APPLICATION NOTE

ATWINC1500 AT Command Reference Guide

Atmel ATWINC1500

Introduction

This guide describes usage of AT commands with the ATWINC1500 Wi-Fi module. The AT command firmware is running simultaneously with the Wi-Fi firmware on the MCU of the ATWINC1500 and these AT commands are used to operate the driver functionality of the ATWINC1500 including Wi-Fi interface, TCP/IP stack and so on.

The reader is supposed to be aware of network protocol and operation of wireless devices.

This guide is currently based on ATWINC1500 AT command firmware version 0.9.1 and ATWINC1500 Wi-Fi firmware version 19.4.4.

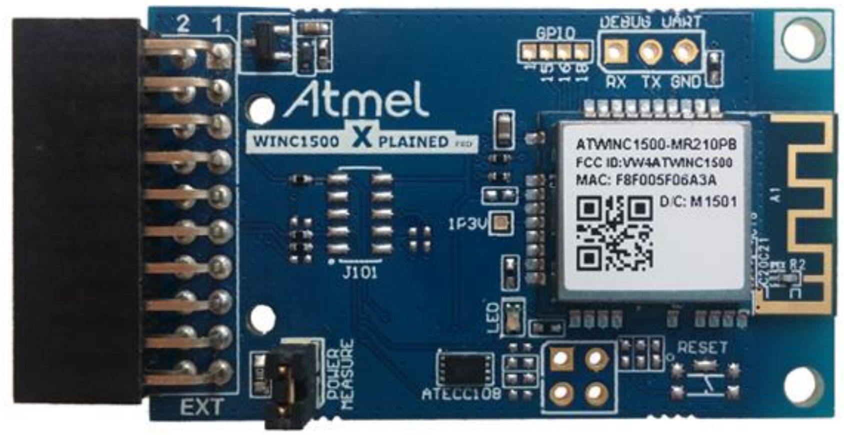


Table of Contents

Icon Key Identifiers

TIP	Useful Tips and Techniques
INFO	Delivers Contextual Information About a Specific Topic
IMPORTANT	Note to Quality and Performance
TO DO	Objectives to be Completed
EXECUTE	Actions to be Executed Out of the Target
RESULT	The Expected Result of an Assignment Step
CAUTION	Procedure Which Can Result in Minor Equipment Damage
WARNING	Procedure With Potential Equipment Damage
DANGER	Procedure With Imminent Equipment Destruction

1 AT Command Architecture

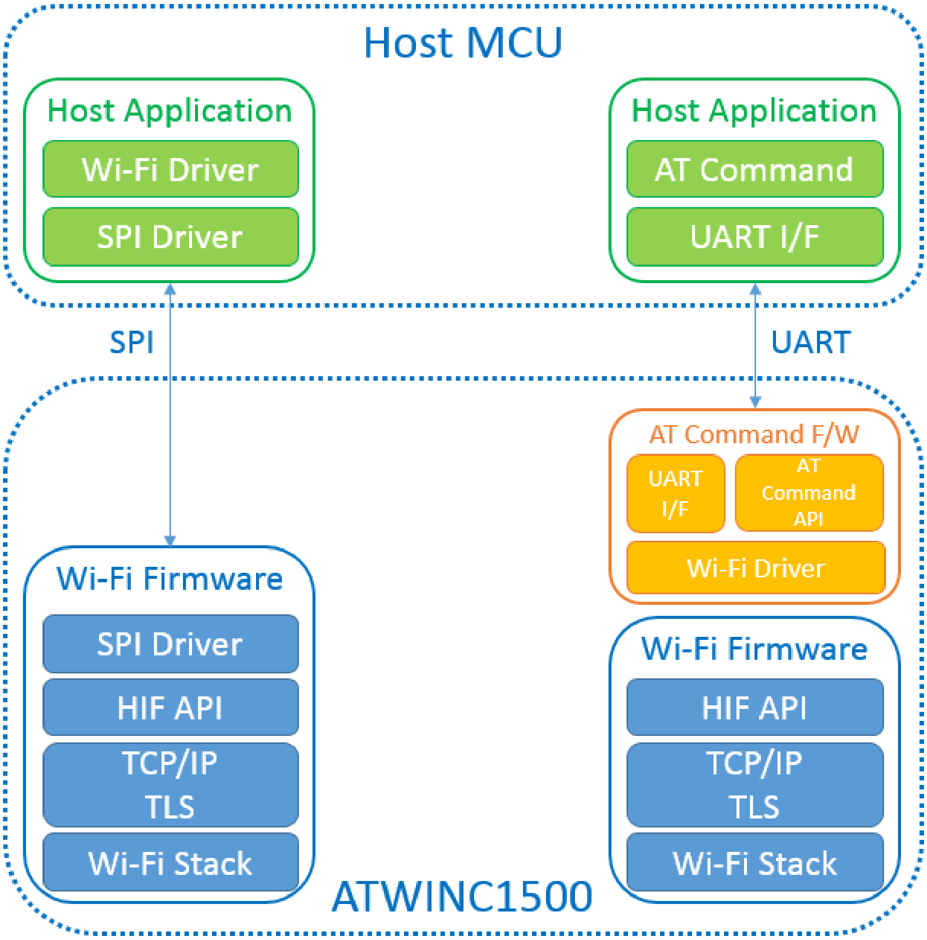
1.1 Software Architecture of ATWINC1500 Solution

On the ATWINC1500 Wi-Fi module, there are two possibilities to implement a Wi-Fi application which has the Wi-Fi firmware driver.

- Application running on a host MCU communicating with the ATWINC1500 through a BUS interface
- Application running on ATWINC1500 simultaneously with the Wi-Fi firmware process

Figure 1-1 shows the difference of software architecture between two methods.

Figure 1-1. Application architecture



1.1.1 Application running on a host MCU

The HIF (Host Interface Layer) is responsible for handling the communication mechanism between the host application and the Wi-Fi firmware. This includes Interrupt handling, DMA control and management of communication logic between the Wi-Fi firmware driver at host MCU and the ATWINC1500 firmware.

1.1.2 Application running on ATWINC1500

The AT command process runs simultaneously with the Wi-Fi firmware process on the MCU of the ATWINC1500 and each of them has its own resources (data and instruction memories).

The communication between the Wi-Fi firmware process and the AT command process is implemented in the AT command firmware through a shared memory area.

The AT command firmware should be downloaded to the ATWINC1500 separately from the Wi-Fi firmware in order to use the AT command interface. But this AT command firmware provides minimized commands due to limited memory size of ATWINC1500.

1.2 Host Application for AT Command

The AT command firmware provides dozens of AT commands to operate Wi-Fi functions. The host application can control Wi-Fi module by sending AT commands to the ATWINC1500 through UART interface.

The AT command set is listed in the following section.

1.3 Communication Interface

UART interface is used for communication between the host MCU and the ATWINC1500. The baud rate and the flow control function can be set with “AT+UART” command. The default setting is as below.

- 115200 bauds
- 8 bit data
- No parity
- One stop bit
- No flow control

1.4 Limitation

INFO

This AT command firmware can be used with ATWINC1500B (Chip ID: 1503A0) or higher revision.

2 Download the Firmware

Wi-Fi driver is coupled with Wi-Fi firmware and the driver is located in the AT command firmware. So the AT command firmware version should be matched up with the Wi-Fi firmware version. You can download each firmware individually into the ATWINC1500.

WARNING Several previous versions of ATWINC1500 AT command firmware (version 0.6.1 ~ 0.7.8) used auto start mode on UART2 by programming an eFuse. These versions are incompatible with the current version so an ATWINC1500 module which has been programmed with one of these versions can't use the current version again.

2.1 Download via Serial Bridge

It's possible to use the SAMD21 Xplained Pro (SAMD21 evaluation kit) as a serial bridge device to download firmware into the ATWINC1500 if you have an ATWINC1500 Xplained Pro. The SAMD21 receives the firmware data from the PC and write it into the ATWINC1500 through SPI interface.

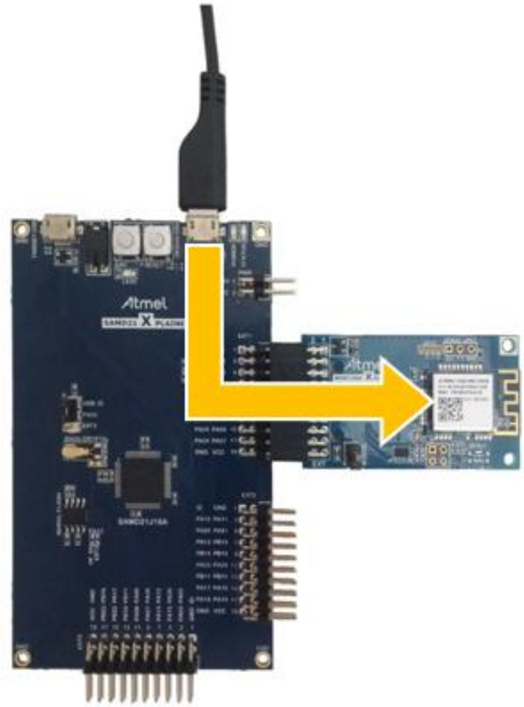
You have to install the Atmel Studio on your PC to launch the download batch file.

It uses the following hardware:

- The SAMD21 Xplained Pro
- The ATWINC1500 on EXT1 header

Connect the SAMD21 (DEBUG USB PORT) to a USB port on your PC.

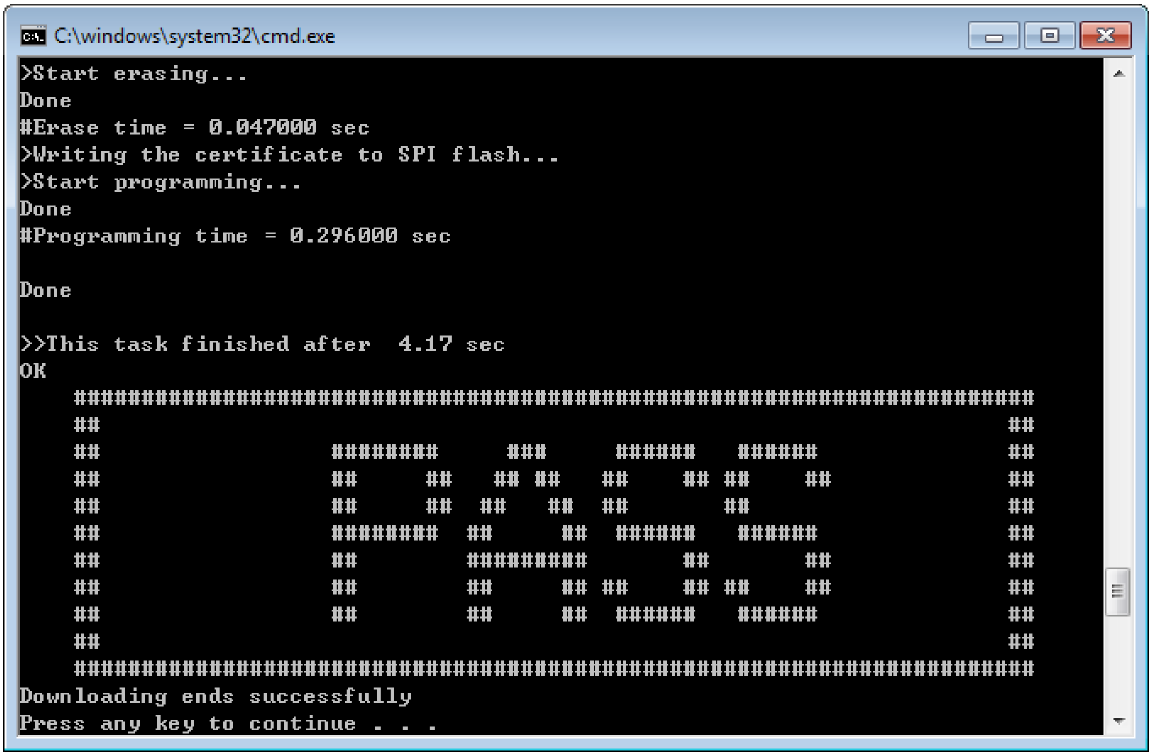
WARNING Do not connect the SAMD21 VIRTUAL COM PORT via terminal program.



