AT Command Manual
NL-SW-LTE-GELS3

NimbeLink Corp

Updated: January 2016

1. Introduction

1.1 Overview

The NimbeLink Skywire NL-SW-LTE-GELS3 end-device certified modem accepts AT commands via the UART and USB COM ports. This document is an overview of the AT commands available to users and provides a high-level summary of their individual capabilities. Many of the AT commands are from the 3GPP 27.007 version 9.7.0 and 27.005 specifications, so for more details regarding each command, you can access the full 3GPP specificatiosn online at www.3GPP.org.

Additionally, the Skywire modem is built upon the Gemalto ELS3 module, Gemalto publishes a complete AT command manual which is only available from Gemalto under NDA.

2. AT Commands

2.1 General

2.1.1 AT+CGMI Manufacturer identity

Command	Possible response(s)
+CGMI	*Cinterion*
	+CME ERROR: <err>
+CGMI=?	*OK*

2.1.2 AT+CGMM Request model identification

Command	Possible response(s)
+CGMM	*ELS31-V*
	+CME ERROR: <err>
+CGMM=?	*OK*

2.1.3 AT+CGMR Request revision identification

Command	Possible response(s)
+CGMR	*ASW4.3.1 [21791] | MFW4.3.1.0c*
	+CME ERROR: <err>
+CGMR=?	*OK*

2.1.4 AT+CGSN Request product serial number

Command	Possible response(s)
+CGSN	*Returns IMEI for example: 004401081656742*
	+CME ERROR: <err>
+CGMR=?	*OK*

2.1.5 AT+CIMI Request IMSI

Command	Possible response(s)
+CIMI	*Returns IMSI for example: 311480239321903*
	+CME ERROR: <err>
	ERROR response may indicate SIM not present
+CIMI=?	*OK*

2.1.6 AT+CSCS Select TE Character Set

Command	Possible response(s)
+CSCS=[<chset>]	
+CSCS?	*Returns current setting*
	+CSCS: "UTF-8"

2.1.7 AT+WS46 PCCA ST-101 Select wireless network

Command	Possible response(s)
+WS46=[<n>]	<n>: integer type 28 E-UTRAN only

+WS46?	*Returns current setting*
	28
+WS46=?	*Returns valid settings*

2.2 Call Control

2.2.1 AT+CEER Extended Error Report

Command	Possible response(s)
+CEER	*+CEER <report>*
+CEER=?	*OK*

2.3 Network Service Commands

2.3.1 AT+CLCK Facility Lock

Command	Possible response(s)
+CLCK=<fac>,<mode>,[,<password>[,<class>]]	*Only SC facility supported* *<fac> values reserved by the present document:* *"SC" SIM (lock SIM/UICC card) (SIM/UICC asks password in MT power-up and when this lock command issued)* *<mode>: integer type* *0 unlock* *1 lock* *2 query status* *<status>: integer type* *0 not active* *1 active* *<passwd>: string type; shall be the same as password specified for the facility from the MT user interface or with command Change Password +CPWD* *<classx> is a sum of integers each representing a class of information (default 7 - voice, data and fax):* *1 voice (telephony)* *2 data (refers to all bearer services; with <mode>=2 this may refer only to some bearer service if TA does not*

	support values 16, 32, 64 and 128) *4 fax (facsimile services)* *8 short message service* *16 data circuit sync* *32 data circuit async* *64 dedicated packet access* *128 dedicated PAD access*
+CLCK=?	*+CLCK: ("SC","FD","PN")*

2.3.2 AT+CNUM Subscriber number

Command	Possible response(s)
+CNUM	*+CNUM: <assigned phone number>*
	OK
+CNUM=?	*OK*

2.3.3 AT+COPN Read Operator Names

Command	Possible response(s)
+COPN	*+COPN: <number>, <string>*
	+COPN: "311480",Verizon Wireless
+COPN=?	*OK*

2.3.4 AT+COPS PLMN selection

Command	Possible response(s)
+COPS=[<mode>[,<format>[,<oper>[,<AcT>]]]]	*+CME ERROR: <err>*
	+COPS: 0,2,"311480",7
+COPS=?	*OK*

2.3.5 AT+CPWD Change password

Command	Possible response(s)
+CPWD=<fac>,<oldpwd>,<new pwd>	*Only SC and P2 facility supported* *+CME ERROR: <err>*
+CPWD=?	*+CPWD: ("SC", 8), ("P2", 8)*

2.4 Mobile Termination Control & Status Commands

2.4.1 AT+CCLK Clock

Command	Possible response(s)
+CCLK=<time>	*<time>: string type value; format is "yy/MM/dd,hh:mm:ss±zz", where characters indicate year (two last digits), month, day, hour, minutes, seconds and time zone (indicates the difference, expressed in quarters of an hour, between the local time and GMT; range -96...+96). E.g. 6th of May 1994, 22:10:00 GMT+2 hours equals to "94/05/06,22:10:00+08"* *+CME ERROR: <err>*
+CCLK?	*+CCLK: <time>* *+CME ERROR: <err>*
Example:	*+CCLK: 16/01/28,15:54:40-24*

2.4.2 AT+CESQ Extended signal quality

Command	Possible response(s)
+CESQ	*+CESQ: 99,99,255,255,22,55*
+CESQ=?	*+CESQ: (99),(99),(255),(255),(0..34,255),(0..97,255)*

2.4.3 AT+CFUN Set phone functionality

Command	Possible response(s)
+CFUN=[<fun>[,<rst>]]	*<fun>: integer type* *0 minimum functionality* *1 full functionality* *4 disable phone both transmit and receive RF circuits, access to SIM card is allowed* *<rst>: integer type* *0 do not reset after change* *1 reset after change*
+CFUN?	*+CFUN: 1*
+CFUN=?	*+CFUN: (0,1,4), (0,1)*

2.4.4 AT+CGPIAF Printing IP address format

Command	Possible response(s)
+CGPIAF=	*TBD*
+CGPIAF?	*+CGPIAF: 0,0,0,0*
+CGPIAF=?	*+CGPIAF: (0,1),(0,1),(0,1),(0,1)*

2.4.5 AT+CLAC List all available AT commands

Command	Possible response(s)
+CLAC	*+CME ERROR: <err>*
	<AT Command1> *[<CR><LF><AT Command2>* *[...]]*
+CLAC=?	*OK*

2.4.6 AT+CPAS List all available AT commands

Command	Possible response(s)
+CPAS	*+CPAS: <pas>* *<pas>: integer type* *0 ready (MT allows commands from TA/TE)* *4 call in progress (MT is ready for commands from TA/TE, but a call is in progress)* *5 asleep (MT is unable to process commands from TA/TE because it is in a low functionality state)*
	+CME ERROR: <err>
+CPAS=?	*+CPAS: (0,4,5)*

2.4.7 AT+CPIN Enter PIN

Command	Possible response(s)
+CPIN=<pin>[,<newpin>]	*+CME ERROR: <err>*
+CPIN?	*+CPIN: READY* *+CME ERROR: <err>*
+CPIN=?	*+CPIN: <pin>[, <new pin>]*

