATE GET message to DDNS server

Usage	AT*ICT*DDNS_UPDATE=<DDNS server> <host name> <DDNS username> <DDNS password> <DDNS timer>
Parameters	DDNS server – Select DDNS server 0 – noip.com 1 – dyndns.com host name - hostnames that you wish to update DDNS username – username(or id) of DDNS server DDNS password – password of DDNS server DDNS timer – DDNS repetition update period (Unit : minute) 0 – 40320 minutes (default value = 28 days) or value
Response	*ICT*DDNS_UPDATE:OK or *ICT*DDNS_UPDATE:ERROR
Example	Host System -> Modem: AT*ICT*DDNS_UPDATE=**1 mycompany.dyndns-free.com mydyndns_id mydyndns_pw 0**<0x0D> Modem -> Host System: *ICT*DDNS_UPDATE:OK<0x0D><0x0A>

2.4.32 AT*ICT*SNTP

Description	Request SNTP GET message to NTP server
Usage	AT*ICT*SNTP
Parameters	SNTP command gets GMT time from multi NTP server. Multi NTP server URL 1. User defined URL (default URL: pool.ntp.org) 2. time.nist.gov 3. time.windows.com
Response	*ICT*SNTP:OK or *ICT*SNTP:ERROR 4
Example	1) Modem was connected with an AP Host System -> Modem: AT*ICT*SNTP<0x0D> Modem -> Host System: *ICT*SNTP:OK<0x0D><0x0A> 2) Not connected Host System -> Modem: AT*ICT*SNTP<0x0D> Modem -> Host System: *ICT*SNTP:ERROR 4<0x0D><0x0A>

2.4.33 AT*ICT*SNTP_GET

Description	Get SNTP Client Properties

Usage	AT*ICT*SNTP_GET=[sntp index]
Parameters	SNTP GET index

Index num	Name	Parameters
0	NTP Server URL or IP ADDRESS	at*ict*sntp_get=0
1	Change GMT value	at*ict*sntp_get=1

Response	*ICT*SNTP_GET:OK or *ICT*SNTP_GET:ERROR
Example	Host System -> Modem: AT*ICT*SNTP_GET=0<0x0D> Modem -> Host System: *ICT*SNTP_GET:OK 9<0x0D><0x0A> or Host System -> Modem: AT*ICT*SNTP_GET=1<0x0D> Modem -> Host System: *ICT*SNTP_GET:OK pool.ntp.org<0x0D><0x0A>

2.4.34 AT*ICT*SNTP_SET

Description	Set SNTP Client Properties
Usage	AT*ICT*SNTP_SET=[sntp index] [value]
Parameters	SNTP SET index NTP Server URL limited length is 32 under.

Index num	Name	Parameters
0	NTP Server URL or IP ADDRESS	at*ict*sntp_set=0 [IP addr or URL]
1	Change GMT value	at*ict*sntp_set=1 [Time Zone difference from GMT]

Response	*ICT*SNTP_SET:OK [Index num] or *ICT*ERROR [Error Code]
Example	1) NTP Server Setting Host System -> Modem: AT*ICT*SNTP_SET=0 time.windows.com<0x0D> Modem -> Host System: *ICT*SNTP_SET:OK<0x0D><0x0A> or Host System -> Modem: AT*ICT*SNTP_SET=0 23.99.222.162<0x0D> Modem -> Host System: *ICT*SNTP_SET:OK<0x0D><0x0A> 2) GMT setting (ex.. Time Zone – Seoul : GMT + 9) Host System -> Modem: AT*ICT*SNTP_GET=1 9<0x0D> Modem -> Host System: *ICT*SNTP_GET:OK<0x0D><0x0A>

2.4.35 AT*ICT*FTPC_SET

Description	Set FTP Client properties for connection
Usage	AT*ICT*FTPC_SET=[ftpc index] {Parameters}
Parameters	FTPC SET index Index num / Name / Parameters 0 / Login Identify / at*ict*ftpc_set=0 [login ID] 1 / Login Password / at*ict*ftpc_set=1 [login Password] Max length Login Identify - 32 byte Login Password - 64byte
Response	*ICT*FTPC_SET:OK or *ICT*ERROR [Error Code]
Example	Save ftp user account information for connection Host System ->Modem : AT*ICT*FTPC_SET=0 ftpc_login<0x0D> Modem -> Host System : *ICT*FTPC_SET:OK<0x0D><0x0A> or Host System ->Modem : AT*ICT*FTPC_SET=1 ftpc_password<0x0D> Modem -> Host System : *ICT*FTPC_SET:OK<0x0D><0x0A>

2.4.36 AT*ICT*FTPC_GET

Description	Get FTP Client properties
Usage	AT*ICT*FTPC_GET=[ftpc index]
Parameters	FTPC GET index Index num / Name / Parameters 0 / Login Identify / at*ict*ftpc_get=0 1 / Login Password / at*ict*ftpc_get=1
Response	*ICT*FTPC_GET:OK [return value] or *ICT*FTPC_GET:OK null → There is no assigned value. or *ICT*ERROR [Error Code]
Example	Get ftp information stored in NV memory Host System ->Modem : AT*ICT*FTPC_GET=0<0x0D> Modem -> Host System : *ICT*FTPC_GET:OK ftpc_login<0x0D><0x0A> or Host System ->Modem : AT*ICT*FTPC_GET=1<0x0D>

	Modem -> Host System : *ICT*FTPC_GET:OK ftpc_password<0x0D><0x0A>

2.4.37 AT*ICT*DHCPDSTART

Description	Start DHCP Server
Usage	AT*ICT*DHCPDSTART
Parameters	None
Response	*ICT*DHCPDSTART:OK or *ICT*DHCPDSTART:ERROR [Error Code]
Example	Host System -> Modem : AT*ICT*DHCPDSTART <0x0D> Modem -> Host System : *ICT*DHCPDSTART:OK<0x0D><0x0A>

2.4.38 AT*ICT*DHCPDSTOP

Description	Stop DHCP Server
Usage	AT*ICT*DHCPDSTOP
Parameters	None
Response	*ICT*DHCPDSTOP:OK or *ICT*DHCPDSTOP:ERROR
Example	Host System -> Modem : AT*ICT*DHCPDSTOP <0x0D> Modem -> Host System : *ICT*DHCPDSTOP:OK<0x0D><0x0A>

2.4.39 AT*ICT*NW_CONN

Description	Remote Network Connection Status
Usage	AT*ICT*NW_CONN
Parameters	[socket descriptor] -1 : not connected 0 ~ 3 : connected socket descriptor
Response	*ICT*NW_CONN:OK [socket descriptor]
Example	Host System -> Modem : AT*ICT*NW_CONN <0x0D> Modem -> Host System : *ICT*NW_CONN:OK -1<0x0D><0x0A> Host System -> Modem : AT*ICT*NW_CONN <0x0D> Modem -> Host System : *ICT*NW_CONN:OK 0<0x0D><0x0A>

2.5 TCP/SSL

2.5.1 AT*ICT*SSL_CLOSE

Description	Close a socket
Usage	AT*ICT*SSL_CLOSE=<socket_descriptor> Refer to Chap 3. Example Sequence for AT Commands
Parameters	Socket descriptor to be returned from AT*ICT*SOCKET command
Response	*ICT*SSL_CLOSE:OK or *ICT*SSL_CLOSE:ERROR <error code>
Example	Host System -> Modem : AT*ICT*SSL_CLOSE=0<0x0D> Modem -> Host System : *ICT*SSL_CLOSE:OK<0x0D><0x0A>

2.5.2 AT*ICT*SSL_CONNECT

Description	Try to connect to remote peer with the remote IP address and the specified port
Usage	AT*ICT*SSL_CONNECT=<socket_descriptor> <ip_addr> <rport> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor ip address: ex) 192.168.100.1 remote port: ex) 50000
Response	*ICT*SSL_CONNECT:OK or *ICT*SSL_CONNECT:ERROR <error code>
Example	Host System -> Modem : AT*ICT*SSL_CONNECT=**0** **192.168.100.1** **50000**<0x0D> Modem -> Host System : *ICT*SSL_CONNECT:OK<0x0D><0x0A>

2.5.3 AT*ICT*SSL_SEND

Description	Transmit a tcp stream data to a socket
Usage	AT*ICT*SSL_SEND=<socket_descriptor> <size> <stream_data> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor size : stream data size - 1460 bytes maximum stream data : payload with escaping character

Response	*ICT*SSL_SEND:OK or *ICT*SSL_SEND:ERROR <error code>
Example	1) There is TCP SERVER operating Host System -> Modem : AT*ICT*SSL_SEND=**0 5 Hello**<0x0D> Modem -> Host System : *ICT*SSL_SEND:OK<0x0D><0x0A>

2.5.4 AT*ICT*SSL_SVR_START

Description	Start TCP SSL server
Usage	AT*ICT*SSL_SVR_START=<socket_descriptor> <port> Refer to Chap 3. Example Sequence for AT Commands
Parameters	Socket descriptor : 0 Port : 1025 ~ 65000
Response	*ICT*SSL_SVR_START:OK or *ICT*SSL_SVR_START:ERROR <error code>
Example	Host System -> Modem : AT*ICT*SSL_SVR_START=0 5000<0x0D> Modem -> Host System : *ICT*SSL_SVR_START:OK<0x0D><0x0A>

2.5.5 AT*ICT*SSL_SVR_CLOSE

Description	Close TCP SSL server
Usage	AT*ICT*SSL_SVR_CLOSE=<socket_descriptor> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor : socket descriptor of SSL_SVR_START
Response	*ICT*SSL_SVR_CLOSE:OK or *ICT*SSL_SVR_CLOSE:ERROR <error code>
Example	Host System -> Modem : AT*ICT*SSL_SVR_CLOSE=**0**<0x0D> Modem -> Host System : *ICT*SSL_SVR_CLOSE:OK<0x0D><0x0A>

2.5.6 AT*ICT*SSL_SVR_SEND

Description	Transmit a tcp stream data to tcp ssl client at a socket
Usage	AT*ICT*SSL_SEND=<socket_descriptor> <size> <stream_data> Refer to Chap 3. Example Sequence for AT Commands
Parameters	socket descriptor : client socket descriptor that is acquired by ICT*SSL_SVR_ACCEPTED event

	size : stream data size - 1460 bytes maximum stream data : payload with escaping character
Response	*ICT*SSL_SVR_SEND:OK or *ICT*SSL_SVR_SEND:ERROR <error code>
Example	1) There is TCP SERVER operating Host System -> Modem : AT*ICT*SSL_SVR_SEND=**1 5 Hello**<0x0D> Modem -> Host System : *ICT*SSL_SVR_SEND:OK<0x0D><0x0A>

2.6 Web Server

2.6.1 AT*ICT*HTTPD_START

Description	Web Server Enable : Current Setting is saved in NV memory and maintained in next time
Usage	AT*ICT*HTTPD_START
Parameters	
Response	*ICT*HTTPD_START:OK or *ICT*HTTPD_START:ERROR <error code>
Example	Host System -> Modem : AT*ICT*HTTPD_START Modem -> Host System : *ICT*HTTPD_START:OK<0x0D><0x0A>

2.6.2 AT*ICT*HTTPD_STOP

Description	Web Server Disable : Current Setting is saved in NV memory and maintained in next time
Usage	AT*ICT*HTTPD_STOP
Parameters	
Response	*ICT*HTTPD_STOP:OK or *ICT*HTTPD_STOP:ERROR <error code>
Example	Host System -> Modem : AT*ICT*HTTPD_STOP Modem -> Host System : *ICT*HTTP_STOP:OK<0x0D><0x0A>

2.7 OTA

2.7.1 AT*ICT*OTA_VERCHECK

Description	Check the lastest firmware version at OTA server (using HTTP or FTP)
Usage	AT*ICT*OTA_VERCHECK=<url>
Parameters	url - URL of OTA firmware location
Response	*ICT*OTA_VERCHECK:OK or *ICT*OTA_VERCHECK:ERROR [error_code] [error_code] 1 - Initialization error 2 - Argument error
Example	Host System -> Modem : AT*ICT*OTA_VERCHECK=http://ota.domain.com:8080/ota/<0x0D> or Host System -> Modem : AT*ICT*OTA_VERCHECK=ftp://ota.domain.com:21/ota/<0x0D> Modem -> Host System : *ICT*OTA_VERCHECK:OK<0x0D><0x0A>

2.7.2 AT*ICT*OTA_REQUEST

Description	Request to download firmware at OTA server (using HTTP)
Usage	AT*ICT*OTA_REQUEST=<url>
Parameters	url - URL of OTA firmware location
Response	*ICT*OTA_REQUEST:OK or *ICT*OTA_REQUEST:ERROR [error_code] [error_code] 1 - Initialization error 2 - Argument error 3 - Server version less than local version 4 - Gettting server version of firmware using AT*ICT*OTA_VERCHECK before update
Example	Host System -> Modem : AT*ICT*OTA_REQUEST=http://ota.domain.com:8080/ota/<0x0D> or Host System -> Modem : AT*ICT*OTA_REQUEST=ftp://ota.domain.com:21/ota/<0x0D> Modem -> Host System : *ICT*OTA_REQUEST:OK<0x0D><0x0A>

2.8 Special Command

2.8.1 AT

Description	Check whether the Host System and the Modem are connected to each other
Usage	AT
Parameters	None
Response	OK
Example	Host System -> Modem : AT<0x0D> Modem -> Host System : OK<0x0D><0x0A>

2.8.2 ATE

Description	Entered commands can be echoed back to the Host System when the ATE command is used.
Usage	ATE<on/off>
Parameters	0 - echo off 1 - echo on
Response	OK or ERROR
Example	Host System -> Modem : ATE1<0x0D> Modem -> Host System : OK<0x0D><0x0A>

2.8.3 ATV (TBD)

Description	
Usage	
Parameters	
Response	
Example	Host System -> Modem : Modem -> Host System :

2.9 Event Information

2.9.1 *ICT*DEVICEREADY

Description	Notify that the Modem is ready to communicate with the HOST SYSTEM
Usage	*ICT*DEVICEREADY
Parameters	
Response	
Example	Modem -> Host System : *ICT*DEVICEREADY<0x0D><0x0A>

2.9.2 *ICT*INITSCAN

Description	Notify that the Modem try connecting to AP
Usage	*ICT*INITSCAN
Parameters	
Response	

2.9.3 *ICT*ASSOCIATED

Description	Notify that the Modem is connected to AP to which the Host System want to connect
Usage	*ICT*ASSOCIATED:<result>
Parameters	Result : 0 - Success 1 - Fail 2 - AP is not found 3 - Timeout 4 - Connection is restricted
Response	
Example	Modem -> Host System : *ICT*ASSOCIATED:0<0x0D><0x0A>

2.9.4 *ICT*DISASSOCIATED

Description	Notify that the Modem is disconnected from AP
Usage	*ICT*DISASSOCIATED
Parameters	
Response	
Example	Modem -> Host System : *ICT*DISASSOCIATED<0x0D><0x0A>

2.9.5 *ICT*SCANIND

Description	Notify that the Modem gives the scan indication to the Host System
Usage	*ICT*SCANIND:<no> <ssid> <bssid> <network_mode> <security> <channel> <rssi>
Parameters	no: index ssid: 32 characters maximum (If there is no SSID, this field displays "NULL" string) bssid network mode: 0 - Infrastructure 1 - IBSS (AdHoc) 2 - reserved 4 - reserved security: 0 - open 1 - WEP 2 - WPA_PSK 3 - WPA_Enterprise 4 - WPA2_PSK 5 - WPA2_Enterprise channel: 1~14 (2.4GHz) rssi
Response	
Example	Modem -> Host System : *ICT*SCANIND:**0 iptimes 00:18:39:22:FB:02 0 2 11 -63**<0x0D><0x0A>

2.9.6 *ICT*SCANRESULT

Description	Notify that the Modem gives the scan result to the Host System
Usage	*ICT*SCANRESULT
Parameters	
Response	
Example	Modem -> Host System : *ICT*SCANRESULT<0x0D><0x0A>

2.9.7 *ICT*RECV

Description	Notify that the Modem has received a tcp stream data at a socket
Usage	*ICT*RECV:<socket descriptor> <remote_ipaddr> <rport> <size> <stream_data>
Parameters	socket descriptor

	remote ip address remote port size : stream data size stream data
Response	
Example	Modem -> Host System : *ICT*RECV:**0 192.168.0.64 9100 10 abcdef12345**<0x0D><0x0A>

2.9.8 *ICT*RECVFROM

Description	Notify that the Modem has received a udp datagram data to a socket
Usage	*ICT*RECVFROM:<socket descriptor> <remote_ipaddr> <rport> <size> <sdatagram_data>
Parameters	socket descriptor remote ip address remote port size : datagram data size datagram data
Response	
Example	Modem -> Host System : *ICT*RECVFROM:**0 192.168.0.103 56362 10** **abcdef12345**<0x0D><0x0A>

2.9.9 *ICT*SSL_RECV

Description	Notify that the Modem has received a tcp ssl stream data at a socket
Usage	*ICT*SSL_RECV:<socket descriptor> <size> <stream data>
Parameters	socket descriptor size : stream data size stream data
Response	
Example	Modem -> Host System : *ICT*SSL_RECV:**0 11 abcdef12345**<0x0D><0x0A>

2.9.10 *ICT*SSL_IND

Description	Notify that the Modem has received a tcp ssl stream data at a socket
Usage	*ICT*SSL_INC:<socket descriptor> <type > <result>
Parameters	socket descriptor type : connect (0), send (1), receive (2) result : OK or ERROR
Response	
Example	Modem -> Host System (connect): *ICT*SSL_IND:**0 0 OK**<0x0D><0x0A> Modem -> Host System (send): *ICT*SSL_IND:**0 1 OK**<0x0D><0x0A>

2.9.11 *ICT*SSL_SVR_ACCEPTED

Description	Notify that the Modem has received to connect of tcp ssl client at a socket
Usage	*ICT*SSL_SVR_ACCEPTED:<socket descriptor>
Parameters	socket descriptor : socket descriptor value is constant integer to minimum value is zero
Response	
Example	Modem -> Host System : *ICT*SSL_SVR_ACCEPTED:**1** <0x0D><0x0A>

2.9.12 *ICT*SSL_SVR_CLOSED

Description	Notify that the Modem has received to close of tcp ssl client at a socket
Usage	*ICT*SSL_SVR_CLOSED:<socket descriptor>
Parameters	socket descriptor : socket descriptor value is constant integer to minimum value is zero
Response	
Example	Modem -> Host System : *ICT*SSL_SVR_CLOSED:**1** <0x0D><0x0A>

2.9.13 *ICT*SSL_SVR_RECV

Description	Notify that the Modem has received a tcp ssl client stream data at a socket
Usage	*ICT*SSL_SVR_RECV:<socket descriptor> <size> <stream_data>
Parameters	socket descriptor : socket descriptor value is constant integer to minimum value is zero size : stream data size stream data
Response	
Example	Modem -> Host System : *ICT*SSL_SVR_RECV:**1 11 abcdef12345**<0x0D><0x0A>

2.9.14 *ICT*SSL_SVR_IND

Description	Notify that the result of current tcp ssl status at a socket
Usage	*ICT*SSL_INC:<socket descriptor> <type> <result>
Parameters	socket descriptor type : connect (0), send (1), receive (2) result : OK or ERROR
Response	
Example	Modem -> Host System (connect): *ICT*SSL_SVR_IND:**0 0 OK**<0x0D><0x0A> Modem -> Host System (send): *ICT*SSL_SVR_IND:**0 1 OK**<0x0D><0x0A>

2.9.15 *ICT*IPALLOCATED

Description	Notify that the Modem IP is allocated

Usage	*ICT*IPALLOCATED:<ip_addr> <subnet> <gateway> <dns>
Parameters	ip address subnet mask default gateway dns server
Response	
Example	Modem -> Host System : *ICT*IPALLOCATED:**192.168.100.100 255.255.255.0 192.168.100.1 164.124.101.2**<0x0D><0x0A>

2.9.16 *ICT*IPRELEASED

Description	Notify that the Modem IP is released
Usage	*ICT*IPRELEASED
Parameters	
Response	
Example	Modem -> Host System : *ICT*IPRELEASED<0x0D><0x0A>

2.9.17 *ICT*CONNECTED

Description	Notify that the Modem TCP client has connected to the remote server
Usage	*ICT*CONNECTED:<socket_descriptor>
Parameters	
Response	
Example	Modem -> Host System : *ICT*CONNECTED:1<0x0D><0x0A>

2.9.18 *ICT*DISCONNECTED

Description	Notify that the remote client is disconnected when working as TCP server
Usage	*ICT*DISCONNECTED:<socket_descriptor> <remote_ipaddr> <rport>
Parameters	socket descriptor remote ip address remote port
Response	
Example	Modem -> Host System : *ICT*DISCONNECTED:**1 192.168.0.64 9100**<0x0D><0x0A>

2.9.19 *ICT*ACCEPTED

Description	Notify that the remote client is connected when working as TCP server
Usage	*ICT*ACCEPTED:<socket_descriptor> <remote_ipaddr> <rport>
Parameters	socket descriptor

	remote ip address remote port
Response	
Example	Modem -> Host System : *ICT*ACCEPTED: **1** **192.168.0.64** **9100**<0x0D><0x0A>

2.9.20 *ICT*CLOSED

Description	Notify that a socket is closed
Usage	*ICT*CLOSED:<socket_descriptor>
Parameters	
Response	
Example	Modem -> Host System : *ICT*CLOSED:1<0x0D><0x0A>

2.9.21 *ICT*TIMEOUT

Description	Notify that TCP connection is timeout. The Default time is 9 seconds (Retransmission time out : 3, Max connection retransmissions : 3) The previous opened socket is automatically closed.
Usage	*ICT*TIMEOUT:<socket_descriptor>
Parameters	
Response	
Example	Modem -> Host System : *ICT*TIMEOUT:0<0x0D><0x0A>

2.9.22 *ICT*REJECTED

Description	Notify that TCP connection was reset by the other end. The previous opened socket is automatically closed.
Usage	*ICT*REJECTED:<socket_descriptor>
Parameters	
Response	
Example	Modem -> Host System : *ICT*REJECTED:0<0x0D><0x0A>

