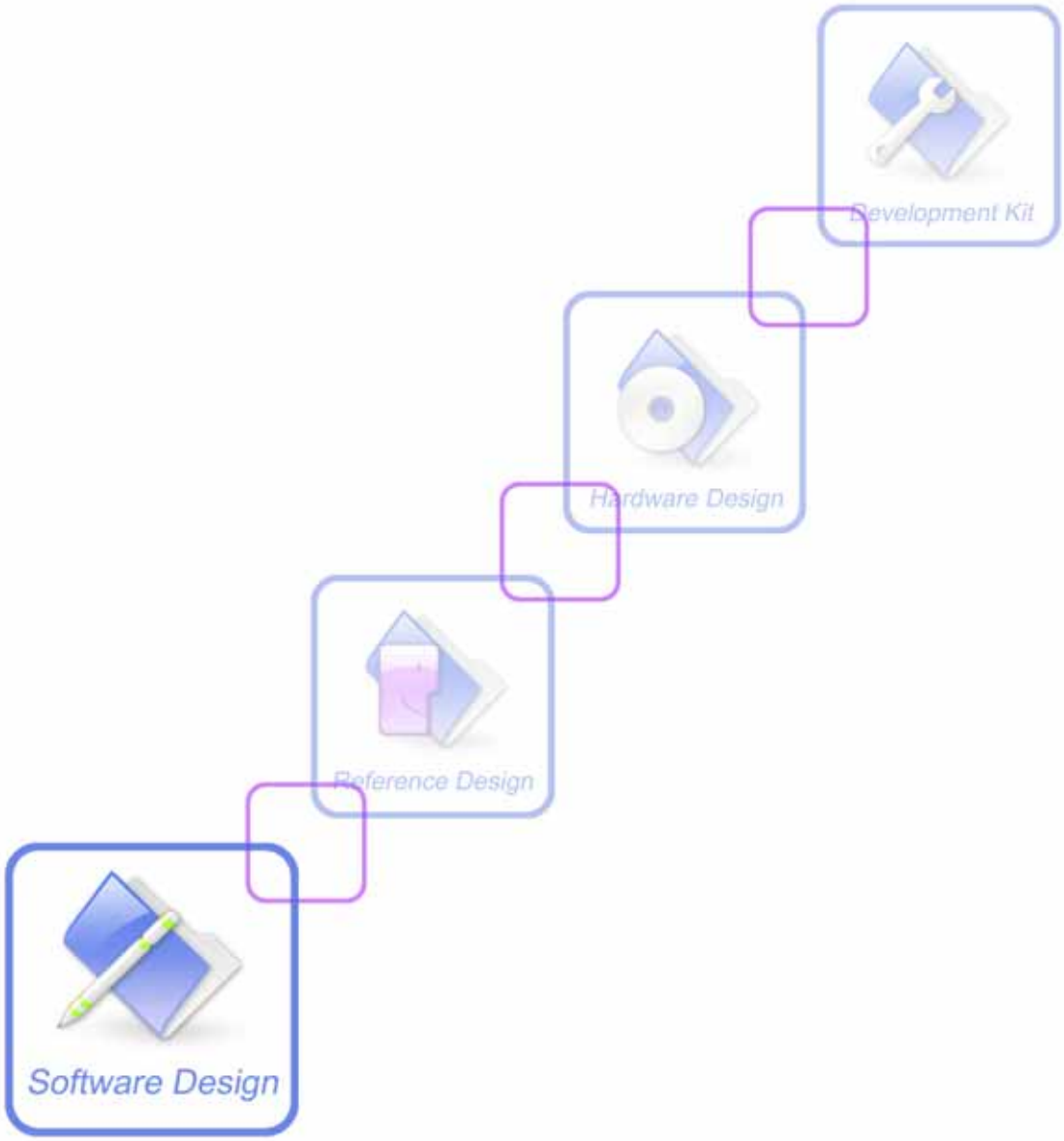
AT# Commands for IP Connectivity

IPConnect_AN_V1.03



Document Title:	AT# Commands for IP Connectivity
Version:	1.03
Date:	2008-8-25
Status:	Release
Document Control ID:	IPConnect_AN_V1.03

General Notes

SIMCOM offers this information as a service to its customers, to support application and engineering efforts that use the products designed by SIMCOM. The information provided is based upon requirements specifically provided to SIMCOM by the customers. SIMCOM has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by SIMCOM within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

Copyright

This document contains proprietary technical information which is the property of SIMCOM Limited., copying of this document and giving it to others and the using or communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights reserved in the event of grant of a patent or the registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Copyright © SIMCOM Limited. 2008

Version history

Version	Chapter	What is new
1.00	New Version	Original Version
1.01	2.3 5.5 6.5 8.2 8.3 5.2/10.8.1 10.8.1/10.8.2	Modify the format to display “MY IP” and “PEER IP” Add to display the value of “TCPMASK” Add to display the value of “UDPMASK” Modify the format to display “NO SERVICE” Correct the sequence of some parameters Delete the “OK” response after AT#LTCPSTART Correct the mistake to use AT#TCPPORT
1.02	2.1.5 2.1.8 2.1.9 2.4 2.6 3.5 4.4 5.1.4 6.1.3 5.2/10.8.1	Modify the legal values of REDIALDELAY Modify the legal values of DIALN1 Modify the legal values of DIALN2 Correct the sequence of the response of VPPP Add CAUTO to the parameters list of VPHY Correct the sequence of the response of VMAIL Correct the sequence of the response of VFTP Modify the legal values of TCPTXDELAY Modify the legal values of UDPTXDELAY Restore the “OK” response for AT#LTCPSTART
1.03	2.1.18	Correct the maximum characters for APNPW to 50
	2.1.20	Correct the maximum characters for APNUN to 50
	2.1.21	Correct VGPRS to VPHY
	4.1.1 5.1.2 6.1.1	Modify the range of port for TCPPORT, UDPPORT and FTPPORT
	8.1	Delete the quotes of version
	9.1	Add the numeric 3087 for information “Ok_Info_ListenAborted”

Contents

1 INTRODUCTION 6

1.1 DEFINITIONS AND ABBREVIATIONS 6
1.2 AT# SOFTWARE INTERFACE 6
1.2.1 Description 6
1.2.2 To set a parameter 6
1.2.3 To check a parameter value 6
1.2.4 To check a parameters category 7
1.2.5 To send a command 7
1.2.6 Response messages 7

2 DIALING SERVICES 8

2.1 PARAMETERS DEFINITION 8
2.1.1 ANSWERMODE 8
2.1.2 CALLBACKTIMER 9
2.1.3 CALLSCREENNUM 9
2.1.4 REDIALCOUNT 10
2.1.5 REDIALDELAY 10
2.1.6 PHYTIMEOUT 11
2.1.7 RINGCOUNT 11
2.1.8 DIALN1 12
2.1.9 DIALN2 12
2.1.10 DIALSELECT 13
2.1.11 ISPPW 13
2.1.12 ISPUN 14
2.1.13 PPPMODE 14
2.1.14 PPPMYIP 14
2.1.15 PPPPEERIP 15
2.1.16 PPPSERVUN 16
2.1.17 PPPSERVPW 16
2.1.18 APNPW 17
2.1.19 APNSERV 17
2.1.20 APNUN 18
2.1.21 GPRSMODE 18
2.1.22 CAUTO 18
2.2 INCOMING CALL MANAGEMENT 19
2.2.1 Answer incoming call #ACCEPT 19
2.2.2 Stop communication #CONNECTIONSTOP 19
2.2.3 Start communication #CONNECTIONSTART 20
2.3 DISPLAY IP ADDRESSES #DISPLAYIP 22
2.4 DISPLAY PPP PARAMETERS #VPPP 23
2.5 DISPLAY GPRS PARAMETERS #VGPRS 23
2.6 DISPLAY PHYSICAL PARAMETERS #VPHY 24

3 SMTP/POP3 E-MAIL SERVICES 25

3.1 PARAMETERS DEFINITION 25
3.1.1 SENDERNAME 25
3.1.2 SENDERADDR 26
3.1.3 CCREC1 / CCREC2 / CCREC3 26
3.1.4 DOMAIN 27
3.1.5 REC1/ REC2 / REC3Definition 27
3.1.6 SUBJ1 / SUBJ2 / SUBJ3 28
3.1.7 BODY1 / BODY2 / BODY3 28
3.1.8 POP3HEADERMODE 29
3.1.9 POP3PORT 29
3.1.10 POP3PW 30
3.1.11 POP3SERV 30
3.1.12 POP3UN 31
3.1.13 SMTPPORT 31

3.1.14 SMTPPW *32*
3.1.15 SMTPSERV *32*
3.1.16 SMTPUN *33*
3.1.17 DNSSERV1 ..... *33*
3.1.18 DNSSERV2 ..... *34*
3.2 RETRIEVE HOST MAIL #GETMAIL 34
3.3 SEND MAIL #SENDMAIL1 / #SENDMAIL2 / #SENDMAIL3 35
3.4 SEND HOST MAIL #PUTMAIL 36
3.5 DISPLAY E-MAIL PARAMETERS #VMAIL1 / #VMAIL2 / #VMAIL3 37
3.6 DISPLAY POP3 PARAMETERS #VPOP3 38
3.7 DISPLAY SMTP PARAMETERS #VSMTP 38
3.8 DISPLAY DNS PARAMETERS #VDNS 39

4 FTP SERVICES 41

4.1 PARAMETERS DEFINITION 41
4.1.1 FTPPORT *41*
4.1.2 FTPMODE *42*
4.1.3 FTPTYPE *42*
4.1.4 FTPSERV *43*
4.1.5 FTPUN *43*
4.1.6 FTPPW *44*
4.1.7 FTPGETFILENAME *44*
4.1.8 FTPGETPATH *45*
4.1.9 FTPPUTFILENAME *45*
4.1.10 FTPPUTPATH *46*
4.2 GET DATA FROM SERVER #FTPGET 46
4.3 PUT DATA TO SERVER #FTPPUT 47
4.4 DISPLAY FTP PARAMETERS #VFTP 48

5 TCP SOCKET SERVICES 50

5.1 PARAMETERS DEFINITION 50
5.1.1 DLEMODE *50*
5.1.2 TCPPORT *51*
5.1.3 TCPSERV *51*
5.1.4 TCPTXDELAY *52*
5.1.5 TCPMASK *52*
5.2 OPEN LISTENING MODE #LTCPSTART 52
5.3 CLOSE LISTENING MODE 54
5.4 OPEN TCP CONNECTION #OTCP 54
5.5 DISPLAY TCP PARAMETERS #VTCP 56

6 UDP SOCKET SERVICES 57

6.1 PARAMETERS DEFINITION 57
6.1.1 UDPPORT *57*
6.1.2 UDPSERV *57*
6.1.3 UDPTXDELAY *57*
6.1.4 UDPMASK *58*
6.2 OPEN A UDP SESSION IN LISTEN MODE #LUDPSTART 58
6.3 CLOSE A LISTENING MODE UDP SESSION # 60
6.4 OPEN AN ACTIVE UDP SESSION #OUDP 60
6.5 DISPLAY UDP PARAMETERS #VUDP 61

7 PING SERVICES 63

7.1 PARAMETERS DEFINITION 63
7.1.1 PINGDELAY *63*
7.1.2 PINGNUM *63*
7.1.3 PINGREMOTE *64*
7.2 START PING REQUEST 64
7.3 DISPLAY PING PARAMETERS #VPING 65

8 MISCELLANEOUS 66

8.1 DISPLAY SOFTWARE VERSION #VVERSION 66
8.2 DISPLAY CURRENT STATUS #VSTATE 66
8.3 DISPLAY ALL PARAMETERS #VALL 67
8.4 ERASE FLASH MEMORY PARAMETER VALUES #DELFLASH 69

9 RESPONSE MESSAGES AND ERROR CODES 70

9.1 RESPONSE MESSAGES 70
9.2 ERROR CODES 70

10 APPENDIX A - GETTING STARTED WITH EXAMPLES 73

10.1 GET SOFTWARE VERSION 73
10.2 AT# INTERFACE 73
10.3 GSM NETWORK REGISTRATION 73
10.4 GPRS NETWORK REGISTRATION 73
10.5 CONNECTION TO THE INTERNET 74
10.5.1 Connection to the Internet Service Provider using GSM data: 74
10.5.2 Connection to the Internet Service Provider using GPRS: 75
10.6 SENDING/RETRIEVING EMAIL 76
10.6.1 Sending an email: AT#PutMail 76
10.6.2 Retrieving an email: AT#Getmail 77
10.7 FTP: DOWNLOAD / UPLOAD FILES 77
10.7.1 Upload a file to a FTP server : AT#FTPput 77
10.7.2 Download a file from a FTP server : AT#FTPget 78
10.8 TCP SOCKET 79
10.8.1 Act as a TCP server: AT#LtcpStart 79
10.8.2 Act as a TCP client: AT#OTCP 80

1 INTRODUCTION

1.1 Definitions and Abbreviations

APN	Access Point Name
<DLE>	Escape character
DNS	Domain Name Service
<ETX>	Escape character [03] in Hex
FTP	File Transfer Protocol
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication
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
Stack	Low-level software
TCP	Transmission Control Protocol
TCP/IP stack	The IP over AT stack (including application protocols) within the TTPCom In-touch module.