2.4.8 AT+CPINR Remaining PIN retries

Command	Possible response(s)
+CPINR	*+CPINR: SIM PIN,3,3* *+CPINR: SIM PUK,10,10* *+CPINR: SIM PIN2,3,3* *+CPINR: SIM PUK2,10,10*
+CPINR=?	*ok*

2.4.9 AT+CRSM Restricted SIM access

Command	Possible response(s)
+CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]	*+CRSM: <sw1>,<sw2>[,<response>]*
	+CME ERROR: <err>
+CRSM=?	*ok*

2.4.10 AT+CSIM Generic SIM access

Command	Possible response(s)
+CSIM=<length>,<command>	*+CSIM: <length>,<response>*
	+CME ERROR: <err>
+CSIM=?	*ok*

2.4.11 AT+CSQ Signal Quality (RSSI)

Command	Possible response(s)
+CSQ	*+CSQ: <rssi>,<ber>*
	+CME ERROR: <err>
+CSQ=?	*+CSQ: (0..31, 99), (99)*

2.4.12 AT+CTZR Automatic Time Zone Reporting

Command	Possible response(s)
+CTZR=<onoff>	*+CTZR:<onoff>* *<onoff>: integer type value indicating:*

	0 disable time zone change event reporting. *1 Enable time zone change event reporting.*
	+CME ERROR: <err>
+CTXR=?	*+CSQ: (0..31, 99), (99)*

2.4.13 AT+CMEC Mobile Termination control mode

Command	Possible response(s)
+CMEC=[<keyp>[,<disp>[,<ind>[,<tscrn>]]]	*+CMEC: <keyp>,<disp>,<ind>,<tscrn>*
	+CME ERROR: <err>
+CMEC=?	*+CMEC: (0),(0),(0),(0)*

2.4.14 AT+CMER Mobile Termination event reporting

Command	Possible response(s)
+CMER=[<mode>[,<keyp>[,<disp>[,<ind>[,<bfr>>[,<tscrn]]]]]	*+CME ERROR: <err>*
+CMER?	*+CMER: <mode>,<keyp>,<disp>,<ind>,<bfr>,<tscrn>*
+CMER=?	*+CMER: (3),(0),(0),(0-2),(0),(0)*

2.4.15 AT+CMEE Report Mobile termination errors

Command	Possible response(s)
+CMEE=<n>	*<n>: integer type.* *0 disable +CME ERROR: <err> result code and use ERROR instead* *1 enable +CME ERROR: <err> result code and use numeric <err> values (refer subclause 9.2)* *2 enable +CME ERROR: <err> result code and use verbose <err> values (refer subclause 9.2)*
+CMEE?	*+CMEE: <n>*
+CMEE=?	*+CMEE: (0,1,2)*

2.5 Packet Domain Commands

2.5.1 AT+CEMODE UE Modes of operation for EPS

Command	Possible response(s)
+CEMODE=[<mode>]	*<mode>: a numeric parameter which indicates the mode of operation* *0 PS mode 2 of operation* *1 CS/PS mode 1 of operation* *2 CS/PS mode 2 of operation* *OK* *ERROR*
+CEMODE?	*+CEMODE: <mode>*
+CEMODE=?	*+CEMODE: (0,2)*

2.5.2 AT+CEREG EPS network registration status

Command	Possible response(s)
+CEREG=[<n>]	<n>: a numeric parameter 0 disable network registration unsolicited result code 1 enable network registration unsolicited result code +CEREG: <stat> 2 enable network registration and location information unsolicited result code +CEREG: <stat>[,<lac>,<rac>,<ci>[,<AcT>]] *OK* *+CME ERROR: <err>*
+CEREG?	+CEREG: <n>,<stat>[,<tac>,<ci>[,<AcT>]]
+CEREG=?	+CEREG: (0,1,2)

2.5.3 AT+CGACT PDP context activate or deactivate

Command	Possible response(s)
+CGACT=[<state>[,<cid>[,<cid>[,...]]]]	*<state>: a numeric parameter that indicates the state of PDP context activation* *0 deactivated* *1 activated* *Other values are reserved and will result in an ERROR response to the execution command.*

	<cid>: a numeric parameter which specifies a particular PDP context definition (see the +CGDCONT and +CGDSCONT commands). *OK* *ERROR*
	+CME ERROR: <err>
+CGACT?	*+CGACT: 1,1* *+CGACT: 2,0* *+CGACT: 3,1* *+CGACT: 4,0* *+CGACT: 8,0*
+CGACT=?	*+CGACT: (0,1)*

2.5.4 AT+CGATT PS Attach or Detach

Command	Possible response(s)
+CGATT=[<state>]	*<state>: : a numeric parameter that indicates the state of PS attachment* *0 detached* *1 attached* *OK* *ERROR*
+CGATT?	*+CGATT: <state>*
+CGATT=?	*+CGATT: (0,1)*

2.5.5 AT+CGAUTH Define PDP context authentication parameters

Command	Possible response(s)
+CGAUTH= TBD	*TBD*
+CGAUTH?	*+CGAUTH: 1,0,"","""* *+CGAUTH: 2,0,"","""* *+CGAUTH: 3,0,"","""* *+CGAUTH: 4,0,"","""* *+CGAUTH: 8,0,"","""*
+CGAUTH=?	*+CGAUTH: (1..8),(0,1,2),(userId max length 30),(password max length 20)*

2.5.6 AT+CGCMOD PDP Context Modify

Command	Possible response(s)

+CGCMOD=[<cid>[,<cid>[,...]]]	*<cid>: a numeric parameter which specifies a particular PDP context definition (see the +CGDCONT and +CGDSCONT commands).* *OK* *ERROR*
+CGCMOD=?	*+CGCMOD: (1,3)*

2.5.7 AT+CGCONTRDP PDP Context Read Dynamic Parameters

Command	Possible response(s)
+CGCONTRDP=<cid>	*+CGCONTRDP: <cid>,<bearer_id>,<apn>[,<source address and subnet mask>[,<gw_addr>[,<DNS_prim_addr>[,<DNS_sec_addr>[,<P-CSCF_prim_addr>[,<P-CSCF_sec_addr>]]]]]]* *[<CR><LF>+CGCONTRDP: <cid>,<bearer_id>,<apn>[,<source address and subnet mask>* *[,<gw_addr>[,<DNS_prim_addr>[,<DNS_sec_addr>[,<P-CSCF_prim_addr>[,<P-CSCF_sec_addr>]]]]]]*
+CGCONTRDP=?	*+CGCONTRDP: (1,3)*

2.5.8 AT+CGDCONT Define PDP context

Command	Possible response(s)
+CGDCONT=[<cid>[,<PDP_type>[,<APN>[,<PDP_addr>[,<d_comp>[,<h_comp>[,<IPv4AddrAlloc>[,<emergency indication>]]]]]]]]	*OK* *ERROR*
+CGDCONT?	*+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[,<IPv4AddrAlloc>[,<emergency indication>]]* *[<CR><LF>+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[,<IPv4AddrAlloc>[,<emergency indication>]]* *[...]]*
+CGDCONT=?	*+CGDCONT: (1,2,3,4,5,6,7,8),"IP",,,,,(0,1),(0,1),(0,1,2),(0,1),(0,1)*

