Porting Guide From eDSoft V3.10 to WIPSoft V3.11

Revision: **007**
Date: **December 17, 2007**

OPEN AT

wavecom
Make it wireless

Operating Systems | Integrated Development Environments | Plug-Ins | Wireless CPUs | Services

Porting Guide From eDSoft V3.10 to WIPSoft V3.11

Reference: WM_DEV_OAT_UGD_027
Revision: 007
Date: December 17, 2007

Trademarks

 ®,WAVECOM®, WISMO®, Open AT® and certain other trademarks and logos appearing on this document, are filed or registered trademarks of Wavecom S.A. in France or in other countries. All other company and/or product names mentioned may be filed or registered trademarks of their respective owners.

Copyright

Overview

The aim of this document is to provide Wavecom customers with a full description of the APIs associated with the Open AT® IP Connectivity library.

Document History

Level	Date	History Of the Evolution	Writer
001	May 22 2006	Creation	Wavecom
002	August 7 2006	Preliminary	Wavecom
003	January 12 2007	Final	Wavecom
004	April 20 2007	Update for WIP Soft V2.02	Wavecom
005	June 06 2007	Update for WIP Step 4	Wavecom
006	October 10 2007	Update for WIP Soft V3.01	Wavecom
007	December 17 2007	Update for WIP Soft V3.11	Wavecom

Contents

1 Introduction

1.1 Related Documents

[1] WIP Open AT IP Connectivity Development Guide (ref WM_DEV_OAT_UGD_021).

[2] AT Commands Interface Guide for IP Connectivity (eDSoft V3.10) (ref WM_ASW_OAT_UGD_011 revision 007).

[3] WIP Open AT Commands User Guide (WM_DEV_OAT_UGD_024)

1.2 Abbreviations

APN	Access Point Name
BCC	Blind Carbon Copy
CC	Carbon Copy
CMUX	Converter Multiplexer
DLE	Escape character having a hex value 0x10
DNS	Domain Name Server
ETX	Escape character having a hex value 0x03.
FTP	File Transfer Protocol
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HTTP	Hyper Text Transfer Protocol
IP	Internet protocol
ISP	Internet Service Provider
OSI	Open System Interconnection
POP	Post Office Protocol
PPP	Point to Point Protocol
PSTN	Public Switched Telephone Network
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol
UART	Universal Asynchronous Receiver Transmitter

1.3 Glossary

AT Mode	The functioning mode of UART, in which anything that is received from UART is treated as AT command.
Bearer	The term used in WIPSoft. It is used to indicate the layer providing the actual transmission of data from one peer to another
Data Mode	The functioning mode of UART, in which everything that is received from the UART is treated as data
eDSoft	An Open AT® application (providing the IP connectivity function to the Wireless CPU®) written using the eDLib IP library
UART	Universal Asynchronous Receiver Transmitter
WIPSoft	An Open AT® application (providing the IP connectivity function to the Wireless CPU®) written using the Wavecom IP library
+++	The escape sequence surrounded with 1 second delay which is used to switch the state of UART from data mode to AT mode

2 AT Commands

This chapter lists all the commands that have been introduced with the WIPSoft along with their usage and functionality. This chapter provides comparison between the commands available in eDSoft and WIPSoft.

2.1 Introduction

The WIPSoft application provides a more consistent interface to the user. The WIPSoft uses the APIs provided by WIPLib and provides custom AT command interface to the external application. This is an Open AT® application that implements the TCP/IP protocols using custom AT commands. This Open AT® application operates in co-operative mode and must be downloaded to the Wavecom Wireless CPU®. The commands are sent from an external application and the corresponding responses are sent back from the Wavecom Wireless CPU® to the external application.

The WIPSoft application maintains a set of protocol identifiers for supported protocols. These identifiers along with the protocol name are listed below in the table:

Protocol Identifier	Protocol
1	UDP
2	TCP client
3	TCP server
4	FTP
5	HTTP
6	SMTP
7	POP3

2.2 TCP/IP Services

2.2.1 TCP/IP Services in eDSoft

The eDSoft application supports multiple services to run at the same time. However, only one instance of a particular service could run at a time. The only exception to this rule is TCP socket service, where 2 sockets could be open at a time. Multiplexing of various services is done using the commands which are used to manipulate the service being used.

2.2.2 TCP/IP Services in WIPSoft

The WIPSoft allows concurrent execution of many services like TCP, UDP and FTP. However, the number of sockets for TCP and UDP and the number of sessions for FTP are limited. At a time, WIPSoft supports the following:

Protocol	Number of Sockets/Sessions
UDP	8
TCP client	8
TCP server	4
FTP session	1
HTTP session	1
SMTP session	1
POP3 session	1

Multiplexing between various services is achieved using the commands which are used to manipulate the service.

2.3 Error Codes

2.3.1 Error Codes in eDSoft

The eDSoft application provides the errors in the following format:

```
#CME ERROR: <Error Code>
```

The Error Code can take values from 34817 to 49158. The following table depicts the description of various error codes.

Error Code	Description of Error code
34817	Bad command: Unknown command
34819	Bad command: Syntax error
34824	Bad command: Write failed
34881	Bad command: Command too long
34882	Bad command: Bad command argument value
34883	Bad command: High level internet configuration only command
35840	Physical layer: Modem is already running
35841	Physical layer: GPRS connection lost
35862	Physical layer: Timeout, no activity on network connection
35865	Physical layer: Module is not attached to the network
35866	Physical layer: Invalid event during activation process
35867	Physical layer: Physical layer connection is currently not active
35868	Physical layer: GPRS connection aborted
35869	Physical layer: Invalid incoming call type
35870	Physical layer: Incoming call CLI not provided
35871	IP Connectivity library: SIM removed
36872	IP Connectivity library internal error: internal resource unavailable
36929	IP Connectivity library: Bad parameter configuration attempt
37120	IP Connectivity library: PPP negotiation failed (client configuration)
37121	IP Connectivity library: PPP negotiation failed (server configuration)