1.2 AT# Software interface

1.2.1 Description

The AT# interface allows to direct the product for the TCP/IP features. All the AT commands remain available in the product.
To be controlled, the eDsoft is using AT# parameters, is waiting for AT# commands and is sending response messages over the serial port. The parameters and commands are not case sensitive.

1.2.2 To set a parameter

The parameter value shall be sent to the eDsoft between quotes (“”) when the value is not an integer and without quotes (“”) when the value is an integer.

AT#parameter=”Value” [CR]
or
AT#parameter=integer[CR]

Examples:
AT#FTPSERV=”myftpserver.com” [CR]
AT#FTPPORT=21[CR]

1.2.3 To check a parameter value

The AT#parameter? Command returns the parameter value

Example:
AT#APNSERV?
#APNSERV: “mobileapn”

1.2.4 To check a parameters category

It is possible to check the parameter values for a category.
List of categories:
VALL: All
VPHY: Physical layer for GSM parameters
VGPRS: GPRS parameters
VPPP: PPP parameters
VDNS: DNS parameters
VTCP: socket TCP parameters
VUDP: socket UDP parameters
VSMTP: SMTP parameters
VPOP3: POP3 parameters
VFTP: FTP parameters
VMAIL<i>: Email combination parameters (<i> = 1, 2 or 3)

Example:
AT#VGPRS (CR)
#APNPW: “username”
#APNSERV: “apnserver”
#APNUN: “apnusername”

1.2.5 To send a command

The eDsoft is waiting for commands to operate and is returning a response message which can be either an information message or an error message.

AT#command[CR]

Example:
AT#GETMAIL [CR]

1.2.6 Response messages

The response messages are information messages or error messages. The AT+CMEE allows to select the response messages format between an error code, and error code + a string code or the verbose mode.

2 DIALING SERVICES

2.1 Parameters Definition

2.1.1 ANSWERMODE

- **Definition**

  The TCP/IP stack manages incoming calls. This parameter defines how the TCP/IP stack will behave when receiving an incoming call.

- **Setting / Getting**

  Set value: AT#ANSWERMODE=<Value>
  Get value: AT#ANSWERMODE? or AT#VPHY, AT#VALL

- **Legal values**

  - **0**: (Ignore) ignores the incoming call. In this case, it is the responsibility of the host to accept/not accept the incoming call by issuing the AT#ACCEPT command.

  - **1**: (Automatic Answer) The TCP/IP stack goes off the hook and accepts the incoming call. As described below, the calling number must match the one specified in the CALLSCREENNUM parameter. The RINGCOUNT parameter shall be > 0

  - **2**: (Static Call-back) The TCP/IP stack ignores the incoming call and then automatically dials (DIALN1 or DIALN2) by issuing an AT#CONNECTIONSTART command. As described below, the calling number must match the one specified in the CALLSCREENNUM parameter. The RINGCOUNT parameter shall be > 0. It also depends on the format of the caller phone number. In case of GPRSMODE is set to 1 (GPRS mode), the call back will be made by a GPRS connection.

  - **3**: (Dynamic Call-back) The TCP/IP stack ignores the incoming call and then automatically dials the calling number by issuing an AT#CONNECTIONSTART command. For this feature, the Caller-ID service is mandatory. As described below, the calling number must match the one specified in the CALLSCREENNUM parameter.
    Even if the GPRSMODE is set to 1, the call back will be a GSM connection to the received Caller-ID number.

- **Default value**

  0 (zero)

- **Note**
  The ANSWERMODE parameter is only usable if the CAUTO parameter is set to 1.

2.1.2 CALLBACKTIMER

- **Definition**

  This parameter defines the number of seconds the TCP/IP stack will wait before an automatic call-back operation occurs after receiving an incoming call. It only applies when the ANSWERMODE parameter is set to an automatic call-back mode (value>1). This timer starts after the end of the ringing signal.

- **Setting / Getting**

  Set value: AT#CALLBACKTIMER=<Value>
  Get value: AT# CALLBACKTIMER? or AT#VPHY, AT#VALL

- **Legal values**

  Integer between 2 and 255 inclusive. This timer is set in seconds.

- **Default value**

  2 (two)

2.1.3 CALLSCREENNUM

- **Definition**

  When receiving an incoming call, the caller identification (Caller ID) service allows the TCP/IP stack to identify the phone number of the remote caller. This information is helpful in preventing unauthorized callers to trigger actions on the TCP/IP stack.
  This parameter allows the user to filter the incoming calls when the ANSWERMODE parameter is set to an automatic mode (value>0). This filtering doesn't apply when the ANSWERMODE parameter is set to 0, in this case it is the host's responsibility to accept or reject the incoming call.
  If an incoming phone number is unauthorized, the TCP/IP stack will ignore it.

- **Setting / Getting**

  Set value : AT#CALLSCREENNUM=<Value>
  Get value : AT#CALLSCREENNUM? or AT#VPHY, AT#VALL

- **Legal values**

  - 0 (zero): No remote caller authorized
  - * (all): No filtering is applied on incoming calls. All the remote phone numbers are authorized. This value must be set when wanting to receive incoming calls while the Caller ID service is not available.
  - Decimal phone number: Only the phone number configured here before is authorized for incoming calls. Alpha-numeric ASCII text string up to 64 characters.

- **Default value**

  0 (zero)

2.1.4 REDIALCOUNT

- **Definition**

  Indicates how many unsuccessful connection attempts the TCP/IP stack will make before terminating the connection attempt.

- **Setting / Getting**

  Set value: AT#REDIALCOUNT=<Value>
  Get value: AT#REDIALCOUNT? or AT#VPHY, AT#VALL

- **Legal values**

  Integer between 0 and 14, inclusive.
  If the value is set to 0, the TCP/IP stack will not make any call retry.

- **Default value**

  5 (five)

2.1.5 REDIALDELAY

- **Definition**

  It controls the delay (in seconds), if any, that will exist between each call retry.

- **Setting / Getting**

  Set value: AT#REDIALDELAY=<Value>
  Get value: AT#REDIALDELAY? or AT#VPHY, AT#VALL

- **Legal values**

  Integer between 0 and 14 inclusive.
  If this parameter is configured to 0, the TCP/IP stack will attempt another connection immediately after terminating the previous unsuccessful attempt.

- **Default value**

  5 (five)

2.1.6 PHYTIMEOUT

- **Definition**

  Used by the TCP/IP stack in order to terminate connections to the telephone line when a long period elapses without activity. “Without activity” is defined as a period when no data is transferred between the Internet and the TCP/IP stack or between the TCP/IP stack and the attached equipment. This timer prevents the telephone line from being tied up if for any reason, some part of the system becomes stuck.

- **Setting / Getting**

  Set value: AT#PHYTIMEOUT=<Value>
  Get value: AT#PHYTIMEOUT? or AT#VPHY, AT#VALL

- **Legal values**

  Integer between 1 and 255 inclusive. This timer is set in minutes.

- **Default value**

  15 (fifteen)

- **Return codes**

  TIMEOUT: The inactivity timer is reached, the GSM/GPRS module product ends the communication.

2.1.7 RINGCOUNT

- **Definition**

  This parameter defines the number of rings that will be waited before an automatic operation occurs when receiving an incoming call. This parameter only applies when the ANSWERMODE parameter is set to an automatic mode (value > 0).
  If the ANSWERMODE parameter is used (value different from 0), the RINGCOUNT value shall be > 0 for being able to use the feature.

- **Setting / Getting**

  Set value: AT#RINGCOUNT=<Value>
  Get value: AT#RINGCOUNT? or AT#VPHY, AT#VALL

- **Legal values**

  Integer between 0 and 15 inclusive.

- **Default**

  0 (zero)

- **Note**
  The RINGCOUNT parameter must be configured in accordance with the ATS0 configuration for not interacting.

2.1.8 DIALN1

- **Definition**

  Primary dial-up phone number to connect with the local ISP. Length depends on country.

- **Setting / Getting**

  Set value: AT# DIALN1=<Value>
  Get value: AT# DIALN1? or AT#VPHY, AT#VALL

- **Legal values**

  Decimal phone numbers.
  Alpha-numeric ASCII text string up to 64 characters

- **Default value**
  There is no default value for this parameter

2.1.9 DIALN2

- **Definition**

  Secondary dial-up number to connect with the local ISP. Length depends on country.

- **Setting / Getting**

  Set value: AT# DIALN2=<Value>
  Get value: AT# DIALN2? or AT#VPHY, AT#VALL

- **Legal values**

  Decimal phone numbers.
  Alpha-numeric ASCII text string up to 64 characters

- **Default value**

  There is no default value for this parameter

2.1.10 DIALSELECT

- **Definition**

  The value of this parameter determines the number called to establish an Internet connection. It configures the TCP/IP stack to use the primary dial-up number or the secondary dial-up number.

- **Setting / Getting**

  Set value: AT# DIALSELECT =<Value>
  Get value: AT# DIALSELECT? or AT#VPHY, AT#VALL

- **Legal values**

  - 1: Use primary dial-up number
  - 2: Use secondary dial-up number

- **Default value**

  1 (one)

2.1.11 ISPPW

- **Definition**

  Password for the ISP account. When communication is initiated and once the physical (modem) connection has been established with the ISP, the TCP/IP stack must provide the ISP with the password associated with the account to be used.

- **Setting / Getting**

  Set value: AT#ISPPW=<Value>
  Get value: AT#ISPPW? or AT#VPPP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 64 characters

- **Default value**

  There is no default value for this parameter

2.1.12 ISPUN

- **Definition**

  User name of the ISP account. When communication is initiated and once the physical (modem) connection has been established with the ISP, the TCP/IP stack must provide the ISP with the user name associated with the account to be used.

- **Setting / Getting**

  Set value: AT#ISPUN=<Value>
  Get value: AT#ISPUN? or AT#VPPP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 64 characters.

- **Default value**

  There is no default value for this parameter

2.1.13 PPPMODE

- **Definition**

  The TCP/IP stack can manage the access layer through different ways. This parameter selects the behaviour the TCP/IP stack must run once the physical layer is successfully established.

- **Setting / Getting**

  Set value: AT#PPPMODE=<Value>
  Get value: AT#PPPMODE? or AT#VPPP, AT#VALL

- **Legal values**

  - **1**: (Standard PPP) the TCP/IP stack behaves as a PPP client for outgoing calls and as a PPP server for incoming calls.
  - **2**: (Reverse PPP) the TCP/IP stack behaves as a PPP server for outgoing calls and as a PPP client for incoming calls.
  - **3**: (PPP client only) the TCP/IP stack always behaves as a PPP client for both outgoing and incoming calls.
  - **4**: (PPP server only) the TCP/IP stack always behaves as a PPP server for both outgoing and incoming calls.
  - **9**: (Specific) this case is reserved for specific behaviour.

- **Default value**

  1 (one)

2.1.14 PPPMYIP

- **Definition**

  When the TCP/IP stack behaves as a PPP server (according to the PPPMODE parameter), it is in charge of the IP address attribution mechanism.
  Once the PPP authentication is successfully achieved, the remote PPP peer asks the TCP/IP stack for an IP address. The related PPP layer, called IPCP, then suggests an IP address for the peer, which

was, previously, stored in the TCP/IP stack parameters. If the remote PPP peer accepts this address, the IP link is then established.
This parameter defines the IP address to be attributed to the TCP/IP stack when the PPP Server mode is running.

- **Setting / Getting**

  Set value: AT#PPPMYIP=<Value>
  Get value: AT#PPPMYIP? or AT#VPPP, AT#VALL

- **Legal values**

  32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx)

- **Default value**
  0.0.0.0

2.1.15 PPPPEERIP

- **Definition**

  When the TCP/IP stack behaves as a PPP server (according to the PPPMODE parameter), it is in charge of the IP address attribution mechanism.
  Once the PPP authentication is successfully achieved, the remote PPP peer asks the TCP/IP stack for an IP address. The related PPP layer, called IPCP, then provides the peer with an IP address previously stored in the TCP/IP stack parameters. If the remote entity accepts this address, the IP link is then established.
  This parameter defines the IP address to be attributed to the remote PPP peer when the PPP Server mode is running.

- **Setting / Getting**

  Set value: AT#PPPPEERIP=<Value>
  Get value: AT#PPPPEERIP? or AT#VPPP, AT#VALL

- **Legal values**

  32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx)

- **Default value**
  0.0.0.0

2.1.16 PPPSERVUN

- **Definition**

  When the TCP/IP stack behaves as a PPP server (according to the PPPMODE parameter), it checks the remote PPP client login/password before to grant access.
  This parameter defines the login that must be specified by the remote PPP client.

- **Setting / Getting**

  Set value: AT#PPPSERVUN=<Value>
  Get value: AT#PPPSERVUN? or AT#VPPP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 64 characters

- **Default value**

  There is no default value for this parameter

2.1.17 PPPSERVPW

- **Definition**

  When the TCP/IP stack behaves as a PPP server (according to the PPPMODE parameter), it checks the remote PPP client login/password before to grant access.
  This parameter defines the password that must be specified by the remote PPP client.

- **Setting / Getting**

  Set value: AT#PPPSERVPW=<Value>
  Get value: AT#PPPSERVPW? or AT#VPPP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 64 characters

- **Default value**

  There is no default value for this parameter

2.1.18 APNPW

- **Definition**

  Access Point Name password parameter (coming with the APNUN) from the GSM operator for providing GPRS access.

- **Setting / Getting**

  Set value: AT#APNPW=<Value>
  Get value: AT#APNPW? or AT#VGPRS, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 50 characters.