	+CGDCONT: (1,2,3,4,5,6,7,8),"IPV6",,,,,,(0,1),(0,1,2),(0,1),(0,1) *+CGDCONT: (1,2,3,4,5,6,7,8),"IPV4V6",,,,,(0,1),(0,1),(0,1,2),(0,1),(0,1)*

2.5.9 AT+CGDSCONT Define Secondary PDP context

Command	Possible response(s)
+CGDSCONT=[<cid>,<p_cid>[,<d_comp>[,<h_comp>]]]	*OK* *ERROR*
+CGDSCONT?	*OK*
+CGDSCONT=?	*+CGDSCONT: (5,6,7),(1,2,3,4,8),(0),(0),(0)*

2.5.10 AT+CGEQOS Define EPS Quality Of Service

Command	Possible response(s)
+CGEQOS=[<cid>[,<QCI>[,<DL_GBR>,<UL_GBR>[,<DL_MBR>,<UL_MBR]]]]	*+CME ERROR: <err>*
+CGEQOS?	*+CGEQOS: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]* *[<CR>>LF>+CGEQOS: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]* *[...]]*
+CGEQOS=?	*+CGEQOS: (1..8), (0..9), (0..256000), (0..256000), (0..256000), (0..256000)*

2.5.11 AT+CGEQOSRDP EPS Quality Of Service Read Dynamic Parameters

Command	Possible response(s)
+CGEQOSRDP=[<cid>]	*+CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]* *[<CR>>LF>+CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]* *[...]]*

+CGEQOSRDP=?	*+CGEQOSRDP: (list of <cid>s associated with active contexts)*

2.5.12 AT+CGEREP Define EPS Quality Of Service

Command	Possible response(s)
+CGEREP=[<mode>[,<bfr>]]	*<mode>: a numeric parameter* *0 buffer unsolicited result codes in the MT; if MT result code buffer is full, the oldest ones can be discarded. No codes are forwarded to the TE.* *1 discard unsolicited result codes when MT-TE link is reserved (e.g. in on-line data mode); otherwise forward them directly to the TE* *2 buffer unsolicited result codes in the MT when MT-TE link is reserved (e.g. in on-line data mode) and flush them to the TE when MT-TE link becomes available; otherwise forward them directly to the TE* *<bfr>: a numeric parameter* *0 MT buffer of unsolicited result codes defined within this command is cleared when <mode> 1 or 2 is entered* *1 MT buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1 or 2 is entered (OK response shall be given before flushing the codes)* *OK* *ERROR*
+CGEREP?	*+CGEREP: <mode>,<bfr>*
+CGEREP=?	*+CGEREP: (0,1,2),(0,1)*

2.5.13 AT+CGPADDR Show PDP Address(es)

Command	Possible response(s)
+CGPADDR=[<cid>[,<cid>[,...]]]	*+CGPADDR: <cid>[,<PDP_addr_1>[,<PDP_addr_2>]]*
+CGPADDR=?	*+CGPADDR: (list of defined <cid>s)*
Example: AT+CGPADDR=3	*+CGPADDR: 3,"100.69.244.115","254.128.0.0.0.0.0.0.0.0.0.73.249.245.27.1"*

2.5.14 AT+CGSCONTRDPSecondary PDP Context Read Dynamic Parameters

Command	Possible response(s)
+CGSCONTRDP=[<cid>]	*+CGSCONTRDP: <cid>,<p_cid>,<bearer_id>* *[<CR><LF>+CGSCONTRDP:* *<cid>,<p_cid>,<bearer_id>* *[...]]*
+CGSCONTRDP=?	*+CGSCONTRDP: (list of <cid>s associated with active contexts)*

2.5.15 AT+CGSMS Select service for mobile originated SMS messages

Command	Possible response(s)
+CGSMS=[<service>]	*<service>: a numeric parameter which indicates the service or service preference to be used* *0 Packet Domain* *2 Packet Domain preferred (use circuit switched if GPRS not available)* *OK* *ERROR*
+CGSMS?	*+CGSMS: <service>*
+CGSMS=?	*+CGSMS: (0,1,2,3)*

2.5.16 AT+CGTFTRDP Traffic Flow Template Read Dynamic Parameters

Command	Possible response(s)
+CGTFTRDP=[<cid>]	*+CGTFTRDP: <cid>,<packet filter identifier>,<evaluation precedence index>,<source address and subnet mask>,<protocol number (ipv4) / next header (ipv6)>,<destination port range>,<source port range>,<ipsec security parameter index (spi)>,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>,<flow label (ipv6)>,<direction>,<NW packet filter Identifier>* *[<CR><LF>+CGTFTRDP: <cid>,<packet filter identifier>,<evaluation precedence*

	index>,<source address and subnet mask>,<protocol number (ipv4) / next header (ipv6)>,<destination port range>, <source port range>,<ipsec security parameter index (spi)>,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>,<flow label (ipv6)>,<direction>,<NW packet filter Identifier> [...]]
+CGTFTRDP=?	*+CGTFTRDP: (list of <cid>s associated with active contexts)*

2.5.17 AT+CGTFT Traffic Flow Template

Command	Possible response(s)
+CGTFT=[<cid>,[<packet filter identifier>,<evaluation precedence index>[,<source address and subnet mask>[,<protocol number (ipv4) / next header (ipv6)>[,<destination port range>[,<source port range>[,<ipsec security parameter index (spi)>[,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>[,<flow label (ipv6)>[,<direction>]]]]]]]]]	*OK* *ERROR*
+CGTFT?	*+CGTFT: <cid>,<packet filter identifier>,<evaluation precedence index>,<source address and subnet mask>,<protocol number (ipv4) / next header (ipv6)>,<destination port range>,<source port range>,<ipsec security parameter index (spi)>,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>,<flow label (ipv6)>,<direction>* *[<CR><LF>+CGTFT: <cid>,<packet filter identifier>,<evaluation precedence index>,<source address and subnet mask>,<protocol number (ipv4) / next header (ipv6)>,<destination port range>, <source port range>,<ipsec security parameter index (spi)>,<type of service*

	(tos) (ipv4) and mask / traffic class (ipv6) and mask>,<flow label (ipv6)>,<direction> [...]]
+CGTFT=?	*+CGTFT: <PDP_type>,(list of supported <packet filter identifier>s),(list of supported <evaluation precedence index>s),(list of supported <source address and subnet mask>s),(list of supported <protocol number (ipv4) / next header (ipv6)>s),(list of supported <destination port range>s),(list of supported <source port range>s),(list of supported <ipsec security parameter index (spi)>s),(list of supported <type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>s),(list of supported <flow label (ipv6)>s),(list of supported <direction>s) [<CR><LF>+CGTFT: <PDP_type>,(list of supported <packet filter identifier>s),(list of supported <evaluation precedence index>s),(list of supported <source address and subnet mask>s),(list of supported <protocol number (ipv4) / next header (ipv6)>s),(list of supported <destination port range>s),(list of supported <source port range>s),(list of supported <ipsec security parameter index (spi)>s),(list of supported <type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>s),(list of supported <flow label (ipv6)>s),(list of supported <direction>s) [...]]*

2.6 Text Mode Commands

2.6.1 AT+CMGD Delete SMS Message

Command	Possible response(s)
+CMGD=<index>[,<delflag>]	*<delflag>: integer type, indicating multiple message deletion request as follows:* *0 Delete the message specified in <index>* *4 Delete all messages from preferred message storage including unread messages.*