2.1.1 Download the Wi-Fi Firmware

Go to the AT command project folder and launch the batch file.

\firmware_tools\download_all_sb_samd21_xplained_pro.bat

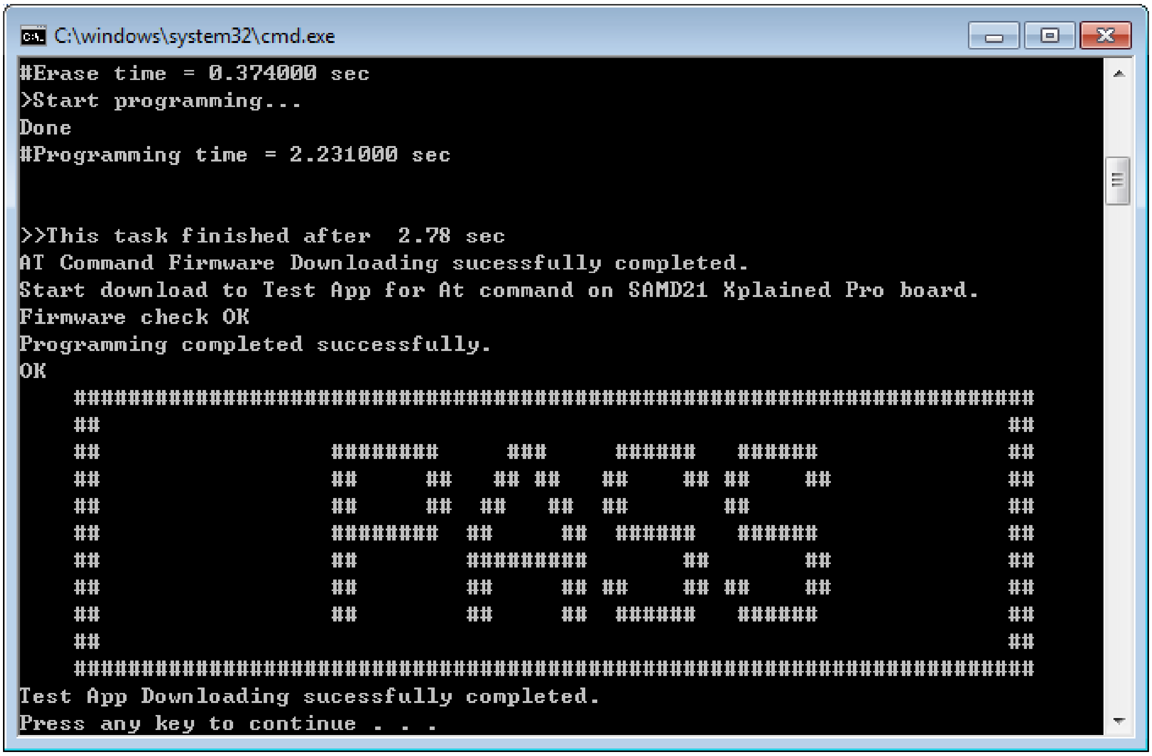


```
>Start erasing...
Done
#Erase time = 0.047000 sec
>Writing the certificate to SPI flash...
>Start programming...
Done
#Programming time = 0.296000 sec

Done

>>This task finished after  4.17 sec
OK
    #######################################################################
    ##                                                                   ##
    ##                 ########     ###     ######   ######              ##
    ##                 ##     ##   ## ##   ##    ## ##    ##             ##
    ##                 ##     ##  ##   ##  ##       ##                   ##
    ##                 ########  ##     ## ######   ######               ##
    ##                 ##        #########       ##       ##             ##
    ##                 ##        ##     ## ##    ## ##    ##             ##
    ##                 ##        ##     ##  ######   ######              ##
    ##                                                                   ##
    #######################################################################
Downloading ends successfully
Press any key to continue . . .
```

2.1.2 Download the AT Command Firmware

Launch the batch file.

\firmware_tools\download_at_cmd_rev_b_samd21_xplained_pro.bat



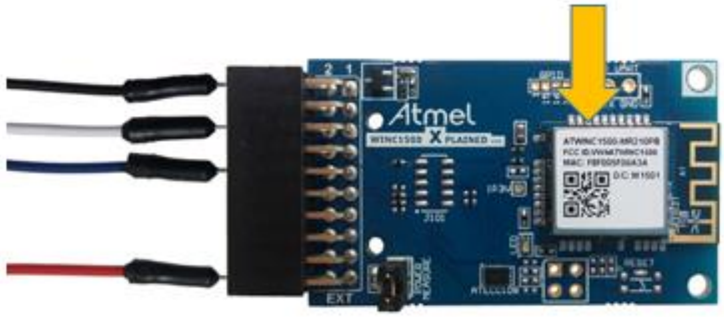
```
#Erase time = 0.374000 sec
>Start programming...
Done
#Programming time = 2.231000 sec


>>This task finished after  2.78 sec
AT Command Firmware Downloading sucessfully completed.
Start download to Test App for At command on SAMD21 Xplained Pro board.
Firmware check OK
Programming completed successfully.
OK
    #######################################################################
    ##                                                                   ##
    ##                 ########     ###     ######   ######              ##
    ##                 ##     ##   ## ##   ##    ## ##    ##             ##
    ##                 ##     ##  ##   ##  ##       ##                   ##
    ##                 ########  ##     ##  ######   ######              ##
    ##                 ##        #########       ##       ##             ##
    ##                 ##        ##     ## ##    ## ##    ##             ##
    ##                 ##        ##     ##  ######   ######              ##
    ##                                                                   ##
    #######################################################################
Test App Downloading sucessfully completed.
Press any key to continue . . .
```

2.2 Download via UART

You can download the firmware directly into the ATWINC1500 through UART interface. Refer to 3.1 UART Connection about UART interface. You should power up the ATWINC1500 correctly before performing downloading. Refer to 3.2 ATWINC1500 Power Up about power up information.

You must not send the start command before start downloading firmware via UART.

Connect UART1 to a USB port on your PC.

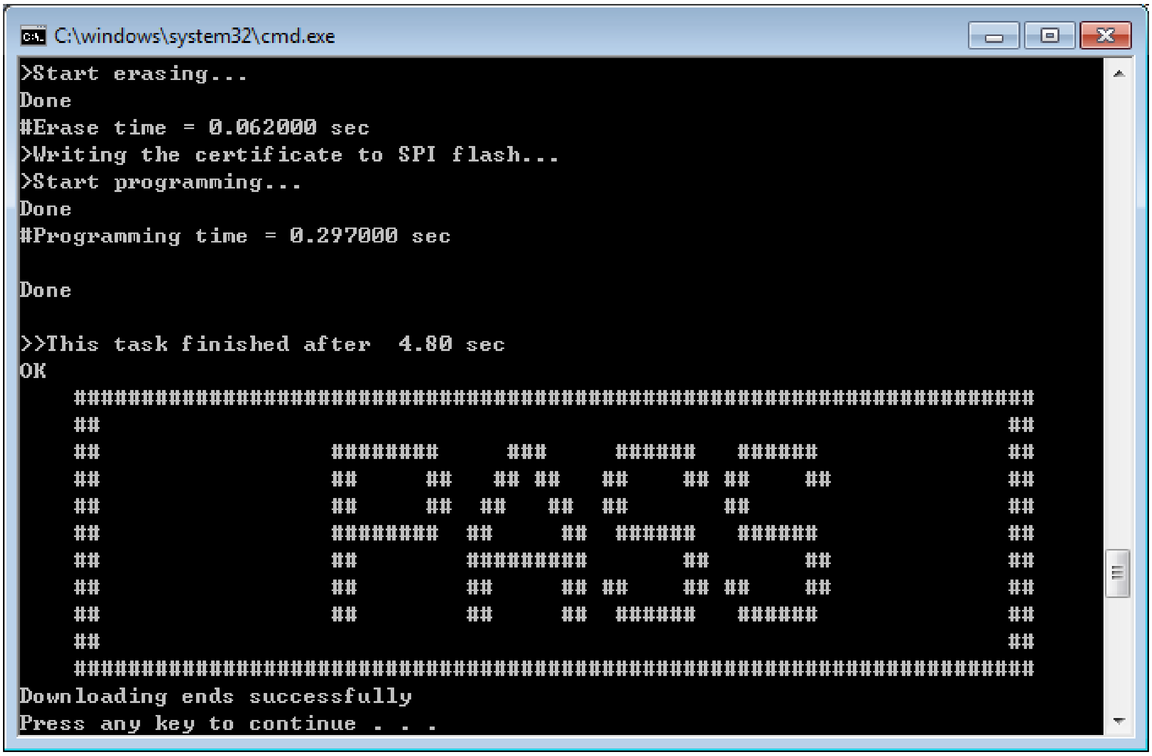


WARNING Do not connect UART1 COM port to a terminal program.

2.2.1 Download the Wi-Fi Firmware

Go to the AT command project folder and launch the batch file.