- **Default value**

  There is no default value for this parameter

2.1.19 APNSERV

- **Definition**

  Access Point Name parameter coming from the GSM operator for providing GPRS access.

- **Setting / Getting**

  Set value: AT#APNSERV=<Value>
  Get value: AT#APNSERV? or AT#VGPRS, AT#VALL

- **Legal value**s

  Alphanumeric ASCII text string up to 100 characters.

- **Default value**

  There is no default value for this parameter

2.1.20 APNUN

- **Definition**

  Access Point Name Username parameter (coming with the APNPW) from the GSM operator for providing GPRS access.

- **Setting / Getting**

  Set value: AT#APNUN=<Value>
  Get value: AT#APNUN? or AT#VGPRS, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 50 characters.

- **Default value**

  There is no default value for this parameter

2.1.21 GPRSMODE

- **Definition**

  Configure the activation of the GSM/GPRS module software for switching between GSM or GPRS.

- **Setting / Getting**

  Set value: AT#GPRSMODE=<Value>
  Get value : AT#GPRSMODE? or AT#VPHY, AT#VALL

- **Legal values**

  - 0: The GSM/GPRS module software is configured for a GSM use
  - 1: The GSM/GPRS module software is configured for a GPRS use

- **Default value**

  1 (one)

2.1.22 CAUTO

- **Definition**
  Control of incoming and outgoing Circuit Switched calls can be changed between the native data stack and the eDstack using the AT#CAUTO command. This command should be entered before using either the eDstack or native data stack, or to specify which stack should handle an incoming call.

- **Setting / getting**
  Set value: AT#CAUTO=<Value>
  Get value: AT#CAUTO? or AT#VPHY, AT#VALL

- **Legal values**
  - 0: (default) The native data stack will handle incoming calls and can be used to make outgoing calls.
  - 1: The eDstack will handle incoming data calls and can be used to make outgoing data calls.

2.2 Incoming Call Management

2.2.1 Answer incoming call #ACCEPT

- **Description**

This command directs the TCP/IP stack to answer an incoming call.
When the TCP/IP stack receives an incoming call, it sends the "RING" messages over the serial port. Depending on the value of the ANSWERMODE parameter the TCP/IP stack may or may not, answer automatically.

If ANSWERMODE is set to 0, it is the host that is responsible for answering the incoming call.

Once the physical layer is up, the TCP/IP stack runs the applicable protocol as specified in the PPPMODE parameter.

The AT#CONNECTIONSTOP command is used to close down the call.

- **Syntax**

Command syntax : AT#ACCEPT

Command	**Possible responses**
AT#ACCEPT *Note : Manual acceptance of an incoming call (incoming call)*	OK ***Note : beginning of the call setting process*** CONNECT <speed> ***Note : Modem speed negotiated between both sides.*** xxx.xxx.xxx.xxx ***Note : IP address indication attributed to the TCP/IP stack*** PPP OK *Note : The software is ready to run IP applications to send/receive data*
AT#ACCEPT *Note : Manual acceptance of an incoming call (incoming call)*	NO CARRIER *Note: The modem handshaking process with the remote host is interrupted or unsuccessful.*
AT#ACCEPT *Note : Manual acceptance of an incoming call (incoming call)*	PPP ERROR *Note: The PPP negotiation has failed (check ISPUN, ISPPW, PPPMODE and the configuration of the PPP peer)*

- **List of parameters**

  ANSWERMODE
  IPSPW
  ISPUN
  PPPMODE
  PPPMYIP
  PPPPEER (if PPPMODE is set in server mode)
  PPPPEERIP
  PPPSENDPW
  PPPSENDUN

2.2.2 Stop communication #CONNECTIONSTOP

- **Description**

This command directs the TCP/IP stack to end a GPRS or GSM communication previously established with a START command or AT#ACCEPT.

- **Syntax**

Command syntax: AT#CONNECTIONSTOP

Command	Possible responses
AT#CONNECTIONSTOP *Note : Disconnect*	OK *Note : Phone line is released*

- **List of parameters**

No TCP-IP parameter is used for the execution of this command.

2.2.3 Start communication #CONNECTIONSTART

- **Description**

This command directs the TCP/IP stack to dial out and establish the connection. Upon receiving this signal, the TCP/IP stack automatically initiates a complete connection session according to the GPRSMODE parameter (selecting the GSM or GPRS mode).

- In GSM mode, the TCP/IP stack will dial the number according to the Dial Option parameter (DIALN1 or DIALN2 depending on DIALSELECT). If an error occurs, the TCP/IP stack automatically redials according to the REDIALCOUNT parameter. Once the physical layer is up, the TCP/IP stack runs the applicable protocol as specified in the PPPMODE parameter.
- In GPRS mode, the TCP/IP stack will establish a GPRS session with the APN using the APNUN, APNPW parameters. Once the GPRS link is up, the product is connected to the Internet. The AT#CONNECTIONSTOP command can be used to close the connection.

Important note: the GPRS attach process must be performed before a CONNECTIONSTART command is sent.

- **Syntax**

Command syntax : AT#CONNECTIONSTART

Command (GSM mode)	Possible responses
AT#CONNECTIONSTART *Note : Request connection to GSM network*	DIALING *Note: The phone line is available* 2124560123 *Note: Dial DIALN1 or DIALN2 number depending on DIALSELECT* CONNECT 9600 *Note: Modem speed negotiated between both side. If the TCP/IP stack is configured for modem only operation (PPPMODE parameter) there are no more return codes* 213.192.200.4 *Note: IP address attributed to the TCP/IP stack* Ok_Info_Ppp *Note: As soon as the TCP/IP stack displays this message, it is ready to receive commands.*

Command (GSM mode)	Possible responses
AT#CONNECTIONSTART	BUSY *Note: A busy signal is detected on the remote site.* *The TCP/IP stack will wait REDIALDELAY seconds and then dials again. This re-dialling will continue until success or until the number of call retries defined in parameter REDIALCOUNT has been reached.*
AT#CONNECTIONSTART	NO ANSWER *Note: There is no response from the remote site.* *TCP/IP stack will wait REDIALDELAY seconds and then dials again. This re-dialling will continue until success or until the number of call retries defined in parameter REDIALCOUNT has been reached.*
AT#CONNECTIONSTART	NO CARRIER *Note: The modem handshaking process with the remote host is interrupted or unsuccessful.* *The TCP/IP stack will wait REDIALDELAY seconds and then dials again. This re-dialling will continue until success or until the number of call retries defined in parameter REDIALCOUNT has been reached.*
AT#CONNECTIONSTART	#CME ERROR: 37120 #CME ERROR: 37121 #CME ERROR: 28980 #CME ERROR: 28981 *Note (only one response at a time): The PPP negotiation has failed (check ISPUN, ISPPW and PPPMODE)* *See paragraph 9.2.*
AT#CONNECTIONSTART	#CME ERROR: 35865 *Note: The product is not registered on the network*
AT#CONNECTIONSTART	#CME ERROR: 35840 *Note: The product is already running (host is connected)*

Command (GPRS mode)	Possible responses
AT#CONNECTIONSTART *Note : Request connection to GPRS network*	213.192.200.4 *Note : IP address attributed to the TCP/IP stack* Ok_Info_GprsActivation *Note: GPRS session established and product connected to the Internet*
AT#CONNECTIONSTART	#CME ERROR: 35866 *Note: All connection attempts will return this message if the GPRS session cannot be established*
AT#CONNECTIONSTART	#CME ERROR: 35865 *Note: The product is not registered on the network*
AT#CONNECTIONSTART	#CME ERROR: 35868 *Note: Aborted GPRS connection, check APN parameters.*

- **List of parameters**

  0 FOR GSM MODE :

DIALN1
DIALN2
DIALSELECT
ISPPW
ISPUN
REDIALCOUNT
REDIALDELAY

1 IF PPPMODE set to 2 or 4 :

PPPMYIP
PPPPEERIP
PPPSERVPW
PPPSERVUN

2 FOR GPRS MODE :

APNPW
APNSERV
APNUN

2.3 Display IP addresses #DISPLAYIP

- **Description**

This command allows the attached host to view the IP addresses that have been attributed during the IPCP phase of the PPP negotiation. Both local and remote PPP peer IP addresses are displayed. This command should be issued only once the PPP OK message has been received from the TCP/IP stack.

- **Syntax**

Command syntax: AT#DISPLAYIP
Response syntax:

#MY IP: xxx.xxx.xxx.xxx
#PEER IP: xxx.xxx.xxx.xxx
OK

Command (GSM mode)	Possible responses
AT#DISPLAYIP *Note : Request for local and remote IP addresses*	#MY IP: 1.2.3.4 #PEER IP: 1.0.3.5 OK *Note : PPPMYIP (IP address attributed to the TCP/IP stack) and PPPPEERIP (IP address attributed to the PPP peer)parameters values*
AT#DISPLAYIP *Note : Request for local and remote IP addresses*	#CME ERROR: 35867 *Note : no IP addresses have been attributed: no active connection or PPP/IPCP negotiation not yet completed*

Command (GPRS mode)	Possible responses
AT#DISPLAYIP *Note : Request for local and remote IP addresses*	#MY IP: 1.2.3.4 #PEER IP: 1.0.3.5 OK *Note : PPPMYIP (IP address attributed to the TCP/IP stack) and PPPPEERIP (IP address attributed to the PPP peer)parameters values*
AT#DISPLAYIP *Note : Request for local and remote IP addresses*	#CME ERROR: 35867 *Note : no IP addresses have been attributed: no active connection or PPP/IPCP negotiation not yet completed*

- **List of parameters**

PPPMYIP
PPPPEERIP

2.4 Display PPP parameters #VPPP

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the PPP layer configuration.

- **Syntax**

Command syntax: AT#VPPP

Command	Possible responses
AT#VPPP	#ISPUN: "myispun" #ISPPW: "myisppwd" #PPPMODE: 1 #PPPMYIP: "0.0.0.0" #PPPPEERIP: "0.0.0.0" #PPPSERVPW: "mypasswd" #PPPSERVUN: "myname" OK

- **List of parameters**

ISPUN
ISPPW
PPPMODE
PPPMYIP
PPPPEERIP
PPPSERVPW
PPPSERVUN

2.5 Display GPRS parameters #VGPRS

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the GPRS configuration.

- **Syntax**

Command syntax : AT#VGPRS

Command	Possible responses
AT#VGPRS	#APNPW: "access" #APNSERV: "a2myoperator.com" #APNUN: "a2b" OK

- **List of parameters**

APNPW
APNSERV
APNUN

2.6 Display physical parameters #VPHY

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the physical layer configuration.

- **Syntax**

Command syntax: AT#VPHY

Command	Possible responses
AT#VPHY	#ANSWERMODE: 0 #CALLBACKTIMER: 2 #CALLSCREENNUM: "0" #DIALN1: "" #DIALN2: "" #DIALSELECT: 1 #GPRSMODE: 1 #PHYTIMEOUT: 15 #REDIALCOUNT: 5 #REDIALDELAY: 5 #RINGCOUNT: 0 #CAUTO: 0 OK

- **List of parameters**

ANSWERMODE
CALLBACKTIMER
CALLSCREENNUM
DIALN1
DIALN2
DIALSELECT
GPRSMODE
PHYTIMEOUT
REDIALCOUNT
REDIALDELAY
RINGCOUNT
CAUTO

3 SMTP/POP3 E-MAIL SERVICES

3.1 Parameters definition

3.1.1 SENDERNAME

- **Definition**

  The sender's literal name (different from the SENDERADDR parameter, which is the sender's e-mail address). This parameter will appear in the header of the e-mail sent by the TCP/IP stack, in the field: 'From: '.

- **Setting / Getting**

  Set value: AT# SENDERNAME =<String>
  Get value: AT# SENDERNAME? or AT#VSMTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 120 characters. The address must be provided in literal format (for instance "SIMCom 246").

- **Default value**

  There is no default value for this parameter.

3.1.2 SENDERADDR

- **Definition**

  To send e-mails, the TCP/IP stack must know the e-mail address of the sender. The “sender” is the hardware platform itself or the optional attached equipment. This e-mail address will appear in the header of the e-mail sent by the TCP/IP stack, in the field ' From: '.

- **Setting / Getting**

  Set value: AT# SENDERADDR =<Value>
  Get value: AT# SENDERADDR? or AT#VSMTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 120 characters. The address must be provided in literal format (for instance dev12345678@web.zyx).

- **Default value**

  There is no default value for this parameter

3.1.3 CCREC1 / CCREC2 / CCREC3

- **Definition**

  The software can send e-mail messages to an additional recipient as a "carbon copy". This parameter contains the e-mail address of the additional recipient.
  This e-mail address will appear in the header of the e-mail sent by the TCP/IP stack in the field ' Cc: '.
  For a given value n, the “CCRECn” parameter is directly associated with the “RECn” parameter. The CCRECn tables can contain ten addresses value per n (n=1, 2 or 3).

- **Setting / Getting**

  Set one value: AT#CCRECi=<Value> (replace i by 1, 2 or 3)
  Set several values (From one to ten) :
  AT#CCRECi=1, “<Value>” <CR>
  AT#CCRECi=2, “<Value>” <CR>
  AT#CCRECi=3, “<Value>” <CR>
  …
  AT#CCRECi=10, “<Value>” <CR>

  Get value: AT#CCRECi? (replace i by 1, 2 or 3) or AT#VMAILi, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 120 characters. The address must be provided in literal format (for instance dev12345678@web.xyz).