Error Code	Description of Error code
37122	IP Connectivity library: Another internal application is already running
37123	IP Connectivity library: Service is running. Unable to set parameter
37952	Distant: TCP session closed (TCP Context cancelled)
37964	Distant: No response from server
37966	Distant: TCP session closed by peer (FIN received from peer)
38016	Distant: Open session attempt failed
38017	Distant: Data send attempt failed
38018	Distant: Close session attempt failed
38022	Distant: Change Directory attempt failed
38023	Distant: File deletion attempt failed
38024	Distant: Data retrieve attempt failed
38025	Distant: Email retrieve attempt failed
38026	Distant: Email header receive failed
38027	Distant: No answer from DNS servers or the domain name resolution could not be completed by the server
38028	Distant: Sender email address rejected by server
38029	Distant: Recipient email address rejected by server
38030	Distant: CC Recipient email address rejected by server
38031	Distant: Email body send request rejected by server
38080	Distant: Username rejected by server
38081	Distant: Password rejected by server
38980	IP Connectivity library: PPP timeout (client configuration)
38981	IP Connectivity library: PPP timeout (server configuration)
49153	Internal error: Open data flow request failed
49154	Internal error: Close data flow request failed
49155	Internal error: Open GPRS session request failed
49156	Internal error: GPRS authentication failed
49157	Internal error: GPRS get IPCP information request failed

Error Code	Description of Error code
49158	Internal error: Open flow confirmation not received

2.3.2 Error Codes in WIPSoft

The WIPSoft application provides the error codes in the standard AT response format. Hence, if +CME error code generation is not enabled, a simple "ERROR" message is returned. In case, the +CME ERROR messages are enabled using +CMEE=1 command, the error codes takes the following format:

```
+CME ERROR: <Error Code>
```

The error code can have values from 800 to 846, 850, 860 to 868 and 880 to 884. The following table depicts the description of various error codes.

Error Code	Description of Error code
800	Invalid option specified
801	Invalid option value
802	Not enough memory
803	Operation not allowed in current stack state
804	Device already open
805	Network interface not available
806	Operation not allowed on the selected bearer
807	Bearer connection failure: line busy
808	Bearer connection failure: no answer
809	Bearer connection failure: no carrier
810	Bearer connection failure: no SIM card present
811	Bearer connection failure: SIM not ready (SIM PIN not given)
812	Bearer connection failure: GPRS network failure
813	Bearer connection failure: PPP LCP negotiation failed
814	Bearer connection failure: PPP authentication failed
815	Bearer connection failure: PPP IPCP negotiation failed
816	Bearer connection failure: PPP peer has terminated the session

Error Code	Description of Error code
817	Bearer connection failure: PPP peer not answering to echo requests
818	Incoming call refused
819	Error on Ping channel
820	Error writing configuration in FLASH memory
821	Error reading configuration in FLASH memory
822-829	Reserved for future use
830	Bad index
831	Bad state
832	Bad port number
833	Bad port state
834	Not implemented
835	Option not supported
836	Memory
837	Bad protocol
838	No more free socket
839	Error during channel creation
840	UDP/TCP socket or FTP/HTTP/SMTP/POP3 session is already active
841	Peer closed
842	Destination host unreachable (whether host unreachable, Network unreachable, response timeout)
843	Connection reset by peer
844	Stack already started
845	Attempt is made to reserve/create a client socket which is already reserved/opened by TCP server/client
846	Internal error: FCM subscription failure
847-849	Reserved for future use
850	Unknown reason
851-859	Reserved for future use

Error Code	Description of Error code
860	Protocol undefined or internal error
861	Username rejected by server
862	Password rejected by server
863	Protocol delete error
864	Protocol list error
865	Authentication error
866	Server not ready error
867	POP3 email retrieving error
868	POP3 email size error
869-879	Reserved for future use
880	SMTP sender email address rejected by server
881	SMTP recipient email address rejected by server
882	SMTP CC recipient email address rejected by server
883	SMTP BCC recipient email address rejected by server
884	SMTP email body send request rejected by server

2.4 Configuration Commands

The WIPSoft application allows better control of IP stack. This is achieved using a set of configurable options available with various WIP AT commands.

2.4.1 IP Stack Initialization and Termination (New)

2.4.1.1 Description

The WIPSoft application provides +WIPCFG command for IP stack initialization and termination. The eDSoft application does not support any initialization command. In eDSoft, the IP stack is initialized when application starts.

2.4.1.2 Syntax

AT+WIPCFG=<Option>

2.4.1.3 Defined Values

Option:

<Option>	Description
1	Start IP stack
0	Stop IP stack

2.4.1.4 Examples

Commands	Possible responses
AT+WIPCFG=1 *Note: Start the IP stack*	OK
AT+WIPCFG=1 *Note: Start the IP stack again*	+CME ERROR: 844 *Note: IP stack already started*
AT+WIPCFG=0 *Note: Stop the IP stack*	+CME ERROR: 802 *Note: Stop procedure failed.*

2.4.2 IP Stack Configuration (New)

The WIPSoft application provides command to configure the values for internal parameters being used by the IP stack. These parameters include

- TTL (Time To Live of IP datagram)
- TOS (Type of Service)
- IP fragment timeout and so on

2.4.2.1 Description

The WIPSoft application provides +WIPCFG command for configuring the internal parameters of IP stack.

2.4.2.2 Syntax

AT+WIPCFG=2,<opt num>,<value>

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	It is recommended to change the default settings of the WIP stack using +WIPCFG only when it is required. Changing the parameter values especially the max number of sockets and the max TCP buffer size with the high values lead to over consumption of the stack memory which causes the WIP Soft to crash. Hence, care must be taken when the default settings of the stack is changed using +WIPCFG command.

2.4.3 Retrieving IP Stack Configuration (New)

2.4.3.1 Description

The WIPSoft application provides AT+WIPCFG? command for retrieving the internal parameters of IP stack.

2.4.3.2 Syntax

AT+WIPCFG?

Refer to [3] for more information on the examples.

2.4.4 IP Stack Configuration Management (New)

2.4.4.1 Description

The WIPSoft application provides +WIPCFG command for storing configuring parameters in FLASH memory.

2.4.4.2 Syntax

```
AT+WIPCFG=4,<mode>
```

Refer to [3] for more information about the parameters and defined values and the examples.

2.5 Bearer Management

2.5.1 Additional Bearers (New)

The WIPSoft introduces a concept of a generic “Bearer”. A “Bearer” actually means a layer which would bear/receive the data sent to it by the IP layer and would forward it to the network. The “Bearer” can be correlated with the physical layer that is present in the OSI layer model.