2.9.23 *ICT*DATAMODE

Description	Notify that the Modem switches from Command Mode to Data Mode
Usage	*ICT*DATAMODE
Parameters	
Response	
Example	Modem -> Host System : *ICT*DATAMODE<0x0D><0x0A>

2.9.24 *ICT*P2P_DEVICE (Optional)

Description	Notify listeners about find a p2p peer device found
Usage	*ICT*P2P_DEVICE:<type> <peer_address>
Parameters	[type] 0 : p2p device found 1 : p2p device lost [peer address] P2P device address
Response	
Example	[Device Found] Modem -> Host System : *ICT*P2P_DEVICE:0 00:01:02:03:04:05<0x0D><0x0A> [Device Lost] Modem -> Host System : *ICT*P2P_DEVICE:1 00:01:02:03:04:05<0x0D><0x0A>

2.9.25 *ICT*P2P_NEG_IND(Optional)

Description	Notify listeners about P2P GO negotiation
Usage	*ICT*P2P_GO_NEG:<peer device address> <type>
Parameters	[peer address] P2P device address [type] 0 : Push Button 1 : Pin – Display 2 : Pin - Keypad
Response	
Example	[Push Button] Modem -> Host System : *ICT*P2P_GO_NEG:**02:01:00:03:04:05 0** <0x0D><0x0A> [Pin] Modem -> Host System : *ICT*P2P_GO_NEG:**02:01:00:03:04:05 1 12345678**<0x0D><0x0A> Or Modem -> Host System : *ICT*P2P_GO_NEG:**02:01:00:03:04:05 1**<0x0D><0x0A>

2.9.26 *ICT*P2P_RESULT_IND(Optional)

Description	Notify listeners about P2P result
Usage	*ICT*P2P_RST_IND:[result]
Parameters	[result] 0 : Success 1 : Failure - Reason
Response	
Example	[Success]

	Modem -> Host System : *ICT*P2P_RST_IND:0<0x0D><0x0A> [Failure] Modem -> Host System : *ICT*P2P_RST_IND:1<0x0D><0x0A>

2.9.27 *ICT*DNSRESPONSE

Description	DNS response of the host name that is to be queried
Usage	*ICT*DNSRESPONSE:<ip addr>
Parameters	Ip address
Response	
Example	Modem -> Host System : *ICT*DNSRESPONSE:210.64.10.25<0x0D><0x0A>

2.9.28 *ICT*HTTPBODY

Description	Received HTML document
Usage	*ICT*HTTPBODY:<body_len> <body>
Parameters	body_len : length of body partition body : body partition
Response	
Example	Modem -> Host System : *ICT*HTTPBODY:**1152** **<html><head>……**<0x0D><0x0A>

2.9.29 *ICT*HTTPCLOSE

Description	Closed session with HTTP server
Usage	*ICT*HTTPCLOSE:OK or *ICT*HTTPCLOSE:ERROR [error_code]
Parameters	
Response	error_code 1 - Unknown error 2 - Failed connection to HTTP server 3 - Incorrect hostname 4 - Connection is closed by remote host 5 - Connection timed out, closing 6 - Response code isn't "200 OK" 7 - Initialization Error 8 - Incorrect argument 9 - Failed memory allocation
Example	Modem -> Host System : *ICT*HTTPCLOSE:OK<0x0D><0x0A> or Modem -> Host System : *ICT*HTTPCLOSE:ERROR 2<0x0D><0x0A>

2.9.30 *ICT*HWPSIND

Description	The Status of Hardware Power Save
Usage	*ICT*HWPSIND:[Status] Refer to Chap7.2 Hardware Power Save on Standalone mode with UART Interface
Parameters	
Response	Status 0 - Operation Mode 1 - Power Save mode
Example	Modem -> Host System : *ICT*HWPSIND:0<0x0D><0x0A>

2.9.31 *ICT*EXTERNALIP

Description	Received external IP address of AP
Usage	*ICT*EXTERNALIP:<ext_ip> Refer to Chap 3.4.5 UPNP_EXTIP
Parameters	ext_ip : external IP address of AP
Response	
Example	Modem -> Host System : *ICT*EXTERNALIP:**192.168.70.50**<0x0D><0x0A>

2.9.32 *ICT*ADDPORTMAPPING

Description	Received result of requesting UPNP_ADDPORTMAP
Usage	*ICT*ADDPORTMAPPING:OK or *ICT*ADDPORTMAPPING:ERROR [error_code]
Parameters	
Response	error_code 1 - Unknown error 2 - Failed connection to UPNP server 4 - Connection is closed by UPNP server 5 - Connection timed out, closing 6 - External port number was already used 7 - Initialization Error 9 - Failed memory allocation 13 - UPNP is not supported
Example	Modem -> Host System : *ICT*ADDPORTMAPPING:OK <0x0D><0x0A>

	or Modem -> Host System : *ICT*ADDPORTMAPPING:ERROR 6<0x0D><0x0A>

2.9.33 *ICT*DELPORTMAPPING

Description	Received result of requesting UPNP_DELPORTMAP
Usage	*ICT*DELPORTMAPPING:OK or *ICT*DELPORTMAPPING:ERROR
Parameters	
Response	
Example	Modem -> Host System : *ICT*DELPORTMAPPING:OK <0x0D><0x0A> or Modem -> Host System : *ICT*DELPORTMAPPING:ERROR <0x0D><0x0A>

2.9.34 *ICT*PINGREPLY

Description	Notify that the Modem gives the ping reply msg to the Host System
Usage	*ICT*PINGREPLY: <Target IP> <Data size> <Response Time>
Parameters	Target IP : Ip address of response STA Data size : response data size Response Time : response time from sending ping request till response receiving
Response	
Example	Modem -> Host System : *ICT*PINGREPLY:**192.168.1.1** 1460 **131**<0x0D><0x0A>

2.9.35 *ICT*DDNSEXTERNALIP

Description	Received external IP address of
Usage	*ICT*DDNSEXTERNALIP:<ddns_ext_ip>
Parameters	ddns_ext_ip : External public IP address of AP
Response	
Example	Modem -> Host System : *ICT*DDNSEXTERNALIP:**210.107.64.131**<0x0D><0x0A>

2.9.36 *ICT*DDNSUPDATE

Description	Notify that the Modem gives the DDNS update result msg to the host
Usage	*ICT*DDNSUPDATE: <Result> <External IP>
Parameters	Result: result of DDNS update

<table>
<tr><td></td><td>External IP: Updated IP address for DDNS server host</td></tr>
<tr><td>Response</td><td>Error code
<table>
<tr><td>result</td><td>SUCCESS/FAIL</td><td>Means</td></tr>
<tr><td>0</td><td>SUCCESS</td><td>good: Completed Update</td></tr>
<tr><td>1</td><td>SUCCESS</td><td>nochg: Although update is completed, the IP address is not changed</td></tr>
<tr><td>2</td><td>FAIL</td><td>Nohost: The hostname does not exist in your account</td></tr>
<tr><td>3</td><td>FAIL</td><td>Badauth: Wrong username or password</td></tr>
<tr><td>4</td><td>FAIL</td><td>Badagent: Agent sent incorrect Http request format</td></tr>
<tr><td>5</td><td>FAIL</td><td>!donator: You have specific options which require a service fee</td></tr>
<tr><td>6</td><td>FAIL</td><td>Abuse: The hostname is blocked due to abuse</td></tr>
<tr><td>7</td><td>FAIL</td><td>911: Service Maintenance is in progress. Please contact with DynDns Support</td></tr>
<tr><td>8</td><td>FAIL</td><td>Unknown Error</td></tr>
</table></td></tr>
<tr><td>Example</td><td>Modem -> Host System : *ICT*DDNSUPDATE:0 210.107.63.131<0x0D><0x0A></td></tr>
</table>

2.9.37 *ICT*OTA_VERSION

Description	Received result of requesting OTA_VERCHECK
Usage	*ICT*VERCHECK:[local] [server]
Parameters	
Response	local - local firmware version server - firmware version at OTA server (0 is error)
Example	Modem -> Host System : *ICT*OTA_VERSION:2647 0 <0x0D><0x0A> or Modem -> Host System : *ICT*OTA_VERSION:2647 2685 <0x0D><0x0A>

2.9.38 *ICT*OTA_UPDATE

Description	Received result of requesting OTA_REQUEST
Usage	*ICT*OTA_UPDATE:OK or

	*ICT*OTA_UPDATE:ERROR [error_code]
Parameters	
Response	error_code 1 - signature error 2 - DRAM CRC error 3 - IRAM CRC error 4 - SF_CODE CRC error 5 - DRAM SIZE error 6 - IRAM SIZE error 7 - SF_CODE SIZE error 9 - OTA server error
Example	Modem -> Host System : *ICT*OTA_UPDATE:OK <0x0D><0x0A> or Modem -> Host System : *ICT*OTA_UPDATE:ERROR 1<0x0D><0x0A>

2.9.39 *ICT*SNTP_RESPONSE

Description	Received result of requesting SNTP_GET
Usage	*ICT*SNTP_RESPONSE:[length] [day name] [month name] [day] [time] [year]
Parameters	
Response	length: length of SNTP response message day name: "Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat" month name: "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec" day: hh:mm:ss year
Example	Modem -> Host System : *ICT*SNTP_RESPONSE:24 Thu Oct 22 11:45:48 2015<0x0D><0x0A> or Modem -> Host System : *ICT*SNTP_RESPONSE:TIMEOUT<0x0D><0x0A>

2.9.40 *ICT*STA_ASSOCIATED

Description	Indication of associated station
Usage	*ICT*STA_ASSOCIATED:[MAC ADDRESS] [RSSI]
Parameters	
Response	MAC ADDRESS: MAC address of Station format: xx:xx:xx:xx:xx:xx RSSI: signal strength of associated station
Example	Modem -> Host System : *ICT*STA_ASSOCIATED:DC:A9:71:37:53:A2 -44 <0x0D><0x0A>

2.10 Error Codes

Error Code	Description
1	ERR_SOCKET_NOT_AVAIL
2	ERR_SOCKET_INVALID
3	ERR_SOCKET_NOT_EXIST
4	ERR_CONNECTION_ESTABLISHMENT
5	ERR_WIFI_CONFIG_PARAM_INVALID
6	ERR_TCPIP_PARAM_INVALID
7	ERR_OUT_OF_MEMORY
8	ERR_GENERAL_PARAM_INVALID
9	ERR_COMMAND_NOT_EXIST
10	ERR_ADDRESS_IN_USE

3 Example Sequence for AT Commands

3.1 Infrastructure Mode

3.1.1 Associate to an AP being set with OPEN (NON ENCRYPTION)

1. Enable DHCP : *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Configure the Encryption : *AT*ICT*CRYPTO=<key_mgmt> <pairwise_cipher> <group_cipher>*
   - **AT*ICT*CRYPTO=0 0 0**
3. Associate with the AP having the specified SSID : *AT*ICT*ASSOCIATE=<ssid> [channel]*
   - **AT*ICT*ASSOCIATE=Standalone**

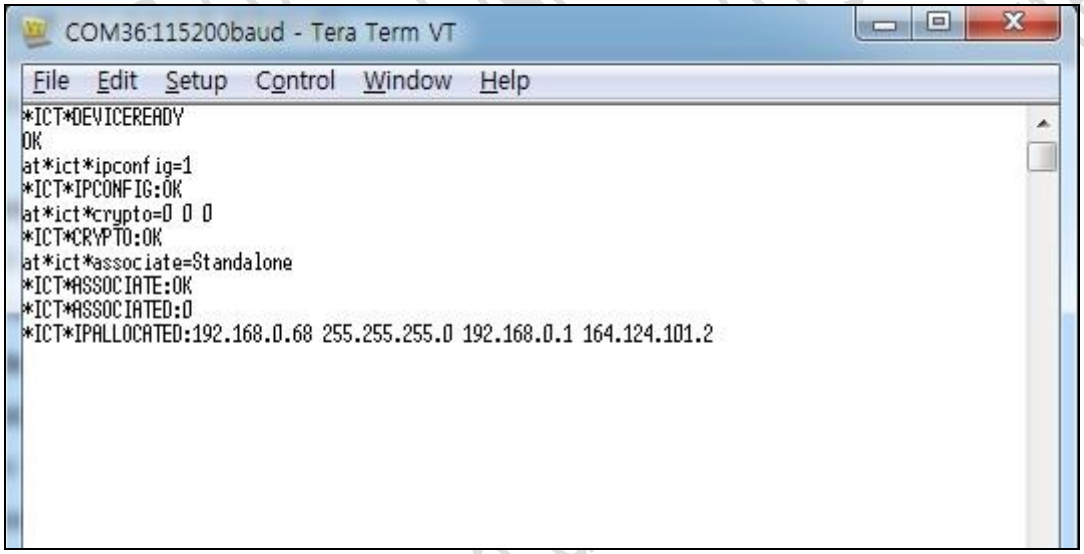


Figure 3. OPEN CONNECT

3.1.2 Associate to an AP being set with WEP

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Configure the Encryption : *AT*ICT*CRYPTO=<key_mgmt> <pairwise_cipher> <group_cipher>*
   - **AT*ICT*CRYPTO=1 0 0**
3. Configure the WEP key : *AT*ICT*WEP=<key_index> <key>*
   - **AT*ICT*WEP=1 12345**
4. Associate with the AP having the specified SSID : *AT*ICT*ASSOCIATE=<ssid> [channel]*
   - **AT*ICT*ASSOCIATE=Standalone**

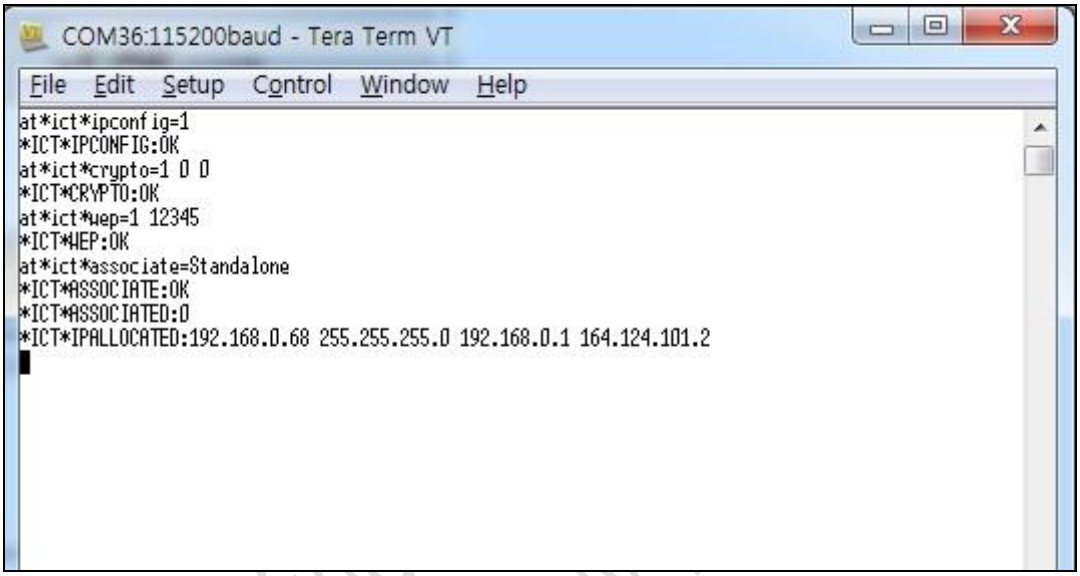


Figure 4. WEP CONNECT

3.1.3 Associate to an AP being set with WPA-PSK with TKIP

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Configure the Encryption : *AT*ICT*CRYPTO=<key_mgmt> <pairwise_cipher> <group_cipher>*
   - **AT*ICT*CRYPTO=2 0 0**
3. Configure the PSK : *AT*ICT*PSK=<passphrase>*
   - **AT*ICT*PSK=12345678**
4. Associate with the AP having the specified SSID : *AT*ICT*ASSOCIATE=<ssid> [channel]*
   - **AT*ICT*ASSOCIATE=Standalone**

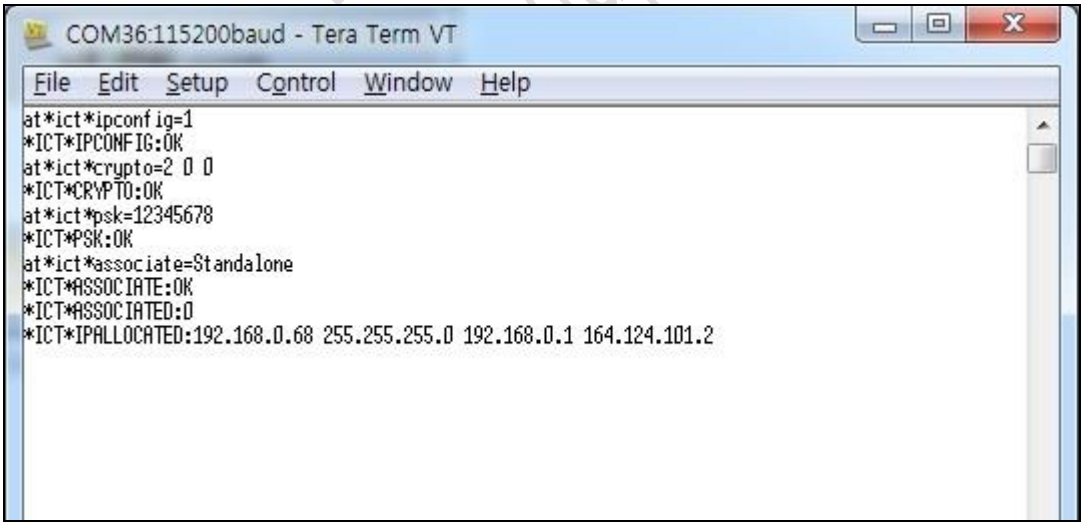


Figure 5. WPA-PSK with TKIP CONNECT

3.1.4 Associate to an AP being set with WPA2-PSK with CCMP

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Configure the Encryption : *AT*ICT*CRYPTO=<key_mgmt> <pairwise_cipher> <group_cipher>*
   - **AT*ICT*CRYPTO=3 1 1**
3. Configure the PSK : *AT*ICT*PSK=<passphrase>*
   - **AT*ICT*PSK=12345678**
4. Associate with the AP having the specified SSID : *AT*ICT*ASSOCIATE=<ssid> [channel]*
   - **AT*ICT*ASSOCIATE=Standalone**

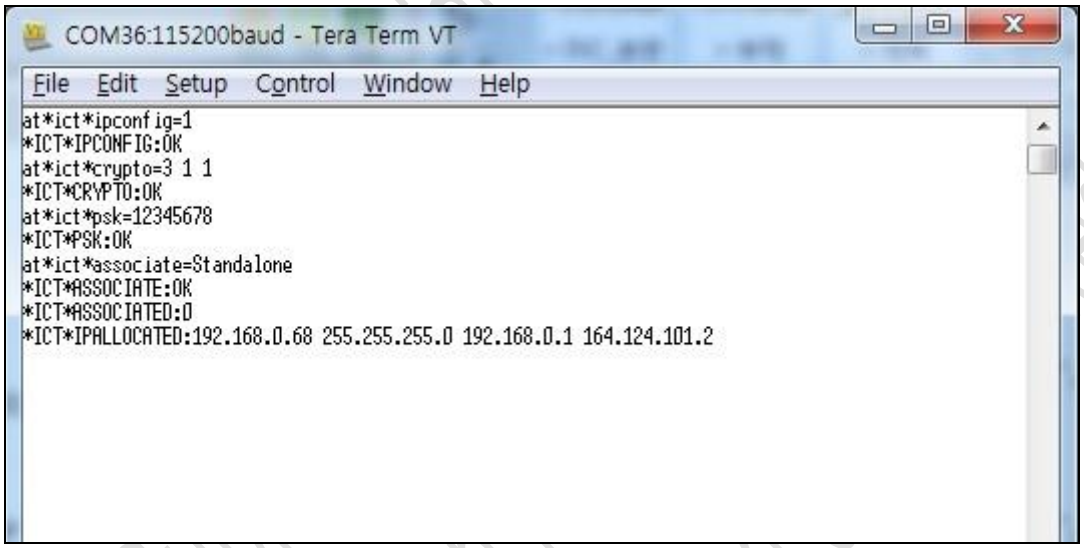


Figure 6. WPA2-PSK with CCMP CONNECT

3.1.5 Associate to an AP using simple connection method

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*
   - **AT*ICT*SCONN=Standalone 12345678**
3. Get current Wireless and Network status : *AT*ICT* NWSTATUS =?*
   - **AT*ICT*NWSTATUS=?**

