ME910C1
AT Commands Reference Guide

80529ST10815A Rev.2-2017-09-14

APPLICABILITY TABLE

PRODUCT
ME910C1-NA
ME910C1-NV
ME910C1-E1
ME910C1-AU
ME910C1-J1
ME910C1-K1
ME910C1-N1
ME910C1-E2
ME910C1-WW
NE910C1-E1

Contents

DISCLAIMER

The information contained in this document is the proprietary information of Telit Communications S.p.A. and its affiliates ("TELIT").

The contents are confidential and any disclosure to persons other than the officers, employees, agents or subcontractors of the owner or licensee of this document, without the prior written consent of Telit, is strictly prohibited.

Telit makes every effort to ensure the quality of the information it makes available. Notwithstanding the foregoing, Telit does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information.

Telit disclaims any and all responsibility for the application of the devices characterized in this document, and notes that the application of the device must comply with the safety standards of the applicable country, and where applicable, with the relevant wiring rules.

Telit reserves the right to make modifications, additions and deletions to this document due to typographical errors, inaccurate information, or improvements to programs and/or equipment at any time and without notice.

Such changes will, nevertheless be incorporated into new editions of this document.

1 Introduction

1.1 Scope

This document is aimed at providing a detailed specification and a comprehensive listing as a **Reference** for the whole set of AT command

1.2 Audience

Readers of this document should be familiar with Telit modules and their ease of controlling by means of AT Commands.

1.3 Contact Information, Support

For general contact, technical support, to report documentation errors and to order manuals, contact Telit's Technical Support Center (TTSC) at:

TS-EMEA@telit.com
TS-NORTHAMERICA@telit.com
TS-LATINAMERICA@telit.com
TS-APAC@telit.com

Alternatively, use:
http://www.telit.com/en/products/technical-support-center/contact.php
For detailed information about where you can buy the Telit modules or for recommendations on accessories and components visit:
http://www.telit.com
To register for product news and announcements or for product questions contact Telit's Technical Support Center (TTSC).
Our aim is to make this guide as helpful as possible. Keep us informed of your comments and suggestions for improvements.
Telit appreciates feedback from the users of our information.

1.4 Document Organization

This document contains the following chapters:

Chapter 1: "Introduction" provides a scope for this document, target audience, contact and support information, and text conventions.

Chapter 2: "Overview" about the aim of this document and implementation suggestions.

Chapter 3: "AT Commands" The core of this **Reference** guide.

1.5 Text Conventions

<u>Danger – This information MUST be followed or catastrophic equipment failure or bodily injury may occur.</u>

Caution or Warning – Alerts the user to important points about integrating the module, if these points are not followed, the module and end user equipment may fail or malfunction.

Tip or Information – Provides advice and suggestions that may be useful when integrating the module.

All dates are in ISO 8601 format, i.e. YYYY-MM-DD.

1.6 Related Documents

- ETSI GSM 07.07 specification and rules
  http://www.3gpp.org/ftp/Specs/archive/07_series/07.07/

- ETSI GSM 07.05 specification and rules
  http://www.3gpp.org/ftp/Specs/archive/07_series/07.05/

- Hayes standard AT command set

1.7 Document History

Revision	Date	SW release	Changes
ISSUE #0	2017-03-10	30.00.xx0-Bxxx	First release
ISSUE #1	2017-08-14	30.00.xx1-Bxxx	Updated applicability table
ISSUE #2	2017-09-14	30.00.xx1-Bxxx 30.00.xx2	Updated applicability table Updated Commands and Aligned to first Mass Production Release 30.00.xx2

2 Overview

2.1 About the document

This document is to describe all AT commands implemented on the Telit wireless modules listed on the applicability table.

3 AT COMMANDS

The Telit wireless modules family can be controlled via the serial interface using the standard AT commands[1].
The Telit wireless modules family is compliant with:

1. Hayes standard AT command set, in order to maintain the compatibility with existing SW programs.
2. ETSI GSM 07.07 specific AT command and GPRS specific commands.
3. ETSI GSM 07.05 specific AT commands for SMS (Short Message Service) and CBS (Cell Broadcast Service)
4. FAX Class 1 compatible commands

Moreover Telit wireless modules family supports also Telit proprietary AT commands for special purposes.
The following is a description of how to use the AT commands with the Telit wireless module family.

3.1 Definitions

The following syntactical definitions apply:

<CR> **Carriage return character**, is the command line and result code terminator character, which value, in decimal ASCII between 0 and 255,is specified within parameter **S3**. The default value is 13.

<LF> **Linefeed character**, is the character recognised as line feed character. Its value, in decimal ASCII between 0 and 255, is specified within parameter **S4**. The default value is 10. The line feed character is output after carriage return character if verbose result codes are used (**V1** option used). Otherwise, if numeric format result codes are used (**V0** option used) it will not appear in the result codes.

<...> Name enclosed in angle brackets is a syntactical element. It does not appear in the command line.

[...] Optional subparameter of a command or an optional part of TA information response is enclosed in square brackets. Brackets themselves do not appear in the command line. When subparameter is not given in AT commands which have a Read command, new value equals to its previous value. In AT commands which do not store the values of any of their subparameters (called *action type* commands), and so don't have a Read command, action should be done on the basis of the recommended default setting of the subparameter.

3.2 AT Command Syntax

The syntax rules followed by Telit implementation of either Hayes AT commands or GSM/WCDMA commands are very similar to those of standard basic and extended AT commands. There are two types of extended command:

---

[1] The AT is an ATTENTION command and is used as a prefix to other parameters in a string. The AT command combined with other parameters can be set up in the communications package or typed in manually as a command line instruction.

- **Parameter type commands**. This type of commands may be "set" (to store a value or values for later use), "read" (to determine the current value or values stored), or "tested" (to determine ranges of values supported). Each of them has a test command (trailing '**=?**') to give information about the type of its subparameters; they also have a Read command (trailing '**?**') to check the current values of subparameters.

- **Action type commands**. This type of command may be "executed" or "tested":

  • "executed" to invoke a particular function of the equipment, which generally involves more than the simple storage of a value for later use

  • "tested" to determine whether or not the equipment implements the Action Command (in this case issuing the correspondent Test command - trailing **=?** - returns the **OK** result code), and if subparameters are associated with the action, the ranges of subparameters values that are supported.

  Action commands do not store the values of any of their possible subparameters.

  Moreover:
  o The response to the Test Command (trailing **=?**) may be changed in the future by Telit to allow the description of new values/functionalities
  o If all the subparameters of a parameter type command **+CMD** (or **#CMD** or **$CMD**) are optional, issuing **AT+CMD=<CR>** (or **AT#CMD=<CR>** or **AT$CMD=<CR>**) causes the **OK** result code to be returned and the previous values of the omitted subparameters to be retained.

3.2.1 String Type Parameters

A string must be enclosed between quotes or it will not be considered as a valid string type parameter input. According to V25.ter, space characters are ignored on the command line and may be used freely for formatting purposes, unless they are embedded on numeric or quoted string constants. Therefore a string containing a space character has to be enclosed between quotes to be considered a valid string type parameter (e.g. typing **AT+COPS=1,0,"A1"** is the same as typing **AT+COPS=1,0,A1**; typing **AT+COPS=1,0,"A BB"** is different from typing **AT+COPS=1,0,A BB**).

A small set of commands always requires writing the input string parameters within quotes. This is explicitly reported in the specific descriptions.

3.2.2 Command Lines

A command line is made up of three elements: the **prefix**, the **body** and the **termination character**.

The **command line prefix** consists of the characters "**AT**" or "**at**", or, to repeat the execution of the previous command line, the characters "**A/**" or "**a/**".

The **termination character** may be selected by a user option (parameter S3), the default being **<CR>**.
The basic structures of the command line are:

- **ATCMD1<CR>** where **AT** is the command line prefix, **CMD1** is the body of a **basic command** (nb: the name of the command never begins with the character "+") and **<CR>** is the command line terminator character

- **ATCMD2=10<CR>** where 10 is a subparameter

- **AT**+**CMD1 ;+CMD2=, ,10<CR>** These are two examples of **extended commands** (nb: the name of the command always begins with the character "+"[2]).
  They are delimit with semicolon. In the second command, the subparameter omitted.

- **+CMD1?<CR>** This is a Read command for checking current subparameter values

- **+CMD1=?<CR>** This is a test command for checking possible subparameter values

These commands might be performed in a single command line as shown below:
    ATCMD1 CMD2=10+CMD1;+CMD2=, ,10;+CMD1?;+CMD1=?<CR>

Anyway it is always preferable to separate into different command lines the basic commands and the extended commands.
Furthermore it is suggested to avoid placing several action commands in the same command line, because if one of them fails, then an error message is received but it is not possible to argue which one of them has failed the execution.

If command **V1** is enabled (verbose responses codes) and all commands in a command line have been performed successfully, result code **<CR><LF>OK<CR><LF>** is sent from the TA to the TE.
If subparameter values of a command are not accepted by the TA or command itself is invalid, or command cannot be performed for some reason, result code **<CR><LF>ERROR<CR><LF>** is sent and no subsequent commands in the command line are processed.

If command **V0** is enabled (numeric responses codes) and all commands in a command line have been performed successfully, result code **0<CR>** is sent from the TA to the TE.
If sub-parameter values of a command are not accepted by the TA or command itself is invalid, or command cannot be performed for some reason, result code **4<CR>** and no subsequent commands in the command line are processed.

---

[2] The set of **proprietary AT commands** differentiates from the standard one because the name of each of them begins with either "**@**", "**#**", "**$**" or "*****". **Proprietary AT commands** follow the same syntax rules as **extended commands**

In case of errors depending on ME operation, **ERROR** (or **4**) response may be replaced by **+CME ERROR: <err> or +CMS ERROR: <err>**.

NOTE: **The command line buffer accepts a maximum of 80 characters. If this number is exceeded none of the commands will be executed and TA returns ERROR.**

3.2.3 ME Error Result Code - +CME ERROR: <err>

This is NOT a command it is the error response to +**Cxxx 3gpp**
TS 27.007 commands.
Syntax: +**CME ERROR: <err>**
Parameter: **<err>** - error code can be either numeric or verbose (see +**CMEE**).The possible values
of **<err>** are reported in the below table:

Numeric Format	Verbose Format
General errors:	
0	phone failure
1	No connection to phone
2	phone-adaptor link reserved
3	operation not allowed
4	operation not supported
5	PH-SIM PIN required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	memory full
21	invalid index
22	not found
23	memory failure
24	text string too long
25	invalid characters in text string
26	dial string too long
27	invalid characters in dial string
30	no network service
31	network time-out
32	network not allowed - emergency calls only
40	network personalization PIN required
41	network personalization PUK required
42	network subset personalization PIN required
43	network subset personalization PUK required
44	service provider personalization PIN required
45	Service provider personalization PUK required
46	corporate personalization PIN required
47	corporate personalization PUK required
General purpose error:	
100	unknown
770	SIM invalid
GPRS related errors to a failure to perform an Attach:	
103	Illegal MS (#3)*

Numeric Format	Verbose Format
106	Illegal ME (#6)*
107	GPRS service not allowed (#7)*
111	PLMN not allowed (#11)*
112	Location area not allowed (#12)*
113	Roaming not allowed in this location area (#13)*
GPRS related errors to a failure to Activate a Context and others:	
132	service option not supported (#32)*
133	requested service option not subscribed (#33)*
134	service option temporarily out of order (#34)*
148	unspecified GPRS error
149	PDP authentication failure
150	invalid mobile class
Easy GPRS® related errors	
550	generic undocumented error
551	wrong state
552	wrong mode
553	context already activated
554	stack already active
555	activation failed
556	context not opened
557	cannot setup socket
558	cannot resolve DN
559	time-out in opening socket
560	cannot open socket
561	remote disconnected or time-out
562	connection failed
563	tx error
564	already listening
Network survey errors	
657	Network survey error (No Carrier)*
658	Network survey error (Busy)*
659	Network survey error (Wrong request)*
660	Network survey error (Aborted)*
Supplementary service related error	
257	network rejected request
258	retry operation
259	invalid deflected to number
260	deflected to own number
261	unknown subscriber
262	service not available
263	unknown class specified
264	unknown network message
AT+COPS test command related error	
680	LU processing
681	Network search aborted
682	PTM mode
AT+WS46 test command related error	
683	Active call state
684	RR connection Established

*(Values in parentheses are 3gpp TS 24.008 cause codes) Message Service Failure Result Code - +CMS ERROR: <err>

3.2.3.1   Message Service Failure Result Code -

+CMS ERROR: <err>

This is NOT a command; it is the error response to +Cxxx 3gpp TS 27.005 commands.
Syntax: +CMS ERROR: <err>
Parameter: <err> - numeric error code. The **<err>** values are reported in the below table:

Numeric Format	Meaning
0...127	3gpp TS 24.011 Annex E-2 values
128...255	3gpp TS 23.040 sub clause 9.2.3.22 values
300	ME failure
301	SMS service of ME reserved
302	operation not allowed
303	operation not supported
304	invalid PDU mode parameter
305	invalid text mode parameter
310	SIM not inserted
311	SIM PIN required
312	PH-SIM PIN required
313	SIM failure
314	SIM busy
315	SIM wrong
316	SIM PUK required
317	SIM PIN2 required
318	SIM PUK2 required
320	memory failure
321	invalid memory index
322	memory full
330	SMSC address unknown
331	no network service
332	network time-out
340	no +CNMA acknowledgement expected
500	unknown error

3.2.4 Information Responses and Result Codes

The TA response, in case of verbose response format enabled, for the previous examples command line could be as shown below:

• Information response to **+CMD1?:**

<CR><LF>+CMD1:2,1,10<CR><LF>

• Information response to **+CMD1=?**

<CR><LF>+CMD1(0-2),(0,1),(0-15)<CR><LF>

• Final result code:

<CR><LF>OK<CR><LF>

Moreover, there are two other types of result codes:

- *Result code*s that inform about progress of TA operation (e.g. connection establishment **CONNECT**)

- *Result code*s that indicate occurrence of an event not directly associated with issuance of a command from TE (e.g. ring indication **RING**).

Here are the basic result codes according to ITU-T V25Ter recommendation:

Result Codes	
Numeric form	Verbose form
0	OK
1	CONNECT
2	RING
3	NO CARRIER
4	ERROR
6	NO DIALTONE
7	BUSY
8	NO ANSWER

3.2.5 Command Response Time-Out

Every command issued to the Telit modules return a result response if response codes are enabled (default). The time needed to process the given command and return the response varies depending on the command type. Commands that do not interact with the SIM or the network, and involve only internal set up settings or readings, have an immediate response, depending on SIM configuration (e.g., number of contacts stored in the phonebook, number of stored SMS), or on the network the command may interact with.

In the table below are listed only the commands for which the interaction with the SIM or the network could lead to long response timings. When not otherwise specified, timing is referred to set command.
For phonebook and SMS writing and reading related commands, timing is referred to commands issued after phonebook sorting is completed.
For DTMF sending and dialing commands timing is referred to module registered on network ("AT+CREG?" answer is "+CREG: 0,1" or "+CREG: 0,5").

Command	Estimated maximum time to get response (Seconds)
+COPS	125 (test command)
+CLCK	15 (SS operation) 5 (FDN enabling/disabling)
+CPWD	15 (SS operation) 5 (PIN modification)
+CPIN	30
+CSCA	5 (read and set commands)
+CSAS	5
+CMGS	120 after CTRL-Z; 1 to get '>' prompt
+CMGW	5 after CTRL-Z; 1 to get '>' prompt
+CMGD	5 (single SMS cancellation) 25 (cancellation of 50 SMS)
+CMGR	5
+CMGL	100
+CGACT	150
+CGATT	140
D	120 (voice call) Timeout set with ATS7 (data call)
+COPN	10
+CRSM	180

+WS46	10
#QDNS	170
#SGACT	150
#SH	10
#SD	140 (DNS resolution + connection timeout set with AT#SCFG)

3.2.6 Command Issuing Timing

The chain '*Command* ➔ *Response*' shall always be respected and a new command must not be issued before the module has terminated all the sending of its response result code (whatever it may be).

This applies especially to applications that "sense" the **OK** text and therefore may send the next command before the complete code **<CR><LF>OK<CR><LF>** is sent by the module.

It is advisable anyway to wait for at least 20ms between the end of the reception of the response and the issue of the next AT command.

If the response codes are disabled and therefore the module does not report any response to the command, then at least the 20ms pause time must be respected.

During command mode, due to hardware limitations, under severe CPU load, the serial port can loose some characters if placed in autobauding at high speeds. Therefore, if you encounter this problem fix the baud rate with **+IPR** command.

3.3 Storage

3.3.1 Factory Profile and User Profiles

The Telit wireless modules store the values set by several commands in the internal non-volatile memory (NVM), allowing to remember these settings even after power off. In the NVM, these values are set either as **factory profile** or as **user profiles**: there are **two customizable user profiles** and **one factory profile** in the NVM of the device. By default, the device will start with user profile 0 equal to factory profile.
For backward compatibility, each profile is divided into two sections.
One **base section** that was present historically and one that is saved and restored on early release of the code, and the **extended section** that includes all the remaining values.

Command **&P** is used to set the profile to be loaded at startup. **&P** instructs the device to load at startup the full profile: **base + extended sections**.
The **&F** command resets to factory profile values only the commands of the base section of profile, while the **&F1** resets to factory profile values the full set of base + extended section commands.

The values set by other commands are stored in NVM outside the profile: some of them are stored always, some other are stored issuing specific commands (**+CSAS**). All of these values are read at power-up.

The values set by following commands are stored in the profile base section:

AUTOBAUD :		**+IPR**
COMMAND ECHO:	**E**	
RESULT MESSAGES:	**Q**	
VERBOSE MESSAGES:		**V**
FLOW CONTROL OPTIONS:	**+IFC**	
DTR (C108) OPTIONS:	**&D**	
DCD (C109) OPTIONS:	**&C**	
POWER SAVING:	**+CFUN**	
S REGISTERS:		**S0; S3;S4;S5**
FLOW CONTROL:	**&K**	
CHARACTER FORMAT:		**+ICF**

The values set by following commands are stored in the profile extended section:

+CSCS, +CMER, +CMEE, +CGEREP, +CGREG, +CMGF, +CEREG, +CSDH, +CNMI, #PSNT, #QSS, #SKIPESC, #SIMDET, #E2SLRI, + CIND, #NITZ, +CLIP, +CLIR

The values set by following commands are automatically stored in NVM, without issuing any storing command and independently from the profile (unique values), and automatically restored at startup:

+WS46 , +COPS, +CPLS, +CGDCONT, +CEMODE, #ALARMPIN, #CCLKMODE, +CALA, #SCFGEXT, #SCFGEXT2, #SCFGEXT3, #SSLEN, #DNS, #HTTPCFG, $GPSGLO, $SLP, $SLPTYPE, $SUPLSEC, $SLPTYPE, $LICLS, $SUPLV, $SUPLSEC, #BND, #SCFG, #TEMPMON, #EPASSW

The values set by following commands are stored in NVM on demand, issuing specific commands and independently from the profile:

+CSCA, **+CSMP**
Stored by +CSAS[3] command and restored by +CRES[4] command.
.

---

[3] Both commands +CSAS and +CRES deal with non-volatile memory, intending for it either the NVM and the SIM storage.

3.4 AT Commands References

3.4.1 Command Line General Format

3.4.1 Command Line Prefixes

3.4.1.1.1   *Starting a Command Line - AT*

AT - Starting A Command Line	
AT	The prefix **AT**, or **at**, is a two-character abbreviation (**ATtention**), always used to start a command line to be sent from TE to TA
Reference	3GPP TS 27.007

3.4.1.1.2   *Last Command Automatic Repetition - A/*

A/ - Last Command Automatic Repetition	
A/	If the prefix **A/** or **a/** is issued, the MODULE immediately executes once again the body of the preceding command line. No editing is possible and no termination character is necessary. A command line may repeated multiple times through this mechanism, if desired. If **A/** is issued before any command line has been executed, the preceding command line is assumed to have been empty (that results in an **OK** result code). **Note:** this command works only at fixed IPR. **Note:** the custom command **#/** has been defined, it causes the last command to be executed again too; but it does not need a fixed IPR.
Reference	V25ter

3.4.3 Hayes Compliant AT Commands

3.4.3.1 Generic Modem Control

3.4.3.1.1 *Set To Factory-Defined Configuration - &F*

&F - Set To Factory-Defined Configuration	

AT&F[<value>]	Execution command sets the configuration parameters to default values specified by manufacturer; it takes in consideration hardware configuration switches and other manufacturer-defined criteria.
	
	Parameter:
	<value>:
	  0 – just factory profile's base section parameters are considered.
	  1 - Either the factory profile base section and the extended section are considered (full factory profile).
	
	Note: if parameter **<value>** is omitted, the command has the same behaviour as **AT&F0**

Reference	V25ter.

3.4.3.1.2 *Soft Reset - Z*

Z - Soft Reset	
ATZ[<n>]	Execution command loads the base section of the specified user profile and the extended section of the default factory profile. **Parameter:** **<n>** 0..1 - user profile number **Note:** any call in progress will terminated. **Note:** if parameter **<n>** omitted, the command has the same behaviour as **ATZ0**.
Reference	V25ter.

3.4.3.1.3 *Default Reset Full Profile Designation - &P*

&P - Default Reset Full Profile Designation	
AT&P[<n>]	Execution command defines which full profile will be loaded on startup.

<table>
<tr><td colspan="2">**&P - Default Reset Full Profile Designation**</td></tr>
<tr><td></td><td>**Parameter:**
<n>
 0...1 – profile number: the wireless module is able to store 2 full configurations (see command **&W**).

Note: differently from command **Z<n>**, which loads just once the desired profile, the one chosen through command **&P** will be loaded on every startup.

Note: if parameter is omitted, the command has the same behaviour as **AT&P0**</td></tr>
<tr><td>**Reference**</td><td>Telit Specifications</td></tr>
</table>

3.4.3.1.4 *Manufacturer Identification - +GMI*

+GMI - Manufacturer Identification	
AT+GMI	Execution command returns the manufacturer identification.
Reference	V.25ter

3.4.3.1.5 *Model Identification - +GMM*

+GMM - Model Identification	
AT+GMM	Execution command returns the model identification.
Reference	V.25ter

3.4.3.1.6 *Revision Identification - +GMR*

+GMR - Revision Identification	
AT+GMR	Execution command returns the software revision identification.
Reference	V.25ter

3.4.3.1.7 *Serial Number - +GSN*

+GSN - Request Product Serial Number Identification	
AT+GSN	Execution command returns the device board serial number. **Note**: The number returned is not the IMSI, it is only the board number.
Reference	V.25ter

3.4.3.1.8 *Display Current Base Configuration and Profile - &V*

&V - Display Current Base Configuration And Profile	
AT&V	Execution command returns some of the base configuration parameters settings.

3.4.3.2 DTE - Modem Interface Control

3.4.3.2.1 *Command Echo - E*

E - Command Echo	

ATE[<n>]	Set command enables/disables the command echo.
	
	Parameter:
	<n>
	  0 - Disables command echo
	  1 - Enables command echo (factory default), hence command sent to the device echo back to the **DTE** before the response is given.
	
	Note: if parameter is omitted, the command has the same behaviour of **ATE0**

ATE?	Read command returns the current value of n.
Reference	V25ter

3.4.3.2.2 *Quiet Result Codes - Q*

Q - Quiet Result Codes	

ATQ[<n>]	Set command enables or disables the result codes.
	
	Parameter:
	<n>
	  0 - enables result codes (factory default)
	  1 - disables result codes
	  2 - disables result codes
	     (only for backward compatibility)

Q - Quiet Result Codes	
	Note: After issuing either **ATQ1** or **ATQ2** every information text transmitted in response to commands is not affected **Note:** if parameter is omitted, the command has the same behaviour of **ATQ0**
Example	*After issuing ATQ1 or ATQ2* AT+CGACT=? **+CGACT: (0-1) nothing is appended to the response**
Reference	V25ter

3.4.3.2.3 *Response Format - V*

V - Response Format	
ATV[<n>]	Set command determines the contents of the header and trailer transmitted with result codes and information responses. It also determines if result codes are transmitted in a numeric form or an alphanumeric form (see [§3.2.4 Information Responses And Result Codes] for the table of result codes). **Parameter:** **<n>** 0 - limited headers and trailers and numeric format of result codes <table><tr><td>information responses</td><td>**<text><CR><LF>**</td></tr><tr><td>result codes</td><td>**<numeric code><CR>**</td></tr></table> 1 - full headers and trailers and verbose format of result codes (factory default) <table><tr><td>information responses</td><td>**<CR><LF>** **<text><CR><LF>**</td></tr><tr><td>result codes</td><td>**<CR><LF>** **<verbose code><CR><LF>**</td></tr></table> **Note:** the **<text>** portion of information responses is not affected by this setting. **Note:** if parameter is omitted, the command has the same behaviour of ATV0
Reference	V25ter

3.4.3.2.5   *Identification Information - I*

I - Identification Information	

ATI[<n>]	Execution command returns one or more lines of information text followed by a result code.
	
	Parameter:
	<n>
	  0 - numerical identifier
	  1 - module checksum
	  2 - checksum check result
	  3 - manufacturer
	  4 - product name
	  5 - DOB version
	
	Note: if parameter is omitted, the command has the same behaviour of **ATI0**

Reference	V25ter

3.4.3.2.6   *Data Carrier Detect (DCD) Control - &C*

&C - Data Carrier Detect (DCD) Control	

AT&C[<n>]	Set command controls the RS232 **DCD** output behaviour.
	
	Parameter:
	<n>
	  0 - **DCD** remains **high** always.
	  1 - **DCD** follows the Carrier detect status: if carrier detected **DCD** is high, otherwise **DCD** is **low**. (factory default)
	  2 – **DCD** is always **high** except for 1sec 'wink' when a data call is disconnected.
	
	Note: if parameter is omitted, the command has the same behaviour of **AT&C0**

Reference	V25ter

3.4.3.2.7   *Data Terminal Ready (DTR) Control - &D*

&D - Data Terminal Ready (DTR) Control	

AT&D[<n>]	Set command controls the Module behaviour to the RS232 **DTR** transitions.
	
	Parameter:
	<n>
	  0 - **DTR** transmitions ignored (factory default)
	  1 - When the MODULE is connected, the **High** to **Low** transition of **DTR** pin sets the device in command mode and the current connection is NOT closed.

&D - Data Terminal Ready (DTR) Control	
	2 - When the MODULE is connected, the **High** to **Low** transition of **DTR** pin sets the device in command mode and the current connection is closed. 3 - **C108/1** operation enabled. 4 - **C108/1** operation disabled. **Note:** if **AT&D2** been issued and the **DTR** has been tied **Low**, autoanswering is inhibited and it is possible to answer only issuing command **ATA**. **Note:** if parameter is omitted, the command has the same behaviour of **AT&D0**
Reference	V25ter

3.4.3.2.8 *Flow Control - &K*

&K - Flow Control	SELINT 2
AT&K[<n>]	Set command controls the RS232 flow control behaviour. **Parameter:** **<n>** **0** - no flow control **3** - hardware bi-directional flow control (both RTS/CTS active) (factory default) **Note**: if parameter is omitted, the command has the same behaviour as AT&K0. **Note:** &K has no Read Command. To verify the current setting of &K, simply check the settings of the active profile issuing AT&V. **Note:** Hardware flow control (AT&K3) is not active in command mode.
Note: &K has no Read Command. To verify the current setting of &K, simply check the settings of the active profile issuing AT&V. **Note:** Hardware flow control (AT&K3) is not active in command mode.	

3.4.3.2.9 *Fixed DTE Interface Rate - +IPR*

+IPR - Fixed DTE Interface Rate	
AT+IPR=<rate>	Set command specifies the **DTE** speed (UART only) at which the device accepts commands during command mode operations. The command could be use to fix the **DTE-DCE** interface speed. **Note**: DTE speed of USB does not change. **Parameter:**

+IPR - Fixed DTE Interface Rate	

	\<rate\>
	  300
	  600
	  1200
	  2400
	  4800
	  9600
	  19200
	  38400
	  57600
	  115200 (default)
	  230400
	  460800
	  921600
	  2900000
	  3200000
	  3686400
	  4000000
	
	If **\<rate\>** specified **DTE-DCE** speed fixed to that speed, hence no speed auto-detection (autobauding) enabled.

AT+IPR?	Read command returns the current value of **+IPR** parameter.
AT+IPR=?	Test command returns the list of supported autodetectable **\<rate\>** values and the list of fixed-only **\<rate\>** values in the format: **+IPR**:(list of supported autodetectable **\<rate\>** values), (list of fixed-only **\<rate\>** values)
Reference	V25ter

3.4.3.2.10   *DTE-Modem Local Flow Control - +IFC*

+IFC - DTE-Modem Local Flow Control	
AT+IFC=\<by_te\>, \<by_ta\>	Set command selects the flow control behaviour of the serial port in both directions: from DTE to modem (\<by_ta\> option) and from modem to DTE (\<by_te\>) **Parameters:** **\<by_te\>** - flow control option for the data received by DTE. **0** - flow control None **2** - C105 (RTS) (factory default) **\<by_ta\>** - flow control option for the data sent by modem **0** - flow control None **2** - C106 (CTS) (factory default) **Note**: only possible commands are AT+IFC=0,0 and AT+IFC=2,2.
AT+IFC?	Read command returns active flow control settings.

AT+IFC=?	Test command returns all supported values of the parameters <by_te> and <by_ta>.
Reference	**V25ter**

3.4.3.2.11 *DTE-Modem Character Framing - +ICF*

<mark>+ICF - DTE-Modem Character Framing</mark>	

AT+ICF=
<format>
[,<parity>]
	Set command defines the asynchronous character framing used when autobauding is disabled.
	The LE9x0 family supports only the 8 Data, 1 Stop setting.
	
	Parameters:
	<format> - determines the number of bits in the data bits, the presence of a parity bit, and the number of stop bits in the start-stop frame.
	3 - 8 Data, 1 Stop (default)
	<parity> - determines how the parity bit is generated and checked, if present; setting this subparameter has no meaning.
	  0 - Odd (not supported)
	  1 - Even (not supported)

AT+ICF?	Read command returns current settings for subparameters **<format>** and **<parity>**. The current setting of subparameter **<parity>** will always represented as 0.
AT+ICF=?	Test command returns the ranges of values for the parameters **<format>** and **<parity>**
Reference	V25ter
Example	AT+ICF = 3 - 8N1 (default) AT+ICF=? +ICF: (3)

3.4.3.3 Call Control

3.4.3.3.1 *Dial - D*

<mark>D – Dial</mark>	
ATD<number>[;]	Execution command starts a call to the phone number given as parameter. **Parameter:** **<number>** - phone number to be dialed **Note:** the numbers accepted are 0-9 and *,#,"A","B","C","+".

D – Dial	
	Note: for backwards compatibility with landline modems modifiers "T", "P","R", ",", "W", "!", "@" are accepted but have no effect.
ATD><str>[;]	Issues a call to phone number which corresponding alphanumeric field is **<str>**; all available memories will search for the correct entry. If "**;**" is present a **voice** call is performed. **Parameter:** **<str>** - alphanumeric field corresponding to phone number. It must be enclosed in quotation marks. **Note:** parameter **<str>** is case sensitive. **Note:** used character set should be the one selected with **+CSCS**.
ATD><mem><n>[;]	Issues a call to phone number in phonebook memory storage **<mem>**, entry location **<n>** (available memories may be queried with **AT+CPBS=?**). If "**;**" is present a **voice** call is performed. **Parameters:** **<mem>** - phonebook memory storage; "SM" - SIM/UICC phonebook "FD" - SIM/USIM fixed dialing phonebook "LD" - SIM/UICC last dialing phonebook "MC" – Missed calls list "RC" - Received calls list "DC" - MT dialled calls list "ME" - MT phonebook "EN" - SIM/USIM (or MT) emergency number(+CPBW is not be applicable for this storage) "ON" - SIM (or MT) own numbers (MSI torage may be available through +CNUM also). "MB" - Mailbox numbers stored on SIM. If this service is provided by the SIM (see **#MBN**). **<n>** - entry location; it should be in the range of locations available in the memory used.
ATD><n>[;]	Issue a call to a phone number on entry location **<n>** of the active phonebook memory storage (see **+CPBS**). If "**;**" is present a **voice** call is performed. **Parameter:** **<n>** - active phonebook memory storage entry location; it should be in the range of locations available in the active phonebook memory storage.
ATDL	Issues a call to the last number dialed.
ATDS=<nr>[;]	Issues a call to the number stored in the MODULE internal phonebook position number **<nr>**. **If "<;" is present a voice** call is performed.

D – Dial	
	Parameter: **<nr>** - internal phonebook position to be called
ATD<number>I[;] **ATD<number>i[;]**	Issues a call overwriting the CLIR supplementary service subscription default value for this call If "**;**" is present a **voice** call is performed. **I** - invocation, restrict CLI presentation **i** - suppression, allow CLI presentation
ATD<number>G[;] **ATD<number>g[;]**	Issues a call checking the CUG supplementary service information for the current call. Refer to **+CCUG** command. If "**;**" is present a **voice** call is performed.
ATD*<gprs_sc> **[*<addr>]** **[*[<L2P>]** **[*[<cid>]]]]#**	This command is specific of GPRS functionality and causes the **MT** to perform whatever actions are necessary to establish communication between the **TE** and the external PDN. **Parameters:** **<gprs_sc>** - GPRS Service Code, a digit string (value 99) which identifies a request to use the GPRS **<addr>** - string that identifies the called party in the address space applicable to the PDP. **<L2P>** - a string which indicates the layer 2 protocol to be used (see **+CGDATA** command). For communications software that does not support arbitrary characters in the dial string, the following numeric equivalents shall be used: 1 - PPP **<cid>** - a digit which specifies a particular **PDP** context definition (see **+CGDCONT** command).
Example	*To dial a number in SIM phonebook entry 6:* ATD>SM6 OK *To have a voice call to the 6-th entry of active phonebook:* ATD>6; OK *To call the entry with alphanumeric field "Name":* ATD>"Name"; OK
Reference	V25ter.

3.4.3.4 S Parameters

Basic commands that begin with the letter "**S**" are known as "**S-Parameters**". The number following the "**S**" indicates the "parameter number" being **referenced**. If the number is not recognized as a valid parameter number, an **ERROR** result code is issued.

If no value is given for the subparameter of an **S-Parameter**, an **ERROR** result code will be issued and the stored value is left unchanged.

> *NOTE:* what follows is a special way to select and set an **S-Parameter:**

1) **ATS*n*<CR>** selects *n* as current parameter number. If the value of *n* is in the range (0, 2, 3, 4, 5, 7, 10, 12, 25, 30, 38), this command establishes **S*n*** as last selected parameter. Every values out of this range and lower than 256 can be used but have no meaning and are maintained only for backward compatibility with landline modems.
2) **AT=<value><CR>** or **ATS=<value><CR>** set the contents of the selected **S-parameter**

Example:
ATS7<CR> establishes **S7** as last selected parameter.

Reference: V25ter and RC56D/RC336D

3.4.3.4.1 *Number of Rings to Auto Answer - S0*

S0 - Number Of Rings To Auto Answer	

ATS0=[<n>]	Set command sets the number of rings required before device automatically answers an incoming call.
	
	Parameter:
	<n> - number of rings
	  0 - auto answer disabled (factory default)
	  1..255 - number of rings required before automatic answer.

ATS0?	Read command returns the current value of **S0 parameter**.
Reference	V25ter

3.4.3.4.2 *Command Line Termination Character - S3*

S3 - Command Line Termination Character	
ATS3=[<char>]	Set command sets the value of the character recognized by the device as command line terminator and generated by the device as part of the header, trailer, and terminator for result codes and information text, along with **S4 parameter**. **Parameter:** **<char>** - command line termination character (decimal ASCII) 0..127 - factory default value is 13 (ASCII **<CR>**) **Note:** the "previous" value of **S3** used to determine the command line termination character for entering the command line containing the **S3** setting command. However the result code issued shall use the "new" value of **S3** (as set during the processing of the command line)
ATS3?	Read command returns the current value of **S3 parameter**. **Note:** the format of the numbers in output is always 3 digits, left-filled with 0s
Reference	V25ter

3.4.3.4.3 *Response Formatting Character - S4*

S4 - Response Formatting Character	
ATS4=[<char>]	Set command sets the value of the character generated by the device as part of the header, trailer, and terminator for result codes and information text, along with the **S3 parameter**. **Parameter:** **<char>** - response formatting character (decimal ASCII) 0..127 - factory default value is 10 (ASCII **LF**) **Note:** if the value of **S4** changed in a command line, the result code issued in response of that command line will use the new value of **S4**.
ATS4?	Read command returns the current value of S4 parameter. **Note:** the format of the numbers in output is always 3 digits, left-filled with 0s
Reference	V25ter

3.4.3.4.4 *Command Line Editing Character - S5*

S5 - Command Line Editing Character	
ATS5=[<char>]	Set command sets the value of the character recognized by the device as a request to delete from the command line the immediately preceding character. **Parameter:**

S5 - Command Line Editing Character	
	<char> - command line editing character (decimal ASCII) 0..127 - factory default value is 8 (ASCII **BS**)
ATS5?	Read command returns the current value of **S5 parameter**. **Note:** the format of the numbers in output is always 3 digits, left-filled with 0s
Reference	V25ter

3.4.4 3GPP TS 27.007 AT Commands

3.4.4.1 General

3.4.4.1.1 *Request Manufacturer Identification - +CGMI*

+CGMI - Request Manufacturer Identification	
AT+CGMI	Execution command returns the "device manufacturer identification code" without command echo.
AT+CGMI=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.4.4.1.2 *Request Model Identification - +CGMM*

+CGMM - Request Model Identification	
AT+CGMM	Execution command returns the "device model identification code" without command echo.
AT+CGMM=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.4.4.1.3 *Request Revision Identification - +CGMR*

+CGMR - Request Revision Identification	
AT+CGMR	Execution command returns "device software revision number" without command echo.
AT+CGMR=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.4.4.1.4 *Request Product Serial Number Identification - +CGSN*

+CGSN - Request Product Serial Number Identification	
AT+CGSN	Execution command returns the product serial number, identified as the IMEI of the mobile, without command echo.
AT+CGSN=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.4.4.1.5 *Select TE Character Set - +CSCS*

+CSCS - Select TE Character Set	
AT+CSCS= [<chset>]	Set command sets the current character set used by the device. **Parameter:** **<chset>** - character set **"GSM"** - "GSM" - GSM default alphabet (3GPP TS 03.38/23.008). **"IRA"** - **international Reference** alphabet (ITU-T T.50) Quoted string (For example, "AB" equals two 8-bit characters with decimal values 65, 66). **"8859-1"** - ISO 8859 Latin 1 character set. **"PCCP437"** - PC character set Code Page 437. **"UCS2"** - 16-bit universal multiple-octet coded character set (ISO/IEC10646). HEX representation (For example,00410042 equals two 16-bit characters with decimal values 65,66).
AT+CSCS?	Read command returns the current value of the active character set.
AT+CSCS=?	Test command returns the supported values for parameter **<chset>**.
Example	AT+CSCS=? +CSCS: ("GSM","IRA","8859-1","PCCP437","UCS2") OK AT+CSCS? +CSCS: "IRA" OK AT+CPBW=1,"8475763000",129,"Lin Zhao" OK AT+CSCS="UCS2" OK AT+CPBR=1 +CPBR: 1,"8475763000",129,004C006E006E0020005A00680061006F

+CSCS - Select TE Character Set

	OK
	AT+CSCS="IRA"
	OK
	AT+CPBR=1
	+CPBR: 1,"8475763000",129,"Lin Zhao"
	OK
Reference	3GPP TS 27.007

3.4.4.1.6 *Request International Mobile Subscriber Identity (IMSI) - +CIMI*

+CIMI - Request International Mobile Subscriber Identify (IMSI)

AT+CIMI	Execution command returns the value of the Internal Mobile Subscriber Identity stored in the SIM without command echo. **Note:** a SIM card must be present in the SIM card housing. Otherwise, the command returns **ERROR**.
AT+CIMI=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.4.4.1.1 *Read ICCID (Integrated Circuit Card Identification) - #CCID*

#CCID - Read ICCID

AT#CCID	Execution command reads on SIM the ICCID (card identification number that provides a unique identification number for the SIM)
AT#CCID=?	Test command returns the **OK** result code.
Example	AT#CCID #CCID: 8982050702100167684F OK

3.4.4.1.2 *Serial and Software Version Number - +IMEISV*

+IMEISV - Serial and Software Version Number

AT+IMEISV	Execution command returns returns the IMEISV (International Mobile station Equipment Identity and Software Version number).
Example	At+imeisv +IMEISV: 3540660590080701

+IMEISV - Serial and Software Version Number	

3.4.4.1.1 PCCA STD-101 Select Wireless Network - +WS46

+WS46 - PCCA STD-101 Select Wireless Network	

AT+WS46=[<n>]	Set command selects the cellular network (Wireless Data Service, WDS) to operate with the **TA** (WDS-Side Stack Selection).
	
