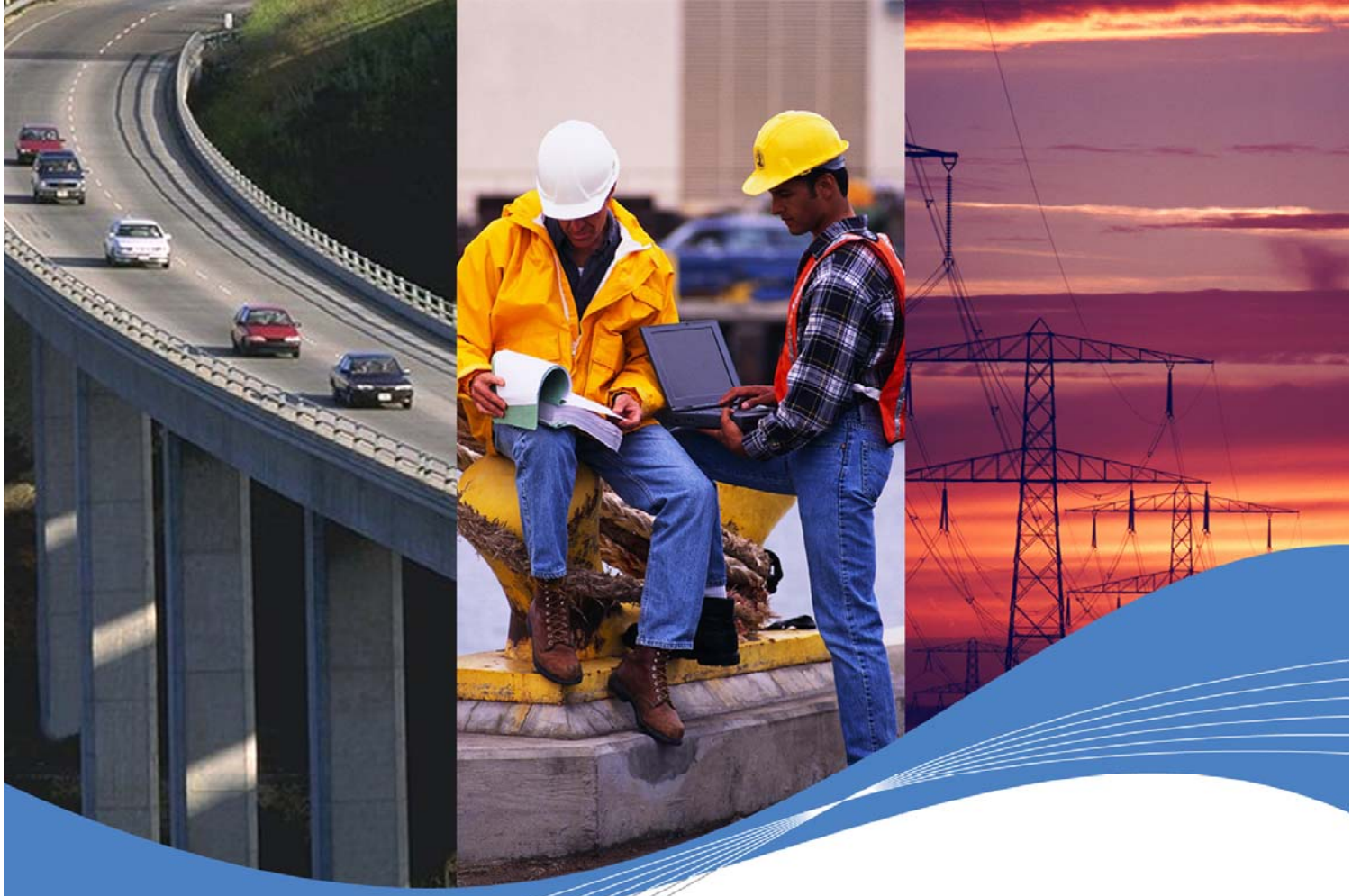

Porting Guide From EdSoft V3.10 to WIPSoft V2.00

Revision: 002
Date: September 2006

OPEN AT

wavecom
Make it wireless

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

PORTING GUIDE FROM EDSOFT V3.10 TO WIP SOFT V2.00

Reference: WM_DEV_OAT_UGD_027

Revision : 002

Date: 26th September 2006

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

No patent liability is assumed with respect to the use of the information contained herein.