In eDSoft, only two bearers are available to establish socket connections. These bearers are:

- GSM bearer: This bearer indicates that a GSM data call will be used to establish the IP connectivity. In this case, GSM data call will act as the physical layer.
- GPRS bearer: This bearer indicates that GPRS session will be used to establish the IP connectivity. In this case, GPRS session will act as the physical layer.

The WIPSoft application extends the above mentioned scenario and provides more bearers using which the IP layer connectivity can be established. The bearers that are available in WIPSoft are:

- GSM bearer: The GSM data call (as mentioned above)
- GPRS bearer: The GPRS bearer (as mentioned above)
- UART1: UART1 can also be used to establish an IP layer connection. A external device (For e.g. PC) can be connected to the Wireless CPU® to transfer TCP/IP data
- UART2: UART2 is used to establish the IP layer connection. This indicates that the client/server is running on the external microprocessor connected to UART2.
- CMUX ports over UARTs: The CMUX ports can also be used to establish the IP layer connection.

2.5.1.1 Associated Commands +WIPBR

2.5.1.1.1 Description

The WIPSoft application provides +WIPBR command to select and open a new available bearer such as UART.

2.5.1.1.2 Syntax

AT+WIPBR=1,<bid>

Refer to [3] for more information about the parameters and defined values and the examples.

2.5.2 Bearer Configuration Commands

2.5.2.1 Deprecated Commands

The following configuration commands are not available in WIPSoft:

- AT#ANSWERMODE
- AT#CALLBACKTIMER
- AT#CALLSCREENNUM
- AT#PHYTIMEOUT
- AT#DIALN2
- AT#DIALSELECT
- AT#REDIALCOUNT
- AT#REDIALDELAY
- AT#PPPPEERIP=<IP>

2.5.2.2 Selecting the GSM/GPRS Bearer

The +WIPBR allows to select between GSM and GPRS bearer.

Old interface
AT#GPRSMODE=<mode> *//Select GSM/GPRS bearer*
New interface
AT+WIPBR=1,<bid> *// Select GSM/GPRS bearer with additional parameters*

Refer to [3] for more information about the parameters and defined values and the examples.

2.5.2.3 Configuring the PPP Mode

The +WIPBR command can be used to configure the PPP mode (client/server)

Old interface
AT#PPPMODE=<mode> *//Select client/server*
New interface
AT+WIPBR=4,<bid>,0,[,<login>,<password>,[,<caller identity>]] *// The parameters <cmdtype>,<bid> and <mode> are mandatory when the bearer is started as a client.* **AT+WIPBR=4,<bid>,1,<login>,<password>,[,<caller identity>]** *//The parameters <cmdtype>,<bid>,<mode>,<login> and <password> are mandatory when the bearer is started as a server.*

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	Several bearer can be opened at the same time but only one bearer can be started at a time

2.5.2.3.1 Configuring ISP Parameters

The +WIPBR command allows to configure the

- number to dial
- user name
- password

Old interface
AT#DIALN1=<number to dial> **AT#ISPUN=<user name>** **AT#ISPPW=<password>**
New interface
AT+WIPBR=2,5,2,<number to dial> **AT+WIPBR=2,5,0,<user name>** **AT+WIPBR=2,5,1,<password>**

2.5.2.4 Configuring the GSM PPP Server Bearer

2.5.2.4.1 Configuring Ring Counts

The +WIPBR command allows to configure the ring count. The WIPSoft command does not allow automatic accept but provides the ring indication only after ring counts specified in <value> parameter have elapsed.

Old interface
AT#RINGCOUNT=<value>
New interface
AT+WIPBR=2,5,5,<value>

2.5.2.4.2 Configuring IP Address of PPP Server

The +WIPBR command allows to configure the IP address assigned to Wireless CPU® itself when in PPP server mode.

Old interface
AT#PPPMYIP=<IP>
New interface
AT+WIPBR=2,<bid>,15,<IP>

2.5.2.4.3 Configuring ISP Authentication Parameters

The +WIPBR command allows to configure the username and password for PPP server. This authentication details should be used by PPP client while connecting to PPP server.

Old interface
AT#PPPSERVUN=<username> **AT#PPPSERVPW=<password>**
New interface
AT+WIPBR=2,5,0,<username> **AT+WIPBR=2,5,1,<password>**

2.5.2.5 Configuring the GPRS Bearer Parameters

The +WIPBR command allows to configure the access point related parameters for GPRS. These parameters include:

- access Point name
- user name
- password
- context id

Old interface
AT#APNSERV=<APN> **AT#APNUN=<username>** **AT#APNPW=<password>** **AT#GPRSCID=<Context id>**
New interface
AT+WIPBR=2,6,11,<APN> **AT+WIPBR=2,6,12,<Context id>** **AT+WIPBR=2,6,0,<username>** **AT+WIPBR=2,6,1,<password>**

2.5.3 Connection Management Commands

2.5.3.1 Deprecated Commands

The following configuration commands are not available in WIPSoft:

- AT#ACCEPT

2.5.3.2 Start the Bearer

The +WIPBR can be used to start the TCP/IP connection procedure.

Old interface
AT#CONNECTIONSTART
New interface
AT+WIPBR=4,<bid>,0,[,<login>,<password>,[,<caller identity>]] *// The parameters <cmdtype>,<bid> and <mode> are mandatory when the bearer is started as a client.* **AT+WIPBR=4,<bid>,1,<login>,<password>,[,<caller identity>]** *//The parameters <cmdtype>,<bid>,<mode>,<login> and <password> are mandatory when the bearer is started as a server.*

2.5.3.3 Stop the Bearer

The +WIPBR can be used to stop the active or outgoing connection.

Old interface
AT#CONNECTIONSTOP
New interface
AT+WIPBR=5,<bid> *//parameter value "5" is used to stop the connection procedure*

2.5.4 Miscellaneous Commands

2.5.4.1 Displaying IP Address

The +WIPBR command can be used to get the current IP address

Old interface
AT#DISPLAYIP
New interface
AT+WIPBR=3,<bid>,15 *//parameter value "15" is used to get the local IP address*

2.5.4.2 Displaying PPP Parameters

The +WIPBR command can be used to get the current PPP parameters

Old interface
AT#VPPP
New interface
AT+WIPBR=3,5,0 *//Username* **AT+WIPBR=3,5,1** *//Password* **AT+WIPBR=3,5,5** *//Ring count* **AT+WIPBR=3,5,15** *//Local IP address* **AT+WIPBR=3,5,16** *//Peer IP address*

NOTE	The local and destination IP address fields will display valid addresses only when the bearer is started and connected, else it will display an address "0.0.0.0".