\firmware_tools\download_all_uart.bat

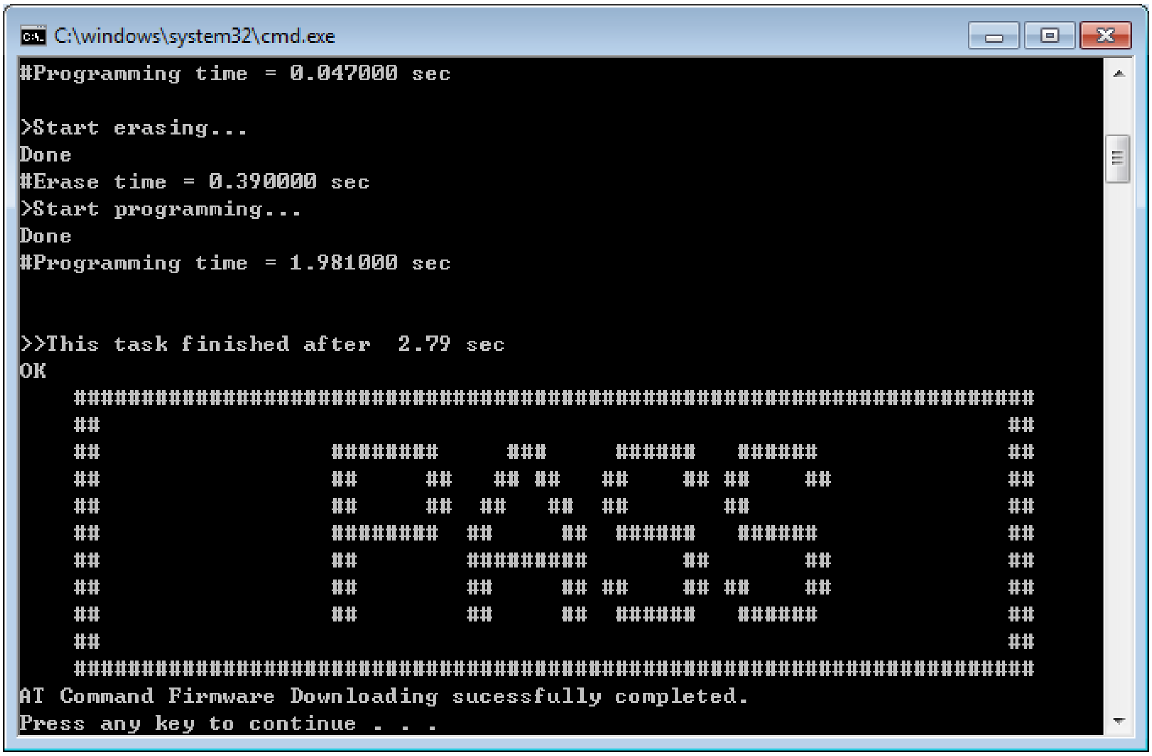


```
>Start erasing...
Done
#Erase time = 0.062000 sec
>Writing the certificate to SPI flash...
>Start programming...
Done
#Programming time = 0.297000 sec

Done

>>This task finished after  4.80 sec
OK
    ########################################################################
    ##                                                                    ##
    ##                 ########     ###     ######   ######               ##
    ##                 ##     ##   ## ##   ##    ## ##    ##              ##
    ##                 ##     ##  ##   ##  ##       ##                    ##
    ##                 ########  ##     ##  ######   ######               ##
    ##                 ##        #########       ##       ##              ##
    ##                 ##        ##     ## ##    ## ##    ##              ##
    ##                 ##        ##     ##  ######   ######               ##
    ##                                                                    ##
    ########################################################################
Downloading ends successfully
Press any key to continue . . .
```

2.2.2 Download the AT Command Firmware

Launch the batch file.

\firmware_tools\download_at_cmd_rev_b_uart.bat



```
#Programming time = 0.047000 sec

>Start erasing...
Done
#Erase time = 0.390000 sec
>Start programming...
Done
#Programming time = 1.981000 sec


>>This task finished after  2.79 sec
OK
    ########################################################################
    ##                                                                    ##
    ##                 ########     ###     ######   ######               ##
    ##                 ##     ##   ## ##   ##    ## ##    ##              ##
    ##                 ##     ##  ##   ##  ##       ##                    ##
    ##                 ########  ##     ##  ######   ######               ##
    ##                 ##        #########       ##       ##              ##
    ##                 ##        ##     ## ##    ## ##    ##              ##
    ##                 ##        ##     ##  ######   ######               ##
    ##                                                                    ##
    ########################################################################
AT Command Firmware Downloading sucessfully completed.
Press any key to continue . . .
```

3 Getting Started with the AT Command Firmware

This chapter introduces how to use the AT command firmware with the ATWINC1500 and it consists of UART connection, power up and starting the AT command firmware.

3.1 UART Connection

There are two UART interfaces on ATWINC1500B or higher revision. UART2 is used for AT command communication and UART1 is used for downloading firmware and sending debug message. UART2 supports H/W flow control mechanism and it has additional pins (RTS/CTS).

Pin assignment of UART1 is described below. Refer to Figure 3-4 for details of pin assignment. On ATWINC1500 Xplained Pro, TX and RX are available on through holes labeled “DEBUG_UART” for easy connection.

ATWINC1500 module pin name	ATWINC1500 Xplained Pro pin name	Function
J14	UART_TXD	TXD
J19	UART_RXD	RXD

Figure 3-1. ATWINC1500 UART1 interface

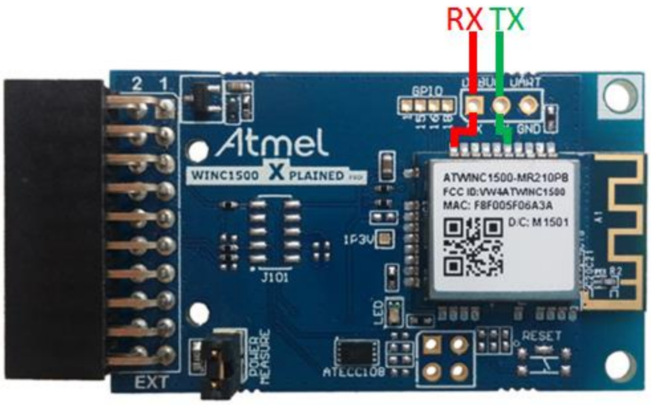
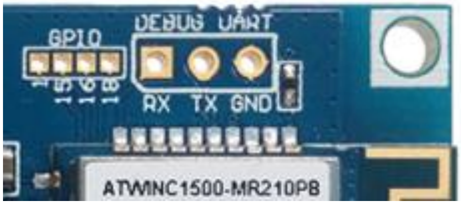


Figure 3-2. Through holes of ATWINC1500 Xplained Pro

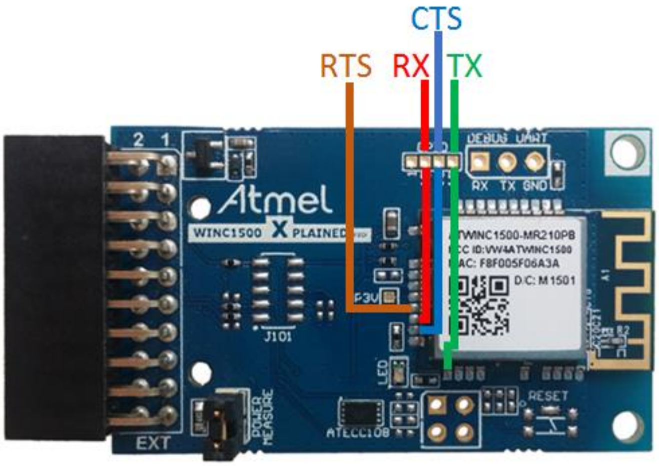


Pin assignment of UART2 is described below. Refer to Figure 3-4 for details of pin assignment. On ATWINC1500 Xplained Pro, TX is available on “GPIO_18” and RX is available on “GPIO_15” for easy connection and these are in the line of 4 through holes labeled “GPIO” next to “DEBUG_UART”.

ATWINC1500 module pin name	ATWINC1500 Xplained Pro pin name	Function
J1	GPIO_6	TXD
J26	GPIO_4	RXD

ATWINC1500 module pin name	ATWINC1500 Xplained Pro pin name	Function
J25	GPIO_3	RTS
J27	GPIO_5	CTS

Figure 3-3. ATWINC1500 UART2 interface



3.2 ATWINC1500 Power Up

3.2.1 Power Up Sequence

You can reset the ATWINC1500 by configuring some pins according to the following information. These pins should rise individually during power up.

Figure 3-4. ATWINC1500 Module Pin Assignment

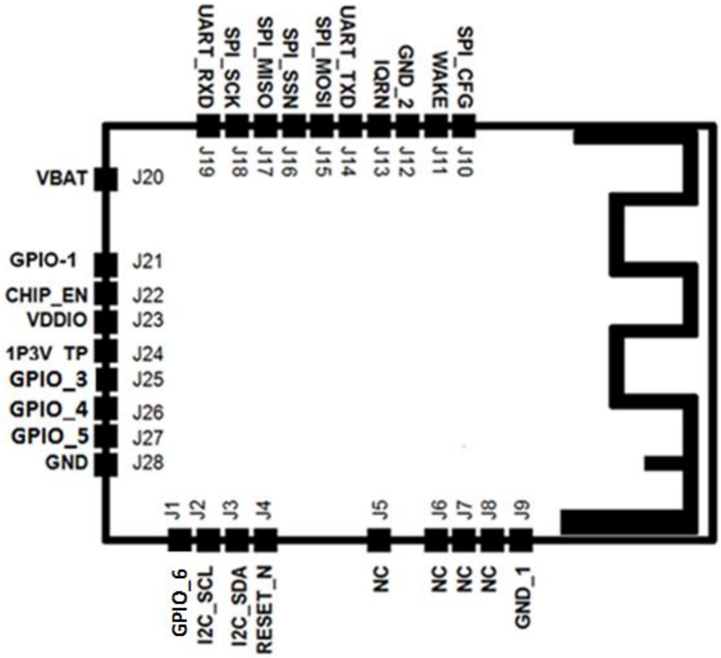
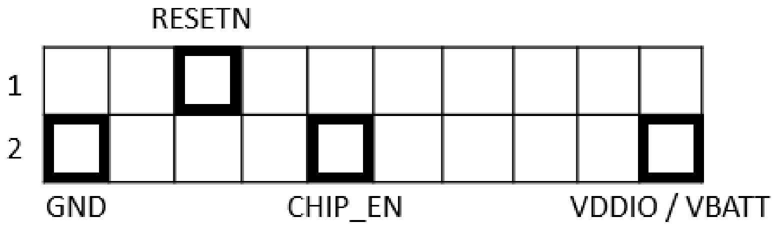


Figure 3-5. ATWINC1500 Xplained Pro 20pin header



The power up sequence for the ATWINC1500 is shown below. The timing parameters are provided in Table 3-1.