	Parameter:
	<n> - integer type, it is the WDS-Side Stack to be used by the **TA**.
	  12 - GSM Digital Cellular Systems (GERAN only)
	  28 E-UTRAN only
	  30 GERAN and E-UTRAN
	
	Note: <n> parameter setting is stored in NVM and available at next reboot.
	Note: 4G only products support **<n>** parameter value 28 only.
	Note: 4G/2G only products support **<n>** parameter values 12, 28 and 30 only. 30 is factory default
	
	Note: for NA (North America) products supporting AT&T requirement 13340 about RAT Balancing and EF-RAT Mode, the value **<n>** stored with AT+WS46 command can be changed and overwritten in case of full SIM read (e.g.: power on, AT+CFUN=4/AT+CFUN=1 sequence, SIM ejection/SIM insertion sequence).

AT+WS46?	Read command reports the currently selected cellular network, in the format: **+ WS46: <n>**
AT+WS46=?	Test command reports the range for the parameter **<n>**.
Reference	3GPP TS 27.007

3.4.4.2 Network Service Handling

3.4.4.2.1 Subscriber Number - +CNUM

+CNUM - Subscriber Number	
AT+CNUM	Execution command returns the MSISDN (if the phone number of the device has been stored in the SIM card) in the format: **+CNUM: <alpha>,<number>,<type>[<CR><LF>** **+CNUM: <alpha>,<number>,<type>[…]]**

+CNUM - Subscriber Number	

	where:
	 <alpha> - alphanumeric string associated to **<number>**; used character set should be the one selected with **+CSCS**.
	 <number> - string containing the phone number in the format **<type>**
	 <type> - type of number:
	   129 - national numbering scheme
	   145 - international numbering scheme (contains the character "+").

AT+CNUM=?	Test command returns the **OK** result code
Example	AT+CNUM +CNUM: "PHONENUM1","2173848500",129 +CNUM: "FAXNUM","2173848501",129 +CNUM: "DATANUM","2173848502",129
Reference	3GPP TS 27.007

3.4.4.2.2 *Read Operator Names - +COPN*

+COPN - Read Operator Names	
AT+COPN	Execution command returns the list of operator names from the **ME** in the format: **+COPN: <numeric1>,<alpha1>[<CR><LF>** **+COPN: <numeric2>,<alpha2>[…]]** **where:** **<numericn>** - string type, operator in numeric format (see **+COPS**) **<alphan>** - string type, operator in long alphanumeric format (see **+COPS**) **Note:** each operator code **<numericn>** that has an alphanumeric equivalent **<alphan>** in the ME memory is returned
AT+COPN=?	Test command returns the **OK** result code
Reference	3GPP TS 27.007

3.4.4.2.3 *Operator Selection - +COPS*

+COPS - Operator Selection	

AT+COPS=
 [<mode>
 [,<format>
 [,<oper>[,< AcT>]]]]
	Set command forces an attempt to select and register the network operator.
	 <mode> parameter defines whether the operator selection is done automatically or it is forced by this command to operator **<oper>**.
	 The operator **<oper>** shall be given in format **<format>**.
	
	 Parameters:
	 <mode>
	   0 - automatic choice (the parameter **<oper>** will be ignored) (factory default)
	   1 - manual choice (**<oper>** field shall be present)

+COPS - Operator Selection	
	2 - deregister from network; the MODULE is kept unregistered until a **+COPS** with **<mode>=0, 1** or **4** is issued 3 - set only **<format>** parameter (the parameter **<oper>** will be ignored) 4 - manual/automatic (**<oper>** field shall be present); if manual selection fails, automatic mode (**<mode>=0**) is entered **<format>** 0 - alphanumeric long form (max length 16 digits) 2 - Numeric 5 or 6 digits [country code (3) + network code (2 or 3)] **<oper>**: network operator in format defined by **<format>** parameter. **<AcT>** access technology selected: 0 GSM 7 E-UTRAN 8 CAT M1 9 E-UTRAN (NB-S1 mode) (NB1)
AT+COPS?	Read command returns current value of **<mode>,<format>,<oper>** and **<AcT>** in format **<format>**; if no operator is selected, **<format>**, **<oper>** and **<AcT>** are omitted **+COPS: <mode>[, <format>, <oper>,< AcT>]**
AT+COPS=?	Test command returns a list of quadruplets, each representing an operator present in the network. The quadruplets in the list are separated by commas: **+COPS: [list of supported (<stat> ,<oper (in <format>=0)>,,** **<oper (in <format>=2)>,< AcT>)s][,,(list of supported <mode>s),** **(list of supported<format>s)]** where **<stat>** - operator availability 0 - unknown 1 - available 2 - current 3 - forbidden Note: since with this command a network scan is done, this command may require some seconds before the output is given.
Reference	3GPP TS 27.007

3.4.4.2.4 *Facility Lock/Unlock - +CLCK*

+CLCK - Facility Lock/Unlock	
AT+CLCK=	Execution command used to lock or unlock a **ME** or a network facility.

+CLCK - Facility Lock/Unlock	

<fac>,
<mode>
[,<passwd>
[,<class>]]
	Parameters:
	<fac> - facility
	  "SC" - SIM (PIN request) (device asks SIM password at power-
	      up and when this lock command issued)
	  "AO" - BAOC (Barr All Outgoing Calls)
	  "OI" - BOIC (Barr Outgoing International Calls)
	  "OX" - BOIC-exHC (Barr Outgoing International Calls except
	      to Home Country)
	  "AI" - BAIC (Barr All Incoming Calls)
	  "IR" - BIC-Roam (Barr Incoming Calls when Roaming outside
	      the home country)
	  "AB" - All Barring services (applicable only for **<mode>=0**)
	  "AG" - All outGoing barring services (applicable only for
	      <mode>=0)
	  "AC" - All inComing barring services (applicable only for
	      <mode>=0)
	  "FD" - SIM fixed dialling memory feature (if PIN2
	      authentication has not been done during the current
	      session, PIN2 is required as **<passwd>**)
	  "PN" - network Personalisation
	  "PU" - network subset Personalisation
	  "PP" - service Provider Personalization
	      (refer 3GPP TS 22.022 [33])
	  "PC" - Corporate Personalization (refer 3GPP TS 22.022 [33])
	  "PF" - lock Phone to the very First inserted SIM/UICC card
	      (also referred in the present document as PH-FSIM)
	      (MT asks password when other than the first SIM/UICC
	      card is inserted)
	<mode> - defines the operation to be done on the facility
	  0 - unlock facility
	  1 - lock facility
	  2 - query status
	<passwd> - shall be the same as password specified for the
	    facility from the **DTE** user interface or with command
	    Change Password **+CPWD**
	<class> - sum of integers each representing a class of
	    information (default is 7)
	  1 - voice (telephony)
	  2 - data (refers to all bearer services)
	  4 - fax (facsimile services)(not supported by LTE)
	  8 - short message service
	 16 - data circuit sync
	 32 - data circuit async
	 64 - dedicated packet access
	128 - dedicated PAD access

+CLCK - Facility Lock/Unlock	

	Note: when **<mode>=2** and command successful, it returns: **+CLCK: <status>[,<class1>[<CR><LF>+CLCK: <status>,<class2> […]]**
	
	Where:
	<status> - the current status of the facility
	  0 - not active
	  1 - active
	<class*n*> - class of information of the facility

AT+CLCK=?	Test command reports all the facilities supported by the device.
Reference	3GPP TS 27.007
Example	*Querying such a facility returns an output on three rows, the first for voice, the second for data, the third for fax:* AT+CLCK =”AO”,2 +CLCK: <status>,1 +CLCK: <status>,2 +CLCK: <status>,4

3.4.4.2.5 *Change Facility Password - +CPWD*

+CPWD - Change Facility Password	

AT+CPWD= <fac>, <oldpwd>, <newpwd>	Execution command changes the password for the facility lock function defined by command Facility Lock **+CLCK**.
	
	Parameters:
	<fac> - facility
	  “SC” - SIM (PIN request)
	  “AB” - All barring services
	  “P2” - SIM PIN2
	    “AC” - All inComing barring services
	    “AG” - All outGoing barring services
	    “AI” – BAIC (Barr All Incoming Calls)
	    “AO” - BAOC (Barr All Outgoing Calls)
	    “IR” – BIC-Roam (Barr Incoming Calls when Roaming
	        outside the home country)
	    “OI” - BOIC (Barr Outgoing International Calls)
	    “OX” - BOIC-exHC (Barr Outgoing International Calls except to Home Country)
	
	<oldpwd> - string type, it shall be the same as password specified for the facility from the ME user interface or with command **+CPWD**.

+CPWD - Change Facility Password	
	<newpwd> - string type, it is the new password **Note:** parameter **<oldpwd>** is the old password while **<newpwd>** is the new one.
AT+CPWD=?	Test command returns a list of pairs **(<fac>,<pwdlength>)** which presents the available facilities and the maximum length of their password (**<pwdlength>**)
Example	at+cpwd=? +CPWD:("AB",4),("AC",4),("AG",4),("AI",4),("AO",4),("IR",4),("OI",4),("OX",4),("SC",8),("P2",8) OK
Reference	3GPP TS 27.007

3.4.4.2.6 *Preferred Operator List - +CPOL*

+CPOL - Preferred Operator List	

AT+CPOL=
[<index>]
[,<format>
[,<oper>
[,<GSM_AcT>,
<GSM_Compact
_AcT>,
<UTRAN_AcT>,
<EUTRAN_AcT>
]]]
	Execution command writes an entry in the SIM list of preferred operators.
	
	Parameters:
	<index> - integer type; the order number of operator in the SIM preferred operator list
	  1..*n*
	<format>
	  2 - numeric **<oper>**
	<oper> - string type
	
	<GSM_AcT> - GSM access technology
	  0 – access technology not selected
	  1 – access technology selected
	
	<GSM_Compact_AcT> - GSM compact access technology
	  0 – access technology not selected
	  1 – access technology selected
	
	<UTRA_AcT> - UTRA access technology
	  0 – access technology not selected
	  1 – access technology selected
	
	<E-UTRAN_AcT*n*> - E-UTRAN access technology:
	  0 access technology not selected
	  1 access technology selected
	
	Note: if **<index>** given but **<oper>** left out, the entry deleted. If **<oper>** given but **<index>** left out, **<oper>** put in the next free location. If only **<format>** given, the format of the **<oper>** in the read command changes. Currently **<GSM_Compact_AcT>** not supported but set value is acceptable.

AT+CPOL?	Read command returns all used entries from the SIM list of preferred operators.

+CPOL - Preferred Operator List	
AT+CPOL=?	Test command returns the whole **<index>** range supported by the SIM and the range for the parameter **<format>**
Reference	3GPP TS 27.007

3.4.4.2.7 *Selection of preferred PLMN list - +CPLS*

+CPLS - Selection of preferred PLMN list +CPLS	

AT+CPLS=<list>	Set command select one PLMN selector with Access Technology list in the SIM card or active application in the UICC (GSM or USIM), that is used by +CPOL command.
	
	Parameter:
	<list>:
	  0 - User controlled PLMN selector with Access Technology EFPLMNwAcT, if not found in the SIM/UICC then PLMN preferred list EFPLMNsel (this file is only available in SIM card or GSM application selected in UICC) (Default)
	  1 - Operator controlled PLMN selector with Access Technology EFOPLMNwAcT
	  2 - HPLMN selector with Access Technology EFHPLMNwAcT

AT+CPLS?	Read command returns the selected PLMN selector list from the SIM/USIM **+CPLS: <list>**
AT+CPLS=?	Test command returns the whole index range supported listsby the SIM./USIM
Reference	3GPP TS 27.007

3.4.4.3 Mobile Equipment Control

3.4.4.3.1 Set Phone Functionality - +CFUN

+CFUN - Set Phone Functionality	
AT+CFUN= [<fun> [,<rst>]]	Set command selects the level of functionality in the ME. **Parameters:** **<fun>** - is the power saving function mode 0 - minimum functionality, NON-CYCLIC SLEEP mode: in this mode, the AT interface is not accessible by UART. Consequently, once you have set **<fun>** level 0, do not send further characters. Otherwise these characters remain in the input buffer and may delay the output of an unsolicited result code. The first wake-up event stops power saving and takes the ME back to full functionality level **<fun>=1**. 1 - mobile full functionality with power saving mechanism (factory default) 2 - disable TX (Not support) 4 - disable both TX and RX 5 – same as 1 6 – mobile reboot Special modes , you can only see them only through the read command and you can't set those mode : 7 – Offline mode 8 – FTM

<rst> - reset flag 0 - do not reset the ME before setting it to <fun> Functionality level. 1- reset the ME before setting it to <fun> functionality level, This option works only with <fun> =1, with other it will return an error.

Note: **AT+CFUN=2** is not supported.

Note: Issuing AT+CFUN=4[,0] actually causes the module to perform either a network deregistration and a SIM deactivation.

Note: power saving mechanism, it reduces the power consumption during the idle time, thus allowing a longer standby time with a given battery capacity.

Note: To place the module in power saving mode, plug out the USB and set the DTR (RS232) line to OFF. Once in power saving, the CTS line switch to the OFF status to signal that the module is really in power saving condition. During the power saving condition, before sending any AT command on the serial line |

+CFUN - Set Phone Functionality	
	enabled the DTR line and wait for the CTS (RS232) line to go in ON status. while the DTR line is ON, the module will not return back in the power saving condition. **Note:** The power saving function does not affect the network behavior of the module. Even during the power save condition, the module remains registered on the network and reachable for incoming calls or SMS. If a call incomes during the power save, then the module will wake up and proceed normally with the unsolicited incoming call code
AT+CFUN?	Read command reports the current setting of **\<fun>**.
AT+CFUN=?	Test command returns the list of supported values for **\<fun>** and **\<rst>**.
Reference	3GPP TS 27.007

3.4.4.3.2 *Enter PIN - +CPIN*

+CPIN - Enter PIN	
AT+CPIN= **\<pin>** **[,\<newpin>]**	Set command sends the device a necessary password before it can be operated (SIM PIN, SIM PUK, PH-SIM PIN, etc.). If the PIN required is SIM PUK or SIM PUK2, the **\<newpin>** is required. This second pin, **\<newpin>** will replace the old pin in the SIM. The command may be used to change the SIM PIN by sending it with both parameters **\<pin>** and **\<newpin>** when PIN request is pending; if no PIN request is pending the command will return an error code and to change the PIN the command **+CPWD** must be used instead. **Parameters:** **\<pin>** - string type value **\<newpin>** - string type value. To check the status of the PIN request use the command **AT+CPIN?** **Note:** If all parameters omitted then the behaviour of Set command is the same as Read command.

AT+CPIN?	Read command reports the PIN/PUK/PUK2 request status of the device in the form:
	+CPIN: \<code>
	where:
	\<code> - PIN/PUK/PUK2 request status code
	   READY - ME is not pending for any password
	   SIM PIN - ME is waiting SIM PIN to be given
	   SIM PUK - ME is waiting SIM PUK to be given
	   PH-SIM PIN - ME is waiting phone-to-SIM card password to be given PH-FSIM PIN - ME is waiting phone-to-very first SIM card password to be given PH-FSIM PUK - ME is waiting phone-to-very first SIM card unblocking password to be given SIM PIN2 - ME is waiting SIM PIN2 to be given; this **\<code>** is returned only when the last executed command resulted in PIN2 authentication failure (i.e. **+CME ERROR: 17**)
	   SIM PUK2 - ME is waiting SIM PUK2 to be intered. this

+CPIN - Enter PIN	
	<code> is returned only when the last executed command resulted in PUK2 authentication failure (i.e. **+CME ERROR: 18**) **PH-NET PIN** - ME is waiting network personalization password to be given **PH-NET PUK** - ME is waiting network personalization unblocking password to be given **PH-NETSUB PIN** - ME is waiting network subset personalization password to be given **PH-NETSUB PUK** - ME is waiting network subset personalization unblocking password to be given **PH-SP PIN** - ME is waiting service provider personalization password to be given **PH-SP PUK** - ME is waiting service provider personalization unblocking password to be given **PH-CORP PIN** - ME is waiting corporate personalization password to be given **PH-CORP PUK** - ME is waiting corporate personalization unblocking password to be given **Note:** Pin pending status at startup depends on PIN facility setting, to change or query the default power up setting use the command **AT+CLCK=SC,<mode>,<pin>**
Example	AT+CMEE=1 OK AT+CPIN? +CME ERROR: 10 *error: you have to insert the SIM* AT+CPIN? +CPIN: READY *you inserted the SIM and device is not waiting for PIN to be given* OK
Note	What follows is a list of the commands which are accepted when ME is pending SIM PIN or SIM PUK

A	&K		+CPIN
D		+GCAP	+CSQ
H	&P	+GCI	+CIND
O		+IPR	+CMER
E	&V	+IFC	+CCLK
I	&W	+ILRR	+CALA
L		+ICF	
M		+CRSM	
P	+DS		
Q		+DR	
	+CGMI		

+CPIN - Enter PIN					
		T	\Q	+CGMM	
		V	+CGMR	+CLAC	
		X	\V	+GMI	+CMEE
		Z	#CGMI	+GMM	+CGREG
		&C	#CGMM	+GMR	+CBC
		&D	#CGMR	+CGSN	+CSDH
		&F	#CGSN	+GSN	+CNMI
		+COPS	+CHUP	+CRC	
		+CLIP	#SHDN		
		+CPAS	+CR	+CREG	
		+CFUN			
	All the above commands, but **+CSDH** and **+CNMI**, can be issued even if ME is waiting for phone-To-SIM card password to be given				
Reference	3GPP TS 27.007				

3.4.4.3.3 *Signal Quality - +CSQ*

+CSQ - Signal Quality	

AT+CSQ	Execution command reports received signal quality indicators in the form:
	
	+CSQ: <rssi>,<sq>
	Where:
	<rssi> - received signal strength indication (3GPP)
	  0: (-113) dBm or less
	  1: (-111) dBm
	  2..30: (-109) dBm..(-53)dBm / 2 dBm per step
	  31: (-51) dBm or greater
	    99: Not known or not detectable
	
	<rssi> - received signal strength indication (TDSCDMA)
	    100: (-116) dBm or less
	    101: (-115) dBm
	    102…191 (-114…-26) dBm
	    191: (-25) dBm or greater
	    199: Not known or not detectable
	
	<sq> - signal quality – as mentioned below:
	2G (GSM) – RXQUAL [bit error rate (in percent)]:
	  0: less than 0.2%
	  1: 0.2% to 0.4%
	  2: 0.4% to 0.8%
	  3: 0.8% to 1.6%

+CSQ - Signal Quality	

	4: 1.6% to 3.2%
	5: 3.2% to 6.4%
	6: 6.4% to 12.8%
	7: more than 12.8%
	  99 - not known or not detectable
	
	3G (UTRAN) – ECIO [in dBm]:
	 0: (-1) to (0)
	 1: (-5) to (-2)
	 2: (-8) to (-6)
	 3: (-11) to (-9)
	  4: (-15) to (-12)
	  5: (-18) to (-16)
	  6: (-22) to (-19)
	  7: (-24) to (-23)
	 99 - not known or not detectable
	
	4G (LTE) – RSRQ [in dBm]:
	 0: (-4) to (-3)
	 1: (-6) to (-5)
	 2: (-8) to (-7)
	 3: (-10) to (-9)
	  4: (-13) to (-11)
	  5: (-15) to (-14)
	  6: (-17) to (-16)
	  7: (-19) to (-18)
	 99 - not known or not detectable

AT+CSQ=?	Test command returns the supported range of values of the parameters **\<rssi\>** and **\<ber\>**. **Note:** although **+CSQ** is an execution command without parameters, 3GPP TS 27.007 requires the Test command to be defined.
Reference	3GPP TS 27.007

3.4.4.3.4 *Extended Signal Quality - +CESQ*

+CESQ – Extended Signal Quality	
AT+CESQ	Execution command reports received signal quality parameters in the form: **+CESQ: \<rxlev\>,\<ber\>,\<rscp\>,\<ecno\>,\<rsrq\>,\<rsrp\>** Where

+CESQ – Extended Signal Quality	
	< rxlev > - received received signal strength level (see 3GPP TS 45.008 subclause 8.1.4).
	0 - rssi < -110 dBm
	1 - -110 dBm ≤ rssi < -109 dBm
	2 - -109 dBm ≤ rssi < -108 dBm
	...
	61 - -50 dBm ≤ rssi < -49 dBm
	62 - -49 dBm ≤ rssi < -48 dBm
	63 - -48 dBm ≤ rssi
	99 - not known or not detectable or if the current serving cell is not a GERAN cell
	<ber> - bit error rate (in percent)
	0...7 - as RXQUAL values in the table in 3GPP TS 45.008 subclause 8.2.4
	99 - not known or not detectable or if the current serving cell is not a GERAN cell.
	<rscp> - received signal code power (see 3GPP TS 25.133 subclause 9.1.1.3 and 3GPP TS 25.123 subclause 9.1.1.1.3).
	0 - rscp < -120 dBm
	1 - -120 dBm ≤ rscp < -119 dBm
	2 - -119 dBm ≤ rscp < -118 dBm
	...
	94 - -27 dBm ≤ rscp < -26 dBm
	95 - -26 dBm ≤ rscp < -25 dBm
	96 - 25 dBm ≤ rscp
	255 - not known or not detectable or if the current serving cell is not a UTRA cell
	<ecno> - ratio of the received energy per PN chip to the total received power spectral density (see 3GPP TS 25.133 subclause).
	0 - Ec/Io < -24 dB
	1 - -24 dB ≤ Ec/Io < -23.5 dB
	2 - -23.5 dB ≤ Ec/Io < -23 dB
	...
	47 - -1 dB ≤ Ec/Io < -0.5 dB
	48 - -0.5 dB ≤ Ec/Io < 0 dB
	49 – 0 dB ≤ Ec/Io
	255 - not known or not detectable detectable or if the current serving cell is not a UTRA cell
	<rsrq> - reference signal received quality (see 3GPP TS 36.133 subclause 9.1.7).
	0 - rsrq < -19.5 dB
	1 - -19.5 dB ≤ rsrq < -19 dB
	2-19 dB ≤ rsrq < -18.5 dB
	...
	32 - -4 dB ≤ rsrq < -3.5 dB
	33 - -3.5 dB ≤ rsrq < -3 dB

+CESQ – Extended Signal Quality	
	34 - -3 dB ≤ rsrq 255 - not known or not detectable detectable or if the current serving cell is not a EUTRA cell **\<rsrp\>** - reference signal received power (see 3GPP TS 36.133 subclause 9.1.4). 0 - rsrp < -140 dBm 1 - -140 dBm ≤ rsrp < -139 dBm 2 - -139 dBm ≤ rsrp < -138 dBm ... 95 - -46 dBm ≤ rsrp < -45 dBm 96 - -45 dBm ≤ rsrp < -44 dBm 97 - -44 dBm ≤ rsrp 255 not known or not detectable detectable or if the current serving cell is not a EUTRA cell
AT+CESQ=?	Test command returns the supported range of values of the parameters **\<rxlev\>, \<ber\>, \<rscp\>, \<ecno\>, \<rsrq\>, \<rsrp\>.**
Reference	3GPP TS 27.007

3.4.4.3.5 *Mobile Equipment Event Reporting - +CMER*

+CMER - Mobile Equipment Event Reporting	
AT+CMER= [\<mode\> [,\<keyp\> [,\<disp\> [,\<ind\> [,\<bfr\>]]]]]	Set command enables/disables sending of unsolicited result codes from TA to TE in the case of indicator state changes (n.b.: sending of URCs in the case of key pressings or display changes are currently not implemented). **Parameters:** **\<mode\>** - controls the processing of unsolicited result codes 0 - discard **+CIEV** Unsolicited Result Codes. 1 - discard **+CIEV** Unsolicited Result Codes when TA-TE link is reserved (e.g. on-line data mode); otherwise forward them directly to the TE. 2 - buffer **+CIEV** Unsolicited Result Codes in the TA when TA-TE link is reserved (e.g. on-line data mode) and flush them to the TE after reservation; otherwise forward them directly to the TE. 3 - forward **+CIEV** Unsolicited Result Codes directly to the TE; when TA is in on-line data mode each **+CIEV** URC is replaced with a **Break** (100 ms), and is stored in a buffer; onche the ME goes into command mode (after **+++** was entered), all URCs stored in the buffer will be output. **\<keyp\>** - keypad event reporting 0 - no keypad event reporting **\<disp\>** - display event reporting 0 - no display event reporting **\<ind\>** - indicator event reporting 0 - no indicator event reporting 2 - indicator event reporting **\<bfr\>** - TA buffer clearing

+CMER - Mobile Equipment Event Reporting	
	0 - TA buffer of unsolicited result codes is cleared when <mode> 1..3 is entered 1 - TA buffer of unsolicited result codes is flushed to the TE when <mode> 1...3 is entered (OK response shall be given before flushing the codes) **Note:** After AT+CMER has been switched on with e.g. AT+CMER=2,0,0,2 command (i.e. <bfr> is 0), URCs for all registered indicators will be issued only first time, if previous <mode> was 0, for backward compatibility. Values shown by the indicators will be current indicators values, not buffered ones. Subsequent AT+CMER commands with <mode> different from 0 and <bfr> equal to 0 will not flush the codes, even if <mode> was set again to 0 before. To flush the codes, <bfr> must be set to 1. Although it is possible to issue the command when SIM PIN is pending, it will answer ERROR if "message" or "smsfull" indicators are enabled in AT+CIND, because with pending PIN it is not possible to give a correct indication about SMS status. To issue the command when SIM PIN is pending you have to disable "message" and "smsfull" indicators in AT+CIND first.
AT+CMER?	Read command returns the current setting of parameters, in the format: **+CMER: <mode>,<keyp>,<disp>,<ind>,<bfr>**
AT+CMER=?	Test command returns the range of supported values for parameters **<mode>, <keyp>, <disp>, <ind>, <bfr>**, in the format: **+CMER: (list of supported <mode>s),(list of supported <keyp>s),(list of supported <disp>s),(list of supported <ind>s),(list of supported <bfr>s)**
Reference	3GPP TS 27.007

3.4.4.3.6 *Clock Management - +CCLK*

+CCLK - Clock Management	
AT+CCLK= **<time>**	Set command sets the real-time clock of the **ME**. **Parameter:** **<time>** - current time as quoted string in the format: **"yy/MM/dd,hh:mm:ss±zz"** **yy** - year (two last digits are mandatory). range is (00..99) **MM** - month (two last digits are mandatory). range is (01..12) **dd** - day (two last digits are mandatory). available ranges are: (01..28) (01..29) (01..30) (01..31) **hh** - hour (two last digits are mandatory).

+CCLK - Clock Management	
	range is (00..23)
	mm - minute (two last digits are mandatory).
	range is (00..59)
	ss - Seconds (two last digits are mandatory).
	range is (00..59)
	±zz - time zone (indicates the difference, expressed in quarter of an hour, between the local time and GMT; two last digits are mandatory), range is -96..+96
AT+CCLK?	Read command returns the current setting of the real-time clock, in the format **<time>**.
	Note: the three last characters of **<time>**, i.e. the time zone information, are returned by **+CCLK?** only if the **#NITZ** URC *'extended'* format has been enabled (see **#NITZ**).
AT+CCLK=?	Test command returns the **OK** result code.
Example	AT+CCLK="02/09/07,22:30:00+00" OK AT+CCLK? +CCLK: 02/09/07,22:30:25 OK
Reference	3GPP TS 27.007

3.4.4.3.7 *Restricted SIM Access - +CRSM*

+CRSM - Restricted SIM Access	

AT+CRSM=
<command>
[,<fileid>
[,<P1>,<P2>,
<P3>[,<data>]]]
	Execution command transmits to the **ME** the SIM **<command>** and its required parameters. **ME** handles internally all **SIM-ME** interface locking and file selection routines. As response to the command, **ME** sends the actual SIM information parameters and response data.
	
	Parameters:
	<command> - command passed on by the **ME** to the SIM
	  176 - READ BINARY
	  178 - READ RECORD
	  192 - GET RESPONSE
	  214 - UPDATE BINARY
	  220 - UPDATE RECORD
	  242 - STATUS
	
	<Fileid> - identifier of an elementary data file on SIM.
	Mandatory for every command except STATUS.
	
	<P1>,**<P2>**,**<P3>** - parameter passed on by the **ME** to the SIM; they are mandatory for every command except GET RESPONSE and STATUS 0..255

+CRSM - Restricted SIM Access	
	<Data> - information to be read/written to the SIM (hexadecimal character format). The response of the command is in the format: **+CRSM: <sw1>,<sw2>[,<response>]** **where:** **<sw1>,<sw2>** - information from the SIM about the execution of the actual command either on successful or failed execution. **<response>** - on a successful completion of the command previously issued it returns the requested data (hexadecimal character format). It's not returned after a successful UPDATE BINARY or UPDATE RECORD command. **Note:** this command requires PIN authentication. However commands READ BINARY and READ RECORD can be issued before PIN authentication and if the SIM is blocked (after three failed PIN authentication attempts) to access the contents of the Elementary Files. **Note:** use only decimal numbers for parameters **<command>**, **<fileid>**, **<P1>**, **<P2>** and **<P3>**.
AT+CRSM=?	Test command returns the **OK** result code
Example	Read binary, ICCID(2FE2) AT+CRSM=176,12258,0,0,10 +CRSM: 144,0,9828507002001107686F4 OK Read record, ADN(6F3A) AT+CRSM=178,28474,1,4,40 +CRSM: 144,0,42434A554EFFFFFFFFFFFFFFFFFFFFFF0681105678 9282FFFFFFFFFFFFFF OK Update Binary, KcGPRS(6F52) AT+CRSM=214,28539,0,0,8,C69018C7958C87 +CRSM: 144,0 OK Update Record, ADN(6F3A)

+CRSM - Restricted SIM Access	
	AT+CRSM=220,28474,9,4,30,657469FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF FFFFFFFFFFFFFFFFFFFFFFFF +CRSM: 144,0 OK Status, FPLMN(6F7B) AT+CRSM=242,28539 +CRSM: 144,0,623C820238218410A0000000871002FFFFFFFF8904 0300FFA5118001318103010A3282011E8304000030E08A01058B032F0609C609 9001C0830101830181
Reference	3GPP TS 27.007, 3GPP TS 11.11/51.011

3.4.4.3.8 *Available AT Commands - +CLAC*

+CLAC - Available AT Commands	
AT+CLAC	Execution command causes the ME to return the AT commands that are available for the user, in the following format: **<AT cmd1>[<CR><LF><AT cmd2>[…]]** **where:** **<AT cmd*n*>** - defines the AT command including the prefix **AT**
AT+CLAC=?	Test command returns the **OK** result code
Reference	3GPP TS 27.007

3.4.4.4 Mobile Equipment Errors

3.4.4.4.1 *Report Mobile Equipment Error - +CMEE*

+CMEE - Report Mobile Equipment Error	
AT+CMEE=[<n>]	Set command enables/disables the report of result code: **+CME ERROR: <err>** As an indication of an error relating to the **+Cxxx** commands issued. When enabled, device related errors cause the **+CME ERROR <err>** result code instead of the default **ERROR** result code. **ERROR** anyway returned normally when the error message related to syntax, invalid parameters, or **DTE** functionality.

+CMEE - Report Mobile Equipment Error	

	Parameter:
	\<n\> - enable flag
	  0 - disable **+CME ERROR:\<err\>** reports, use only **ERROR** report.
	  1 - enable **+CME ERROR:\<err\>** reports, with **\<err\>** in numeric format
	  2 - enable **+CME ERROR: \<err\>** reports, with **\<err\>** in verbose format
	Note: default value after powerup is "0"

AT+CMEE?	Read command returns the current value of subparameter **\<n\>**: **+CMEE: \<n\>**
AT+CMEE=?	Test command returns the range of values for subparameter **\<n\>**
Note	**+CMEE** has no effect on the final result code **+CMS**
Reference	3GPP TS 27.007

3.4.4.5 Commands for GPRS

3.4.4.5.1 *GPRS Attach or Detach - +CGATT*

+CGATT - GPRS Attach Or Detach	

AT+CGATT= [<state>]	Execution command used to attach the terminal to, or detach the terminal from, the GPRS service depending on the parameter **<state>**.
	
	Parameter:
	<state> - state of GPRS attachment
	  0 - detached
	  1 - attached

AT+CGATT?	Read command returns the current GPRS service state.
AT+CGATT=?	Test command requests information on the supported GPRS service states.
Example	AT+CGATT? +CGATT: 0 OK AT+CGATT=? +CGATT: (0,1) OK AT+CGATT=1 OK
Reference	3GPP TS 27.007

3.4.4.5.2 *GPRS Event Reporting - +CGEREP*

+CGEREP - GPRS Event Reporting	
AT+CGEREP= [<mode>[,<bfr>]]	Set command enables or disables sending of unsolicited result codes **+CGEV: XXX** (see below) from **TA** to **TE** in the case of certain events occurring in the **TA** or the network. **Parameters**: **<mode>** - controls the processing of **URCs** specified with this command 0 - Buffer unsolicited result codes in the **TA**. If **TA** result code buffer is full, the oldest one can be discarded. No codes are forwarded to the **TE**.

+CGEREP - GPRS Event Reporting	
	1 - Discard unsolicited result codes when **TA-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 **TA** when **TA-TE** link is reserved (e.g. in on-line data mode) and flush them to the **TE** when **TA-TE** link becomes available; otherwise forward them directly to the **TE**. **\<bfr\>** - controls the effect on buffered codes when **\<mode\> 1** or **2** is entered: 0 - **TA** buffer of unsolicited result codes defined within this command is cleared when **\<mode\>=1** or **2** is entered. 1 - **TA** 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) **Unsolicited Result Codes** The following unsolicited result codes and the corresponding events are defined: **+CGEV: REJECT \<PDP_type\>, \<PDP_addr\>** A network request for PDP context activation occurred when the **TA** was unable to report it to the **TE** with a +**CRING** unsolicited result code and was automatically rejected **+CGEV: NW REACT \<PDP_type\>, \<PDP_addr\>, [\<cid\>]** The network has requested a context reactivation. The \<cid\> that was used to reactivate the context is provided if known to **TA** **+CGEV: NW DEACT \<PDP_type\>, \<PDP_addr\>, [\<cid\>]** The network has forced a context deactivation. The \<cid\> that was used to activate the context is provided if known to **TA** **+CGEV: ME DEACT \<PDP_type\>, \<PDP_addr\>, [\<cid\>]** The mobile equipment has forced a context deactivation. The \<cid\> that was used to activate the context is provided if known to **TA** **+CGEV: NW DETACH** The network has forced a GPRS detach. This implies that all active contexts have been deactivated. These are not reported separately **+CGEV: ME DETACH** The mobile equipment has forced a GPRS detach. This implies that all active contexts have been deactivated. These are not reported separately **+CGEV: ME CLASS \<class\>** The mobile equipment has forced a change of MS class. The highest available class is reported (see **+CGCLASS**)
AT+CGEREP?	Read command returns the current \<mode\> and \<bfr\> settings, in the format:

+CGEREP - GPRS Event Reporting	
	+CGEREP: <mode>,<bfr>
AT+CGACT=?	Test command reports the supported range of values for the +CGEREP command parameters.
Reference	3GPP TS 27.007

3.4.4.5.3 *GPRS Network Registration Status - +CGREG*

+CGREG - GPRS Network Registration Status	

AT+CGREG=[<n>]	Set command controls the presentation of an unsolicited result code
	+CGREG: (see format below).
	
	Parameter:
	<n> - result code presentation mode
	  0 - disable network registration unsolicited result code
	  1 - enable network registration unsolicited result code; if there is a change in the terminal GPRS network registration status, it is issued the unsolicited result code:
	
	  +CGREG: <stat>
	
	  where:
	  <stat> - registration status
	    0 - not registered, terminal is not currently searching a new operator to register to
	    1 - registered, home network
	    2 - not registered, but terminal is currently searching a new operator to register to
	    3 - registration denied
	    4 - unknown
	    5 - registered, roaming
	  2 - enable network registration and location information unsolicited result code; if there is a change of the network cell, it is issued the unsolicited result code:
	
	  +CGREG: <stat>[,<lac>,<ci>[,<AcT>,<rac>]]
	
	  where:
	  <stat> - registration status (see above for values)
	  <lac> - Local Area Code (when <AcT> indicates value 0 to 6) or tracking area code (when <AcT> indicates value 7)
	  <ci> - cell ID in hexadecimal format.
	  <AcT>: access technology of the registered network:
	    0 GSM
	    2 UTRAN
	    3 GSM w/EGPRS

+CGREG - GPRS Network Registration Status	

	4 UTRAN w/HSDPA
	    5 UTRAN w/HSUPA
	    6 UTRAN w/HSDPA and HSUPA
	
	<rac>: string type; one byte routing area code in hexadecimal format
	
	 4 - enable network registration and location information unsolicited result code:
	
	+CGREG: <stat>[,<lac>,<ci>[,<AcT>,<rac>]]
	
	where:
	
	4 - enable network registration and location information unsolicited result code:
	+CGREG: <stat>[,[<lac>],[<ci>],[<AcT>],[<rac>]
	[,,[,[<Active-Time>],[<Periodic-RAU>],[<GPRS-READY-timer>]]]]
	
	where:
	
	<Active-Time> - string type; one byte in an 8 bit format. Indicates the Active Time value (T3324) allocated to the UE in GERAN/UTRAN. The Active Time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes). For the coding and the value range, see the GPRS Timer 2 IE in 3GPP TS 24.008.
	
	<Periodic-RAU> - string type; one byte in an 8 bit format. Indicates the extended periodic RAU value (T3312) allocated to the UE in GERAN/UTRAN. The extended periodic RAU value s coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see the GPRS Timer 3 IE in 3GPP TS 24.008.
	
	<GPRS-READY-timer> - string type; one byte in an 8 bit format. Indicates the GPRS READY timer value (T3314) allocated to the UE in GERAN/UTRAN. The GPRS READY timer value is coded as one byte (octet 2) of the GPRS Timer information element coded as bit format (e.g. "01000011" equals 3 decihours or 18 minutes). For the coding and the value range, see the GPRS Timer IE in 3GPP TS 24.008.
	
	Note: **<lac>, <Ci>, <AcT>** and **<rac>** are reported only if **<mode>=2** or **4** and the mobile is registered on some network cell.

AT+CGREG?	Read command returns the status of result code presentation mode **<n>** and the integer **<stat>** which shows whether the network has currently indicated the registration of the terminal in the format: **+CGREG: <n>,<stat>[,[<lac>],[<ci>],[<AcT>],[<rac>]** **[,,[,[<Active-Time>],[<Periodic-RAU>],[<GPRS-READY-timer>]]]]**

+CGREG - GPRS Network Registration Status	
	Note: **\<lac\>, \<Ci\>, \<AcT\>** and **\<rac\>** are reported only if **\<mode\>=2** or **4** and the mobile is registered on some network cell.
AT+CGREG=?	Test command returns supported values for parameter **\<n\>**
Reference	3GPP TS 27.007

3.4.4.5.4 *Printing IP Address Format - +CGPIAF*

+CGPIAF - Printing IP Address Format	

AT+CGPIAF= [\<IPv6_AddressFormat\> [,\<IPv6_SubnetNotation\> [,\<IPv6_leadingZeros\> [,\<IPv6_compressZeros\>]]]]	Set command decides what the format to print IPv6 address parameter.
	
	Parameters:
	\<IPv6_AddressFormat\> - decides the IPv6 address format. Relevant for all AT command parameters, that can hold an IPv6 address.
	  0 – Use IPv4-like dot-notation. IP addresses, and subnetwork mask if applicable, are dot-separated.
	  1 – Use IPv6-like colon-notation. IP address, and subnetwork mask if applicable and when given explicitly, are separated by a space.
	