- **Default value**

  There is no default value for this parameter.

3.1.4 DOMAIN

- **Definition**

  When sending an e-mail message, the TCP/IP stack must provide the SMTP server with the domain name of the sender. In some cases, this domain name may be different from the domain name included in the sender's e-mail address.

- **Setting / Getting**

  Set value: AT#DOMAIN=<Value>
  Get value: AT#DOMAIN? or AT#VSMTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 120 characters.

- **Default value**

  There is no default value for this parameter

3.1.5 REC1/ REC2 / REC3Definition

To send e-mail messages, the TCP/IP stack must know the e-mail address of at least one recipient. Each e-mail address will appear in the header of the e-mail sent by the TCP/IP stack, in the field ' To: '.
The RECi parameter can hold a maximum of 10 e-mail addresses, each e-mail address being at the most 120 characters long.

- **Setting one e-mail address / resetting the parameter / Getting**

  Set value / reset the parameter: AT#RECi=”Value” (replace i by 1, 2 or 3)
  Get value: AT#RECi? (replace i by 1, 2 or 3) will send the list of the RECi parameters values that are configured.
  or AT#VMAILi, AT#VALL

- **Legal values to RECi (i = 1, 2 or 3)**

  Alphanumeric ASCII text string up to 120 characters. The address must be provided in literal format (for instance dev12345678@web.xyz).

- **Setting one to ten e-mail addresses to the RECi parameter / resetting the parameter**

  Each RECi parameter (i = 1, 2 or 3) can contain ten email addresses according to an index.
  To set one to ten e-mail addresses to the RECi parameter, enter the
  AT#RECi= 1, “<value_Rec1 n°1>” <CR> command.
  AT#RECi= 2, “<value_Rec1 n°2>”<CR> command,
  …
  AT#RECi= 10, “<value_Rec1 n°10>” <CR> command

  Each e-mail address has to be an alphanumeric ASCII text string, in literal format (for instance dev12345678@web.xyz).

- **Default value**

  There is no default value for this parameter.

3.1.6 SUBJ1 / SUBJ2 / SUBJ3

- **Definition**

  These parameters contain pre-defined subjects that will be used by the TCP/IP stack to compose the e-mail header.

- **Setting / Getting**

  Set value: AT#SUBJi=<Value> (replace i by 1, 2 or 3)
  Get value: AT#SUBJi? (replace i by 1, 2 or 3) or AT#VMAILi, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 120 characters.

- **Default value**

  There is no default value for this parameter

3.1.7 BODY1 / BODY2 / BODY3

- **Definition**

  These parameters store pre-defined message bodies. They allow the host application to send pre-defined e-mail combinations

- **Setting / Getting**

  Set value: AT#BODYi=<Value> (replace i by 1, 2 or 3)
  Get value: AT#BODYi? (replace i by 1, 2 or 3)

- **Legal values**

  The body content has to be entered after the AT#BODY1<CR> command. It has to be an alphanumeric ASCII text string up to 120 characters followed by the following character : 1A (in ASCIIcode), generated in a keyboard by CTRL+C escape sequence.
  Example:

  AT#BODY1<CR>
  Text string
  <CTRL+C>

- **Default value**

There is no default value for this parameter

3.1.8 POP3HEADERMODE

- **Definition**

  When receiving an e-mail message, the TCP/IP stack can be configured to send (or otherwise) the POP3 header through the serial port. The POP3 header contains the From, Cc and Subject fields.

- **Setting / Getting**

  Set value: AT#POP3HEADERMODE=<Value>
  Get value: AT#POP3HEADERMODE? or AT#VPOP3, AT#VALL

- **Legal values**

  - 0: the e-mail header will not be sent over the serial port while retrieving
  - 1: the e-mail header will be sent over the serial port while retrieving

- **Default value**

  1 (one)

3.1.9 POP3PORT

- **Definition**

  To reach the POP3 server, the TCP/IP stack must know the port of the POP3 server used for the e-mail retrieving.

- **Setting / Getting**

  Set value: AT#POP3PORT=<Value>
  Get value: AT#POP3PORT? or AT#VPOP3, AT#VALL

- **Legal values**

  5 digits (each digit between 0 and 9 inclusive). Note that numbers above 65,535 are illegal as the port identification fields are 16 bits long in the TCP header.

- **Default value**

  110 (one hundred and ten)

- **Note**

  This parameter should be changed only upon request of your network administrator. It applies for network infrastructure including Firewalls, Proxy or specific TCP port translation.

3.1.10 POP3PW

- **Definition**

  Password for POP3 account. To retrieve e-mail messages sent to a specified e-mail address, the TCP/IP stack must know the POP3 password that has been set for that e-mail account.

- **Setting / Getting**

  Set value: AT#POP3PW=<Value>
  Get value: AT#POP3PW? or AT#VPOP3, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 64 characters.

- **Default value**

  There is no default value for this parameter

3.1.11 POP3SERV

- **Definition**

  To retrieve e-mail messages, the TCP/IP stack must know the address of the POP3 server that is to be used. The POP3 server must be the one where the specified e-mail account is hosted (which is not necessarily maintained by the local ISP).

- **Setting / Getting**

  Set value: AT#POP3SERV=<Value>
  Get value: AT#POP3SERV? or AT#VPOP3, AT#VALL

- **Legal values**

  - 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx) or
  - alphanumeric ASCII text string up to 120 characters if DNS is available.

- **Default value**

  There is no default value for this parameter

3.1.12 POP3UN

- **Definition**

  User-name for POP3 account. To retrieve e-mail messages sent to a specified e-mail address, the TCP/IP stack must know the POP3 user name that has been set for that e-mail account.

- **Setting / Getting**

  Set value: AT#POP3UN=<Value>
  Get value: AT#POP3UN? or AT#VPOP3, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 64 characters.

- **Default value**

  There is no default value for this parameter

3.1.13 SMTPPORT

- **Definition**

  To reach the SMTP server, the TCP/IP stack must know the port of the SMTP server used for the e-mail sending.

- **Setting / Getting**

  Set value: AT#SMTPPORT=<Value>
  Get value: AT#SMTPPORT? or AT#VSMTP, AT#VALL

- **Legal values**

  From 1 to 5 digits (each digit between 0 and 9 inclusive). Note that numbers above 65,535 are illegal as the port identification fields are 16 bits long in the TCP header.

- **Default value**

  25 (twenty five)

- **Note**
  This parameter should be changed only upon request of your network administrator. It applies for network infrastructure including firewalls, proxy or specific TCP port translation.

3.1.14 SMTPPW

- **Definition**

  SMTP password: To send e-mail messages, some SMTP servers use an authentication process. In these cases, the TCP/IP stack will provide the SMTP password (associated to the SMTP user name) for the e-mail sending process. If this parameter is an empty string, the authentication mode is inactive. If both this parameter and the SMTPUN parameter are not empty, the authentication mode is active.
  Note : the TCP/IP stack only supports the LOGIN authentication mechanism.

- **Setting / Getting**

  Set value: AT#SMTPPW=<Value>
  Get value: AT#SMTPPW? or AT#VSMTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 64 characters.

- **Default value**

  There is no default value for this parameter

3.1.15 SMTPSERV

- **Definition**

  To send e-mail messages the TCP/IP stack must know the address of the SMTP server that is to be used. In most cases, the local ISP maintains the SMTP server.

- **Setting / Getting**

  Set value: AT#SMTPSERV=<Value>
  Get value: AT#SMTPSERV? or AT#VSMTP, AT#VALL

- **Legal values**
  - 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx) or
  - alphanumeric ASCII text string up to 120 characters if DNS is available.

- **Default value**

  There is no default value for this parameter

3.1.16 SMTPUN

- **Definition**

  SMTP User Name:
  To send e-mail messages, some SMTP servers use an authentication process. In these cases, the TCP/IP stack will provide the SMTP user name (associated with a SMTP password) for the e-mail sending process. If this parameter is an empty string, the authentication mode is inactive. If both this parameter and the SMTPPW parameter are not empty, the authentication mode is active.
  Note : the TCP/IP stack only supports the LOGIN authentication mechanism.

- **Setting / Getting**

  Set value: AT#SMTPUN=<Value>
  Get value: AT#SMTPUN? or AT#VSMTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 64 characters.

- **Default value**

  There is no default value for this parameter

3.1.17 DNSSERV1

- **Definition**

  In order to translate the server names from literal format into IP addresses, the TCP/IP stack implements the Domain Name System (DNS) protocol. The DNS Server IP address must be specified to the TCP/IP stack.

- **Setting / Getting**

  Set value: AT#DNSSERV1=<Value>
  Get value: AT#DNSSERV1? or AT#VDNS, AT#VALL

- **Legal values**

  32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx)

- **Default value**

  0.0.0.0

3.1.18 DNSSERV2

- **Definition**

  In order to translate the server names from literal format into IP addresses, the TCP/IP stack implements the Domain Name System (DNS) protocol. The DNS Server IP address has to be specified to the TCP/IP stack. This secondary DNS server is used in case of the primary DNS server does not respond to a request.

- **Setting / Getting**

  Set value: AT#DNSSERV2=<Value>
  Get value: AT#DNSSERV2? or AT#VDNS, AT#VALL

- **Legal values**

  32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx)

- **Default value**

  0.0.0.0

3.2 Retrieve host mail #GETMAIL

- **Description**

This command allows the attached host to direct the TCP/IP stack to retrieve the first mail present in the POP3 server list.
Once an IP link is established, the attached host can retrieve an e-mail message at any time (except when the TCP/IP stack is already in a process using TCP resources).
This command is similar to a "check e-mail box" feature issued by a standard messaging client on a PC.

- **Syntax**

Command syntax: AT#GETMAIL

Command	Possible responses
AT#GETMAIL *Note : Retrieve mail*	Ok_Info_Mail <mail content> *Note : This message is issued when one e-mail message is located in the specified POP3 mailbox.* *Depending on the POP3HEADERMODE parameter, the TCP/IP stack sends the e-mail header over the serial port to the attached host.* *The (CR)(LF)(.)(CR)(LF) sequence finally indicates the end of the e-mail body.*
AT#GETMAIL	Ok_Info_NoMail *Note: There is no e-mail to retrieve in the POP3 mailbox*
AT#GETMAIL	#CME ERROR: 38027 *Note: The address of the POP3 server has not been resolved by the secondary DNS server.* *TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong POP3 server address has been filled in.*

Command	Possible responses
AT#GETMAIL	#CME ERROR: <value> *Note: An error has occurred during the communication with the remote POP3 server. It may also happen during the data transfer after the MAIL message. In this case it is preceded by a (CR)(LF)(.)(CR)(LF) sequence.* *This error can be due to one of the following reason:* • *the DNS servers are not able to resolve the POP3 server address* • *the POP3 server is temporarily out of service* • *the authentication (POP3UN, POP3PW) is not valid* *Please refer to paragraph 9.2*

- **List of parameters**

POP3HEADERMODE
POP3PORT
POP3UN
POP3PW
POP3SERV

3.3 Send mail #SENDMAIL1 / #SENDMAIL2 / #SENDMAIL3

- **Description**

This command sends one of the 3 pre-defined e-mail combinations.
Once an IP link is established, the attached host can direct the TCP/IP stack to send an e-mail message at any time (except when the TCP/IP stack is already in a process using TCP resources).
The header of this e-mail is built using the REC1/2/3, CCREC1/2/3 and SUBJ1/2/3 parameters while the body is filled in the BODY1/2/3 parameter.
This command is similar to a "send e-mail" operation issued by a standard messaging client on a PC.
Note #SENDMAILi is used for #SENDMAIL1 or #SENDMAIL2 or #SENDMAIL3.

- **Syntax**

Command syntax: AT#SENDMAILi

Command	Possible responses
AT#SENDMAIL1 *Note : Send predefined mail #1*	OK *Note : Mail 1 has been successfully sent*
AT#SENDMAIL2 *Note : Send predefined mail #2*	OK *Note : Mail 2 has been successfully sent*
AT#SENDMAIL3 *Note : Send predefined mail #3*	OK *Note : Mail 3 has been successfully sent*
AT#SENDMAIL2	#CME ERROR: 38027 *Note: The address of the SMTP server has not been resolved by the secondary DNS server.* *TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong SMTP server address has been filled in.*

Command	Possible responses
AT#SENDMAIL1	#CME ERROR: <value> *Note: An error has occurred during the communication with the remote SMTP server. It may also happen during the data transfer (after the OK message).* *This error can be due to one of the following reason:* • *the DNS servers are not able to resolve the SMTP server address* • *the SMTP server is temporarily out of service* • *Wrong SMTP server* • *the authentication (SMTPUN, SMTPPW) is not valid* • *an e-mail address specified in REC1 or CCREC1 is not valid* • *The SMTP server does not allow the relay* See paragraph 9.2

- **List of parameters**

REC1 or REC2 or REC3
CCREC1 or CCREC2 or CCREC3
SUBJ1 or SUBJ2 or SUBJ3
BODY1 or BODY2 or BODY3
SENDERADDR
SENDERNAME
DOMAIN
SMTPPORT
SMTPSERV
SMTPPW
SMTPUN

3.4 Send host mail #PUTMAIL

- **Description**

This command allows the attached host to send an e-mail message containing body text passed to the TCP/IP stack over the serial port.
Once an IP link is established, the attached host can send an e-mail message at any time (except when the TCP/IP stack is already in a process using TCP resources).
The header of this e-mail is built using the REC1, CCREC1 and SUBJ1 parameters.
This command is similar to a "send e-mail" operation issued by a standard messaging client on a PC.