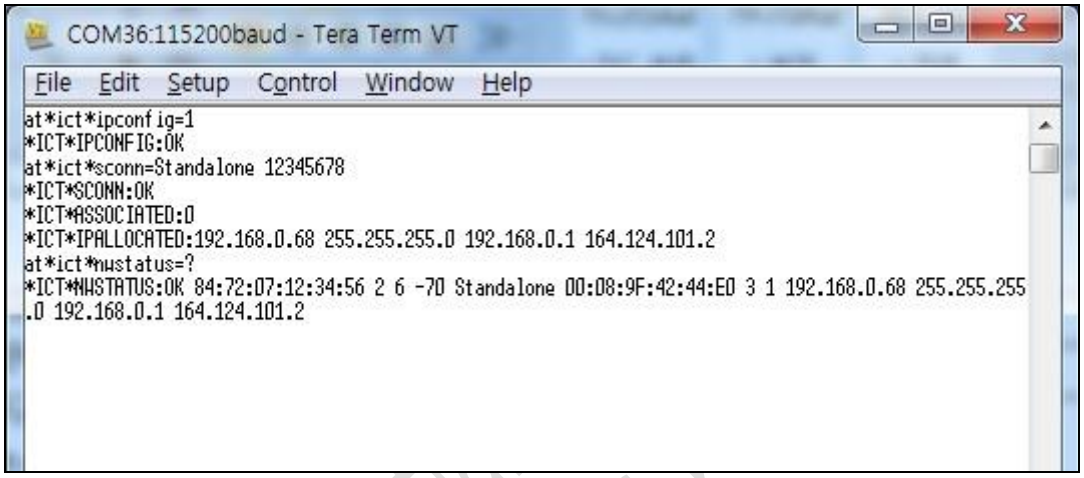
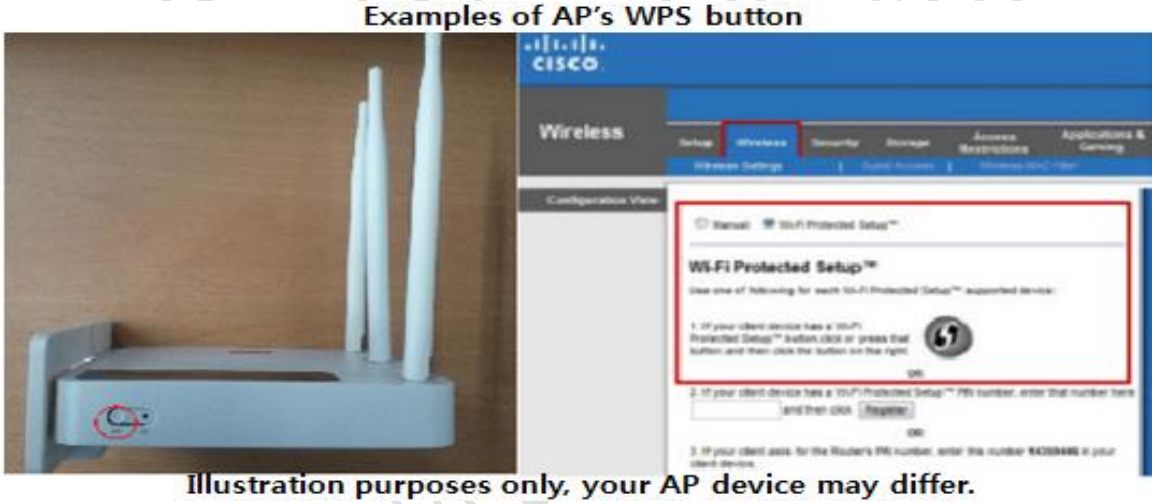


Figure 7. Simple Connect

3.1.6 Associate to an AP using WPS_PBC (PUSH)

1. Enable DHCP : *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Configure the AP's WPS PUSH BUTTON:



3. Associate with the AP having specified WPS_PBC: *AT*ICT*WPS_PBC=[any | bssid]*
   - **AT*ICT*WPS_PBC=any**

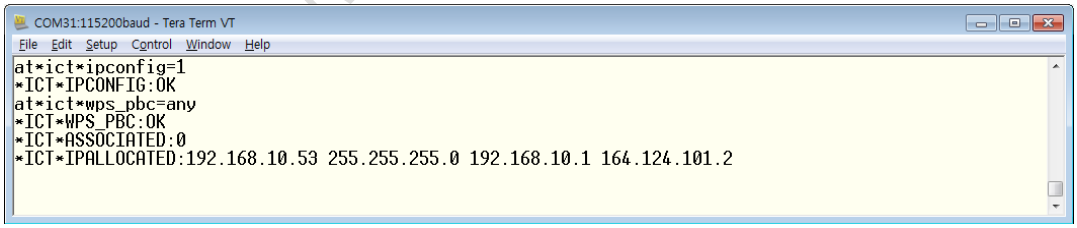


Figure 8. WPS PUSH BUTTON CONNECT

3.2 IBSS Mode (Optional)

3.2.1 Create or Associate to an ADHOC(IBSS) being set with OPEN (NON ENCRYPTION)

1. Disable DHCP and input static IP: *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=0 192.168.0.10 255.255.255.0 192.168.0.254 0.0.0.0**
2. Associate with the IBSS having the specified SSID : *AT*ICT*ADSTART=<ssid> <channel> [key_mgmt] [passphrase]*
   - **AT*ICT*ADSTART=ADHOC_STDA 11**
3. Get current Wireless and Network status : *AT*ICT* NWSTATUS =?*
   - **AT*ICT*NWSTATUS=?**

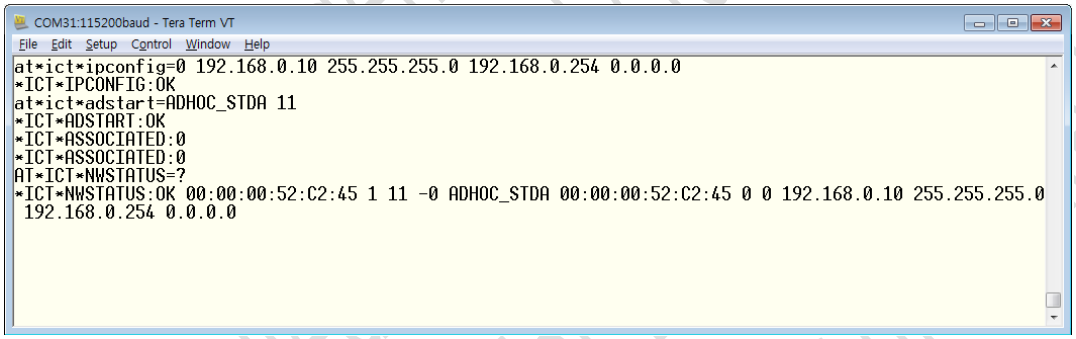


Figure 9. OPEN with ADHOC mode

3.2.2 Create or Associate to an ADHOC(IBSS) being set with WEP

1. Disable DHCP and input static IP: *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=0 192.168.0.10 255.255.255.0 192.168.0.254 0.0.0.0**
2. Associate with the IBSS having the specified SSID : *AT*ICT*ADSTART=<ssid> <channel> [key_mgmt] [passphrase]*
   - **AT*ICT*ADSTART=ADHOC_STDA 11 1 1234567890**
3. Get current Wireless and Network status : *AT*ICT* NWSTATUS =?*
   - **AT*ICT*NWSTATUS=?**

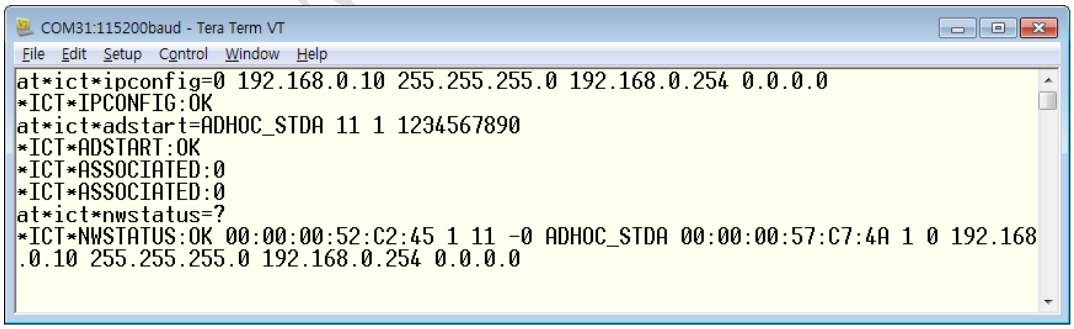


Figure 10. WEP with ADHOC mode

3.3 SoftAP Mode

3.3.1 Create an AP being set with OPEN (NON ENCRYPTION)

1. Enable DHCP : *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - ▶ **AT*ICT*IPCONFIG=1**
2. Create and AP having the specified SSID: *AT*ICT*APSTART=<essid> <channel> [key_mgmt] [pairwise_cipher] [group_cipher] [passphrase]*
   - ▶ **AT*ICT*APSTART=Standalone 1**
3. Get current Wireless and Network status : *AT*ICT* NWSTATUS =?*
   - ▶ **AT*ICT*NWSTATUS=?**

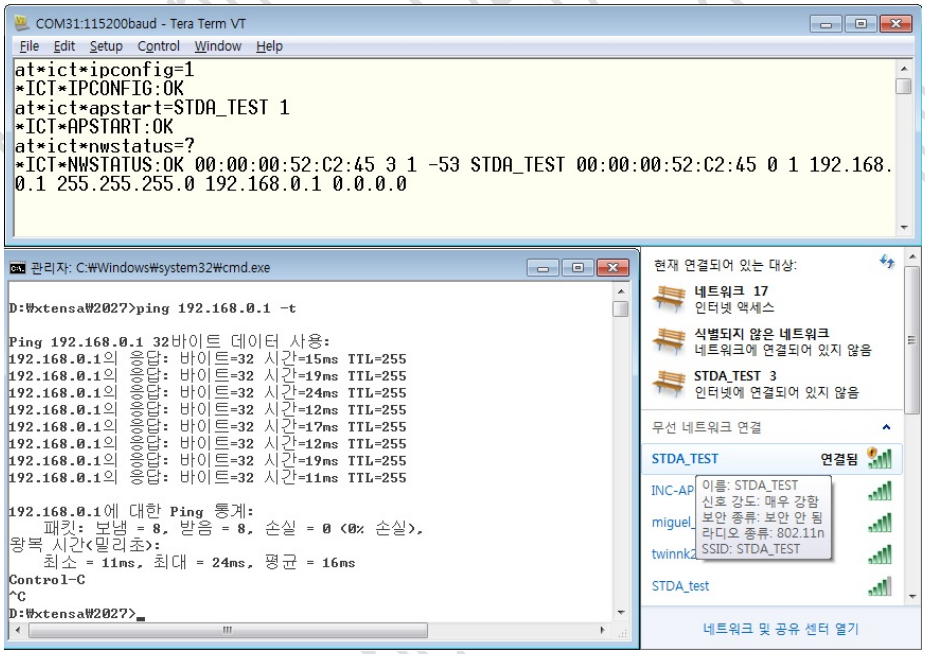


Figure 11. OPEN with AP mode

3.3.2 Create an AP being set with WPA-PSK with TKIP

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - ▶ **AT*ICT*IPCONFIG=1**
2. Associate with the AP having the specified SSID with WPA-PSK(TKIP) encryption:
   *AT*ICT*APSTART=<essid> <channel> [key_mgmt] [pairwise_cipher] [group_cipher] [passphrase]*
   - ▶ **AT*ICT*APSTART=SDTA_TEST 1 2 0 0 12345678**
3. Get current Wireless and Network status: *AT*ICT* NWSTATUS =?*
   - ▶ **AT*ICT*NWSTATUS=?**

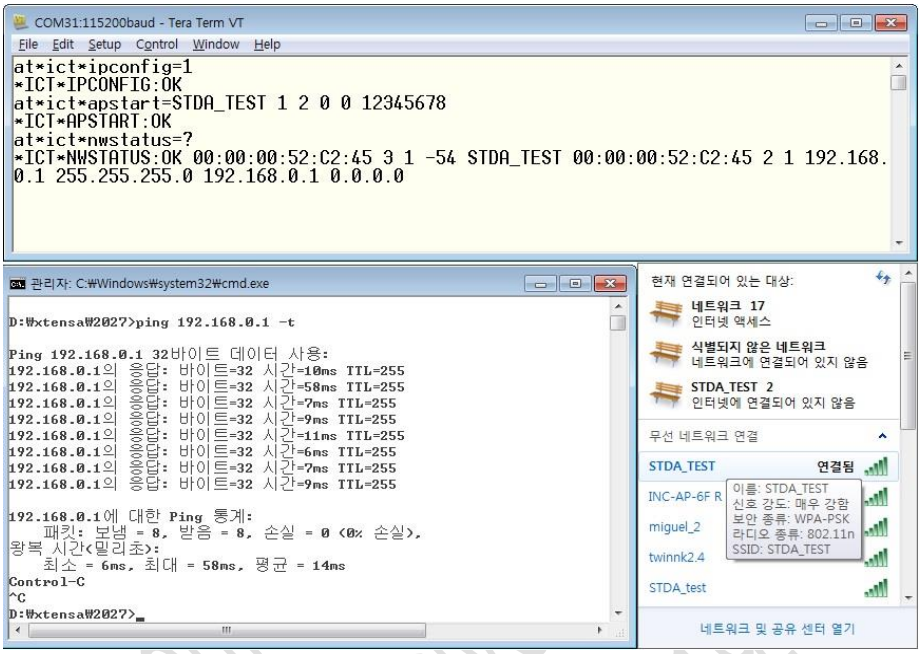


Figure 12. WPA-PSK - TKIP with AP mode

3.3.3 Create an AP being set with WPA2-PSK with CCMP

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*

   ▶ **AT*ICT*IPCONFIG=1**

2. Associate with the AP having the specified SSID with WPA2-PSK(CCMP) encryption:

   *AT*ICT*APSTART=<essid> <channel> [key_mgmt] [pairwise_cipher] [group_cipher] [passphrase]*

   ▶ **AT*ICT*APSTART= SDTA_TEST 1 3 1 1 12345678**

3. Get current Wireless and Network status: *AT*ICT* NWSTATUS =?*

   ▶ **AT*ICT*NWSTATUS=?**

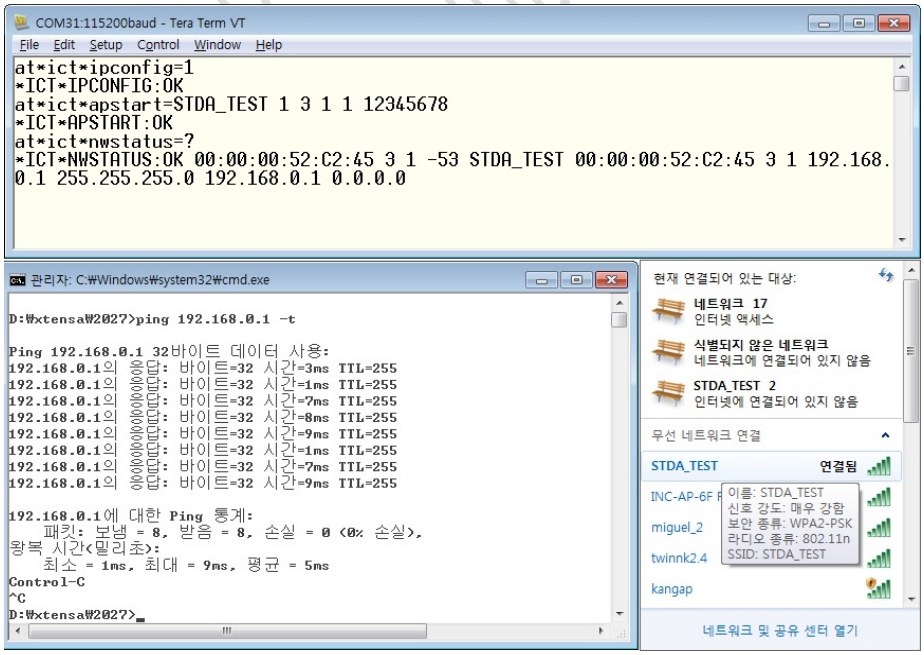
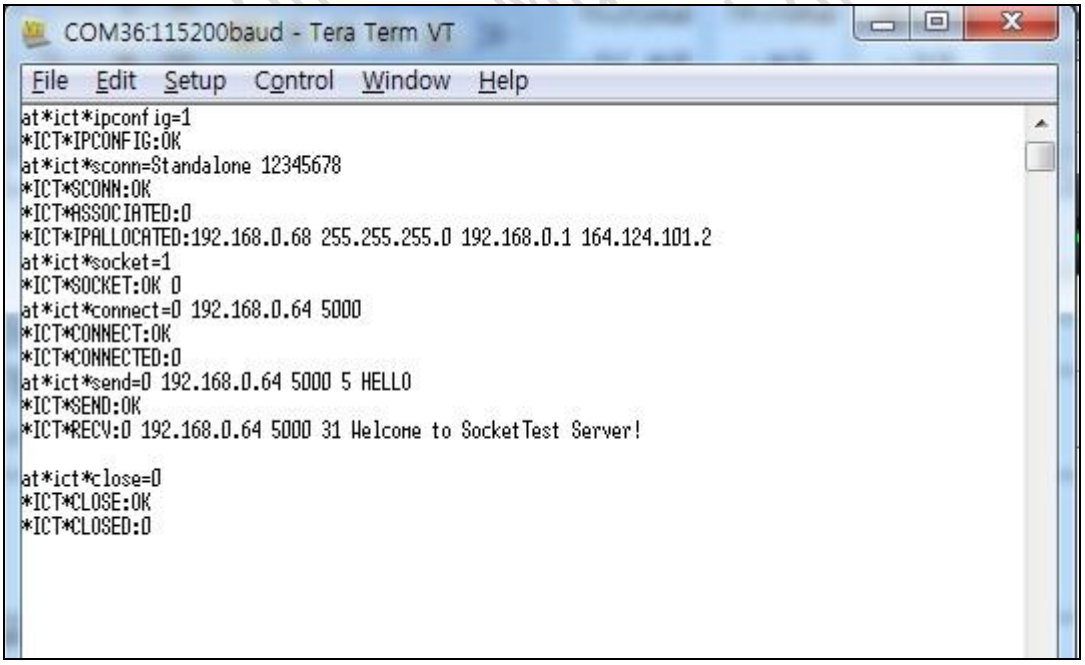


Figure 13. WPA2-PSK - CCMP with AP mode

3.4 TCP & UDP socket

3.4.1 TCP Client

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*
   - **AT*ICT*SCONN=Standalone 12345678**
3. Open a Socket - TCP socket : *AT*ICT*SOCKET=<socket_type>*
   - **AT*ICT*SOCKET=1**
4. Connect the TCP socket on the remote IP address and the specified port :
   *AT*ICT*CONNECT=<socket_descriptor> <ip_addr> <rport>*
   - **AT*ICT*CONNECT=0 192.168.0.64 5000**
5. Transmit a byte stream to a socket : *AT*ICT*SEND=<socket_descriptor> <ip_addr> <rport> <size> <stream_data>*
   - **AT*ICT*SEND=0 192.168.0.64 5000 5 HELLO**
6. Close a socket : *AT*ICT*CLOSE=<socket_descriptor>*
   - **AT*ICT*CLOSE=0**



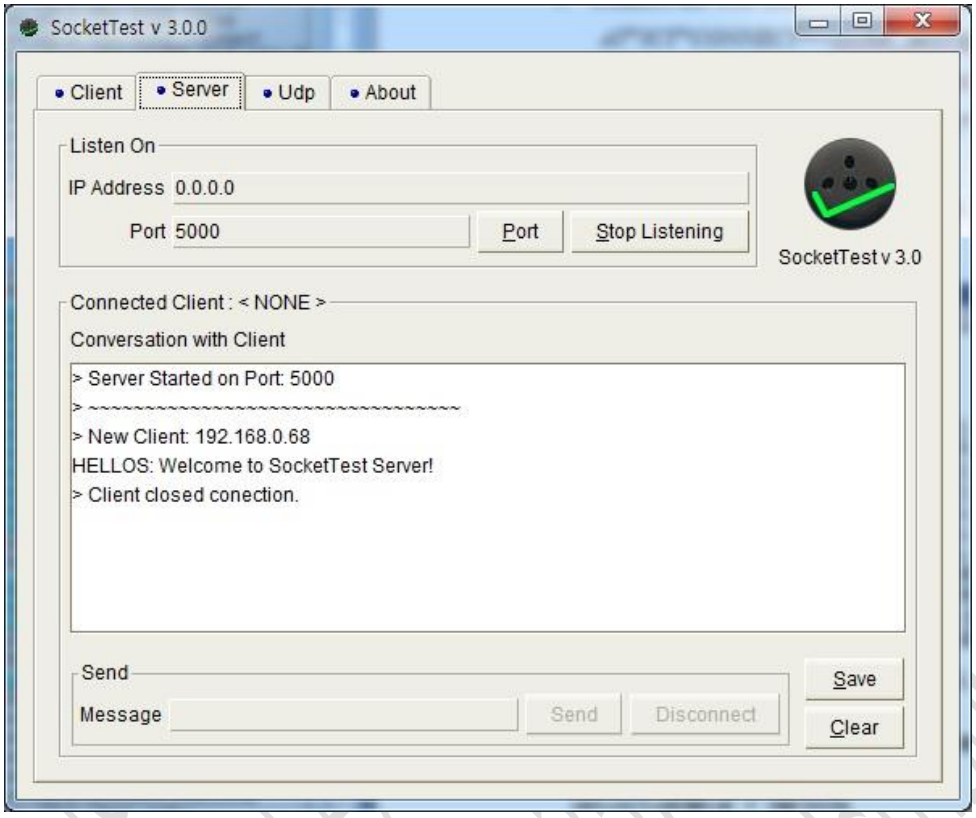
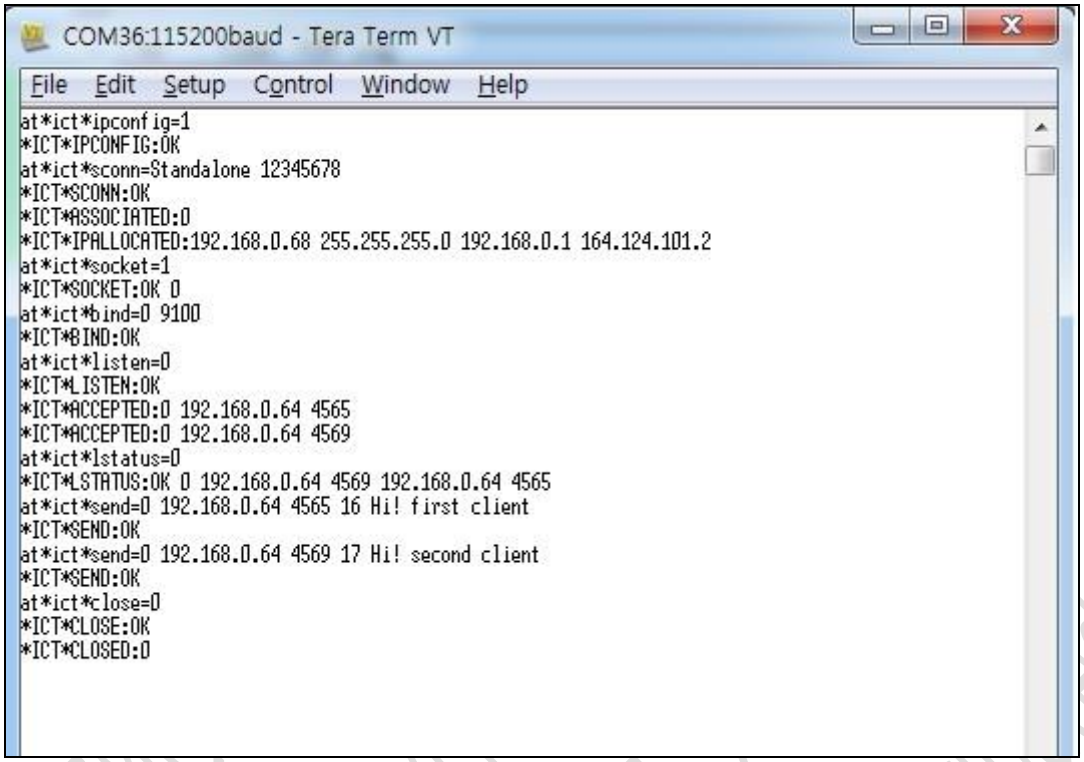