	\<IPv6_SubnetNotation\> - decides the subnet-notation for \<remote address and subnet mask\> Setting does not apply if IPv6 address format **\<IPv6_AddressFormat\>** = 0.
	  0 – Both IP address, and subnet mask are started explicitly, separated by a space.
	  1 – The printout format is applying /(forward slash) subnet-prefix Classless Inter-Domain Routing (CIDR) notation.
	
	\<IPv6_LeadingZeros\> - decides whether leading zeros are omitted or not. Setting does not apply if IPv6 address format **\<IPv6_AddressFormat\>** = 0.
	  0 – Leading zeros are omitted.
	  1 – Leading zeros are included.
	
	\<IPv6_CompressZeros\> - decides whether 1-n instances of 16-bit- zero-values are replaced by only "::". This applies only once. Setting does not apply if IPv6 address format **\<IPv6_AddressFormat\>** = 0.
	  0 – No zero compression.
	  1 – Use zero compression.

AT+CGPIAF?	Read command returns the current parameter setting.
AT+CGPIAF=?	Test command returns values supported as compound parameter setting.
Example	AT+CGPIAF=0,0,0,0 OK AT#SGACT=1,1 #SGACT: 252.1.171.171.205.205.239.224.0.0.0.0.0.0.0.1 OK

+CGPIAF - Printing IP Address Format	
	at+CGPIAF=1,0,0,0 OK AT#SGACT=1,1 #SGACT: FC01:ABAB:CDCD:EFE0:0:0:0:1 OK
Reference	3GPP TS 27.007

3.4.4.5.5 *Define PDP Context - +CGDCONT*

+CGDCONT - Define PDP Context	

AT+CGDCONT=
[<cid>
[,<PDP_type>
[,<APN>
[,<PDP_addr>
[,<d_comp>
[,<h_comp>
[,<pd1>
[,…[,pdN]]]]]]]]]
	Set command specifies PDP context parameter values for a PDP context identified by the (local) context identification parameter <cid>.
	
	Parameters:
	<cid> - (PDP Context Identifier) numeric parameter which specifies a particular PDP context definition.
	  1..*max* - where the value of *max* is returned by the Test command.
	<PDP_type> - (Packet Data Protocol type) a string parameter which specifies the type of packet data protocol.
	  "IP" - Internet Protocol
	  "PPP" - Point to Point Protocol
	  "IPV6" - Internet Protocol, Version 6
	  "IPV4V6" - Virtual <PDP_type> introduced to handle dual IP stack UE capability.
	<APN> - (Access Point Name) a string parameter which is a logical name that is used to select the GGSN or the external packet data network. If the value is null or omitted, then the subscription value will be requested.
	<PDP_addr> - a string parameter that identifies the terminal in the address space applicable to the PDP. The allocated address may be read using the **+CGPADDR** command.
	<d_comp> - numeric parameter that controls PDP data compression.
	  0 - off (default if value is omitted)
	  1 – on
	  2 - V.42bis
	<h_comp> - numeric parameter that controls PDP header compression.
	  0 - off (default if value is omitted)
	  1 – on
	  2 - RFC1144 (applicable for SNDCP only)
	  3- RFC2507
	  4- RFC3095 (applicable for PDCP only)

+CGDCONT - Define PDP Context	
	<pd1>, …, **<pdN>** - zero to N string parameters whose meanings are specific to the **<PDP_type>** **Note:** a special form of the Set command, **+CGDCONT=<cid>**, causes the values for context number **<cid>** to become undefined.
AT+CGDCONT?	Read command returns the current settings for each defined context in the format: **+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>, <h_comp>[,<pd1>[,…[,pdN]]][<CR><LF>+CGDCONT: <cid>, <PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp> [,<pd1>[,…[,pdN]]][…]]**
AT+CGDCONT=?	Test command returns values supported as a compound value
Example	AT+CGDCONT=1,"IP",*"APN"*,"10.10.10.10",0,0 OK AT+CGDCONT? +CGDCONT: 1,"IP","*APN*","10.10.10.10",0,0 OK AT+CGDCONT=? +CGDCONT: (1-16),"IP",,,(0-2),(0-4) +CGDCONT: (1-16),"PPP",,,(0-2),(0-4) +CGDCONT: (1-16),"IPV6",,,(0-2),(0-4) OK
Reference	3GPP TS 27.007

3.4.4.5.6 *Define Secondary PDP Context - +CGDSCONT*

+CGDSCONT – Define Secondary PDP Context	
AT+CGDSCONT= [<cid> ,<p_cid> [,<d_comp> [,<h_comp>]]]	Set command specifies PDP context parameter values for a Secondary PDP context identified by the (local) context Identification parameter, <cid>. The number of PDP contexts that may be in a defined state at the same time is given by the range returned by the test command. In EPS the command is used to define traffic flows. A special form of the set command, +CGDSCONT= <cid> causes the values for context number <cid> to become undefined. **Parameters:** <**cid**>: a numeric parameter which specifies a particular PDP context definition (see +**CGDCONT** command)

+CGDSCONT – Define Secondary PDP Context	
	<p_cid>: a numeric parameter which specifies a particular PDP context definition which has been specified by use of the +CGDCONT command. The parameter is local to the TE-MT interface. <d_comp>: a numeric parameter that controls PDP data compression (applicable for SNDCPonly) (refer 3GPP TS 44.065 [61]) 0 - off (default if value is omitted) 1 - on (manufacturer preferred compression) 2 - V.42bis 3 - V.44 Other values are reserved. <h_comp>: a numeric parameter that controls PDP header compression (refer 3GPP TS 44.065 [61] and 3GPP TS 25.323 [62]) 0 - off (default if value is omitted) 1 - on (manufacturer preferred compression) 2 - RFC1144 (applicable for SNDCP only) 3 - RFC2507 4 - RFC3095 (applicable for PDCP only) Other values are reserved.
AT+CGDSCONT?	The read command returns the current settings for each defined context in the format: **AT+CGDSCONT:** **<cid>,<p_cid>,<d_comp>,<h_comp>[<CR><LF>+CGDSCONT:<cid>,<p_cid>,<d_comp>,<h_comp> [...]]**
AT+CGDSCONT=?	Test command returns the supported range of values of parameters

3.4.4.5.7 *PDP Context Activate Or Deactivate - +CGACT*

+CGACT - PDP Context Activate Or Deactivate	

AT+CGACT= [<state>[,<cid> [,<cid>[,…]]]]	Execution command is used to activate or deactivate the specified PDP context(s)
	
	 Parameters:
	 <state> - indicates the state of PDP context activation
	   0 - deactivated
	   1 - activated
	 <cid> - a numeric parameter which specifies a particular PDP context definition (see **+CGDCONT** command)
	
	 Note: at least three **<cid>**s can be activated at the same time.

+CGACT - PDP Context Activate Or Deactivate	
	Note: if no **<cid>**s are specified, the activation form of the command activates at least the first three defined contexts. The deactivation form deactivates all the active contexts.
AT+CGACT?	Read command returns the current activation state for all the defined PDP contexts in the format: **+CGACT: <cid>,<state>[<CR><LF>+CGACT: <cid>,<state>[…]]**
AT+CGACT=?	Test command reports information on the supported PDP context activation states parameters in the format: **+CGACT: (0,1)**
Example	AT+CGACT=1,1 OK AT+CGACT? +CGACT: 1,1 OK
Reference	3GPP TS 27.007

3.4.4.5.8 *PDP Context Modify - +CGCMOD*

+CGCMOD action command syntax	
AT+CGCMOD= **[<cid>** **[,<cid>** **[,…]]]**	It has no effect and is included only for backward compatibility with landline modems **Possible Response(s):** OK ERROR The execution command used to modify the specified PDP context (s) with repect to QoS profiles and TFTs. After command has completed, the MT returns to V.250 online data state. If the requested modification for any specified context cannot be achieved, an ERROR or +CME ERROR response is returned. Extended error responses are enabled by the +CMEE command. For EPS, the modification request for an EPS bearer resource will be answered by the network by an EPS bearer Modification request. The request must be accepted by the MT before the PDP context effectively changed. If no <cid>s are specified the activation form of the command modifies all active contexts. The test command returns a list of <cid>s associated with active contexts. **Defined Values** **<cid>:** a numeric parameter which specifies a particular PDP context definition (see the +CGDCONT and +CGDSCONT commands).

+CGCMOD action command syntax	
AT+CGCMOD=?	+CGCMOD: (list of <cid>s associated with active contexts)

3.4.4.5.9 *Show PDP Address - +CGPADDR*

+CGPADDR - Show PDP Address	
AT+CGPADDR= [<cid>[,<cid> [,…]]]	Execution command returns a list of PDP addresses for the specified context identifiers in the format: **+CGPADDR: <cid>,<PDP_addr>[<CR><LF>+CGPADDR: <cid>, <PDP_addr>[…]]** **Parameters:** **<cid>** - a numeric parameter which specifies a particular PDP context definition (see **+CGDCONT** command). If no **<cid>** specified, the addresses for all defined contexts are returned. **<PDP_addr>** - a string that identifies the terminal in an address space applicable to the PDP. The address may be static or dynamic. For a static address, it will be the one set by the **+CGDCONT** command when the context was defined. For a dynamic address it will be the one assigned during the last PDP context activation that used the context definition referred to by **<cid>**;**<PDP_addr>** is omitted if none is available
AT+CGPADDR=?	Test command returns a list of defined **<cid>**s.
Example	 OK AT+CGPADDR=1 +CGPADDR: 1,"xxx.yyy.zzz.www" OK AT+CGPADDR=? +CGPADDR: (1) OK
Reference	3GPP TS 27.007

3.4.4.5.10 *Set mode of operation for EPS - +CEMODE*

+CEMODE – Set mode of operation for EPS.	

AT+CEMODE=[<mode>]	Set command configures the mode of operation for EPS.
	
	Parameter:
	<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
	  3 : PS mode 1 of operation
	
	Note: the default value depends on product and the support of VoLTE.
	
	Note: the definition for UE modes of operation can be found in 3GPP TS 24.30.

AT+CEMODE?	Read command returns the currently configured values, in the format: **+CEMODE**: **< mode >** **Note:** The read command will return right values after set command, but effectively the mode of operation changes after power cycle.
AT+CEMODE=?	Test command returns the supported range of values of parameters **< mode>.**
Note	
Example	AT+CEMODE=1 OK AT+CEMODE? +CEMODE: 1 OK

3.4.5 3GPP TS 27.005 AT Commands for SMS and CBS

3.4.5.1 General Configuration

3.4.5.1.1 *Select Message Service - +CSMS*

+CSMS - Select Message Service	

AT+CSMS=
<service>
	Set command selects messaging service **<service>**. It returns the types of messages supported by the **ME**:
	
	Parameter:
	<service>
	 0 - The syntax of SMS AT commands is compatible with 3GPP TS 27.005 Phase 2
	     version 4.7.0 (factory default)
	 1 - The syntax of SMS AT commands is compatible with 3GPP TS 27.005 Phase 2+
	     version.
	Set command returns the types of messages supported by the **ME**:
	
	+CSMS: <mt>,<mo>,<bm>
	
	where:
	<mt> - mobile terminated messages support
	 0 - type not supported
	 1 - type supported
	<mo> - mobile originated messages support
	 0 - type not supported
	 1 - type supported
	<bm> - broadcast type messages support
	 0 - type not supported
	 1 - type supported

AT+CSMS?	Read command reports current service setting along with supported message types in the format: **+CSMS: <service>,<mt>,<mo>,<bm>** **where:** **<service>** - messaging service (see above) **<mt>** - mobile terminated messages support (see above) **<mo>** - mobile originated messages support (see above) **<bm>** - broadcast type messages support (see above)
AT+CSMS=?	Test command reports the supported value of the parameter **<service>**.
Example	AT+CSMS=1 +CSMS: 1,1,1 OK AT+CSMS?

+CSMS - Select Message Service	
	+CSMS: 1,1,1,1 OK
Reference	3GPP TS 27.005; 3GPP TS 03.40/23.040; 3GPP TS 03.41/23.041

3.4.5.1.2 *Preferred Message Storage - +CPMS*

+CPMS - Preferred Message Storage	

AT+CPMS= <memr>[,<memw> [,<mems>]]	Set command selects memory storages **<memr>**, **<memw>** and **<mems>** to be used for reading, writing, sending and storing SMs.
	
	 Parameters:
	 <memr> - memory from which messages are read and deleted
	   "ME" – SMS memory storage in Flash
	   "SM" – SIM SMS memory storage (default)
	   "SR" – Status Report message storage (in SIM EF-SMSR file exists otherwise in the RAM volatile memory)
	
	 Note: "SR" non volatile memory is cleared when another SIM card is inserted. It is kept, even after a reset, while the same SIM card is inserted.
	
	 <memw> - memory to which writing and sending operations are made
	   "ME" – SMS memory storage in Flash
	   "SM" – SIM SMS memory storage (default)
	
	 <mems> - memory to which received SMs are preferred to be stored
	   "ME" – SMS memory storage in Flash
	   "SM" – SIM SMS memory storage (default)
	
	 The command returns the memory storage status in the format:
	
	 +CPMS: <usedr>,<totalr>,<usedw>,<totalw>,<useds>,<totals>
	
	 where:
	 <usedr> - number of SMs stored into **<memr>**
	 <totalr> - max number of SMs that **<memr>** can contain
	 <usedw> - number of SMs stored into **<memw>**
	 <totalw> max number of SMs that **<memw>** can contain
	 <useds> - number of SMs stored into **<mems>**
	 <totals> - max number of SMS that **<mems>** can contain

AT+CPMS?	Read command reports the message storage status in the format: **+CPMS: <memr>,<usedr>,<totalr>,<memw>,<usedw>,<totalw>, <mems>,<useds>,<totals>**

+CPMS - Preferred Message Storage	
	Where: <memr>, <memw> and **<mems>** are the selected storage memories for reading, writing and storing respectively.
AT+CPMS=?	Test command reports the supported values for parameters **<memr>, <memw>** and **<mems>**
Example	AT+CPMS? +CPMS: "ME",27, 50,"ME",27, 50,"SR",1,20 OK AT+CPMS="SM","ME","SM" +CPMS: 1,20,27, 50,1,20 OK AT+CPMS? +CPMS: "SM",1,20,"ME",27, 50,"SM",1,20 OK *(You have 1 out of 255 SMS SIM positions occupied)*
Reference	3GPP TS 27.005

3.4.5.1.3 *Message Format - +CMGF*

+CMGF - Message Format	

AT+CMGF= [<mode>]	Set command selects the format of messages used with send, list, read and write commands.
	
	Parameter:
	<mode>
	  0 - PDU mode, as defined in 3GPP TS 3.40/23.040 and
	    3GPP TS 3.41/23.041 (factory default)
	  1 - text mode

AT+CMGF?	Read command reports the current value of the parameter **<mode>**.
AT+CMGF=?	Test command reports the supported value of **<mode>** parameter.
Example	AT+CMGF=1 OK
Reference	3GPP TS 27.005

3.4.5.2 Message Configuration

3.4.5.2.1 *Service Center Address - +CSCA*

+CSCA -Service Center Address	

AT+CSCA=
\<number\>
[,\<type\>]
	Set command sets the Service Center Address to use for mobile originated SMS transmissions.
	
	Parameter:
	\<number\> - SC phone number in the format defined by **\<type\>**
	\<type\> - the type of number
	  129 - national numbering scheme
	  145 - international numbering scheme (contains the character "+")
	
	Note: to use the SM service, is mandatory to set a Service Center Address at which service requests directed.
	
	Note: in Text mode the settings is used by send & write commands; in PDU mode, setting is used by the same commands, but only when the length of the SMSC address coded into the **\<pdu\>** parameter equals zero.
	
	Note: the current settings are stored through **+CSAS**

AT+CSCA?	Read command reports the current value of the SCA in the format: **+CSCA: \<number\>,\<type\>** **Note:** If SCA is not present, the device reports an error message.
AT+CSCA=?	Test command returns the **OK** result code.
Example	AT+CSCA="821029190903",145 OK AT+CSCA? +CSCA: "+821029190903",145 OK
Reference	3GPP TS 27.005

3.4.5.2.2 *EPS – Network Registarion Status - +CEREG*

+CEREG – Network Registarion Status	
+CEREG=[**\<n\>**]	The set command controls the presentation of an unsolicited result code +CEREG: \<stat\> when \<n\>=1 and there is a change in the MT's EPS network registration status in E-UTRAN, or unsolicited result code. **+CEREG:** \<stat\>[,[\<tac\>],[\<ci\>],[\<AcT\>]] when \<n\>=2 and there is a change of the network cell in E-UTRAN. The parameters \<AcT\>, \<tac\> and \<ci\> are sent only if available.The value \<n\>=3 further extends the unsolicited result code with

+CEREG – Network Registarion Status	
	[,<cause_type>,<reject_cause>], when available, when the value of <stat> changes. Refer subclause 9.2 for possible <err> values. **Note:** If the EPS MT in GERAN/UTRAN/E-UTRAN also supports circuit mode services and/or GPRS services, the +CREG command and +CREG: result codes and/or the +CGREG command and +CGREG: result codes apply to the registration status and location information for those services. **Possible response(s):** **+CME ERROR: <err>**
+CEREG?	**Defined values:** **<n>:** integer type 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>[,[<tac>],[<ci>],[<AcT>]]** 3 - enable network registration, location information and EMM cause value information unsolicited result code. **+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]** **<stat>:** integer type; indicates the EPS registration status 0 - not registered, MT is not currently searching an operator to register to. 1 - registered, home network. 2 - not registered, but MT is currently trying to attach or searching an operator to register to. 3 - registration denied. 4 - unknown (e.g. out of E-UTRAN coverage). 5 - registered, roaming. 6 - registered for "SMS only", home network (not applicable) 7 - registered for "SMS only", roaming (not applicable). 8 - attached for emergency bearer services only (See Note 2). 9 - registered for "CSFB not preferred", home network (not applicable). 10 - registered for "CSFB not preferred", roaming (not applicable). **Note 2:** 3GPP TS 24.008 [8] and 3GPP TS 24.301 [83] specify the condition when the MS is considered as attached for emergency bearer services. **<tac>:** string type; two byte tracking area code in hexadecimal format (e.g. "00C3" equals 195 in decimal). **<ci>:** string type; four byte E-UTRAN cell ID in hexadecimal format. **<AcT>:** integer type; indicates the access technology of the serving cell. 0 - GSM (not applicable) 1 - GSM Compact (not applicable) 2 - UTRAN (not applicable)

+CEREG – Network Registarion Status	
	3 - GSM w/EGPRS (see Note 3) (not applicable)
	4 - UTRAN w/HSDPA (see Note 4) (not applicable)
	5 - UTRAN w/HSUPA (see Note 4) (not applicable)
	6 - UTRAN w/HSDPA and HSUPA (see Note 4) (not applicable)
	7 - E-UTRAN
	Note 3: 3GPP TS 44.060 [71] specifies the System Information messages which give the information about whether the serving cell supports EGPRS.
	Note 4: 3GPP TS 25.331 [74] specifies the System Information blocks which give the information about whether the serving cell supports HSDPA or HSUPA.
	<cause_type>: integer type; indicates the type of <reject_cause>.
	0 - Indicates that <reject_cause> contains an EMM cause value, see 3GPP TS 24.301 [83] Annex A.
	1 - Indicates that <reject_cause> contains a manufacturer-specific cause.
	<reject_cause>: integer type; contains the cause of the failed registration. The value is of type as defined by <cause_type>.
	+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>[,<cause_type>,<reject_cause>]]]
+CEREG=?	Test command returns values supported as a compound value.
	+CEREG: (list of supported <n>s)
Reference	3GPP TS 27.007

3.4.5.2.3   *PDP Context Read Dynamic Parameters +CGCONTRDP*

+CGCONTRDP parameter command syntax	
AT+CGCONTRDP = [<p_cid>]	**Possible response(s):** **+CGCONTRDP:** <p_cid>,<bearer_id>,<apn>[,<ip_addr>, <subnet_mask>[,<gw_addr>[,<DNS_prim_addr>[,<DNS_sec_addr> [,<P-CSCF_prim_addr>[,<P-CSCF_sec_addr>]]]]]][<CR><LF> **+CGCONTRDP:** <p_cid>,<bearer_id>,<apn>[,<ip_addr>, <subnet_mask>[,<gw_addr>[,<DNS_prim_addr>[,<DNS_sec_addr> [,<PCSCF_prim_addr>[,<PCSCF_sec_addr>]]]]]][…]] **Description:** The execution command returns the relevant information: <bearer_id>, <apn>, <ip_addr>, <subnet_mask>,<gw_addr>, <DNS_prim_addr>, <DNS_sec_addr>, <P-CSCF_prim_addr> and <P-CSCF_sec_addr> for a non- secondary PDP Context established by the network with the primary context identifier <p_cid>. If the context can't be found an ERROR response is returned. If the parameter <p_cid> omitted, the relevant information for all established PDP contexts returned. **Note:** The dynamic part of the PDP context will only exist if established by the network. The test command returns a list of <p_cid>s associated with active contexts.

+CGCONTRDP parameter command syntax	
	Defined values: **<p_cid> -** a numeric parameter specifies a particular non secondary PDP context definition. The parameter is local to the TE-MT interface and used in other PDP context-related commands. The <p_cid> rage is: (1 <= <p_id> <= 24) or (100 <= <p_id> <= 179) **<bearer_id> -** a numeric parameter identifies the bearer, EPS Bearer in EPS and NSAPI in UMTS/GPRS. **<APN> -** a string parameter which is a logical name that was used to select the GGSN or the external packet data network. **<ip_addr> -** a string parameter shows the IP Address of the MT. The string is given as dot-separated numeric (0-255) parameters on the form: "a1.a2.a3.a4" for IPv4 or "a1.a2.a3.a4.a5.a6.a7.a8" for IPv6. If the MT has dual stack capabilities the string shows first the dot separated IPv4 Address followed by the dot Separated IPv6 Global Prefix Address. The IPv4 address and the IPv6 address parameters are separated by space: "a1.a2.a3.a4 a1:a2:a3:a4:a5:a6:a7:a8" **<subnet_mask> -** a string parameter shows the subnet mask for the IP Address of the MT. The string given as dot-separated numeric (0-255) parameters. If the MT has dual stack capabilities the string shows the dot separated IPV4 subnet mask followed by the dot Separates IPV6 subnet mask. The subnet masks are separates by space. **<gw_addr> -** a string parameter shows the Gateway Address of the MT. The string is given as dot-separated numeric (0-255) parameters. If the MT has dual stack capabilities the parameter shows first the dot separated IPV4 Gateway address followed by the dot separated IPV6 Gateway Address. The gateway addresses are separated by space. **<DNS_prim_addr> -** a string parameter which shows the IP Address of the primary DNS Server. If the MT has dual stack capabilities the parameter shows first the dot separated IPV4 Address, followed by the dot separated IPV6 Address of DNS Server. **<DNS_sec_addr> -** a string parameter which shows the IP address of the secondary DNS Server. If the MT has dual stack capabilities the parameter shows first the dot separated IPV4 Address, followed by the dot separated IPV6 Address of DNS Server.

+CGCONTRDP parameter command syntax	
	<P_CSCF_prim_addr> - a string parameter which shows the IP Address of the primary P-CSCF Server. If the MT has dual stack capabilities the parameter shows first the dot separated IPV4 Address, followed by the dot separated IPV6 primary Address of P-CSCF Server. **<P_CSCF_sec_addr> -** a string parameter which shows the IP Address of the secondary P-CSCF Server. If the MT has dual stack capabilities the parameter shows first the dot separated IPV4 Address, followed by the dot separated IPV6 Address of P-CSCF Server.
+CGCONTRDP=?	**+CGCONTRDP:** (list of <p_cid>s associated with active contexts)

3.4.5.2.4 *Secondary PDP Context Read Dynamic Parameters - +CGSCONTRDP*

+CGSCONTRDP - parameter command syntax	
AT+CGSCONTRDP= **[<cid>]**	**Possible response(s):** **+CGSCONTRDP:**<cid>,<p_cid>,<bearer_id>[<CR><LF> **+CGSCONTRDP:** <cid>, <p_cid>,<bearer_id> […]] The execution command returns <p_cid> and <bearer_id> for a given <cid>. If the context cannot be found an ERROR response returned.If the parameter <cid> omitted, the <cid>, <p_cid> and <bearer_id> returned for all established PDP contexts. In EPS, the Traffic Flow parameters returned. **Note:** Parameters for network initiated PDP contexts returned as well. The dynamic part of the PDP context will only exist if established by the network. **Defined values:** **<cid>** a numeric parameter which specifies a particular PDP context or Traffic Flows definition. The parameter is local to the TE-MT interface and is used in other PDP context-related commands. **<p_cid>** a numeric parameter which specifies a particular PDP context definition or default EPS context Identifier which has been specified by use of the +CGDCONT command. The parameter is local to the TE-MT interface. **<bearer_id>** a numeric parameter which identifies the bearer, EPS Bearer in EPS and NSAPI in UMTS/GPRS.
+CGSCONTRDP=?	**+CGSCONTRDP:** (list of <cid>s associated with active contexts) The test command returns a list of <cid>s associated with active contexts.

3.4.5.2.1 *Set Text Mode Parameters - +CSMP*

+CSMP - Set Text Mode Parameters	

AT+CSMP=
[<fo>
[,<vp>
[,<pid>
[,<dcs>]]]]
	Set command is used to select values for additional parameters for storing and sending SMs when the text mode is used (**AT+CMGF=1**)
	
	Parameters:
	<fo> - depending on the command or result code:
	  first octet of 3GPP TS 03.40/23.040 SMS-DELIVER, SMS-SUBMIT (default 17), SMS-STATUS-REPORT, or SMS-COMMAND (default 2) in integer format.
	<vp> - depending on SMS-SUBMIT **<fo>** setting:
	  3GPP TS 03.40/23.040 TP-Validity-Period either in integer format (default 167) or in quoted time-string format.
	<pid> - 3GPP TS 03.40/23.040 TP-Protocol-Identifier in integer format.
	<dcs> - depending on the command or result code:
	  3GPP TS 03.38/23.038 SMS Data Coding Scheme (default 0), or Cell Broadcast Data Coding Scheme.
	
	Note: the current settings are stored through **+CSAS**

AT+CSMP?	Read command reports the current setting in the format: **+CSMP: < fo>,<vp>,<pid>,<dcs>**
AT+CSMP=?	Test command returns the **OK** result code.
Example	*Set the parameters for an outgoing message with 24 hours of validity period and default properties:* AT+CSMP=17,167,0,0 OK
Reference	3GPP TS 27.005; 3GPP TS 03.40/23.040; 3GPP TS 03.38/23.038

3.4.5.2.1 *Show Text Mode Parameters - +CSDH*

+CSDH - Show Text Mode Parameters	

AT+CSDH=
[<show>]
	Set command controls whether detailed header information is shown in text mode (**AT+CMGF=1**) result codes.
	
	Parameter:
	<show>
	  0 - do not show header values defined in commands **+CSCA** and **+CSMP** (**<sca>**, **<tosca>**, **<fo>**, **<vp>**, **<pid>** and **<dcs>**) nor **<length>**, **<toda>** or **<tooa>** in **+CMT**, **+CMGL**, **+CMGR** result codes for SMS-DELIVERs and SMS-SUBMITs in text mode. For SMS-COMMANDs in **+CMGR** result code do not show **<pid>**, **<mn>**, **<da>**, **<toda>**, **<length>** or **<cdata>**
	  1 - show the values in result codes

AT+CSDH?	Read command reports the current setting in the format:

+CSDH - Show Text Mode Parameters	
	+CSDH: <show>
AT+CSDH=?	Test command reports the supported range of values for parameter **<show>**
Reference	3GPP TS 27.005

3.4.5.2.2 *Save Settings - +CSAS*

+CSAS - Save Settings	
AT+CSAS[= <profile>]	Execution command saves settings which have been made by the **+CSCA**, **+CSMP** and **+CSCB** commands in local non-volatile memory. **Parameter:** **<profile>** 0 - it saves the settings to NVM (factory default). 1..n - SIM profile number; the value of n depends on the SIM(le9x0 allow to store up to 5). **Note:** certain settings may not be supported by the SIM. Therefore, the settings always saved to the NVM, regardless the value of **<profile>**. **Note:** If parameter is omitted the settings are saved in the non-volatile memory.
AT+CSAS=?	Test command returns the possible range of values for the parameter **<profile>**.
Reference	3GPP TS 27.005

3.4.5.2.3 *Restore Settings - +CRES*

+CRES - Restore Settings	
AT+CRES[= [<profile>]	Execution command restores message service settings saved by +CSAS command from either NVM or SIM. **Parameter:** **<profile>** 0 - it restores message service settings from NVM. 1..n - it restores message service settings from SIM. The value of n depends on the SIM (le9x0 allow to store up to 5). **Note:** certain settings may not be support by the SIM and therefore they are always restored from NVM, regardless the value of <profile>.

+CRES - Restore Settings	
	Note: If parameter is omitted the command restores message service settings from NVM.
AT+CRES=?	Test command returns the possible range of values for the parameter <profile>.
Reference	3GPP TS 27.005

3.4.5.3 Message Receiving and Reading

3.4.5.3.1 *New Message Indications to Terminal Equipment - +CNMI*

+CNMI - New Message Indications To Terminal Equipment	
AT+CNMI= [<mode> [,<mt>[,<bm> [,<ds> [,<bfr>]]]]]	Set command selects the behaviour of the device on how the receiving of new messages from the network indicated to the **DTE**. **Parameter:** **<mode>** - unsolicited result codes buffering option 0 - Buffer unsolicited result codes in the **TA**. If **TA** result code buffer is full, indications may be buffered in some other place or the oldest indications may be discarded and replaced with the new received indications. 1 - Discard indication and reject new received message unsolicited result codes when **TA-TE** link is reserved, otherwise forward them directly to the **TE**. 2 - Buffer unsolicited result codes in the TA in case the **DTE** is busy and flush them to the TE after reservation. Otherwise, forward them directly to the TE. 3 - if **<mt>** is set to 1 an indication via 100 ms break is issued when a SMS is received while the module is in GPRS online mode. It enables the hardware ring line for 1 s. too. **Note:** In <mode> field, "3" not supported. **<mt>** - result code indication reporting for SMS-DELIVER 0 - No SMS-DELIVER indications are routed to the TE and message is stored. 1 - If SMS-DELIVER is stored into ME/TA, indication of the memory location is routed to the TE using the following unsolicited result code: **+CMTI: <memr>,<index>** where: **<memr>** - memory storage where the new message is Stored: "SM" , "ME" **<index>** - location on the memory where SMS is stored. 2 - SMS-DELIVERs (except class 2 messages and messages in the message waiting indication group) are routed directly to the TE using the following unsolicited result code:

+CNMI - New Message Indications To Terminal Equipment	

	(PDU Mode)
	 +CMT: <alpha>,<length><CR><LF><pdu>
	where:
	<alpha> - alphanumeric representation of
	 originator/destination number corresponding to
	 the entry found in MT phonebook. used character
	 set should be the one selected with command +CSCS.
	<length> - PDU length
	<pdu> - PDU message
	
	 (TEXT Mode)
	 +CMT:<oa>,<alpha>,<scts>[,<tooa>,<fo>,<pid>,<dcs>,
	 ***<sca>,<tosca>,<length>]*<CR><LF><data>**
	 (the information written in italics will be present
	 depending on **+CSDH** last setting)
	where:
	<oa> - originating address, string type converted in
	 the currently selected character set (see +CSCS)
	 <alpha> - alphanumeric representation of **<oa>**, used
	 character set should be the one selected with command
	 +CSCS.
	 <scts> - arrival time of the message to the SC
	 <tooa>, <tosca> - type of number **<oa>** or *<sca>:*
	 129 - number in national format
	 145 - number in international format(contains the "+")
	<fo> - first octet of 3GPP TS 03.40/23.040
	<pid> - Protocol Identifier
	<dcs> - Data Coding Scheme
	<sca> - Service Centre address, string type, converted
	 in the currently selected character set (see +CSCS)
	<length> - text length
	<data> - TP-User-Data
	    • If **<dcs>** indicates that GSM03.38/23.038 default alphabet is used and **<fo>** indicates that GSM03.40/23.040 TP-User-Data-Header-Indication is not set (bit 6 of **<fo>** is 0), each character of GSM/WCDMA alphabet will be converted into current TE character set (see +CSCS)
	    • If **<dcs>** indicates that 8-bit or UCS2 data coding scheme is used or **<fo>** indicates that GSM03.40/23.040 TP-User-Data-Header-Indication is set (bit 6 of **<fo>** is 1), each 8-bit octet will be converted into two IRA character long hexadecimal number (e.g. octet 0x2A will be converted as two characters 0x32 0x41)
	
	Class 2 messages and messages in the message waiting indication group (stored message) result in indication as defined in **<mt>=1**.
	
	Acknowledge for the received SMS-DELIVER SM is sent

+CNMI - New Message Indications To Terminal Equipment	
	to network immediately when +CSMS <service> is set to '0' or when +CSMS <service> is set to '1', acknowledge is sent via +CNMA command during predefine time-out, an error is sent to network in case timeout expire, Next +CMT response is depend on acknowledge of current received +CMT response in case +CSMS <service> parameter set to '1'.

3 - Class 3 SMS-DELIVERs are routed directly to TE using unsolicited result codes defined in **<mt>=2**. Messages of other data coding schemes result in indication as defined in **<mt>=1**.

<bm> - broadcast reporting option

  0 - Cell Broadcast Messages are not sent to the **DTE**

  2 - New Cell Broadcast Messages are sent to the **DTE** with the unsolicited result code:

  (PDU Mode)
    +CBM: <length><CR><LF><PDU>
    where:
    <length> - PDU length
    <PDU> - message PDU

    (TEXT Mode)
    +CBM:<sn>,<mid>,<dcs>,<pag>,<pags><CR><LF><data>
    where:
    <sn> - message serial number
    <mid> - message ID
    <dcs> - Data Coding Scheme
    <pag> - page number
    <pags> - total number of pages of the message
    <data> - CBM Content of Message

- If **<dcs>** indicates that GSM03.38/23.038 default alphabet is used , each character of GSM/WCDMA alphabet will be converted into current TE character set (see +CSCS)
- If **<dcs>** indicates that 8-bit or UCS2 data coding scheme is used, each 8-bit octet will be converted into two IRA character long hexadecimal number (e.g. octet 0x2A will be converted as two characters 0x32 0x41)

<ds> - SMS-STATUS-REPORTs reporting option

  0 - status report receiving is not reported to the **DTE** and messages are stored

  1 - the status report is sent to the **DTE** with the following unsolicited result code:

    (PDU Mode)
    +CDS: <length><CR><LF><PDU>
    where:
    <length> - PDU length
    <PDU> - message PDU

+CNMI - New Message Indications To Terminal Equipment	

	(TEXT Mode)
	+CDS: <fo>,<mr>,<ra>,<tora>,<scts>,<dt>,<st>
	where:
	<fo> - first octet of the message PDU
	  <mr> - message **Reference** number
	  <ra> - recipient address, string type, represented in the currently selected character set (see +CSCS)
	  <tora> - type of number **<ra>**
	  <scts> - arrival time of the message to the SC
	  <dt> - sending time of the message
	  <st> - message status as coded in the PDU
	
	Acknowledge for the received SMS-STATUS-REPORT SM is sent to network immediately when +CSMS <service> is set to '0' or when +CSMS <service> is set to '1', acknowledge is sent via +CNMA command during pre-defined timeout,
	an error is sent to network in case timeout expire,
	Next +CDS response is depend on acknowledge of current received +CDS response in case +CSMS <service> parameter set to '1'.
	
	  2 - if a status report is stored, then the following unsolicited result code is sent:
	    +CDSI: <memr>,<index>
	
	 where:
	    <memr> - memory storage where the new message is stored "SR"
	    <index> - location on the memory where SMS is stored
	  <bfr> - buffered result codes handling method:
	  0 - **TA** buffer of unsolicited result codes defined within this command is flushed to the **TE** when **<mode>=1..3** is entered (**OK** response shall be given before flushing the codes)
	 1 - **TA** buffer of unsolicited result codes defined within this command is cleared when **<mode>=1..3** is entered.

AT+CNMI?	Read command returns the current parameter settings for **+CNMI** command in the form: **+CNMI: <mode>,<mt>,<bm>,<ds>,<bfr>**
AT+CNMI=?	Test command reports the supported range of values for the **+CNMI** command parameters.
Reference	3GPP TS 27.005
Example	AT+CMGF=1 OK AT+CNMI=1,2,0,1,0 OK *Received message from network* +CMT: "+821020955219",,"07/07/26,20:09:07+36"

+CNMI - New Message Indications To Terminal Equipment	
	TEST MESSAGE
Note	**DTR** signal is ignored, hence the indication is sent even if the **DTE** is inactive (**DTR** signal is **Low**). In this case the unsolicited result code may be lost so if MODULE remains active while **DTE** is not, at **DTE** startup is suggested to check whether new messages have reached the device meanwhile with command **AT+CMGL=0** that lists the new messages received.

3.4.5.3.2 *List Messages - +CMGL*

+CMGL - List Messages	

AT+CMGL= [=<stat>]	Execution command reports the list of all the messages with status value **<stat>** stored into **<memr>** message storage (**<memr>** is the message storage for read and delete SMs as last settings of command +CPMS).
	
	The parameter type and the command output depend on the last settings of command **+CMGF** (message format to be used)
	
	(PDU Mode)
	Parameter:
	<stat>
	  0 - new message
	  1 - read message
	  2 - stored message not yet sent
	  3 - stored message already sent
	  4 - all messages.
	
	Each message to be listed is represented in the format:
	+CMGL: <index>,<stat>,<alpha>,<length><CR><LF><pdu>
	
	where:
	<index> - message position in the memory storage list.
	<stat> - status of the message
	<alpha> - string type alphanumeric representation of
	<da> or **<oa>**, corresponding to an entry found in the phonebook; used character set is the one selected with command +CSCS.
	<length> - length of the PDU in bytes
	<pdu> - message in PDU format according to 3GPP TS 3.40/23.040
	
	(Text Mode)
	Parameter:
	<stat>
	  "REC UNREAD" - new message
	  "REC READ" - read message
	  "STO UNSENT" - stored message not yet sent

+CMGL - List Messages	

	"STO SENT" - stored message already sent
	"ALL" - all messages.
	
	Each message to be listed is represented in the format (the information written in italics will be present depending on +CSDH last setting):
	
	+CMGL: <index>,<stat>,<oa/da>,<alpha>,<scts>[,<tooa/toda>, <length>]<CR><LF> <data>
	
	Where:
	<index> - message position in the storage
	<stat> - message status
	<oa/da> - originator/destination address, string type , represented in the currently selected character set (see +CSCS)
	<alpha> - string type alphanumeric representation of **<da>** or **<oa>**, corresponding to an entry found in the phonebook; used character set is the one selected with command +CSCS.
	<scts> - TP-Service Centre Time Stamp in Time String Format
	<tooa/toda> - type of number **<oa/da>**
	  129 - number in national format
	  145 - number in international format (contains the "+")
	<length> - text length
	<data> - TP-User-Data
	  • If **<dcs>** indicates that GSM03.38/23.038 default alphabet is used , each character of GSM/WCDMA alphabet will be converted into current TE character set (see +CSCS)
	  • If **<dcs>** indicates that 8-bit or UCS2 data coding scheme is used, each 8-bit octet will be converted into two IRA character long hexadecimal number (e.g. octet 0x2A will be converted as two characters 0x32 0x41)
	
	Each message delivery confirm is represented in the format**:**
	
	+CMGL: <index>,<stat>,<fo>,<mr>,,,<scts>,<dt>,<st>
	
	Where:
	<index> - message position in the storage
	<stat> - message status
	<fo> - first octet of the message PDU
	<mr> - message **Reference** number
	<scts> - arrival time of the message to the SC
	<dt> - sending time of the message
	<st> - message status as coded in the PDU
	
	Note: If parameter is omitted the command returns the list of sms with "**REC UNREAD**" status.