Overview

The aim of this document is to provide Wavecom customers with a detailed porting guide to ease transition from the AT# AT commands used in the former TCP/IP and Internet Protocol Plug In to the new "AT+WIP" AT commands used to address new Wavecom TCP / IP and Internet Plug In.

Document History

Level	Date	History of the evolution	Writer
001	24th August 2006	Creation	Wavecom
002	26th September 2006	Preliminary	Wavecom

Contents

1 Introduction

1.1 Related Documents

- WIP Open AT IP Connectivity Development Guide (ref WM_DEV_OAT_UGD_021 revision 004).
- AT Commands Interface Guide For IP Connectivity (eDSoft V3.10) (ref WM_ASW_OAT_UGD_011 revision 007).

1.2 Abbreviations

Abbreviation	Description
APN	Access Point Name
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
IP	Internet protocol
ISP	Internet Service Provider
POP	Post Office Protocol
PPP	Point to Point Protocol
PSTN	Public Switched Telephone network
SMTP	Simple Mail Transfer Protocol
TCP	Transmission control protocol
Bearer	The term used in wipSoft used to indicate the layer providing the a transmission of data from one peer to another.

1.3 Glossary

AT commands	Set of standard modem commands.
wipSoft	An Open AT® application (providing the IP connectivity fu to the Wireless CPU) written using the Wavecom IP librar
eDSoft	An Open AT® application (providing the IP connectivity fu to the Wireless CPU) written using the eDLib IP library.
UART	Universal Asynchronous Receiver Transmitter
Data Mode	The functioning mode of UART, in which everything that i from the UART is treated as data.
AT Mode	The functioning mode of UART, in which anything that is from UART is treated as AT command.
+++	The escape sequence surrounded with 1 second delay wh used to switch the state of UART from data mode to AT m

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 protocol
2	TCP socket in client mode
3	TCP socket in server mode
4	FTP protocol

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 socket	8
TCP client	8
TCP server	4
FTP 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.

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 818. 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
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 proto
838	No more free socket
839-849	Reserved for future use
850	Unknown reason
851	Bad state

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:

Value specified in Option field	Description
1	Initialize the IP stack
0	Stop the IP stack

2.4.1.4 Examples

Commands	Possible responses
AT+WIPCFG=1 Note: Start the IP stack	OK
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, <option number>, <value>
```

2.4.2.3 Defined Values

The <option number> and <value> fields can have the following values and meanings:

Opt num	Description	Value range
0	Time to live for outgoing datagrams	0..255
1	Default TOS of outgoing datagrams	0..255
2	Time to live in seconds of incomplete fragments	1..65535
3	Number of segments of initial TCP window	1..65535
4	Default MSS (Maximum segment size) for off-link connections	0..65535

2.4.2.4 Examples

Commands	Possible responses
AT+WIPCFG=2,0,10 Note: configure the TTL parameter of outgoing datagrams to 10 units.	OK
AT+WIPCFG=2,5,10	+CME ERROR: 800 Note: Incorrect Option Number
AT+WIPCFG=2,0,300	+CME ERROR: 801 Note: Incorrect Option value

2.4.3 Retrieving IP stack configuration (New)

2.4.3.1 Description

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

2.4.3.2 Syntax

AT+WIPCFG?

2.4.3.3 Examples

Commands	Possible responses
AT+WIPCFG?	<CR><LF> +WIPCFG: 0, 10 <CR><LF> +WIPCFG: 1, 5 <CR><LF> +WIPCFG: 2, 3 <CR><LF> +WIPCFG: 3, 2 <CR><LF> +WIPCFG: 4, 2 <CR><LF> +WIPCFG: 5, <value><CR><LF> +WIPCFG: 6, <value><CR><LF> +WIPCFG: 7, <value><CR><LF> +WIPCFG: 8, <value><CR><LF> +WIPCFG: 9, <value><CR><LF> +WIPCFG: 10, <value><CR><LF> +WIPCFG: 11, <value><CR><LF>

	OK

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>
```