Figure 14. TCP Client

3.4.2 TCP Server

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*
   - **AT*ICT*SCONN=Standalone 12345678**
3. Open a Socket - TCP socket : *AT*ICT*SOCKET=<socket_type>*
   - **AT*ICT*SOCKET=1**
4. Bind the socket to the specified port : *AT*ICT*BIND=<socket_descriptor> <lport>*
   - **AT*ICT*BIND=0 9100**
5. Listen TCP socket on the local IP address and the specified port : *AT*ICT*LISTEN=<socket_desccriptor>*
   - **AT*ICT*LISTEN=0**
6. Retrieve active connection established with a listening TCP socket : *AT*ICT*LSTATUS=<socket_desccriptor>*
   - **AT*ICT*LSTATUS=0**



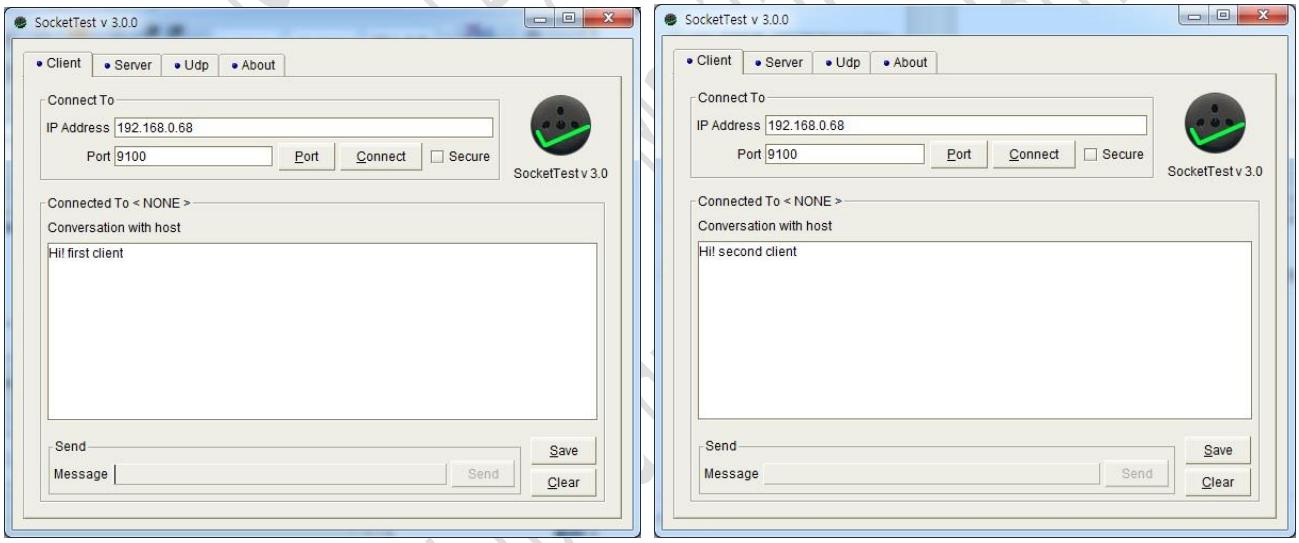


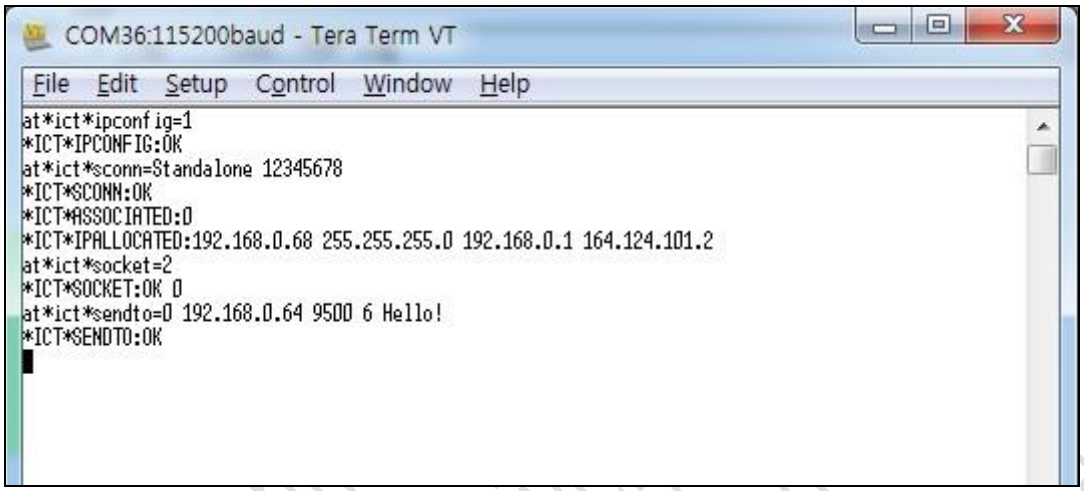
Figure 15. TCP Server

3.4.3 UDP Client

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - ▶ **AT*ICT*IPCONFIG=1**
2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*
   - ▶ **AT*ICT*SCONN=Standalone 12345678**
3. Open a Socket - UDP socket : *AT*ICT*SOCKET=<socket_type>*
   - ▶ **AT*ICT*SOCKET=2**
4. Send a UDP datagram data to a socket on the remote IP address and the specified port :

AT*ICT*SENDTO=<socket_descriptor> <ip_addr> <rport> <size> <datragram_data>

▶ **AT*ICT*SENDTO=0 192.168.0 64 9500 6 Hello!**



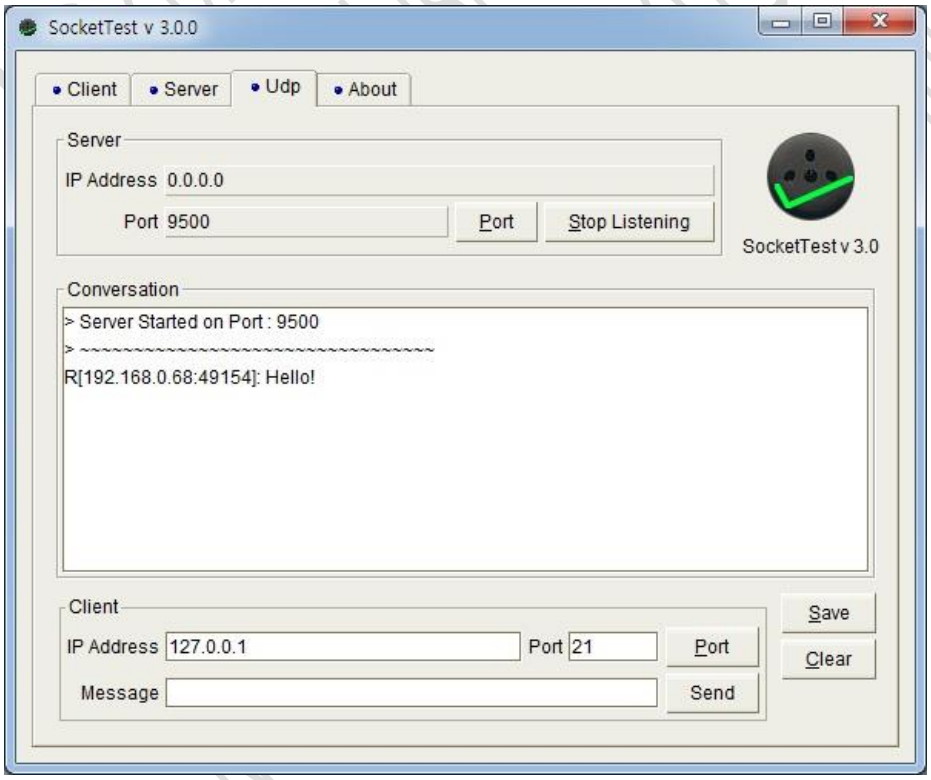


Figure 16. UDP Client

3.4.4 UDP Server

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*

   ▶ **AT*ICT*IPCONFIG=1**

2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*

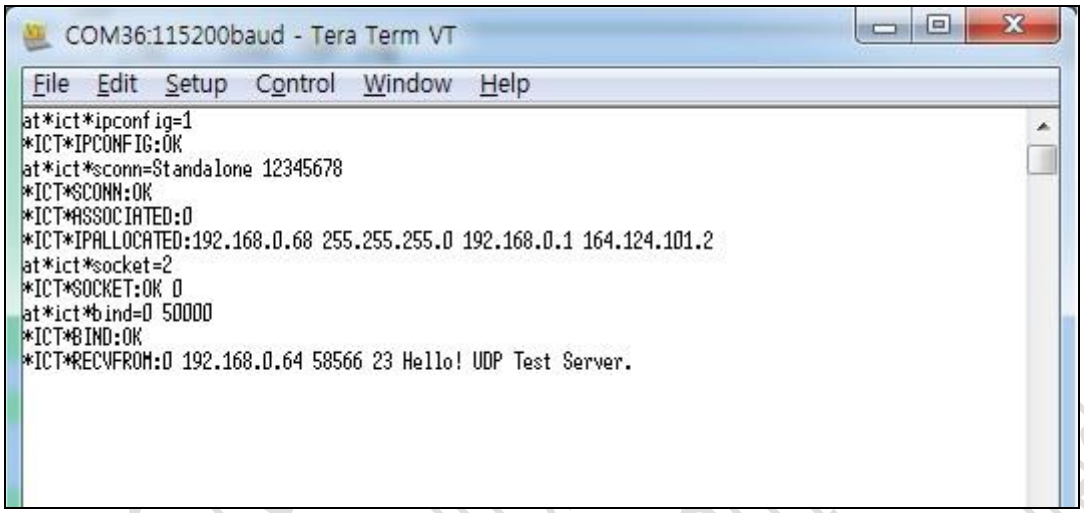
   ▶ **AT*ICT*SCONN=Standalone 12345678**

3. Open a Socket - UDP socket : *AT*ICT*SOCKET=<socket_type>*

▶ **AT*ICT*SOCKET=2**

4. Bind the socket to the specified port : *AT*ICT*BIND=<socket_descriptor> <lport>*

▶ **AT*ICT*BIND=0 50000**



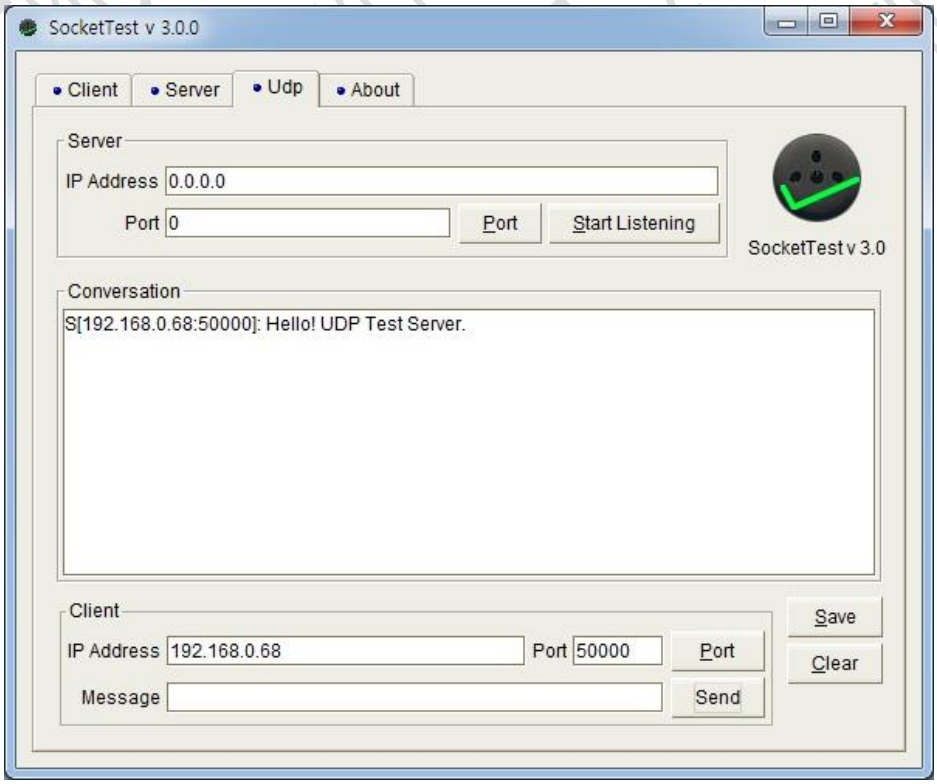


Figure 17. UDP Server

3.4.5 UPNP_EXTIP

1. Enable DHCP : *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*

▶ **AT*ICT*IPCONFIG=1**

2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*

▶ **AT*ICT*SCONN=Standalone 12345678**

3. Get external IP address of AP : *AT*ICT*UPNP_EXTIP*

▶ **AT*ICT*UPNP_EXTIP**

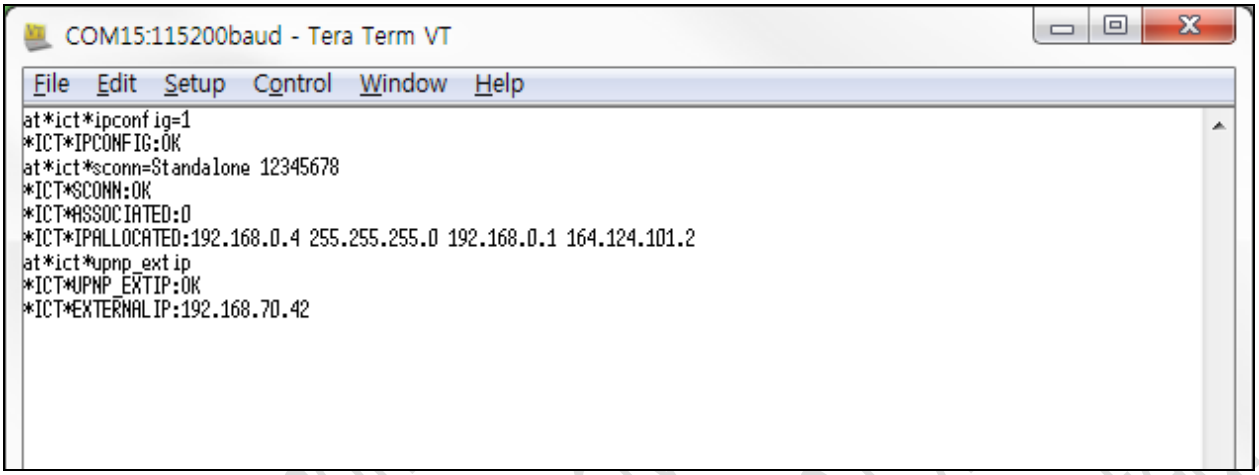


Figure 18. UPNP_EXTIP

3.4.6 UPNP_ADDPORTMAP

1. Enable DHCP : *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*

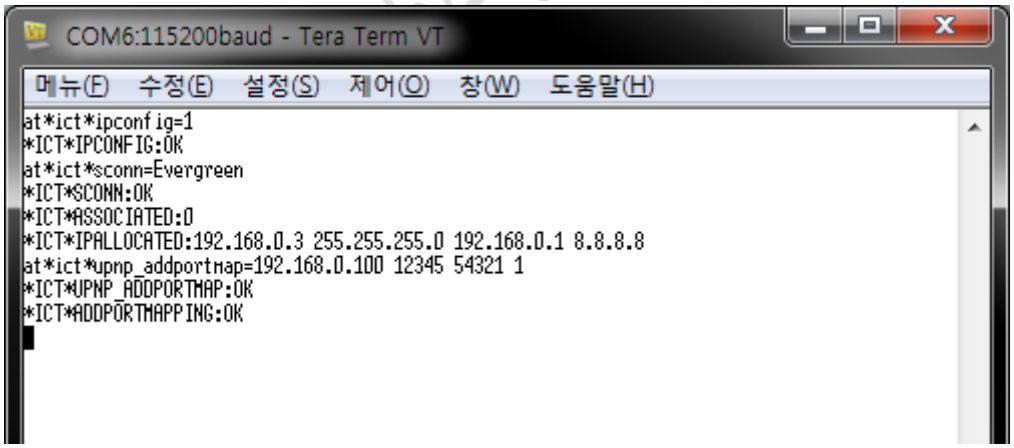
▶ **AT*ICT*IPCONFIG=1**

2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*

▶ **AT*ICT*SCONN=Standalone 12345678**

3. Add portmapping in AP using UPNP : *AT*ICT*UPNP_ADDPORTMAP=<ip> <port_int> <port_ext> <protocol> [description]*

▶ **AT*ICT*UPNP_ADDPORTMAP=192.168.0.100 12345 54321 1**



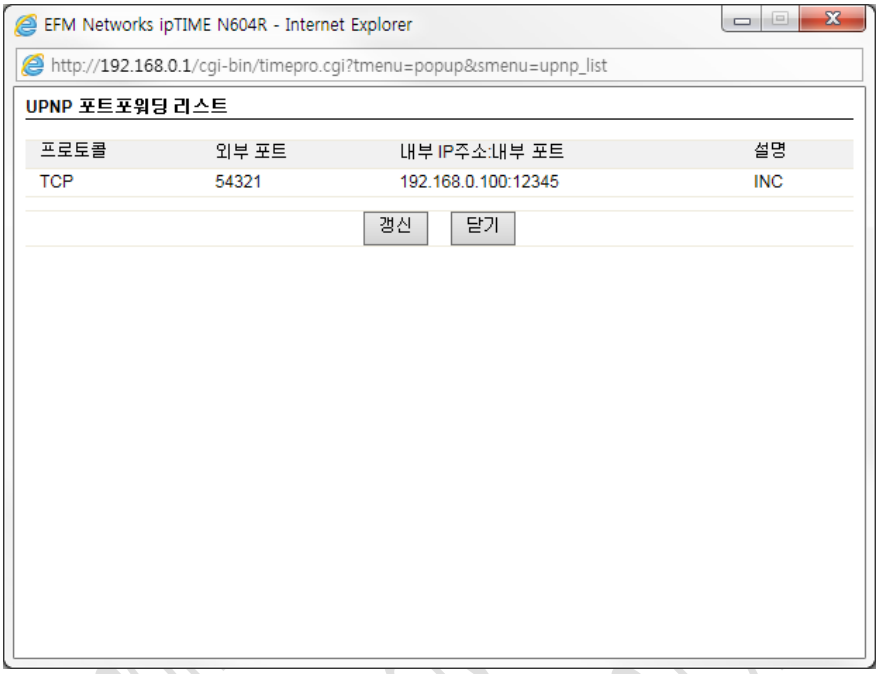


Figure 19. UPNP_ADDPORTMAP

3.4.7 UPNP_DELPORTMAP

1. Enable DHCP : *AT*ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*
   - **AT*ICT*SCONN=Standalone 12345678**
3. Delete portmapping in AP using UPNP : *AT*ICT*UPNP_DELPORTMAP=<port_ext> <protocol>*
   - **AT*ICT*UPNP_DELPORTMAP=54321 1**

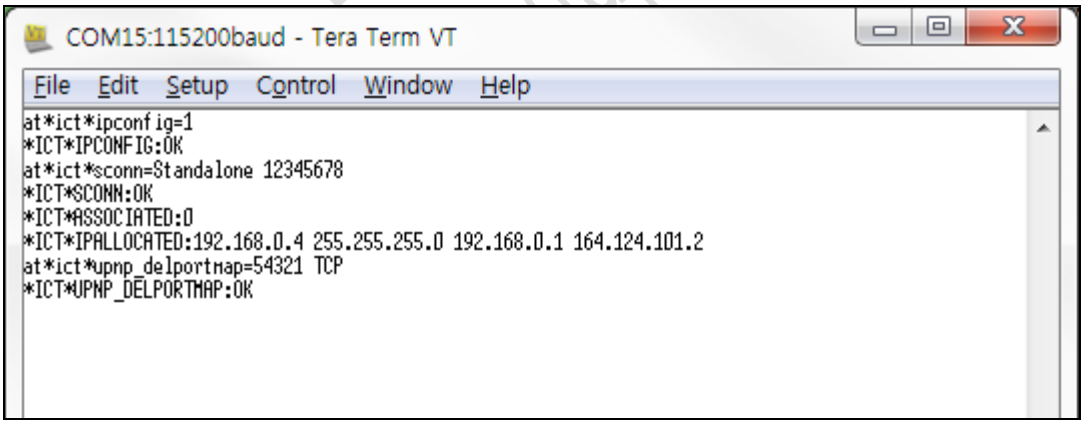


Figure 20. UDP Server

3.5 TCP & SSL

3.5.1 TCP SSL Client

1. Enable DHCP : **ICT*IPCONFIG=<dhcp mode> [ip] [subnet] [gateway] [dns]*
   - **AT*ICT*IPCONFIG=1**
2. Connect to an AP simply without setting security: *AT*ICT*SCONN=<essid> [passphrase]*
   - **AT*ICT*SCONN=Standalone 12345678**
3. Connect the TCP socket on the remote IP address and the specified port :
   *AT*ICT*SSL_CONNECT=<socket_descriptor> <ip_addr> <rport>*
   - **AT*ICT*SSL_CONNECT=0 192.168.0.64 5000**
4. Transmit a byte stream to a socket : *AT*ICT*SSL_SEND=<socket_descriptor> <size> <stream_data>*
   - **AT*ICT*SSL_SEND=0 5 HELLO**
5. Received a byte stream from a socket:**ICT*SSL_RECV=<socket_descriptor><size><stream_data>*
   - ***ICT*SSL_RECV=0 7 GOODBYE**
6. Close a socket : *AT*ICT*SSL_CLOSE=<socket_descriptor>*
   - **AT*ICT*SSL_CLOSE=0**