AT+CMGL=?	Test command returns a list of supported **<stat>**s

+CMGL - List Messages	
Example	AT+CMGF=1 *Set Text mode* OK AT+CMGL +CMGL: 1,"REC UNREAD","+821020955219",,"07/07/26,20:05:11+36" SMS Test message +CMGL: 2,"REC UNREAD","+821020955219",,"07/07/26,20:05:58+36" SMS Test message... +CMGL: 3,"REC UNREAD","+821020955219",,"07/07/26,20:06:37+36" SMS Test Message.. +CMGL: 4,"REC UNREAD","+821020955219",,"07/07/26,20:07:43+36" TEST MESSAGE.. +CMGL: 5,"REC UNREAD","+821020955219",,"07/07/26,20:09:07+36" TEST MESSAGE OK AT+CMGF=0 *Set PDU mode* OK AT+CMGL=2 +CMGL: 0,2,,24 079128019291903011640A8110567892820000A70CF4F29C0E6A97E7F3F0B90 C +CMGL: 1,2,,21 079128019291903011640A8110516529700000A709027A794E77B95C2E +CMGL: 26,2,,17 0881280100990102591164 0A8110567892820014A704C7D1B1DB OK
Reference	3GPP TS 27.005

3.4.5.3.3   *Read Message - +CMGR*

+CMGR - Read Message	
AT+CMGR= <index>	Execution command reports the message with location value **<index>** from **<memr>** message storage (**<memr>** is the message storage for read and delete SMs as last settings of command +CPMS). **Parameter:** **<index>** - message index. The output depends on the last settings of command **+CMGF** (message format to be used) **(PDU Mode)** The output has the following format:

+CMGR - Read Message

+CMGR: <stat>,<alpha>,<length><CR><LF><pdu>

Where:
<stat> - status of the message
  0 - new message
  1 - read message
  2 - stored message not yet sent
  3 - stored message already sent
<alpha> - string type alphanumeric representation of **<da>** or **<oa>**,
      corresponding to an entry found in the phonebook; used character set is
      the one selected with command +CSCS.
<length> - length of the PDU in bytes.
<pdu> - message in PDU format according to 3GPP TS 3.40/23.040.

The status of the message and entire message data unit **<pdu>** returned.

(Text Mode)
Output format for received messages (the information written in *italics* will be
present depending on **+CSDH** last setting)*:*
**+CMGR: <stat>,<oa>,<alpha>,<scts>*[,<tooa>,<fo>,<pid>,*
<dcs>,<sca>,<tosca>,<length>]<CR><LF><data>**

Output format for sent messages:
**+CMGR: <stat>,<da>,<alpha>*[,<toda>,<fo>,<pid>,<dcs>,[*
<vp>],<sca>,<tosca>,<length>]<CR><LF><data>**

Output format for message delivery confirm:
+CMGR: <stat>,<fo>,<mr>,,,<scts>,<dt>,<st>

where:
<stat> - status of the message
  "REC UNREAD" - new received message unread
  "REC READ" - received message read
  "STO UNSENT" - message stored not yet sent
  "STO SENT" - message stored already sent
<fo> - first octet of the message PDU
<mr> - message **Reference** number
<scts> - arrival time of the message to the SC
<dt> - sending time of the message
<st> - message status as coded in the PDU
<pid> - Protocol Identifier
<dcs> - Data Coding Scheme
<oa> - Originator address, string type represented in the currently selected
      character set (see +CSCS)
<da> - Destination address, string type represented in the currently selected
      character set (see +CSCS)

+CMGR - Read Message	

	<alpha> - string type alphanumeric representation of **<da>** or **<oa>**, corresponding to an entry found in the phonebook; used character set is the one selected with command +CSCS.
	 <sca> - Service Centre number
	 <tooa>,*<toda >*,*<tosca>* - type of number **<oa>**,**<da>**,*<sca>*
	  129 - number in national format
	  145 - number in international format (contains the "+")
	 <length> - text length
	 <data> - TP-User_data
	 • If **<dcs>** indicates that GSM03.38/23.038 default alphabet is used , each character of GSM/WCDMA alphabet will be converted into current TE character set (see +CSCS)
	 • If **<dcs>** indicates that 8-bit or LE9x0 data coding scheme is used, each 8-bit octet will be converted into two IRA character long hexadecimal number (e.g. octet 0x2A will be converted as two characters 0x32 0x41)
	
	 Note: in both cases if status of the message is 'received unread', status in the storage changes to 'received read'.

AT+CMGR=?	Test command returns the **OK** result code
Example	AT+CMGF=0 AT+CMGR=1 +CMGR: 2,,21 0791280192291903011640A8110516529700000A709027A794E77B95C2E OK AT+CMGF=1 OK AT+CMGR=3 +CMGR: "REC READ","+821020955219",,"07/07/19,10:06:34+36" test message/..... OK
Reference	3GPP TS 27.005

3.4.5.1 Message Sending and Writing

3.4.5.1.1 *Send Message - +CMGS*

+CMGS - Send Message	
(PDU Mode) **AT+CMGS=** **<length>**	**(PDU Mode)** Execution command sends to the network a message. **Parameter:** **<length>** - length of the PDU to be sent in bytes (excluding the SMSC address octets) 7..164 After command line is terminated with **<CR>**, the device responds sending a four character sequence prompt: **<CR><LF><greater_than><space> (IRA 13, 10, 62, 32)** and waits for the specified number of bytes. **Note:** the **DCD** signal shall be in **ON** state while PDU is given. **Note:** the echoing of given characters back from the TA is controlled by echo command **E** **Note:** the **PDU** shall be hexadecimal format (each octet of the **PDU** is given as two IRA character long hexadecimal number) and given in one line. **Note:** when the octet length of the SMSC address (given in the **PDU**) equals zero, the SMSC address set with command +CSCA is used. In this case, the SMSC Type-of-Address octet shall not be present in the **PDU**. To send the message issue **Ctrl-Z** char (**0x1A** hex). To exit without sending the message issue **ESC** char (**0x1B** hex). If message is successfully sent to the network, then the result is sent in the format: **Note:** Optionally (when +CSMS <service> value is 1 and network supports) <scts> is returned: **+CMGS: <mr>[, <scts>]** **Where:** **<mr>** - message **Reference** number. **<scts>** - TP-Service Centre Time Stamp in Time String Format. **Note:** if message sending fails for some reason, an error code reported. **Note:** to ensure that during the command execution, which may take several seconds, no other SIM interacting commands issued, care must taken.

(Text Mode)
AT+CMGS=
<da>
[,<toda>]
	(Text Mode)
	Execution command sends to the network a message.
	Parameters:
	<da> - destination address, string type represented in the currently selected character set (see +CSCS).
	<toda> - type of destination address
	  129 - number in national format
	  145 - number in international format(contains the "+")

+CMGS - Send Message
After command line is terminated with **<CR>**, the device responds sending a four character sequence prompt: **<CR><LF><greater_than><space> (IRA 13, 10, 62, 32)** After this prompt text can be entered; the entered text should be formatted as follows: • if current **<dcs>** (see +CSMP) indicates that GSM03.38/23.038 default alphabet is used and current **<fo>** (see +CSMP) indicates that 3GPP TS 03.40/23.040 TP-User-Data-Header-Indication is not set, then ME/TA converts the entered text into GSM/WCDMA alphabet, according to 3GPP TS 27.005, Annex A; **backspace** can be used to delete last character and **carriage returns** can be used. • if current **<dcs>** (see +CSMP) indicates that 8-bit or UCS2 data coding scheme is used or current **<fo>** (see +CSMP) indicates that 3GPP TS 03.40/23.040 TP-User-Data-Header-Indication is set, the entered text should consist of two IRA character long hexadecimal numbers which ME/TA converts into 8-bit octet (e.g. the **'asterisk'** will be entered as **2A** (**IRA50** and **IRA65**) and this will be converted to an octet with integer value **0x2A**) **Note:** the **DCD** signal shall be in **ON** state while text entered. **Note:** the echoing of entered characters back from the TA is controlled by echo command **E** To send the message issue **Ctrl-Z** char (**0x1A** hex). To exit without sending the message issue **ESC** char (**0x1B** hex). If message is successfully sent to the network, then the result is sent in the format: **Note** : Optionally (when +CSMS <service> value is 1 and network supports) <scts> is returned: **+CMGS: <mr>[, <scts>]** **Where:** **<mr>** - message **Reference** number. **<scts>** - TP-Service Centre Time Stamp in Time String Format. **Note:** if message sending fails for some reason, an error code reported. **Note:** care must be taken to ensure that during the command execution, which may take several seconds, no other SIM interacting commands are issued. **Note:** it is possible to save a concatenation of at most 16 SMs; the maximum number of chars depends on the <dcs>: 2448 chars; 2144 chars if 8-bit

+CMGS - Send Message	
	is used;1072 chars if UCS2 is used
AT+CMGS=?	Test command resturns the **OK** result code.
Note	To avoid malfunctions is suggested to wait for the **+CMGS: <mr>** or **+CMS ERROR: <err>** response before issuing further commands.
Example	*Set PDU mode* AT+CMGF=0 AT+CMGS=18 > 0881280100990102591155078110209055127F90000A704F4F29C0E +CMGS: 124 OK *Set text mode* AT+CMGF=1 AT+CSMP=17,167,0,0 AT+CMGS="01090255219",129 >TEST MESSAGE +CMGS:125 OK
Reference	3GPP TS 27.005
AT+CMSS=?	Test command resturns the **OK** result code.
Note	To avoid malfunctions is suggested to wait for the **+CMSS: <mr>** or **+CMS ERROR: <err>** response before issuing further commands.
Example	AT+CMGF=1 OK AT+CMGW="0165872928" > test message... +CMGW: 28 AT+CMSS=28 +CMSS: 136 OK
Reference	3GPP TS 27.005

3.4.5.1.2 *Write Message to Memory - +CMGW*

+CMGW - Write Message To Memory	
(PDU Mode) **AT+CMGW=** **<length>** **[,<stat>]**	**(PDU Mode)** Execution command writes in the **<memw>** memory storage a new message. **Parameter:**

+CMGW - Write Message To Memory	

	<length> - length in bytes of the PDU to be written.
	  7..164
	<stat> - message status.
	  0 - new message
	  1 - read message
	  2 - stored message not yet sent (default)
	  3 - stored message already sent
	
	The device responds to the command with the prompt '>' and waits for the specified number of bytes.
	
	To write the message issue **Ctrl-Z** char (**0x1A** hex).
	To exit without writing the message issue **ESC** char (**0x1B** hex).
	
	If message is successfully written in the memory, then the result is sent in the format:
	
	+CMGW: <index>
	
	where:
	<index> - message location index in the memory **<memw>**.
	
	If message storing fails for some reason an "error" code reported.
	
	Note: to ensure that during the command execution, no other SIM interacting commands issued care must be taken of.

(Text Mode)
AT+CMGW[=<da>
[,<toda>
[,<stat>]]]
	(Text Mode)
	Execution command writes in the **<memw>** memory storage a new message.
	
	Parameters:
	<da> - destination address, string type represented in the currently selected
	        character set (see +CSCS).
	<toda> - type of destination address.
	  129 - number in national format
	  145 - number in international format (contains the "+")
	<stat> - message status.
	  "REC UNREAD" - new received message unread
	  "REC READ" - received message read
	  "STO UNSENT" - message stored not yet sent (default)
	  "STO SENT" - message stored already sent
	
	After command line is terminated with **<CR>**, the device responds sending a four character sequence prompt:
	
	<CR><LF><greater_than><space> (IRA 13, 10, 62, 32)

+CMGW - Write Message To Memory	
	After this prompt text can be entered; the entered text should be formatted as follows: • if current **<dcs>** (see +CSMP) indicates that GSM03.38/23.038 default alphabet is used and current **<fo>** (see +CSMP) indicates that 3GPP TS 03.40/23.040 TP-User-Data-Header-Indication is not set, then ME/TA converts the entered text into GSM/WCDMA alphabet, according to 3GPP TS 27.005, Annex A; **backspace** can be used to delete last character and **carriage returns** can be used. • if current **<dcs>** (see +CSMP) indicates that 8-bit or UCS2 data coding scheme is used or current **<fo>** (see +CSMP) indicates that 3GPP TS 03.40/23.040 TP-User-Data-Header-Indication is set, the entered text should consist of two IRA character long hexadecimal numbers which ME/TA converts into 8-bit octet (e.g. the **'asterisk'** will be entered as **2A** (**IRA50** and **IRA65**) and this will be converted to an octet with integer value **0x2A**) **Note:** the **DCD** signal shall be in ON state while text entered. **Note:** the echoing of entered characters back from the TA is controlled by echo command **E** To write the message issue **Ctrl-Z** char (**0x1A** hex). To exit without writing the message issue **ESC** char (**0x1B** hex). If message is successfully written in the memory, then the result is sent in the format: **+CMGW: <index>** **where:** **<index>** - message location index in the memory **<memw>**. If message storing fails for some reason, an error code reported. **Note:** care will taken to ensure that during the command execution, no other SIM interacting commands issued. **Note:** it is possible to save a concatenation of at most 16 SMs; the maximum number of chars depends on the <dcs>: 2448 chars; 2144 chars if 8-bit is used;1072 chars if UCS2 is used **Note**: in text mode, not only SUBMIT messages can be stored in SIM, but also DELIVER messages. The type of saved message depends upon the current <fo> parameter (see +CSMP). For a DELIVER message, current <vp> parameter (see +CSMP) is used to set the message Service Centre Time Stamp <scts>, so it has to be an absolute time string, e.g. "09/01/12,11:15:00+04".

+CMGW - Write Message To Memory	
	SUBMIT messages can only be stored with status "STO UNSENT" or "STO SENT"; DELIVER messages can only be stored with status "REC UNREAD" or "REC READ".
AT+CMGW=?	Test command returns the **OK** result code.
Reference	3GPP TS 27.005
Example	AT+CMGF=0 *set PDU mode* OK AT+CMGW=18 > 088128010099010259115507811020905512F90000A704F4F29C0E +CMGW: 29 OK AT+CMGF=1 *set text mode* OK AT+CSMP=17,167,0,0 OK AT+CSCA="821029190903",145 OK AT+CMGW="0165872928" > test message... +CMGW: 28
Note	To avoid malfunctions is suggested to wait for the **+CMGW: <index>** or **+CMS ERROR: <err>** response before issuing further commands.

3.4.5.1.3 *Delete Message - +CMGD*

+CMGD - Delete Message	

AT+CMGD=
<index>
[,<delflag>]
	Execution command deletes from memory **<memr>** the message(s).
	
	Parameter:
	<index> - message index in the selected storage **<memr>**
	<delflag> - an integer indicating multiple message deletion request.
	  0 (or omitted) - delete message specified in **<index>**
	  1 - delete all read messages from **<memr>** storage, leaving unread messages and stored mobile originated messages (whether sent or not) untouched
	  2 - delete all read messages from **<memr>** storage and sent mobile originated messages, leaving unread messages and unsent mobile originated messages untouched
	  3 - delete all read messages from **<memr>** storage, sent and unsent mobile originated messages, leaving unread messages untouched

+CMGD - Delete Message	
	4 - Delete all messages from **\<memr\>** storage. **Note:** if **\<delflag\>** is present and not set to 0 then **\<index\>** is ignored and ME shall follow the rules for **\<delflag\>** shown above.
AT+CMGD=?	Test command shows the valid memory locations and optionally the supported values of **\<delflag\>**. **+CMGD: (supported \<index\>s list)[,(supported \<delflag\>s list)]**
Example	AT+CMGD=? +CMGD: (0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50),(0-4) OK AT+CMGD=11 *Delete message in 10th record* OK AT+CMGD=1,4 *Delete all messages* OK
Reference	3GPP TS 27.005

3.4.6 PSM and eDRX Commands

3.4.6.1 PSM AT Commands

3.4.6.1.1 *Power Saving Mode Setting - +CPSMS*

+CPSMS – Power Saving Mode Setting	

AT+CPSMS= [<mode>[,<Requested_Periodic-RAU>[,<Requested_GPRS-READY-timer>[,<Requested_Periodic-TAU>[,<Requested_Active-Time>]]]]]	The set command controls the setting of the UEs power saving mode (PSM) parameters. The command controls whether the UE wants to apply PSM or not, as well as the requested extended periodic RAU value and the requested GPRS READY timer value in GERAN/UTRAN, the requested extended periodic TAU value in E-UTRAN and the requested Active Time value.
	
	Parameters:
	<mode> - integer type. Indication to disable or enable the use of PSM in the UE.
	  0 - Disable the use of PSM
	  1 - Enable the use of PSM
	
	<Requested_Periodic-RAU> - string type; one byte in an 8 bit format. Requested extended periodic RAU value (**T3312**) to be allocated to the UE in GERAN/UTRAN. The requested extended periodic RAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see the GPRS Timer3 IE in 3GPP TS 24.
	
	<Requested_GPRS-READY-timer> - string type; one byte in an 8 bit format. Requested GPRS READY timer value (**T3314**) to be allocated to the UE in GERAN/UTRAN. The requested GPRS READY timer value is coded as one byte (octet 2) of the GPRS Timer information element coded as bit format (e.g. "01000011" equals 3 decihours or 18 minutes). For the coding and the value range, see the GPRS Timer IE in 3GPP TS 24.008.
	
	<Requested_Periodic-TAU> - string type; one byte in an 8 bit format. Requested extended periodic TAU value (**T3412**) to be allocated to the UE in E-UTRAN. The requested extended periodic TAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see the GPRS Timer 3 IE in 3GPP TS 24.008.
	
	<Requested_Active-Time> - string type; one byte in an 8 bit format. Requested Active Time value (**T3324**) to be allocated to the UE. The requested Active Time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes).

+CPSMS – Power Saving Mode Setting	
	For the coding and the value range, see the GPRS Timer 2 IE in 3GPP TS 24.008. **Note**: A special form of the command can be given as **AT+CPSMS=** (with all parameters omitted). In this form, the parameter **<mode>** will be set to 0, the use of PSM will be disabled and data for all parameters in command +CPSMS will be removed or, if available, set to the manufacturer specific default values.
AT+CPSMS?	Read command returns the current parameter values in the format: **+CPSMS: <mode>,[<Requested_Periodic-RAU>],[<Requested_GPRS-READY-timer>],[<Requested_Periodic-TAU>],[<Requested_Active-Time>]**
AT+CPSMS=?	Test command reports the range for the parameters in the format: **+CPSMS: (list of supported <mode>s),(list of supported <Requested_Periodic-RAU>s),(list of supported <Requested_GPRS-READY-timer>s),(list of supported <Requested_Periodic-TAU>s),(list of supported <Requested_Active-Time>s)**
Example	In E-UTRAN AT+CPSMS? +CPSMS: 1,,,"01100001","01100010" OK It means that Power Saving Mode is set to enabled and module enters in PSM after a minute (T3324 = 33) and stay in this mode for two minute (T3412 = 162). AT+CPSMS=1,,,162,33 OK AT+CPSMS=0 OK It means that PSM is set to disable, the module does not go to Power Saving Mode in any case. AT+CPSMS=? +CPSMS: (0-1),(<Units(0-6)><TimerValue(0-31)> in bits), (<Units(0-2)><TimerValue(0-31)> in bits), (<Units(0-6)><TimerValue(0-31)> in bits),

+CPSMS – Power Saving Mode Setting	
	(<Units(0-2)><TimerValue(0-31)> in bits)
Notes	How to manage timer values octet. T3412 value: Bits 5 to 1 represent the binary coded timer value. Bits 6 to 8 defines the timer value unit as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 10 minutes 0 0 1 value is incremented in multiples of 1 hour 0 1 0 value is incremented in multiples of 10 hours 0 1 1 value is incremented in multiples of 2 seconds 1 0 0 value is incremented in multiples of 30 seconds 1 0 1 value is incremented in multiples of 1 minute 1 1 0 value is incremented in multiples of 320 hours 1 1 1 value indicates that the timer is deactivated. Example: 10100010 -> 101 means values is incremented in multiples of 1 minute, 10 means 2 -> the obtained value is 2 minutes T3324 value: Bits 5 to 1 represent the binary coded timer value. Bits 6 to 8 defines the timer value unit for the GPRS timer as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 2 seconds 0 0 1 value is incremented in multiples of 1 minute 0 1 0 value is incremented in multiples of decihours 1 1 1 value indicates that the timer is deactivated.
Reference	3GPP TS 27.007

3.4.6.1 eDRX AT Commands

3.4.6.1.1 eDRX setting - +CEDRXS

+CEDRXS - eDRX setting	
AT+CEDRXS=[<mode>,[<AcT-type>[,<Requested_eDRX_value>]]]	Set command controls the setting of the UEs eDRX parameters. The command controls whether the UE wants to apply eDRX or not, as well as the requested eDRX value for each specified type of access technology. Parameter:

+CEDRXS - eDRX setting	
	\<mode\>: disable or enable the use of eDRX in the UE. This parameter is applicable to all specified types of access technology, i.e. the most recent setting of \<mode\> will take effect for all specified values of \<AcT\>. 0 - disable the use of eDRX 1 - enable the use of eDRX 2 - enable the use of eDRX and enable the unsolicited result code 3 - disable the use of eDRX and discard all parameters for eDRX or, if available, reset to the manufacturer specific default values If **\<mode\>=2** and there is a change in the eDRX parameters provided by the network, the unsolicited result code reports: **+CEDRXP: \<AcT-type\>[,\<Requested_eDRX_value\>[,\<NW-provided_eDRX_value\>[,\<Paging_time_window\>]]]** **\<AcT-type\>**: type of access technology. 1 – CAT M1 2 - GSM (A/Gb mode) 3 - UTRAN (Iu mode) 4 - E-UTRAN (WB-S1 mode) 5 - E-UTRAN (NB-S1 mode) **\<Requested_eDRX_value\>**: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008, Table 10.5.5.32/3GPP TS 24.008. **\<NW-provided_eDRX_value\>**: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 Table 10.5.5.32/3GPP TS 24.008. **\<Paging_time_window\>**: string type; half a byte in a 4 bit format. The paging time window referes to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008). For the coding and the value range, see the Extended DRX

+CEDRXS - eDRX setting	
	parameters information element in 3GPP TS 24.008 Table 10.5.5.32/3GPP TS 24.008.
AT+CEDRXS?	Read command returns the current settings for each defined value of **<AcT-type>**.
AT+CEDRXS=?	Test command returns the supported **<mode>**s and the value ranges for the access technology and the requested eDRX value as compound values.
Reference	3GPP TS 27.007

3.4.6.1.2 *eDRX read dynamic parameters - +CEDRXRDP*

+CEDRXRDP - eDRX read dynamic parameters	
AT+CEDRXRDP	Execution command returns **<AcT-type>, <Requested_eDRX_value>**, **<NW-provided_eDRX_value>** and **<Paging_time_window>** if eDRX is used for the cell that the MS is currently registered to, in the format: **+CEDRXP: <AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]** If the cell that the MS is currently registered to is not using eDRX, **<AcT-type>**=0 is returned. Parameter: **<AcT-type>**: type of access technology. 0 - Access technology is not using eDRX 1 – CAT M1 2 - GSM (A/Gb mode) 3 - UTRAN (Iu mode) 4 - E-UTRAN (WB-S1 mode) 5 - E-UTRAN (NB-S1 mode) **<Requested_eDRX_value>**: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008, Table 10.5.5.32/3GPP TS 24.008. **<NW-provided_eDRX_value>**: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008). For the

+CEDRXRDP - eDRX read dynamic parameters	
	coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 Table 10.5.5.32/3GPP TS 24.008. **<Paging_time_window>**: string type; half a byte in a 4 bit format. The paging time window referes to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008). For the coding and the value range, see the Extended DRX parameters information element in 3GPP TS 24.008 Table 10.5.5.32/3GPP TS 24.008.
AT+CEDRXRDP=?	Test command returns the supported **<mode>**s and the value ranges for the access technology and the requested eDRX value as compound values.
Reference	3GPP TS 27.007

3.4.7 Custom AT Commands

3.4.7.1 Configuration AT Commands

3.4.7.1.1 *Manufacturer Identification - #CGMI*

#CGMI - Manufacturer Identification	
AT#CGMI	Execution command returns the "device manufacturer identification code" with command echo.
AT#CGMI=?	Test command returns the **OK** result code.
Example	AT#CGMI #CGMI: Telit OK

3.4.7.1.2 *Model Identification - #CGMM*

#CGMM - Model Identification	
AT#CGMM	Execution command returns the "device model identification" code with command echo.
AT#CGMM=?	Test command returns the **OK** result code.
Example	AT#CGMM #CGMM:LE920 OK

3.4.7.1.3 *Revision Identification - #CGMR*

#CGMR - Revision Identification	
AT#CGMR	Execution command returns "device software revision number" with command echo.
AT#CGMR=?	Test command returns the **OK** result code.
Example	AT#CGMR #CGMR: 08.01.005 OK

3.4.7.1.4 *Product Serial Number Identification - #CGSN*

#CGSN - Product Serial Number Identification	
AT#CGSN	Execution command returns the product serial number, identified as the IMEI of the mobile, with command echo.

AT#CGSN[=<snt>]	Set command causes the TA to return IMEI (International Mobile station Equipment Identity number) and related information to identify the MT that the TE connected to.
	Parameter:
	<snt> - indicating the serial number type that has been requested.
	
	    0 returns <sn>
	
	    1 returns the IMEI (International Mobile station Equipment Identity)
	
	    2 returns the IMEISV (International Mobile station Equipment Identity and Software Version number)
	
	    3 returns the SVN (Software Version Number)
	
	where:
	
	<sn> **-** Indicate the product "serial number", identified as the IMEI of the mobile, with command echo.
	
	<imei> - string type in decimal format indicating the IMEI. . IMEI is composed of Type Allocation Code (TAC) (8 digits), Serial Number (SNR) (6 digits) and the Check Digit (CD) (1 digit). Character set used in <imei> is as specified by command Select TE Character Set +CSCS.
	
	<imeisv> - string type in decimal format indicating the IMEISV. The 16 digits of IMEISV are composed of Type Allocation Code (TAC) (8 digits), Serial Number (SNR) (6 digits) and the software version (SVN) (2 digits). Character set used in <imeisv> is as specified by command Select TE Character Set +CSCS.
	
	<svn> - string type in decimal format indicating the current SVN which is a part of IMEISV. Character set used in <svn> is as specified by command Select TE Character Set +CSCS.

AT#CGSN=?	Test command returns the **OK** result code.
Example	AT#CGSN #CGSN: 358677008900540

#CGSN - Product Serial Number Identification	
	OK

3.4.7.1.5 *International Mobile Subscriber Identity (IMSI) - #CIMI*

#CIMI - International Mobile Subscriber Identity (IMSI)	
AT#CIMI	Execution command returns the international mobile subscriber identity, identified as the IMSI number, with command echo.
AT#CIMI=?	Test command returns the **OK** result code.
Example	AT#CIMI #CIMI: 450050209516643 OK

3.4.7.1.6 *Display PIN Counter - #PCT*

#PCT - Display PIN Counter	

AT#PCT	Execution command reports the PIN/PUK or PIN2/PUK2 input remaining attempts, depending on **+CPIN** requested password in the format:
	
	#PCT: <n>
	
	where:
	<n> - remaining attempts
	  0 - the SIM is blocked.
	  1..3 - if the device is waiting either SIM PIN or SIM PIN2 to be given.
	  1..10 - if the device is waiting either SIM PUK or SIM PUK2 to be given.

AT#PCT=?	Test command returns the OK result code.
Example	AT+CPIN? +CPIN: SIM PIN OK AT#PCT *Check PIN remained counter* #PCT: 3 OK AT+CPIN=1111 *Input incorrect PIN number* +CME ERROR: incorrect password AT#PCT #PCT: 2

3.4.7.1.7 *Software ShutDown - #SHDN*

#SHDN - Software Shutdown	
AT#SHDN	Execution command causes device detach from the network and shut down. Before definitive shut down an **OK** response is returned. **Note:** when issuing the command any previous activity terminated and the device will not respond to any further command. **Note:** to turn it on again Hardware pin ON/OFF must be tied **low**. **Note:** The maximum time to shutdown the device, completely is 25 seconds.
AT#SHDN=?	Test command returns the OK result code.

3.4.7.1.8 *Rebbots the unit - #REBOOT*

#REBOOT – Rebbots the unit	
AT#REBOOT	Execution command reboots immediately the unit. It can be used to reboot the system after a remote update of the script in order to have the new one running. **Note:** if AT#REBOOT follows an AT command that stores some parameters in NVM, it is recommended to insert a delay of at least 5 seconds before to issue AT#REBOOT, to permit the complete NVM storing. **Note:** AT#REBOOT is an obsolete AT command; please refer to AT#ENHRST to perform a module reboot.
AT#REBOOT=?	Test command returns **OK** result code.
Example	AT#REBOOT OK Module Reboots … AT#REBOOT OK AT#REBOOT=? OK
Reference	

3.4.7.1.9 *Periodic Reset - #ENHRST*

#ENHRST – Periodic Reset	
AT#ENHRST=	Set command enables/disables the unit reset after <delay> minutes.

<mod>[,<delay>]	**Parameters:** **<mod>** 0 – disables the unit reset (factory default) 1 – enables the unit reset only for one time 2 – enables the periodic unit reset **<delay>** - time interval after that the unit reboots; numeric value in minutes **Note:** the settings are saved automatically in NVM only if old or new mod is 2. Any change from 0 to 1 or from 1 to 0 is not stored in NVM **Note:** the particular case AT#ENHRST=1,0 causes the immediate module reboot. In this case if AT#ENHRST=1,0 follows an AT command that stores some parameters in NVM, it is recommended to insert a delay of at least 5 seconds before to issue AT#ENHRST=1,0, to permit the complete NVM storing.
AT#ENHRST?	Read command reports the current parameter settings for #EHNRST command in the format: #EHNRST: <mod>[,<delay>,<remainTime>] **<remainTime>** - time remaining before next reset
AT#ENHRST=?	Test command reports supported range of values for parameters <mod> and <delay>. AT#ENHRST=? #ENHRST: (0-2),(0-11000) OK
Examples	AT#ENHRST=1,60 …. Module reboots after 60 minutes … AT#ENHRST=1,0 …. Module reboots now … AT#ENHRST=2,60 …. Module reboots after 60 minutes and indefinitely after every following power on …

3.4.7.1.10　　*General Purpose Input/Output Pin Control - #GPIO*

#GPIO - General Purpose Input/Output Pin Control	
AT#GPIO=[<pin>, <mode>[,<dir>[,<save>]]]	Execution command sets the value of the general purpose output pin GPIO<pin> according to <dir> and <mode> parameter. Not all configurations for the three parameters are valid. **Parameters:** **<pin>** - GPIO pin number; supported range is from 1 to a value that depends on the hardware.

#GPIO - General Purpose Input/Output Pin Control	
	<mode> - its meaning depends on <dir> setting: 0 - if <dir>=0 – INPUT, remove any Pull-up/Pull-down - output pin cleared to 0 (Low) if <dir>=1 - OUTPUT 1 - if <dir>=0 – INPUT, if <dir>=0 – INPUT, remove any Pull-up/Pull-down - output pin set to 1 (High) if <dir>=1 - OUTPUT 2 - Reports the read value from the input pin if <dir>=0 – INPUT - Reports the read value from the input pin if <dir>=1 - OUTPUT **Note**: Reports a no meaning value if ALTERNATE FUNCTION 3 - if <dir>=0 – INPUT, enable Pull-Up 4 - if <dir>=0 – INPUT, enable Pull-Down **<dir>** - GPIO pin direction 0 - pin direction is INPUT 1 - pin direction is OUTPUT 2,3,4,5,6,7,8,9 - pin direction is Alternate Function ALT1, ALT2, ALT3, ALT4, ALT5, ALT6, ALT7,ALT8 respectively this direction. **<save>** - GPIO pin save configuration 0 – pin configuration is not saved 1 – pin configuration is saved **Note:** when <save> is omitted the configuration is stored **Note:** when <mode>=2 (and <dir> is omitted) the command reports the direction and value of pin GPIO<pin> in the format: **#GPIO:** <dir>,<stat> **where:** **<dir>** - current direction setting for the GPIO<pin> **<stat>** - logic value read from pin GPIO<pin> in the case the pin <dir> is set to input; • logic value present in output of the pin GPIO<pin> in the case the pin <dir> is currently set to output; • no meaning value for the pin GPIO<pin> in the case the pin <dir> is set to alternate function or Tristate pull down. **Note:**

#GPIO - General Purpose Input/Output Pin Control	
	While using the pins in the alternate function, the GPIO read/write access to that pin is not accessible and shall be avoided. While pins in used by ALT function it cannot be used as GPIO out or by any other ALT function until the original ALT released it or we change it to GPIO in(default state). **Note:** "ALT1" Reserved. "ALT2" value is valid for all GPIOs: alternate function is "Alarm Pin" "ALT3" value is valid for all GPIOs as "TempMon Pin" "ALT4" value is valid for all GPIOs as "AD_Det Pin" "ALT5" value is valid for all GPIOs as "AD_rep Pin" "ALT6" value is valid for all GPIOs as "FASTSHDN" "ALT7" value is valid for GPIO4 as "WKIO" "ALT8" value is valid for ALL "FRATTRIGGER"
AT#GPIO?	Read command reports the read direction and value of all GPIO pins, in the format: **#GPIO: <dir>,<stat>[<CR><LF>#GPIO: <dir>,<stat>[…]]** **where:** <dir> - as seen before <stat> - as seen before If <mode> = 3,4 the ouput format is **#GPIO:<dir>,<stat>,<mode>[<CR><LF>#GPIO:<dir>,<stat>,<mode>[…]]**
AT#GPIO=?	Test command reports the supported range of values of the command parameters. **<pin range>,<mode range>,<dir range>,<save>**
Examples	at#gpio=? #GPIO: (1-6),(0-4),(0-1),(0,1) OK at#gpio=3,1,1 // setting gpio_3 as output and value is HIGH OK at#gpio=4,1,1,1 // setting gpio_4 as output and value is HIGH an GPIO pin save configuration OK at#gpio=3,2 // report gpio_3 state #GPIO: 1,1 OK at#gpio? // read command

#GPIO - General Purpose Input/Output Pin Control	
	#GPIO: 1,1 // gpio 1 is output and output value is HIGH #GPIO: 0,0 #GPIO: 1,1 #GPIO: 0,0 #GPIO: 0,1 // gpio 5 is input and input value is HIGH #GPIO: 1,0 OK
Note	All gpio's states (mode and direction) are saved in nv memory, so the last state is restored after power cycle.

3.4.7.1.11 *Read Analog/Digital Converter Input - #ADC*

#ADC - Read Analog/Digital Converter Input	

AT#ADC=
[<adc>,<mode>
[,<dir>]]
	Execution command reads pin**<adc>** voltage, converted by ADC, and outputs it in the format:
	
	#ADC: <value>
	
	where:
	<value> - pin**<adc>** voltage, expressed in mV
	
	Parameters:
	<adc> - index of pin
	  1 - available for LE9x0 family
	  2 - available for LE9x0 family
	  3 - available for LE9x0 family
	<mode> - required action
	  2 - query ADC value
	<dir> - direction; its interpretation is currently not implemented
	  0 - no effect.
	
	Note: The command returns the last valid measure.

AT#ADC?	Read command reports all pins voltage, converted by ADC, in the format: **#ADC: <value>[<CR><LF>#ADC: <value>[…]]**
AT#ADC=?	Test command reports the supported range of values of the command parameters **<adc>**, **<mode>** and **<dir>**.

3.4.7.1.12   *Battery and Charger Status - #CBC*

#CBC- Battery And Charger Status	

AT#CBC	Execution command returns the current Battery and Charger state in the format:
	
	#CBC: \<ChargerState\>,\<BatteryVoltage\>
	
	where:
	\<ChargerState\> - battery charger state
	  0 - charger not connected
	  1 - charger connected and charging
	  2 - charger connected and charge completed
	\<BatteryVoltage\> - battery voltage in units of ten millivolts: it is the real battery voltage only if charger is not connected; if the charger is connected this value depends on the charger voltage.

AT#CBC=?	Test command returns the **OK** result code.

3.4.7.1.13   *V24 Output Pins Configuration - #V24CFG*

#V24CFG - V24 Output Pins Configuration	

AT#V24CFG=\<pin\>, \<mode\>	Set command sets the AT commands serial port interface output pins mode.
	
	Parameters:
	\<pin\> - AT commands serial port interface hardware pin:
	  0 – **DCD** (Data Carrier Detect)
	  1 – **CTS** (Clear To Send)
	  2 – **RI** (Ring Indicator)
	  3 – **DSR** (Data Set Ready)
	
	\<mode\> - AT commands serial port interface hardware pins mode:
	  0 – AT commands serial port mode: the V24 pins are controlled by the serial port device driver (default)
	  1 – GPIO mode: the V24 output pins can be managed through the **AT#V24** command
	
	Note: when **\<mode\>=1,** the V24 pins, both output and input, can be set to control an external GNSS receiver through the **AT$GPSGPIO** command.
	
	Note: changing V24 pins configuration may affect the cellular module functionality set through **AT+CFUN**.

AT#V24CFG?	Read command returns the current configuration for all the pins (both output and input) in the format: **#V24CFG: \<pin1\>,\<mode1\>[\<CR\>\<LF\>\<CR\>\<LF\>** **#V24CFG: \<pin2\>,\<mode2\>[...]]**

#V24CFG - V24 Output Pins Configuration	
	Where: **<pin*n*>** - AT command serial port interface HW pin **<mode*n*>** - AT commands serial port interface hardware pin mode
AT#V24CFG=?	Test command reports supported range of values for parameters **<pin>** and **<mode>**.

3.4.7.1.14 *V24 Output Pins Control - #V24*

#V24 - V24 Output Pins Control	
AT#V24= **<pin>[,<state>]**	Set command sets the AT commands serial port interface output pins state. **Parameters:** **<pin>** - AT commands serial port interface hardware pin: **0** - DCD (Data Carrier Detect) **1** - CTS (Clear To Send) **2** - RI (Ring Indicator) **3** - DSR (Data Set Ready) **4** - DTR (Data Terminal Ready). This is not an output **pin:** we maintain this value only for backward compatibility, but trying to set its state raises the result code"ERROR" (not yet implemented) **5** - RTS (Request To Send). This is not an output pin: we maintain this value only for backward compatibility, but trying to set its state raises the result code "ERROR" <state> - State of AT commands serial port interface output hardware pins(0,1,2,3) when pin is in GPIO mode (see #V24CFG): **0** - Low **1** - High **Note:** if <state> is omitted the command returns the actual state of the pin <pin>.
AT#V24?	Read command returns actual state for all the pins (either output and input) in the format: **#V24: <pin1>,<state1>[<CR><LF>** **#V24: <pin2>,<state2>[...]]** **Where:** **<pin*n*>** - AT command serial port interface HW pin. **<state*n*>** - AT commands serial port interface hardware pin state.
AT#V24=?	Test command reports supported range of values for parameters <pin> and <state>.