2.5.4.3 Displaying GPRS Parameters

The +WIPBR command can be used to get the current GPRS parameters

Old interface
AT#VGPRS
New interface
AT+WIPBR=3,6,0 *//Username* **AT+WIPBR=3,6,1** *//Password* **AT+WIPBR=3,6,12** *//Context id* **AT+WIPBR=3,6,11** *// APN*

2.5.4.4 Displaying Physical Layer Parameters

The +WIPBR command can be used to get the current physical layer parameters such as APN, IP address, dial number.

Old interface
AT#VPHY
New interface
AT+WIPBR=3,<bid>,11 *//APN* **AT+WIPBR=3,<bid>,15** *//IP address* **AT+WIPBR=3,<bid>,2** *//Phone number to dial*

2.5.5 Bearer Configuration Management (New)

2.5.5.1 Description

The WIPSoft application provides +WIPBR command for storing configuring parameters in FLASH memory.

2.5.5.2 Syntax

AT+WIPBR=6,<bid>,<mode>

Refer to [3] for more information about the parameters and defined values and the examples.

2.6 TCP Sockets

2.6.1 Deprecated Commands

Following command is not available in WIPSoft for TCP sockets:

- AT#TCPTXDELAY

2.6.2 Socket Configuration Commands

2.6.2.1 Additional Configuration Command (New)

2.6.2.1.1 Description

The WIPSoft application provides additional options which can be used to configure the way the socket behaves.

2.6.2.1.2 Syntax

AT+WIPOPT=<protocol>,<idx>,2,<optnum>,<optval>

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	It is possible to configure option value using +WIPOPT command only when the TCP socket (given by <idx>) is active else it returns error.

2.6.2.2 Configuring TCP Parameters

The +WIPCREATE allows to configure the TCP parameters such as port number, IP address.

Old interface
AT#TCPORT=1,"port"
AT#TCPSERV=1,"IP address"
//In case of TCP client, these parameters are for remote server
//In case of TCP server, TCP/IP library will listen to this port and allow
//the IP address mentioned in the TCPSERV to connect

New interface
//For TCP client **AT+WIPCREATE=2,<communication index>,<peer IP>,<peer port>** *//For TCP server* **AT+WIPCREATE=3,<server index>,<local port>,<from idx>,<to idx>** *//<from idx> and <to idx> indicates minimum and maximum index for spawned TCP sockets*

NOTE	The +WIPCREATE command configures and creates the socket at the same time. Configuration cannot be done separately in WIPSoft.

2.6.3 Socket Creation/Termination Commands

2.6.3.1 Socket Creation

The +WIPCREATE command can be used to create TCP socket in client or server mode.

Server spawns a new socket whenever a client wants to communicate. The clients will be assigned an index based on the <from idx> and <to idx> that is specified along with the +WIPCREATE command. <from idx> indicates the minimum index that will be used between the server and the client. For the subsequent client connections the consecutive indexes till the <to idx> will be used. For example,

- Create server socket using command +WIPCREATE=3,1,80,5,10. Here the <from idx> is specified as 5 and <to idx> as 10.
- Server spawns a socket with communication index 5 when the first client connects to the server. All the communication with this client will be done through the spawned socket with the index as 5.
- Server spawns a socket with communication index 6 when the second client request for connection with the server. All the communication with this client will be done through the spawned socket with the index as 6.

Old interface
//For TCP client **AT#OTCP** *//For TCP Server* **AT#LTCPSTART**
New interface
//For TCP client, **AT+WIPCREATE=2,<communication index>,<peer IP>,<peer port>** *//For TCP server* **AT+WIPCREATE=3,<server index>,<local port>,<from idx>,<to idx>** *//From <idx> and <to idx> indicates minimum and maximum index for spawned TCP sockets*

2.6.3.2 Socket Termination

The +WIPCLOSE command can be used to terminate TCP socket in client or server mode.

Old interface
//For TCP client **[ETX character]** *//For TCP server* **AT#LTCPSTOP**
New interface
//For TCP client **AT+WIPCLOSE=2,<idx>** *//For TCP server* **AT+WIPCLOSE=3,<idx>**

Refer to [3] for more information about the parameters and defined values and the examples.

2.6.4 Data Transfer Command

The +WIPDATA command can be used to transfer the data to/from socket. This command switches the state of UART to data mode and allows reading/writing of data. There are 2 different modes available for data transfer and are described below:

Mode	Description
0	unmap: This parameter is used to switch the UART (mapped to continuous mode) to AT mode.
1	continuous: This parameter is used to switch the UART* to data mode.
2	continuous transparent: This parameter is used to switch the UART* to data mode. In this mode,[DLE]/[ETX] are considered as normal data and not as special characters.

NOTE	In continuous mode, [ETX] character will shutdown the socket. In case [DLE]/[ETX] character needs to be transmitted as data, it should be preceded by [DLE] character. Similarly, [DLE]/[ETX] characters received from the TCP/IP stack from the internet are sent to the host through the serial port preceded by a [DLE] character. In continuous transparent mode, [DLE]/[ETX] characters are considered as normal data and not as special characters. To close the sockets, switch the Wireless CPU® to AT command mode and use +WIPCLOSE command. The UART switches back to AT mode due to "+++" sequence or +WIPDATA=1,x,0 command.

Old interface
AT#OTCP CONNECT ... *//Module switches to data mode immediately after socket is created*
New interface
AT+WIPDATA=2,<idx>,<mode> *//Module can switch back and forth between AT mode and data mode as often as //wished.* *//Switches to data mode can happen on different UARTs. For instance, a socket can be created with +WIPCREATE on UART1, then the switch to data mode with +WIPDATA on UART2.* *//In continuous mode, <ETX> character must be escaped by <DLE> character.*

Old interface
AT#DLEMODE=<mode> *//Mode can be set as 0 or 1. This mode is done for the management of [DLE]/[ETX] characters for data transfer.*
New interface
AT+WIPDATA=2,<idx>,2 *//When the <mode> is set to 1, it is continuous mode(same as when the DLEMODE is set to 1)* *//When the <mode> is set to 2, it is the continuous transparent mode(same as when the DLEMODE is set to 0).*

Refer to [3] for more information about the parameters and defined values and the examples.