Figure 3-6. Power up sequence

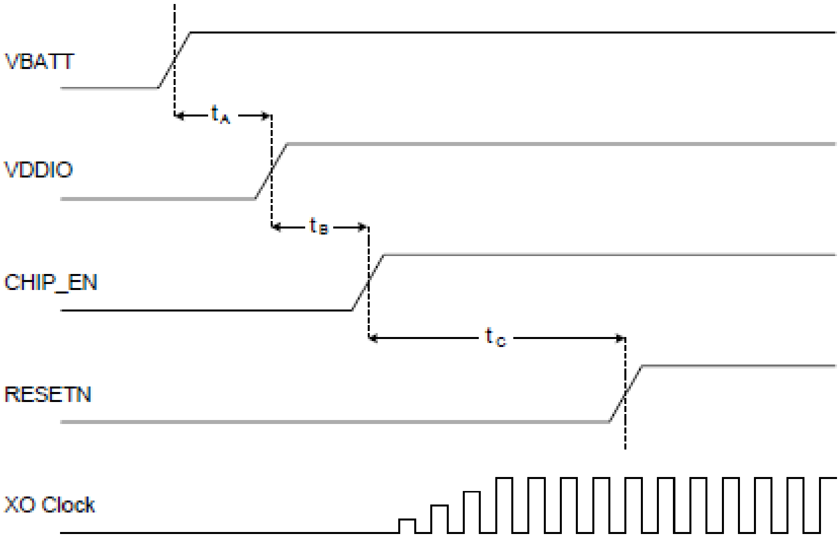


Table 3-1. Power up sequence timing

Parameter	Min	Max	Units	Description	Notes
t_A	0		ms	VBATT rise to VDDIO rise	VBATT and VDDIO can rise simultaneously or can be tied together. VDDIO must not rise before VBATT.
T_B	0		ms	VDDIO rise to CHIP_EN rise	CHIP_EN must not rise before VDDIO. CHIP_EN must be driven high or low, not left floating.
T_C	5		ms	CHIP_EN rise to RESETN rise	This delay is needed because XO clock must stabilize before RESETN removal. RESETN must be driven high or low, not left floating.

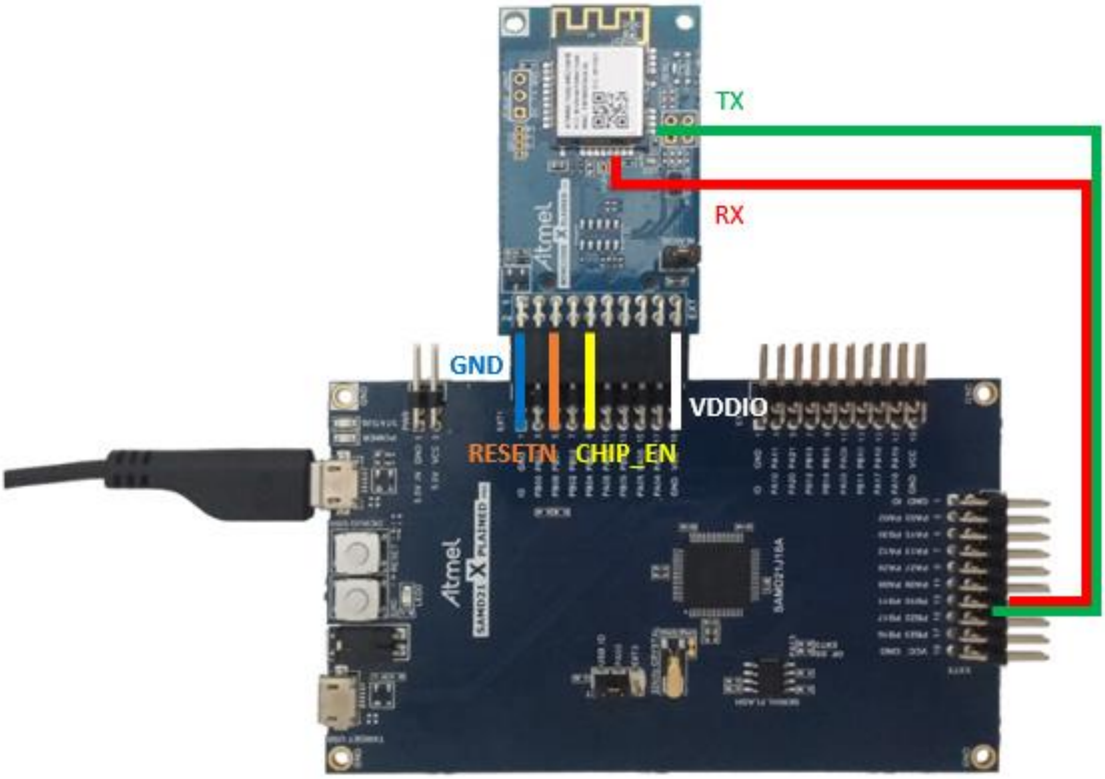
3.2.2 Power Up Example – SAMD21

If you have an SAMD21 Xplained Pro board connected to the ATWINC1500, you can reset the ATWINC1500 by setting CHIP_EN pin and RESETN pin as below.

```
void winc_bsp_reset(void)
{
        port_pin_set_output_level(CONF_WINC_PIN_CHIP_ENABLE, false);
        port_pin_set_output_level(CONF_WINC_PIN_RESET, false);
        winc_bsp_sleep(100);
        port_pin_set_output_level(CONF_WINC_PIN_CHIP_ENABLE, true);
        winc_bsp_sleep(10);
        port_pin_set_output_level(CONF_WINC_PIN_RESET, true);
        winc_bsp_sleep(100);
}
```

Figure 3-6 describes an example of H/W connection between SAMD21 Xplained Pro and ATWIC1500 Xplained Pro.

Figure 3-7. H/W connection with SAMD21 Xplained Pro



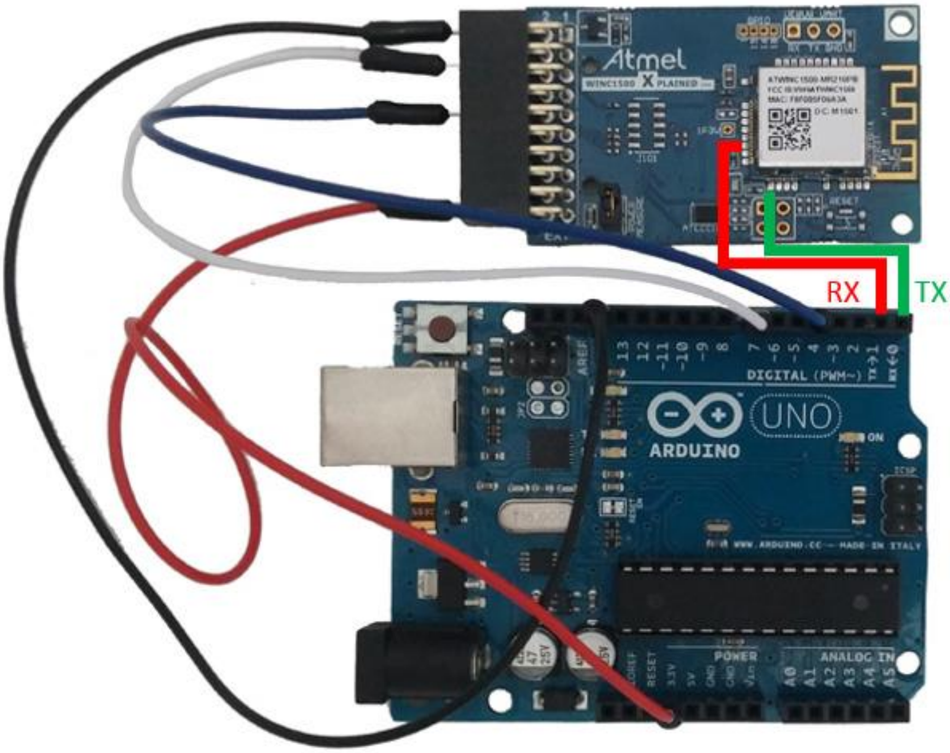
3.2.3 Power Up Example – Arduino Uno

If you have an Arduino Uno board connected to the ATWINC1500, you can reset the ATWINC1500 by setting CHIP_EN pin and RESETN pin as below.

```
void winc_bsp_reset(void)
{
        digitalWrite(4, LOW);
        digitalWrite(7, LOW);
        delay(100);
        digitalWrite(4, HIGH); // CHIP_EN
        delay(10);
        digitalWrite(7, HIGH); // RESET
        delay(100);
}
```

Figure 3-7 describes an example of H/W connection between Arduino Uno and ATWIC1500 Xplained Pro.

Figure 3-8. H/W connection with Arduino Uno

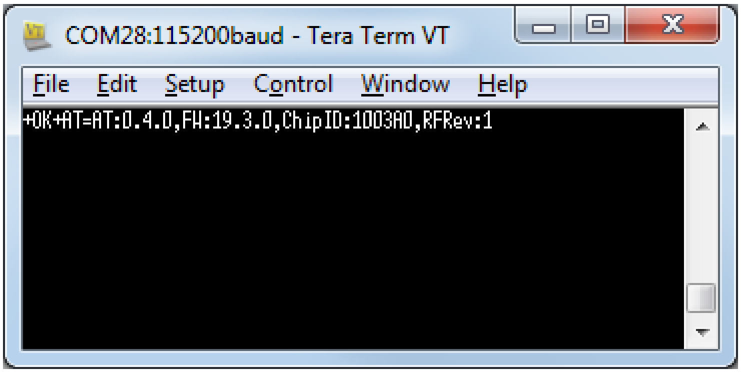


3.3 Start the AT Command Firmware

After the ATWINC1500 is reset, you need to send the start command to start the AT command firmware. For details about the start command, refer to 5.1 Start Command.

Open a terminal tool (Tera Term, Hyper Terminal and so on) and connect to UART2 COM port. Now you can operate the ATWINC1500 with AT commands with the terminal tool.

Figure 3-9. Terminal tool



```
+OK+AT=AT:0.4.0,FW:19.3.0,ChipID:1003A0,RFRev:1
```

WARNING To use H/W flow control with the terminal tool, you have to set the flow control option to “hardware”. Check your terminal tool’s settings.

4 AT Command General Format

4.1 AT Command Syntax

Commands are not case sensitive. All commands and responses except data stream commands end with <CR><LF>.

4.1.1 Command Mode

The AT commands described in this guide shall follow one of the following formats.

- Command with no parameters

  An AT command with no parameter associated with it shall take the form:

  AT+*CMD*<CR><LF>

  Where, CMD is the command name and <CR><LF> are the "\r\n".
- Command with parameters

  An AT command with parameters associated with it shall take the form:

  AT+*CMD*=*<P1>*,*<P2>*,…,*{Pn}*<CR><LF>

  Where, P1~Pn are the list of parameters corresponding to this CMD. A parameter represented with "< >" is essential while a parameter represented with "{ }" is optional and can be omitted.
- Query command syntax

  The command syntax will be returned in the response if the command name is followed by '!' like below:

  AT+*CMD*!<CR><LF>

  This is useful to check each command format.
- Query command setting value

  The current setting value of the command will be returned in the response if the command name is followed by '?' like below:

  AT+*CMD*?<CR><LF>

  Available commands for this syntax are listed below:

  - LOG
  - ECHO
  - SCANCH
  - SSCNOPT
  - SNTP
  - PSMODE
  - LSNINT
  - DHCP
  - UART

4.1.2 Data Stream Mode

In this mode, the ASCII character SOH (0x1, Start of heading) is used at the beginning and the end of command line. For details, refer to 6 Data Stream.