2.4.4.3 Defined Values

The <mode> field can have the following values and meanings:

Opt num	Description
0	Configuration storage (when existing) is freed
1	Stores IP configuration parameters

Note	Executing +WIPCFG=1 will apply default parameters when existing. It is possible to change option values at run time using +WIPCFG=2,<optnum>,<optvalue>.

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 exertenal 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 bearers such as UART.

2.5.1.1.2 Syntax

AT+WIPBR=1, <bid>

2.5.1.1.3 Defined Values

The <bid> field can take the values as defined in the following table.

bid	Bearer
1	UART1*
2	UART2*
3	N.A.
4	N.A.
5	GSM
6	GPRS
11..14	CMUX port over UART1*
21..24	CMUX port over UART2*

*New bearers

Note	Opening bearer only consists in associating the IP protocol stack with the specified bearer. The corresponding bearer setup has to be done through the adequate already existing AT commands (please refer to +WMFM commands for UART1 and 2, +CMUX command for CMUX virtual ports and GSM/GPRS AT commands).

2.5.1.1.4 Examples

Commands	Possible responses
AT+WIPBR=6 Note: Select GPRS as the bearer	OK
AT+WIPBR=0	+CME ERROR: 800 Note: Invalid option
AT+WIPBR=6	+CME ERROR: 803 Note: Already Open
AT+WIPBR=5	+CME ERROR: 802 Note: Memory error
AT+WIPBR=3	+CME ERROR: 804 Note: Not available on this platform.

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

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=2,<bid>,<mode>, <other params> **// Select GSM/GPRS bearer with additional parameters**

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>,<mode>, <other params> **// Mode determines the client or server**

The <other params> field can take the values depending on the <mode> and the bearer type as defined in the following table.

Bid	Mode	Other params
1..3,11..14,21..24	0	None
1..3,11..14,21..24	1	<PPP login>, <PPP password>
5	0	None
5	1	<login>,<password>[,<caller identity>]
6	0	None

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=2,5,0, <number to dial>,<login>,<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 IP address of PPP client

The +WIPBR command allows to configure the IP address assigned to the peer when Wireless CPU is configured as a PPP server.

Old interface
AT#PPPPEERIP=<IP>
New interface
AT+WIPBR=2,5,16,<IP>

2.5.2.4.4 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>,<mode>,<other params> **//parameter value "4" is used to start the connection procedure**

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`

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>
```

2.5.5.3 Defined Values

For more details on available <bid> parameters, please refer section 2.5.1.13. The <mode> field can have the following values and meanings:

Opt num	Description
0	Configuration storage (when existing) is freed
1	Stores bearer configuration parameters

Note	Executing +WIPBR=1,<bid> will open bearer <bid> with default parameters of the bearer when existing. It is possible to change option values at run time using +WIPBR=2,<bid>,<opt num>,<value>

2.6 TCP Sockets

2.6.1 Deprecated commands

Following commands are not available in wipSoft for TCP sockets:

- AT#DLEMODE
- 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>
```

2.6.2.1.3 Defined values

- protocol: Specifies the protocol on which the option is to be applied. Please refer to section 2.1 for more information on the values that can be taken by "protocol" field.

- IDX: Specifies the index of the socket on which the particular setting has to be performed.