	OK *ERROR* *+CMS ERROR: <err>*
+CMGD=?	*+CMGD: (0-9),(0-4)*

2.6.2 AT+CMGF SMS Message Format

Command	Possible response(s)
+CMGF=[<mode>]	*<mode>: integer type* *0 PDU mode* *1 text mode* *OK* *ERROR*
+CMGF?	*+CMGF: <mode>*
+CMGF=?	*+CMGF: (0,1)*

2.6.3 AT+CMGL List SMS Messages

Command	Possible response(s)
+CMGL[=<stat>]	***if text mode (+CMGF=1), command successful and SMS-SUBMITs and/or SMS-DELIVERs:*** *+CMGL:* *<index>,<stat>,<oa/da>,[<alpha>],[<scts>][,<tooa/toda>,* *<length>]<CR><LF><data>[<CR><LF>* *+CMGL:* *<index>,<stat>,<da/oa>,[<alpha>],[<scts>][,<tooa/toda>,* *<length>]<CR><LF><data>[...]]* ***if text mode (+CMGF=1), command successful and SMS-STATUS-REPORTs:*** *+CMGL:* *<index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>* *[<CR><LF>* *+CMGL:* *<index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>* *[...]]* ***if text mode (+CMGF=1), command successful and SMS-COMMANDs:*** *+CMGL: <index>,<stat>,<fo>,<ct>[<CR><LF>* *+CMGL: <index>,<stat>,<fo>,<ct>[...]]* ***if text mode (+CMGF=1), command successful and CBM storage:***

	+CMGL: *<index>,<stat>,<sn>,<mid>,<page>,<pages>* *<CR><LF><data>[<CR><LF>* *+CMGL:* *<index>,<stat>,<sn>,<mid>,<page>,<pages>* *<CR><LF><data>[...]]* ***otherwise:*** *+CMS ERROR: <err>*
+CMGL=?	*+CMGL: (list of supported <stat>s)*

2.6.4 AT+CMGR Read SMS Message

Command	Possible response(s)
+CMGR=<index>	***if text mode (+CMGF=1), command successful and SMS-DELIVER:*** *+CMGR:* *<stat>,<oa>,[<alpha>],<scts>[,<tooa>,<fo>,<pid>,<dcs>,* *<sca>,<tosca>,<length>]<CR><LF><data>* ***if text mode (+CMGF=1), command successful and SMS-SUBMIT:*** *+CMGR:* *<stat>,<da>,[<alpha>][,<toda>,<fo>,<pid>,<dcs>,[<vp>],* *<sca>,<tosca>,<length>]<CR><LF><data>* ***if text mode (+CMGF=1), command successful and SMS-STATUS-REPORT:*** *+CMGR:* *<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,* *<dt>,<st>* ***if text mode (+CMGF=1), command successful and SMS-COMMAND:*** *+CMGR:* *<stat>,<fo>,<ct>[,<pid>,[<mn>],[<da>],[<toda>],<length>* *<CR><LF><cdata>]* ***if text mode (+CMGF=1), command successful and CBM storage:*** *+CMGR:* *<stat>,<sn>,<mid>,<dcs>,<page>,<pages><CR><LF><data>* ***otherwise:*** *+CMS ERROR: <err>*
+CMGR=?	*OK*

2.6.5 AT+CMGS Send SMS Message

Command	Possible response(s)

if text mode (+CMGF=1): +CMGS=<da>[,<toda>]<CR> text is entered<ctrl-Z/ESC>	***if text mode (+CMGF=1) and sending successful:*** *+CMGS: <mr>[,<scts>]* ***if sending fails:*** *+CMS ERROR: <err>*
+CMGS=?	*+CMGS: (0-4294967295)*

2.6.6 AT+CMGW Write SMS Message to Memory

Command	Possible response(s)
if text mode (+CMGF=1): +CMGW[=<oa/da>[,<tooa/toda>[,<stat>]]]<CR> text is entered<ctrl-Z/ESC>	*+CMGW: <index>* *+CMS ERROR: <err>*
+CMGW=?	*+CMGW: (0-4294967295),(0-4294967295)*

2.6.7 AT+CMSS Send SMS Message from Storage

Command	Possible response(s)
+CMSS=<index>[,<da>[,<toda>]]	***if text mode (+CMGF=1) and sending successful:*** *+CMSS: <mr>[,<scts>]* ***if sending fails:*** *+CMS ERROR: <err>*
+CMSS=?	*OK*

2.6.8 +CMTI Notifies Arrival of New SMS

Unsolicited Response	Possible response(s)
+CMTI	*Notifies Arrival of New SMS*

2.6.9 AT+CNMA New Message Acknowledgement to ME/TA

Command	Possible response(s)
if text mode (+CMGF=1): +CNMA	*+CMS ERROR: <err>*
+CNMA?	*OK*
+CNMA=?	*+CNMA: (0,1,2)*

2.6.10 AT+CNMI New Message Indications to TE

Command	Possible response(s)
if text mode (+CMGF=1): +CNMA	*+CMS ERROR: <err>*
+CNMA?	*OK*
+CNMA=?	*+CNMA: (0,1,2)*

2.7 CMCC Specific Commands

2.7.1 AT^MODE=1 System Reporting Mode

Command	Possible response(s)
^MODE=1	*^MODE: <sys_mode>,<sys_submode>* *<sys_mode>: a numeric parameter which indicates the system mode* *0 No Service* *1 Reserved* *2 Reserved* *4 Reserved* *3 GSM/GPRS mode* *5 WCDMA mode* *15 TD-SCDMA mode* *16 FDD-LTE mode* *17 TDD-LTE mode* *<sys_submode): a numeric parameter which indicates the system sub mode* *0 No Service* *1 GSM mode* *2 GPRS mode* *3 EDGE mode* *4 WCDMA mode* *5 HSDPA mode* *6 HSUPA mode* *7 HSUPA and HSDPA mode* *8 TD-SCDMA mode* *9 HSPA+ mode* *25 TDD-LTE mode* *26 FDD-LTE mode* *Example:* *AT^MODE=1* *^MODE: 16,26* *OK*

2.7.2 AT^DIALMODE PS Access Mode

Command	Possible response(s)
^DIALMODE?	*Returns the PS access mode recommended by the device:* *Example:* *AT^DIALMODE?* *^DIALMODE: 1*

	OK
^DIALMODE=?	*REturns acceptable parameters.* *Example:* *AT^DIALMODE=?* *^DIALMODE: (1)* *OK*

2.7.3 AT^SYSINFO Request System Information

Command	Possible response(s)
^SYSINFO	*Returns system information.* *^SYSINFO:* *<service_status>,<service_domain><roaming>,<system_mode>,<sim_state>,<reserved>,<system_submode> OK* *<service_status>:* *0 No Service* *1 Limited Service* *2 Normal Service* *3 Limited Regional Service* *<service_domain>:* *0 No Service* *1 Only CS service* *2 Only PS service* *3 CS and PS service* *<roaming>:* *0 No Roaming* *1 Roaming* *<system_mode>:* *0 No Service* *1 Reserved* *2 Reserved* *3 GSM/GPRS mode* *4 Reserved* *5 WCDMA mode* *15 TD-SCDMA mode* *16 FDD-LTE mode* *17 TDD-LTE mode* *<sim_state>:* *0 USIM is invalid* *1 USIM is present and valid* *255 USIM absent or present without PIN Verification* *<reserved>:* *0 Reserved* *<system_submode>:* *0 No Service* *1 GSM mode* *2 GPRS mode*