2.6.5 Miscellaneous Commands

2.6.5.1 Displaying TCP Configuration Parameters

The +WIPOPT command can be used to get the current TCP configuration parameters

Old interface
AT#VTCP
New interface
//For TCP client **AT+WIPOPT=2,<idx>,1,0** *//Port* **AT+WIPOPT=2,<idx>,1,8** *//TTL* *//For TCP Server* **AT+WIPOPT=3,<idx>,1,0** *//Port* **AT+WIPOPT=3,<idx>,1**,8 *//TTL*

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	It is possible to retrieve option value using +WIPOPT command only when the TCP socket (given by <idx>) is active else it returns error.

2.7 UDP sockets

2.7.1 Deprecated Commands

The following commands are not available in WIPSoft for UDP sockets:

- AT#UDPTXDELAY
- AT#LUDPSTART
- AT#LUDPSTOP

2.7.2 Configuration Commands

2.7.2.1 Configuring UDP Parameters

The +WIPCREATE command can be used to configure parameters such as port number, IP address.

NOTE	The +WIPCREATE command configures and creates the socket at the same time. Configuration cannot be done separately in WIPSoft.

Old interface
AT#UDPPORT=<port number> **AT#UDPSERV=<IP address>**
New interface
AT+WIPCREATE=1,<communication index>[,<local port>][,<peer IP>, <peer port>]

2.7.3 Socket Creation/Termination Commands

2.7.3.1 Creating a UDP Socket

The +WIPCREATE command can be used to create a UDP socket.

Old interface
AT#OUDP
New interface
AT+WIPCREATE=1,<communication index>[,<local port>][,<peer IP>, <peer port>]

Refer to [3] for more information about the parameters and defined values and the examples.

2.7.3.2 Terminating a UDP Socket

The +WIPCLOSE command can be used to terminate a UDP socket

Old interface
[ETX Character]
New interface
AT+WIPCLOSE=1,<idx>

Refer to [3] for more information about the parameters and defined values and the examples.

2.7.4 Data Transfer Command

The +WIPDATA command can be used to transfer the data to/from socket. This command switches the state of UART to data mode and allows reading/writing of data.

UDP is a connectionless protocol and hence there is no way to detect or cause a shutdown. However, an [ETX] character is used to mark the boundaries of datagrams.

All data written on an UDP socket is collected till an [ETX] character is encountered or the maximum size of the datagram[1] is reached and will be sent as a single datagram. Similarly when reading data, all data will be read till an [ETX] character is encountered which indicates the end of the datagram.

There are 2 different modes available for data transfer and are described below:

Mode	Description
0	unmap: This parameter is used to switch the UART (mapped to continuous mode) to AT mode.
1	continuous: This parameter is used to switch the UART* to data mode. In this mode, size of the buffer need not be mentioned.

NOTE	In continuous mode, [ETX] character will mark the end of datagram. In case [DLE]/[ETX] character needs to be transmitted as data, it should be preceded by [DLE] character. Similarly, [DLE]/[ETX] characters received by the internet are sent to the host through the serial port preceded by a [DLE] character. To close the sockets, switch the Wireless CPU® to AT command mode and use +WIPCLOSE command. The UART switches back to AT mode due to "+++" sequence or +WIPDATA=1,x,0 command. Using +WIP AT commands, when receiving several UDP datagrams on an IP bearer, +WIPDATA indication is sent once for the first received datagram. Next indication (for next remaining UDP datagram to read) is sent once the first datagram have been read (using +WIPDATA command)

Old interface
AT#OUDP CONNECT ... *//Module switches to data mode immediately after socket is created*
New interface
AT+WIPDATA=1,<idx>,<mode> *//Module switches to data mode manually*

Refer to [3] for more information about the parameters and defined values and the examples.

[1] *Maximum size of an UDP datagram has been fixed to 5840 Bytes. This limit is an arbitrary one. Nevertheless, note that smaller the datagram is the surer it will reach the aimed destination. Note that UDP is not a reliable transport layer.*

2.7.5 Miscellaneous Commands

2.7.5.1 Displaying UDP Parameters

The +WIPOPT command can be used to get the current UDP parameters.

Old interface
AT#VUDP
New interface
AT+WIPOPT=1,<idx>,1,0 *//Port* **AT+WIPOPT=1,<idx>,1,8** *//TTL*

Refer to [3] for more information about the parameters and defined values and the examples.

 NOTE	It is possible to retrieve option value using +WIPOPT command only when the UDP socket (given by <idx>) is active else it returns error.

2.8 FTP Service

2.8.1 Configuration Commands

2.8.1.1 Configuring FTP Server Parameters

The +WIPCREATE command can be used to configure parameters such as port number, IP address of FTP server

Old interface
AT#FTPPORT=<port number> **AT#FTPUN=<username>** **AT#FTPPW=<password>** **AT#FTPSERV=<IP address>**
New interface
AT+WIPCREATE=4,<index>,<server>[,<peer_port>],<username>, <password>[,<account>]

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	The +WIPCREATE command configures and creates the FTP session at the same time. Configuration cannot be done separately in WIPSoft

2.8.1.2 Configuring FTP Transfer Parameters

The +WIPOPT command can be used to configure FTP transfer related parameters such as mode of transfer.

Old interface
AT#FTPTYPE=<type of data transfer> **AT#FTPMODE=<mode of data transfer>**
New interface
AT+WIPOPT=4,<idx>,2,40,<optval> **AT+WIPOPT=4,<idx>,2,41,<optval>**

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	It is possible to configure option value using +WIPOPT command only when the FTP session (given by <idx>) is active else it returns error.

2.8.1.3 Configuring Parameters Related to File Upload

The +WIPFILE command can be used to set the file name to be uploaded.

Old interface
AT#FTPPUTFILENAME=<filename> **AT#FTPPUTPATH=<path of file>**
New interface
AT+WIPFILE=4,<idx>,2,<filename> *//Filename contains both path as well as file name*

NOTE	The +WIPFILE command sets the file name to be uploaded and uploads the file at the same time. The file name cannot be set separately in WIPSoft.

2.8.1.4 Configuring Parameters Related to File Download

The +WIPFILE command can be used to set the file name to be downloaded.