- **Syntax**

Command syntax : AT#PUTMAIL

Command	Possible responses
AT#PUTMAIL *Note: You have to configure only receiver address1, copy address1, and subject1 before or during the session, but the content (body) of the e-mails typed when the TCP/IP session is established. Content is not echoed.*	Ok_Info_WaitingForData *Note: A SMTP session has been successfully opened with the remote SMTP server.* *Once the TCP/IP stack has displayed this message, all the data received on the serial port is sent within the e-mail body.* ***The (CR)(LF)(.)(CR)(LF) sequence sent by the attached host indicates the TCP/IP stack the end of the e-mail body.***
<content> *Note: Content is not written when typing.*	

Command	Possible responses
<CR><LF> . <CR><LF> *Note: Termination sequence <CR> = <Enter>, <LF> = <Ctrl Enter>*	OK *Note : The mail has been successfully sent*
AT#PUTMAIL	#CME ERROR: 38027 *Note: The address of the SMTP server has not been resolved by the secondary DNS server.* *TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong SMTP server address has been filled in.*
AT#PUTMAIL	#CME ERROR: <value> *Note: An error has occurred during the communication with the remote SMTP server. It may also happen during the data transfer (after the OK message).* *This error can be due to one of the following reason:* • *the DNS servers are not able to resolve the SMTP server address* • *the SMTP server is temporarily out of service* • *the authentication (SMTPUN, SMTPPW) is not valid* • *an e-mail address specified in REC1 or CCREC1 is not valid* • *there has been an inactivity period of 50 seconds on the serial port* *See paragraph 9.2*

- **List of parameters**

REC1
CCREC1
SUBJ1
SENDERADDR
SENDERNAME
DOMAIN
SMTPPORT
SMTPSERV
SMTPPW
SMTPUN

3.5 Display e-mail parameters #VMAIL1 / #VMAIL2 / #VMAIL3

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the e-mail combinations, configuration.
Note: The BODY1/BODY2/BODY3 parameter values are not displayed. Their values can be displayed using the AT#BODY1? / AT#BODY2? / AT#BODY3? commands.

Note1: #VMAIL**i** is used for #VMAIL1 or #VMAIL2 or #VMAIL3.

- **Syntax**

Command syntax : AT#VMAILi

Command	Possible responses
AT#VMAIL1 *Note : View predefined (nb 1) mail header elements.*	#BODY1 : "Report from Machine 1" #CCREC1 : « maintenance@domain.com » … #CCREC1 : « « #REC1 : « alarme1@domain.com« … #REC1 : « « #SUBJ1 : «Alarm Message from 12345 « OK

- **List of parameters**

BODYi (BODY1 or BODY2 or BODY3)
CCRECi (CCREC1 or CCREC2 or CCREC3)
RECi (REC1 or REC2 or REC3)
SUBJi (SUBJ1 or SUBJ2 or SUBJ3)

3.6 Display POP3 parameters #VPOP3

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the e-mail retriever configuration.

- **Syntax**

Command syntax: AT#VPOP3

Command	Possible responses
AT#VPOP3 *Note : View POP3 parameters*	#POP3HEADERMODE: 1 #POP3PORT: 110 #POP3PW: "mypop3passwd" #POP3SERV: "pop3server" #POP3UN: "mypop3un" OK

- **List of parameters**

POP3HEADERMODE
POP3PORT
POP3PW
POP3SERV
POP3UN

3.7 Display SMTP parameters #VSMTP

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the e-mail sender configuration.

- **Syntax**

Command syntax: AT#VSMTP

Command	Possible responses
AT#VSMTP *Note : Comment on the aim of this syntax*	#DOMAIN: "a2myoper.com" #SENDERADDR: "toto@myoper.com" #SENDERNAME: "toto" #SMTPPORT: 25 #SMTPPW: "mysmtppw" #SMTPSERV: "smtp. a2myoper.com" #SMTPUN: "mysmtpun" OK

- **List of parameters**

DOMAIN
SENDERADDR
SENDERNAME
SMTPPORT
SMTPPW
SMTPSERV
SMTPUN

3.8 Display DNS parameters #VDNS

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the DNS servers configuration.

- **Syntax**

Command syntax: AT#VDNS

Command	Possible responses
AT#VDNS	#DNSSERV1: "0.0.0.0" #DNSSERV2: "0.0.0.0" OK

- **List of parameters**

DNSSERV1
DNSSERV2

4 FTP SERVICES

4.1 Parameters definition

4.1.1 FTPPORT

- **Definition**

  To reach the FTP server, the TCP/IP stack must know the control port of the FTP server used for file transfer.

- **Setting / Getting**

  Set value: AT#FTPPORT=<Value>
  Get value: AT#FTPPORT? or AT#VFTP, AT#VALL

- **Legal values**

  0 ~ 65535.

- **Default value**

  21 (twenty one)

- **Note**

  This parameter should be changed only upon request of your network administrator. It applies for network infrastructure including Firewalls, Proxy or specific TCP port translation.

4.1.2 FTPMODE

- **Definition**

  Define the FTP soft behaviour for file transfer.
  Active or Passive mode.

- **Setting / Getting**

  Set value: AT#FTPMODE=<Value>
  Get value: AT#FTPMODE? or AT#VFTP, AT#VALL

- **Legal values**

  0 : Ftp active mode
  1 : Ftp passive mode

- **Default value:**

  1 (one)

4.1.3 FTPTYPE

- **Definition**

  Before transferring files from a specified FTP server, the TCP/IP stack must specify the type of data to be transferred within the FTP session.

- **Setting / Getting**

  Set value: AT#FTPTYPE=<Value>
  Get value: AT#FTPTYPE? or AT#VFTP, AT#VALL

- **Legal values**

  - A : for FTP ASCII sessions
  - I : for FTP Binary sessions (upper case ‘i’ char)

- **Default value**

  I (upper case ‘i’ char)

- **Note**

  When this value is set to A, all the data sent by the TCP/IP stack to the FTP server is made of 7 bits characters (NVT-ASCII: the MSB is set to 0). As a consequence binary data containing 8 bits characters will be corrupted during the transfer if the FTPTYPE is set to A.

4.1.4 FTPSERV

- **Definition**

  FTP server address. To connect to an FTP server to download files, the TCP/IP stack must know the address of the FTP server that is to be used.

- **Setting / Getting**

  Set value: AT#FTPSERV=<Value>
  Get value: AT#FTPSERV? or AT#VFTP, AT#VALL

- **Legal values**

  - 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx) or
  - alphanumeric ASCII text string up to 120 characters if DNS is available

- **Default value**

  There is no default value for this parameter

4.1.5 FTPUN

- **Definition**

  Before transferring files from a specified FTP server, the TCP/IP stack must open an FTP session using a valid FTP user name

- **Setting / Getting**

  Set value: AT#FTPUN=<Value>
  Get value: AT#FTPUN? or AT#VFTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 64 characters.

- **Default value**

  There is no default value for this parameter

4.1.6 FTPPW

- **Definition**

  Before transferring files from a specified FTP server, the TCP/IP stack must open an FTP session using a valid FTP password.

- **Setting / Getting**

  Set value: AT#FTPPW=<Value>
  Get value: AT#FTPPW? or AT#VFTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 64 characters

- **Default value**

  There is no default value for this parameter

4.1.7 FTPGETFILENAME

- **Definition**

  In order to download a file from the FTP server, the TCP/IP stack must know the name of the relevant file.

- **Setting / Getting**

  Set value: AT#FTPGETFILENAME=<Value>
  Get value: AT#FTPGETFILENAME? or AT#VFTP, AT#VALL

- **Legal values**

  Alphanumeric ASCII text string up to 120 characters.

- **Default value**

  There is no default value for this parameter

4.1.8 FTPGETPATH

- **Definition**

  In order for the TCP/IP stack to get a file from the FTP server, the TCP/IP stack must know the path of the relevant file. For example, it could be: /list

- **Setting / Getting**

  Set value: AT#FTPGETPATH=<Value>
  Get value: AT#FTPGETPATH? or AT#VFTP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 120 characters.

- **Default value**

  "." : means in the root directory

- **Note**

  Depending on the FTP server, the value can be used for getting a file from the root directory of the FTP server

4.1.9 FTPPUTFILENAME

- **Definition**

  In order for the TCP/IP stack to upload a file to the FTP server, the TCP/IP stack must know the name of the relevant file.

- **Setting / Getting**

  Set value: AT#FTPPUTFILENAME=<Value>
  Get value: AT#FTPPUTFILENAME? or AT#VFTP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 120 characters.

- **Default value**

  There is no default value for this parameter

4.1.10 FTPPUTPATH

- **Definition**

  In order for the TCP/IP stack to upload a file to the FTP server, the TCP/IP stack must know the path of the relevant file. For example, it could be : /list

- **Setting / Getting**

  Set value: AT#FTPPUTPATH=<Value>
  Get value: AT#FTPPUTPATH? or AT#VFTP, AT#VALL

- **Legal values**

  Alpha-numeric ASCII text string up to 120 characters

- **Default value**

  ".": means in the root directory

- **Note**

  Depending on the FTP server, the value can be used for getting a file from the root directory of the FTP server

4.2 Get data from server #FTPGET

- **Description**

This command, sent by the attached host, directs the TCP/IP stack to connect to the specified FTP server and to retrieve the specified file from this server. Once the operation is completed, the TCP/IP stack closes the FTP connection.
Once an IP link is established, the attached host can retrieve a file from a FTP server at any time (except when the TCP/IP stack is already in a process using TCP resources).
This command is similar to a GET operation (with an automatic connect/disconnect) issued by a standard FTP client on a PC. The TCP/IP stack handles the global FTP get process by itself.

Note: Each <ETX> character present in the payload data of the FTP flow will be coded by the TCP/IP stack on the serial port as <DLE><ETX>. Each <DLE> character will be coded as <DLE><DLE>. The attached host must then decode the FTP flow to remove these escape characters.

- **Syntax**

Command syntax: AT#FTPGET

Command	Possible responses
AT#FTPGET *Note : Start data reception*	Ok_Info_DataBegin *Note : The server is ready to send data to the TCP/IP stack. For the attached host, it notifies the switch from command to data mode.* DATA *Note: The data transmitted from the FTP server to the TCP/IP stack is sent over the serial port.* <ETX> *Note: Once the file transfer finished, the TCP/IP stack sends an ETX character over the serial port to notify the attached host the end of transfer : switch from data to command mode* OK *Note: The FTP process was successfully completed.*
AT#FTPGET	#CME ERROR: 38027 *Note: The address of the FTP server has not been resolved by the secondary DNS server. The TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong FTP server address has been filled in.*
AT#FTPGET	#CME ERROR: <value> *Note: The connection to the FTP server failed (see paragraph 9.2).If this error occurs once the data transfer started, it is preceded by an ETX character*

- **List of parameters**

  FTPGETFILENAME
  FTPGETPATH
  FTPPORT
  FTPSERV
  FTPTYPE
  FTPMODE
  FTPPW
  FTPUN

4.3 Put data to server #FTPPUT

- **Description**

This command sent by the attached host directs the TCP/IP stack to connect to the specified FTP server and to upload the data received on the serial port to the specified file on this server. Once the operation has completed, the TCP/IP stack closes the FTP connection.
Once an IP link is established, the attached host can send a file to a FTP server at any time (except when the TCP/IP stack is already in a process using TCP resources).
This command is similar to a PUT operation (with an automatic connect/disconnect) issued by a standard FTP client on a PC. The TCP/IP stack handles the global FTP put process by itself.
Note: The TCP/IP stack will only interpret an <ETX> character as the end of the file to be transferred if it's not preceded by a <DLE> character. As a consequence the attached host must send <ETX> characters preceded by <DLE> characters and it must also code <DLE> characters in <DLE><DLE>.

- **Syntax**

Command syntax: AT#FTPPUT

Command	**Possible responses**
AT#FTPPUT *Note : Start data sending*	Ok_Info_WaitingForData *Note : TCP/IP stack is ready to send data from the serial port to the remote FTP server. TCP/IP stack then immediately transfers all the data sent by the attached host to the remote FTP server.* ***To notify TCP/IP stack that all data has been sent, the attached host must send the <ETX> character*** <ETX> *Note: Notification from the host for end of data : switch from data mode to command mode* OK *Note: The FTP process was successfully completed*
AT#FTPPUT	#CME ERROR: 38027 *Note: The address of the FTP server has not been resolved by the secondary DNS server. TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong FTP server address has been filled in.*
AT#FTPPUT	#CME ERROR: <value> *Note: The connection to the FTP server failed (see paragraph 9.2).If this error occurs once the data transfer started, it is preceded by an ETX character*

- **List of parameters**

FTPPUTFILENAME
FTPPUTPATH
FTPPORT
FTPSERV
FTPTYPE
FTPMODE
FTPPW
FTPUN

4.4 Display FTP parameters #VFTP

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the FTP client configuration.