4.2 Response

After executing the command a response will be sent through UART interface and it shall follow one of the following cases.

- Success response

  This indicates the command was completed without any error and it shall take the form:

  +OK+*CMD=R1,R2,…,Rn*<CR><LF>

  Where, CMD is the given command name and R1~Rn are the list of return values. There can be no return value according to the command.

- Error response

  This indicates there was an error during executing the command and it shall take the form:

  +ERROR+*CMD*:*CODE*<CR><LF>

  Where, CODE is the error code.

- Event response

  Event response is sent when the AT command firmware receives a Wi-Fi event from the Wi-Fi firmware and it shall take the form:

  +EVT+*CMD=DATA*<CR><LF>

  Where, DATA is related information to the event and it can be omitted according to the event.

5 AT Command Reference

5.1 Start Command

Start the AT command firmware. This command should be sent in HEX format (not in ASCII format).

Command syntax	A5 01 F2 00 00 0C 00 0C 00 61 2F 52 EF
Response	+OK+AT=AT:<AT command firmware version>,FW:<Wi-Fi firmware version>,ChipID:<chip ID>,RFRev<RF revision ID>

WARNING Before the host application starts AT command operation it must send the start command to start the AT command firmware.

5.2 Common Commands

5.2.1 Check the AT Command Firmware

Check the state of the AT command firmware. If the AT command firmware is working then the host MCU may receive a response.

Command syntax	AT
Response	+OK+AT=Ver:<AT command firmware version>

5.2.2 Firmware Version

Get the Wi-Fi firmware version.

Command syntax	AT+VER
Response	+OK+VER=AT:<AT command firmware version>, FW:<Wi-Fi firmware version>

5.2.3 Command List

Show the list of all available AT commands with the syntax.

Command syntax	AT+LIST
Response	AT+<Command>=<parameter>,<parameter>,… … +OK+LIST=<the number of commands>

5.2.4 Configure Log State

Enable or disable AT command firmware log and Wi-Fi firmware log. These logs are used for debugging purpose and they are sent through UART1.

Command syntax	AT+LOG=<level>
Command parameters	**level** Log level.

	0(default): Disable all logs. 1: Enable AT command firmware log only. 2: Enable Wi-Fi firmware log only. 3: Enable AT command firmware log and Wi-Fi firmware log.
Response	+OK+LOG
Example	AT+LOG=1

5.2.5 Echo

Configure echo message print mode for each command line. Echo message is not printed in data stream mode.

Command syntax	AT+ECHO=<mode>
Command parameters	**mode** Echo message mode. 0(default): No echo message. 1: Print echo message when receiving <CR><LF> through UART interface. 2: Print echo message in real time.
Response	+OK+ECHO
Example	AT+ECHO=2

⚠ **WARNING**

Echo mode #2 can be used only with some terminal tools which are support real time input (e.g. Tera Term or Hyper Terminal).

5.3 Wi-Fi Commands

5.3.1 Chip ID

Get Chip ID of the Wi-Fi module.

Command syntax	AT+CHIPID
Response	+OK+CHIPID=<chip ID>,<revision ID>

5.3.2 Get IP Address

Get the current IP address. This command can be used in station mode, AP mode and provision mode.

Command syntax	AT+GETIP
Response	**Station mode** +OK+GETIP=<IP >,<subnet>,<gateway>,<DNS> **Other modes** +OK+GETIP=<IP >

5.3.3 Set IP Address

Set static IP addresses. This setting is used for Ipv4 network settings in the station mode. This command disables DHCP client.

Command syntax	AT+SETIP=<ip>,<subnet>,<gateway>,<dns>
Command parameters	**ip** Static IP address. **Subnet** Subnet IP address. **Gateway** Gateway IP address. **Dns** DNS IP address.
Response	+OK+SETIP
Example	AT+SETIP=192.168.0.22,255.255.255.0,192.168.0.1,110.253.212.23

5.3.4 Get MAC Address

Get the MAC address of the Wi-Fi module.

Command syntax	AT+GETMAC
Response	+OK+GETMAC=<MAC address>

5.3.5 Set MAC Address

Set an MAC address temporarily for the Wi-Fi module. This command can be used in the idle mode.

Command syntax	AT+SETMAC=<address>
Command parameters	**address** User MAC address.
Response	+OK+SETMAC
Example	AT+SETMAC=a0:b0:c0:d0:e0:f0

WARNING MAC address is initialized to the original value after rebooting.

5.3.6 Scan Aps

Scan all Aps and show the list with details.

Command syntax	AT+SCAN={channel}
Command parameters	**channel** Specify the channel for scan.
Response	+EVT+SCAN=<index>,<RSSI>,

	<authentication type: 1(Open)/2(WPA/WPA2)/3(WEP)4(Enterprise)>, <channel ID>,<BSSID>,<SSID> … +OK+SCAN=<the number of found Aps>

5.3.7 Set Scan Channels

Set scan channels. Wi-Fi scanning can be performed on all channels or on specific requested channels.

Command syntax	AT+SCANCH=<channel>,…
Command parameters	**channel** The channel number to scan. Plural channels can be set and 1 to 14 (CH_1 to CH_14) are available. Scan channel range should be changed according to the regulatory domain. Refer to the following information. all: Scan all channels.
Response	+OK+SCANCH
Example	AT+SCANCH=all AT+SCANCH=1,6,11

INFO

Scan channel range of regulatory domain is as below:
FCC(North America) : CH_1 to CH_11
ETSI(Europe) : CH_1 to CH_13
TELEC(Japan) : CH_1 to CH_14

5.3.8 Set Scan Option

Set channel scan option.

Command syntax	AT+SSCNOPT=<slots>,<slot time>,<probes>,<rssi>
Command parameters	**slots** The number of slots for every channel. The minimum value is 2. **Slot time** Time in millisecond that Wi-Fi module will wait on every channel listening for the frames on air. The available range is 10 ~ 250. **Probes** The number of probe requests to be sent per channel scan slot. The available range is 0 ~ 2. **Rssi** The RSSI threshold of the AP which will be connected to directly. The available range is -99 ~ -1.
Response	+OK+SSCNOPT

Example	AT+SSCNOPT=2,20,1,-50

5.3.9 RSSI

Get RSSI of the current connection. This command can't be used in the idle mode.

Command syntax	AT+RSSI
Response	+OK+RSSI=<RSSI value>

5.3.10 DNS Lookup

Get host IP address by domain name. This command can be used in the station mode.

Command syntax	AT+RESDNS=<host>
Command parameters	**host** Host domain name.
Response	+OK+RESDNS=<host domain name>,<host IP address>
Example	AT+RESDNS=www.atmel.com

5.3.11 SNTP Client

Enable/Disable the internal SNTP client.

Command syntax	AT+SNTP=<enable>
Command parameters	**enable** Enable or disable the SNTP client. 0: Disable. 1: Enable.
Response	+OK+SNTP
Example	AT+SNTP=1

INFO

The SNTP is enabled by default at start-up. The SNTP client at the Wi-Fi firmware is used to sync the system clock to the UTC time from a well-known time servers (e.g. "time-c.nist.gov"). The SNTP client uses a default update cycle of 1 day.
If there is an RTC on the host MCU, the SNTP could be disabled. After the system clock is synchronized once, the clock is counted even if the SNTP client is disabled again**.**

5.3.12 Set System Time

Set system time in the Wi-Fi module.

Command syntax	AT+SETTIME=<time>
Command parameters	**time**

	UTC value in seconds.
Response	+OK+SETTIME
Example	AT+SETTIME=1434413832

INFO System time uses an epoch of January 1, 1900 as same as NTP.

5.3.13 Get System Time

Get system time from the Wi-Fi module.

Command syntax	AT+GETTIME
Response	+OK+GETTIME=2015/06/16 03:16:27

INFO Time format in the response is “YYYY/MM/DD hh:mm:ss” (UTC).

5.3.14 Power Save Mode

Set the Wi-Fi module to power save mode.

Command syntax	AT+PSMODE=<mode>
Command parameters	**mode** Power save mode. 0: Power save mode is disabled. 1: M2M_PS_H_AUTOMATIC. Power saving is done automatically by the Wi-Fi module. All system clock and CPU of the Wi-Fi module are on but 802.11 radio, MAC and PHY are disabled. 2: M2M_PS_DEEP_AUTOMATIC. Power saving is done automatically by the Wi-Fi module. System clock, 802.11 radio, MAC and PHY are disabled.
Response	+OK+PSMODE
Example	AT+PSMODE=1

INFO Manual power save mode is not supported in the AT command firmware.

INFO Deep sleep mode (M2M_PS_DEEP_AUTOMATIC) actually operates in STA mode.

WARNING Host can wake the Wi-Fi module up from deep sleep mode by UART2 TX and deep sleep mode has a little latency to wake up. This latency can cause loss in first several UART characters and we recommend giving 100ms delay at least between wakeup and the first character of a command. This delay is needed before every command in deep sleep mode.
M2M_PS_H_AUTOMATIC mode has no latency for wakeup.

5.3.15 Set Listen Interval

Set listen interval for power saving. This interval is given in units of beacon period. Periodically after the listen interval fires, the Wi-Fi module wakes up and listens to the beacon and check for any buffered frames for it from the AP.

Command syntax	AT+LSNINT=<interval>
Command parameters	**interval** Listen interval in beacon period count.
Response	+OK+LSNINT
Example	AT+LSNINT=2

INFO Power save mode (PSMODE) shall be set before running this command**.**

WARNING This command should be called once after the Wi-Fi module is reset.

5.3.16 Ping

Send ping request to the given IP address. The result event will be returned asynchronously. This command can be used in the station mode.

Command syntax	AT+PING=<host>,{ttl}
Command parameters	**host** Domain name or IP address of the target destination for the ping request. **Ttl** IP TTL value for the ping request. Available range is 0 ~ 255. If omitted or set to 0, the default value is used.
Response	+OK+PING +EVT+PING=<Destination IP>,<Round trip time>, <Error code: 0(Success)/1(Unreached)/2(Timeout)>
Example	AT+PING=www.atmel.com AT+PING=192.168.0.100

5.3.17 OTA Firmware Upgrade

Perform over-the-air upgrade of the Wi-Fi firmware and the AT command firmware.

WARNING OTA upgrade function for AT command firmware is available only with the ATWINC1510 which has 8MB flash memory and we will provide this command later. It will returns an error response in the ATWINC1500 which has 4MB flash memory.

5.4 Mode Commands

5.4.1 DHCP Client

Enable or disable the DHCP client after connection. If this is enabled the DHCP IP address will be obtained after the ATWINC1500 is connected to an AP in the station mode. This command can be used only in the idle mode.