Old interface
AT#FTPGETFILENAME=<filename> **AT#FTPGETPATH=<path of file>**
New interface
AT+WIPFILE=4,<idx>,1,<filename> *//Filename contains both path as well as file name*

NOTE	The +WIPFILE command sets the file name to be downloaded and downloads the file at the same time. The file name cannot be set separately in WIPSoft

2.8.2 Uploading a File

The +WIPFILE command can be used to upload a file to the FTP server. The <ETX> character indicates end of the data in the file that is being transferred.

Old interface
AT#FTPPUT <data>
New interface
AT+WIPFILE=4,<idx>,2,<filename> <data>

NOTE	In case [ETX]/[DLE] character needs to be transmitted, it should be preceded by [DLE] character. A single [DLE] character which is not preceded by a [DLE] character will be ignored. A single [ETX] character which is not preceded by a [DLE] character will indicate end of data. If a "+++" escape sequence is sent during file transfer, it is interpreted as an [ETX] character.

Refer to [3] for more information about the parameters and defined values and the examples.

2.8.3 Downloading a File

The +WIPFILE command can be used to download a file from the FTP server. The <ETX> character indicates end of the data in the file that is being transferred.

Old interface
AT#FTPGET <data>
New interface
AT+WIPFILE=4,<idx>,1,<filename> <data>

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	In case [ETX]/[DLE] character is received, it will be preceded by a [DLE] character before sending it to the mapped UART. If a "+++" escape sequence is sent during file transfer, it is interpreted as an [ETX] character.

2.8.4 Miscellaneous Commands

2.8.4.1 Displaying FTP Related Parameters

The +WIPOPT command can be used to display the parameters related to FTP.

Old interface
AT#VFTP
New interface
AT+WIPOPT=4,<idx>,1,40 **AT+WIPOPT=4,<idx>,1,41**

NOTE	It is possible to retrieve option value using +WIPOPT command only when the FTP session (given by <idx>) is active else it returns error.

2.8.5 Closing a FTP Connection

The +WIPCLOSE command can be used to close the FTP session.

Old interface
//Session closes automatically after the file is downloaded, in case //of upload, the session is closed after data transfer
New interface
AT+WIPCLOSE=4,<idx>

Refer to [3] for more information about the parameters and defined values and the examples.

2.9 PING Service

2.9.1 Deprecated Commands

The following command is not available in WIPSoft for UDP sockets:

- AT#VPING

2.9.2 Configuration Command

2.9.2.1 Configuring PING Related Parameters

The +WIPPING can be used to configure the PING related parameters.

NOTE	The +WIPPING command configures parameters and creates the PING session at the same time. Configuration cannot be done separately in WIPSoft

Old interface
AT#PINGDELAY=<interval> **AT#PINGNUM=<repeat>** **AT#PINGREMOTE=<host>**
New interface
AT+WIPPING=<host>[,<repeat>,<interval>,[<timeout>,[<nwrite>,[ttl>]]]]

2.9.3 PING Session Creation Command

2.9.3.1 Creating a PING Session

The +WIPPING can be used to ping a remote server.

Old interface
AT#PING
New interface
AT+WIPPING=<host>[,<repeat>,<interval>,[<timeout>,[<nwrite>,[ttl>]]]]

Refer to [3] for more information about the parameters and defined values and the examples.

2.10 SMTP Service

2.10.1 Deprecated Commands

The following command is not available in WIPSoft for SMTP service:

- AT#DOMAIN
- AT#DOMAIN?
- AT#BODYi
- AT#BODYi?
- AT#SMTPPORT?
- AT#SMTPPW?
- AT#SMTPSERV?
- AT#SMTPUN?
- AT#SENDMAILi
- AT#VSMTP
- AT#VMAILi
- AT#DNSSERV1
- AT#DNSSERV2
- AT#VDNS

2.10.2 Configuration Command

2.10.2.1 Configuring SMTP Server Parameters

The +WIPCREATE command can be used to configure parameters such as port number, SMTP server address, username and password.

Old interface
AT#SMTPSERV **AT#SMTPPORT** **AT#SMTPUN** **AT#SMTPPW**
New interface
AT+WIPCREATE=<mode>,<index>,<server>[,<peer port>][,<username>,<password>]

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	The +WIPCREATE command configures and creates the SMTP session at the same time. Configuration of parameters required for establishing SMTP session cannot be done separately in WIPSoft.

2.10.2.2 Configuring SMTP eMail Sending Parameters

The +WIPOPT command can be used to configure SMTP parameters such as

- sender name
- sender mail address
- receiver mail address
- CC receiver mail address
- mail subject

Old interface	
AT#SENDERADDR	*//Configure sender address*
AT#SENDERNAME	*//Configure sender name*
AT#RECi	*//Configure receiver address*
AT#CCRECi	*//Configure CC receiver address*
AT#SUBJi	*//Configure mail subject*
New interface	
AT+WIPOPT=6,<index>,2,61,"sender addr"	*//Configure sender address*
AT+WIPOPT=6,<index>,2,62,"sender name:	*//Configure sender name*
AT+WIPOPT=6,<index>,2,63,"receiver addr"	*//Configure receiver address*
AT+WIPOPT=6,<index>,2,64,"CC receiver addr"	*//Configure CC receiver address*
AT+WIPOPT=6,<index>,2,66,"mail subject"	*//Configure mail subject*

NOTE	The +WIPOPT command allows to configure multiple receiver and CC receiver address. But only one mail subject can be configured using this command. It is possible to configure option value using +WIPOPT command only when the SMTP session (given by <idx>) is active else it returns error.

Refer to [3] for more information about the parameters and defined values and examples.

2.10.3 Retrieving DNS Server Name

The +WIPBR command can be used to get the name of the DNS server used for translating the SMTP server name from literal format to IP address format.

Old interface
AT#DNSSERV1? **AT#DNSSERV2?**
New interface
AT+WIPBR=3,<bid>,17 **AT+WIPBR=3,<bid>,18**

Refer to [3] for more information about the parameters and defined values and examples.

2.10.4 Sending Mail

The +WIPFILE command can be used to send mail to different recepients configured using +WIPOPT command. The <ETX> character indicates end of mail data.