- **Syntax**

Command syntax: AT#VFTP

Command	Possible responses
AT#VFTP *Note : View FTP parameters*	#FTPGETFILENAME: "" #FTPGETPATH: "." #FTPMODE: 1 #FTPPORT: 21 #FTPPUTFILENAME: "Testseb3.txt" #FTPPUTPATH: "." #FTPPW: "mypass" #FTPSERV: "mytestwebsite.com" #FTPTYPE: “I” #FTPUN: "myname" OK

- **List of parameters**

FTPGETFILENAME
FTPGETPATH
FTPMODE
FTPPORT
FTPPUTFILENAME
FTPPUTPATH
FTPPW
FTPSERV
FTPTYPE
FTPUN

5 TCP SOCKET SERVICES

Ten TCP sockets may be active simultaneously. And the id is from 1 to 10.

5.1 Parameters definition

5.1.1 DLEMODE

- **Definition**

  When performing a socket TCP, the attached host has the choice to code (or otherwise) the ETX character.

- **Setting / Getting**

  Set value: AT#DLEMODE= id,<Value>
  Get value: AT#DLEMODE=id or AT#VTCP=id, AT#VALL

- **Legal id**

  {1 - 10}

- **Legal values**

  - **0**: When DLEMODE is set to 0, no specific process is needed on [ETX] characters.
    It means that it is not possible for a host to request a end of connection or to receive a clear indication of “end of connection” from the TCP/IP stack.
  - **1**: When DLEMODE is set to 1, the [ETX] character is interpreted as a request or an indication of “end of connection”.

    As a consequence, [ETX] characters that belong to the payload data, must be sent by the host on the serial port preceded by a DLE character. Similarly ETX characters received by the TCP/IP stack from the Internet are sent to the host through the serial port preceded by a DLE character

- **Default value**

  1 (one)

5.1.2 TCPPORT

- **Definition**

  To exchange data over TCP, the TCP/IP stack must know the port of the remote peer used for the TCP session.

- **Setting / Getting**

  Set value: AT#TCPPORT= id,<Value>
  Get value: AT#TCPPORT=id or AT#VTCP=id, AT#VALL

- **Legal id**

  {1 - 10}

- **Legal values**

  0~65535

- **Default value**

  0 (zero)

5.1.3 TCPSERV

- **Definition**

  To exchange data over TCP, the TCP/IP stack must know the address of the remote TCP server (or host) that is to be used.

- **Setting / Getting**

  Set value: AT#TCPSERV=id,<Value>
  Get value: AT#TCPSERV=id or AT#VTCP=id, AT#VALL

- **Legal id**

  {1 - 10}

- **Legal values**

  - 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx) or
  - alpha-numeric ASCII text string up to 120 characters if DNS is integrated.

- **Default value**
  There is no default value for this parameter

5.1.4 TCPTXDELAY

- **Definition**

  This parameter determines the time delay introduced before sending a TCP frame that has not been entirely filled with user data. The time is entered in milliseconds and it should be noted that a value of '0' initiates the sending of a TCP frame as soon as possible after the reception of a single character value from the host.

- **Setting / Getting**

  Set value: AT#TCPTXDELAY=id,<Value>
  Get value: AT#TCPTXDELAY=id or AT#VTCP=id, AT#VALL

- **Legal id**

  {1 - 10}

- **Legal values**

  Integer multiple of 100 milliseconds between 0 and 30000 inclusive.

- **Default value**

  100 (one hundred milliseconds)

5.1.5 TCPMASK

- **Definition**

  This parameter determines the mask used for the IP filtering for an incoming TCP socket session.

- **Setting / Getting**

  Set value: AT#TCPMASK=,<Value>
  Get value: AT#TCPMASK=id or AT#VTCP=id, AT#VALL

- **Legal id**

  {1 - 10}

- **Legal values**

  IP address

- **Default value**

  0.0.0.0

5.2 Open listening mode #LTCPSTART

- **Description**

This command informs the TCP/IP stack to open a listening TCP connection on the specified TCP port. Once an IP link is established, the attached host can open a listening TCP socket at any time (except when the TCP/IP stack is already in a process using TCP resources).

The TCP connection will be active upon reception of a TCP connection request. This request is sent a valid TCP peer (as specified using the TCPSERV and TCPMASK commands) which implement TCP address and network filtering.

Description of the IP Filtering Mechanism:

Two parameters are mandatory in order to enable IP filtering when the module is entering listening mode. These are, the TCPSERV and TCPMASK parameters.

Two cases are possible:

- No IP filtering:

* If TCPMASK="0.0.0.0", there is no IP filtering no matter what the value of TCPSERV is set to.
* If TCPSERV="0.0.0.0", there is no IP filtering no matter what the value of TCPMASK is set to.
* If the TCPSERV value is not specified in the correct format (i.e. as a 32-bit number in dotted-decimal notation) there is no IP filtering no matter what the value of TCPMASK is set to.

- IP filtering:

* filtering a particular network:
TCPSERV is an IP address of a network and TCPMASK contains the IP address of the mask corresponding to the specified network. (refer to RFC 950).
* filtering a single IP address:
This enables only a single IP address to connect to the module. The correct configuration is: TCPSERV="single IP address" and TCPMASK="255.255.255.255"

Once opened, this TCP connection may be closed by the remote TCP peer or by the attached host via sending an ETX character on the serial port (depending on the DLEMODE parameter).

Note

- The LTCPSTART command can be aborted before an incoming TCP request has been received by issuing any character on the serial port

- If the DLEMODE parameter is set to 1, the TCP/IP stack will only interpret an <ETX> character from the serial port, as a close request, if a <DLE> character does not precede it.

As a consequence, the attached host must send <ETX> characters preceded by <DLE> characters and it must also code <DLE> characters in <DLE><DLE>.

Similarly, each <ETX> character in the TCP-frame payload data (Rx data) , will be coded by the TCP/IP stack on the serial port as <DLE><ETX>. Each <DLE> character will be coded as <DLE><DLE>. The attached host must then decode the TCP socket flow to remove these escape characters.

- If the DLEMODE parameter is set to 0, the TCP/IP stack does not close the TCP connection through serial port (unless an error occurs). In this case <ETX> character sent through serial port has not effect.

- **Syntax**

Command syntax: AT#LTCPSTART=id, with id one of {1 – 10}

Command	Possible responses
AT#LTCPSTART=1	OK *Notifies the host that the TCP/IP is in listening mode*
Note : listen mode previously activated. Incoming Session opening by a remote peer.	Ok_Info_WaitingForData *Note : This message signals that a remote allowed TCP peer has opened the TCP socket. The TCP connection is now opened. All the data from the attached host / remote TCP server is then immediately transferred by the TCP/IP stack to the remote TCP server / attached host. Depending on the DLEMODE value, the attached host may close this TCP connection by sending an ETX character.* *If the remote TCP server closes the connection, the TCP/IP stack issues an ETX character on the serial port.*
Session previously opened by a remote peer.	Ok_Info_SocketClosed OK *Note: The TCP socket is closed*
Note : listen mode previously activated.	Ok_Info_ListenAborted The listen mode is aborted. A Character has been sent over the serial port to end the listen mode
AT#LTCPSTART=1	#CME ERROR: <value> *Note: An error has occurred during the TCP connection. This connection is being closed. If this error occurs once the TCP connection opened, it is preceded by an ETX character. See paragraph 9.2*

- **List of parameters**

DLEMODE
TCPPORT
TCPSERV
TCPTXDELAY
TCPMASK

5.3 Close listening mode

- **Description**

The listening mode is stopped as soon as one character is sent over the serial port.
The TCP/IP sends information over the serial port: Ok_Info_ListenAborted

5.4 Open TCP connection #OTCP

- **Description**

This command sent by the attached host directs the TCP/IP stack to open a TCP connection to the specified TCP server.

Once an IP link is established, the attached host can open a TCP connection at any time (except when the TCP/IP stack is already in a process using TCP resources).

This TCP connection may be closed by the remote TCP server or by the attached host by sending an ETX character on the serial port (**depending on the DLEMODE parameter**).

Notes on DLEMODE value:

Depending on the DLEMODE value, the attached host may close this TCP connection by sending an ETX character.

- If the DLEMODE parameter is set to 1, the TCP/IP stack will only interpret an <ETX> character as a close request if it's not preceded by a <DLE> character. As a consequence the attached host must send <ETX> characters preceded by <DLE> characters and it must also code <DLE> characters in <DLE><DLE>. Similarly, each <ETX> character present in the payload data of the TCP frame, will be coded by the TCP/IP stack on the serial port as <DLE><ETX>. Each <DLE> character will be coded as <DLE><DLE>. The attached host must then decode the TCP socket flow to remove these escape characters.

- If the DLEMODE parameter is set to 0, the TCP/IP stack will never close the TCP connection (unless an error occurs).

If the remote TCP server closes the connection, the TCP/IP stack sends an ETX character on the serial port.

- **Syntax**

Command syntax: AT#OTCP =id, with id one of{1 – 10}

Command	**Possible responses**
AT#OTCP=1 *Note : Request opening of the TCP socket 1*	Ok_Info_WaitingForData *Note : This message signals that the TCP socket has been opened. All the data from the attached host / remote TCP server is then immediately transferred by the TCP/IP stack to the remote TCP server / attached host. Depending on the DLEMODE value, the attached host may close this TCP connection by sending an ETX character. If the remote TCP server closes the connection, the TCP/IP stack issues an ETX character on the serial port.*
	Ok_Info_SocketClosed OK *Note: The TCP socket is closed*
AT#OTCP=2	#CME ERROR: 38027 *Note: The address of the server has not been resolved by the secondary DNS server. TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong server address has been filled in.*
AT#OTCP=2	#CME ERROR: <value> *Note: An error has occurred during the TCP connection. This connection is being closed. If this error occurs once the TCP connection opened, it is preceded by an ETX character. See paragraph 9.2*

- **List of parameters**

DLEMODE
TCPPORT
TCPSERV
TCPTXDELAY
TCPMASK

5.5 Display TCP parameters #VTCP

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the TCP socket configuration.

- **Syntax**

Command syntax: AT#VTCP =id, with id one of{1 – 10}

Command	Possible responses
AT#VTCP=2 *Note : View TCP parameters of the TCP socket 2*	#DLEMODE: 2,1 #TCPPORT: 2,0 #TCPSERV: 2,"" #TCPTXDELAY: 2,1 #TCPMASK: 2,"0.0.0.0" OK

- **List of parameters**

DLEMODE
TCPSERV
TCPPORT
TCPTXDELAY
TCPMASK

6 UDP SOCKET SERVICES

6.1 Parameters definition

6.1.1 UDPPORT

- **Definition**

  - Local UDP port number if UDP session is initiated in listen mode
  - Remote UDP number if UDP session is initiated in active mode.

- **Setting / Getting**

  Set value: AT#UDPPORT=<Value>
  Get value: AT#UDPPORT? or AT#VUDP, AT#VALL

- **Legal values**

  0 ~ 65535.

- **Default value**

  0 (zero)

6.1.2 UDPSERV

- **Definition**

  - IP address filter if the UDP session is initiated in listen mode. This means that the remote must have a defined UDPSERV IP address.
  - remote IP address if the UDP session is initiated in active mode.

  Note: no IP filter is applied if parameter value is “255.255.255.255”

- **Setting / Getting**

  Set value: AT#UDPSERV=<Value>
  Get value: AT#UDPSERV? or AT#VUDP, AT#VALL

- **Legal values**

  - 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx) or
  - alpha-numeric ASCII text string up to 120 characters if DNS is integrated.

- **Default value**

  There is no default value for this parameter

6.1.3 UDPTXDELAY

- **Definition**

  This parameter determines the delay before sending a UDP data-gram that has not been entirely filled with user data. The delay is expressed in milliseconds.

The '0' value initiates the sending of a UDP data-gram as soon as possible after the reception of a single character value from the host.

- **Setting / Getting**

  Set value: AT#UDPTXDELAY=<Value>
  Get value: AT#UDPTXDELAY? or AT#VUDP, AT#VALL

- **Legal values**

  Integer, multiple of 100 milliseconds and between 0 and 30000 inclusive.

- **Default value**

  100 (100 milliseconds)

6.1.4 UDPMASK

- **Definition**

  This parameter determines the mask used for the IP filtering for an incoming UDP socket session.

- **Setting / Getting**

  Set value: AT#UDPMASK=<Value>
  Get value : AT#UDPMASK？ or AT#VUDP, AT#VALL

- **Legal id**

- **Legal values**

  IP address

- **Default value**

  0.0.0.0

6.2 Open a UDP session in listen mode #LUDPSTART

- **Description**

This command informs the TCP/IP stack to open a listening UDP connection on the specified UDP port. Once an IP link is established, the attached host can open a listening UDP socket at any time (except when the TCP/IP stack is already in a process using UDP resources).

The UDP connection will be active upon reception of a UDP connection request. This request is sent a valid UDP peer (as specified using the UDPSERV and UDPMASK commands) which implement UDP address and network filtering.

Description Of The IP Filtering Mechanism:

Two parameters are mandatory in order to enable IP filtering when the module is entering listening mode. These are, the UDPSERV and UDPMASK parameters.

Two cases are possible:

- No IP filtering:

* If UDPMASK="0.0.0.0", there is no IP filtering no matter what the value of UDPSERV is set to.
* If UDPSERV="0.0.0.0", there is no IP filtering no matter what the value of UDPMASK is set to.
* If the UDPSERV value is not specified in the correct format(i.e. as a 32-bit number in dotted-decimal notation) there is no IP filtering no matter what the value of UDPMASK is set to.