Command syntax	AT+DHCP=<enable>
Command parameters	**enable** Enable or disable the DHCP client. 0: Disable 1: Enable
Response	+OK+DHCP
Example	AT+DHCP=1

5.4.2 Connect

Connect to an AP. After successful connecting, the Wi-Fi service mode will be changed to the station mode.

Command syntax	AT+CONN=<ssid>,<security>,{index}/{user name},{key},{channel}
Command parameters	**ssid** Access point name. **security** Security type of connection to the AP. 1: Open 2: WPA/WPA2-Personal 3: WEP 4: WPA-Enterprise **index / user name** WEP key index(WEP) or user name(WPA-Enterprise). **Key** WEP key(WEP) or password(WPA-Personal and WPA-Enterprise). **Channel** AP listen channel.
Response	+OK+CONN
Example	AT+CONN=OPEN_SSID,1 AT+CONN=WPA_SSID,2,abcd1234 AT+CONN=WEP_SSID,3,1,abcd1234

5.4.3 Default Connection

Connect to the last successfully connected AP from cached connections. This command can be used in the idle mode.

Command syntax	AT+DEFCONN
Response	+OK+DEFCONN

5.4.4 Get Mode

Get the current Wi-Fi service mode. Return value can be one of the following modes.

- "STA": station mode
- "AP": AP mode
- "P2P": P2P mode
- "PROV": Provision mode
- "WPS": WPS mode
- "IDLE": Initialized mode

Command syntax	AT+MODE
Response	+OK+MODE=<mode>

5.4.5 Disable Mode

Disable the current Wi-Fi service mode and set to the idle mode. This command closes and initializes all sockets.

Command syntax	AT+DISMODE
Response	+OK+DISMODE

5.4.6 AP Mode

Setup AP mode. This command can be used in the idle mode. The command syntax is different between the security types.

Command syntax	**OPEN type:**
	AT+AP=<ssid>,<visibility>,<security>,{ip},{channel}
	WEP type:
	AT+AP=<ssid>,<visibility>,<security>,{key},{ip},{channel}

Command parameters	**ssid** SSID of the AP. **Visibility** AP visibility. 0: Visible 1: Hidden **security** Security type.

	1: Open(Not secured) 3: WEP **key** WEP key for the SSID. This option is available only with WEP type. **Ip** IP server address of the AP. Default is 192.168.0.1. **channel** AP listen channel. Default is 11.
Response	+OK+AP
Example	AT+AP=AP_OPEN,0,1 AT+AP=AP_WEP,0,3,68656c6c6f AT+AP=AP_WEP,0,3,hello

INFO

AP mode supports OPEN and WEP security only. The AP can only support a single associated station and further connect attempts will be rejected.

INFO

The key index of the WEP type is fixed to 1. Also, the key size of the WEP type supports 40 bits and 104 bits as ASCII character (5 or 13 characters) or hex character (10 or 26 characters).

INFO

The last address byte of IP server should 1 like 192.168.0.1 otherwise an error response will be returned. The last address byte of client IP will be 100 like 192.168.0.100.

5.4.7 P2P Mode

Setup P2P mode. This command can be used in the idle mode.

Command syntax	AT+P2P=<name>,<channel>
Command parameters	**name** Device name. **channel** P2P listen channel. Available channels are 1, 6 and 11.
Response	+OK+P2P
Example	AT+P2P=P2P_NAME,6

5.4.8 HTTP Provision

Start HTTP provision mode. The ATWINC1500 is placed in AP mode in this method and command parameters indicate ATWINC1500 AP setting.

Command syntax	**OPEN type:**

	AT+PROV=<ssid>,<visibility>,<security>,{ip},{channel}, {domain},{redirect} **WEP type:** AT+PROV=<ssid>,<visibility>,<security>, {key},{ip},{channel},{domain},{redirect}
Command parameters	**ssid** SSID of the ATWINC1500 AP. **Visibility** AP visibility. 0(default): Visible 1: Hidden **security** Security type. 1: Open(Not secured) 3: WEP **key** WEP key for the SSID. **Ip** AP IP server address. Default is 192.168.0.1. **channel** AP listen channel. Default is 11. **Domain** Provision web page domain name. This will be "prov.atmel.com" if not given. **Redirect** HTTP redirection. If enabled, all HTTP traffic will be redirected to the HTTP provisioning web page. 0: Disable redirection. 1(default): Enable redirection.
Response	+OK+PROV
Example	AT+PROV=AP_PROV,0,1 AT+PROV= AP_PROV,0,3,123456789a,192.168.0.1,1,prov.atmel.com,1

INFO

Only Aps with WPA/WPA2-Personal security (passphrase based) and no security (Open network) can be provisioned. WEP and WPA-Enterprise Aps cannot be provisioned.

INFO The last address byte of IP server should 1 like 192.168.0.1 otherwise an error response will be returned. The last address byte of client IP will be 100 like 192.168.0.100.

5.4.9 WPS

Setup WPS(Wi-Fi Protected Setup) mode. This command can be used in the idle mode.

Command syntax	AT+WPS=<trigger>,<pin>
Command parameters	**trigger** WPS trigger method. 0: PIN method. 4: Push button method. **Pin** Pin number. This is used only with PIN method.
Response	+OK+WPS **Success** +EVT+WPS=1,<ssid>,<auth type>,<psk>,<channel> **Fail** +EVT+WPS=0
Example	AT+WPS=0,1234 AT+WPS=4

5.5 Socket Commands

5.5.1 TCP Client Socket

Create a TCP client socket and connect to the remote peer.

Command syntax	AT+TCPCLI=<remote ip>,<remote port>,{flag},{auto recv}
Command parameters	**remote ip** Remote host IP address. **Remote port** Remote host port number. **Flag** Socket flag. 0(default): Normal socket. 1: Used for SSL session. **Auto recv** Receive data automatically after connection. 0: Not receive data automatically 1(default): Receive data automatically

Response	+OK+TCPCLI=<socket handler>
Example	AT+TCPCLI=216.58.216.36,80

ℹ **INFO**

ATWINC1500 supports 7 TCP sockets and 4 UDP sockets. The socket handler in the response is 1 ASCII character in hexadecimal format so it can be '0' ~ 'A'. This is same in other commands.

ℹ **INFO**

ATWINC1500 Wi-Fi firmware includes several trusted root certificates. You may need to download another root certificate to set up a TLS connection and you can refer to Atmel-42417-Getting-Started-Guide-for-AT-WINC1500WiFi-using-SAMD21-Xplained-Pro_UserGuide.pdf for details.

5.5.2 TCP Server Socket

Create a TCP server socket.

Command syntax	AT+TCPSVR=<local port>,{auto recv}
Command parameters	**local port** Local host port number. **Auto recv** Receive data automatically after binding. 0: Not receive data automatically 1(default): Receive data automatically
Response	+OK+TCPSVR=<socket handler>
Example	AT+TCPSVR=80

5.5.3 UDP Socket

Create a UDP socket.

Command syntax	AT+UDP=<local port>,{remote ip},{remote port},{auto recv}
Command parameters	**local port** Local host port number. **Remote ip** Remote host IP address. If this parameter is not given then 255.255.255.255 will be set to this parameter. **Remote port** Remote host port number. If this parameter is not given then the local host port number will be set to this parameter. **Auto recv**

	Receive data automatically after socket binding. 0: Not receive data automatically. 1(default): Receive data automatically.
Response	+OK+UDP=<socket handler>
Example	AT+UDP=8000 AT+UDP=8000,192.168.0.100,10001

5.5.4 Configure UDP Socket

Change remote IP address and remote port number of a UDP socket.

Command syntax	AT+UDPCFG=<socket>,<remote ip>,<remote port>
Command parameters	**socket** UDP socket handler to be configured. **Remote ip** Remote host IP address. **Remote port** Remote host port number.
Response	+OK+UDPCFG
Example	AT+UDPCFG=7,108.61.73.243,80

5.5.5 IP Multicast Join

Join multicast group with a UDP socket.

Command syntax	AT+ADDMS=<socket>,<ip>
Command parameters	**socket** UDP socket handler to join a multicast group. **Ip** IP address of the multicast group.
Response	+OK+ADDMS
Example	AT+ADDMS=7,224.100.100.100

5.5.6 IP Multicast Leave

Leave multicast group.

Command syntax	AT+DROPMS=<socket>
Command parameters	**socket** UDP socket handler to leave a multicast group.
Response	+OK+DROPMS

Example	AT+DROPMS=7

5.5.7 Set UDP Send Callback

Enable or disable the send event callbacks for a UDP socket. Since UDP is unreliable by default the user maybe interested (or not) in receiving a message of each data sending. Anyway the user might want to disable the send event callback for UDP socket to enhance the socket connection throughput.

Command syntax	AT+UDPCB=<socket>,<enable>
Command parameters	**socket** UDP socket handler to change its option. **Enable** Enable or disable callback. 0: Disable. 1: Enable.
Response	+OK+UDPCB
Example	AT+UDPCB=7,1

5.5.8 Socket Status

Get socket status.

Command syntax	AT+SOCSTS=<socket>
Command parameters	**socket** Socket handler to be identified.
Response	**TCP server socket** +OK+SOCSTS=<socket handler>,<0(disconnected) / 1(connected)>, T,S,<local port> **TCP client socket** +OK+SOCSTS=<socket handler>,<0(disconnected) / 1(connected)>, T,C,<remote ip>,<remote port> **UDP socket** +OK+SOCSTS=<socket handler>,<0(disconnected) / 1(connected)>, U,<local port>,<remote ip>,<remote port>, <multicast address(if set)>
Example	AT+SOCSTS=7

5.5.9 Close Socket

Close a socket.

Command syntax	AT+CLOSE=<socket>
Command parameters	**socket**

	Socket handler to be closed.
Response	+OK+CLOSE
Example	AT+CLOSE=7

5.5.10 Close All Sockets

Close all connected sockets.

Command syntax	AT+CLOSEALL
Response	+OK+CLOSEALL

5.6 HTTP Client Commands

5.6.1 Open HTTP Client

Create a socket for an HTTP client and connect to a server.

Command syntax	AT+HTTPOPEN=<url>,{port}
Command parameters	**url** Server URL. **Port** Server port number. Default value is 80.
Response	+OK+HTTPOPEN=<socket handler>
Example	AT+HTTPOPEN=www.atmel.com,80 AT+HTTPOPEN=https://www.google.com

INFO

The AT command firmware supports only one HTTP client instance**.** The previous client should be closed before opening a new client. Refer to "HTTPCLOSE" command about closing the HTTP client.

5.6.2 Configure HTTP header

Set an HTTP header field.

Command syntax	AT+HTTPCFG=<field>,<value>
Command parameters	**field** HTTP header field name to set. **Value** Value for the field.
Response	+OK+HTTPCFG
Example	AT+HTTPCFG=Accept,text/plain

5.6.3 Get HTTP header data

Get all or a specific HTTP header data.