Old interface
AT#PUTMAIL Ok_Info_WaitingForData <data> <CR><LF>.<CR><LF> *//End of mail data*
New interface
AT+WIPFILE=6,<index>,2 CONNECT <data> <ETX> *//End of mail data*

NOTE	In case [ETX]/[DLE] character needs to be transmitted, it should be preceded by [DLE] character. A single [DLE] character which is not preceded by a [DLE] character will be ignored. A single [ETX] character which is not preceded by a [DLE] character will indicate end of data.If a "+++" escape sequence is sent during data transfer, it is interpreted as an [ETX] character.

2.10.5 Additional Commands

2.10.5.1 Configuring BCC Receiver Address (New)

2.10.5.1.1 Description

The WIPSoft provides +WIPOPT command for configuring BCC receiver address. This command allows configuring multiple BCC receiver address using "," as separator.

2.10.5.1.2 Syntax

```
AT+WIPOPT=6,<index>,2,65,<BCC receiver address>
```

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to configure BCC receiver address using +WIPOPT command only when the SMTP session (given by <idx>) is active else it returns error.

2.10.5.2 Retrieving BCC Receiver Address (New)

2.10.5.2.1 Description

The WIPSoft provides +WIPOPT command for retrieving BCC receiver address.

2.10.5.2.2 Syntax

```
AT+WIPOPT=6,<index>,1,65
```

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to retrieve BCC receiver address using +WIPOPT command only when the SMTP session (given by <idx>) is active else it returns error.

2.10.5.3 Retrieving Last Protocol Error Code and Error String (New)

2.10.5.3.1 Description

The WIPSoft provides +WIPOPT command for retrieving last protocol error code and error string.

2.10.5.3.2 Syntax

```
AT+WIPOPT=6,<index>,1,60
```

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to retrieve last protocol error code using +WIPOPT command only when the SMTP session (given by <idx>) is active else it returns error.

2.10.5.4 Configuring Mail Format Header (New)

2.10.5.4.1 Description

The WIPSoft provides +WIPOPT command for configuring mail format header option.

2.10.5.4.2 Syntax

```
AT+WIPOPT=6,<index>,2,67,< Format mail header >
```

Refer to [3] for more information about the parameters and defined values.

NOTE	Setting this option to zero indicates that the customer application is in charge of formatting the mail header to attach documents (see RFC 2822 for Standard for the Format of ARPA Internet Text Messages for formatting details).

2.10.5.5 Retrieving Mail Format Header (New)

2.10.5.5.1 Description

The WIPSoft provides +WIPOPT command for retrieving mail format header option.

2.10.5.5.2 Syntax

```
AT+WIPOPT=6,<index>,1,67
```

Refer to [3] for more information about the parameters and defined values.

2.10.6 Miscellaneous Commands

2.10.6.1 Displaying SMTP Related Parameters

The +WIPOPT command can be used for retrieving the parameters related to SMTP.

Old interface	
AT#SENDERADDR?	*//Get sender address*
AT#SENDERNAME?	*//Get sender name*
AT#RECi?	*//Get receiver address*
AT#CCRECi?	*//Get CC receiver address*
AT#SUBJi?	*//Get mail subject*
New interface	
AT+WIPOPT=6,<index>,1,61	*//Get sender address*
AT+WIPOPT=6,<index>,1,62	*//Get sender name*
AT+WIPOPT=6,<index>,1,63	*//Get receiver address*
AT+WIPOPT=6,<index>,1,64	*//Get CC receiver address*
AT+WIPOPT=6,<index>,1,66	*//Get mail subject*

NOTE	It is possible to retrieve option value using +WIPOPT command only when the SMTP session (given by <idx>) is active else it returns error.

2.10.7 Closing a SMTP Connection

The +WIPCLOSE command can be used to close the SMTP session.

Old interface
//Session closes automatically after the mail is sent.
New interface
AT+WIPCLOSE=6,<index>

Refer to [3] for more information about the parameters and defined values and the examples.

2.11 POP3 Service

2.11.1 Deprecated Commands

The following commands are not available in WIPSoft for POP3 service:

- AT#POP3HEADERMODE
- AT#POP3HEADERMODE?
- AT#POP3PORT?
- AT#POP3PW?
- AT#POP3SERV?
- AT#POP3UN?
- AT#VPOP3
- AT#DNSSERV1
- AT#DNSSERV2
- AT#VDNS

2.11.2 Configuring POP3 Server Parameters

The +WIPCREATE command can be used to configure parameters such as port number, POP3 server address, username and password.

Old interface
AT#POP3SERV **AT#POP3PORT** **AT#POP3UN** **AT#POP3PW**
New interface
AT+WIPCREATE=<mode>,<index>,<server>[,<peer port>][,<username>,<password>]

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	The +WIPCREATE command configures and creates the POP3 session at the same time. Configuration of parameters required for establishing POP3 session cannot be done separately in WIPSoft.

2.11.3 Retrieving DNS Server Name

The +WIPBR command can be used to get the name of the DNS server used for translating the POP3 server name from literal format to IP address format.

Old interface
AT#DNSSERV1? **AT#DNSSERV2?**
New interface
AT+WIPBR=3,<bid>,17 **AT+WIPBR=3,<bid>,18**

2.11.4 Retrieving Mail

The +WIPFILE command can be used to retrieve mail. The <ETX> character indicates end of email data.

Old interface
AT#GETMAIL
New interface
AT+WIPFILE=7,<index>,3,<mail id>

NOTE	In case [ETX]/[DLE] character is received, it will be preceded by a [DLE] character before sending it to the mapped UART. If a "+++" escape sequence is sent during data transfer, it is interpreted as an [ETX] character.

2.11.5 Additional Commands

2.11.5.1 Deleting Mail ID Without Retrieving Mail Data (New)

2.11.5.1.1 Description

The WIPSoft provides +WIPOPT command to delete mail id without retrieving mail data.

2.11.5.1.2 Syntax

AT+WIPOPT=7,<index>,2,74,<mail id>

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to delete mail id using +WIPOPT command only when the POP3 session (given by <idx>) is active else it returns error.

2.11.5.2 Retrieving Last Protocol Error Code and Error String (New)

2.11.5.2.1 Description

The WIPSoft provides +WIPOPT command for retrieving last protocol error code and string.

2.11.5.2.2 Syntax

AT+WIPOPT=7,<index>,1,70

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to retrieve last protocol error code using +WIPOPT command only when the POP3 session (given by <idx>) is active else it returns error.