3.4.7.1.15 *Cell Monitor - #MONI*

#MONI - Cell Monitor	
AT#MONI= **<number>**	Set command sets one cell out of seven, in a neighbour of the serving cell including it, from which extract GSM/WCDMA/LTE-related information. **Parameter:** **<number>** **<GSM network>** **0..6** - it is the ordinal number of the cell, in a neighbour of the serving cell (default 0, serving cell). **7** - it is a special request to obtain GSM-related information from the whole set of seven cells in the neighbour of the serving cell. **<WCDMA network>** **0** – it is the active set **1** – it is the candidate set **2** – it is the syncronized neighbour set **3** – it is the unsyncronized neighbour set **4..6** – it is not available **7** - it is a special request to obtain WCDMA-related information from the all sets. **<LTE network>** **0** – it is the serving cell **1** – it is the intra-frequency cells **2** – it is the inter-frequency cells **3** – it is the WCDMA neighbour cells **4** – it is the GSM neighbour cells **5..6** – it is not available **7** - it is a special request to obtain LTE-related information from the all available neighbour cells.
AT#MONI	Read command reports the following GSM/WCDMA-related information for selected cell and dedicated channel (if exists). **<GSM network>** a) When extracting data for the serving cell and the network name is known the format is: **(GSM network)** **#MONI: <netname> BSIC:<bsic> RxQual:<qual> LAC:<lac> Id:<id>** **ARFCN:<arfcn> PWR:<dBm> TA:<timadv>** **(WCDMA network)** **#MONI: <netmame> PSC:<psc> RSCP:<rscp> LAC:<lac> Id:<id>** **EcIo:<ecio> UARFCN:<uarfcn> PWR:<dBm> DRX:<drx> SCR:<scr>** **URA:<ura_id>**

#MONI - Cell Monitor	
	(LTE network) **#MONI: \<netmame> RSRP:\<rsrp> RSRQ:\<rsrq> TAC:\<tac> Id:\<id> EARFCN:\<earfcn> PWR:\<dBm> DRX:\<drx>** b) When the network name is unknown, the format is: (GSM network) **#MONI: Cc:\<cc> Nc:\<nc> BSIC:\<bsic> RxQual:\<qual> LAC:\<lac> Id:\<id> ARFCN:\<arfcn> PWR:\<dBm> TA:\<timadv>** (WCDMA network) **#MONI: Cc:\<cc> Nc:\<nc> PSC:\<psc> RSCP:\<rscp> LAC:\<lac> Id:\<id> EcIo:\<ecio> UARFCN:\<uarfcn> PWR:\<dBm> DRX:\<drx> SCR:\<scr> URA:\<ura_id>** (LTE network) **#MONI: Cc:\<cc> Nc:\<nc> RSRP:\<rsrp> RSRQ:\<rsrq> TAC:\<tac> Id:\<id> EARFCN:\<earfcn> PWR:\<dBm> DRX:\<drx>** c) When extracting data for an adjacent cell, the format is: (GSM network) **#MONI: Adj Cell\<n> [LAC:\<lac> Id:\<id>] ARFCN:\<arfcn> PWR:\<dBm> dBm** (WCDMA network) **#MONI: PSC:\<psc> RSCP:\<rscp> EcIo:\<ecio> UARFCN:\<uarfcn> SCR:\<scr>** (LTE network) (LTE intra-frequency and inter-frequency cells) **#MONI: RSRP:\<rsrp> RSRQ:\<rsrq> PhysCellId:\<physicalCellId> EARFCN:\<earfcn> PWR:\<dBm>** (LTE WCDMA neighbour cells) **#MONI: PSC:\<psc> RSCP:\<rscp> EcIo:\<ecio> UARFCN:\<uarfcn> SCR:\<scr>** (LTE GSM neighbour cells) **#MONI: Adj Cell\<n> BSIC:\<bsic> ARFCN:\<arfcn> PWR:\<dBm>** **where:** **\<netname>** - name of network operator **\<cc>** - country code **\<nc>** - network operator code **\<n>** - progressive number of adjacent cell **\<bsic>** - base station identification code **\<qual>** - quality of reception 0..7 **\<lac>** - localization area code **\<id>** - cell identifier **\<physicalCellId>** - physical cell identifier **\<pid>** - physical cell identifier **\<arfcn>** - assigned radio channel

#MONI - Cell Monitor	
	<dBm> - received signal strength in dBm **<timadv>** - timing advance **<psc>** - primary ~~synchronisation~~ scrambling code **<rscp>** - Received Signal Code Power in dBm **<ecio>** - chip energy per total wideband power in dBm **<uarfcn>** - UMTS assigned radio channel **<drx>** - Discontinuous reception cycle length **<scr>** - Scrambling code **<rsrp> -** Reference Signal Received Power **<rsrq> -** Reference Signal Received Quality **<tac> -** Tracking Area Code **<earfcn> -** E-UTRA Assigned Radio Channel **<ura_id> -** UTRAN Registration Area Identity **Note:** TA: **<timadv>** reported only for the serving cell. When the last setting done is **AT#MONI=7**, then the Read command reports the above information for each of the cells in the neighbour of the serving cell, formatting them in a sequence of **<CR><LF>-terminated** strings. Currently, it is available in case of GSM network.
AT#MONI=?	Test command reports the maximum number of cells, in a neighbour of the serving cell excluding it, from which we can extract GSM/WCDMA-related information, along with the ordinal number of the current selected cell, in the format: **#MONI: (<MaxCellNo>,<CellSet>)** where: **<MaxCellNo>** - maximum number of cells in a neighbour of the serving cell and excluding it from which we can extract GSM-related informations. This value is always **6**. **<CellSet>** - the last setting done with command **#MONI**.
Note	The refresh time of the measures is preset to 3 sec. The timing advance value is meaningful only during calls or GPRS transfers active.

3.4.7.1.16 *Packet Service Network Type - #PSNT*

#PSNT – Packet Service Network Type	
AT#PSNT=<mode>	Set command enables/disables unsolicited result code for packet service network type (PSNT). **Parameter:**

#PSNT – Packet Service Network Type	
	\<mode\> 0 - disable PSNT unsolicited result code (factory default) 1 - enable PSNT unsolicited result code 2 - PSNT unsolicited result code enabled; read command reports HSUPA and HSDPA related info. **Note: \<mode\>** parameter setting is stored in NVM.
AT#PSNT?	Read command reports the \<mode\>,\<nt\> and HSUPA and HSDPA related info in the format: (\<mode\> = 2) #PSNT: \<mode\>,\<nt\>,\<is_hsupa_available\>,\<is_hsupa_used\>,\<is_hsdpa_available\>,\<is_hsdpa_used\> (\<mode\> = 0 or \<mode\> = 1) #PSNT: \<mode\>,\<nt\> **Where:** **\<mode\>** 0 - PSNT unsolicited result code disabled 1 - PSNT unsolicited result code enabled 2 - PSNT unsolicited result code enabled; read command reports HSUPA and HSDPA related info **\<nt\>** - network type 0 - GPRS network 1 - EGPRS network 2 - WCDMA network 3 - HSDPA network 4 - LTE network 5 - unknown or not registered **\<is_hsupa_available\>** - HSUPA available 0 – HSUPA is not supported by network 1 – HSUPA is supported by network **\<is_hsupa_used\>** - HSUPA used 0 – HSUPA is not in use 1 – HSUPA is in use **\<is_hsdpa_available\>** - HSDPA available 0 – HSDPA is not supported by network 1 – HSDPA is supported by network **\<is_hsdpa_used\>** - HSPA used 0 – HSDPA is not in use 1 – HSDPA is in use **Note:** when the type of network is HSPA, the indication is certainly valid during traffic, while it could be not valid in idle because it depends on network broadcast parameters.

#PSNT – Packet Service Network Type	
AT#PSNT=?	Test command returns the range of supported **<mode>**s.

3.4.7.1.17 *Read Current Network Status in 3G Network - #RFSTS*

#RFSTS – Read current network Status in 3G network	
AT#RFSTS	Read current status in the format **(GSM network)** **#RFSTS:** <PLMN>,<ARFCN>,<RSSI>,<LAC>,<RAC>,[<TXPWR>],<MM>,<RR>,<NOM>,<CID>,<IMSI>,[<NetNameAsc>],<SD>,<ABND>[CR,LF] [CR,LF] <table><tr><td>Parameter</td><td>GSM Example</td><td>description</td></tr><tr><td>PLMN</td><td>"450 05"</td><td>Country code and operator code(MCC, MNC</td></tr><tr><td>ARFCN</td><td>114</td><td>GSM Assigned Radio Channel</td></tr><tr><td>RSSI</td><td>-67</td><td>Received Signal Strength Indication</td></tr><tr><td>LAC</td><td>2011</td><td>Localization Area Code</td></tr><tr><td>RAC</td><td>11</td><td>Routing Area Code</td></tr><tr><td>TXPWR</td><td>1</td><td>Tx Power (In traffic only)</td></tr><tr><td>MM</td><td>19</td><td>Mobility Management</td></tr><tr><td>RR</td><td>0</td><td>Radio Resource</td></tr><tr><td>NOM</td><td>1</td><td>Network Operator Mode</td></tr><tr><td>CID</td><td>2825220</td><td>Cell ID</td></tr><tr><td>IMSI</td><td>"450050203619261"</td><td>International Mobile Station ID</td></tr><tr><td>NetNameAsc</td><td>"SKTelecom"</td><td>Operation Name, Quoted string type or "" if name is unknown</td></tr><tr><td>SD</td><td>3</td><td>Service Domain (0 : No Service, 1 : CS only, 2 : PS only, 3 :</td></tr><tr><td>ABND</td><td>2</td><td>Active Band (1 : GSM 850, 2 : GSM 900, 3 : DCS 1800, 4 1900)</td></tr></table> **(WCDMA network)** **#RFSTS:** <PLMN>,<UARFCN>,<PSC>,<Ec/Io>,<RSCP>,<RSSI>,<LAC>,<RAC>,[<TXPWR>],<DRX>,<MM>,<RRC>,<NOM>,[<BLER>],<CID>,<IMSI>,[<NetNameAsc>],<SD>,[<CsAccess>],[<PsAccess>],<nAST>[,<nUARFCN><nPSC>,<nEc/Io>,,,,],<ABND>[CR,LF] [CR,LF] <table><tr><td>Parameter</td><td>WCDMA Example</td><td>description</td></tr></table>

#RFSTS – Read current network Status in 3G network

PLMN	"450 05"	Country code and operator code(MCC, MNC)	
UARFCN	10737	UMTS Assigned Radio Channel	
PSC	75	Active PSC(Primary Synchronization Code)	
Ec/Io	-7.0	Active Ec/Io(chip energy per total wideband power in dBm)	
RSCP	-74	Active RSCP (Received Signal Code Power in dBm)	
RSSI	-67	Received Signal Strength Indication	
LAC	2011	Localization Area Code	
RAC	11	Routing Area Code	
TXPWR	1	Tx Power (In traffic only)	
DRX	64	Discontinuous reception cycle Length(cycle length : display using ms)	
MM	19	Mobility Management	
RRC	0	Radio Resource Control	
NOM	1	Network Operator Mode	
BLER	005	Block Error Rate (005 means 0.5 %)	
CID	2B1C04	Cell ID (IN HEX)	
IMSI	"450050203619261"	International Mobile Station ID	
NetNameAsc	"SKTelecom"	Operation Name, Quoted string type or "" if network name is unknown	
SD	3	Service Domain (0 : No Service, 1 : CS only, 2 : PS only, 3 : CS+PS)	
CsAccess	3	Circuit Switch Access (0: Normal calls only, 1: Emergency calls only, 2: No calls, 3: All calls)	
PsAccess	3	Packet Switch Access (0: Normal calls only, 1: Emergency calls only, 2: No calls, 3: All calls)	
nAST	3	Number of Active Set(Maximum 6)	
nUARFCN		UARFCN of n-th active set	
nPSC		PSC of n-th active set	
nEc/Io		Ec/Io of n-th active Set	
ABND	0	Active Band (1 : 2100 MHz, 2 : 1900 MHz, 3 : 850 MHz, 4 : 900 MHz , 5 : 1700 MHz, 6 : 800 MHz, 7 : 1800 MHz)	

(LTE network)
#RFSTS:
<PLMN>,<EARFCN>,<RSRP>,<RSSI>,<RSRQ>,<TAC>,[<TXPWR>],<DRX>,<MM>,<RRC>,<CID>,<IMSI>,[<NetNameAsc>],<SD>,<ABND>,<SINR>[CR,LF] [CR,LF]

#RFSTS – Read current network Status in 3G network				
	Parameter	LTE Example	description	
	PLMN	"262 25"	Country code and operator code(MCC, MNC)	
	EARFCN	6400	E-UTRA Assigned Radio Channel	
	RSRP	-99	Reference Signal Received Power	
	RSSI	-76	Received Signal Strength Indication	
	RSRQ	-7	Reference Signal Received Quality	
	TAC	40A5	Tracking Area Code	
	TXPWR	0	Tx Power (In traffic only)	
	DRX	64	Discontinuous reception cycle Length(cycle length : display using ms)	
	MM	19	Mobility Management	
	RRC	0	Radio Resource Control	
	CID	0000007	Cell ID	
	IMSI	"262011242110776"	International Mobile Station ID	
	NetNameAsc	" Telekom.de "	Operation Name, Quoted string type or "" if network name is unknown	
	SD	3	Service Domain (0: No Service, 1: CS only, 2: PS only, 3: CS+PS)	
	ABND	20	Active Band (1..63) 3GPP TS 36.101	
	SINR	93	Signal-to-Interface plus Noise Ratio (range 0 - 250)	
	Note: nSAT - Number of active set, Maximum is 6. **Note:** If nSAT value is 1, it means that active set number 1. Module does not display after parameters of nSAT. **Note:** TXPWR of GSM network means 1 tx burst **Note:** MM - Mobility Management States are: 0 - NULL 3 - LOCATION_UPDATE_INITIATED 5 - WAIT_FOR_OUTGOING_MM_CONNECTION 6 - CONNECTION_ACTIVE 7 - IMSI_DETACH_INITIATED 8 - PROCESS_CM_SERVICE_PROMPT 9 - WAIT_FOR_NETWORK_COMMAND 10 - LOCATION_UPDATE_REJECTED 13 - WAIT_FOR_RR_CONNECTION_LU 14 - WAIT_FOR_RR_CONNECTION_MM 15 - WAIT_FOR_RR_CONNECTION_IMSI_DETACH 17 - REESTABLISHMENT_INITIATED 18 - WAIT_FOR_RR_ACTIVE				

#RFSTS – Read current network Status in 3G network	
	19 - IDLE 20 - WAIT_FOR_ADDITIONAL_OUTGOING_MM_CONNECTION 21 - WAIT_FOR_RR_CONNECTION_REESTABLISHMENT 22 - WAIT_FOR_REESTABLISH_DECISION 23 - LOCATION_UPDATING_PENDING 25 - CONNECTION_RELEASE_NOT_ALLOWED **Note:** RR - Radio Resource States are: 0 - INACTIVE 1 - GOING_ACTIVE 2 - GOING_INACTIVE 3 - CELL_SELECTION 4 - PLMN_LIST_SEARCH 5 - IDLE 6 - CELL_RESELECTION 7 - CONNECTION_PENDING 8 - CELL_REESTABLISH 9 - DATA_TRANSFER 10 - NO_CHANNELS 11 - CONNECTION_RELEASE 12 - EARLY_CAMPED_WAIT_FOR_SI 13 - W2G_INTERRAT_HANDOVER_PROGRESS 14 - W2G_INTERRAT_RESELECTION_PROGRESS 15 - W2G_INTERRAT_CC_ORDER_PROGRESS 16 - G2W_INTERRAT_RESELECTION_PROGRESS 17 - WAIT_FOR_EARLY_PSCAN 18 - GRR 19 - G2W_INTERRAT_HANDOVER_PROGRESS 21 - W2G_SERVICE_REDIRECTION_IN_PROGRESS 22 - RESET 29 - FEMTO 30 - X2G_RESEL 31 - X2G_RESEL_ABORTED 32 - X2G_REDIR 33 - G2X_REDIR 34 - X2G_CGI 35 - X2G_CCO_FAILED 36 - X2G_CCO_ABORTED 37 - X2G_CCO_FAILED_ABORTED 38 - RR_INVALID **Note:** RRC (WCDMA) - Radio Resource Control States are: 0 - RRC_STATE_DISCONNECTED

#RFSTS – Read current network Status in 3G network	
	1 - RRC_STATE_CONNECTING
	2 - RRC_STATE_CELL_FACH
	3 - RRC_STATE_CELL_DCH
	4 - RRC_STATE_CELL_PCH
	5 - RRC_STATE_URA_PCH
	6 - RRC_STATE_WILDCARD
	7 - RRC_INVALID
	Note: RRC (LTE) - Radio Resource Control States are:
	0 - RRC_IDLE
	1 - RRC_CONNECTED

3.4.7.1.18   *Temperature Monitor- #TEMPMON*

#TEMPMON - Temperature Monitor	
AT#TEMPMON **=** **<mod>[,<urcmo** **de>[,<action>[,<** **GPIO>]]]**	Set command sets the behavior of the module internal temperature monitor. **Parameters:** **<mod>** 0 - sets the command parameters. 1 - triggers the measurement of the module internal temperature, reporting the result in the format: **#TEMPMEAS: <level>,<value>** **where:** **<level>** - threshold level -2 - Extreme temperature lower bound. -1 - Operating temperature lower bound. 0 - normal temperature. 1 - Operating temperature upper bound. 2 - Extreme temperature upper bound. (see Note 1) **<value> -** actual temperature expressed in Celsius degrees. Setting of the following optional parameters has meaning only if <mod>=0: **<urcmode>** - URC presentation mode. (Default 1) 0 - It disables the presentation of the temperature monitor URC. 1 - It enables the presentation of the temperature monitor URC, whenever the module internal temperature reaches either operating or extreme levels. The unsolicited message is in the format: **#TEMPMEAS: <level>,<value>**

	where: **<level>** and **<value>** are as before. **<action>** - sum of integers, each representing the action to be done whenever the module internal temperature reaches either operating or extreme levels (default is 1). 0 - (00) - No action. 1 - (01) - Activating of thermal mitigation according to thermal configuration file. 2 - (10) - Output pin <GPIO> is tied HIGH when operating temperature bounds are reached; when the temperature is back to normal the output pin <GPIO> is tied LOW. If this <action> is required, it is mandatory to set the <GPIO> parameter too. 3- (11) - This value contains <action=1> and <action=2> i.e. activate thermal mitigation and a GPIO indication. If this <action> is required, it is mandatory to set the <GPIO> parameter too. **<GPIO>** - GPIO number. Valid range is any TGPIO pin as described in #GPIO command. This parameter is needed and required only if <action>=2 or 3 are enabled.
AT#TEMPMON?	Read command reports the current parameter settings for #TEMPMON command in the format: **#TEMPMON: <urcmode>,<action> [,<GPIO>]**
AT#TEMPMON=?	Test command reports the supported range of values for parameters **<mod>, <urcmode>, <action>,** and **<GPIO>**
Notes	1. Thresholds levels are defined in #TEMPCFG command. See there for detailed description on thermal mitigation configuration. 2. Last <action> setting is saved in the 'config.ini' file ('mitigate'/'none mitigate'), and in the NVM ('gpio indication'/'none gpio indication'). 3. Last <GPIO> is saved in the NVM. 4. Thermal mitigation is disabled automatically when using laboratory test SIM.

3.4.7.1.19   *Temperature Monitor CONFIGURATION - #TEMPCFG*

#TEMPCFG – Temperature Monitor Configuration	
AT#TEMPCFG= <etlz_clr>,<etlz>,<etlz_act_in>,<otlz_clr>,<otlz>,<otlz_act_in>,<otnz_clr>,<otnz>,<otnz_act_in>,<otuz_clr>,<otuz>,<otuz_act_in>,<etuz_clr>,<etuz>,<etuz_act_in>	Set command sets the Temperature zones used in the #TEMPMON command. **Parameters**: **<etlz_clr>**: Extreme low zone temperature threshold clear. Has only one valid value: -273°C. see notes **<etlz>**: Extreme low zone temperature threshold. Default value -33°C. **<etlz_act_in>**: Extreme low zone action info. Default value 0. **<otlz_clr>**: Operate low zone temperature threshold clear. Default value -35°C.

	\<otlz\>: Operate low zone temperature threshold. Default value -28°C. **\<otlz_act_in \>**: Operate low zone action info. Default value 0. \<**otnz_clr\>**: Operate normal zone temperature threshold clear**.** Default value -30°C. **\<otnz\>**: Operate normal zone temperature threshold**.** Default value 95°C. **\<otnz_act_in\>**: Operate normal zone action info. Default value 0. \<**otuz_clr\>**: Operate up zone temperature threshold clear. Default value 93°C. **\<otuz\>**: Operate up zone temperature threshold. Default value 100°C. **\<otuz_act_in\>**: Operate up zone action info. Default value 3. \<**etuz_clr\>**: Extreme up zone temperature threshold clear. Default value 98°C. **\<etuz\>**: Extreme up zone temperature threshold. Has only one valid value: 528°C. see notes **\<etuz_act_in\>**: Extreme up zone action info. Default value 3. See notes for detailed description of thermal mitigation configuration.
AT#TEMPCFG?	Read command reports the current parameter setting for #TEMPCFG command in the format: #TEMPCFG: **\<etlz_clr\>,\<etlz\>,\<etlz_act_in\>,\<otlz_clr\>,\<otlz\>,\<otlz_act_in\>,\<otnz_clr\>,\<otnz\>,\<otnz_act_in\>,\<otuz_clr\>,\<otuz\>,\<otuz_act_in\>,\<etuz_clr\>,\<etuz\>,\<etuz_act_in\>**
AT#TEMPCFG =?	Test command reports the supported range values for parameters \<x_**clr**\>,\<x\>,\<x_**action_info**\>. Where "x" is substitute for **"etlz"**, **"otlz"**, **"otnz"**, **"otuz"**, **"etuz"**. Values are: #TEMPCFG: (-40-100),(-40-100),(0-5)

Notes:
After setting new values, there must be an execution of power cycle or #REBOOT command in order for the mitigation algorithm to operate with them.
Thermal mitigation mechanism works like this:
• The whole temperature scale is divided into 5 states (zones).
• Each measured temperature should be belonging to a particular state called the "**current state**".
  State is defined by the following fields:
-    "**thresholds**" – upper temperature boundary of the state. Values are in °C.

- "**thresholds_clr**" – lower temperature boundary of the state. Values are in °C.
- "**actions**" – indicator that indicates if an action should be taken or not in the "**current state**".

Values are: "**none**"/"**mitigate**".

"**action_info**" – thermal mitigation action type that should be taken care if "**actions**" field is "**mitigate**".

Values are:

0 - No mitigation action is required.
1 - Mitigation action - data throttling (reducing uplink baud rate).
2 - Mitigation action - TX backoff (reducing MTPL - Max Tx Power Limit).
3 - Emergency Calls Only.
4 - RF OFF. RX and TX circuits automatically disabled (using +CFUN=4).
5- Automatic shutdown. Module is powered off.

It is **prohibited** for user to set '**action info**' equal to '**5**'in '**operate normal zone**'. This comes in order to prevent setting 'Automatic shutdown' by mistake to the normal operating temperature range.
"**+CME ERROR: operation not supported**" error will be received as a response.

Here is the graph that illustrates the temperature states and their limits:
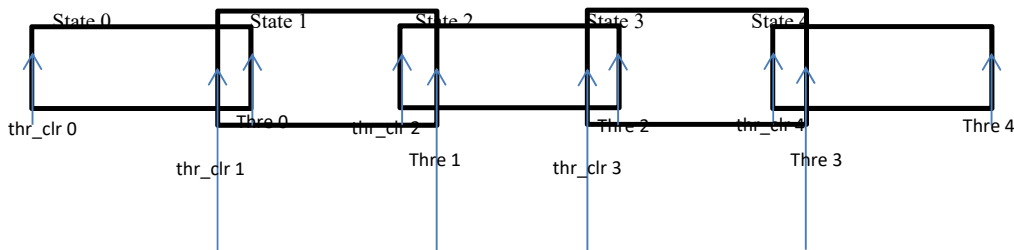
When temperature exceeds the "**current state**" "**threshold**", the thermal mitigation algorithm searches the next state when this temperature is **lower** than "**threshold**". After it finds it, the "**current state**" is updated to that "**state**" and then it checks whether "**action**" is "**mitigate**". If yes, then it activates the mitigation according to the "**action info**" of the "**current state**".

When temperature decreases below "**threshold_clr**" then it does the same algorithm as above, but in the opposite direction. It searches the next state when this temperature is **greater** than "**threshold_clr** ", updates the "**current state**" to that state, and activates mitigation as described above.

There are 2 rules in that states definition should follow:
1. Overlap between 2 adjacent states of at least 2 deg, i.e. ("thre state(x)" – "thre_clr state(x+1)") >= 2
2. Every state shall have "free" temperature range which has no part in any overlap range. This range should be at least 2 deg, i.e. ("thre_clr(x+2)" – "thre(x)") >= 2 .

Rule '1' comes to ensure hysteresis in the transition between two states.
Rule '2' comes to ensure a minimum range for a stable state.

State 0 is **'Extreme low zone'**.
State 1 is **'Operate low zone'**.
State 2 is **'Operate normal zone'**.

State 3 is '**Operate up zone**'.
State 4 is '**Extreme up zone**'.

etlz_clr – Extreme low zone threshold clear is enforced to have value of '-273'. Module doesn't operate in such temperature, but this value is logically set in order to define clearly 'thermal state' to temperatures below -40 deg.

etuz – Extreme up zone threshold is enforced to have value of '528'. Module doesn't operate in such temperature, but this value is logically set in order to define clearly 'thermal state' to temperatures above 100 deg.

"**#TEMPMON**" set command changes field "**actions**" to "**mitigate**" or "**none**" to all zones.

All above parameters are saved in a configuration file in the module file system.

Examples:
AT#TEMPCFG= -273,-33,3,-35,-28,2,-30,80,0,78,90,3,88,528,3
OK
Explanation:

zone	Thr_clr	Thr	Action info
'**Extreme low zone**'	-273	-33	3 – emergency call only
'**Operate low zone**'	-35	-28	2 – TX backoff
'**Operate normal zone**'	-30	80	0 – no mitigation
'**Operate up zone**'	78	90	3 - emergency call only
'**Extreme up zone**'	88	528	3 - emergency call only

All zones have hysteresis and free temperature range.

AT#TEMPCFG=-273,-33,3,-35,-28,2,-30,**80**,0,**79**,90,3,88,528,3
+CME ERROR: operation not supported
Explanation:

zone	Thr_clr	Thr	Action info
'**Extreme low zone**'	-273	-33	3 – emergency call only
'**Operate low zone**'	-35	-28	2 – TX backoff
'**Operate normal zone**'	-30	**80**	0 – no mitigation
'**Operate up zone**'	**79**	90	3 - emergency call only
'**Extreme up zone**'	88	528	3 - emergency call only

('**Thr**' of '**Operate normal zone**') - ('**Thr_clr**' of '**Operate up zone**') = 1 < 2
Rule 1 was broken - Hysteresis is lower than 2 deg.

AT#TEMPCFG=-273,-33,3,-35,-28,2,-30,**80**,0,78,90,3,**81**,528,3
+CME ERROR: operation not supported
Explanation:

zone	Thr_clr	Thr	Action info
'**Extreme low zone**'	-273	-33	3 – emergency call only
'**Operate low zone**'	-35	-28	2 – TX backoff
'**Operate normal zone**'	-30	**80**	0 – no mitigation
'**Operate up zone**'	78	90	3 - emergency call only

| 'Extreme up zone' | 81 | 528 | 3 - emergency call only |

('**Thr_clr**' of '**Extreme up zone**') - ('**Thr**' of '**Operate normal zone**') = 1 < 2
Rule 2 was broken - free temperature range is lower than 2 deg.

3.4.7.1.20   *Wake from Alarm Mode - #WAKE*

#WAKE - Wake From Alarm Mode	
AT#WAKE= [<opmode>]	Execution command stops any eventually present alarm activity and, if the module is in **alarm mode**, it exits the **alarm mode** and enters the **normal operating mode**. **Parameter:** **<opmode>** - operating mode 0 - normal operating mode; the module exits the **alarm mode**, enters the **normal operating mode**, any alarm activity is stopped (e.g. alarm tone playing) and an **OK** result code is returned. **Note:** The "alarm mode" is indicated by hardware pin **CTS** to the **ON** status and **DSR** to the **OFF** status, while the "power saving" status is indicated by a **CTS - OFF** ,**DSR - OFF** and USB_VBUS – OFF status. The normal operating status is indicated by **DSR – ON or USB_VBUS – ON status**. **Note:** during the **alarm mode** the device will not make any network scan and will not register to any network and therefore is not able to dial or receive any call or SM, the only commands that can be issued to the MODULE in this state are the **#WAKE** and **#SHDN**, every other command must not be issued during this state.
AT#WAKE?	Read command returns the **operating status** of the device in the format: **#WAKE: <status>** **where:** **<status>** 0 - Normal operating mode 1 - Alarm mode or normal operating mode with some alarm activity.
AT#WAKE=?	Test command returns the OK result code.

3.4.7.1.21   *AlARM PIN - #ALARMPIN*

#ALARMPIN – Alarm Pin	
AT#ALARMPIN= <pin>	Set command sets the GPIO pin for the ALARM pin **Parameters:** **<pin>** defines which GPIO shall be used as ALARM pin. For the <pin> actual range, check the "Hardware User Guide". Default value is 0, which means no ALARM pin set. **Note:** the setting is saved in NVM **Note:** ALARM pin function of a GPIO corresponds to ALT2 function of the GPIO, So it can be also set through AT#GPIO command, ALT2 function.
AT#ALARMPIN?	Read command returns the current parameter settings for #ALARMPIN command in the format: #ALARMPIN: <pin>
AT#ALARMPIN= ?	Test command reports the supported range of values for parameter <pin>.

3.4.7.1.22   *Serving Cell Information - #SERVINFO*

#SERVINFO - Serving Cell Information	
AT#SERVINFO	Execution command reports information about serving cell, in the format: **(GSM network)** **#SERVINFO:<BARFCN>,<dBM>,[<NetNameAsc>],<NetCode>,<BSIC>,<LAC>,<TA>,<GPRS>[,[<PB-ARFCN>],[<NOM>],<RAC>,[PAT]]** **(WCDMA network)** **#SERVINFO:<UARFCN>,<dBM>,[<NetNameAsc>],<NetCode>,<PSC>,<LAC>,<DRX>,<SD>,<RSCP>,<NOM>,<RAC>,<URA>** **(LTE network)** **#SERVINFO:<EARFCN>,<dBM>,[<NetNameAsc>],<NetCode>,<PhysicalCellId>,<TAC>,<DRX>,<SD>,<RSRP>** **where:** **<B-ARFCN>** - BCCH ARFCN of the serving cell **<dBM>** - received signal strength in dBm **<NetNameAsc>** - operator name, quoted string type or "" if nerwork name is unknown. **<NetCode>** - country code and operator code, hexadecimal representation **<BSIC>** - Base Station Identification Code **<LAC>** - Localization Area Code **<TA>** - Time Advance: it's available only if a GSM or GPRS is

#SERVINFO - Serving Cell Information	

	running
	<GPRS> - GPRS supported in the cell
	    0 - not supported
	    1 - supported
	
	The following informations will be present only if GPRS is supported in the cell
	
	<PB-ARFCN> - Not supported by 3GPP. PBCCH ARFCN of the serving cell; it'll be printed only if PBCCH is supported by the cell, otherwise the label **"hopping"** will be printed
	<NOM> - Network Operation Mode
	.."I"
	 "II"
	.."III"
	<RAC> - Routing Area Color Code
	<PAT> - Priority Access Threshold
	..0
	..3..6
	<UARFCN> - UMTS ARFCN of the serving cell
	<PSC> - Primary ~~Synchronisation~~ Scrambling Code
	<DRX> - Discontinuous reception cycle length
	<SD> - Service Domain
	 0 – No Service
	 1 – CS Only
	 2 – PS Only
	 3 – CS & PS
	<RSCP> - Received Signal Code Power in dBm
	<EARFCN> - E-UTRA Assigned Radio Channel
	<PhysicalCellId> - Physical Cell ID
	<TAC> - Tracking Area Code
	<RSRP> - **Reference** Signal Received Power
	<URA> - UTRAN Registration Area Identity

3.4.7.1.23 *Query SIM Status - #QSS*

#QSS - Query SIM Status	
AT#QSS= [**<mode>**]	Set command enables/disables the Query SIM Status unsolicited indication in the ME. **Parameter**: **<mode>** - type of notification 0 - disabled (factory default); it's possible only to query the current SIM status through Read command **AT#QSS?**

#QSS - Query SIM Status	

	1 - enabled; the ME informs at every SIM status change through the following basic unsolicited indication:
	
	#QSS: <status>
	
	where:
	<status> - current SIM status
	  0 - SIM NOT INSERTED
	  1 - SIM INSERTED
	
	2 - enabled; the ME informs at every SIM status change through the following unsolicited indication:
	
	#QSS: <status>
	
	where:
	<status> - current SIM status
	  0 - SIM NOT INSERTED
	  1 - SIM INSERTED
	  2 - SIM INSERTED and PIN UNLOCKED
	  3 - SIM INSERTED and READY (SMS and Phonebook access are possible).

AT#QSS?	Read command reports whether the unsolicited indication **#QSS** is currently enabled or not, along with the SIM status, in the format: **#QSS: <mode>,<status>** (**<mode>** and **<status>** are described above)
AT#QSS=?	Test command returns the supported range of values for parameter **<mode>**.

3.4.7.1.24 *Network Timezone - #NITZ*

#NITZ - Network Timezone	
AT#NITZ= **[<val>** **[,<mode>]]**	Set command enables/disables (a) automatic date/time updating, (b) Full Network Name applying and (c) **#NITZ** URC; moreover it permits to change the **#NITZ** URC format. Date and time information can be sent by the network after GSM registration or after GPRS attach. **Parameters:** **<val>** 0 - disables (a) automatic data/time updating, (b) Full Network Name applying and (c) **#NITZ** URC; moreover it sets the **#NITZ** URC *'basic'* format (see **<datetime>** below)(factory default)

#NITZ - Network Timezone	
	1..15 - as a sum of: 1 - enables automatic date/time updating 2 - enables Full Network Name applying (not supported by LE) 4 - it sets the **#NITZ** URC *'extended'* format (see **<datetime>** below) 8 - it sets the **#NITZ** URC *'extended'* format with Daylight Saving Time(DST) support (see **<datetime>** below) **<mode>** 0 - disables **#NITZ** URC (factory default) 1 - enables **#NITZ** URC; after date and time updating the following unsolicited indication is sent: **#NITZ: <datetime>** **where:** **<datetime>** - string whose format depends on subparameter **<val>** **"yy/MM/dd,hh:mm:ss"** - *'basic'* format, if **<val>** is in (0..3) **"yy/MM/dd,hh:mm:ss±zz"** - *'extended'* format, if **<val>** is in (4..7) **"yy/MM/dd,hh:mm:ss±zz,d"** - *'extended'* format with DST support, if **<val>** is in (8..15) **where:** **yy** - year **MM** - month (in digits) **dd** - day **hh** - hour **mm** - minute **ss** - second **zz** - time zone (indicates the difference, expressed in quarter of an hour, between the local time and GMT; two last digits are mandatory, range is -47..+48) **d** – Number of hours added to the local TZ because of Daylight Saving Time (summertime) adjustment: range is 0-2. **Note:** If the DST information isn't sent by the network, then the **<datetime>** parameter has the format **"yy/MM/dd,hh:mm:ss±zz"**
AT#NITZ?	Read command reports whether (a) automatic date/time updating, (b) Full Network Name applying, (c) **#NITZ** URC (as well as its format) are currently enabled or not, in the format: **#NITZ: <val>,<mode>**
AT#NITZ=?	Test command returns supported values of parameters **<val>** and **<mode>**.

3.4.7.1.25   *Clock Mode - #CCLKMODE*

#CCLKMODE – Clock Mode	
AT#CCLKMODE= <mode>	Set command enables the local time or the UTC time in AT+CCLK and AT#CCLK commands and in #NITZ URC

#CCLKMODE – Clock Mode	

	Parameter:
	<mode> - time and date mode
	  0 - Local time + local time zone offset (default)
	  1 – UTC time + local time zone offset
	
	Note: the setting is saved automatically in NVM.

AT#CCLKMODE?	Read command reports whether the local time or the UTC time is enabled, in the format: **#CCLKMODE: <mode>** (**<mode>** described above)
AT#CCLKMODE=?	Test command reports the supported range of values for parameter <**mode**>
Example:	at#cclkmode? #CCLKMODE: 0 OK #NITZ: 13/03/05,15:20:33+04,0 at+cclk? +CCLK: "13/03/05,15:20:37+04" OK at#cclkmode=1 OK at+cclk? +CCLK: "13/03/05,14:20:45+04" OK at#cclkmode? #CCLKMODE: 1 OK #NITZ: 13/03/05,14:20:53+04,0 at+cclk? +CCLK: "13/03/05,14:20:55+04" OK at#cclkmode=0 OK at+cclk? +CCLK: "13/03/05,15:20:59+04" OK

3.4.7.1.26   *Calculate and update date and time - #NTP*

#NTP – Calculate and update date and time	

AT#NTP=
<NTPaddr>,
<NTPPort>,
<update_module_clock>,
<timeout>[,<TimeZone>]
	This command permits to calculate and update date and time through NTP protocol(RFC2030), sending a request to a NTP server.
	
	Parameters:
	
	<NTPaddr> - address of the NTP server, string type. This parameter can be either:
	  - any valid IP address in the format: "xxx.xxx.xxx.xxx"
	  - any host name to be solved with a DNS query
	
	<NTPPort> - NTP server port to contact
	 1..65535
	
	<update_module_clock>
	 0 - no update module clock
	 1 – update module clock
	
	<timeout> - waiting timeout for server response in seconds
	1..10
	
	<TimeZone> - Time Zone (indicates the difference, expressed in quarter of an hour, between the local time and GMT), range is -47..+48; default is 0.
	