3.5.2 TCP SSL Sequence Example

1. SSL Server Connect
   - **AT*ICT*SSL_CONNECT=0 192.168.0.64 5000**
   - ***ICT*SSL_CONNECT:OK**
   - ***ICT*SSL_IND:0 0 OK**
2. Message Send
   - **AT*ICT*SSL_SEND=0 5 HELLO**
   - ***ICT*SSL_SEND:OK**
   - ***ICT*SSL_IND:0 1 OK**
3. Message Receive
   - ***ICT*SSL_RECV:0 7 GOODBYE**

3.5.3 TCP SSL Server

1. Start TCP SSL Server at the specified port : *AT*ICT*SSL_SVR_START=<socket_descriptor> <port>*
   - **AT*ICT*SSL_SVR_START=0 5000**
2. Accept client connection: *ICT*SSL_SVR_ACCEPTED=<socket_descriptor><ip address> <port>*
   - ***ICT*SSL_SVR_ACCEPTED=1**
3. Received a byte stream from a socket:**ICT*SSL_SVR_RECV=<socket_descriptor><size><stream_data>*

- ***ICT*SSL_SVR_RECV=1 5 HELLO**

4. Transmit a byte stream to a socket : *AT*ICT*SSL_SVR_SEND=<socket_descriptor> <size> <stream_data>*
   - **AT*ICT*SSL_SVR_SEND=1 7 GOODBYE**
5. Received a close event from a socket:**ICT*SSL_SVR_CLOSED=<socket_descriptor>*
   - ***ICT*SSL_SVR_CLOSED=1**
6. Close TCP SSL server: *AT*ICT*SSL_SVR_CLOSE=<socket_descriptor>*
   - **AT*ICT*SSL_SVR_CLOSE=0**

3.5.4 TCP SSL Server Sequence Example

1. SSL Server Start
   - **AT*ICT*SSL_SVR_START=0 5000**
   - ***ICT*SSL_SVR_START:OK**
2. SSL Client#1 Connect
   - ***ICT*SSL_SVR_ACCEPTED:1 192.168.0.100 34567**
3. SSL Client#2 Connect
   - ***ICT*SSL_SVR_ACCEPTED:2 192.168.0.101 45678**
4. Message Send to SSL Client #2
   - **AT*ICT*SSL_SVR_SEND=2 5 HELLO**
   - ***ICT*SSL_SVR_SEND:OK**
   - ***ICT*SSL_SVR_IND:2 1 OK**
5. Message Receive from SSL Client #1
   - ***ICT*SSL_SVR_RECV:1 7 GOODBYE**
6. SSL Client#1 Close
   - ***ICT*SSL_SVR_CLOSED:1**
7. SSL Server Stop
   - **AT*ICT*SSL_SVR_CLOSE=0**
   - ***ICT*SSL_SVR_CLOSE:OK**
   - ***ICT*SSL_SVR_CLOSED:2 (Client #2)**
   - ***ICT*SSL_SVR_CLOSED:0 (SSL Server)**