	3 EDGE mode *4 WCDMA mode* *5 HSDPA mode* *6 HSUPA mode* *7 HSUPA and HSDPA mode* *8 TD_SCDMA mode* *9 HSPA+ mode* *25 TDD-LTE mode* *26 FDD-LTE mode* *Example:* *AT^SYSINFO* *^SYSINFO: 1,2,1,16,255,0,26* *OK*

2.7.4 AT^HVER Hardware Version

Command	Possible response(s)
^HVER	*Returns hardware version.* *Example:* *AT^HVER* *^HVER: version1.0* *OK*

2.7.5 AT^RFSW RF State Switch State

Command	Possible response(s)
^RFSW	*Returns state of RF switch.* *0 RF switch is off* *1 RF switch is on* *Example:* *^RFSW: 1* *OK*

2.7.6 AT^CSG CSG Selection

Command	Possible response(s)

^CSG=n	*Manual CSG selection.returns maximum access capacity of the Mobile Terminal.* *Example:* *TBD*
^CSG=?	*Example:* *TBD*

2.7.7 AT^AUTOATT Automatic Attach

Command	Possible response(s)
^AUTOATT=<auto_att>	*Command enables or disables automatic EPS attach after AT+CFUN=1. If automatic EPS attach is disable when device is started, then AT+CGATT must be issued to initiate EPS attach.* *<auto_att>:* *0 Disable automatic EPS attach* *1 Enable automatic EPS attach* *Example:* *AT^AUTOATT=1* *OK*
^AUTOATT?	*Returns current state:* *Example:* *AT^AUTOATT?* *^AUTOATT:1* *OK*
^AUTOATT=?	*Returns acceptable inputs* *Example:* *AT^AUTOATT=?* *^AUTOATT: (0,1)* *OK*

2.7.8 AT+BGLTEPLMN Background Search for LTE PLMNs

Command	Possible response(s)
+BGLTEPLMN=?	*Similar to AT+COPS=? command. AT+BGLTEPLMN=? can be used even when a SIM card is not present.* *+BGLTEPLMN: [list of supported (stat, long_oper, short_oper, num_oper [,act,rsvp)] [, list of supported (format)]*

	Example: *AT+BGLTEPLMN=?* *ERROR* *Not supported in firmware.*
	+CME ERROR:err

2.7.9 AT^RESET Hard Reset

Command	Possible response(s)
^RESET	*Performs hardware reset.* *Example:* *AT^RESET* *OK*

2.7.10 AT+VZWAPNE Edit APN Table

Command	Possible response(s)
+VZWAPNE=<wapn>,<apncl>, <apnni>,<apntype>,<apnb>, apned>,<apntime>	*Sets APN entries.* *<wapn>:* *0 Take no action* *1 Edit APN entry number 1.* *2 Edit APN entry number 2.* *n Edit APN entry number n.* *<apncl>:* *1 APN class number 1.* *2 APN class number 2.* *n APN class number n.* *<apnni>:* *VZWIMS Verizon Wireless IMS PDN* *VZWADMIN Verizon Wireless Administrative PDN* *VZWINTERNET Verizon Wireless Internet PDN* *VZWAPP Verizon Wireless Application PDN* *<apntype>:* *IP IPV4* *IPv6 IPv6 type* *IPv4v6 IPv4 and IPv6 type* *<apnb>:* *LTE LTE bearer used* *<apned>:* *Enabled The APN is enabled.* *Disabled The APN is disabled.* *<apntime>:* *0 Indicates no-ending timer.* *x Timer value in minutes.* *Example:*

	AT+VZWAPNE=1,1,"VZWINTERNET","IP","LTE", "Enabled",0 *OK*
+VZWAPNE=?	*Returns possible APN ranges.* *+VZWAPNE:<apncl>1, <apntype>1, <apnnb>1, <apned>1, <apntime>1,* *<apncl>2, <apntype>2, <apnnb>2, <apned>2, <apntime>2, ..., <apncl>n, <apntype>n, <apnnb>n, <apned>n, <apntime>n +CME ERROR:err* *Example:* *AT+VZWAPNE=?* *+VZWAPNE:* *(0,1,2,3,4),(1,2,3,4),,("IP","IPv6","IPv 4v6"),("LTE"),("Enabled","Disabled"),,* *OK*
+VZWAPNE?	*Returns current APN settings* *Example:* *AT+VZWAPNE?* *+VZWAPNE:* *1,VZWIMS,IPv4v6,LTE,Enabled,0* *2,VZWADMIN,IPv4v6,LTE,Enabled,0* *3,VZWINTERNET,IPv4v6,LTE,Enabled,0* *4,VZWAPP,IPv4v6,LTE,Enabled,0* *0,VZWEMERGENCY,IPv4v6,LTE,Enabled,0*

2.7.11 AT+VZWRSRP Read RSRP Values

Command	Possible response(s)
+VZWRSRP?	*Returns all cell tower RSRP values the device is measuring up to 8 cells.* *+VZWRSRP:<cellID>1, <EARFCN>1, <RSRP>1, <cellID>2, <EARFCN>2, <RSRP>2, ..., <cellID>n, <EARFCN>n, <RSRP>n* *Example:* *AT+VZWRSRP?* *+VZWRSRP: 425,2050,-100.10* *OK*
	+CME ERROR:err

2.7.12 AT+VZWRSRQ Read RSRQ Values

Command	Possible response(s)
+VZWRSRQ?	*Returns all cell tower RSRQ values the device is measuring up to 8 cells.*

	+VZWRSRQ:<cellID>1, <EARFCN>1, <RSRQ>1, <cellID>2, <EARFCN>2, <RSRQ>2, ..., <cellID>n, <EARFCN>n, <RSRQ>n *Example:* *AT+VZWRSRQ?* *+VZWRSRQ: 425,2050,-11.30* *OK*
	+CME ERROR:err

2.7.13 AT+CPIN2 Enter PIN2

Command	Possible response(s)
+CPIN2=<pin>[,<newPin>[<aid>]]	*Sends the USIM the PIN2 and PUK2 if the SIM supports PUK2.* *Example:* *AT+CPIN2=1234* *OK*
	+CME ERROR:err
+CPIN?	*Example:* *AT+CPIN?* *+CPIN: READY* *OK*
+CPIN=?	*AT+CPIN=?* *+CPIN: <pin>[, <new pin>]* *OK*

2.7.14 AT+SCPWD Change PIN with <aid>

Command	Possible response(s)
+SCPWD=<pin>,<newPin>[,<aid>]<fac>,<pin>,<newPin>[,<aid>]	*Allows changing PIN with <aid>.* *<fac>: Only "SC" is supported* *<pin>: Value of PIN* *<newpin>: value of new PIN code* *<aid>: Application ID if omitted USIM application is used* *Example:* *AT+SCPWD="SC", 1234,5678,A0000000871004010203040506070809*