	Note: the Time Zone is applied directly in the Date and Time received by the NTP Server, that is, by definition, GMT+0

AT#NTP=?	Test command reports the supported range of values for parameters **<NTPaddr>,<NTPPort>,<update_module_clock>, <timeout>** and **<TimeZone>**
Example	at#ntp="ntp1.inrim.it",123,1,2,4 #NTP: 12/01/27,14:42:38+04 OK …. at+cclk? +CCLK: "12/01/27,14:42:39+04" OK

3.4.7.1.27   *Alarm Management - +CALA*

+CALA - Alarm Management	
AT+CALA=<time>[,<n>[,<type>[,<text>[,<recurr>[,<silent>]]]]]	Set command stores in the internal Real Time Clock an alarm time with respective settings. It is possible to set up a recurrent alarm for one or more days in the week. • Currently just one alarm can be set. • DO NOTE! Alarms are not supported after disconnecting from power. Coin cell are supported. In case of a power cut, alarm will be deleted and needs to be re-set. When the RTC time reaches the alarm time then the alarm starts, the behavior of the MODULE depends upon the setting <type> and if the device was already ON at the moment when the alarm time had come. **Parameters:** **<time>** - current alarm time as quoted string in the same format as defined for +CCLK command (i.e. "yy/MM/dd,hh:mm:ss±zz"), unless the **<recurr>** parameter is used: in this case <time> must not contain a date (i.e."hh:mm:ss±zz") **<n>** - index of the alarm 0 - The only value supported is 0. **<type>** - alarm behaviour type **0 -** reserved for other equipment use. **1 -** the MODULE simply wakes up fully operative as if the ON/OFF button had been pressed. If the device is already ON at the alarm time, then it does nothing.(Default) **2 -** The MODULE wakes up in "alarm mode" if at the alarm time it was off, otherwise it remains fully operative. In both cases the MODULE issues an unsolicited code every 3s: **+CALA: <text>** Where <text> is the +CALA optional parameter previously set. The device keeps on sending the unsolicited code every 3s until a #WAKE or #SHDN command is received or a 90 seconds timer expires. If the device is in "alarm mode" and it does not receive the #WAKE command within 90s then it shuts down. (default) **3 -** The MODULE wakes up in "alarm mode" if at the alarm time it was off, otherwise it remains fully operative. In both cases the MODULE starts playing the alarm tone on the selected path for the ringer (see command #SRP). The device keeps on playing the alarm tone until a #WAKE or #SHDN command is received or a 90 s time-out occurs. If the device is in "alarm mode" and it does not receive the #WAKE command within 90s then it shuts down. If alarm expires during a call alarm sound will stop when the call is disconnected. **4 -** The MODULE wakes up in "alarm mode" if at the alarm time it was off, otherwise it remains fully operative. In both cases the MODULE brings the alarmpin high, provided that one has been set (using #ALARMPIN or #GPIO), and keeps it in this state until a #WAKE or #SHDN command is received or a 90 seconds timer expires. If

	the device is in "alarm mode" and it does not receive the #WAKE command within 90s then it shuts down. 5 - The MODULE will make both the actions as for type=2 and <type>=3. 6 - The MODULE will make both the actions as for type=2 and <type>=4. 7 - The MODULE will make both the actions as for type=3 and <type>=4. **8 -** The MODULE wakes up in "alarm mode" if at the alarm time it was off, otherwise it remains fully operative. In both cases the MODULE sets High the RI output pin. The RI output pin remains High until next #WAKE issue or until a 90s timer expires. If the device is in "alarm mode" and it does not receive the #WAKE command within 90s. After that it shuts down. **<text>** - unsolicited alarm code text string. It has meaning only if <type> is equal to 2 or 5 or 6. **<recurr>** - string type value indicating day of week for the alarm in one of the following formats: "<1..7>[,<1..7>[, …]]" - it sets a recurrent alarm for one or more days in the week; the digits 1 to 7 corresponds to the days in the week (Monday is 1). "0" - it sets a recurrent alarm for all days in the week. **<silent>** - integer type indicating if the alarm is silent or not. **0 - the alarm will not be silent;** **1 -** the alarm will be silent. **Note:** a special form of the Set command, +CALA="", deletes an alarm in the ME **Note**: The "alarm mode" is indicated by hardware pin CTS to the ON status and DSR to the OFF status, while the "power saving" status is indicated by a CTS - OFF ,DSR - OFF and USB_VBUS – OFF status. The normal operating status is indicated by DSR – ON or USB_VBUS – ON status. During the "alarm mode" the device will not make any network scan and will not register to any network and therefore is not able to dial or receive any call or SMS. The only commands that can be issued to the MODULE in this state are the #WAKE and #SHDN, every other command must not be issued during this state.
AT+CALA?	Read command returns the list of current active alarm settings in the ME, in the format: **[+CALA: <time>,<n>,<type>,[<text>],<recurr>,<silent>]** **Note: on READ command <time> does not include the time zone.**
AT+CALA=?	Test command returns the list of supported index values (currently just 0), alarm types, maximum length of the text to be displayed, maximum length of <recurr> and supported <silent>s, in the format: **+CALA: (list of supported <n>s),(list of supported <type>s),<tlength>,<rlength>,(list of supported <silent>s)**

Example	AT+CALA="02/09/07,23:30:00+00" OK
Reference	3gpp TS 27.007

3.4.7.1.28 Delete Alarm - +CALD

+CALD - Delete alarm	
AT+CALD=<n>	*Execution command deletes an alarm in the ME* *Parameter:* *<n> - alarm index* *0*
AT+CALD=?	*Test command reports the range of supported values for <n> parameter.*
Reference	*3G TS 27.007*

3.4.7.1.29 Generic SIM access - +CSIM

+CSIM – Generic SIM access	
AT+CSIM=<lock>	Between two successive +CSIM command the SIM-ME interface must be locked to avoid commands can modify wrong SIM file. The locking and unlocking of the SIM-ME interface must be done explicitly respectively at the beginning and at the end of the +CSIM commands sequence. **Parameters**: <**lock**>=1 locking of the interface <**lock**>=0 unlocking of the interface In case that TE application does not use the unlock command in a certain timeout value, ME releases the locking.
AT+CSIM=<length>, <command>	The ME shall send the <**command**> as it is to the SIM/UICC. As response to the command, ME sends back the actual SIM/UICC <**response**> to the TA as it is. **Parameters**: <**lenght**>: number of the characters that are sent to TE in <command> or <response> (two times the actual length of the command or response) <**command**>: command passed on by the ME to the SIM/UICC in the format as described in GSM TS 11.11 or 3G TS 31.101 (hexadecimal character format) The response of the command is in the format: **+CSIM: <length>,<response>** where:

+CSIM – Generic SIM access	
	<**response**> : response to the command passed on by the SIM to the ME in the format as described in GSM TS 11.11 or 3G TS 31.101 (hexadecimal character format). Error case: *+CME ERROR: <err>* possible <err> values (numeric format followed by verbose format): 3 operation not allowed *(operation mode is not allowed by the ME, wrong interface lock/unlock status)* 4 operation not supported *(wrong format or parameters of the command)* 13 SIM failure *(SIM no response)*
AT+CSIM=?	Test command returns the **OK** result code.
Example	*Lock SIM interface* **AT+CSIM=1** **OK** **2G SIM (TS 11.11):** AT#ENAUSIM? +ENAUSIM: 0 OK *STATUS* AT+CSIM=10,A0F2000016 +CSIM:48,"000002A87F2002000000000000099300220800838A838A9000" OK *SELECT EF 6F07* AT+CSIM=14,A0A40000026F07 +CSIM: 4,"9F0F" OK *GET RESPONSE* AT+CSIM=10,A0C000000F +CSIM: 34,"000000096F0704001A001A010200009000" OK *SELECT EF 6F30* AT+CSIM=14,A0A40000026F30

+CSIM – Generic SIM access	
	+CSIM: 4,"9F0F" OK *READ BINARY* AT+CSIM=10,A0B00000FC +CSIM:508,"FFFFFF13008313009013005413003013006513003813008013018013000113110913013013009813007713005913004313008113009513014013002313 0016330420130041FFFFFFFFFFFF21436542F41922F28822F201FFF9000" OK **3G UICC (3G TS 31.101):** AT#ENAUSIM? +ENAUSIM: 1 OK *STATUS* AT+CSIM=10,A0F2000016 +CME ERROR: operation not supported *STATUS* AT+CSIM=10,80F2000016 +CSIM:48,"623F8202782183027FF08410A0000000871002FFFFFF9000" OK *SELECT EF 6F07 No Data Returned* AT+CSIM=18,00A4080C047F206F07 +CSIM: 4,"9000" OK *SELECT EF 6F30 Return FCP Template* AT+CSIM=18,00A40804047F206F30 +CSIM: 4,"6120" OK

+CSIM – Generic SIM access	
	GET RESPONSE AT+CSIM=10,00C0000020 +CSIM:68,"621E8202412183026F30A506C00140DE01008A01058B036F06048002006988009000" OK *READ BINARY* AT+CSIM=10,00B0000069 +CSIM:214,"02F81012F47022F83082F63082F64022F60192F31412F60313006132F40102F20162 F21032F23002F60182F41012F91042F41902F46102F40242F22092F52072F22062F03062F86032F0 1032F11042F01032F80217F60127F42027F43027F44027F24337F62037F0209000" OK **Unlock SIM interface** AT+CSIM=0 OK
Note	After the locking of the SIM-ME interface (AT+CSIM=1) the SIM will be accessible only by AT+CSIM commands (#QSS: 0). The GSM and GPRS services will be automatically deregistered to avoid the TE commands alter the GSM application. They will be automatically reconditioned after the unlocking of the SIM-ME interface. After the unlocking of the SIM-ME interface if PIN is required it will be necessary to enter it another time.

3.4.7.1.30   Select Band - #BND

#BND - Select Band	

AT#BND=
<band> [,<UMTS band>[,<LTE band>]]
	Set command selects the current GSM, UMTS and LTE bands.
	
	Parameter
	<band>:
	  0 - GSM 900MHz + DCS 1800MHz (default value)
	  1 - GSM 900MHz + PCS 1900MHz
	  2 - GSM 850MHz + DCS 1800MHz
	  3 - GSM 850MHz + PCS 1900MHz
	  4 - GSM 900 MHz + DCS 1800 MHz + PCS 1900 MHz
	  5 - GSM 900 MHz + DCS 1800 MHz + PCS 1900 MHz + GSM 850 MHz

#BND - Select Band	
	<UMTS band>: 0 - 1900 / 2100MHz(FDD I) (default value depending on product) 1 - 1900MHz(FDD II) (default value depending on product) 2 - 850MHz(FDD V) 3 - 2100MHz(FDD I) + 1900MHz(FDD II) + 850MHz(FDD V) 4 - 1900MHz(FDD II) + 850MHz(FDD V) 5 - 900MHz(FDD VIII) 6 - 2100MHz(FDD I) + 900MHz(FDD VIII) 7 – 1700/ 2100MHz(FDD IV, AWS) **<LTE band>** values in the range 1 – 4294967295 as a sum of: 1 - B1 2 - B2 4 - B3 8 - B4 … i - B(2exp(i-1)) … 2147483648 - B32 **Note**: not all products support all the values of parameter **<band>**: please refer to test command to find the supported range of values. **Note**: not all products support all the values of parameter **<UMTS band>**: please refer to test command to find the supported range of values. **Note**: not all products support all the values of parameter **<LTE band>**: please refer to test command to find the supported range of values (maximum value is the sum representation of supported bands). **Note**: for 4G only product use fixed unused value 0 for **<band>** and **<UMTS band>** parameters. **Note**: for 4G/3G only product use fixed unused value 0 for **<band>** parameter. **Note**: for 4G/2G only product use fixed unused value 0 for **<UMTS band>** parameter.
AT#BND?	Read command returns the current selected band in the format: **#BND: <band>,<UMTS band>,<LTE band>**
AT#BND=?	Test command returns the supported range of values of parameters **<band>,** **<UMTS band>** and **<LTE band>**.

3.4.7.1.31   *Skip Escape Sequence - #SKIPESC*

#SKIPESC - Skip Escape Sequence	

AT#SKIPESC= [<mode>]	Set command enables/disables skipping the escape sequence +++ while transmitting during a data connection.
	
	Parameter:
	<mode>
	  0 - does not skip the escape sequence; its transmission is enabled (factory default).
	  1 - skips the escape sequence; its transmission not enabled.
	
	Note: in case of an FTP connection, the escape sequence not transmitted, regardless of the command setting.

AT#SKIPESC?	Read command reports whether escape sequence skipping is currently enabled or not, in the format: **#SKIPESC: <mode>**
AT#SKIPESC=?	Test command reports supported range of values for parameter **<mode>**.

3.4.7.1.32   *PPP-GPRS Connection Authentication Type - #GAUTH*

#GAUTH - PPP-GPRS Connection Authentication Type	

AT#GAUTH= [<type>]	Set command sets the authentication type either for PPP-GPRS connections.
	
	Parameter:
	<type>
	  0 - no authentication
	  1 - PAP authentication
	  2 - CHAP authentication
	  3 – AUTO authentication (PAP or CHAP or no authentication according to host application), factory default
	
	Note: if the settings on the server side (the host application) of the PPP are not compatible with the AT#GAUTH setting, then the PDP Context Activation will use no authentication.

AT#GAUTH?	Read command reports the current authentication type, in the format: **#GAUTH: <type>**
AT#GAUTH=?	Test command returns the range of supported values for parameter **<type>**.

3.4.7.1.33   *SIM Detection Mode-#SIMDET*

#SIMDET - SIM Detection Mode	

AT#SIMDET=
\<mode\>
	Set command specifies the SIM Detection mode.
	
	Parameter:
	\<mode\> - SIM Detection mode
	  0 - ignore SIMIN pin and simulate the status
	    "SIM Not Inserted"
	  1 - ignore SIMIN pin and simulate the status
	    "SIM Inserted"
	  2 - automatic SIM detection through SIMIN Pin (default)

AT#SIMDET?	Read command returns the currently selected Sim Detection Mode in the format:
	
	#SIMDET: \<mode\>,\<simin\>
	
	where:
	\<mode\> - SIM Detection mode, as before
	\<simin\> - SIMIN pin real status
	  0 - SIM not inserted
	  1 - SIM inserted

AT#SIMDET=?	Test command reports the supported range of values for parameter \<mode\>

3.4.7.1.34   *Write to I2C - #I2CWR*

#I2CWR – Write to I2C	
AT#I2CWR= \<sdaPin\>, \<sclPin\>, \<deviceId\>, \<registerId\>, \<len\>	This command is used to Send Data to an I2C peripheral connected to module GPIOs **\<sdaPin\>**: GPIO number for SDA . Valid range is "any input/output pin" (see "Hardware User's Guide".) **\<sclPin\>**: GPIO number to be used for SCL. Valid range is "any output pin" (see "Hardware User's Guide"). **\<deviceId\>**: address of the I2C device, with the LSB, used for read\write command. It doesn't matter if the LSB is set to 0 or to 1. 10 bit addressing supported. Value has to be written in hexadecimal form (without 0x). **\<registerId\>**: Register to write data to , range 0..255. Value has to be written in hexadecimal form (without 0x). **\<len\>**: number of data to send. Valid range is 1-254. The module responds to the command with the prompt '>' and awaits for the data to send.

#I2CWR – Write to I2C	
	To complete the operation send Ctrl-Z char (0x1A hex); to exit without writing the message send ESC char (0x1B hex). Data shall be written in Hexadecimal Form.
	If data are successfully sent, then the response is OK. If data sending fails for some reason, an error code is reported. Example if CheckAck is set and no Ack signal was received on the I2C bus.
	Note: At the end of the execution GPIO will be restored to the original setting (check AT#GPIO Command) Device address, register address where to read from\ write to, and date bytes have to be written in hexadecimal form without 0x.
AT#I2CWR=?	Test command returns the range of each parameter.
Example	AT#I2CWR=2,3,30,10,14 > 00112233445566778899AABBCCDD<ctrl-z> OK Set GPIO2 as SDA, GPIO3 as SCL; Device I2C address is 0x30; 0x10 is the address of the first register where to write I2C data; 14 data bytes will be written since register 0x10.

3.4.7.1.35 Read to I2C - #I2CRD

#I2CRD – Read to I2C	
AT#I2CRD= <sdaPin>, <sclPin>, <deviceId>, <registerId>, <len>	This command is used to Send Data to an I2C peripheral connected to module GPIOs **<sdaPin >:** GPIO number for SDA . Valid range is "any input/output pin" (see Test Command.) **<sclPin>:** GPIO number to be used for SCL. Valid range is "any output pin" (see Command Test). **<deviceId>:** address of the I2C device, with the LSB, used for read\write command. It doesn't matter if the LSB is set to 0 or to 1. 10 bit addressing supported. Value has to be written in hexadecimal form (without 0x before). **<registerId>:** Register to read data from, range 0..255. Value has to be written in hexadecimal form (without 0x before). **<len>:** number of data to receive. Valid range is 1-254. Data Read from I2C will be dumped in Hex:

#I2CRD – Read to I2C	
	Note: If data requested are more than data available in the device, dummy data (normally 0x00 or 0xff) will be dumped. **Note**: At the end of the execution GPIO will be restored to the original setting (check AT#GPIO Command) **Note**: device address, register address where to read from\ write to, and date bytes have to be written in hexadecimal form without 0x.
AT#I2CRD=?	Test command reports the supported list of currently available <service>s.
Example	AT#I2CRD=2,3,20,10,12 #I2CRD: 00112233445566778899AABBCC OK

3.4.7.1.36 *Enable Test Mode command in not signalling mode - #TESTMODE*

#TESTMODE – Enable Test Mode command in not signalling mode	
AT#TESTMODE=<command>	The command allows setting module in not signaling mode. The functionality has to be first activated by sending **AT#TESTMODE="TM"**, which sets the module in Test Mode. Only after this set, **AT#TESTMODE** can be used with the other allowed commands. To exit from Test Mode and go back to Operative Mode, the command **AT#TESTMODE ="OM"** has to be sent. **Parameter**: **<command>**: • "*TM*"→ forces the module in Test Mode; • "*OM*"→ forces the module in Operative Mode **4G Commands**: • "*INIT4G*"→ initialize Radio for 4G transmission • "*TX4G*"→ starts the 4G module transmission if Radio is initialized • "PL4G <power> → change the 4G transmission power has the range -400:200 in tenth of dBm • "RL4G" → provides the Rx power level for the channel set with ""CH4G *<earfcn dl>*" command. "DE*INIT4G*"→ de-initialize Radio for 4G transmission • "CH4G *<earfcn> <bw>*"→ changes the 4G earfcn ul or dl for transmitting or receiving, and sets the bandwidth: {table below}

LTE_EARFCN UL	Band
18000 ÷ 18599	1

18600 ÷ 19199	2
19200 ÷ 19949	3
19950 ÷ 20399	4
20400 ÷ 20649	5
21450 ÷ 21799	8
23010 ÷ 23179	12
23180 ÷ 23279	13
23730 ÷ 23849	17
24000 ÷ 24149	19
24150 ÷ 24449	20
24450 ÷ 24599	21
26690 ÷ 27039	26
27210 ÷ 27659	28

LTE_EARFCN DL	Band
0 ÷ 599	1
600 ÷ 1199	2
1200 ÷ 1949	3
1950 ÷ 2399	4
2400 ÷ 2649	5
3450 ÷ 3799	8
5010 ÷ 5179	12
5180 ÷ 5279	13
5730 ÷ 5849	17
6000 ÷ 6149	19
6150 ÷ 6449	20
6450 ÷ 6599	21
8690 ÷ 9039	26
9210 ÷ 9659	28

<bw>	Band (MHz)
0	1,4
1	3,0
2	5,0

Note:
- Bands support varies depending on the product

Note : it's not possible to read RX power level during an ongoing TX

AT#TESTMODE?	Read command reports the currently selected **<command>** in the format: **#TESTMODE: <TestModeStatus>** Where: **<TestModeStatus>** can assume the following values:

	- 1 if the module is in Test Mode - 0 if the module is in Operative Mode
AT#TESTMODE=?	Test command returns the **OK** result code

3.4.7.1.37 *Update PLMN List - #PLMNUPDATE*

#PLMNUPDATE – Update PLMN List	
AT#PLMNUPDATE=[<action>,<MCC>,<MNC>[,<PLMNname>]]	Set command adds a new entry or updates an existing entry of the module PLMN list. **Parameter**: <**action**> - command action 0 - remove the entry with selected <MCC> and <MNC>. Parameter <PLMNname> will be ignored 1 - update the entry with selected <MCC> and <MNC> if it is already present, otherwise add it. 2 – remove all entries. Parameters <MCC> and <MNC> are not used in this case. <**MCC**> - Mobile Country Code. String value, length 3 digits. <**MNC**> - Mobile Network Code. String value, min length 2 digits, max length 3 digits. <**PLMNname**> - Name of the PLMN; string value, max length 30 characters. **Note**: the entries will be saved in NVM. **Note**: this command supports up to 30 entries. **Note**: entries added or updated with **#PLMNUPDATE** are effective only if **#PLMNMODE** is set to 2.
AT#PLMNUPDATE?	Read command returns the list of entries added or updated with set command, in the format: **#PLMNUPDATE**: <MCC>,<MNC>,<PLMNname> **#PLMNUPDATE**: <MCC>,<MNC>,<PLMNname> … **OK** **Note**: the entries are in increasing order by MCC and MNC
AT#PLMNUPDATE=?	Test command returns the range of <**action**> parameter and the maximum length of <**MCC**>, <**MNC**> and <**PLMNname**> parameters.

3.4.7.1.38   *Select IoT technology - #WS46*

#WS46 - Select IoT technology	

AT#WS46=<n>	Set command selects the IoT technology to operate with.
	
	Parameter:
	<n> - integer type, it is the technology to be used:
	  0 – CAT-M1
	  1 – NB-IoT
	  2 – CAT-M1 and NB-IoT
	
	Note: <n> parameter setting is stored in NVM and available at next reboot.

AT#WS46?	Read command reports the currently selected technology, in the format: **#WS46: <n>**
AT#WS46=?	Test command reports the range for the parameter <n>.

3.4.7.1.39   *Set Active Firmware Image - #FWSWITCH*

#FWSWITCH – Set Active Firmware Image	
AT#FWSWITCH =<image_number> [,<storage_conf>]	Set command allows enabling a specific firmware image on products embedding 2 different firmware images. **Parameters**: **<image_number>** - Firmware Image To Be Enabled **<storage_conf>** - Setting Storage Configuration 1 – Save the **<image_number>** value in NVM (Default)
AT#FWSWITCH?	Read command reports the current active firmware image: **#FWSWITCH =<image_number>**
AT#FWSWITCH=?	Test command reports the range of supported values for parameters **<image_number>,<storage_conf>**
Example	Switch to Image 1: AT#FWSWITCH =1,1 OK

Note	This AT command performs a system reboot.

3.4.7.2   Multisocket AT Commands

3.4.7.2.1   *Socket Status - #SS*

#SS - Socket Status	

AT#SS	Execution command reports the current status of the sockets in the format:
	
	#SS: <connId>,<state>,<locIP>,<locPort>,<remIP>,<remPort>
	[<CR><LF><connId>,<state>,<locIP>,<locPort>,<remIP>,<remPort>
	[…]]
	
	where:
	<connId> - socket connection identifier
	  1..6
	<state> - actual state of the socket:
	  0 - Socket Closed.
	  1 - Socket with an active data transfer connection.
	  2 - Socket suspended.
	  3 - Socket suspended with pending data.
	  4 - Socket listening.
	  5 - Socket with an incoming connection. Waiting for the user accept or shutdown command.
	  6 – Socket in opening process. 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 use 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.

AT#SS=?	Test command returns the **OK** result code.

3.4.7.2.2   *Socket Info - #SI*

#SI - Socket Info	
AT#SI[=<connId>]	Execution command is used to get information about socket data traffic.

#SI - Socket Info	
	Parameters: **<connId>** - socket connection identifier 1..6 The response format is: **#SI: <connId>,<sent>,<received>,<buff_in>,<ack_waiting>** **where:** **<connId>** - socket connection identifier, as before **<sent>** - total amount (in bytes) of sent data since the last time the socket connection identified by **<connId>** has been opened **<received>** - total amount (in bytes) of received data since the last time the socket connection identified by **<connId>** has been opened **<buff_in>** - total amount (in bytes) of data just arrived through the socket connection identified by **<connId>** and currently buffered, not yet read **<ack_waiting>** - total amount (in bytes) of sent and not yet acknowledged data since the last time the socket connection identified by **<connId>** has been opened **Note:** not yet acknowledged data are available only for TCP connections. The value **<ack_waiting>** is always 0 for UDP connections. **Note:** issuing **#SI<CR>** causes getting information about data traffic of all the sockets, the response format is: **#SI: <connId1>,<sent1>,<received1>,<buff_in1>,<ack_waiting1>** **<CR><LF>…** **#SI: <connId6>,<sent6>,<received6>,<buff_in6>,<ack_waiting6>**
AT#SI=?	Test command reports the range for parameter **<connId>**.
Example	AT#SI #SI: 1,123,400,10,50 #SI: 2,0,100,0,0 #SI: 3,589,100,10,100 #SI: 4,0,0,0,0 #SI: 5,0,0,0,0 #SI: 6,0,98,60,0 OK Sockets 1,2,3,6 are opened with some data traffic. For example socket 1 has 123 bytes sent, 400 bytes received, 10 byte waiting to be read and 50 bytes waiting to be acknowledged from the remote side. AT#SI=1 #SI: 1,123,400,10,50 OK We have information only about socket number 1

3.4.7.2.3 *Socket Type - #ST*

#ST – Socket Type	
AT#ST [=<ConnId>]	Set command reports the current type of the socket (TCP/UDP) and its direction (Dialer / Listener) **Parameter:** **<ConnId>** - socket connection identifier 1..6 The response format is: **#ST: <connId>,<type>,<direction>** **Where**: **<connId>** - socket connection identifier 1..6 **<type>** - socket type 0 – No socket 1 – TCP socket 2 – UDP socket **<direction>** - direction of the socket 0 – No 1 – Dialer 2 – Listener **Note:** issuing #ST<CR> causes getting information about type of all the sockets; the response format is: #ST: <connId1>,<type1>,<direction1> <CR><LF> … #ST: <connId6>,< type 6>,< direction 6>
AT#ST=?	Test command reports the range for parameter <connId>.
Example	single socket: AT#ST=3 #ST: 3,2,1 Socket 3 is an UDP dialer.

3.4.7.2.4 *Context Activation - #SGACT*

#SGACT - Context Activation	
AT#SGACT= <cid>,<stat> [,<userId>, <pwd>]	Execution command is used to activate the specified PDP context, followed by binding data application to the PS network. Also, it is used to deactivate the PDP context and unbind data application from PS network

#SGACT - Context Activation	

	Parameters:
	<cid> - a numeric parameter which specifies a particular PDP context definition (see **+CGDCONT** command)
	
	<stat>
	  0 - deactivate the context
	  1 - activate the context
	<userId> - string type,used only if the context requires it
	
	<pwd> - string type, used only if the context requires it
	
	Execution command returns a list of IP addresses for the specified context identifiers in the format:
	If IP or IPV6 PDP context:
	#SGACT: <ipAddr>
	
	For DUAL STACK IPV4V6 PDP context:
	#SGACT: [<ipAddrV4>],[<ipAddrV6>]
	
	Where:
	<ipAddr> - ip address ipv4 or ipv6
	<ipAddrV4> - ip address ipv4(if v4 PDP context activated)
	<ipAddrV6> - ip address ipv6(if v6 PDP context activated)
	
	Note: context activation/deactivation returns ERROR if there is not any socket associated to it (see AT#SCFG).
	Note: In LTE network, default PDP context(cid 1) is activated by piggybacking on LTE attach procedure and maintained until detached from NW. This command with cid 1 is just binding or unbinding application to the default PDP context.
	Note: If the unsolicited result code for obtaining IP address was enabled (urcmode value) using #SGACTCFG command, on start up and
	due to USB enumeration timing the unsolicited may not appear , user should manually use +CGPADDR command to see the IP address.

AT#SGACT?	Returns the state of all the five contexts, in the format:
	
	#SGACT: <cid1>,<Stat1><CR><LF>
	…
	#SGACT: <cid5>,<Stat5>
	
	where:
	<cid*n*> - as **<cid>** before
	<stat*n*> - context status
	  0 - context deactivated
	  1 - context activated

AT#SGACT=?	Reports the range for the parameters **<cid>** and **<stat>**

3.4.7.2.5 *Socket Shutdown - #SH*

#SH - Socket Shutdown	
AT#SH=<connId>	This command is used to close a socket. **Parameter:** **<connId>** - socket connection identifier 1..6 **Note:** a socket connection can be closed only when it is in suspended mode (with pending data too). Trying to close an active socket connection will produce an error.
AT#SH=?	Test command returns the **OK** result code.

3.4.7.2.6 *Socket Configuration - #SCFG*

#SCFG - Socket Configuration	

AT#SCFG=<connId>,<cid>, <pktSz>,<maxTo>,<connTo>,<txTo>	Set command sets the socket configuration parameters.
	
	Parameters:
	<connId> - socket connection identifier
	  1..6
	<cid> - - a numeric parameter which specifies a particular PDP context definition (see +CGDCONT command)
	<pktSz> - packet size to be used by the TCP/UDP/IP stack for data sending. Used for online data mode only.
	  0 - automatically chosen by the device.
	  1..1500 - packet size in bytes.
	<maxTo> - exchange timeout(or socket inactivity time); if there's no data exchange within this timeout period the connection is closed
	  0 - no timeout
	  n(1...65535) - timeout value in seconds (default 90 s.)
	<connTo> - connection timeout; if we can't establish a connection to the remote within this timeout period, an error is raised.
	  0 - no timeout
	  n(10...1200) - timeout value in hundreds of milliseconds (default 600)
	<txTo> - data sending timeout; data are sent even if they're less than max packet size , after this period. Used for online data mode only.
	  0 - no timeout
	  1..255- timeout value in hundreds of milliseconds (default 50)
	  256 – set timeout value in 10 milliseconds
	  257 – set timeout value in 20 milliseconds
	  258 – set timeout value in 30 milliseconds

#SCFG - Socket Configuration	
	259 – set timeout value in 40 milliseconds 260 – set timeout value in 50 milliseconds 261 – set timeout value in 60 milliseconds 262 – set timeout value in 70 milliseconds 263 – set timeout value in 80 milliseconds 264 – set timeout value in 90 milliseconds **Note:** these values are automatically saved in NVM.
AT#SCFG?	Read command returns the current socket configuration parameters values for all the six sockets, in the format: **#SCFG: <connId1>,<cid1>,<pktsz1>,<maxTo1>,<connTo1>,<txTo1>** **<CR><LF>** **. . .** **#SCFG: <connId6>,<cid6>,<pktsz6>,<maxTo6>,<connTo6>,<txTo6>** **<CR><LF>**
AT#SCFG=?	Test command returns the range of supported values for all the subparameters.
Example	at#scfg? #SCFG: 1,1,300,90,600,50 #SCFG: 2,2,300,90,600,50 #SCFG: 3,2,250,90,600,50 #SCFG: 4,1,300,90,600,50 #SCFG: 5,1,300,90,600,50 #SCFG: 6,1,300,90,600,50 OK

3.4.7.2.7 *Context activation configuration extended - #SGACTCFGEXT*

#SGACTCFGEXT - context activation configuration extended	
AT#SGACTCFGEXT= <cid>, <abortAttemptEnable> [,<unused> [,<unused> [,<unused>]]]	Execution command is used to enable new features related to context activation. **Parameters:** <cid> - a numeric parameter which specifies a particular PDP context definition (see +CGDCONT command) **<abortAttemptEnable>** 0 – old behavior: no abort possible while attempting context activation 1 – abort during context activation attempt is possible by sending a byte on the serial port. It takes effect on successive GPRS context activation attempt through #SGACT command in the following manner.

	While waiting for AT#SGACT=<cid>,1 response(up to 150 s) is possible to abort attempt by sending a byte and get back AT interface control(NO CARRIER indication). **Note:** If we receive delayed CTXT ACTIVATION ACCEPT after abort, network will be automatically informed of our aborted attempt through relative protocol messages(SM STATUS) and will also close on its side. Otherwise, if no ACCEPT is received after abort, network will be informed later of our PDP state through other protocol messages (routing area update for instance). **Note:** the command is not effective while the context is already open.
AT#SGACTCFGEXT?	Read command reports the state of all the five contexts, in the format: **#SGACTCFGEXT:** <cid1>,< abortAttemptEnable1 >,0,0,0<CR><LF> … **#SGACTCFGEXT:** <cid5>,< abortAttemptEnable5 >,0,0,0<CR><LF> **where:** **<cid*n*>** - as <cid> before <abortAttemptEnable *n*> - as <abortAttemptEnable> before. **Note:** values are automatically saved in NVM
AT#SGACTCFGEXT=?	Test command reports supported range of values for all parameters

3.4.7.2.8 *PAD forward character - #PADFWD*

PAD forward character - #PADFWD	
AT#PADFWD= <char>[,<mode>]	This command sets the char that immediately flushes pending data to socket, opened with AT#SD command. **Parameters:** **<char>** a number, from 0 to 255, that specifies the asci code of the char used to flush data **<mode>** flush mode, 0 – normal mode (default) 1 – reserved **Note:** use AT#PADCMD to enable the socket char-flush activity.

PAD forward character - #PADFWD	
AT#PADFWD?	Read command reports the currently selected <char> and <mode> in the format: #PADFWD: <char>,mode
AT#PADFWD=?	Test command reports the supported range of values for parameters <char> and <mode>.

- **Feature PAD currently applicable for UART only**

3.4.7.2.1 *PAD command features - #PADCMD*

PAD command features - #PADCMD	
AT#PADCMD= <mode>	This command sets features of the pending data flush to socket, opened with AT#SD command. **Parameters:** **<mode>:** Bit 1: 1 - enable forwarding; 0 – disable forwarding; Other bits reserved. **Note:** forwarding depends on character defined by AT#PADFWD
AT#PADCMD?	Read command reports the currently selected <mode> in the format: #PADCMD: mode
AT#PADCMD=?	Test command reports the supported range of values for parameter <mode>.

- **Feature PAD currently applicable for UART only**

3.4.7.2.2 *Socket Configuration Extended - #SCFGEXT*

#SCFGEXT - Socket Configuration Extended	
AT#SCFGEXT= <connId>, <srMode>, <dataMode>, <keepalive> [,<ListenAutoRsp>[,<sendDataMode>]]	Set command sets the socket configuration extended parameters. **Parameters:** **<connId>** - socket connection identifier 1..6 **<srMode>** - **SRing** URC mode 0 - normal mode (default): **SRING : <connId>** **where:** **<connId>** - socket connection identifier, as before 1 - data amount mode: **SRING : <connId>,<recData>**

#SCFGEXT - Socket Configuration Extended	
	where: **\<connId\>** - as before **\<recData\>** - amount of data received on the socket connection 2 - data view mode: **SRING : \<connId\>,\<recData\>,\<data\>** **where:** **\<connId\>** - **\<recData\>** - as before **\<data\>** - received data; the presentation format depens on the subparameter **\<dataMode\>** value **\<dataMode\>** - "data view mode" presentation format 0 - data represented as text (default) 1 - data represented as sequence of hexadecimal numbers (from 00 to FF) 3 – Data view with UDP datagram informations: **SRING : \<remoteIP\>,\<remotePort\>\<connId\>,\<recData\>,** **\<dataLeft\>,\<data\>** same as before with \<remoteIP\>,\<remotePort\> and \<dataLeft\> that means the number of bytes left in the UDP datagram **\<keepalive\> - TCP keepalive** timer timeout -The interval between two keepalive transmissions in idle condition. 0 - **TCP keepalive** timer is deactivated (default) 1..240 - **TCP keepalive** timer timeout in minutes \<**ListenAutoRsp**\> - Set the listen auto-response mode, that affects the commands AT#SL and AT#SLUDP 0 - Deactivated (default) 1 – Activated \<**sendDataMode**\>- data mode for sending data in command mode(AT#SSEND) 0 - data represented as text (default) 1 - data represented as sequence of hexadecimal numbers (from 00 to FF) Each octet of the data is given as two IRA character long hexadecimal number **Note**: **KeepAlive Interval -** Interval between two successive keepalive retransmissions, if acknowledgement to the previous keepalive transmission is not received. Non configurable value: 75 sec. **KeepAlive Probes** - The number of unacknowledged retransmissions to send out before closing socket. Non configurable value: 9 retransmissions. **Note:** these values are automatically saved in NVM

#SCFGEXT - Socket Configuration Extended	
	Note: for the behaviour of AT#SL and AT#SLUDP in case of auto-response mode or in case of no auto-response mode, see the description of the two commands.
AT#SCFGEXT?	Read command returns the current socket extended configuration parameters values for all the six sockets, in the format: **#SCFGEXT: <connId1>,<srMode1>,<dataMode1>,<keepalive1>,** <ListenAutoRsp1>,<sendDataMode1>,<CR><LF> **...** **#SCFGEXT: <connId6>,<srMode6>,<dataMode6>,<keepalive6>** <ListenAutoRsp6>,<sendDataMode6>,
AT#SCFGEXT=?	Test command returns the range of supported values for all the subparameters
Example	Socket 1 set with data view sring, text data mode, a keepalive time of 30 minutes and listen auto-response set. Socket 3 set with data amount sring, hex recv data mode, no keepalive and listen auto-response not set. Socket 4 set with hex recv and send data mode at#scfgext? #SCFGEXT: 1,2,0,30,1,0 #SCFGEXT: 2,0,0,0,0,0 #SCFGEXT: 3,1,1,0,0,0 #SCFGEXT: 4,0,1,0,0,1 #SCFGEXT: 5,0,0,0,0,0 #SCFGEXT: 6,0,0,0,0,0 OK

3.4.7.2.3 *Socket Configuration Extended 2 - #SCFGEXT2*

#SCFGEXT2 - Socket Configuration Extended	
AT#SCFGEXT2= **<connId>,** **<bufferStart>** **[,<abortConnAttempt>** **[, unused_B>** **[,<unused_C>** **[,<noCarrierMode>]]]]**	Set command sets the socket configuration extended parameters for features not included in #SCFGEXT command. **Parameters:** <**connId**> - socket connection identifier 1..6 <**bufferStart**> - Set the sending timeout method based on new data received from the serial port (<**txTo**> timeout value is set by #SCFG command). Restart of transmission timer done when new data received from the serial port. **0** – old behaviour for transmission timer (#SCFG command 6th parameter old behaviour, start only first time if new data are received from the serial port) **1** – new behaviour for transmission timer : Restart when new data received from serial port

#SCFGEXT2 - Socket Configuration Extended	
	Note: is necessary to avoid overlapping of the two methods. Enabling new method, the old method for transmission timer (#SCFG) automatically disabled to avoid overlapping.
	Note: check if new data have been received from serial port is done with a granularity that is directly related to #SCFG <**txTo**> setting with a maximum period of 1 sec.
	<**abortConnAttempt**> - Enable connection attempt(#SD) abort before CONNECT (online mode) or OK (command mode) **0** – Not possible to intterupt connection attempt **1** – It is possible to interrupt the connection attempt (<**connTo**> set by #SCFG or DNS resoultion running if required) and give back control to AT interface by reception of a character. As soon as the control given to the AT interface, the ERROR message will be received on the interface itself.
	Note: values automatically saved in NVM.
	<**noCarrierMode**> - permits to choose NO CARRIER indication format when the socket is closed as follows **0** – NO CARRIER (default) Indication is sent as usual, without additional information **1** – NO CARRIER:<**connId**> Indication of current <connId> socket connection identifier is added **2** – NO CARRIER:<**connId**>,<**cause**> Indication of current <connId> socket connection identifier and closure <cause> added. For possible <cause> values, see also #SLASTCLOSURE
	Note: like #SLASTCLOSURE, in case of subsequent consecutive closure causes received, the original disconnection cause indicated. **Note**: in the case of command mode connection and remote closure with subsequent inactivity timeout closure without retrieval of all available data(#SRECV or SRING mode 2), it is indicated cause 1 for both possible FIN and RST from remote. parameters values for all the six sockets, in the format: #SCFGEXT2:<**connId1**>,<**bufferStart1**>,<**abortConnAttempt**>,0,0,0 <CR><LF> ... #SCFGEXT2:<**connId1**>,<**bufferStart1**>,<**abortConnAttempt**>,0,0,0
AT?	
AT#SCFGEXT2=?	Test command returns the range of supported values for all the subparameters
Example	AT#SCFGEXT2=1,1 OK AT#SCFGEXT2=2,1 OK

#SCFGEXT2 - Socket Configuration Extended	
	AT#SCFGEXT2? #SCFGEXT2: 1,1,0,0,0,0 #SCFGEXT2: 2,1,0,0,0,0 #SCFGEXT2: 3,0,0,0,0,0 #SCFGEXT2: 4,0,0,0,0,0 #SCFGEXT2: 5,0,0,0,0,0 #SCFGEXT2: 6,0,0,0,0,0 OK AT#SCFG? #SCFG: 1,1,300,90,600,50 #SCFG: 2,1,300,90,600,50 #SCFG: 3,1,300,90,600,50 #SCFG: 4,2,300,90,600,50 #SCFG: 5,2,300,90,600,50 #SCFG: 6,2,300,90,600,50 OK AT#SCFG=1,1,300,90,600,30 OK Current configuration: socket with connId 1 and 2 are configured with new transmission timer behaviour. <txTo> corresponding value has been changed (#SCFG) for connId 1, for connId 2 has been left to default value.

3.4.7.2.4 *Socket Configuration Extended 3 - #SCFGEXT3*

#SCFGEXT3 - Socket Configuration Extended 3	
AT#SCFGEXT3= <connId>,<immRsp>[,<closureTypeCmdM odeEnabling>[,< fastsring>[,<unused_ C>[,<unused_D>]]]]]	Set command sets the socket configuration extended parameters for features not included in #SCFGEXT command nor in #SCFGEXT2 command. **Parameters:** <**connId**> - socket connection identifier 1..6 <**immRsp**> - Enables AT#SD command mode immediate response 0 – factory default, means that AT#SD in command mode (see AT#SD) returns after the socket is connected 1 – Means that AT#SD in command mode returns immediately. Then the state of the connection can be read by the AT command AT#SS <**closureTypeCmdModeEnabling**> - It has no effect and is included only for backward compatibility 0 – factory default <**fastsring**> - It has no effect and is included only for backward compatibility 0 – factory default **Note:** parameter is saved in NVM
AT#SCFGEXT3?	Read command returns the current socket extended configuration parameters values for all the six sockets, in the format:

#SCFGEXT3 - Socket Configuration Extended 3	
	#SCFGEXT3: <connId1>,<immRsp1>,<closureTypeCmdModeEnabling1>, <fastsring1>,0,0<CR><LF> . . . #SCFGEXT3: <connId6>,<immRsp6>,<closureTypeCmdModeEnabling6>, <fastsring6>,0,0<CR><LF>
AT#SCFGEXT3=?	Test command returns the range of supported values for all the parameters.

3.4.7.2.5 Socket Dial - #SD

#SD - Socket Dial	
AT#SD=<connId>, <txProt>,<rPort>, <IPaddr> [,<closureType> [,<lPort> [,<connMode> [,<txTime> [,<userIpType>]]]]]	Execution command opens a remote connection via socket. **Parameters:** **<connId>** - socket connection identifier 1..6 **<txProt>** - transmission protocol 0 - TCP 1 - UDP **<rPort>** - remote host port to contact 1..65535 **<IPaddr>** - address of the remote host, string type. This parameter can be either: - any valid IP address in the format: "xxx.xxx.xxx.xxx" - any host name to be solved with a DNS query - any valid IPv6 address in the format: xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx or xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx **<closureType>** - socket closure behaviour for TCP 0 - local host closes immediately when remote host has closed (default)255 - local host closes after an escape sequence (+++) **<lPort>** - UDP connections local port 1..65535 **<connMode>** - Connection mode 0 - online mode connection (default) 1 - command mode connection **<txTime>** - Adjusting a time interval for series of UDP data packets will be uploaded. 0 – Time interval is not requested (default) 1..1000 – Time interval in milliseconds. **<userIpType>** - ip type for socket to open 0 – no ip type chosen;[default] 1 – ipv4. 2 – ipv6.