Command syntax	AT+HTTPCFGGET={field}
Command parameters	**field** HTTP header field name to inquire. All header data will be returned if this parameter is omitted.
Response	+OK+HTTPCFGGET=text/plain
Example	AT+HTTPCFGGET=Accept

5.6.4 Delete HTTP header data

Reset all HTTP header data or delete a specific HTTP header data.

Command syntax	AT+HTTPCFGDEL={field}
Command parameters	**field** HTTP header field name to delete its data. All header data will be deleted if this parameter is omitted.
Response	+OK+HTTPCFGDEL
Example	AT+HTTPCFGDEL=Accept

5.6.5 HTTP Request

Send an HTTP request to a server.

Command syntax	AT+HTTPREQ=<method>,<uri>,{data}
Command parameters	**method** HTTP request method. Possible values are GET, POST, DELETE, PUT, OPTIONS and HEAD. **Uri** The resource URI which the request applies. **Data** Request data.
Response	+OK+HTTPREQ +EVT+RECV=… (Refer to 7.1.10 Socket Data Receive)
Example	AT+HTTPREQ=GET,/js/converter.html

5.6.6 HTTP GET Request

Send an HTTP GET request. This command provides a simple way to run HTTPOPEN command and HTTPREQ command. If an HTTP client already exists then this command will close it and create a new client. The HTTP client will be closed after receiving an HTTP response.

Command syntax	AT+HTTPGET=<url>,{port}

Command parameters	**url** Request URI. **Port** Server port number. Default value is 80.
Response	+OK+HTTPGET=<socket handler> +EVT+RECV=… (Refer to 7.1.10 Socket Data Receive)
Example	AT+HTTPGET=www.atmel.com,80 AT+HTTPGET=www.atmel.com/Images/atmel.png

5.6.7 Close HTTP Client

Close the current HTTP client socket and reset the data.

Command syntax	AT+HTTPCLOSE
Response	+OK+HTTPCLOSE

5.7 System Commands

5.7.1 Configure UART interface

Configure UART interface settings.

Command syntax	AT+UART=<id>,<baud rate>,{flow control}
Command parameters	**Id** Target UART ID to be configured. 0: UART1 1: UART2 **baud rate** Baud rate. The maximum is 921600 and default value is 115200. **Flow control** Enable or disable hardware flow control function. Default is 0. This parameter is available for UART2 only. 0: Disable. 1: Enable.
Response	+OK+UART
Example	AT+UART=0,57600 AT+UART=1,9600,1

INFO

Host may not receive the response because UART setting was changed. Configure the host UART settings and reconnect to the ATWINC1500.

6 Data Stream

The host application can request to send or receive data through a socket. For these operation, the AT command firmware provides the data stream mode and this has a different format from normal AT commands. The ASCII character SOH (0x1, Start of heading) is used as an indicator for the data stream at both the beginning and the end of the command line. Data stream sequence starts with <SOH> and ends with <SOH>E. <CR> and <LF> are not necessary at the end of the data stream command line.

WARNING When unexpected data loss on UART occurs during sending stream command AT command firmware may keep waiting for the end of stream command even though host sends another command with <CR> and <LF>. Host can escape this situation by sending <SOH>E again.

6.1 Send Data

Request to send data through a socket. The ASCII character 'S' is used as the directive of this method.

EVT response indicates the command has been executed and shows the result (success or failure) of socket operation while ERROR response indicates command wasn't executed due to an error.

Command syntax	<SOH>**S**<socket handler><length><data><SOH>E
Command parameters	**socket handler** Socket handler to send data. 1 ASCII character in hexadecimal format. Available values are 0 ~ A. **length** The length of data to be sent. 4 ASCII characters in decimal format. Maximum length is 1400 bytes. **Data** Data stream to be sent.
Response	+EVT+SEND=… (Refer to 7.1.9 Socket Data Send) +ERROR+SEND=<error code>
Example	<SOH>S10005Hello<SOH>E <SOH>SA0010abcdefghij<SOH>E

There is another method to specify address and port number of the remote host when sending data through a UDP socket. This is not available on a TCP socket. The address and the port number are one-time parameters and not saved to the configuration of the UDP socket. The ASCII character 'U' is used as the directive of this method.

Command syntax	<SOH>**U**<socket handler><length>,<remote ip>,<remote port>, <data><SOH>E
Command parameters	**socket handler** UDP socket handler to send data. 1 ASCII character in hexadecimal format. Available values are 7 ~ A. **length**

	The length of data to be sent. 4 ASCII characters in decimal format. Maximum length is 1400 bytes. **Remote ip** Remote host IP address. **Remote port** Remote host port number. **Data** Data stream to be sent.
Response	+EVT+SEND=… (Refer to 7.1.9 Socket Data Send) +ERROR+SEND=<error code>
Example	<SOH>U70005,192.168.0.100,8000,Hello<SOH>E

6.2 Receive Data

Request to receive data through a socket. The ASCII character ‘R’ is used as the directive of this method. Received data is followed by the ASCII character ETB (0x17, End of transmit block) in the response.

EVT response indicates the command has been executed and shows the result (success or failure) of socket operation while ERROR response indicates command wasn’t executed due to an error.

Command syntax	<SOH>**R**<socket handler><SOH>E
Command parameters	**socket handler** Socket handler to receive data. 1 ASCII character in hexadecimal format. Available values are 0 ~ A.
Response	+EVT+RECV=… (Refer to 7.1.10 Socket Data Receive) +ERROR+RECV=<error code>
Example	<SOH>R1<SOH>E

6.3 Data Stream Error Response

UART interface can cause data loss and this is especially important in the data stream mode because a lot of data is continuously transferred. If there is an error in the stream data, an error response will be sent to notify the host MCU of the error. Related error responses are described below.

- Header error
  Command header of the data stream mode has socket handler parameter and data length parameter.
  If socket handler parameter or data length parameter has an invalid format, below error response will be sent.
  +ERROR+STREAM:-109
- Data length error
  If total data length received through UART is different from the value of data length parameter, below error response will be sent.
  +ERROR+STREAM:-110

7 Response

7.1 Event Response

An event response is sent when a Wi-Fi event is received. You can handle these events according to the host application's state.

7.1.1 Connected

This event indicates the ATWINC1500 is connected to a remote device in AP, P2P and provision mode.

Syntax	+EVT+CONN=<ip>
Parameters	**ip** IP address for the remote device.

7.1.2 Disconnected

This event indicates the ATWINC1500 is disconnected from an AP or a remote device.

Syntax	+EVT+DISCONN

7.1.3 DHCP

This event sends dynamic IP addresses obtained from an AP.

Syntax	+EVT+DHCP=<ip>,<subnet>,<gateway>,<dns>
Parameters	**ip** IP address assigned to the device. **Subnet** Subnet mask for the local area network. **Gateway** IP of the Default internet gateway. **Dns** IP for the DNS server.

7.1.4 Scan Result

This event sends AP's information which is found by "AT+SCAN" command. For details, refer to SCAN command.

7.1.5 Ping Result

This event sends the result of "AT+PING" command. For details, refer to PING command.

7.1.6 Provision Result

This event sends the result of HTTP provisioning.

Syntax	**Success:** +EVT+PROV=1,<ssid>,<security>

	Fail: +EVT+PROV=0
Parameters	**ssid** SSID of the provisioned AP. **Security** Security type of the provisioned AP.

7.1.7 WPS Result

This event indicates WPS operation is completed.

Syntax	+EVT+WPS=<result>,<ssid>,<security>,<psk>,<channel>
Parameters	**result** The result of obtaining the target AP's information. 0: Fail. Following parameters are omitted. 1: Success. **Ssid** SSID of the target AP. **Security** Security type of the target AP. 1: Open 2: WPA/WPA2-Personal **psk** PSK obtained from the target AP. **Channel** Channel index of the target AP.

7.1.8 Socket Accept

This event indicates a server socket accepted a client socket connection.

Syntax	+EVT+ACCEPT=<socket>,<ip>,<port>,<af>
Parameters	**socket** Client socket handler. **Ip** Remote device's IP address **port** Port number of the client socket. **Af** Address family. The only supported value for this is AF_INET (2).

7.1.9 Socket Data Send

This event shows the result of sending data through a socket.

Syntax	**Success:** +EVT+SEND=<socket handler>,<sent size> **Failure:** +EVT+SEND=<socket handler>,<error code>
Parameters	**socket handler** Socket handler. **Sent size** Sent data size. **Error code** Error reason.

7.1.10 Socket Data Receive

This event shows the result of receiving data through a socket.

Syntax	**TCP socket success:** +EVT+RECV=<socket handler>,<received size>,T, <received data><ETB> **UDP socket success:** +EVT+RECV=<socket handler>,<received size>,U, <remote address>,<remote port>,<received data><ETB> **Failure:** +EVT+RECV=<socket handler>,<error code>
Parameters	**socket handler** Socket handler. **Received size** Received data size. **Received data** Received data. **Remote address** IP address of the remote host. **Remote port** Port number of the remote host. **Error code** Error reason.

⚠ **WARNING** We don’t recommend sending other command during receiving data because it can cause command response confusion.

7.2 Error Response

This section explains response error codes and each meaning.

7.2.1 Common Error Code

These error codes indicate system operation errors or AT command operation errors.

- -100: It needs to reboot the host.
- -101: Firmware mismatch.
- -102: The command is not registered.
- -103: Wrong number of parameters.
- -104: The command operation is not available in the current state.
- -105: Invalid argument.
- -106: Command parameter buffer overflow.
- -107: System error.
- -108: Cannot find any AP for scanning or connecting.
- -109: Invalid stream header in data stream command.
- -110: Invalid stream data. Some data might be lost on UART transfer.
- -111: Command processing time-out.
- -112: NV memory operation error.
- -113: UART TX buffer overflow.

7.2.2 Wi-Fi Error Code

These error codes indicate wrong parameters or error results of Wi-Fi function.

- -200: Invalid SSID for configuring AP mode.
- -201: Invalid channel number for configuring AP mode.
- -202: Invalid DHCP server for configuring AP mode.
- -203: Invalid key index for configuring AP mode.
- -204: Invalid key size for configuring AP mode.
- -205: Invalid authentication type for configuring AP mode. Only OPEN mode and WEP mode are supported.
- -206: Wi-Fi is not connected yet.
- -207: Cannot find a domain with DNS lookup.
- -250: Fail to perform the scan operation.
- -251: Fail to join the BSS.
- -252: Fail to authenticate with the AP.
- -253: Fail to associate with the AP.
- -254: Another connection request in progress.

7.2.3 Socket Error Code

These error codes indicate wrong parameters or error results of socket function.

- -300: Invalid socket handler.
- -301: Invalid socket address.
- -302: Socket address is already in use so cannot bind on the given address.
- -303: Exceed the maximum number of TCP sockets.
- -304: Exceed the maximum number of UDP sockets.