	OK
	+CME ERROR:err
AT+SCPWD=?	*+CPWD: ("SC", 8, "aid"), ("P2", 8, "aid")* *OK*

2.7.15 AT+IMSCLI Send CLI to IMS Client

Command	Possible response(s)
+IMSCLI=”cli command”	*Command sends CLI comamdn to IMS client through an AT command.* *Example:* *AT+IMSCLI="sendMessage 1234 SMS_content save=1"* *+IMSCLI: Message status: 0, ID: 1* *OK*
	+CME ERROR:err

2.7.16 AT+IMSSMSSEND Send SMS over IMS

Command	Possible response(s)
+IMSSMSSEND=<to>,<text>[,<save>[,<priority>[,<cbk number>]]]	*Sends an SMS with the IMS client through an AT command.* *<to>: String, destination phone number* *<text>: String, body of the message* *<save>: Integer,* *0 do not save SMS message, default* *1 save the SMS message* *<priority>: Integer* *<cbk number>: String, callback number* *Example:* *AT+IMSSMSSEND="123456789","This is the SMS content"* *OK*
	+CME ERROR:err

2.7.17 AT+IMSSTATEGET Get IMS state information

Command	Possible response(s)

+IMSSTATEGET?	*Returns internal state of IMS.* *Example:* *AT+IMSSTATEGET?* *OK*
+IMSSTATEGET=?	*Example:* *AT+IMSSTATEGET=?* *+IMSSTATEGET: ("SIMSTORE","SIP")* *OK*
+IMSSTATEGET=<state>,<value>	*<state>: String. Name of state to query* *<value>: String. Specific to current state* *Examples:* *"SIMSTORE", "INIT" Initialization* *"SIMSTORE", "INITIALIZED" Initialized* *"SIMSTORE", "WAIT_SIM" SIM is not ready* *"SIMSTORE", "WAIT_STORE" Storage on SIM is not ready* *"SIMSTORE", "READY" Storage is ready* *"SIP", "REGISTERING" Registration in progress* *"SIP", "REGISTERED" Registered on IMS server* *"SIP", "DEREGISTERING" De-registration in progress* *"SIP", "WAIT_DEREGISTER ED" De-registration in progress* *"SIP", "DEREGISTERED" De-registered from IMS server* *Example:* *AT+IMSSTATEGET="SIP"* *+IMSSTATEGET: SIP,DEREGISTERED* *OK*

2.7.18 AT+IMSCFGSMS IMS SMS configuration

Command	Possible response(s)
+IMSCFGSMS=[<mo-type>,[,<mo-char-set>[,<mo-ack-req]]]	*Command configures teh SMS parameters over IMS.* *<mo-type>:String. MO message format.* *"3GPP" GSM format* *"3GPP2" CDMA format* *"TEXT" SIP text messaging* *<mo-char-set>: String. Force text encoding of MO SMS.* *"AUTO" Encoding is auto-selected (depends on used symbols)*

	"7BIT_ASCII" *"7BIT_GSM"* *"7BIT_IA5"* *"8BIT_LATIN"* *"16BIT_UCS2"* *<mo-ack-req>: Integer. Send TP-SRR or BearerReplyOption* *0 do not request* *1 do request* *Example:* *AT+VZWRSRP?* *+VZWRSRP: 425,2050,-100.10* *OK* *Examples:* *AT+IMSCFGSMS="3GPP"* *+IMSCFGSMS: "3GPP","AUTO",0* *OK* *AT+IMSCFGSMS=,"7BIT_IA5",1* *+IMSCFGSMS: "3GPP","7BIT_IA5",1* *OK* *AT+IMSCFGSMS?* *+IMSCFGSMS: "3GPP","7BIT_IA5",1 OK*
+IMSCFGSMS?	*Example:* *AT+IMSCFGSMS?* *+IMSCFGSMS: "3GPP2","AUTO",0* *OK*
+IMSCFGSMS=?	*Example:* *AT+IMSCFGSMS=?* *+IMSCFGSMS:* *("3GPP","3GPP2"),("AUTO","ASCII","GSM","IA5","LATIN1","UCS-2"),(0,1)* *OK*

2.7.19 AT+TISTESTMODE TIS test mode

Command	Possible response(s)
+TISTESTMODE=<enable>,<server>,<port>	*Command enables or disables the TIS test mode of the device.* *<enable>: Integer* *0 disabled* *1 enabled* *<server>: Integer* *0 client mode*

	1 server mode *<port>: Integer. UDP port which teh client can connect, shall be in range of [8883-8889]* *Example:* *AT+TISTESTMODE=1,1,8883* *+TISTESTMODE: TIS has been enabled on [1,1,8883].* *OK* *AT+TISTESTMODE=0* *+TISTESTMODE: TIS has been disabled* *OK*
+TISTESTMODE?	*Example:* *AT+TISTESTMODE?* *+TISTESTMODE: 0* *OK*
AT+TISTESTMODE=?	*Example:* *AT+TISTESTMODE=?* *+TISTESTMODE: (0,1), (0,1), (8883, 8889)* *OK*

2.7.20 AT+SQNAUTOCONNECT Auto-Connect

Command	Possible response(s)
+SQNAUTOCONNECT=<autoconnect>	*Command changes autoconnect mode of the device. When enabled, the device will automatically set the device to maximum functionality(equivalent to AT+CFUN=1) after each reboot. This setting persists across reboots.* *<autoconnect>: Integer.* *0 Autoconnect disabled* *1 Autoconnect enabled* *Example:* *AT+SQNAUTOCONNECT=1* *OK*
AT+SQNAUTOCONNECT?	*AT+SQNAUTOCONNECT?* *+SQNAUTOCONNECT: 1* *OK*
AT+SQNAUTOCONNECT=?	*AT+SQNAUTOCONNECT=?*

	+SQNAUTOCONNECT: (0,1),(0,1) *OK*

2.7.21 AT+SQNSMSSIPLASTCODE Get Last SMS SIP Status Code

Command	Possible response(s)
+SQNSMSSIPLASTCODE?	*Returns last SIP status code defined in RFC 3261 (SIP: Session Initiation Protocol)* *Value Description* *1xx Provisional Responses* *2xx Successful Responses* *3xx Redirection Responses* *4xx Client Failure Responses* *5xx Server Failure Responses* *6xx General failure Responses* *Example:* *AT+SQNSMSSIPLASTCODE?* *+SQNSMSSIPLASTCODE: 200* *OK*

2.7.22 AT+SQNSTKAPNE Change STK APN Configuration

Command	Possible response(s)
+SQNSTKAPNE= <apnname>, <iptype>, <enabled>	*If the UICC STK application does not provide the APN in the open channel, this configuration will be used instead. This setting is not persistent across reboot.* *<apnname>: String, the APN* *<iptype>: String, IP type description* *ip IPv4 only* *ipv6 IPv6 only* *ipv4v6 IPv4 and IPv6* *<enabled>: Integer,* *0 APN is disabled* *1 APN is enabled* *Example:* *AT+SQNSTKAPNE="stktest","ipv4v6",1* *OK*
+SQNSTKAPNE?	*Example:* *AT+SQNSTKAPNE?* *+SQNSTKAPNE: 0,0,0*