#SD - Socket Dial	
	Note: <userSockType> this parameter only valid when <ipaddr> is domain name and dual stack connection is open by (AT#sgact). **Note: when <userSockType> is "no ip type chosen**" ipv6 will be requested firstly. When ipv6 DNS server doesn't support so ipv4 will be requested. **Note: <closureType>** parameter is valid for TCP connections only and has no effect (if used) for UDP connections. **Note: <lPort>** parameter is valid for UDP connections only and has no effect (if used) for TCP connections. **Note:** if we set **<connMode>** to **online mode connection** and the command is successful we enter in **online data mode** and we see the intermediate result code **CONNECT**. After the **CONNECT** we can suspend the direct interface to the socket connection (nb 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's possible to resume it in every moment (unless the socket inactivity timer timeouts, see **#SCFG**) by using the **#SO** 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 there are input data arrived through a connected socket and not yet read because the module entered **command mode** before reading them (after an escape sequence or after **#SD** 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 **#SCFGEXT** setting); it's possible to read these data afterwards issuing **#SRECV**. Under the same hypotheses it's possible to send data while in **command mode** issuing **#SSEND**. **Note:** <txTime> parameter is valid for UDP connections only and has no effect (if used) for TCP connections. For slow servers it is recommended to adjust the time interval for uploading series of data packets in order to do not lose data. The following data packet will be sent after the previous data packet's time interval has been expired.
AT#SD=?	Test command reports the range of values for all the parameters.
Example	Open socket 1 in online mode AT#SD=1,0,80,"www.google.com",0,0,0 CONNECT … Open socket 1 in command mode AT#SD=1,0,80,"www.google.com",0,0,1 OK

3.4.7.2.6 *Socket Accept - #SA*

#SA - Socket Accept	
AT#SA=<connId> [,<connMode>]	Execution command accepts an incoming socket connection after an URC **SRING: <connId>** **Parameter:** **<connId>** - socket connection identifier 1..6 **<connMode>** - Connection mode, as for command **#SD**. 0 - online mode connection (default) 1 - command mode connection **Note:** the **SRING** URC has to be a consequence of a **#SL** issue
AT#SA=?	Test command reports the range of values for all the parameters.

3.4.7.2.7 *Socket Restore - #SO*

#SO - Socket Restore	
AT#SO=<connId>	Execution command resumes socket connection which has been suspended by the escape sequence. **Parameter:** **<connId>** - socket connection identifier 1..6
AT#SO=?	Test command reports the range of values for **<connId>** parameter.

3.4.7.2.8 *Socket Listen - #SL*

#SL - Socket Listen	
AT#SL=<connId>, <listenState>, <listenPort> [,<lingerT>]	This command opens/closes a socket listening for an incoming connection on a specified port. **Parameters:** **<connId>** - socket connection identifier 1..6 **<listenState>** - 0 - closes socket listening 1 - starts socket listening **<listenPort>** - local listening port 0..65535 **<lingerT>** - linger time 0 - immediate closure after remote closure 255 - local host closes only after an escape sequence (+++) **Note:** if successful, commands returns a final result code **OK**. If the ListenAutoRsp flag has not been set through the command AT#SCFGEXT (for the specific connId), then, when a TCP connection request comes on the input

#SL - Socket Listen	
	port, if the sender is not filtered by internal firewall (see **#FRWL**), an URC is received: **+SRING : <connId>** Afterwards we can use **#SA** to accept the connection or **#SH** to refuse it. If the ListenAutoRsp flag has been set, then, when a TCP connection request comes on the input port, if the sender is not filtered by the internal firewall (see command **#FRWL**), the connection is automatically accepted: the **CONNECT** indication is given and the modem goes into **online data mode**. If the socket is closed by the network the following URC is received: **#SKTL: ABORTED**
AT#SL?	Read command returns all the actual listening TCP sockets.
AT#SL=?	Test command returns the range of supported values for all the subparameters.
Example	Next command opens a socket listening for TCP on port 3500 without. AT#SL=1,1,3500 OK

3.4.7.2.9 *Detect the Cause of a Socket disconnection - #SLASTCLOSURE*

#SLASTCLOSURE – Detect the cause of a socket disconnection	
AT#SLASTCLOSURE=[<connId>]	Execution command reports socket disconnection cause. **Parameters:** **<connId>** - socket connection identifier 1..6 The response format is: **#SLASTCLOSURE: <connId>,<cause>** **where:** **<connId>** - socket connection identifier, as before **<cause>** - socket disconnection cause: **0** – not available(socket has not yet been closed) **1**.- remote host TCP connection close due to FIN/END: normal remote disconnection decided by the remote application **2** -.remote host TCP connection close due to RST, all others cases in which the socket is aborted without indication from peer (for instance because peer doesn't send ack after maximum number of retransmissions/peer is no more alive). All these cases include all the "FATAL" errors after recv or send on the TCP socket(named as different from EWOULDBLOCK) **3**.- socket inactivity timeout

#SLASTCLOSURE – Detect the cause of a socket disconnection	
	4.- network deactivation(PDP context deactivation from network)
	Note: any time socket is re-opened, last disconnection cause is reset. Command report 0(not available).
	Note: user closure cause(#SH) is not considered and if a user closure is performed after remote disconnection, remote disconnection cause remains saved and is not overwritten.
	Note: if more consecutive closure causes are received, the original disconnection cause is saved. (For instance: if a TCP FIN is received from remote and later a TCP RST because we continue to send data, FIN cause is saved and not overwritten)
	Note: also in case of <closureType>(#SD) set to 255, if the socket has not yet been closed by user after the escape sequence, #SLASTCLOSURE indicates remote disconnection cause if it has been received.
	Note: in case of UDP, cause 2 indicates abnormal(local) disconnect. Cause 3 and 4 are still possible. (Cause 1 is obviously never possible)
	Note: in case of command mode connection and remote closure with subsequent inactivity timeout closure without retrieval of all available data(#SRECV or SRING mode 2), it is indicated cause 1 for both possible FIN and RST from remote.
AT#SLASTCLOSURE=?	Test command reports the supported range for parameter <connId>

3.4.7.2.10 Socket Listen UDP - #SLUDP

#SLUDP - Socket Listen UDP	
AT#SLUDP= <connId>, <listenState>[, <listenPort>]	This command opens/closes a socket listening for an incoming UDP connection on a specified port. **Parameters:** **<connId>** - socket connection identifier 1..6 **<listenState>** - 0 - closes socket listening 1 - starts socket listening **<listenPort>** - local listening port 1..65535 **Note:** if successful, the command returns a final result code OK.

#SLUDP - Socket Listen UDP	
	If the ListenAutoRsp flag has not been set through the command AT#SCFGEXT (for the specific connId), then, when an UDP connection request comes on the input port, if the sender is not filtered by internal firewall (see #FRWL), an URC is received: +SRING : <connId> Afterwards we can use #SA to accept the connection or #SH to refuse it. If the ListenAutoRsp flag has been set, then, when an UDP connection request comes on the input port, if the sender is not filtered by the internal firewall (see command #FRWL), the connection is automatically accepted: the **CONNECT** indication is given and the modem goes into **online data mode**. If the socket is closed by the network the following URC is received: **#SLUDP: ABORTED** **Note:** when closing the listening socket <listenPort> is a don't care parameter
AT#SLUDP?	Read command returns all the actual listening UDP sockets.
AT#SLUDP=?	Test command returns the range of supported values for all the subparameters.
Example	Next command opens a socket listening for UDP on port 3500. AT#SLUDP=1,1,3500 OK

3.4.7.2.11 Receive Data in Command Mode - #SRECV

#SRECV – Received Data in Command Mode	
AT#SRECV=<connId>, <maxByte>,[<UDPInfo>]	Execution command permits the user to read data arrived through a connected socket, but buffered and not yet read because the module entered **command mode** before reading them; the module is notified of these data by a **SRING** URC, whose presentation format depends on the last **#SCFGEXT** setting. **Parameters:** **<connId>** - socket connection identifier 1..6 **<maxByte>** - max number of bytes to read 1..1500 <**UDPInfo**> 0 – UDP information disabled (default) 1 – UDP information enabled: data are read just until the end of the UDP datagram and the response carries information about the remote IP address and port and about the remaining bytes in the datagram. AT#SRECV=<connId>,<maxBytes>,1 #SRECV: <remoteIP>,<remotePort><connId>,<recData>, <dataLeft> data

#SRECV – Received Data in Command Mode	
	Note: issuing **#SRECV** when there's no buffered data raises an error.
AT#SRECV=?	*Test command returns the range of supported values for parameters: <connId> <maxByte> and <UDPInfo>*
Example	SRING URC (<srMode> be 0, <dataMode> be 0) telling data have just come through connected socket identified by <connId>=1 and are now buffered SRING: 1 Read in text format the buffered data AT#SRECV=1,15 #SRECV: 1,15 stringa di test OK Or: if the received datagram, received from <IPaddr and <IPport> is of 60 bytes AT#SRECV=1,15,1 #SRECV: <IPaddr>,<IPport>,1,15,45 stringa di test OK SRING URC (<srMode> be 1, <dataMode> be 1) telling 15 bytes data have just come through connected socket identified by <connId>=2 and are now buffered SRING: 2,15 Read in hexadecimal format the buffered data AT#SRECV=2,15 #SRECV: 2,15 737472696e67612064692074657374 OK SRING URC (<srMode> be 2, <dataMode> be 0) displaying (in text format) 15 bytes data that have just come through connected socket identified by <connId>=3; it's no necessary to issue #SRECV to read the data; no data remain in the buffer after this URC SRING: 3,15, stringa di test

3.4.7.2.12   *Send UDP data to a specific remote host - #SSENDUDP*

#SSENDUDP – send UDP data to a specific remote host	
AT#SSENDUDP= **<connId>,** **<remoteIP>,** **<remotePort>**	This command permits, while the module is in command mode, to send data over UDP to a specific remote host. UDP connection has to be previously completed with a first remote host through #SLUDP / #SA. Then, if we receive data from this or another host, we are able to send data to it. Like command #SSEND, the device responds with '>' and waits for the data to send. **Parameters:** **<connId>** - socket connection identifier 1..6 **<remoteIP>** - IP address of the remote host in dotted decimal notation, string type: "xxx.xxx.xxx.xxx" **<remotePort>** - remote host port 1..65535 **Note:** after SRING that indicates incoming UDP data and issuing #SRECV to receive data itself, through #SS is possible to check last remote host (IP/Port). **Note**: if successive resume of the socket to online mode Is performed (#SO), connection with first remote host is restored as it was before.
AT#SSENDUDP=?	Test command reports the supported range of values for parameters **<connId>, <remoteIP> and <remotePort>**.
Example	Starts listening on <LocPort>(previous setting of firewall through #FRWL has to be done) AT#SLUDP=1,1,<LocPort> OK SRING: 1 // UDP data from a remote host available AT#SA=1,1 OK SRING: 1 AT#SI=1 #SI: 1,0,0,23,0 // 23 bytes to read OK AT#SRECV=1,23 #SRECV:1,23 message from first host OK AT#SS=1 #SS: 1,2,<LocIP>,<LocPort>,<RemIP1>,<RemPort1> OK

#SSENDUDP – send UDP data to a specific remote host	
	AT#SSENDUDP=1,<RemIP1>,<RemPort1>
	>response to first host
	OK
	SRING: 1 // UDP data from a remote host available
	AT#SI=1
	#SI: 1,22,23,24,0 // 24 bytes to read
	OK
	AT#SRECV=1,24
	#SRECV:1,24
	message from second host
	OK
	AT#SS=1
	#SS: 1,2,<LocIP>,<LocPort>,<RemIP2>,<RemPort2>
	OK
	Remote host has changed, we want to send a reponse:
	AT#SSENDUDP=1,<RemIP2>,<RemPort2>
	>response to second host
	OK

3.4.7.2.13 *Send UDP data to a specific remote host extended - #SSENDUDPEXT*

#SSENDUDPEXT – send UDP data to a specific remote host extended	
AT#SSENDUDPEXT= <connId>, <bytestosend>, <remoteIP>, <remotePort>	This command permits, while the module is in command mode, to send data over UDP to a specific remote host including all possible octets(from 0x00 to 0xFF) As indicated about #SSENDUDP: UDP socket has to be previously opened through #SLUDP / #SA, then we are able to send data to different remote hosts. Like #SSENDEXT, the device responds with the prompt '> ' and waits for the data to send, operation is automatically completed when <bytestosend> have been sent. **Parameters:** **<connId>** - socket connection identifier 1..6 **<bytestosend>** - number of bytes to be sent 1-1500 **<remoteIP>** - IP address of the remote host in dotted decimal notation, string type: "xxx.xxx.xxx.xxx" **<remotePort>** - remote host port 1..65535
AT#SSENDUDPEXT=?	Test command reports the supported range of values for parameters **<connId>,<bytestosend>,<remoteIP> and <remotePort>**

3.4.7.2.14 *Send Data in Command Mode - #SSEND*

#SSEND – Send Data in Command Mode	
AT#SSEND= <connId>	Execution command permits, while the module is in **command mode,** to send data through a connected socket. **Parameters:** **<connId>** - socket connection identifier 1..6 - The device responds to the command with the prompt '**>**' and waits for the data to send. To complete the operation send **Ctrl-Z** char (**0x1A** hex); to exit without writing the 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's possible to use **#SSEND** only if the connection was opened by **#SD**, else the ME is raising an error **Note:** a byte corresponding to BS char(0x08) is treated with its corresponding meaning; therefore previous byte will be cancelled(and BS char itself will not be sent)
AT#SSEND=?	Test command returns OK message.
Example	Send data through socket number 2 AT#SSEND=2 >Test<CTRL-Z> OK

3.4.7.2.15   *Send data in Command Mode extended - #SSENDEXT*

#SSENDEXT - Send Data In Command Mode extended	
AT#SSENDEXT= <connId>, <bytestosend>	Execution command permits, while the module is in command mode, to send data through a connected socket including all possible octets (from 0x00 to 0xFF). **Parameters:** <**connId**> - socket connection identifier 1..6 **<bytestosend>** - number of bytes to be sent Please refer to test command for range The device responds to the command with the prompt '>' <greater_than><space> and waits for the data to send. When <bytestosend> bytes have been sent, operation is automatically completed. If data are successfully sent, then the response is **OK**. If data sending fails for some reason, an error code is reported. **Note:** it's possible to use **#SSENDEXT** only if the connection was opened by **#SD**, else the ME is raising an error. **Note:** all special characters are sent like a generic byte. (For instance: 0x08 is simply sent through the socket and don't behave like a BS, i.e. previous character is not deleted)
AT#SSENDEXT=?	Test command returns the range of supported values for parameters < **connId** > and

#SSENDEXT - Send Data In Command Mode extended	
	\<bytestosend\>

Example	Open the socket in command mode:
	at#sd=1,0,\<port\>,"IP address",0,0,1
	OK
	Give the command specifying total number of bytes as second **Parameter:**
	at#ssendext=1,256
	\> .... ; // Terminal echo of bytes sent is displayed here
	OK
	All possible bytes (from 0x00 to 0xFF) are sent on the socket as generic bytes.

3.4.7.2.16 *Context Activation and Configuration - #SGACTCFG*

#SGACTCFG - Context Activation and Configuration	
AT#SGACTCFG= **<cid>**, **<retry>**, **[,<delay>** **[,<urcmode>]]**	Execution command is used to enable or disable the automatic activation/reactivation of the context for the specified PDP context, to set the maximum number of attempts and to set the delay between an attempt and the next one. The context is activated automatically after every GPRS Attach or after a NW PDP CONTEXT deactivation if at least one IPEasy socket is configured to this context (sees AT#SCFG). **Parameters:** **<cid>** - a numeric parameter which specifies a particular PDP context definition (see +CGDCONT command) **<retry>** - numeric parameter which specifies the maximum number of context activation attempts in case of activation failure. The value belongs to the following range: 0 - 15 0 - disable the automatic activation/reactivation of the context (default) **<delay>** - numeric parameter which specifies the delay in seconds between an attempt and the next one. The value belongs to the following range: 180 - 3600 **<urcmode>** - URC presentation mode 0 - disable unsolicited result code (default) 1 - enable unsolicited result code, after an automatic activation/reactivation, of the local IP address obtained from the network. It has meaning only if <auto>=1. The unsolicited message is in the format: #SGACT: <ip_address> Reporting the local IP address obtained from the network. **Note:** the URC presentation mode <urcmode> is related to the current AT instance only. Last <urcmode> setting is saved for every instance as extended profile parameter, thus it is possible to restore it even if the multiplexer control channel is released and set up, back and forth. **Note:** <retry> and <delay> setting are global parameter saved in NVM **Note:** if the automatic activation is enabled on a context, then it is not allowed to modify by the command AT#SCFG the association between the context itself and the socket connection identifier; all the other parameters of command AT#SCFG are modifiable while the socket is not connected
AT#SGACTCFG?	Read command reports the state of all the five contexts, in the format: #SGACTCFG: <cid1>,<retry1>,<delay1>, < urcmode >CR><LF> …

	#SGACTCFG: <cid5>,<retry5>,<delay5>,< urcmode > where: <cid*n*> - as <cid> before <retry*n*> - as <retry> before <delay*n*> - as <delay> before < urcmode > - as < urcmode > before
AT#SGACTCFG=?	Test command reports supported range of values for parameters <cid>>,<retry>,<delay>and < urcmode >

3.4.7.3 SSL Commands

3.4.7.3.1 *Enable a SSL socket - #SSLEN*

#SSLEN – Enable a SSL socket	
AT#SSLEN=<SSId>,<Enable>	This command enables a socket secured by SSL **Parameters**: **<SSId>** - Secure Socket Identifier 1 – Until now SSL block manages only one socket **<Enable>** 0 – deactivate secure socket [default] 1 – activate secure socket **Note**: if secure socket is not enabled only test requests can be made for every SSL command except #SSLS (SSL status) which can be issued also if the socket is disabled. Read commands can be issued if at least a <SSId> is enabled. **Note**: these values are automatically saved in NVM. **Note**: a SSL socket cannot be disabled by issuing #SSLEN=1,0 if it is connected.
AT#SSLEN?	Read command reports the currently enable status of secure socket in the format: **#SSLEN: <SSId>,<Enable><CR><LF>** **<CR><LF>** **OK**
AT#SSLEN=?	Test command returns the range of supported values for all the parameters:

#SSLEN – Enable a SSL socket	
	#SSLEN: (1),(0,1)

3.4.7.3.2 *Opens a socket SSL to a remote server - #SSLD*

#SSLD – Opens a socket SSL to a remote server	

AT#SSLD=<SSId>,<rPort>,<IPAddress>,<ClosureType>[,<connMode>[,<Timeout>]]	Execution command opens a remote connection via socket secured through SSL.
	Parameters:
	<SSId> - Secure Socket Identifier
	  1 - Until now SSL block manage only one socket
	<rPort> - Remote TCP port to contact 1..65535
	<IPAddress> - string type, address of SSL server
	<ClosureType> - how to close SSL socket
	  0 – Until now only closure type 0 supported. SSL session id and keys are free.
	<connMode> - connection mode
	  0 – online mode connection.
	  1 – command mode connection (factory default).
	<Timeout> - time-out in 100 ms units. It represents the maximum allowed TCP inter-packet delay. It means that, when more data is expected during the handshake, the module awaits <Timeout> * 100 msecs for the next packet. If no more data can be read, the module gives up the handshake and raises an ERROR response.
	
	Note: if we set **<connMode>** to **online mode connection** and the command is successful we enter into **online data mode** and we see the intermediate result code **CONNECT**.
	After the **CONNECT** we can suspend the direct interface to the socket connection (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 by using the **#SSLO** 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: **<Timeout>** is the total handshake timeout or, in other words, it is not the absolute maximum time between the #SSLD issue and the CONNECT/OK/ERROR response. Though by changing this parameter you can limit the handshake duration (for example in case of congested network or busy server), there's no way to be sure to get the command response within a certain amount of time, because it depends on the TCP connection time, the handshake time and the computation time (which depends on the authentication mode and on the size of keys and certificates).10..5000 - hundreds of ms (factory default is 100)
	
	Note: If secure socket is not enabled, only test requests can be made

#SSLD – Opens a socket SSL to a remote server	
	Note: if timeout is not set for SSL connection the default timeout value, set by **AT#SSLCFG,** is used. **Note**: in online mode the socket is closed after an inactivity period (configurable with #SSLCFG, with a default value of 90 seconds), and the '**NO CARRIER**' message is printed. **Note**: in online mode data is transmitted as soon as the data packet size is reached or as after a transmission timeout. Both these parameters are configurable by using **#SSLCFG** **Note**: Before opening a SSL connection, make sure to have stored the needed secure data (Certificate, CA certificate, private key), using **AT#SSLSECDATA**, for the security level set through **AT#SSLSECCFG**. **Note**: Before opening a SSL connection the GPRS context must have been activated by **AT#SGACT=x,1** **Note**: The PDP context definition that will be used, is set by **AT#SSLCFG** command
AT#SSLD=?	Test command returns the range of supported values for all the parameters: **#SSLD: (1),(1-65535),,(0),(0,1),(10-5000)**
Example	Start command mode: **AT#SSLD=1,8500,"84.94.194.21",0,1** **OK** Start online mode: **AT#SSLD =1,8500,"84.94.194.21",0,0** **OK** **CONNECT** Configure correct PDP context with AT#SSLCFG command: **at#sgact=3,1** **#SGACT: XX.XXX.XXX.XXX** **OK** ** Note the second parameter of sslcfg ** **at#sslcfg=1,3,300,90,100,50,0,0,0,0** **OK** **at#ssld=1,<port>,"IP or URL",0,0** **CONNECT**

3.4.7.3.3 *Send data through a SSL socket - #SSLSEND*

#SSLSEND – Send data through a SSL socket	
AT#SSLSEND= **<SSId>[,< Timeout >]**	This command allows sending data through a secure socket. **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manage only one socket. **<Timeout>** - socket send timeout, in 100 ms. units. 10..5000 - hundreds of ms. (factory default is 100) The device responds to the command with the prompt '>' and waits for the data to send. To complete the operation send Ctrl-Z char (0x1A hex); to exit without writing the 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 1023. **Note**: If secure socket is not enabled using **AT#SSLEN** only test requests can be made. **Note**: If timeout is not set for SSL connection the default timeout value, set by **AT#SSLCFG**, is used. **Note**: Before sending data through the SSL connection it has to be established using **AT#SSLD**
AT#SSLSEND=?	Test command returns the range of supported values for all the parameters: **#SSLSEND: (1),(10-5000)**

3.4.7.3.4 *Read data from a SSL socket - #SSLRECV*

#SSLRECV – Read data from a SSL socket	
AT#SSLRECV= <SSId>,<MaxNumByte>[,<TimeOut>]	This command allows receiving data from a secure socket. **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manage only one socket. **<MaxNumByte> -** max number of bytes to read 1..1000 **< Timeout > -** time-out in 100 ms units 10..5000 - hundreds of ms (factory default is 100) If no data are received the device responds: #SSLRECV: 0<CR><LF> TIMEOUT<CR><LF> <CR><LF> OK If the remote host closes the connection the device responds: #SSLRECV: 0<CR><LF> DISCONNECTED<CR><LF> <CR><LF> OK If data are received the device responds: #SSLRECV: NumByteRead<CR><LF> ...(Data read)... <CR><LF> <CR><LF> OK **Note**: if secure socket is not enabled using **AT#SSLEN** only test requests can be made. **Note**: if timeout is not set for SSL connection the default timeout value, set through **AT#SSLCFG,** is used. **Note**: before receiving data from the SSL connection it has to be established using **AT#SSLD**.
AT#SSLRECV=?	Test command returns the range of supported values for all the parameters: **#SSLRECV: (1),(1-1000),(10-5000)**

3.4.7.3.5 *Report the status of a SSL socket - #SSLS*

#SSLS - Report the status of a SSL socket	
AT#SSLS=<SSId>	This command reports the status of secure sockets. **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manages only one socket If secure socket is connected the device responds to the command in following format: **#SSLS: <SSId>,<ConnectionStatus>,<CipherSuite>** **Note:** ConnectionStatus will equal 2 otherwise: **#SSLS: <SSId>,<ConnectionStatus>** **Note**: ConnectionStatus value will be equal 0 or 1. **ConnectionStatus** possible values are: 0 – Socket Disabled 1 – Connection closed 2 – Connection open **CipherSuite** possible values are: 0 - Chiper Suite is chosen by remote Server [default] 1 - TLS_RSA_WITH_RC4_128_MD5 2 - TLS_RSA_WITH_RC4_128_SHA 3 - TLS_RSA_WITH_AES_256_CBC_SHA 4 - TLS_RSA_WITH_NULL_SHA **Note**: This command can be issued, even if the <SSId> is not enabled.
AT#SSLS=?	Test command returns the range of supported values for all the parameters. #SSLS: (1)
Example	**AT#SSLS=1** **#SSLS: 1,1** **OK** **AT#SSLS=1** **#SSLS: 1,2,0** **OK**

3.4.7.3.6 *Close a SSL socket - #SSLH*

#SSLH – Close a SSL socket	
AT#SSLH= <SSId>[,<ClosureType>]	This command allows closing the SSL connection. **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manage only one socket. **< ClosureType >:** how to close SSL socket 0 – Until now, only closure type 0 supported. SSL session id and keys are free. **Note**: if secure socket is not enabled using **AT#SSLEN** only test requests can be made.
AT#SSLH=?	Test command returns the range of supported values for all the parameters: **#SSLH: (1),(0)**

3.4.7.3.7 *Restore a SSL socket after a +++ - #SSLO*

#SSLO - Restore a SSL socket after a +++	
AT#SSLO=<SSId>	This command allows to restore a SSL connection (online mode) suspended by an escape sequence (+++). After the connection restore, the **CONNECT** message is printed. Please note that this is possible even if the connection has been started in command mode (#SSLD with <connMode> parameter set to 1). **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manage only one socket. **Note**: if secure socket is not enabled using **AT#SSLEN** only test requests can be made. **Note**: Before opening a SSL connection the GPRS context must have been activated by **AT#SGACT=X, 1**. **Note**: if an error occur during reconnection the socket cannot be reconnected then a new connection has to be done.
AT#SSLO=?	Test command returns the range of supported values for all the parameters: **#SSLO: (1)**

3.4.7.3.8 *Configure general parameters of a SSL socket - #SSLCFG*

#SSLCFG – Configure general parameters of a SSL socket	

AT#SSLCFG=
<SSId>,<cid>,<pktSz>,<maxTo>,<defTo>,<txTo>
[,<skipHostMismatch>[,<UNUSED_2>[,<UNUSED_3>[,<UNUSED_4>]]]]
	This command allows configuring SSL connection parameters.
	
	Parameters:
	<SSId> - Secure Socket Identifier
	  1 - Until now SSL block manages only one socket
	<cid> - a numeric parameter which specifies a particular PDP context definition (see +**CGDCONT** command)
	<pktSz> - packet size to be used by the SSL/TCP/IP stack for data sending.
	  0 - select automatically default value (300).
	  1..1500 - packet size in bytes.
	<maxTo> - exchange timeout (or socket inactivity timeout); in online mode, if there's no data exchange within this timeout period the connection is closed.
	  0 - no timeout
	  1..65535 - timeout value in seconds (default 90 s.)
	<defTo> - Timeout that will be used by default whenever the corresponding parameter of each command is not set.
	  10…5000 - Timeout in tenth of seconds (default 100).
	<txTo> - data sending timeout; in online mode after this period data are sent also if they're less than max packet size.
	  0 - no timeout
	  1..255 - timeout value in hundreds of milliseconds (default 50).
	<skipHostMismatch> - ignores Host Mismatch alert.
	  0 - Do not ignore
	  1 – Ignore (default)..
	
	Note: If secure socket is not enabled (using **#SSLEN)** only test requests can be made.
	
	Note: these values automatically saved in NVM.

AT#SSLCFG?	Read command reports the currently selected parameters in the format: **#SSLCFG:** **<SSId1>,<cid>,<pktSz>,<maxTo>,<defTo><txTo>,<skipHostMismatch>,0,0,0**
AT#SSLCFG=?	Test command returns the range of supported values for all the parameters. **#SSLCFG: (1),(1-5),(0-1500),(0-65535),(10-5000),(0-255),(0-1),(0),(0),(0)**

3.4.7.3.9 *Configure security parameters of a SSL socket – #SSLSECCFG*

#SSLSECCFG – Configure security parameters of a SSL socket	
AT#SSLSECCFG= <SSId>,<CipherSuite>,<SecLevel>	This command allows configuring SSL connection parameters. **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manage only one socket **<CipherSuite>** 0 - Chiper Suite is chosen by remote Server [default] 1 - TLS_RSA_WITH_RC4_128_MD5 2 - TLS_RSA_WITH_RC4_128_SHA 3 - TLS_RSA_WITH_AES_256_CBC_SHA 4 - TLS_RSA_WITH_NULL_SHA **<SecLevel>** 0 - No authentication [default] 1 - Manage server authentication 2 - Manage server and client authentication if requested by the remote server **Note**: if no authentication is set no security data are needed (Client certificate, Server CAcertificate and Client private key). **Note**: if only server authentication is managed then Server CAcertificate has to be stored through **AT#SSLSECDATA**. **Note**: if server and client authentication are managed then client certificate and private key, and server CAcertificate have to be stored through **AT#SSLSECDATA**. Please note that private keys with password are not Supported. **Note**: only "rsa_sign" certificates are supported by the Telit Module in client authentication. The remote server must support this certificate type, otherwise the handshake will fail. **Note**: if secure socket is not enabled using **#SSLEN** only test requests can be made. Read command can be issued if at least a <SSId> is enabled. **Note**: these values are automatically saved in NVM.
AT#SSLSECCFG?	Read command reports the currently selected parameters in the format: **#SSLSECCFG: <SSId1>,<CipherSuite>,<SecLevel>**
AT#SSLSECCFG=?	Test command returns the range of supported values for all the parameters. **#SSLSECCFG: (1),(0-2),(0-2)**

3.4.7.3.10 *Manage the security data - #SSLSECDATA*

#SSLSECDATA - Manage the security data	
AT#SSLSECDATA= <SSId>,<Action>,<DataType>[,<Size>]	This command allows to store, delete and read security data (Certificate, CAcertificate, private key) into NVM. **Parameters:** **<SSId>** - Secure Socket Identifier 1 - Until now SSL block manages only one socket.

#SSLSECDATA - Manage the security data	
	<Action> - Action to do. 0 – Delete data from NVM. 1 – Store data into NVM. 2 – Read data from NVM. **<DataType>** 0 – Certificate. 1 – CA certificate. 2 – RSA Private key. **<Size>** - Size of security data to be stored 1..2047 If the **<Action> parameter** is 1 (store data into NVM) the device responds to the command with the prompt '>' and waits for the data to store. To complete the operation send Ctrl-Z char (0x1A hex); to exit without writing the message send ESC char (0x1B hex). If data are successfully stored, then the response is OK; if it fails for some reason, an error code is reported. If the **<Action>** parameter is 2 (read data from NVM), data specified by **<DataType>** parameter is shown in the following format: **#SSLSECDATA: <connId>,<DataType>** **<DATA>** **OK** If **<DataType>** data has not been stored (or it has been deleted) the response has the following format: **#SSLSECDATA: <connId>,<DataType>** **No data stored** **OK** **Note**: Secured data has to be in PEM format. **Note**: private keys with password ARE NOT supported. **Note**: only "rsa_sign" certificates are supported by the Telit Module in client authentication. The remote server must support this certificate type, otherwise the handshake will fail. **Note**: **<size>** parameter is mandatory if the <write> action is issued, but it has to be omitted for <delete> or <read> actions are issued. **Note**: if secure socket is not enabled using **AT#SSLEN** only test requests can be made. **Note**: If socket is connected an error code is reported.
AT#SSLSECDATA?	Read command reports what security data are stored in the format: **#SSLSECDATA:<SSId1>,<CertIsSet>,<CAcertIsSet>,<PrivKeyIsSet>** **<CertIsSet>, <CAcertIsSet>, <PrivKeIsset>** are 1 if related data are stored into NVM otherwise 0
AT#SSLSECDATA=?	Test command returns the range of supported values for all the parameters: **#SSLSECDATA: (1),(0-2),(0-2),(1-2047)**

3.4.7.4 FTP AT Commands

3.4.7.4.1 *FTP Time – Out - #FTPTO*

#FTPTO - FTP Time-Out	
AT#FTPTO= [<tout>]	Set command sets the time-out used when opening either the FTP control channel or the FTP traffic channel. **Parameter:** <tout> - time-out in 100 ms units 100..5000 - hundreds of ms (factory default is 100) **Note:** The parameter is not saved in NVM.
AT#FTPTO?	Read command returns the current FTP operations time-out, in the format: #FTPTO: <tout>
AT#FTPTO=?	Test command returns the range of supported values

3.4.7.4.2 *FTP Open - #FTPOPEN*

#FTPOPEN - FTP Open	
AT#FTPOPEN= [<server:port>, <username>, <password>, <mode>]	Execution command opens an FTP connection toward the FTP server. **Parameters:** <server:port> string type, address and port of FTP server (factory default port 21), in the format: "ipv4" / "ipv4:port" "ipv6" / "[ipv6]" / "[ipv6]:port" "dynamic_name" / "dynamic_name:port" <username> string type, authentication user identification string for FTP. <password> string type, authentication password for FTP. <mode> 0 - active mode (factory default) 1 - passive mode **Note:** In FTP Open case, the solution dependency limits the maximum time out to

	1200 (120 seconds). The FTPTO value that exceed 1200 is considered as 1200.
	Note: Before opening FTP connection the GPRS must been activated with AT# SGACT
AT#FTPOPEN=?	Test command returns the OK result code.

3.4.7.4.3 *FTP Close - # FTPCLOSE*

# FTPCLOSE - FTP Close	
AT#FTPCLOSE	Execution command closes an FTP connection.
AT#FTPCLOSE=?	Test command returns the OK result code.

3.4.7.4.4 *FTP Config - #FTPCFG*

#FTPCFG – FTP Config	
AT#FTPCFG= <tout>, <IPPignoring> [,<FTPSEn> [,<FTPext>]]	**<tout>** - time-out in 100 ms units 100..5000 – hundreds of ms (factory default is 100) Set command set the time-out used when opening either the FTP control channel or the FTP traffic channel. **Note:** The parameter is not saved in NVM. **Note:** if parameter <tout> is omitted the behavior of Set command is the same as Read command. **<IPPignoring>** 0 - No IP Private ignoring. During a FTP passive mode connection client uses the IP address received from server, even if it is a private IPV4 address. 1 - IP Private ignoring enabled. During a FTP passive mode connection if the server sends a private IP address the client doesn't consider this and connects with server using the IP address used in AT#FTPOPEN. **<FTPSEn>** 0 – Disable FTPS security: all FTP commands will perform plain FTP connections. **< FTPext >** 0 – always use EPRT and EPSV commands(default) 1 – if both module and server ipv4 use PORT and PASV commands Option added to pass-through firewall that is unaware of the extended FTP commands for FTPPUT, FTPLIST, FTPAPP, FTPGET
AT#FTPCFG?	Read command reports the currently selected parameters in the format: AT#FTPCFG=<tout>,<IPPignoring>,<FTPSEn>,<FTPext>

AT#FTPCFG=?	Test command reports the sypported range of values for parameter(s) <tout>,<IPPignoring>,<FTPSEn>,<FTPext>

3.4.7.4.5 *FTP Put - #FTPPUT*

#FTPPUT – FTP Put	
AT#FTPPUT= [[<filename>] [,<connMode>]]	Execution command, issued during an FTP connection, opens a data connection and starts sending <filename> file to the FTP server. If the data connection succeeds, a CONNECT indication is sent, otherwise a NO CARRIER indication is sent. **Note**: if we set <connMode> to 1, the data connection is opened and we remain in command mode and we see the result code OK (instead of CONNECT) **Parameter:** **<filename>** - string type, name of the file (maximum length 200 characters) **<connMode>** 0 – online mode 1 – command mode **Note:** use the escape sequence +++ to close the data connection. **Note:** The command causes an ERROR result code to be returned if no FTP connection has been opened yet.
AT#FTPPUT=?	Test command reports the maximum length of <filename> and the supported range of values of <connMode>. The format is: #FTPPUT:<length>,(list of supported <connMode>s) **where:** **<length>** - integer type value indicating the maximum length of <filename>

3.4.7.4.6 *FTP Get - #FTPGET*

#FTPGET – FTP Get	
AT#FTPGET= [<filename>]	Execution command, issued during an FTP connection, opens a data connection and starts getting a file from the FTP server. If the data connection succeeds a CONNECT indication is sent, otherwise a NO CARRIER indication is sent. The file received on the serial port. **Parameter:**

	<filename> - file name, string type.
	Note: The command causes an ERROR result code to be returned if no FTP connection has been opened yet.
AT#FTPGET=?	Test command returns the OK result code.

3.4.7.4.7 *FTP Get in command mode - #FTPGETPKT*

#FTPGETPKT - FTP Get in command mode	
AT#FTPGETPKT= <filename> [,<viewMode>]	Execution command issued during an FTP connection, opens a data connection and starts getting a file from the FTP server while remaining in command mode. The data port is opened and we remain in command mode and we see the result code OK. Retrieval from FTP server of "remotefile" is started, but data are only buffered in the module. It's possible to read data afterwards issuing #FTPRECV command. **Parameter:** **<filename>** - file name, string type. (maximum length: 200 characters). <viewMode> - permit to choose view mode (text format or Hexadecimal) 0 – text format (default) 1 – hexadecimal format **Note:** The command causes an ERROR result code to be returned in case no FTP connection has been opened yet. **Note**: Command closure should always be handled by application. In order to avoid download stall situations a timeout should be implemented by the application.
AT#FTPGETPKT?	Read command reports current download state for <filename> with <viewMode> chosen, in the format: #FTPGETPKT: <remotefile>,<viewMode>,<eof> <eof> 0 – file currently being transferred 1 – complete file has been transferred to FTP client
AT#FTPGETPKT=?	Test command returns the OK result code.

3.4.7.4.8 *FTP Type - #FTPTYPE*

#FTPTYPE - FTP Type	
AT#FTPTYPE= [<type>]	Set command, issued during an FTP connection, sets the file transfer type.

	Parameter: **\<type\>** - file transfer type: 0 - binary 1 - ascii **Note:** The command causes an ERROR result code to be returned if no FTP connection has been opened yet.
#FTPTYPE?	Read command returns the current file transfer type, in the format: #FTPTYPE: \<type\>
#FTPTYPE=?	Test command returns the range of available values for

3.4.7.4.9 *FTP Delete - #FTPDELE*

#FTPDELE - FTP Delete	
AT#FTPDELE= **[\<filename\>]**	Execution command, issued during an FTP connection, deletes a file from the remote working directory. **Parameter:** **\<filename\>** - string type, it's the name of the file to delete. **Note:** The command causes an ERROR result code to be returned if no FTP connection has been opened yet.
AT#FTPDELE=?	Test command returns the OK result code.

3.4.7.4.10 *FTP Print Working Directory - #FTPPWD*

#FTPPWD - FTP Print Working Directory	
AT#FTPPWD	Execution command, issued during an FTP connection, shows the current working Directory on FTP server. **Note:** The command causes an ERROR result code to be returned if no FTP Connection has been opened yet.
AT#FTPPWD=?	Test command returns the OK result code.

3.4.7.4.11 *FTP Change Working Directory - #FTPCWD*

#FTPCWD - FTP Change Working Directory	
AT#FTPCWD= **[\<dirname\>]**	Execution command, issued during an FTP connection, changes the working Directory on FTP server.

	Parameter: <dirname> - string type, it's the name of the new working directory. **Note:** The command causes an ERROR result code to be returned if no FTP Connection has been opened yet.
AT#FTPCWD=?	Test command returns the OK result code.

3.4.7.4.12 *FTP List - #FTPLIST*

#FTPLIST - FTP List	
AT#FTPLIST[= **[<name>]]**	Execution command, issued during an FTP connection, opens a data connection and starts getting from the server the list of contents of the specified directory or the properties of the specified file. **Parameter:** <name> - string type, it's the name of the directory or file. **Note:** The command causes an ERROR result code to be returned if no FTP connection has been opened yet. **Note**: issuing AT#FTPLIST<CR> opens a data connection and starts getting from the server the list of contents of the working directory.
AT#FTPLIST=?	Test command returns the OK result code.

3.4.7.4.13 *Get file size from FTP - #FTPFSIZE*

#FTPFSIZE – Get file size from FTP	
AT#FTPFSIZE= **<filename>**	Execution command, issued during an FTP connection, permits to get file size of <filename> file. **Note:** #FTPTYPE=0 command has to be issued before #FTPFSIZE command, to set file transfer type to binary mode.
AT#FTPFSIZE=?	Test command returns the OK result code.

3.4.7.4.14 *FTP Append - #FTPAPP*

#FTPAPP – FTP Append	
AT#FTPAPP= **[<filename>** **[,<connMode>]]**	Execution command, issued during an FTP connection, opens a data connection and append data to existing <filename> file.