4 WebUI Screen Shots for Standalone Mode

4.1 WebUI STA Mode Screen shots

4.1.1 System

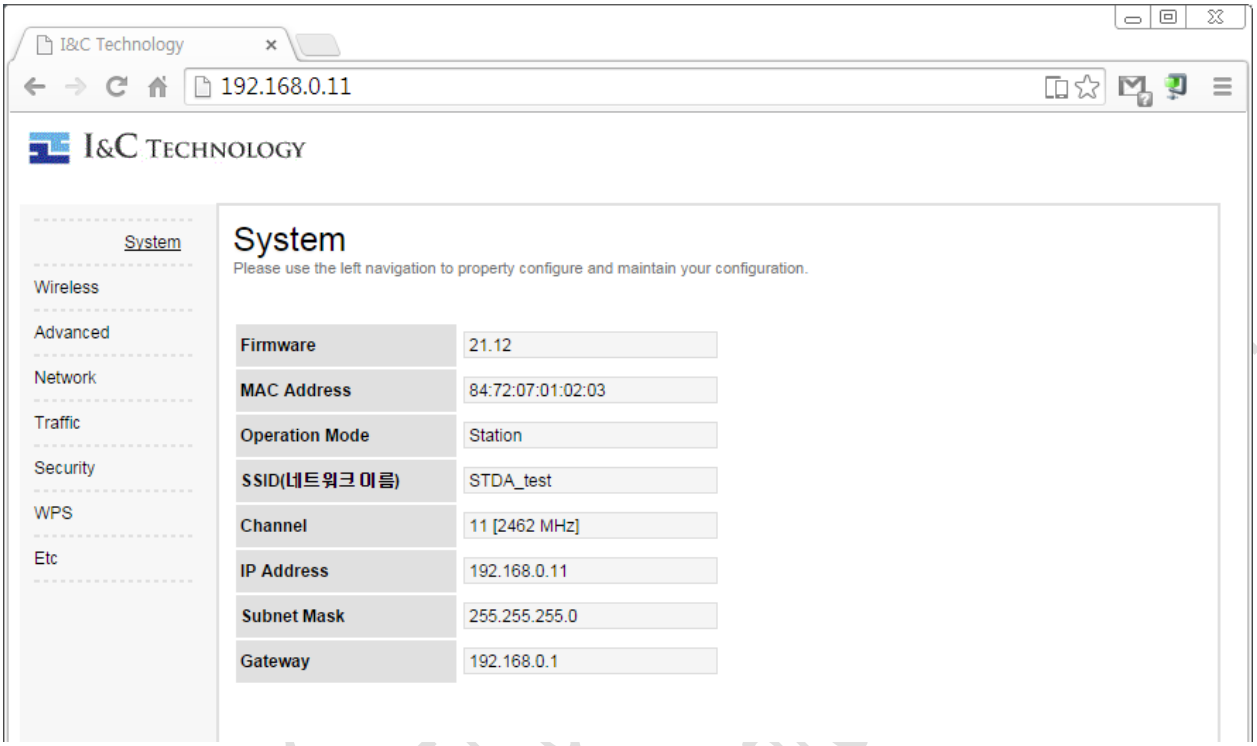


Figure 21. STA Mode System Menu for WebUI

4.1.2 Wireless

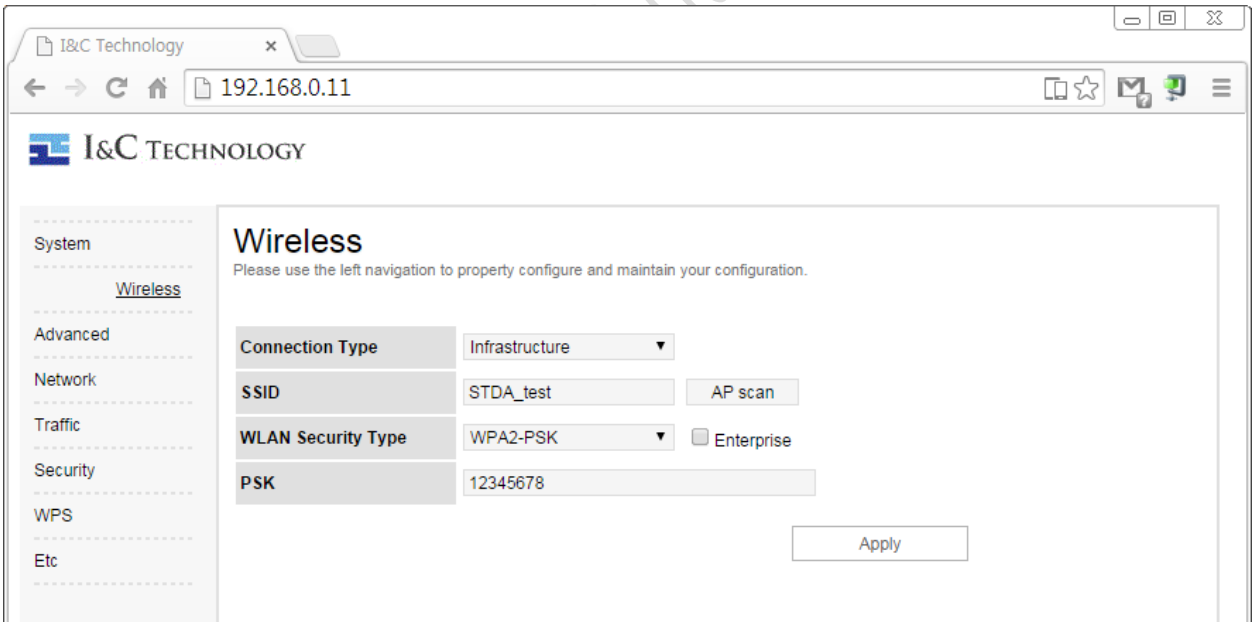


Figure 22. STA Mode Wireless Menu for WebUI

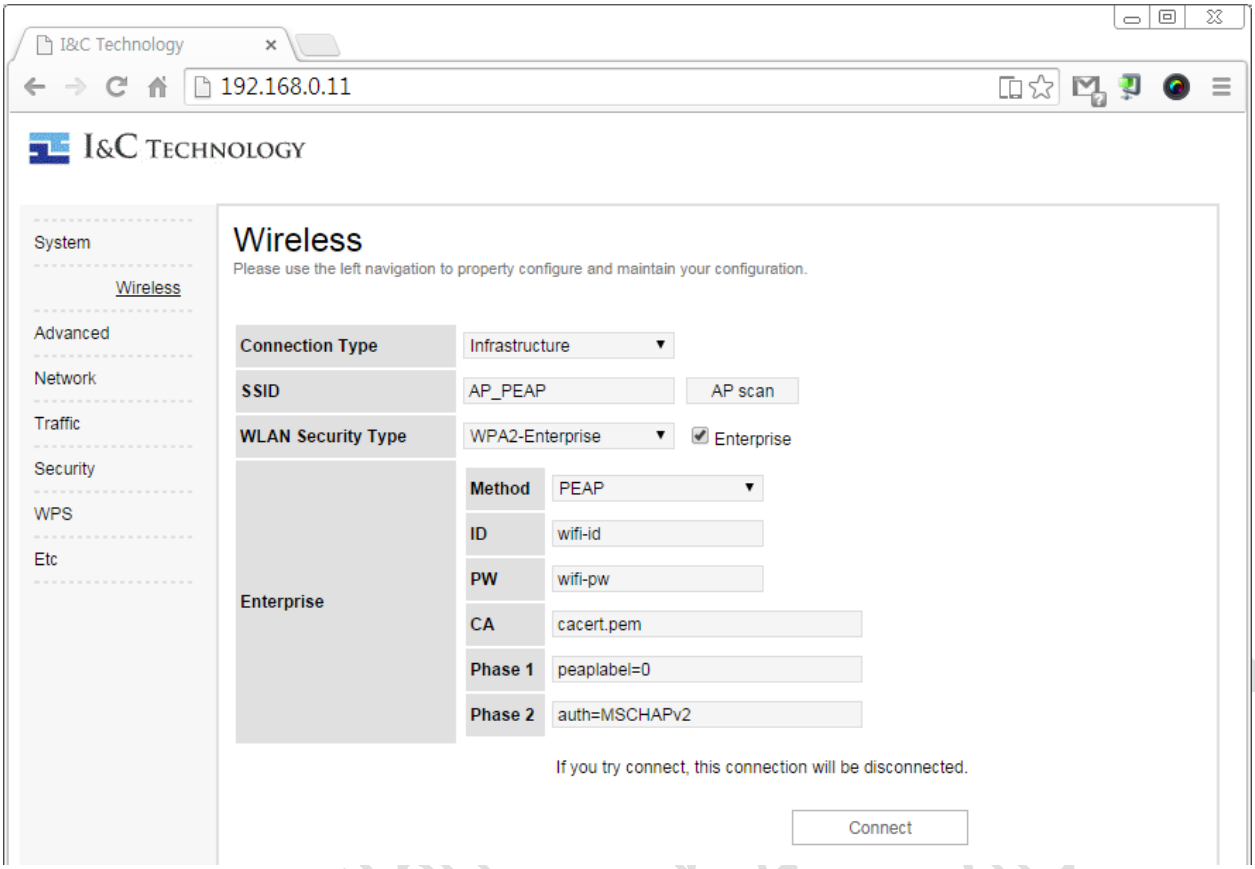


Figure 23. STA Mode Wireless Menu using Enterprise Security

4.1.3 Network

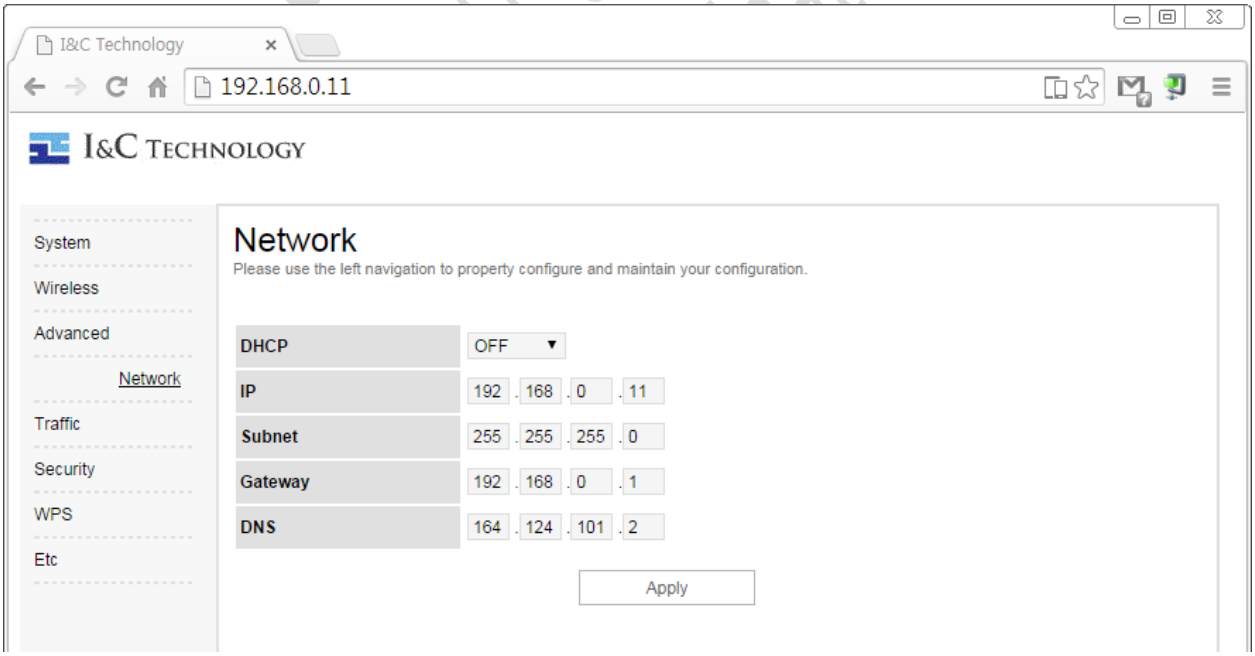


Figure 24. STA Mode Network Menu for WebUI

4.1.4 Traffic

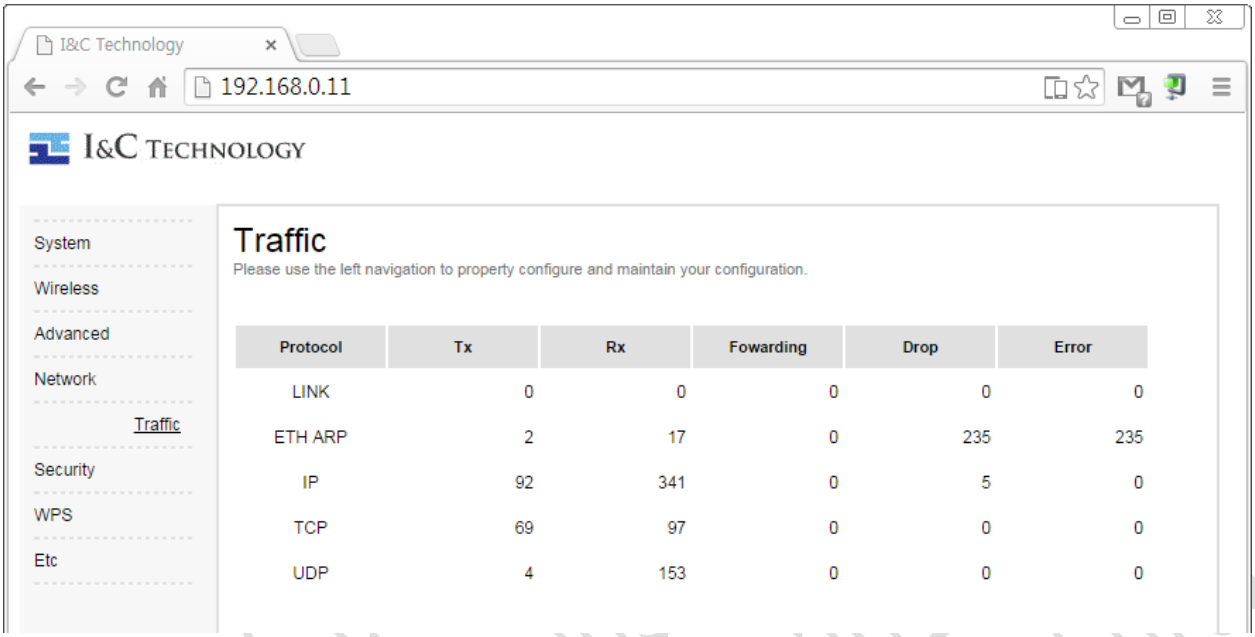


Figure 25. STA Mode Traffic Menu for WebUI

4.1.5 Security

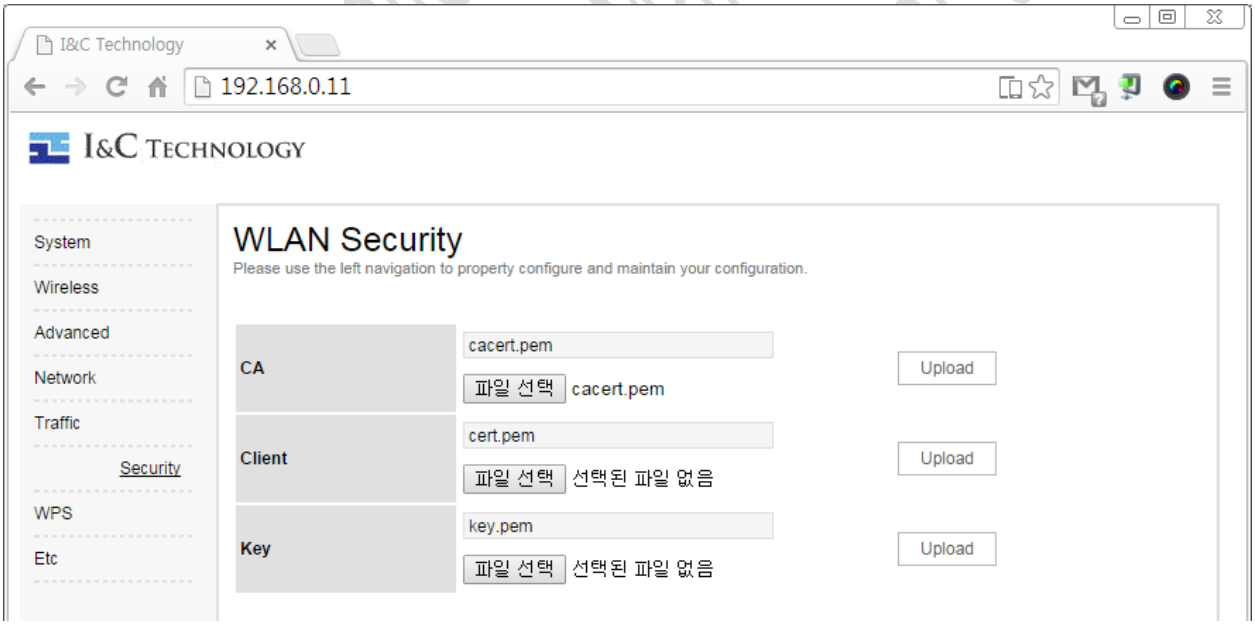


Figure 26. STA Mode Security File Upload Menu for WebUI

4.1.6 WPS

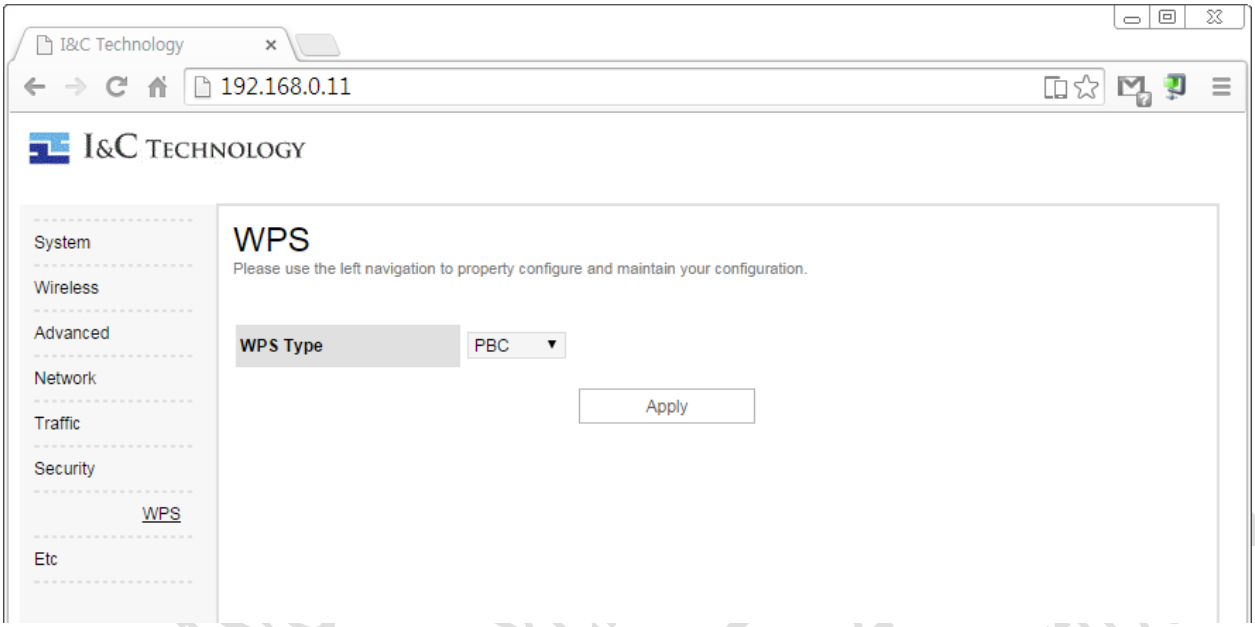


Figure 27. STA Mode WPS Menu for WebUI

4.1.7 Etc

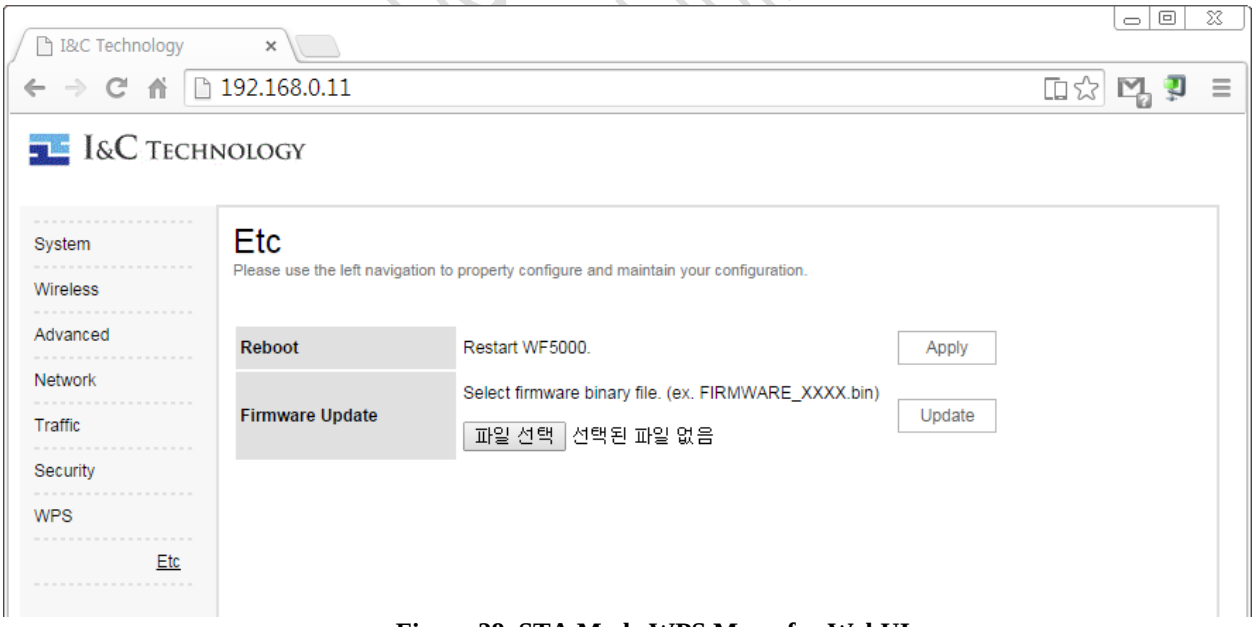


Figure 28. STA Mode WPS Menu for WebUI

4.2 WebUI AP Mode Screen Shot

4.2.1 System

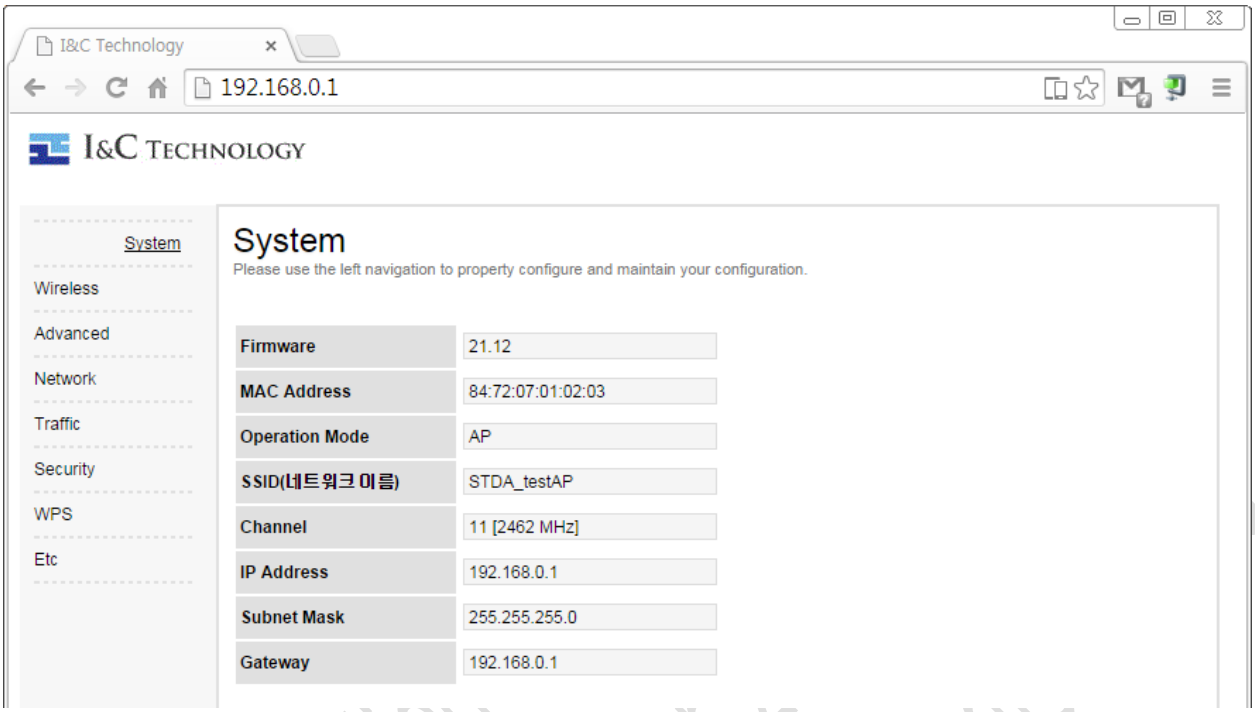


Figure 29. AP Mode System Menu for WebUI

4.2.2 Wireless

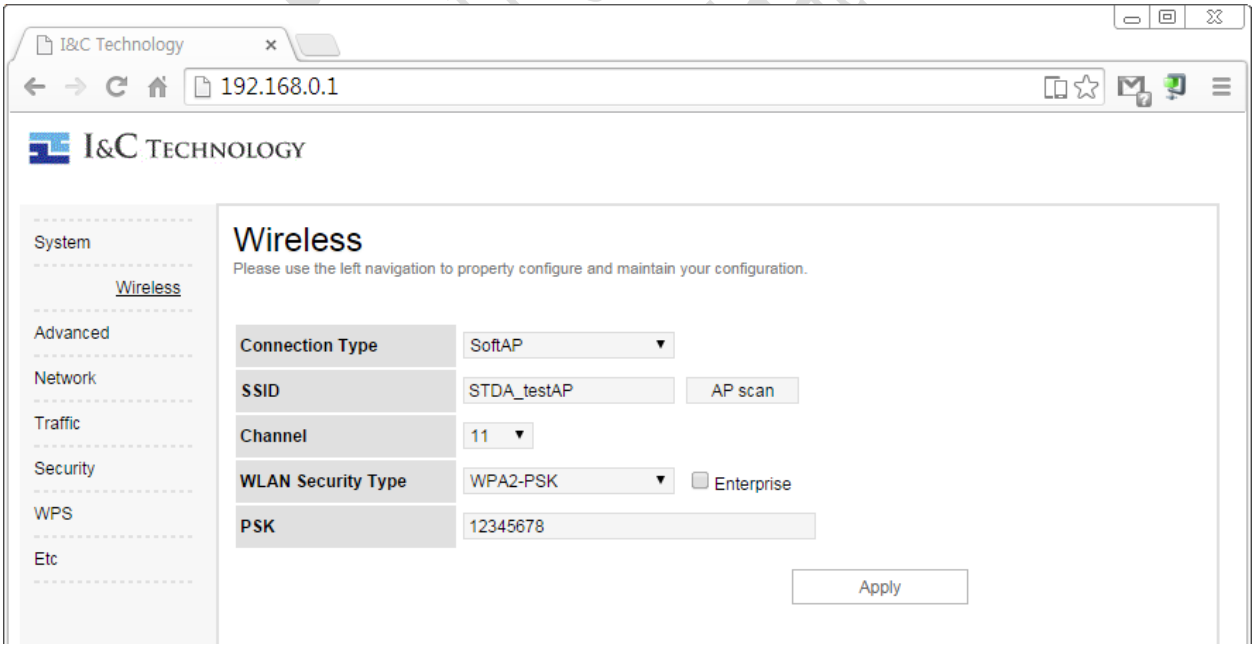


Figure 30. AP Mode Wireless Menu for WebUI

4.2.3 Advanced

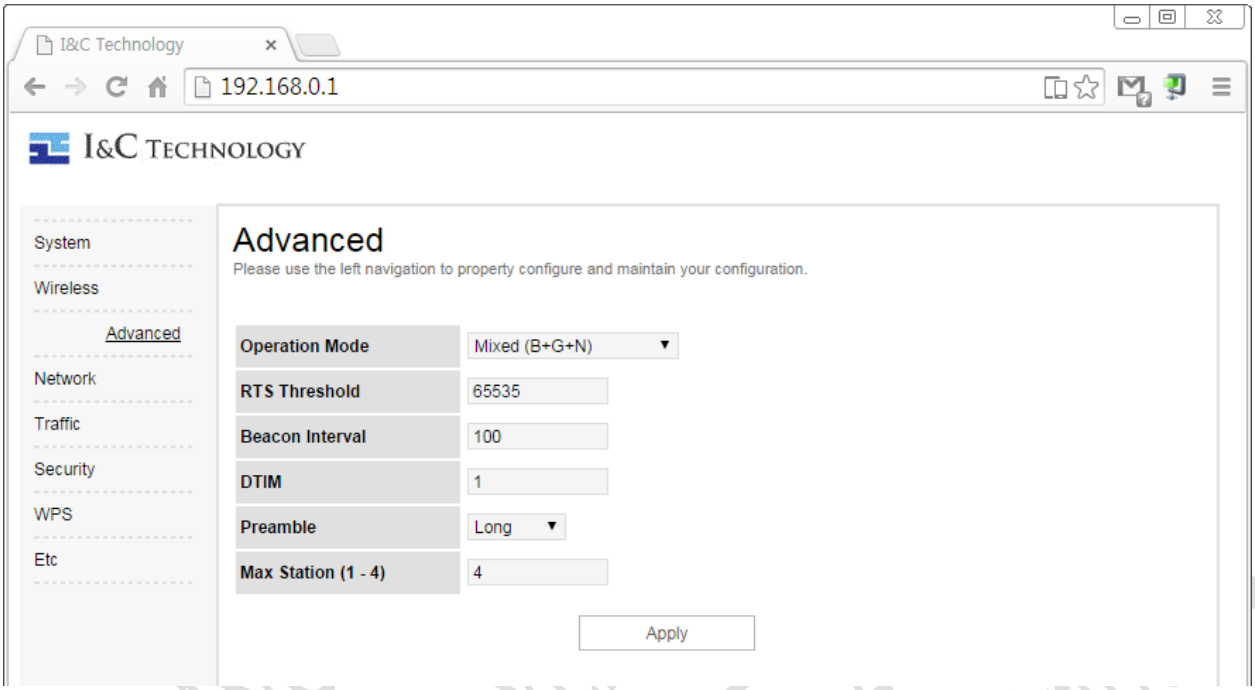


Figure 31. AP Mode Advanced Menu for WebUI

5 Firmware Upgrade

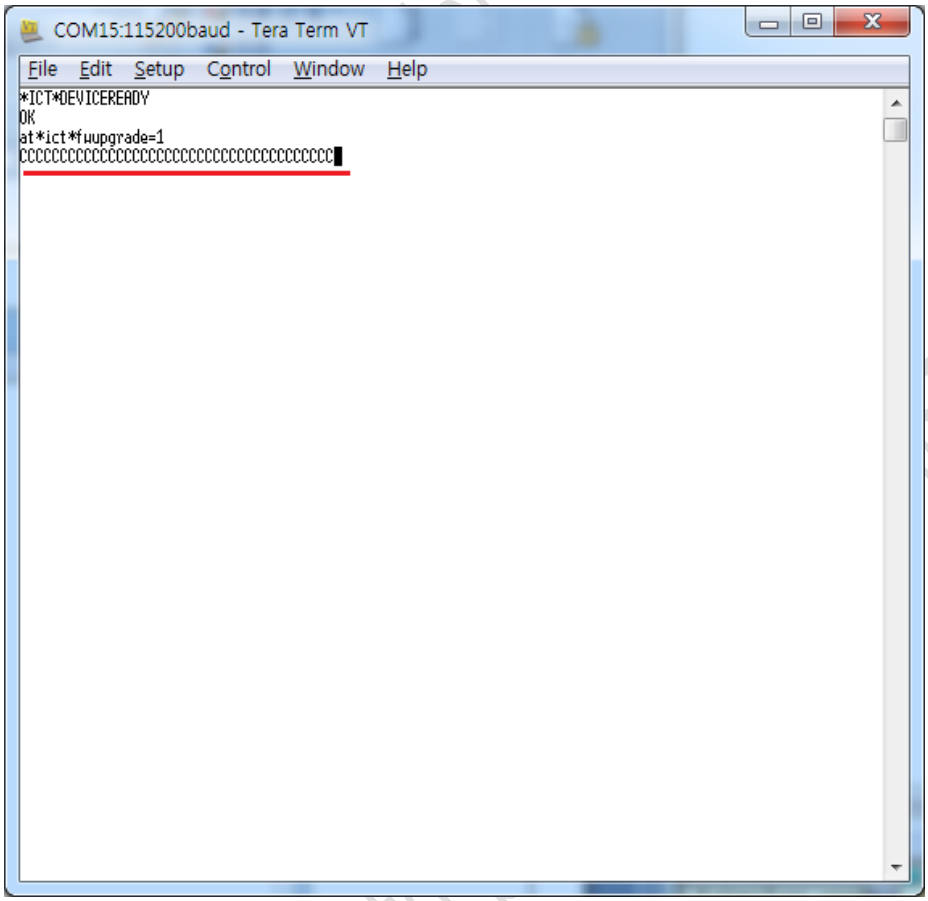
5.1 XMODEM

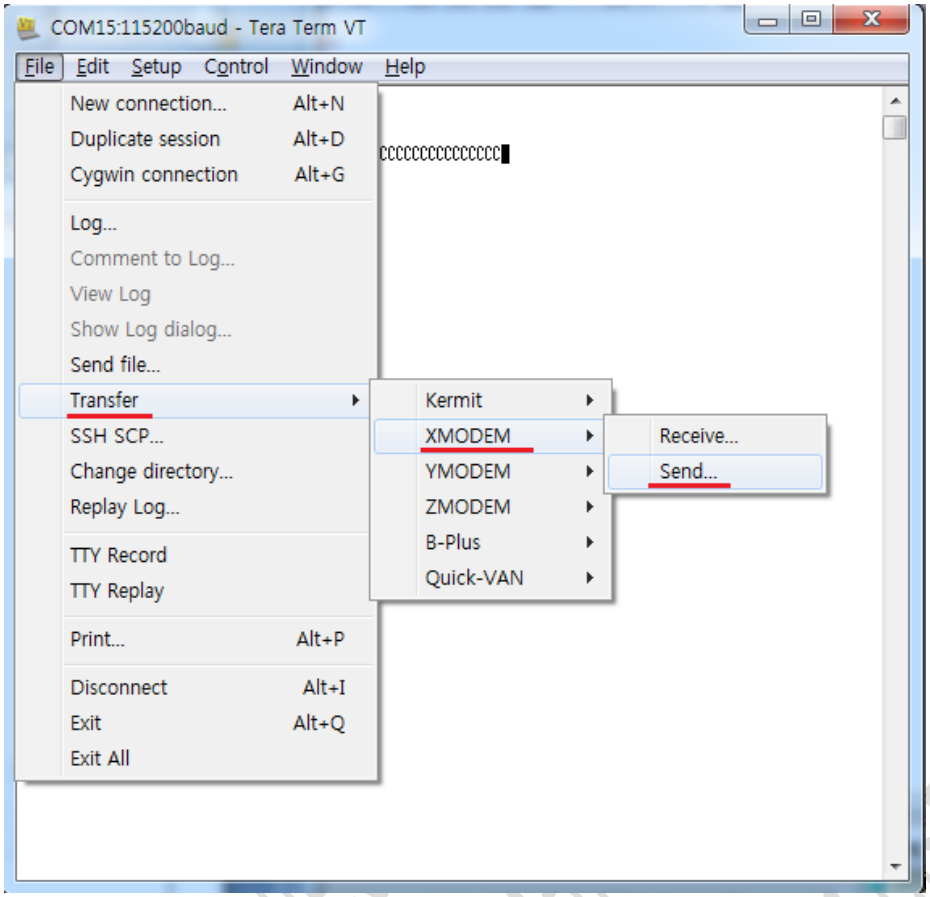
1. Enter firmware upgrade mode : *AT*ICT*FWUPGRADE*

   ▶ **AT*ICT*FWUPGRADE=<bank number>**

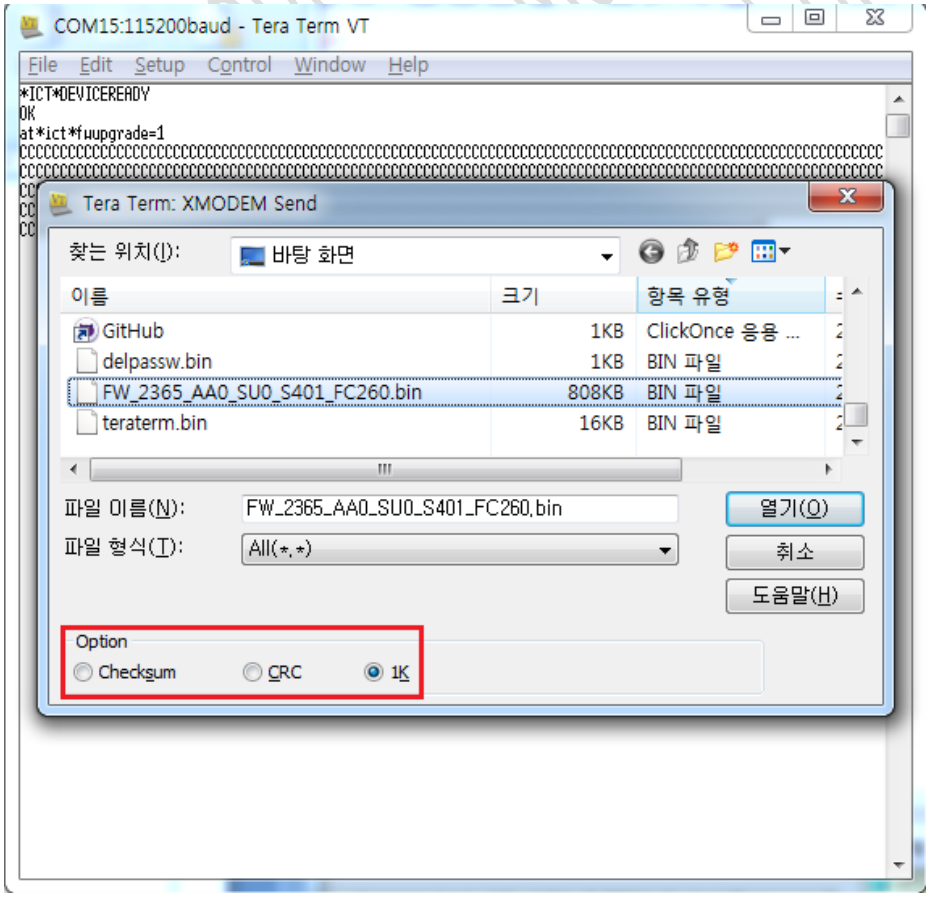
2. After "CCC..." characters display, In Tera Term menu, Choose as below