- IP filtering:

* filtering a particular network:
UDPSERV is an IP address of a network and UDPMASK contains the IP address of the mask corresponding to the specified network. (refer to RFC 950).
* filtering a single IP address:
This enables only a single IP address to connect to the module. The correct configuration is: UDPSERV="single IP address" and UDPMASK="255.255.255.255"

Once opened, this UDP connection may be closed by the remote UDP peer or by the attached host via sending an ETX character on the serial port (depending on the DLEMODE parameter).

Note

- The LUDPSTART command can be aborted before an incoming UDP request has been received by issuing any character on the serial port

- If the DLEMODE parameter is set to 1, the TCP/IP stack will only interpret an <ETX> character from the serial port, as a close request, if a <DLE> character does not precede it.

As a consequence, the attached host must send <ETX> characters preceded by <DLE> characters and it must also code <DLE> characters in <DLE><DLE>.

Similarly, each <ETX> character in the UDP-datagram payload data (Rx data), will be coded by the TCP/IP stack on the serial port as <DLE><ETX>. Each <DLE> character will be coded as <DLE><DLE>. The attached host must then decode the TCP socket flow to remove these escape characters.

- If the DLEMODE parameter is set to 0, the TCP/IP stack does not close the TCP connection through serial port (unless an error occurs). In this case <ETX> character sent through serial port has not effect.

- **Syntax**

Command syntax: AT#LUDPSTART

Command	Possible responses
AT#LUDPSTART	OK *Notifies the host that the listen mode is activated*
Note: listen mode previously activated. Incoming UDP datagrams sent from a remote host	Ok_Info_WaitingForData *Note : This message signals that a remote allowed UDP peer has sent its first data-gram to local UDP socket. The UDP connection is now effective. All data from the attached host / remote UDP peer is immediately transferred by the TCP/IP stack to the remote UDP peer / attached host. The attached host may close this UDP session by sending an <ETX> character.* *If an error occurs, the TCP/IP stack issues an <ETX> character on the serial port before sending the appropriate error message.*
Note: listen mode previously activated. Incoming UDP datagrams sent from a remote host	Ok_Info_WaitingForData … … … <ETX> Ok_Info_SocketClosed OK *Note: The UDP socket is closed.*
Note: listen mode previously activated.	Ok_Info_ListenAborted The listen mode is aborted. A Character has been sent over the serial port to end the listen mode
AT#LUDPSTART	#CME ERROR: <value> *Note: An error has occurred during the UDP session creation. If this error occurs once the UDP session is effective, it is preceded by an <ETX> character. See paragraph 9.2*

- **List of parameters**

UDPPORT
UDPSERV
UDPTXDELAY
UDPMASK

6.3 Close a listening mode UDP session

- **Description**

Any character sent over the serial port directs the TCP/IP stack to stop the a UDP listening mode state (previously launched by the AT#LUDPSTART command).
The TCP/IP sends an information over the serial port: Ok_Info_ListenAborted

6.4 Open an active UDP session #OUDP

- **Description**

Once an IP link is established, and if no other TCP/IP stack resource is active, this command sent by the attached host directs the TCP/IP stack to initiate a UDP session in active mode on the specified UDP remote port UDPPORT to the specified remote IP address UDPSERV.
The host can then transmit to the UDPPORT or UDPSERV address. Data-grams can only be received from this UDPPORT or UDPSERV address.

The host can stop the UDP session by issuing an <ETX> character.

Note: The TCP/IP stack will only interpret an <ETX> character as a close request if it is not preceded by a <DLE> character. As a consequence, an <ETX> character must be sent (even in payload data) as <DLE><ETX>, and <DLE> character as <DLE><DLE>. The attached host must then decode the UDP socket flow to remove these escape characters.

- **Syntax**

Command syntax: AT#OUDP

Command	**Possible responses**
AT#OUDP *Note : Request opening of UDP socket*	Ok_Info_WaitingForData *Note : This message signals that the UDP socket has been opened. All data from the attached host / remote UDP peer is immediately transferred by the TCP/IP stack to the remote UDP peer / attached host. The attached host may close this UDP session by sending an <ETX> character.* *If an error occurs, the TCP/IP stack issues an <ETX> character on the serial port before sending the appropriate error message.*
AT#OUDP	Ok_Info_WaitingForData … … … <ETX> Ok_Info_SocketClosed OK *Note: The UDP socket is closed.*
AT#OUDP	#CME ERROR: 38027 *Note: The address of the remote UDP peer has not been resolved by the secondary DNS server. TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong remote UDP peer address has been filled in.*
AT#OUDP	#CME ERROR: <value> *Note: An error has occurred during the UDP session creation. If this error occurs once the UDP session is effective, it is preceded by an <ETX> character. See paragraph 9.2*

- **List of parameters**

UDPPORT
UDPSERV
UDPTXDELAY
UDPMASK

6.5 Display UDP parameters #VUDP

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the UDP socket configuration.

- **Syntax**

Command syntax: AT#VUDP

Command	Possible responses
AT#VUDP *Note : View UDP parameters*	#UDPPORT: 0 #UDPSERV: "" #UDPTXDELAY: 1 #UDPMASK: "0.0.0.0" OK

- **List of parameters**

UDPPORT
UDPSERV
UDPTXDELAY
UDPMASK

7 PING SERVICES

7.1 Parameters definition

7.1.1 PINGDELAY

- **Definition**

  Waiting delay, in seconds, before an echo request is considered as not replied to.

- **Setting / Getting**

  Set value: AT#PINGDELAY=<Value>
  Get value: AT#PINGDELAY? or AT#VPING, AT#VALL

- **Legal values**

  From 1 to 255 inclusive.

- **Default value**

  1 (one)

7.1.2 PINGNUM

- **Definition**

  Number of PING echo requests to issue to PINGREMOTE.

- **Setting / Getting**

  Set value: AT#PINGNUM=<Value>
  Get value: AT#PINGNUM? or AT#VPING, AT#VALL

- **Legal values**

  From 1 to 255 inclusive.

- **Default value**

  4 (four)

7.1.3 PINGREMOTE

- **Definition**

  IP address or alpha-num ASCII text string, up to 120 characters long if DNS is available.

- **Setting / Getting**

  Set value: AT#PINGREMOTE=<Value>
  Get value: AT#PINGREMOTE? or AT#VPING, AT#VALL

- **Legal values**

  - 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx) or
  - alpha-numeric ASCII text string up to 120 characters long if DNS is integrated.

- **Default value**

  There is no default value for this parameter

7.2 Start PING request

- **Description**

Once an IP link is established, and if no other TCP/IP stack resource is active, this command, sent by the attached host, directs the TCP/IP stack to start PING requests.

- **Syntax**

Command syntax: AT#PING

Command	**Possible responses**
AT#PING	"yahoo.com" is alive : time = 900 ms " yahoo.com " is alive : time = 900 ms " yahoo.com " is alive : time = 800 ms " yahoo.com " is alive : time = 800 ms " yahoo.com " is alive : time = 1000 ms No answer from "yahoo.com " " yahoo.com " is alive : time = 900 ms " yahoo.com " is alive : time = 800 ms " yahoo.com " is alive : time = 900 ms " yahoo.com " is alive : time = 900 ms OK *Note: TCP/IP stack sends PINGNUM = 10 request*
AT#PING	#CME ERROR: 38027 *Note: The address of the remote has not been resolved by the DNS servers. TCP/IP stack is not able to reach the primary and secondary DNS servers or a wrong remote address has been fill in*
AT#PING	#CME ERROR: <value> *Note: An error has occurred during the PING requests* *Please refer to paragraph 9.2*

- **List of parameters**

PINGDELAY
PINGNUM
PINGREMOTE

7.3 Display PING parameters #VPING

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters related to the PING configuration.

- **Syntax**

Command syntax: AT#VPING

Command	Possible responses
AT#VPING *Note : View PING parameters*	#PINGDELAY: 1 #PINGNUM: 4 #PINGREMOTE: “” OK

- **List of parameters**

PINGDELAY
PINGNUM
PINGREMOTE

8 MISCELLANEOUS

8.1 Display software version #VVERSION

- **Description**

This command directs the TCP/IP stack to display the software version.

- **Syntax**

Command syntax: AT#VVERSION
Response syntax: *<Filename> – <File size> – <Date and time of generation>*

Command	Possible responses
AT#VVERSION *Note : Request TCP/IP stack version*	#EDLIB VERSION: edLib_V0.1 Jul 31 2007 04:56:28 OK

- **List of parameters**

No TCP/IP parameters.

8.2 Display current status #VSTATE

- **Description**

This command directs the TCP/IP stack to display the current status of the GSM/GPRS module product.

- **Syntax**

Command syntax: AT#VSTATE

Command	Possible responses
AT#VSTATE	#STATE: “IDLE” OK *Note : Idle state*
AT#VSTATE	#STATE: “DIALING” OK *Note : Dialling the ISP. Not yet connected*
AT#VSTATE	#STATE: “AUTHENTICATING” OK *Note : Connection. Not yet PPP negotiated (PPP OK message)*
AT#VSTATE	#STATE: “CONNECTED” OK *Note : Connected to Internet. An IP address has been attributed to the TCP/IP stack*
AT#VSTATE	# STATE: “NO SERVICE” *Note: No radio connection made*
AT#VSTATE	#STATE: “DISCONNECTING” *Note: In the process of disconnecting from the current communication process.*

Command	Possible responses
AT#VSTATE	#STATE:"CHECKING" *Note: Either, in the process of setting up an outgoing communication session over GSM or GPRS, or in the process of answering an incoming call (manually or automatically) over GSM.*

- **List of parameters**

No TCP/IP parameters.

8.3 Display all parameters #VALL

- **Description**

This command directs the TCP/IP stack to display all the AT# parameters.
The parameters are displayed by blocks of categories separated by a <CR><LF> sequence, all at the same time.

- **Syntax**

Command syntax: AT#VALL

Command	Possible responses
AT#VALL	#EDLIB VERSION: edLib_V0.1Jul 31 2007 04:56:28
	#ANSWERMODE: 0 #CALLBACKTIMER: 2 #CALLSCREENNUM: "0" #DIALN1: "" #DIALN2: "" #DIALSELECT: 1 #GPRSMODE: 1 #PHYTIMEOUT: 15 #REDIALCOUNT: 5 #REDIALDELAY: 5 #RINGCOUNT: 0 #CAUTO: 0
	#ISPUN: "" #ISPPW: "" #PPPMODE: 1 #PPPMYIP: "0.0.0.0" #PPPPEERIP: "0.0.0.0" #PPPSERVPW: "" #PPPSERVUN: ""
	#APNPW: "access" #APNSERV: "a2bouygtel.com" #APNUN: "a2b"
	#DNSSERV1: "0.0.0.0" #DNSSERV2: "0.0.0.0"

Command	Possible responses
	#FTPGETFILENAME: "" #FTPGETPATH: "." #FTPMODE: 1 #FTPPORT: 21 #FTPPUTFILENAME: "" #FTPPUTPATH: "." #FTPPW: "mypassword" #FTPSERV: "mytestwebsite.com" #FTPTYPE: "I" #FTPUN: "mylogin"
	#POP3HEADERMODE: 1 #POP3PORT: 110 #POP3PW: "" #POP3SERV: "" #POP3UN: ""
	#DOMAIN: "a2bouygtel.com" #SENDERADDR: "toto@bouygtel.com" #SENDERNAME: "toto" #SMTPPORT: 25 #SMTPPW: "" #SMTPSERV: "smtp.a2bouygtel.com" #SMTPUN: ""
	#BODY1: "" #CCREC1: "" #REC1: "lesavecom.com" #SUBJ1: "Test"
	#BODY2: "" #CCREC2: "" #REC2: "" #SUBJ2: ""
	#BODY3: "9:1234567890abcd" #CCREC3: "" #REC3: "" #SUBJ3: ""

Command	Possible responses
	#DLEMODE: 1,1 #TCPPORT: 1,0 #TCPSERV: 1,"" #TCPTXDELAY: 1,1 #TCPMASK: 1,"0.0.0.0" #DLEMODE: 2,1 #TCPPORT: 2,0 #TCPSERV: 2,"" #TCPTXDELAY: 2,1 #TCPMASK: 2,"0.0.0.0" … #DLEMODE: 10,1 #TCPPORT: 10,0 #TCPSERV: 10,"" #TCPTXDELAY: 10,1 #TCPMASK: 10,"0.0.0.0"
	#UDPPORT : 0 #UDPSERV : "" #UDPTXDELAY:1 #UDPMASK: "0.0.0.0"
	#PINGDELAY:1 #PINGNUM: 4 #PINGREMOTE: "" OK

- **List of parameters**

All parameters.

8.4 Erase flash memory parameter values #DELFLASH

- **Description**

This command erases the contents of parameter flash memory. Used prior to a 'hard reset' of the module, it results in the default values being written into the parameter memory.
Note: the current parameter values remain visible until the 'hard reset' has completed.

- **Syntax**

Command syntax: AT#DELFLASH

Command	Possible responses
AT#DELFLASH *Note : Delete flash memory contents*	OK

- **List of parameters**

No TCP/IP parameters.