- optnum and optval: Option number and option value respectively as specified in the following table (here 'R' specifies that the option can only be read, 'RW' specifies that the option can be read and written)

opt num	value format	Meaning	UDP	TCP client	TCP server
0	0..65535	Port to be used by the socket.	R	R	R
1	0..65535	Peer port which is used for the socket.	RW	R	-
2	string	IP address of the peer.	RW	R	-
3	0..1	Indicates whether the socket is bound or not to a particular UART.	R	-	-
4	0..65535	Minimum amount of available space that must be available in the emission buffer before triggering a write event.	-	RW	RW
5	0..65535	Minimum amount of available space that must be available in the reception buffer before triggering a read event.	-	RW	RW
6	0.65535	Number of bytes that can currently be read on that socket.	R	R	-
7	0..1	When set, TCP packets are sent immediately, even if the buffer is not full enough.	-	RW	RW

8	0..255	Time to live for this socket.	RW	Rw	RW
9	0..255	Type of service for this socket.	RW	RW	RW

2.6.2.1.4 Examples

Commands	Possible responses
AT+WIPOPT=2,1,2,8,20 Note: TTL for socket	OK

2.6.2.2 Configuring TCP parameters

The +WIPCREATE allows to configure the TCP 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#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**

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 the 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>

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.

Old interface
AT#OTCP CONNECT … //Module switches to data mode immediately after socket is created
New interface

AT+WIPDATA=2,1,1
//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

Old interface
with +WIPDATA on UART2. //In continuous mode, <ETX> character must be escaped by <DLE> character.

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>,0 //Port** **AT+WIPOPT=2,<idx>,1 //IP address** **//For TCP Server** **AT+WIPOPT=3,<idx>,0 //Port** **AT+WIPOPT=3,<idx>,1 //IP address**

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>`
`AT+WIPCREATE=1,<communication index>,<peer IP>,<peer port>`
`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>`
`AT+WIPCREATE=1,<communication index>,<peer IP>,<peer port>`
`AT+WIPCREATE=1,<communication index>,<local port>,<peer IP>,<peer port>`

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>`

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. 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	1. During continuous mode, [ETX] character will terminate the session. In case [ETX] character needs to be transmitted, it should be preceded by [DLE] character. To close all sockets at once, "+++" sequence should be sent followed by +WIPCLOSE command. 2. The UART switches back to AT mode due to "+++" sequence or +WIPDATA=1,x,0 command.

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

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>,0 //Port` `AT+WIPOPT=1,<idx>,1 //IP address`

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

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

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>]

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>`

2.8.1.3 Configuring parameters related to file upload

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

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.

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`

2.8.1.4 Configuring parameters related to file download

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

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

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**

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	1. In case [ETX] character needs to be transmitted, it should be preceded by [DLE] character. 2. If a "+++" escape sequence is sent during file transfer, it is interpreted as an [ETX] character.

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>**

Note	1. In case [ETX] character needs to be transmitted, it should be preceded by [DLE] character. 2. 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`

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>`

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>]]]

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>]]

2.10 SMTP/POP3 service

2.10.1 SMTP/POP3 service commands in eDSoft

The eDSoft application provides commands which can be used to send/receive emails using SMTP/POP3 protocol. Following commands are present to send/receive emails.

- AT#SENDERNAME
- AT#SENDERADDR
- AT#CCREC1/CCREC2/CCREC3
- AT#DOMAIN
- AT#REC1,REC1ADD/REC2,REC2ADD/REC3,REC3ADD
- AT#SUBJ1/SUBJ2/SUBJ3
- AT#BODY1/BODY2/BODY3
- AT#POP3HEADERMODE
- AT#POP3PORT
- AT#POP3PW
- AT#POP3SERV
- AT#POP3UN
- AT#SMTPPORT
- AT#SMTPPW
- AT#SMTPSERV
- AT#SMTPUN

- AT#DNSSERV1
- AT#DNSSERV2
- AT#GETMAIL
- AT#SENDMAIL1/SENDMAIL2/SENDMAIL3
- AT#PUTMAIL
- AT#VMAIL1/VMAIL2/VMAIL3
- AT#VPOP3
- AT#VSMTP
- AT#VDNS

2.10.2 SMTP/POP3 service commands in wipSoft

Currently, the wipSoft application does not support the SMTP/POP3 service commands. These commands will be added in future releases of wipSoft application.

2.11 Miscellaneous commands

2.11.1 Deprecated commands

The following miscellaneous commands are not available in the wipSoft:

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

2.11.2 IP stack version information

The +WIPCFG command can be 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