- -306: Invalid socket argument
- -307: Exceed the maximum number of TCP passive listening sockets.
- -309: The requested socket operation is not valid in the current socket state.
- -311: Destination address is required.
- -312: The socket is closed by the peer.
- -313: The socket operation has timed out.
- -314: No buffer space available to be used for the requested socket operation.

7.2.4 Process Error Code

These error codes indicate Wi-Fi firmware operation failure.

- -400: Fail to operate connection.
- -401: Fail to operate disconnection.
- -402: Fail to operate default connection.
- -403: Fail to enable the DHCP client.
- -404: Fail to disable the DHCP client.
- -405: Fail to set the static IP address.
- -406: Fail to enable AP mode.
- -407: Fail to disable AP mode.
- -408: Fail to enable P2P mode.
- -409: Fail to disable P2P mode.
- -410: Fail to enable provision mode.
- -411: Fail to disable provision mode.
- -412: Fail to enable WPS mode.
- -413: Fail to disable WPS mode.
- -414: Fail to set the device name.
- -415: Fail to request scanning.
- -416: Fail to get scan result.
- -417: Fail to set scan region.
- -418: Fail to set scan options.
- -419: Fail to request RSSI.
- -420: Fail to set power saving mode.
- -421: Fail to set Wi-Fi listen interval.
- -422: Fail to get system time.
- -423: Fail to set system time.
- -424: Fail to enable SNTP.
- -425: Fail to disable SNTP.
- -426: Fail to set TX power.
- -427: Fail to set RX power.
- -428: Fail to set battery power.
- -429: Fail to resolve the host name.
- -430: Fail to send ping request.
- -480: Fail to switch Wi-Fi firmware during OTA.
- -481: Fail to rollback firmware during OTA.
- -482: Fail to start OTA operation.

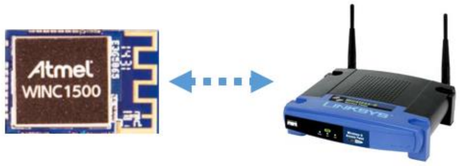
- -483: OTA generic fail.
- -484: Invalid or malformed download URL for OTA.
- -485: Invalid rollback image for Wi-Fi firmware.
- -486: Flash size on device is not enough for OTA.
- -487: OTA operation is already enabled.
- -488: OTA firmware upgrade is in progress.
- -489: OTA verification fail.
- -490: OTA connection error.
- -491: OTA server error (file not found or else).

8 Use Cases

8.1 Check AT Command

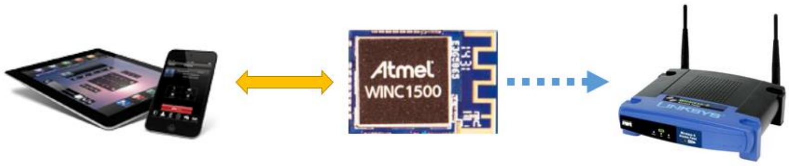
1. [SEND]Check WINC: → AT
2. [WAIT]ATWINC1500 response: ← +OK+AT=Ver:0.1.0

8.2 Default Connection



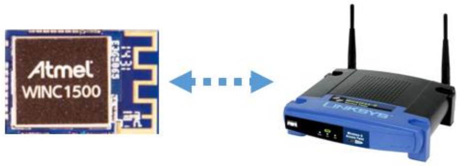
1. [SEND]Default connection: → AT+DEFCONN
2. [WAIT]Connection response: ← +OK+DEFCONN
3. [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100

8.3 HTTP Provision



1. [SEND]Setup provision mode: → AT+PROV=AP_PROV,0,1
2. [WAIT]Setup response: ← +OK+PROV
3. [WAIT](Connected to a mobile phone) Connection event: ← +EVT+CONN=192.168.0.100
4. [WAIT](Setup provision information) Provision response: ← +OK+PROV=DEMO_AP,2,123456789a
5. [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.66

8.4 Wi-Fi Protected Setup (WPS)



1. [SEND]Setup WPS mode: → AT+WPS=4
2. [WAIT]Setup response: ← +OK+WPS
3. [WAIT]WPS event: ← +EVT+WPS=DEMO_AP,2,12345678,4
4. [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100

8.5 TCP Client Socket

Atmel
WINC1500

1. [SEND]Connect to an AP: → AT+CONN=DEMO_AP,2,12345678
2. [WAIT]Connection response: ← +OK+CONN

   [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100
3. [SEND]Create a TCP client socket: → AT+TCPCLI=206.190.36.45,80
4. [WAIT]Socket response: ← +OK+TCPCLI=0
5. [SEND]Send data to the remote device: → <SOH>S00005HELLO<SOH>E
6. [WAIT]Send data response: ← +EVT+SEND=0,5

8.6 TCP Server Socket

Atmel
WINC1500

1. [SEND]Connect to an AP: → AT+CONN=DEMO_AP,2,12345678
2. [WAIT]Connection response: ← +OK+CONN
3. [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100
4. [SEND]Create a TCP server socket: → AT+TCPSVR=8000
5. [WAIT]Socket response: ← +OK+TCPSVR=0
6. [WAIT]Socket accept event: ← +EVT+ACCEPT=1,192.168.0.6,64305,2
7. [SEND]Send data to the remote device: → <SOH>S10005HELLO<SOH>E
8. [WAIT]Send data response: +EVT+SEND=1,5
9. [WAIT]Receive data event: ← +EVT+RECV=1,10,T,abcdefghij<ETB>

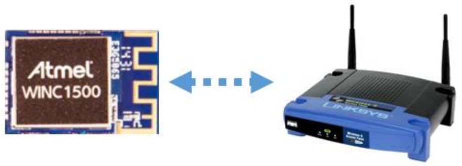
8.7 UDP Socket

Atmel
WINC1500

1. [SEND]Connect to an AP: → AT+CONN=DEMO_AP,2,12345678
2. [WAIT]Connection response: ← +OK+CONN
3. [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100
4. [SEND]Create a UDP server socket: → AT+UDP=8000

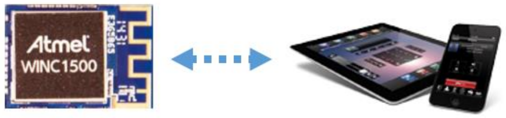
5. [WAIT]Socket response: ← +OK+UDP=7
6. [WAIT]Receive data event: ← +EVT+RECV=7,10,U,192.168.0.123,53107,abcdefghij<ETB>
7. [SEND]Send data to the remote device: → <SOH>S70005HELLO<SOH>E
8. [WAIT]Send data response: ← +EVT+SEND=7,5

8.8 Mode Change



1. [SEND]Setup AP mode: → AT+CONN=DEMO_AP,2,12345678
2. [WAIT]Connection response: ← +OK+CONN
3. [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100
4. [SEND]Disable AP mode: → AT+DISMODE
5. [WAIT]Disable mode response: ← +OK+DISMODE
6. [SEND]Setup P2P mode: → AT+P2P=P2P_TEST,6
7. [WAIT]P2P response: ← +OK+P2P

8.9 P2P Connection



1. [SEND]Setup P2P mode: → AT+P2P=P2P_TEST,6
2. [WAIT]P2P response: ← +OK+P2P
3. [WAIT](Connected to a mobile phone) Connection event: ← EVT+CONN=192.168.49.176

8.10 HTTP Client

Atmel
WINC1500

1. [SEND]Connect to an AP: → AT+CONN=DEMO_AP,2,12345678
2. [WAIT]Connection response: ← +OK+CONN

   [WAIT]DHCP event: ← +EVT+DHCP=192.168.0.100
3. [SEND]Open HTTP client connection: → AT+HTTPOPEN=www.atmel.com
4. [WAIT]HTTP socket connection response: ← +OK+HTTPOPEN=0
5. [SEND]HTTP request: → AT+HTTPREQ=GET,/
6. [WAIT]Receive data: ← +EVT+RECV=0,1446,T,<data><ETB>…

9 Revision History

Doc Rev.	Date	Comments
XXXXX	05/2015	Initial document release.
XXXXX	07/2015	Release ATWINC1500 AT command firmware version 0.4.0. - Wi-Fi firmware version 19.3.0.
XXXXX	09/2015	Release ATWINC1500 AT command firmware version 0.6.0. - Wi-Fi firmware version 19.4.0. - Change response error codes.
XXXXX	09/2015	Release ATWINC1500 AT command firmware version 0.6.1. - Change UART interface configuration.
XXXXX	09/2015	Release ATWINC1500 AT command firmware version 0.6.5. - Integrate UDPSVR and UDPCLI into UDP command.
XXXXX	09/2015	Release ATWINC1500 AT command firmware version 0.7.1. - Remove FWUG command. - ADD ‘U’ command option in data stream mode.
XXXXX	09/2015	Release ATWINC1500 AT command firmware version 0.7.2. - Remove BATPWR command.
XXXXX	10/2015	Release ATWINC1500 AT command firmware version 0.7.6.
XXXXX	10/2015	Release ATWINC1500 AT command firmware version 0.8.0. - Wi-Fi firmware version 19.4.2. - Enable the start command. - Add HTTPGET command.
XXXXX	10/2015	Release ATWINC1500 AT command firmware version 0.8.2. - Wi-Fi firmware version 19.4.3. - Modify the parameter of LOG command. - Remove TXPWR and RXPWR commands.
XXXXX	10/2015	Release ATWINC1500 AT command firmware version 0.8.3. - Add details about UART through holes on ATWINC1500 Xplained Pro.
XXXXX	11/2015	Release ATWINC1500 AT command firmware version 0.8.4. - Remove WEP key index parameter in AP command and PROV command.
XXXXX	12/2015	Release ATWINC1500 AT command firmware version 0.8.6. - Remove IP parameter in DROPMS command.
XXXXX	12/2015	Release ATWINC1500 AT command firmware version 0.8.7. - Add warnings for incompatible firmware versions (2 Download the Firmware), wakeup in deep sleep mode (5.3.14 Power Save Mode) and stream command (6 Data Stream).
XXXXX	01/2016	Release ATWINC1500 AT command firmware version 0.8.8. - Add description about server/client IP address for AP and PROV commands.
XXXXX	03/2016	Release ATWINC1500 AT command firmware version 0.9.1. - Update description about ssid for AP command.

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**

DISCLAIMER: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. EXCEPT AS SET FORTH IN THE ATMEL TERMS AND CONDITIONS OF SALES LOCATED ON THE ATMEL WEBSITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS AND PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and products descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

SAFETY-CRITICAL, MILITARY, AND AUTOMOTIVE APPLICATIONS DISCLAIMER: Atmel products are not designed for and will not be used in connection with any applications where the failure of such products would reasonably be expected to result in significant personal injury or death (“Safety-Critical Applications”) without an Atmel officer's specific written consent. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Atmel products are not designed nor intended for use in military or aerospace applications or environments unless specifically designated by Atmel as military-grade. Atmel products are not designed nor intended for use in automotive applications unless specifically designated by Atmel as automotive-grade.