2.11.5.3 Retrieving Total Number of Mails (New)

2.11.5.3.1 Description

The WIPSoft provides +WIPOPT command for retrieving total number of mails available in the mail box.

2.11.5.3.2 Syntax

AT+WIPOPT=7,<index>,1,71

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to retrieve total number of mails using +WIPOPT command only when the POP3 session (given by <idx>) is active else it returns error.

2.11.5.4 Retrieving Total Mail Size (New)

2.11.5.4.1 Description

The WIPSoft provides +WIPOPT command for retrieving total mail size available in the mail box.

2.11.5.4.2 Syntax

```
AT+WIPOPT=7,<index>,1,72
```

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to retrieve total mail size using +WIPOPT command only when the POP3 session (given by <idx>) is active else it returns error.

2.11.5.5 Retrieving List of Available Mails (New)

2.11.5.5.1 Description

The WIPSoft provides +WIPOPT command for retrieving list of available mails in the mail box.

2.11.5.5.2 Syntax

```
AT+WIPOPT=7,<index>,1,73
```

Refer to [3] for more information about the parameters and defined values.

NOTE	It is possible to retrieve list of available mails using +WIPOPT command only when the POP3 session (given by <idx>) is active else it returns error.

2.11.5.6 Retrieving Mail Without Deleting it from the Server (New)

2.11.5.6.1 Description

The WIPSoft provides +WIPFILE command for retrieving mail without deleting it from the server.

2.11.5.6.2 Syntax

```
AT+WIPFILE=7,<index>,1,<mail id>
```

Refer to [3] for more information about the parameters and defined values.

2.11.6 Closing a POP3 Connection

The +WIPCLOSE command can be used to close the POP3 session.

Old interface
//Session closes automatically after the mail is retrieved.
New interface
AT+WIPCLOSE=7,<index>

Refer to [3] for more information about the parameters and defined values and the examples.

2.12 New WIPSoft Service

This section lists the new services that are provided by WIPSoft and are not available in eDSoft.

2.12.1 HTTP Service

2.12.1.1 Creation of HTTP Session

2.12.1.1.1 Description

The +WIPCREATE command configures and creates the HTTP session at the same time. The +WIPCREATE command can be used to configure parameters such as port number, HTTP proxy server address, username, password and HTTP header information.

2.12.1.1.2 Syntax

```
AT+WIPCREATE=<mode>,<index>,[<server>[,<peer port>]],
[<username>,<password>,[<headers list>[…] ] ]
```

Refer to [3] for more information about the parameters and defined values and the examples.

2.12.1.2 Configuring HTTP Parameters

2.12.1.2.1 Description

The +WIPOPT command is used to configure parameters of HTTP service.

2.12.1.2.2 Syntax

```
AT+WIPOPT=<protocol>,<index>,<action>,<optnum>,<optval>
```

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	It is possible to configure option value using +WIPOPT command only when the HTTP session (given by <idx>) is active else it returns error.

2.12.1.3 Retrieving HTTP Parameters

2.12.1.3.1 Description

The +WIPOPT command is used to retrieve parameters of HTTP service.

2.12.1.3.2 Syntax

AT+WIPOPT=<protocol>,<index>,<action>,<optnum>

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	It is possible to retrieve option value using +WIPOPT command only when the HTTP session (given by <idx>) is active else it returns error.

2.12.1.4 Uploading a File

2.12.1.4.1 Description

The +WIPFILE command can be used to upload a file to the HTTP server. <ETX> character is used to indicate end of data.

2.12.1.4.2 Syntax

AT+WIPFILE=5,<index>,2,<filename>[,<username>, <password>][, <headers list>[…]] *//HTTP PUT method*

or

AT+WIPFILE=5,<index>,4,<filename>[,<username>, <password>][, <headers list>[…]] *//HTTP POST method*

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	In case [ETX]/[DLE] character needs to be transmitted, it should be preceded by [DLE] character. A single [DLE] character which is not preceded by a [DLE] character will be ignored. A single [ETX] character which is not preceded by a [DLE] character will indicate end of data.If a "+++" escape sequence is sent during data transfer, it is interpreted as an [ETX] character.

2.12.1.5 Downloading a File

2.12.1.5.1 Description

The +WIPFILE command can be used to retrieve a file from the HTTP server. <ETX> character is used to indicate end of data in the file that is being downloaded.

2.12.1.5.2 Syntax

AT+WIPFILE=5,<index>,1,<filename>[,<username>, <password>][, <headers list>[…]] *//HTTP GET method*

Refer to [3] for more information about the parameters and defined values and the examples.

NOTE	In case [ETX]/[DLE] character is received, it will be preceded by a [DLE] character before sending it to the mapped UART. If a "+++" escape sequence is sent during data transfer, it is interpreted as an [ETX] character.

2.12.1.6 Deleting a File

2.12.1.6.1 Description

The +WIPFILE command can be used to delete a file in the HTTP server.

2.12.1.6.2 Syntax

AT+WIPFILE=5,<index>,3,<filename>[,<username>, <password>][, <headers list>[…]] *//HTTP DELETE method*

Refer to [3] for more information about the parameters and defined values and the examples.

2.12.1.7 Closing a HTTP Session

2.12.1.7.1 Description

The +WIPCLOSE command can be used to close the HTTP session.

2.12.1.7.2 Syntax

AT+WIPCLOSE=5,<index>

Refer to [3] for more information about the parameters and defined values and the examples.

2.13 Miscellaneous Commands

2.13.1 Deprecated Commands

The following miscellaneous commands are not available in the WIPSoft:

- AT#VSTATE
- AT#VALL
- AT#DELFLASH

2.13.2 IP Stack Version Information

The +WIPCFG command is used to get the version information for the TCP/IP library.

Old interface
AT#VVERSION
New interface
AT+WIPCFG=3

wavecom

Make it wireless

WAVECOM S.A. - 3 esplanade du Foncet - 92442 Issy-les-Moulineaux Cedex - France - Tel: +33(0)1 46 29 08 00 - Fax: +33(0)1 46 29 08 08
Wavecom, Inc. - 4810 Eastgate Mall - Second Floor - San Diego, CA 92121 - USA - Tel: +1 858 362 0101 - Fax: +1 858 558 5485
WAVECOM Asia Pacific Ltd. - 4/F, Shui On Centre - 6/8 Harbour Road - Hong Kong - Tel: +852 2824 0254 - Fax: +852 2824 025