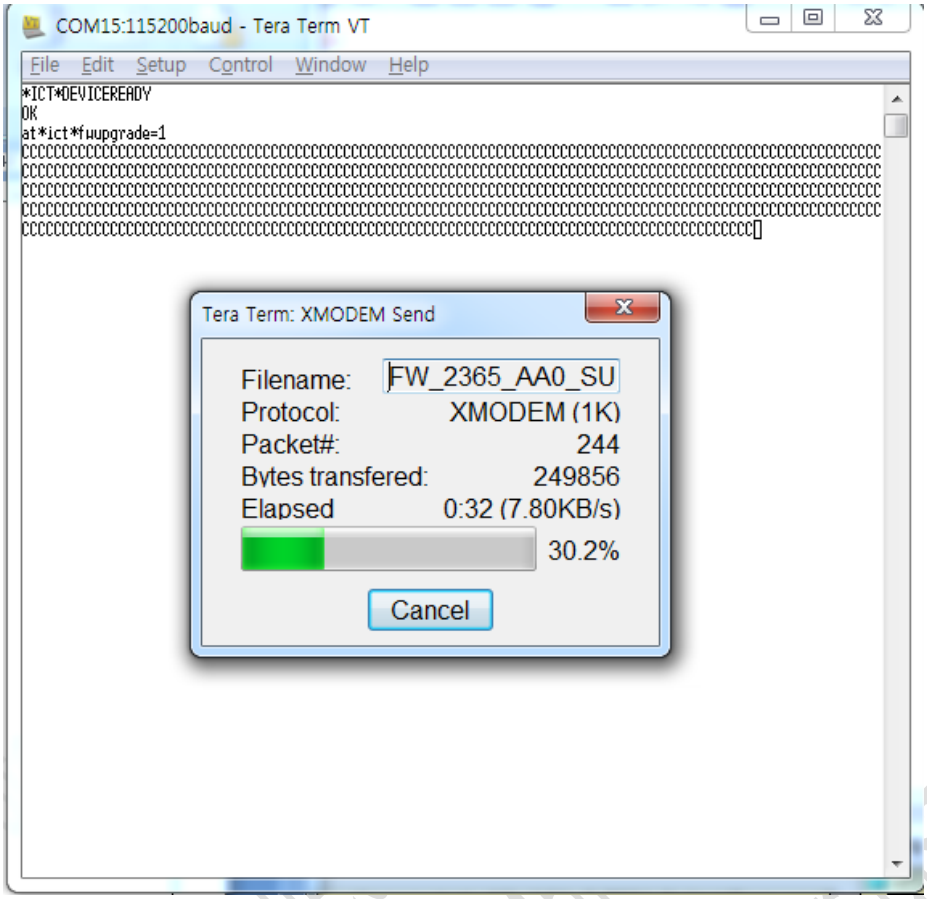
   MENU -> Transfer -> XMODEM -> Send



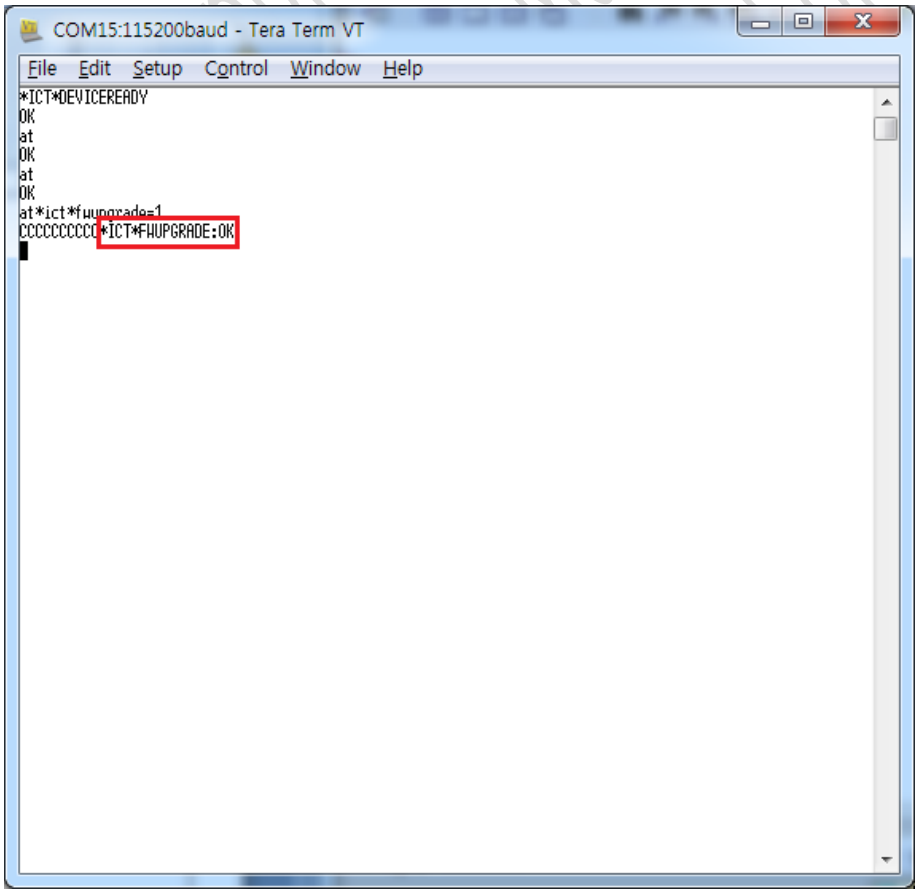


3. Choose a Firmware binary file (i.e, FW_2365_AA0_SU0_S401_FC260.bin) and select a 1K in option.



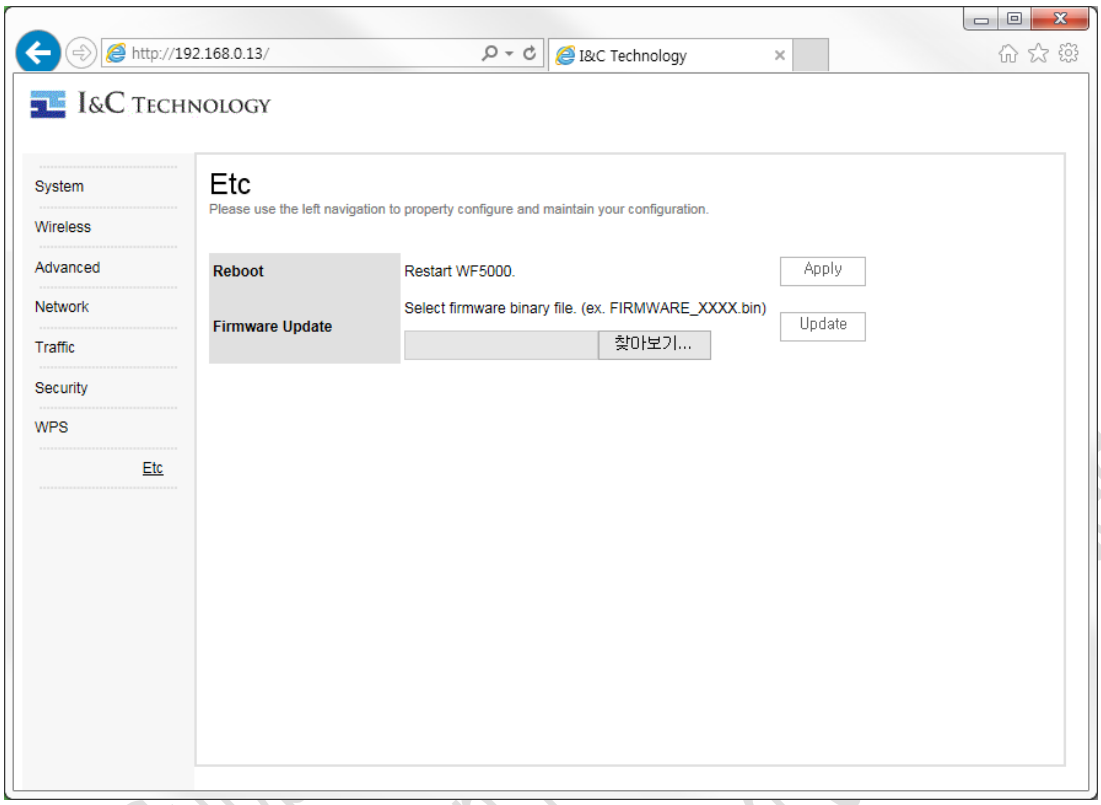


4. After end of transfer, "*ICT*FWUPGRADE:OK" display.

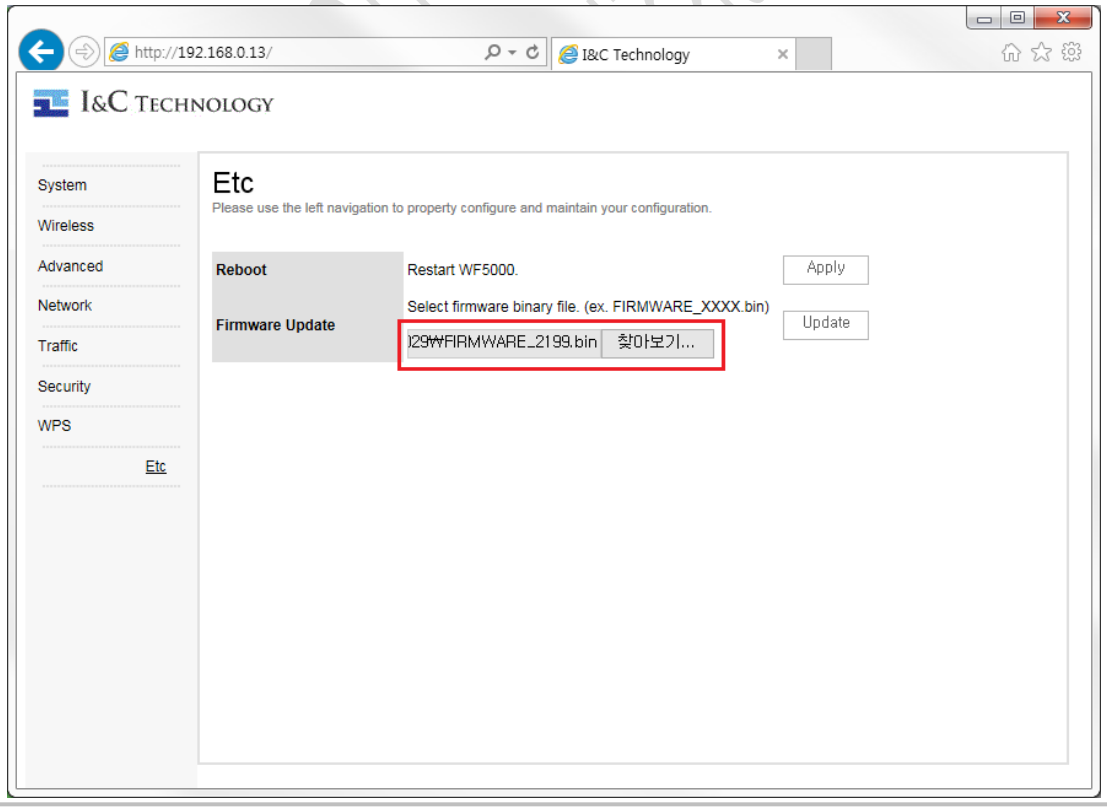


5.2 Web UI

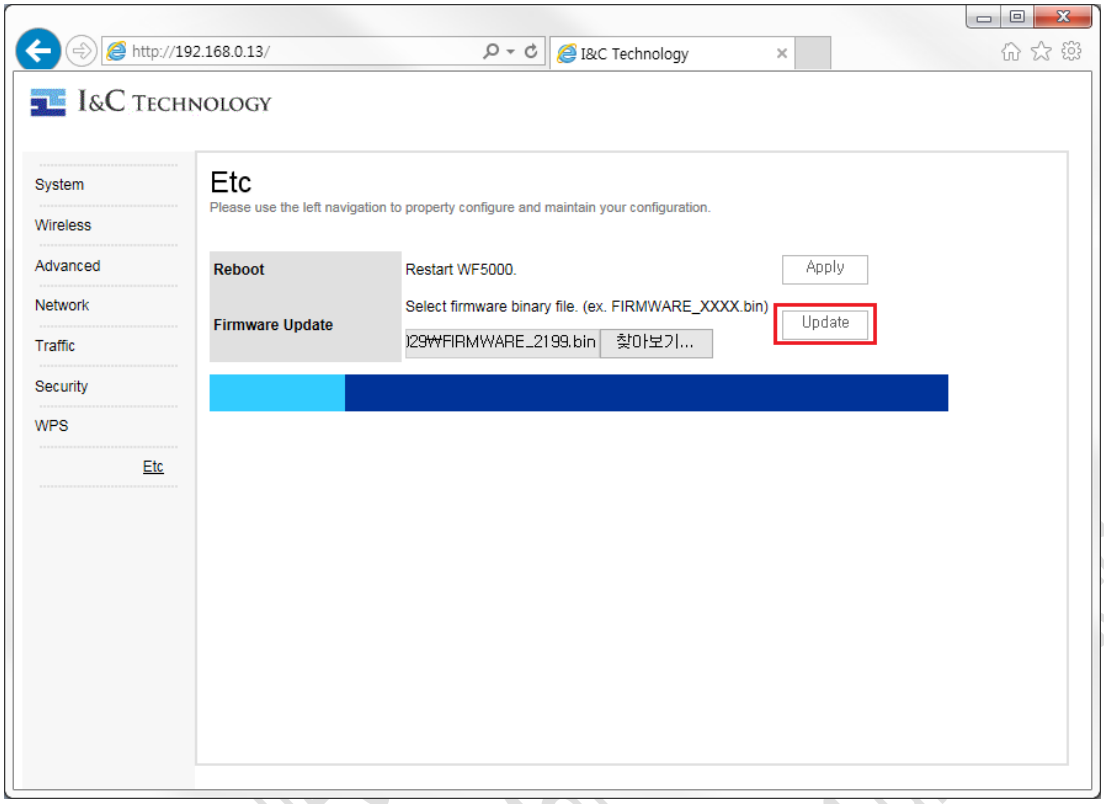
1. Browse “Etc” page of Web UI.



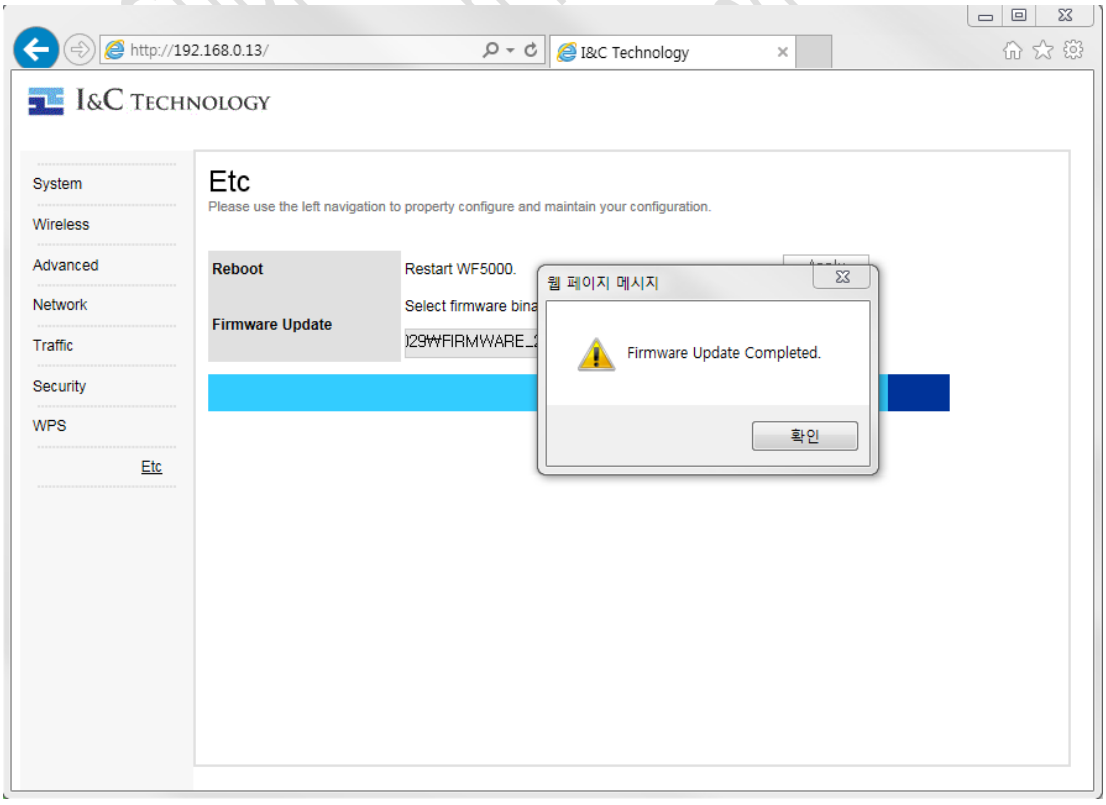
2. Select new firmware binary file.



3. Submit by “Update” button and Wait uploading file.



4. Check the message box to finish success or fail.



6 TestBench Screen Shots for Standalone Mode

6.1 Random Data Loopback test

6.1.1 Summary

A random data loopback test application is used to loopback communications between the PC and a Modem connected to an AP. This program is used to confirm whether data transmission is completed without error by sending data generated randomly by the Testbench program from the PC to the Modem via Wi-Fi and then sending the data to the PC via UART, and integrity of the received data is checked to confirm whether data transmission successfully.

① Send data generated randomly with the Testbench program on a PC (host) automatically to a Modem via Wi-Fi.

② Send the random data received from the Modem to the PC via UART.

③ On the PC, received random data via UART do integrity check to compare first sent data via Wi-Fi. If the data received via UART are identical to the data sent via Wi-Fi, then send the data to the Modem via UART.

④ The Modem send to data received via UART to the PC via Wi-Fi.

⑤ On the PC, compare the data received via Wi-Fi with those received via UART. If they are identical, then add 1 to the counter, and repeat the same test.

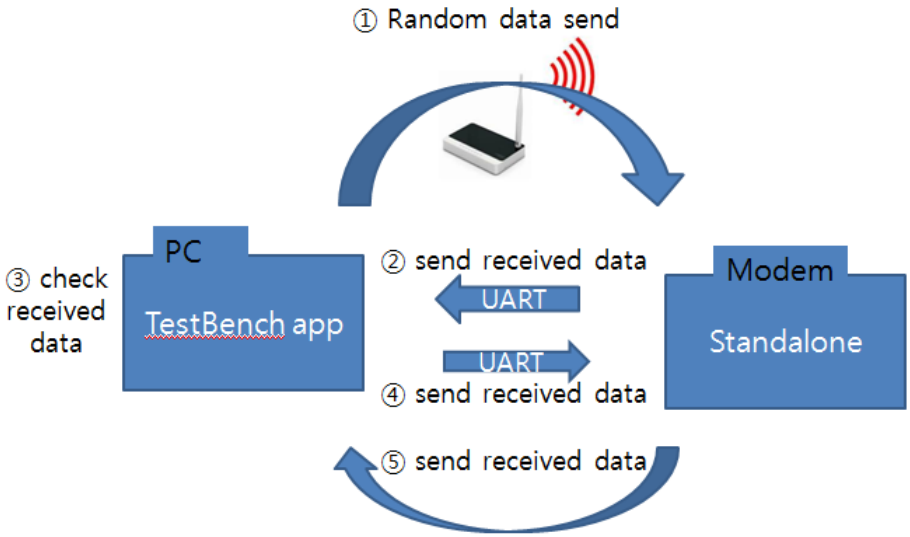


Figure 32. Chart of Random Data Loopback Test

6.1.2 TEST SCREEN

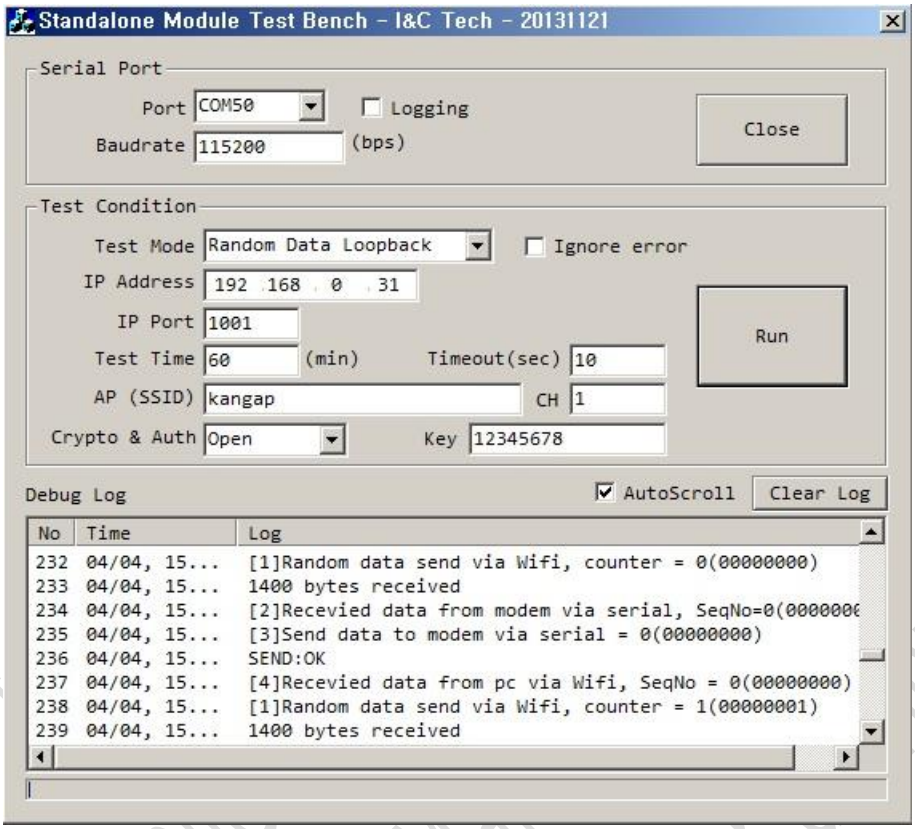


Figure 33. Random Data Loopback Test Screen for Standalone

6.2 Random Data Modem Tx

6.2.1 Summary

The random data Modem Tx is a test of sending random data generated by the Testbench program at the PC to the Modem via UART, and resending the received random data at the Modem to the PC via Wi-Fi. The Testbench program checks the integrity of data whether the initial data sent via UART are identical to the received data via UART.

① Send random generated data with the Testbench program on a PC(host) automatically to the Modem via UART.

② Send the random data received from the Modem to the PC via Wi-Fi.

③ On the PC, progress an integrity check to compare whether the random data received via Wi-Fi are identical to the first data sent via UART, if they are identical, then add up the count, and repeat the same test.

.