9 RESPONSE MESSAGES AND ERROR CODES

9.1 Response messages

Standard AT messages		
Numeric	**Verbose**	**Description**
0	OK	Operation or command success
3	NO CARRIER	No physical layer connection
7	BUSY	Destination busy
8	NO ANSWER	No answer from destination
4	ERROR	Operation or command unsuccessful
2	RING	Incoming call indication
10	CONNECT 300	Physical layer connected at 300 baud
11	CONNECT 1200	Physical layer connected at 1200 baud
12	CONNECT 1200/75	Physical layer connected at 1200/75 baud
13	CONNECT 2400	Physical layer connected at 2400 baud
14	CONNECT 4800	Physical layer connected at 4800 baud
15	CONNECT 9600	Physical layer connected at 9600 baud
16	CONNECT 14400	Physical layer connected at 14400 baud

Information messages		
Numeric	**Verbose**	**Description**
1025	Ok_Info_DataBegin	Start of data
1028	Ok_Info_WaitingForData	Send data
3074	Ok_Info_SocketClosed	Socket connection closed successfully
3072	Ok_Info_NoMail	No mail to retrieve on server
3073	Ok_Info_Mail	Mail ready to be retrieved on server
3077	Ok_Info_Ppp	PPP connection successful
3086	Ok_Info_GprsActivation	GPRS connection successful
3087	Ok_Info_ListenAborted	Listen mode aborted by the host (Socket UDP and socket TCP)

9.2 Error codes

Error codes		
Numeric	**Verbose**	**Description**
34817	Invalid command: unknown command	Invalid command : Unknown command
34819	Invalid command: syntax error	Invalid command : Syntax error
34824	Invalid command: writing failed	Invalid command : Writing failed
34881	Invalid command: command too long	Invalid command : Command too long
34882	Invalid command: invalid argument value	Invalid command : Bad command argument value
35840	Physical layer: modem already running	Physical layer : Modem is already running
35841	Physical layer: GPRS session lost	Physical layer : GPRS session lost.
35842	Physical layer: open GPRS session request failed	Physical error : Open GPRS session request failed

Error codes		
Numeric	**Verbose**	**Description**
35843	Physical layer: GPRS authentication failed	Physical error : GPRS authentication failed : wrong APN username or/and password
35862	Physical layer: inactivity timeout	Physical layer : inactivity timeout, no activity on network connection
35865	Physical layer: module not attached to the network	Physical layer : Module is not attached to the network
35866	Physical layer: invalid event during activation process	Physical layer : Invalid event during the activation process
35867	Physical layer: connection not active	Physical layer : Physical layer connection is currently not active
35868	Physical layer: GPRS connection aborted	Physical layer : GPRS connection aborted
35869	Physical layer: invalid incoming call type	Physical layer : Invalid incoming call type
35870	Physical layer: incoming call CLI not provided	Physical layer : Incoming call CLI not provided
35871	IP library: SIM removed	IP Connectivity library: SIM removed
36872	IP library: internal error - resource not available	IP Connectivity library internal error : internal resource not available.
36929	IP library: invalid parameter configuration attempt	IP Connectivity library : invalid parameter configuration attempt
37120	IP library: PPP negotiation failed - client configuration	IP Connectivity library : PPP negotiation failed (client configuration)
37121	IP library: PPP negotiation failed - server configuration	IP Connectivity library: PPP negotiation failed (server configuration)
37122	IP library: another internal application is running	IP Connectivity library: Another internal application is already running
37123	IP library: service running - unable to set parameters	IP Connectivity library: Service is running. Unable to set parameter
37124	IP library: data buffer oversized	IP Connectivity library: Data Buffer oversized
37125	IP library: no UDP datagram received	IP Connectivity library: No UDP datagram received
37952	Distant: TCP session closed - TCP context cancelled	Distant : TCP session closed (TCP Context cancelled)
37964	Distant: no response from server	Distant : No response from server
37966	Distant: TCP session closed by peer	Distant : TCP session closed by peer (FIN received from peer)
38016	Distant: session opening attempt failed	Distant : Session opening attempt failed
38017	Distant: data sending attempt failed	Distant : Data sending attempt failed
38018	Distanet: session closing attempt failed	Distant : Session closing attempt failed
38022	Distant: directory change attempt failed	Distant : Directory change attempt failed
38023	Distant: file deletion attempt failed	Distant : File deletion attempt failed
38024	Distant: data retrieval attempt failed	Distant : Data retrieval attempt failed
38025	Distant: email retrieval attempt failed	Distant : Email retrieval attempt failed
38026	Distant: email header reception failed	Distant : Email header reception failed
38027	Distant: no answer from DNS server	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	Distant : Sender email address rejected by server
38029	Distant: recipient email address rejected by server	Distant : Recipient email address rejected by server

Error codes		
Numeric	**Verbose**	**Description**
38030	Distant: cc recipient email address rejected by server	Distant : CC Recipient email address rejected by server
38031	Distant: email body sending request rejected by server	Distant : Email body sending request rejected by server
38080	Distant: username rejected by server	Distant : Username rejected by server
38081	Distant: password rejected by server	Distant : Password rejected by server
38082	Distant: invalid format response from server	Distant: the response from the remote server is not in a valid format.
38083	Distant: invalid POP3 banner format	Distant: POP3 banner sent by the server is not in a valid format
38084	Distant: POP3 Stat command failed	Distant: POP3 STAT command failed in the remote server
38085	Distant: POP3 Retr command failed	Distant: POP3 RETR command failed in the remote server
38086	Distant: POP3 Dele command failed	Distant : POP3 DELETE command failed in the remote server
38087	Distant: POP3 Quit command failed	Distant: POP3 QUIT command failed in the remote server
38088	Distant: FTP authentication failed	Distant: FTP Authentication failed in the remote server: user or/and password rejected
38089	Distant: SMTP authentication mechanism not supported	Distant: SMTP Authentication mechanism not supported by the server
38980	IP library: PPP timeout - client configuration	IP Connectivity library: PPP timeout (client configuration)
38981	IP library: PPP timeout – server	IP Connectivity library: PPP timeout (server configuration)

10 APPENDIX A - GETTING STARTED WITH EXAMPLES

10.1 Get Software Version

Commands	Responses
AT#VVERSION *Note: check the TCP/IP stack version*	#EDLIB VERSION : "edLib_V0.1 Jul 31 2007 04:56:28"

10.2 AT# Interface

Commands	Responses
AT#PPPmode=1 *Note: Set a parameter value*	OK
AT#PPPmode? *Note: Request a parameter value*	#PPPMODE: 1 OK

10.3 GSM Network Registration

Commands	Responses
AT+CPIN=xxxx *Note: Set a the pin code*	OK
AT+CREG=1 *Note: Ask for registration state*	OK +CREG: 1

10.4 GPRS Network Registration

Commands	Responses
AT+CGREG=1 *Note: Ask for GPRS registration state*	OK +CGREG:0
AT+CGATT=1 *Note: GPRS attach*	+CGREG: 2 +CGREG: 1

10.5 Connection to the Internet

10.5.1 Connection to the Internet Service Provider using GSM data:

Commands	Responses
AT#GPRSMODE=0 *Note: Activate the GSM data mode*	OK
AT#DIALN1=”0860000000” *Note: Set the ISP phone number*	OK
AT#ISPUN=”login” *Note: Set the ISP login*	OK
AT#ISPPW=”password” *Note: Set the ISP password*	OK
AT#DNSSERV1=”xxx.xxx.xxx” *Note: Set the DNS server address (coming with the ISP subscription)*	OK
AT#ConnectionStart *Note:* ***Launch the ISP connection***	DIALING 0860000000 *Note: The remote modem answers to the call.* CONNECT 9600 213.30.30.30 *Note: Dynamic IP address attributed by the network* Ok_Info_Ppp *Note: The PPP link is established.*
AT#ConnectionStop *Note: When required, stop the Internet connection*	OK

10.5.2 Connection to the Internet Service Provider using GPRS:

Commands	Responses
AT#GPRSMODE=1 *Note: Activate the GPRS mode*	OK
AT#APNSERV=”apnserver.com” *Note: Set the APN server (GPRS provider)*	OK
AT#APNUN=”login” *Note: Set the APN login*	OK
AT#APNPW=”password” *Note: Set the APN password*	OK
AT#ConnectionStart *Note: Launch the GPRS connection*	213.30.30.30 *Note: Dynamic IP address attributed by the network* Ok_Info_GprsActivation *Note: The GPRS/IP link is established.*
AT#ConnectionStop *Note: When required, stop the Internet connection*	OK

10.6 Sending/retrieving email

10.6.1 Sending an email: AT#PutMail

Commands	Responses
AT#SMTPSERV=”smtp.domain.com” *Note: SMTP server used*	OK
AT#DOMAIN=”domain.com” *Note: Domain name*	OK
AT#SENDERNAME=”Test module” *Note: Sender Name*	OK
AT#SENDERADDR”module@domain.com” *Note: Sender email address*	OK
AT#REC1=recipient@domain.com *Note: Recipient email address*	OK
AT#CCREC1=ccrecipient@domain.com *Note: Carbon Copy recipient*	OK
AT#SUBJ1=”Email Subject” *Note: Email Subject*	OK
AT#PUTMAIL *Note: **Send an email** (type the email text, and then the end sequence)*	Ok_Info_WaitingForData *Note: The software is ready to receive incoming data (not echoed)* *At the end of data , the [CR][LF].[CR][LF] sequence ends the email. This sequence can be sent by a keyboard by :* *ENTER* *CTRL+ENTER* *.* *ENTER* *CTRL+ENTER*
. *Note: end sequence sent*	OK *Note: The email is successfully sent*

10.6.2 Retrieving an email: AT#Getmail

Commands	Responses
AT#POP3SERV=”pop3.domain.com” *Note: POP3 server used*	OK
AT#POP3UN=”module@domain.com” *Note: POP3 username (not always the complete email address). It is the POP3 login*	OK
AT#POP3PW=”password” *Note: POP3 password*	OK
AT#GETMAIL *Note: **Retrieve an email***	Ok_Info_Mail *Note: The software switches from command mode to data mode for receiving the email content.* Data Data . *Note: At the end of data , the [CR][LF].[CR][LF] sequence notifies the end of the data mode* OK *Note: The email is successfully retrieved*

10.7 FTP: Download / upload files

10.7.1 Upload a file to a FTP server : AT#FTPput

Commands	Responses
AT#FTPSERV=”pop3.domain.com” *Note: FTP server used*	OK
AT#FTPUN=”ftplogin” *Note: FTP username*	OK
AT#FTPPW=”ftppassword” *Note: FTP password*	OK
AT#FTPPUTFILENAME=”upload.txt” *Note: Name of the file that will be written in the FTP server*	OK
AT#FTPPUTPATH=”.” *Note: Path in the server where the file will be written.*	OK
AT#FTPput *Note: **FTP put***	Ok_Info_WaitingForData *Note: Switch from command to data mode. The host can send the data that will compose the file. (Data not echoed).* *To notify the end of data, the host has to send the [ETX] character (CTRL+C in a keyboard). This character is echoed.*
# *Note: End of data notified*	OK

10.7.2 Download a file from a FTP server : AT#FTPget

Commands	Responses
AT#FTPSERV=”pop3.domain.com” *Note: FTP server used*	OK
AT#FTPUN=”ftplogin” *Note: FTP username*	OK
AT#FTPPW=”ftppassword” *Note: FTP password*	OK
AT#FTPGETFILENAME=”upload.txt” *Note: Name of the file stored in the FTP server*	OK
AT#FTPget *Note: **FTP get***	Ok_Info_DataBegin *Note: Switch from command to data mode. The data are sent over the serial port* *Note: The end of data is notified by the [ETX] character sent over the serial port. It switches from the data to command mode.* OK

10.8 TCP Socket

Open a TCP socket between two machines. One machine acts as a caller (TCP client), and one as a listener (TCP server). Both machines have to be connected to the Internet and set on the same TCP port.

10.8.1 Act as a TCP server: AT#LtcpStart

Commands	Responses
AT#TCPSERV=1,"255.255.255.255" *Note: No filter of the incoming TCP client*	OK
AT#TCPPORT=1,23 *Note: TCP port between the TCP client and the TCP server must be the same.*	OK
AT#ltcpstart=1 *Note: **Launch the listening mode**, waiting for an incoming TCP connection from a TCP client.*	OK Ok_Info_WaitingForData *Note: Message sent over the serial port in case of successful TCP socket opening (Telnet for example in the IP address of the GSM/GPRS module)*
Data *Note: Data flow is bi-directional.*	Data *Note: Data flow is bi-directional.*
# *Note: The socket can be closed locally by the attached host sending an [ETX] character (CTRL+C in a keyboard).*	OK
	# *Note: The socket can be closed by the remote* Ok_Info_SocketClosed OK

Note: the closing of the socket is performed either locally or remotely (refer to only one of the two closing examples provided above).

10.8.2 Act as a TCP client: AT#OTCP

Commands	Responses
AT#TCPSERV=1,"xxx.xxx.xxx.xxx" *Note: No filter of the incoming TCP client*	OK
AT#TCPPORT=1,23 *Note: TCP port between the TCP client and the TCP server must be the same.*	OK
AT#otcp=1 *Note: **Open, as a TCP client, a socket TCP** with a remote TCP server*	Ok_Info_WaitingForData *Note: Message notifying the socket opening and the switch in data mode.*
Data *Note: Data flow is bi-directional.*	Data *Note: Data flow is bi-directional.*
# *Note: The socket can be closed locally by the attached host sending an [ETX] character (CTRL+C in a keyboard).*	OK
	# *Note: The socket can be closed by the remote* Ok_Info_SocketClosed OK

Contact us:

Shanghai SIMCom Ltd.
Add: SIM Technology Building, No. 700, Yishan Road, Shanghai,P. R. China 200233
Tel: +86 21 5427 8900
Fax: +86 21 5427 6035
URL: www.sim.com