	OK
+SQNSTKAPNE=?	*Example:* *AT+SQNSTKAPNE=?* *+SQNSTKAPNE: apnname,(ip, ipv6, ipv4v6), (0,1)* *OK*

2.7.23 AT+SQNDPRMODE Set Dynamic Power Reduction

Command	Possible response(s)
+SQNDPRMODE=<mode>	*Sets the Dynamic Power Reduction for Body SAR algorithm.* *<mode>: Integer.* *0 DPR for Body SAR is driven by a hardware GPIO (defined in PSI configuration).* *1 DPR for Body SAR is enabled* *2 DPR for Body SAR is disabled* *Example:* *AT+SQNDPRMODE=1* *OK*
+SQNDPRMODE=?	*Example:* *AT+SQNDPRMODE=?* *+SQNDPRMODE: (0-2)* *OK*
+SQNDPRMODE?	*Example:* *AT+SQNDPRMODE?* *+SQNDPRMODE: 1* *OK*

2.7.24 AT+SQNUSBENUM USB Enumeration Configuration

Command	Possible response(s)
+SQNUSBENUM=<mode>	*Sends an SMS with the IMS client through an AT command.* *<mode>: String, see options below.* *"auto" The device is enumerated in mass storage if the Host OS is not supporting neither ECM nor RNDIS (in this order) to install drivers. If it supports one of them, the device is enumerated in the first supported mode.* *"ecm" The device is enumerated as an ECM network device.*

	"ecm+acm" The device is enumerated as an ECM network device with an ACM port. *"ecm+acm+ecm" The device is enumerated as an ECM network device with an ACM port and a second network interface for debug.* *"ecm+ecm" The device is enumerated as an ECM network device with a second network interface for debug.* *"mbim_win8_only_ums_win_only->ecm" On windows 8.x Os, the device is enumerated in MBIM. Else the device is enumerated first as a mass storage in order to install drivers (on windows OS only) then, after ejection, the device is enumerated in ecm.* *"mbim_win8_only_ums_win_only->ecm+acm " On windows 8.x Os, the device is enumerated in MBIM. Else the device is enumerated first as a mass storage in order to install drivers (on windows OS only) then, after ejection, the device is enumerated in ecm+acm.* *"mbim_win8_only_ums_win_only->ecm+acm +ecm" On windows 8.x Os, the device is enumerated in MBIM. Else enumeration will be ums first (on windows OS only) then, after ejection, the device is enumerated in ecm+acm+ecm.* *"mbim" If MBIM is supported, the device is enumerated in MBIM.* *"mbim+ecm" If MBIM is supported, the device is enumerated in MBIM. Else device is enumerated in ecm* *"mbim+ecm+acm" If MBIM is supported, the device is enumerated in MBIM. Else device is enumerated in ecm+acm* *"rndis" The device is enumerated directly as an RNDIS network device.* *"ums_win_only->ecm" On windows Os, the device is enumerated first as a mass storage in order to install drivers. Then, when the CDROM is ejected, the device is re-enumerated in ecm.* *"ums_win_only->ecm+acm" On windows Os, the device is enumerated first as a mass*

	storage in order to install drivers. Then, when the CDROM is ejected, the device is re-enumerated in ecm+acm. *"ums_win_only->ecm+acm+ecm" On windows Os, the device is enumerated first as a mass storage in order to install drivers. Then, when the CDROM is ejected, the device is re-enumerated in ecm+acm+ecm.* *Example:* *AT+SQNUSBENUM="ecm+acm+ecm"* *OK*
+SQNUSBENUM=?	*Example:* *("none","ecm+acm+ecm","ecm+acm","ecm+ecm","ecm","mbim","mbim+ecm","mbim+ecm+acm","ums_win_only->ecm","ums_win_only->ecm+acm","ums_win_only->ecm+acm+ecm","mbim_win8_only_ums_win_only->ecm","mbim_win8_only_ums_win_only->ecm+acm","mbim_win8_only_ums_win_only->ecm+acm+ecm","rndis","rndis_win_only","auto","ums->ecm","ums->ecm+acm","ums->ecm+acm+ecm")[,1]* *OK*
+SQNUSBENUM?	*Example:* *AT+SQNUSBENUM?* *ecm+acm+ecm* *OK*

2.7.25 AT+IMSSMSSEND Send SMS over IMS

Command	Possible response(s)
+SQNDIAL=<"telNum">	*Initiates outbound call to "telNum".* *<telNum>: String, destination phone number* *Response:* *+SQNDIAL:<result>* *<result>: String, result of command.* *NO ANSWER* *NO CARRIER* *BUSY* *FAIL* *CONNECT* *Example:* *AT+SQNDIAL="+1272456"* *+SQNDIAL: CONNECT*

2.7.26 AT+SQNLASTCALLFAILCAUSE Get Last Call Failure Cause

Command	Possible response(s)
+SQNLASTCALLFAILCAUSE	*Retrieves the last voice call failure error cause.* *Response:* *+SQNLASTCALLFAILCAUSE:<errCause>* *<errCause>: Integer. Refer to TS 24.008 Annex. H: 3GPP specific cause values for call control*

2.8 Socket Data Commands

2.8.1 AT+SQNSD Socket Dial

Command	Possible response(s)
+SQNSD=<connId>,<txProt>,<rPort>,<IPaddr>[,<closureType> [,<lPort> [,<connMode>]]]	*Command opens remote connection via socket.* ***Note:*** *if we set <connMode> to online mode connection and the command is successful, we enter the online data mode and we see the intermediate result code CONNECT. After the CONNECT, we can suspend the direct interface to the socket connection (note that the socket stays open) using the escape sequence (+): the module moves back to command mode and we receive the final result code OK after the suspension. After such a suspension, it is possible to resume it at every moment (unless the socket inactivity timer timeouts, see +SQNSCFG) by using the +SQNSO command with the corresponding <connId>.* ***Note:*** *if we set <connMode> to command mode connection and the command is successful, the socket is opened and we remain in command mode and we see the result code OK.* ***Note:*** *if some input data arrive through a connected socket and are not yet read because the module entered command mode be- fore reading them (after an escape*

	sequence or after +SQNSD has been issued with <connMode> set to command mode connection), these data are buffered and we *receive the SRING URC (SRING presentation format depends on the last +SQN- SCFGEXT setting). It is possible to read these data afterwards issuing +SQNSRECV. Under the same hypotheses, it is possible to send data while in command mode issuing +SQNSSEND.* *<connId>: Integer. Socket connection identifier, range of 1-6* *<txProt>: Integer. Transmission protocol.* *0 TCP* *1 UDP* *<rPort>: Integer. Remote host port number to contact. Range of 0-65535* *<IPaddr>: String. Address of the remote host. Any valid IP address in the format "xxx.xxx.xxx.xxx" or any host name to be sovled with a DNS query.* *<closureType>: Integer. Socket closure behavior for TCP. Has no effect for UDP connections.* *0 Local host closes immediately when remote host has closed(default)* *255 Local host closes after an escape sequence(+)* *<lPort>: Integer. UDP connection local port has no effect on TCP connections. Range of 0-65535.* *<connMode>: Integer. Connection mode.* *0 Online mode (default)* *1 Command mode connection* *Example:* *AT+SQNSD=3,0,80,"thingspace.io"* *CONNECT*
+SQNSD=?	*Example:* *AT+SQNSD=?* *+SQNSD:* *(1-6),(0-1),(0-65535),,(0,255),(0-65535),(0-1)* *OK*