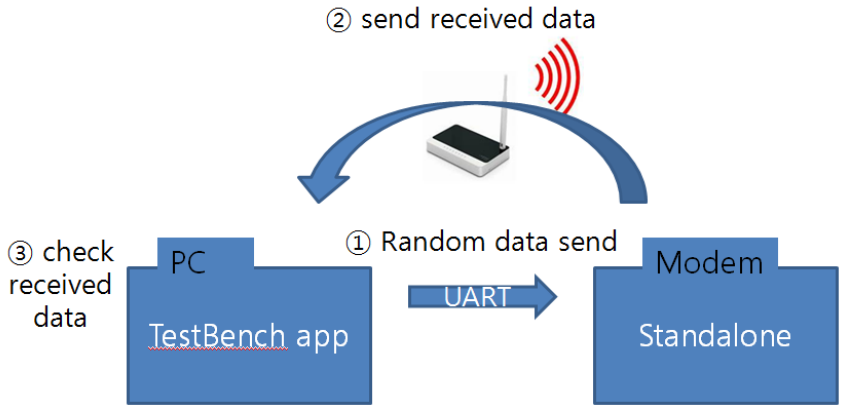


Figure 34. Chart of Random Data Modem Tx

6.2.2 TEST SCREEN

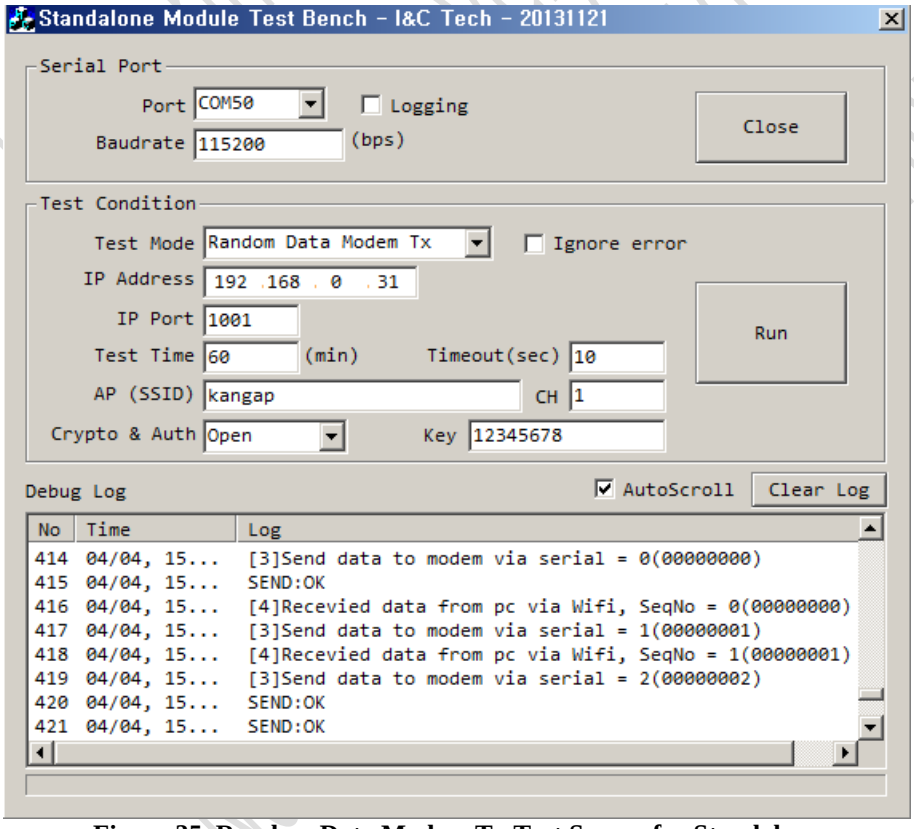


Figure 35. Random Data Modem Tx Test Screen for Standalone

6.3 Random Data Modem Rx

6.3.1 Summary

The random data Modem Rx is a test of sending random data generated by the Testbench program at the PC to the Modem via Wi-Fi, and resending the received random data at the Modem to the PC via UART.
The Testbench program checks the integrity of data whether the initial data sent via Wi-Fi are identical to the received data via UART.

① Send random generated data with the Testbench program on a PC(host) automatically to the Modem via Wi-Fi.

② Send the random data received from the Modem to the PC via UART.

③ On the PC, progress an integrity check to compare whether the random data received via UART are identical to the first data sent via Wi-Fi, if they are identical, then add up the count, and repeat the same test.

.

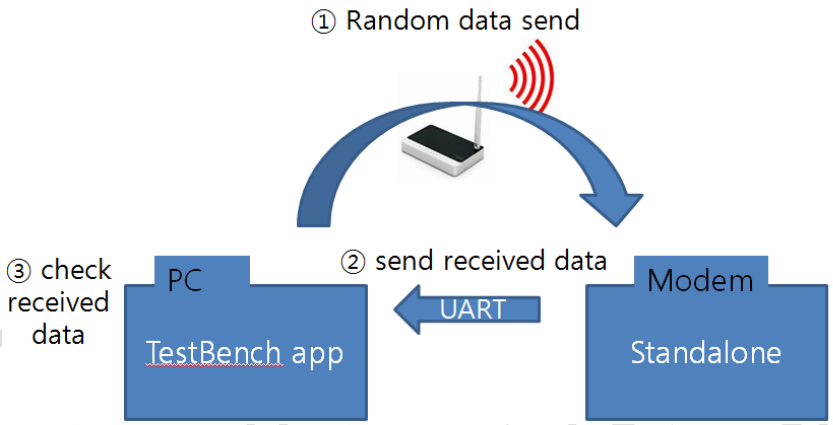


Figure 36. Chart of Random Data Modem Rx

6.3.2 TEST SCREEN

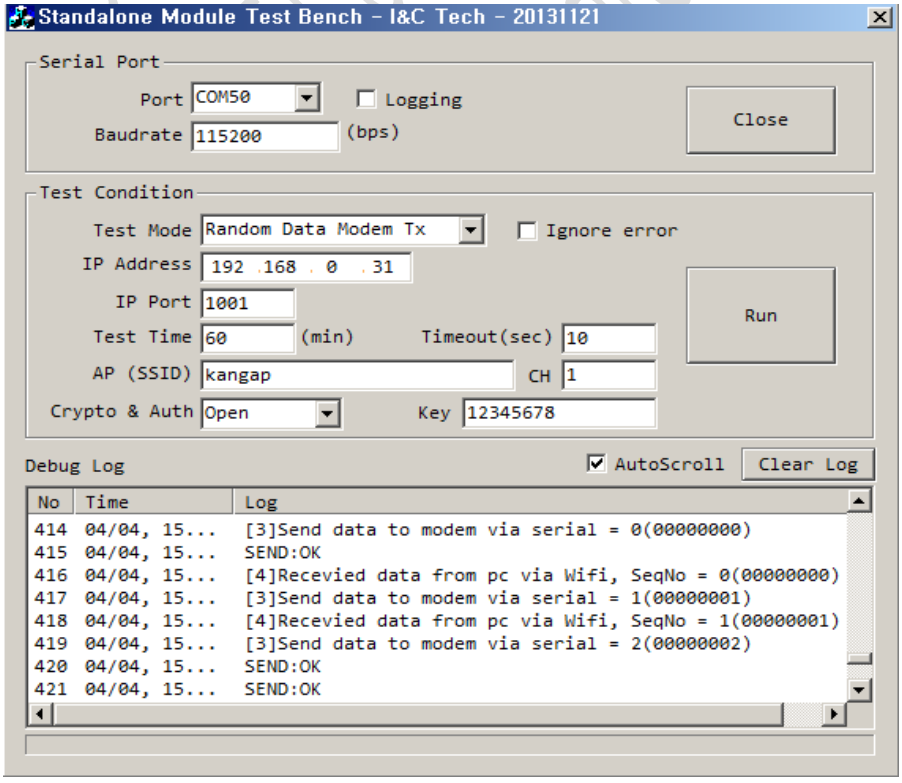


Figure 37. Random Data Modem Rx Test Screen for Standalone

6.4 Repeat Association

6.4.1 Summary

The repeat association is repetitive association test that is connected to an AP after association and get an IP address from DHCP server, and then it disconnects the AP and tries to connect again.

① Send a command at the PC (host) to the Test bench program to connect to an AP via UART.

② The Modem tries to connect AP to a SSID configured after scanning.

③ When the association succeeds and allocated an IP address from the AP, then the Modem notifies the Testbench program via UART.

④ The Testbench program sends a disassociation command to the Modem via UART, and disconnects with the AP, and adds up the count.

⑤ Repeat from 1 to 4.

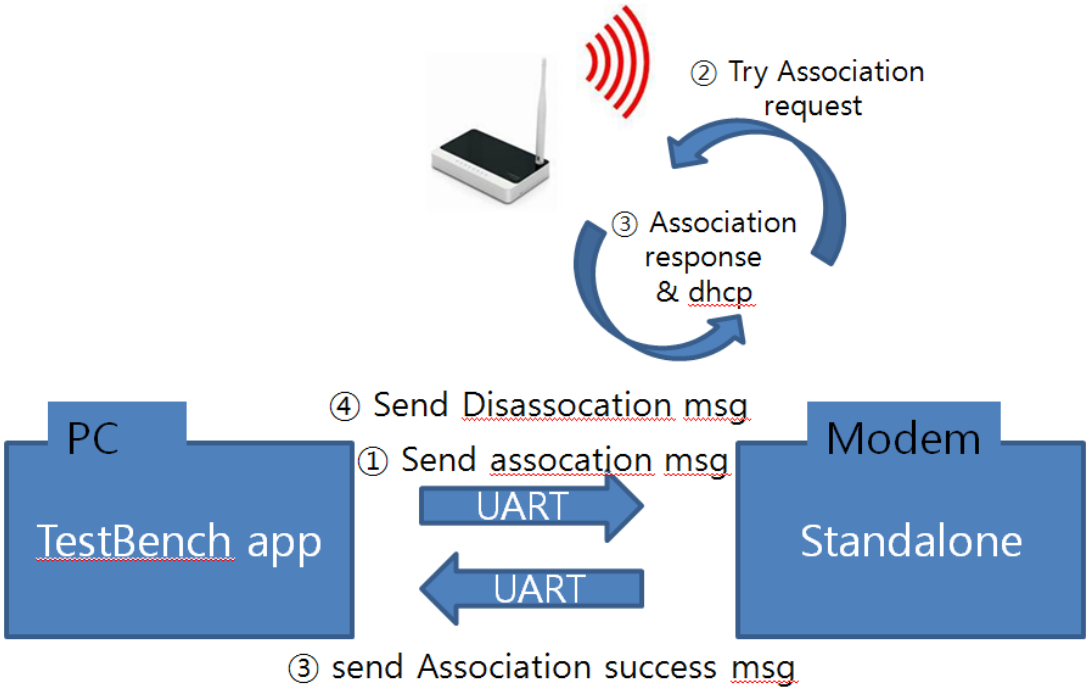


Figure 38. Chart of Repeat Association Test

6.4.2 TEST SCREEN

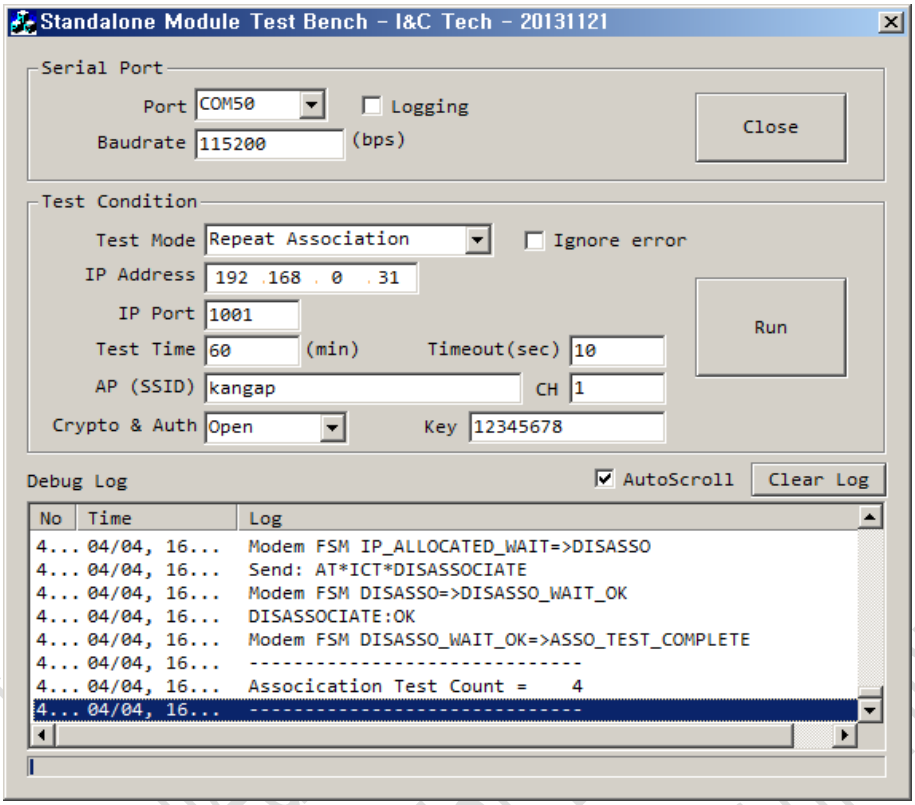


Figure 39. Repeat Association Test Screen for Standalone

7 Application Protocol Program for Standalone Mode

7.1 Hardware Power Save on Standalone mode with UART interface

7.1.1 Summary

If Hardware Power Save is enabled on Modem side, Modem could be stayed in PS mode. In this case, Modem could not wake up by AT commands through UART interface. One of solutions to solve it is making a wired connection between RXD pin of UART used by AT command and SDIO command pin. It is a simple way to be awake from PS mode through the SDIO wake up line. Below describes the sequence that Modem is wake up by an AT command when RXD pin of UART2 (GPIO13) is directly connected to SDIO command pin.

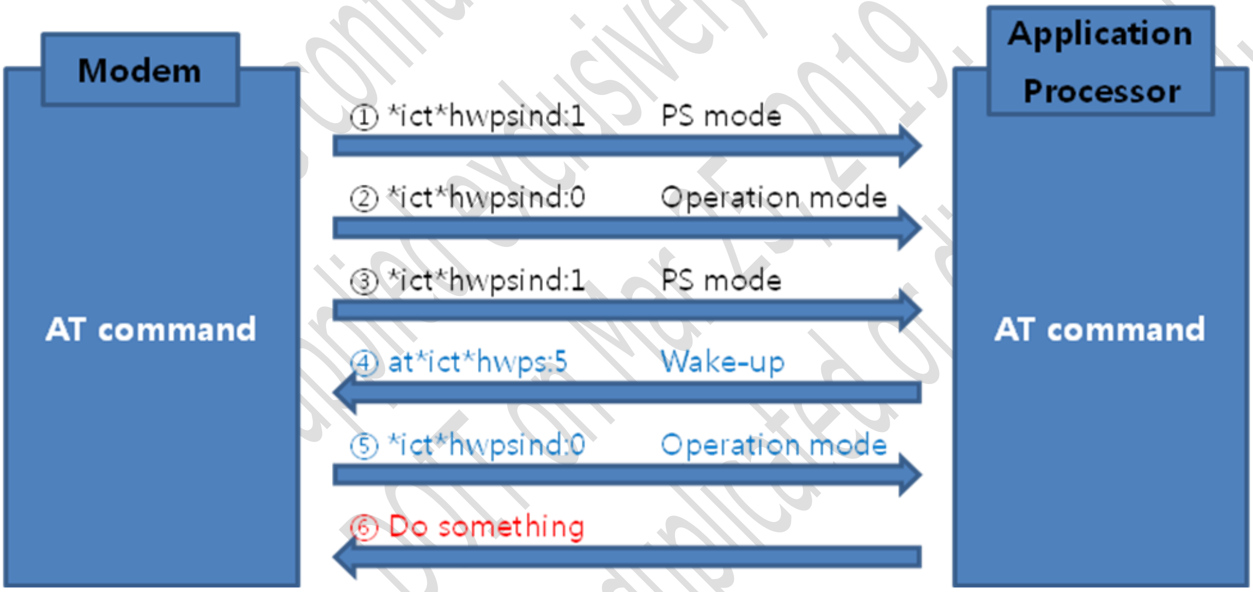


Figure 40. The Sequence to wake modem up from PS mode by AT command

8 Binary UART Protocol for Standalone Mode(Optional)

8.1 Binary UART protocols

8.1.1 Summary

Type of Protocol can be composed of Traffic and command as operation. Packet's PID is composed of Command and Indication. Command message was used to send to the Modem for executing the command. The Indication, Modem used this command when transferring data asynchronously to the Host System. The ACK or NACK packet can identify a result of doing instruction except some Indication was required a response. System default protocol is AT command mode. If you want to use the Binary Protocol mode, you must switch modes AT command.

<table>
<tr><th>AT command -> Binary protocol</th><th>Binary protocol -> AT command</th></tr>
<tr><td>AT*ICT*UARTPROTO
HEX:
41 54 2A 49 43 54 2A 55 41 52 54 50 52 4F 54 4F 0D</td><td>Packet Frame (6 bytes) : Host → Modem
<table>
<tr><td>0xF0</td><td>0x24
(36)</td><td>0x0000
(2bytes)</td><td>Data</td><td>0x24</td><td>0xE0</td></tr>
</table>
HEX:
F0 24 00 00 24 E0</td></tr>
</table>

8.1.2 Setup for use Binary UART protocol

The Modem and Host System, you must set the properties of the UART in order to UART communication.

1. Data: 8bit
2. Parity: None
3. Stop Bit: One
4. Handshake: None
5. Baud rate: 115200 bps

8.1.3 General Packet Format

Data that is sent to the UART in order to execute the different command types, will send by packed specific format. The transmission unit in this category is called PACKET. PACKET has the following structure.

<table>
<tr><td>SYNC(0xF0)</td><td>PID(1byte)</td><td>Length(2byte)</td><td>Data(0~n byte)</td><td>CheckSum
(1byte)</td><td>EndMark
(0xE0)</td></tr>
</table>

Figure 41. Binary Protocol PACKET structure

In this case, *Sync* and *Endmark* means beginning and end of the Frame. *Sync* and *Endmark* has a size of each 1bytes. The PID can see the kind of the PID Table that describes the characteristics of each Packet.

Length means size of real data packet that has 2bytes volume and Little Endian format (Low byte comes first). It is transferred only instructed in Length, if Length is of 0, Data does not exist, and Data location was replaced by CheckSum. All data types passed to the data must be processed in Little Endian low data are located in the first. CheckSum means a value obtained by adding 1 byte format to Data from PID. If the total size is greater than the 1 byte then take only low byte. In part that handles the protocol of the Packet, and the transition to the Sync state input as soon as it receives the data that exceed the specified format, to ignore all the data until the receiving next Sync data.

For more details, refer to the “WF5000_Binary_UART_Protocol_User_Guide" document.

9 UDAP Discovery Protocol

9.1 Overview

WF5000 device supports device discovery protocol. The discovery process is a cal l and response process utilizing the UDP communications protocol. The managing processor makes a network broadcast, using UDP, targeting the UDAP port on WF5000 device. The UDP packet has a predefined format that uniquely defines the UDAP discovery request.

All WF5000 devices receiving the packet will then attempt to respond, again using UDP protocol. Their response will contain specific information about them including their MAC address, Information, like the IP address can be extracted from the UDP packet header

9.2 Discovery Request

The host attempting to discover the WF5000 devices must send the following data packet via a UDP broadcast or unicast.

Item	Description
Communication Protocol	UDP broadcast or UDP unicast
Target Port	UDP Port (User Defined Port) [default: 47556]
Data	0x49, 0x26, 0x43, 0x20, 0x54, 0x65, 0x63, 0x68, 0x6e, 0x6f, 0x6c, 0x6f, 0x67, 0x79, 0x2e, 0x2e
Message Options	See Discovery Request Message Option Format

9.2.1 Message Option Format

Item	Length	Value	Description
Get	2 bytes	0x1000	Get WF5000 general information
Get specific item	Reserved	Reserved	Get specific WF5000 information
Set	Reserved	Reserved	
Trap	Reserved	Reserved	

9.3 Discovery Response

Upon receipt of the discovery request the WF5000 unit will response via a UDP unicast, providing device specific information. The broadcast response will typically be sent within 200msec of the request being received.
WF5000 device will response to all requests (all received broadcast discovery request packets) and there for may be seen multiple times. It is possible for broadcast UDP packets to be duplicated by routes within the network. It is therefore necessary for the host to be able to handle this possibility.

Item	Description
Communication Protocol	UDP unicast
Source IP	WF5000 IP address is the UDP packet source address
Target Port	Discovery Request packet source port
Data	See Discovery Response Data Format
Message Options	See Discovery Response Message Option Format

9.3.1 Data Format

Item	Length	Type	Description
SSID	32 bytes	CHAR	AP's SSID
BSSID	6 bytes	HEX	AP's BSSID
Frequency	2 bytes	INT	Center frequency
RSSI	2 bytes	INT	AP's RSSI

MAC Address	6 bytes	HEX	MAC address of WF5000
Serial Number	32 bytes	CHAR	The Serial Number is stored in flash memory. The Serial Number is set by SETMIB AT command.
Version	4bytes	INT	WF5000's firmware version
Device Type	2bytes	INT	Device Type (Options)
Device Code	2bytes	INT	Device Code (Options)
Model Name	32bytes	CHAR	Model Name (Options)
Manufacture	16bytes	CHAR	Manufacture (Options)

9.3.2 Message Option Format

Not available

Item	Length	Value	Description

9.4 Discovery Notification

The WF5000 send the host information, the following data packet via a UDP unicast.

Item	Description
Communication Protocol	UDP unicast
Target Port	UDP Port (User Defined Port) [default: 47556]
Data (Magic Code)	0x49, 0x26, 0x43, 0x20, 0x54, 0x65, 0x63, 0x68, 0x6e, 0x6f, 0x6c, 0x6f, 0x67, 0x79, 0x2e, 0x2e
Message Options	See Discovery Notification Message Option Format
Data	See Discovery Notification Data Format

9.4.1 Message Option Format

Item	Length	Value	Description
Trap	2 bytes	0x4000	Trap information of WF5000

9.4.2 Data Format

Item	Length	Type	Description
SSID	32 bytes	CHAR	AP's SSID
BSSID	6 bytes	HEX	AP's BSSID
Frequency	2 bytes	INT	Center frequency
RSSI	2 bytes	INT	AP's RSSI
MAC Address	6 bytes	HEX	MAC address of WF5000
Serial Number	32 bytes	CHAR	The Serial Number is stored in flash memory. The Serial Number is set by SETMIB AT command.
Version	4bytes	INT	WF5000's firmware version
Device Type	2bytes	INT	Device Type (Options)