2.8.2 AT+SQNSO Socket Restore

Command	Possible response(s)

+SQNSO=<connId>	*Resumes connection which has been suspended by the escape sequence.* *<connId>: Integer. 1-6* *Example:* *AT+SQNSO=1*
+SQNSO=?	*AT+SQNSO=?* *+SQNSO: (1-6)* *OK*

2.8.3 AT+SQNSCFG Socket Configuration

Command	Possible response(s)
+SQNSCFG=<connId>, <cid>, <pktSz>, <maxTo>, <connTo>, <txTo>	*Sets socket configuration parameters.* *<connId>: Integer. Socket connection identifier range 1-6* *<cid>: Integer. PDP context identifier range 1-5* *<pktSz>: Integer. Packet size to be used by the TCP/UDP/IP stack for data sending. Used for online data modes only. Range 0-1500* *0 automatically chosen by device* *1-1500 packet size in bytes* *<maxTo>: Integer. Exchange timeout. If there is no data exchange within this timeout period the connection is closed. Range of 0-65535.* *0 no timeout* *1-65535 timeout value in seconds (default 90seconds)* *<connTo>: Integer. Connection timeout. If we can't establish a connection to the remote within this timeout period, an error is raised. Range 0, 10-1200.* *0 no timeout* *12-1200 timeout value in hundreds of milliseconds (default 600)* *<txTo>: Integer. Data sending timout. Data are sent even if they are less than max packet size after this period. Used for online data mode only. Range 0-255.* *0 no timeout* *1-255 timout value in hundreds of milliseconds (default 50)* *Example:* *AT+SQNSCFG=3,3,300,90,600,5* *OK*
+SQNSCFG?	*Example:* *AT+SQNSCFG?* *+SQNSCFG: 1,1,300,90,600,50*

	+SQNSCFG: 2,1,300,90,600,50 *+SQNSCFG: 3,1,300,90,600,50* *+SQNSCFG: 4,1,300,90,600,50* *+SQNSCFG: 5,1,300,90,600,50* *+SQNSCFG: 6,1,300,90,600,50* *OK*
+SQNSCFG=?	*AT+SQNSCFG=?* *(1-6),(0-5),(0-1500),(0-65535),(0,10-1200),(0-255)* *OK*

2.8.4 AT+SQNSCFGEXT Socket Configuration Extended

Command	Possible response(s)
+SQNSCFGEXT=<connId>, <srMode>, <dataMode>, <keepalive>	*Sets the socket configuration extended parameters.* *<connId>: Integer. Socket connection identifier range 1-6* *<srMode>: Integer. Sring URC mode. Range 0-2* *0 normal mode (default),* *SRING : <connId>* *1 data amount mode,* *SRING : <connId>,<recData>* *2 data view mode,* *SRING : <connId>,<recData>,<data>* *<dataMode>: Integer. Data presentation mode.* *0 data represented as text (default)* *1 data represented as sequence of hexadecimal numbers (from 00 to FF)* *<keepalive>: Integer. Not used. Range 0-240*
+SQNSCFGEXT?	*Example:* *AT+SQNSCFGEXT?* *+SQNSCFGEXT: 1,0,0,0,0,0* *+SQNSCFGEXT: 2,0,0,0,0,0* *+SQNSCFGEXT: 3,0,0,0,0,0* *+SQNSCFGEXT: 4,0,0,0,0,0* *+SQNSCFGEXT: 5,0,0,0,0,0* *+SQNSCFGEXT: 6,0,0,0,0,0* *OK*
+SQNSCFGEXT=?	*Example:* *AT+SQNSCFGEXT=?* *+SQNSCFGEXT:* *(1-6),(0-2),(0-1),(0-240),(0),(0)*

	OK

2.8.5 AT+SQNSH Socket Shutdown

Command	Possible response(s)
+SQNSH=<connId>	*Shuts down socket. Socket can only be closed when it is in suspended mode. Trying to close an active socket will produce an error.* *<connId>: Integer. Socket connection identifier range 1-6* *Example:* *AT+SQNSH=1* *OK*
+SQNSH=?	*Example:* *AT+SQNSH=?* *+SQNSH: (1-6)* *OK*

2.8.6 AT+SQNRECV Receive Data in Command Mode

Command	Possible response(s)
+SQNSRECV=<connId>,<maxByte>	*Command allows user to read data arrived through a connected socket, buffered and not yet read because the module entered 'command mode' before reading them. The module is notified of the data by SRING URC whose format depends on the last +SQNSCFGEXT setting.* *<connId>: Integer. Socket connection identifier range 1-6* *<maxByte>: Integer. Max number of bytes to read. Range 1-1500* *Example:* *AT+SQNSRECV=1,10*
+SQNSRECV=?	*Example:* *AT+SQNSRECV=?* *+SQNSRECV: (1-6),(1-1500)* *OK*

2.8.7 AT+SQNSSEND Send Data in Command Mode

Command	Possible response(s)
+SQNSSEND=<connId>. . . <CTRL+Z>	*Allows device to send data through a connected socket in command mode.* *To complete the operation, send Ctrl-Z char (0x1A hex). To exit without writing message, send ESC char (0x1B hex). If data are successfully sent, then the response is OK. If data sending fails for some reason, an error code is reported.* *Note: the maximum number of bytes to send is 1500* *Note: it is possible to use +SQNSEND only if the connection was opened by +SQNSD, else the device will raise an error.* *Note: A byte corresponding to BS char (0x08) is treated with its corresponding meaning; therefore the previous byte will be cancelled (and the BS char itself will not be sent)* *<connId>: Integer. Socket connection identifier range 1-6*
+SQNSSEND=?	*Example:* *AT+SQNSSEND=?* *+SQNSSEND: (1-6)* *OK*

2.8.8 AT+SQNSS Socket Status

Command	Possible response(s)
+SQNSS	*Reports current status of teh sockets in the format:* *+SQNSS:* *<connId>,<state>,<locIP>,<locPort>,<remIP>,<remPort>* *[<CR><LF><connId>,<state>,<locIP>,<locPort>,<remIP>,<remPort> [. . .]]* *<connId>: Integer. Socket connection identifier range 1-6* *<state>: Integer. Actual state of the socket.* *0 Socket closed* *1 Socket active data connection* *2 Socket suspended* *3 Socket suspended with pending data* *4 Not used*

	5 Socket with incomign connection. Waiting for the 'user accept' or 'shutdown' command. *6 Socket opening in progress. The socket is not in closed state but still not in active or suspended or suspended with pending data state.* *<locIP>: Ip address associated by the context activation to the socket* *<locPort>: two meanings* *- The listening port if we put the socket in listen mode* *- The local port for the connection if we used the socket to connect to a remote machine* *<remIP>: when we are connected to a remote machine this is the remote IP address* *<remPort>: it is the port we are connected to on the remote machine.* *Example:* *AT+SQNSS* *+SQNSS: 1,0* *+SQNSS: 2,0* *+SQNSS: 3,0* *+SQNSS: 4,0* *+SQNSS: 5,0* *+SQNSS: 6,0* *OK*