	If the data connection succedds, a CONNECT indication is sent, Afterward a NO CARRIER indication is sent when the socket is closed. **Note:** if we set <connMode> to 1, the data connection is opened and we remain in command mode and we see the result code OK (instead of CONNECT) **Parameters:** **<filename>** – string type, name of the file. **<connMode>** 0 – online mode 1 – command mode **Note:** use the escape sequence +++ to close the data connection **Note**: The command causes an ERROR result code to be returned if no FTP connection has been opened yet.
AT#FTPAPP=?	Test command reports the maximum length of <filename> and the supported range of values of <connMode>. The format is: **#FTPAPP:<length>,(list of supported <connMode>s)** **where:** **<length>** – integer type value indicating the maximum length of <filename>

3.4.7.4.15   *Set restart position for FTP GET - #FTPREST*

#FTPREST – Set restart position for FTP GET	
AT#FTPREST= **<restartposition>**	Set command sets the restart position for successive #FTPGET (or #FTPGETPKT) command. It permits to restart a previously interrupted FTP download from the selected position in byte. **Parameters:** **<restartposition>** – position in byte of restarting for successive #FTPGET (or #FTPGETPKT) **Note:** It's necessary to issue #FTPTYPE=0 before successive #FTPGET (or #FTPGETPKT) to set binary file transfer type. **Note:** Setting <restartposition> has effect on successive FTP download. After successive successfully initiated #FTPGET (or #FTPGETPKT) command, <restartposition> is automatically reset.

	Note: value set for <restartposition> has effect on next data transfer (data port opened by #FTPGET or #FTPGETPKT). Then <restartposition> value is automatically assigned to 0 for next download.
AT#FTPREST?	Read command returns the current <restartposition> #FTPREST:<restartposition>
AT#FTPREST=?	Test command returns the OK result code.

3.4.7.4.16 *Receive Data In Command Mode - #FTPRECV*

#FTPRECV – Receive Data In Command Mode	
AT#FTPRECV= <blocksize>	Execution command permits the user to transfer at most <blocksize> bytes of remote file, provided that retrieving from the FTP server has been started with a previous #FTPGETPKT command, onto the serial port. This number is limited to the current number of bytes of the remote file which have been transferred from the FTP server. **Parameter:** **<blocksize>** – max number of bytes to read 1..3000 **Note:** it's necessary to have previously opened FTP data port and started download and buffering of remote file through #FTPGETPKT command. **Note:** issuing #FTPRECV when there's no FTP data port opened raises an error. **Note**: data port will stay opened if socket is temporary waiting to receive data (FTPRECV returns 0 and FTPGETPTK gives an EOF 0 indication).
AT#FTPRECV?	Read command reports the number of bytes currently received from FTP server, in the format: #FTPRECV:<available>
AT#FTPRECV=?	Test command reports the supported range of values for parameter < blocksize >
Example	AT#FTPRECV? #FTPRECV: 3000 OK *Read required part of the buffered data:* AT#FTPRECV=400 #FTPRECV:400

	Text row number 1 * 11111111111111111111111111 * Text row number 2 * 22222222222222222222222222 * Text row number 3 * 33333333333333333333333333 * Text row number 4 * 44444444444444444444444444 * Text row number 5 * 55555555555555555555555555 * Text row number 6 * 66666666666666666666666666 * Text row number 7 * 77777777777777777777777777 * Text row number 8 * 88888888888888888888 OK AT#FTPRECV=200 #FTPRECV:200 88888 * Text row number 9 * 99999999999999999999999999 * Text row number 10 * AAAAAAAAAAAAAAAAAAAAAAAAAA* Text row number 11 * BBBBBBBBBBBBBBBBBBBBBBBBBB * Text row number 12 * CCCCCCCCCCCCCCCC OK *Note: to check when you have received complete file it's possible to use AT#FTPGETPKT read command:* AT#FTPGETPKT? #FTPGETPKT:sample.txt,0,1 OK *(you will get <eof> set to 1)*

3.4.7.4.17   *FTP Append Extended - #FTPAPPEXT*

#FTPAPPEXT - FTP Append Extended	
AT#FTPAPPEXT= <bytestosend> [,<eof>]	This command permits to send data on a FTP data port while the module is in command mode. FTP data port has to be previously opened through #FTPPUT (or #FTPAPP) with <connMode> parameter set to command mode connection. **Parameters:** <bytestosend> - number of bytes to be sent 1..1500 <eof> - data port closure

	0 – normal sending of data chunk 1 – close data port after sending data chunk The device responds to the command with the prompt <greater_than><space> and waits for the data to send. When <bytestosend> bytes have been sent, operation is automatically completed. If (all or part of the) data are successfully sent, then the response is: #FTPAPPEXT:<sentbytes> OK Where <sentbytes> are the number of sent bytes. Note: <sentbytes> could be less than <bytestosend> If data sending fails for some reason, an error code is reported.
AT#FTPAPPEXT=?	Test command reports the supported range of values for parameters <**bytestosend**> and <**eof**>
Example	AT#FTPOPEN="IP",username,password OK AT#FTPPUT=<filename>,1 *(the new param 1 means that we open the connection in command mode)* OK *Here data socket will stay opened, but interface will be available (command mode)* AT#FTPAPPEXT=Size >… *write here the binary data. As soon Size byte are written, data are sent and OK is returned* #FTPAPPEXT:<SentBytes> OK …….. *Last #FTPAPPEXT will close the data socket, because second (optional) parameter has this meaning:* AT#FTPAPPEXT=Size,1 >… *write here the binary data. As soon Size byte are written, data are sent and OK is returned* #FTPAPPEXT:<SentBytes> OK *If the user has to reopen the data port to send another (or append to the same) file, he can restart with the FTPPUT (or FTPAPP).* *Then FTPAPPEXT, … to send the data chunks on the reopened data port.*

	Note: if while sending the chunks the data port is closed from remote, user will be aware of it because #FTPAPPEXT will indicatd ERROR and cause (available if previously issued the command AT+CMEE=2) will indicate that socket has been closed. *Also in this case obviously, data port will have to be reopened with FTPPUT and so on...(same sequence)*

3.4.7.4.18 *FTP Read Message - #FTPMSG*

#FTPMSG - FTP Read Message	
AT#FTPMSG	Execution command returns the last response from the server.
AT#FTPMSG=?	Test command returns the OK result code.

3.4.7.5 AT Commands

3.4.7.5.1 *Query DNS - #QDNS*

#QDNS - Query DNS	
AT#QDNS= [<host name>]	Execution command executes a DNS query to solve the host name into an IP address. **Parameter**: **<host name>** - host name, string type. If the DNS query is successful then the IP address will be reported in the result code, as follows: **#QDNS: <host name>,<IP address>** where **<host name>** - string type **<IP address>** - string type, in the format **"xxx.xxx.xxx.xxx"** **Note**: the command has to activate the GPRS context if it was not previously activated. In this case the context is deactivated after the DNS query.
AT#QDNS=?	Test command returns the **OK** result code.
Note	This command requires that the authentication parameters are correctly set and that the GPRS network is present.

3.4.7.5.2 *DNS Response Caching - #CACHEDNS*

#CACHEDNS - DNS Response Caching	
AT#CACHEDNS= [<mode>]	Set command enables caching a mapping of domain names to IP addresses, as does a resolver library. **Parameter:** **<mode>** 0 - caching disabled; it cleans the cache too 1 - caching enabled **Note:** the validity period of each cached entry (i.e. how long a DNS response remains valid) is determined by a value called the **Time To Live** (**TTL**), set by the administrator of the DNS server handing out the response. **Note:** it is recommended to clean the cache, if command **+CCLK** had been issued while the DNS Response Caching was enabled.
AT#CACHEDNS?	Read command reports whether the DNS Response Caching is currently enabled or not, in the format: **#CACHEDNS: <mode>**
AT#CACHEDNS=?	Test command returns the currently cached mapping along with the range of available values for parameter **<mode>**, in the format: **#CACHEDNS: [<hostn1>,<IPaddr1>,[…,[<hostnn>,<IPaddrn>,]]](0,1)** **where:** **<hostnn>** - hostname, string type **<IPaddrn>** - IP address, string type, in the format "xxx.xxx.xxx.xxx"

3.4.7.5.3 *Manual DNS Selection - #DNS*

#DNS – Manual DNS Selection	
AT#DNS=<cid>, <primary>, <secondary>	Set command allows to manually set primary and secondary DNS servers for a PDP context defined by **+CGDCONT.** **Parameters:** **<cid>** - a numeric parameter which specifies a particular PDP context definition (see **+CGDCONT** command) **Ipv4**- **manual primary DNS server**, string type, in the format "xxx.xxx.xxx.xxx" used for the specified cid; we're using this value instead of the **primary DNS server** come from the network (default is "0.0.0.0") **Ipv6**- **manual primary DNS server**, string type, in the format "xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx" used for the specified cid; we're using this value instead of the **primary DNS server** come from the network (default is "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0"). Ipv6 can also be in HEX format: "**xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx**" **<secondary>**

#DNS – Manual DNS Selection	
	Ipv4- **manual primary DNS server**, string type, in the format **"xxx.xxx.xxx.xxx"** used for the specified cid; we're using this value instead of the **primary DNS server** come from the network (default is "0.0.0.0") **Ipv6**- **manual primary DNS server**, string type, in the format **"xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx"** used for the specified cid; we're using this value instead of the **primary DNS server** come from the network (default is "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0"). Ipv6 can also be in HEX format: **"xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx"** **Note:** if **<primary> is "0.0.0.0"** and **<secondary> is not "0.0.0.0"**, then issuing **AT#DNS=…** raises an error. **Note:** if **<primary> is "0.0.0.0"** were using the **primary DNS server** come from the network as consequence of a context activation. **Note:** if **<primary> is not "0.0.0.0"** and **<secondary> is "0.0.0.0"**, then were using only the **manual primary DNS server.** **Note:** the context identified by **<cid>** has to be previously defined, elsewhere issuing **AT#DNS=…** raises an error. **Note:** the context identified by **<cid>** has to be not activated yet, elsewhere issuing **AT#DNS=…** raises an error.
AT#DNS?	Read command returns the manual DNS servers set either for every defined PDP context and for the single GSM context (only if defined), in the format: **[#DNS: <cid>,<primary>,<secondary>[<CR><LF>** **#DNS: <cid>,<primary>,<secondary>]]** **In case +cgdcont** determined as ipv4v6 the format is **[#DNS: <cid>,<primary ip4>,<primary ip6>,<secondary ip4>,<secondary ip6>[<CR><LF>** **#DNS: <cid>,<primary ip4>,<primary ip6>,<secondary ip4>,<secondary ip6>]]**
AT#DNS=?	Test command reports the supported range of values for the **<cid>** parameter only, in the format: **#DNS: (1-5),,**

3.4.7.5.4 *Socket Listen Ring - #E2SLRI*

#E2SLRI - Socket Listen Ring	
AT#E2SLRI=[<n>]	Set command enables/disables the Ring Indicator pin response to a Socket Listen connect and if enabled the duration of the negative going pulse generated on receipt of connect. **Parameter:** **<n>** - RI enabling 0 - RI disabled for Socket Listen connect (factory default) 50..1150 - RI enabled for Socket Listen connect; a negative going pulse is generated on receipt of connect and <n> is the duration in ms of this pulse
AT#E2SLRI?	Read command reports whether the Ring Indicator pin response to a Socket Listen connect is currently enabled or not, in the format: #E2SLRI: <n>
AT#E2SLRI=?	Test command returns the allowed values for parameter <status>.

3.4.7.5.5 *Firewall Setup - #FRWL*

#FRWL - Firewall Setup	

AT#FRWL=
[<action>,
<ip_address>,
<net mask>]
	Execution command controls the internal firewall settings.
	
	Parameters:
	<action> - command action
	  0 - remove selected chain
	  1 - add an **ACCEPT** chain
	  2 - remove all chains (**DROP** everything); **<ip_addr>** and **<net_mask>** has no meaning in this case.
	<ip_addr> - remote address to be added into the **ACCEPT** chain; string type, it can be any valid IP address in the format: xxx.xxx.xxx.xxx
	<net_mask> - mask to be applied on the **<ip_addr>**; string type, it can be any valid IP address mask in the format: xxx.xxx.xxx.xxx
	
	Command returns **OK** result code if successful.
	
	Note: the firewall applies for incoming (listening) connections only.
	Firewall general policy is **DROP**, therefore all packets that are not included into an **ACCEPT** chain rule will be silently discarded.
	When a packet comes from the IP address **incoming_IP**, the firewall chain rules will be scanned for matching with the following criteria:
	
	incoming_IP & <net_mask> = <ip_addr> & <net_mask>

#FRWL - Firewall Setup	
	If criteria matched, then the packet is accepted and the rule scan is finished; if criteria not matched for any chain the packet silently dropped.
AT#FRWL?	Read command reports the list of all **ACCEPT** chain rules registered in the Firewall settings in the format: **#FRWL: <ip_addr>,<net_mask>** **#FRWL: <ip_addr>,<net_mask>** **….** **OK**
AT#FRWL=?	Test command returns the allowed values for parameter **<action>.**
Example	*Let assume we want to accept connections only from our devices which are on the IP addresses ranging from* *197.158.1.1 to 197.158.255.255* *We need to add the following chain to the firewall:* AT#FRWL=1,"197.158.1.1","255.255.0.0" OK

3.4.7.5.6 *GPRS Data Volume - #GDATAVOL*

#GDATAVOL - GPRS Data Volume	

AT#GDATAVOL=
[<mode>]
	Execution command reports, for every active PDP context, the amount of data the last GPRS session received and transmitted, or it will report the total amount of data received and transmitted during all past GPRS sessions, since last reset.
	
	Parameter:
	<mode>
	 0 - it resets the GPRS data counter for the all the available PDP contexts (1-16)
	 1 - it reports the last GPRS session data counter for the all the set PDP contexts (i.e. all the PDP contexts with APN parameter set using +CGDCONT), in the format:
	
	  #GDATAVOL: <cid*n*>,<tot*n*>,<sent*n*>,<received*n*>[<CR><LF>
	  #GDATAVOL: <cid*m*>,<tot*m*>,<sent*m*>,<received*m*>[…]]
	
	 where:
	   <cid*n*> - PDP context identifier
	    1..16 - numeric parameter which specifies a particular PDP context definition
	   <tot*n*> - number of bytes either received or transmitted in the last GPRS session for **<cid*n*>** PDP context;
	   <sent*n*> - number of bytes transmitted in the last GPRS session for **<cid*n*>** PDP context;
	   <received*n*> - number of bytes received in the last GPRS session for **<cid*n*>** PDP context;

#GDATAVOL - GPRS Data Volume	
	2 - it reports the total GPRS data counter, since last reset, for the all the set PDP contexts (i.e. all the PDP context with APN parameter set using **+CGDCONT**), in the format: **#GDATAVOL: <cid*n*>,<tot*n*>,<sent*n*>,<received*n*>[<CR><LF>** **#GDATAVOL: <cid*m*>,<tot*m*>,<sent*m*>,<received*m*>[…]]** **where:** **<cid*n*>** - PDP context identifier 1..16 - numeric parameter which specifies a particular PDP context definition **<tot*n*>** - number of bytes either received or transmitted, in every GPRS session since last reset, for **<cid*n*>** PDP context; **<sent*n*>** - number of bytes transmitted, in every GPRS session since last reset, for **<cid*n*>** PDP context; **<received*n*>** - number of bytes received, in every GPRS session since last reset, for **<cid*n*>** PDP context; **Note:** last GPRS session counters not saved in NVM so they are loosen at power off. **Note:** total GPRS session counters saved on NVM.
AT#GDATAVOL=?	Test command returns the range of supported values for parameter **<mode>.**

3.4.7.5.7 *DNS from Network - #NWDNS*

#NWDNS – DNS from Network	
AT#NWDNS= **[<cid>[,<cid>** **[,…]]]**	Execution command returns either the primary and secondary DNS addresses for the GSM context (if specified) and/or a list of primary and secondary DNS addresses for the specified PDP context identifiers. **Parameters:** **<cid>** - a numeric parameter which specifies a particular PDP context definition (see +**CGDCONT** command) **Note:** if no <cid> is specified, the DNS addresses for all defined contexts are returned. **Note:** issuing the command with more than 6 parameters raises an error. **Note:** the command returns only one row of information for every specified <cid>, even if the same <cid> is present more than once. The command returns a row of information for every specified <cid> whose context has been already defined. No row is returned for a <cid> whose context has not been defined yet. Response format is: **#NWDNS:** <cid>,<PDNSaddress>,<SDNSaddress>[<CR><LF> **#NWDNS:** <cid>,<PDNSaddress>,<SDNSaddress> […]] **where:** **<cid>** - context identifier, as before <PDNSaddress>,<SDNSaddress> - primary and secondary DNS addresses set through AT#DNS command. If not set, they are

#NWDNS – DNS from Network	
	the primary and secondary DNS addresses assigned during the PDP (or GSM) context activation.
AT#NWDNS=?	Test command returns a list of defined <cid>s.

3.4.7.6 Easy Scan® Extension AT Commands

Note: it is strongly suggested to issue all the Easy Scan® Extension AT commands with NO SIM inserted, to avoid a potential conflict with normal module operations, such as "incoming call", "periodic location update, "periodic routing area update" and so on.

3.4.7.6.1 *Network survey - #CSURV*

#CSURV - Network Survey	
AT#CSURV[= [<s>,<e>]]	Execution command allows to perform a quick survey through channels belonging to the band selected by last **#BND** command issue, starting from channel **<s>** to channel **<e>**. Issuing **AT#CSURV<CR>**, a full band scan is performed. **Parameters:** **<s>** - starting channel **<e>** - ending channel After issuing the command the device responds with the string: **Network survey started…** and, after a while, a list of informations, one for each received carrier, is reported, each of them in the format: In 2G **(For BCCH-Carrier)** **arfcn:** <arfcn> bsic: <bsic> rxLev: <rxLev> ber: <ber> mcc: <mcc> mnc: <mnc> lac: <lac> cellId: <cellId> cellStatus: <cellStatus> numArfcn: <numArfcn> arfcn: [<arfcn1> ..[<arfcn64>]] [numChannels: <numChannels> array: [<ba1> ..[<ba32>]] [pbcch: <pbcch> [nom: <nom> rac: <rac> spgc: <spgc> pat: <pat> nco: <nco> t3168: <t3168> t3192: <t3192> drxmax: <drxmax> ctrlAck: <ctrlAck> bsCVmax: <bsCVmax> alpha: <alpha> pcMeasCh: <pcMeasCh>]]] <CR><LF><CR><LF><CR><LF>

where:
\<arfcn\> - C0 carrier assigned radio channel (BCCH - Broadcast Control Channel).
\<bsic\> - base station identification code; if #CSURVF last setting is 0, **\<bsic\>** is a decimal number, else it is a 2-digits octal number.
\<rxLev\> - decimal number; it is the reception level (in dBm).
\<ber\> - decimal number; it is the bit error rate (in %).
\<mcc\> - hexadecimal 3-digits number; it is the mobile country code.
\<mnc\> - hexadecimal 2-digits number; it is the mobile network code.
\<lac\> - location area code; if #CSURVF last setting is 0, **\<lac\>** is a decimal number, else it is a 4-digits hexadecimal number.
\<cellId\> - cell identifier; if #CSURVF last setting is 0, **\<cellId\>** is a decimal number, else it is a 4-digits hexadecimal number.
\<cellStatus\> - string type; it is the cell status
..**CELL_SUITABLE** - C0 is a suitable cell.
CELL_LOW_PRIORITY - the cell is low priority based on the received system information.
CELL_FORBIDDEN - the cell is forbidden.
CELL_BARRED - the cell is barred based on the received system information.
CELL_LOW_LEVEL - the cell **\<rxLev\>** is low.
CELL_OTHER - none of the above e.g. exclusion timer running, no BCCH available…etc.
\<numArfcn\> - number of valid channels in the Cell Channel Description.
\<arfcn*n*\> - arfcn of a valid channel in the Cell Channel Description (*n* is in the range **1..\<numArfcn\>**)
\<numArfcn\> - decimal number; it is the number of valid channels in the Cell Channel Description.
\<arfcn*n*\> - decimal number; it is the arfcn of a valid channel in the Cell Channel Description (*n* is in the range **1..\<numArfcn\>**).
\<numChannels\> - decimal number; it is the number of valid channels in the BCCH Allocation list; the output of this information for non-serving cells depends on last #CSURVEXT setting:
If **#CSURVEXT=0** this information is displayed only for serving cell.
If **#CSURVEXT=1** or **2** this information is displayed also for every valid scanned BCCH carrier.
\<ba*n*\> - decimal number; it is the arfcn of a valid channel in the BA list (*n* is in the range **1..\<numChannels\>**); the output of this information for non-serving cells depends on last #CSURVEXT setting:
Iif **#CSURVEXT=0** this information is displayed only for serving cell.
If **#CSURVEXT=1** or **2** this information is displayed also for every valid scanned BCCH carrier.

(The following informations will be printed only if GPRS is supported in the cell)
\<pbcch\> - packet broadcast control channel
0 - pbcch not activated on the cell
1 - pbcch activated on the cell

<nom> - network operation mode
1
2
3
<rac> - routing area code
0..255 -
<spgc> - SPLIT_PG_CYCLE support
..0 - SPLIT_PG_CYCLE is not supported on CCCH on this cell
..1 - SPLIT_PG_CYCLE is supported on CCCH on this cell
<pat> - priority access threshold
0 -
3..6 -
<nco> - network control order
0..2 -
<t3168> - timer 3168
<t3192> - timer 3192
<drxmax> - discontinuous reception max time (in seconds)
<ctrlAck> - packed control ack
<bsCVmax> - blocked sequenc countdown max value
<alpha> - alpha parameter for power control
<pcMeasCh> - type of channel which shall be used for downlink measurements for power control
0 - BCCH
1 - PDCH
(For non BCCH-Carrier)
arfcn: <arfcn> rxLev: <rxLev>

where:
<arfcn> - decimal number; it is the RF channel
<rxLev> - decimal number; it is the receiption level (in dBm)
In 3G
uarfcn: <uarfcn> rxLev: <rxLev> mcc: <mcc> mnc: <mnc> scr code: <scrcode> cellId: <cellId> lac: <lac> cellStatus: <cellStatus>
<CR><LF><CR><LF><CR><LF>
where:
<uarfcn> - The carrier frequency is designated by the UTRA Absolute Radio Frequency Channel Number.
<rxLev> - decimal number; it is the receiption level (in dBm).
<mcc> - hexadecimal 3-digits number; it is the mobile country code.
<mnc> - hexadecimal 2-digits number; it is the mobile network code.
<scrcode> - decimal number; it is the scrambling code
<cellId> - cell identifier; if #CSURVF last setting is 0, **<cellId>** is a decimal number, else it is a 4-digits hexadecimal number.

	<lac> - location area code; if #CSURVF last setting is 0, **<lac>** is a decimal number, else it is a 4-digits hexadecimal number. **<cellStatus>** - string type; it is the cell status. **CELL_SUITABLE** - C0 is a suitable cell. **CELL_LOW_PRIORITY** - the cell is low priority based on the received system information. **CELL_FORBIDDEN** - the cell is forbidden. **CELL_BARRED** - the cell is barred based on the received system information. **CELL_LOW_LEVEL** - the cell **<rxLev>** is low. **CELL_OTHER** - none of the above e.g. exclusion timer running, no BCCH available…etc. In 4G (partly implemented) Currently work only if module camped on LTE cell. For serving cell: earfcn: <earfcn> rxLev: <rxLev> mcc: <mcc> mnc: <mnc> cellId: <cellId> tac: <tac> For neighbor cell: earfcn: <earfcn> rxLev: <rxLev> cellId: <cellId> **Where:** <earfcn> - E-UTRA Assigned Radio Channel <tac> - Tracking Area Code. if #CSURVF last setting is 0, <cellId> is a decimal number, else it is a 4-digits hexadecimal number Lastly, the **#CSURV** output ends in two ways, depending on the last **#CSURVF** setting: if **#CSURVF=0** or **#CSURVF=1** The output ends with the string: **Network survey ended** if **#CSURVF=2** the output ends with the string: **Network survey ended (Carrier: <NoARFCN> BCCh: <NoBCCh>)** **Where**: **<NoARFCN>** - number of scanned frequencies **<NoBCCH>** - number of found BCCh
Example	(2G) AT#CSURV Network survey started… arfcn: 48 bsic: 24 rxLev: -52 ber: 0.00 mcc: 610 mnc: 1 lac: 33281 cellId: 3648 cellStatus: CELL_SUITABLE numArfcn: 2 arfcn: 30 48 numChannels: 5 array: 14 19 22 48 82 arfcn: 14 rxLev: 8 Network survey ended

	OK
	(wcdma)
	at#csurv
	Network survey started ...
	uarfcn: 10812 rxLev: -87 mcc: 450 mnc: 08 scr code: 6528 cellId: 10683976 lac: 5121 cellStatus: CELL_LOW_PRIORITY
	uarfcn: 10713 rxLev: -87 mcc: 450 mnc: 05 scr code: 1200 cellId: 2171648 lac: 8209 cellStatus: CELL_LOW_PRIORITY
	Network survey ended
	OK
Note	1. The command is executed within max. 2 minute.

3.4.7.6.2 *Network Survey (Numeric Format) - #CSURVC*

#CSURVC - Network Survey (Numeric Format)	
AT#CSURVC[=[<s>,<e>]]	Execution command allows to perform a quick survey through channels belonging to the band selected by last #BND command issue, starting from channel **<s>** to channel **<e>**. Issuing **AT#CSURVC<CR>**, a full band scan is performed. **Parameters:** **<s>** - starting channel **<e>** - ending channel After issuing the command the device responds with the string: **Network survey started…** and, after a while, a list of informations, one for each received carrier, is reported, each of them in the format: In 2G **(For BCCH-Carrier)** **<arfcn>,<bsic>,<rxLev>,<ber>,<mcc>,<mnc>,<lac>,<cellId>,** **<cellStatus>,<numArfcn>[,<arfcn1> ..[<arfcn64>]]** **[,<numChannels>[,<ba1> ..[<ba32>]]][,<pbcch>[,<nom>,<rac>,<spgc> <pat>** **<nco> <t3168> <t3192> <drxmax> <ctrlAck>** **<bsCVmax>,<alpha>,<pcMeasCh>]]]** **<CR><LF><CR><LF><CR><LF>** **where:** **<arfcn>** - C0 carrier assigned radio channel (BCCH - Broadcast Control Channel). **<bsic>** - base station identification code; if #CSURVF last setting is 0. **<bsic>** is a decimal number, else it is a 2-digits octal number. **<rxLev>** - decimal number; it is the receiption level (in dBm). **<ber>** - decimal number; it is the bit error rate (in %). **<mcc>** - hexadecimal 3-digits number; it is the mobile country code. **<mnc>** - hexadecimal 2-digits number; it is the mobile network code.

<lac> - location area code; if #CSURVF last setting is 0, **<lac>** is a decimal number, else it is a 4-digits hexadecimal number.

<cellId> - cell identifier; if #CSURVF last setting is 0, **<cellId>** is a decimal number, else it is a 4-digits hexadecimal number.

<cellStatus> - string type; it is the cell status

..0 - C0 is a suitable cell (CELL_SUITABLE).

1 - the cell is low priority based on the received system information (CELL_LOW_PRIORITY).

2 - the cell is forbidden (CELL_FORBIDDEN).

3 - the cell is barred based on the received system information (CELL_BARRED).

4 - the cell **<rxLev>** is low (CELL_LOW_LEVEL).

5 - none of the above e.g. exclusion timer running, no BCCH available…etc.. (CELL_OTHER).

<numArfcn> - decimal number; it is the number of valid channels in the Cell Channel Description

<arfcn*n*> - decimal number; it is the arfcn of a valid channel in the Cell Channel Description (*n* is in the range **1..<numArfcn>**)

<numChannels> - decimal number; it is the number of valid channels in the BCCH Allocation list; the output of this information for non-serving cells depends on last #CSURVEXT setting:

If **#CSURVEXT=0** this information is displayed only for serving cell

If **#CSURVEXT=1 or 2** this information is displayed also for every valid scanned BCCH carrier.

<ba*n*> - decimal number; it is the arfcn of a valid channel in the BA list (*n* is in the range **1..<numChannels>**); the output of this information for non-serving cells depends on last #CSURVEXT setting:

If **#CSURVEXT=0** this information is displayed only for serving cell

If **#CSURVEXT=1 or 2** this information is displayed also for every valid scanned BCCH carrier.

(The following informations will be printed only if GPRS is supported in the cell)

<pbcch> - packet broadcast control channel

0 - pbcch not activated on the cell

1 - pbcch activated on the cell

<nom> - network operation mode

1

2

3

<rac> - routing area code

0..255

<spgc> - SPLIT_PG_CYCLE support

..0 - SPLIT_PG_CYCLE is not supported on CCCH on this cell

..1 - SPLIT_PG_CYCLE is supported on CCCH on this cell

<pat> - priority access threshold.

	0 3..6 **<nco>** - network control order. 0..2 - **<t3168>** - timer 3168. **<t3192>** - timer 3192. **<drxmax>** - discontinuous reception max time (in seconds). **<ctrlAck>** - packed control ack. **<bsCVmax> -** blocked sequenc countdown max value. **<alpha> -** alpha parameter for power control. **<pcMeasCh> -** type of channel which shall be used for downlink measurements for power control. 0 - BCCH 1 - PDCH **(For non BCCH-Carrier)** **<arfcn>,<rxLev>** **where:** **<arfcn>** - decimal number; it is the RF channel **<rxLev>** - decimal number; it is the reception level (in dBm) In 3G **<uarfcn>,<rxLev>,<mcc>,<mnc>,<scrcode>,<cellId>,<lac>,<cellStatus>** **<CR><LF><CR><LF><CR><LF>** **where:** **<uarfcn>** - The carrier frequency is designated by the UTRA Absolute Radio Frequency Channel Number **<rxLev>** - decimal number; it is the reception level (in dBm) **<mcc>** - hexadecimal 3-digits number; it is the mobile country code **<mnc>** - hexadecimal 2-digits number; it is the mobile network code **<scrcode>** - decimal number; it is the scrambling code **<cellId>** - cell identifier; if #CSURVF last setting is 0, **<cellId>** is a decimal number, else it is a 4-digits hexadecimal number **<lac>** - location area code; if #CSURVF last setting is 0, **<lac>** is a decimal number, else it is a 4-digits hexadecimal number **<cellStatus>** - string type; it is the cell status ..CELL_SUITABLE - C0 is a suitable cell. **CELL_LOW_PRIORITY** - the cell is low priority based on the received system information. **CELL_FORBIDDEN** - the cell is forbidden. **CELL_BARRED** - the cell is barred based on the received system information. **CELL_LOW_LEVEL** - the cell **<rxLev>** is low. CELL_OTHER - none of the above e.g. exclusion timer running, no BCCH available…etc. In 4G (partly implemented)

	Currently work only if module camped on LTE cell. For serving cell: <earfcn>,<rxLev, <mcc>,<mnc>, <cellId>,<tac> For neighbor cell: <earfcn>,<rxLev>,<cellId> **Where:** **<earfcn>** - E-UTRA Assigned Radio Channel **<tac>** - Tracking Area Code. if #CSURVF last setting is 0, **<cellId>** is a decimal number, else it is a 4-digits hexadecimal number The last information from **#CSURVC** depends on the last **#CSURVF** setting: When **#CSURVF=0** or **#CSURVF=1** The output ends with the string "**Network survey ended**" when **#CSURVF=2** the output ends with the string "**Network survey ended (Carrier: <NoARFCN> BCCh: <NoBCCh>)** **Where:** **<NoARFCN>** - number of scanned frequencies **<NoBCCH>** - number of found BCCh
Example	AT#CSURVC Network survey started… 48,24,-52,0.00,610,1,33281,3648,0,2,30 48,5,14 19 22 48 82 14,8 Network survey ended OK
Note	The command is executed within max. 2 minute. The information provided by #CSURVC is the same as that provided by #CSURV. The difference is that the output of #CSURVC is in numeric format only.

3.4.7.6.3 *Network Survey Format - #CSURVF*

#CSURVF - Network Survey Format	
AT#CSURVF= [<format>]	Set command controls the format of the numbers output by all the Easy Scan®. **Parameter:** **<format>** - numbers format 0 - Decimal 1 - Hexadecimal values, no text (for formats 0 and 1 - the output ends with the string: "Network survey ended") 2 - Hexadecimal values with text the output ends with the string: Network survey ended (Carrier: <NoARFCN> BCCh: <NoBCCh>)

	Where: **<NoARFCN>** - number of scanned frequencies **<NoBCCH>** - number of found BCCh
AT#CSURVF?	Read command reports the current number format, as follows: #CSURVF: <format>
AT#CSURVF=?	Test command reports the supported range of values for the parameter <format>.

3.4.7.6.4 *<CR><LF> Removing On Easy Scan® Commands Family -#CSURVNLF*

#CSURVNLF - <CR><LF> Removing On Easy Scan® Commands Family	
AT#CSURVNLF= [<value>]	Set command enables/disables the automatic <CR><LF> removing from each information text line. **Parameter:** **<value>** 0 - disables <CR><LF> removing; they'll be present in the information text (factory default) 1 - remove <CR><LF> from imformation text
AT#CSURVNLF?	Read command reports whether automatic <CR><LF> removing is currently enabled or not, in the format: <value>
AT#CSURVNLF=?	Test command reports the range of values for parameter <value>.

3.4.7.6.5 *Extended network survey - #CSURVEXT*

#CSURVEXT - Extended network survey	

**AT#CSURVEXT
[=<value>]**
	Set command enables/disables extended network survey.
	
	Parameter:
	<value>
	  0 - disables extended network survey (factory default)
	  1 - enables extended network survey; all the network survey execution commands (**#CSURV, #CSURVC**) display the BAList for every valid scanned BCCh carrier
	  2 - enables extended network survey; all the network survey execution commands (**#CSURV, #CSURVC**) display the BAList for every valid scanned BCCh carrier and, if GPRS is supported in the cell, they report some GPRS informations carried by the System Information 13 of the BCCh
	  3 - enables more extended network survey**;** all the network survey execution commands (**#CSURV, #CSURVC**). It displays transmit power level, receiving level access min, Cell Reselection Offset, Penalty Time, T3212 Periodic Location Update Timer and Cell Reselection Offset

#CSURVEXT - Extended network survey	
AT#CSURVEXT?	Read command reports whether extended network survey is currently enabled or not, in the format: **<value>**
AT#CSURVEXT=?	Test command reports the range of values for parameter **<value>.**
Notes and Platform limits	**#CSURVEXT** configuration has effect on 2G cells only.

3.4.7.7 HTTP AT COMMANDS

3.4.7.7.1 *Configure HTTP parameters - #HTTPCFG*

#HTTPCFG – configure HTTP parameters	
AT#HTTPCFG=<prof_id>[,<server_address>[,<server_port>[,<auth_type>[,<username>[,<password>[,<ssl_enabled>[,<timeout>[,<cid>]]]]]]]]	This command sets the parameters needed to the HTTP connection **Parameters:** **<prof_id>** - Numeric parameter indicating the profile identifier. Range: 0-2 **<server_address>** - String parameter indicating the IP address of the HTTP server. This parameter can be either: - any valid IP address in the format: "xxx.xxx.xxx.xxx" - any valid IPv6 address in the format: "xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx.xxx" Or "XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX" - any host name to be solved with a DNS query Default: "" for first and second profile; "m2mlocate.telit.com" for third profile. **<server_port>** - Numeric parameter indicating the TCP remote port of the HTTP server to connect to. Default: 80 for first and second profile; 9978 for third profile. Range 1...65535. **<auth_type>** - Numeric parameter indicating the HTTP authentication type. 0 – no authentication (default) 1 – basic authentication **<username>** - String parameter indicating authentication user identification string for HTTP. **<password>** - String parameter indicating authentication password for HTTP.

#HTTPCFG – configure HTTP parameters	
	<ssl_enabled> - Numeric parameter indicating if the SSL encryption is enabled. 0 – SSL encryption disabled (default) 1 – SSL encryption enabled (not yet implemented and not available for setting) **<timeout>** - Numeric parameter indicating the time interval in seconds to wait for receiving data from HTTP server. Range: (1- 65535). Default: 120. **<cid>** - a numeric parameter which specifies a particular PDP context definition (see +CGDCONT command) **Note:** a special form of the Set command, #HTTPCFG=<prof_id>, causes the values for profile number <prof_id> to reset to default values. **Note:** if the SSL encryption is enabled, the <cid> parameter has to be set to 1. **Note:** only one profile can use the SSL encryption. **Note:** values are automatically saved in NVM.
AT#HTTPCFG?	Read command returns the current settings for each defined profile in the format: #HTTPCFG: <prof_id>,<server_address>,<server_port>,<auth_type>,<username>,<password>,<ssl_enabled>,<timeout>,<cid><CR><LF>[<CR><LF>#HTTPCFG: <prof_id>,<server_address>,<server_port>,<auth_type>,<username>,<password>,<ssl_enabled>,<timeout>,<cid>]<CR><LF>[…]]
AT#HTTPCFG =?	Test command returns the supported range of parameters <prof_id>, <server_port>, <auth_type>, <ssl_enabled>, <timeout> and <cid> and the maximum length of <server_address>, <username> and <password> parameters in the format: #HTTPCFG: (list of supported <prof_id>s),<s_length>,(list of supported <server_port>s), (list of supported <auth_type>s),<u_length>,<p_length>,(list of supported <ssl_enabled>s),(list of supported <timeout>s),(list of supported <cid>s) **where:** **<s_length>** - integer type value indicating the maximum length of parameter <server_address>. **<u_length>** - integer type value indicating the maximum length of parameter <username>. **<p_length>** - integer type value indicating the maximum length of parameter <password>.

3.4.7.7.2 *Send HTTP GET, HEAD or DELETE request - #HTTPQRY*

#HTTPQRY – send HTTP GET, HEAD or DELETE request	
AT#HTTPQRY= <prof_id>,<command >,<resource>[,<extra_ header_line>]	Execution command performs a GET, HEAD or DELETE request to HTTP server. **Parameters:** **<prof_id>** - Numeric parameter indicating the profile identifier - Range: 0-2 **<command>** - Numeric parameter indicating the command requested to HTTP server: 0 – GET 1 – HEAD 2 – DELETE **<resource>** - String parameter indicating the HTTP resource (uri), object of the request. **<extra_header_line>** - String parameter indicating optional HTTP header line. If sending ends successfully, the response is OK; otherwise an error code is reported. **Note:** the HTTP request header sent with #HTTPQRY always contains the "Connection: close" line, and it cannot be removed. When the HTTP server answer is received, then the following URC is put on the serial port: #HTTPRING: <prof_id>,<http_status_code>,<content_type>,<data_size> **Where:** **<prof_id>** is defined as above. **<http_status_code>** is the numeric status code, as received from the server (see RFC 2616). **<content_type>** is a string reporting the "Content-Type" header line, as received from the server (see RFC 2616). **<data_size>** is the byte amount of data received from the server. If the server doesn't report the "Content-Length:" header line, the parameter value is 0. **Note:** if there are no data from server or the server doesn't answer within the time interval specified in <timeout> parameter of #HTTPCFG command, then the URC #HTTPRING <http_status_code> parameter has value 0.
AT#HTTPQRY=?	Test command reports the supported range of values for the parameters <prof_id> and <command> and the maximum length of <resource> parameter in the format: #HTTPQRY: (list of supported <prof_id>s),(list of supported <command>s),<r_length>,<m_length> **where:**

#HTTPQRY – send HTTP GET, HEAD or DELETE request	
	<r_length> - integer type value indicating the maximum length of parameter <resource>. **<m_length>** - integer type value indicating the maximum length of parameter <extra_header_line>.

3.4.7.7.3 *Send HTTP POST or PUT request - #HTTPSND*

#HTTPSND – send HTTP POST or PUT request	
AT#HTTPSND= **<prof_id>,<command >,<resource>,<data_le n>[,<post_param>[,<e xtra_header_line>]]**	Execution command performs a POST or PUT request to HTTP server and starts sending data to the server. The device shall prompt a three character sequence <greater_than><greater_than><greater_than> (IRA 62, 62, 62)after command line is terminated with <CR>; after that the data can be entered from TE, sized <data_len> bytes. **Parameters:** **<prof_id>** - Numeric parameter indicating the profile identifier. Range: 0-2 **<command>** - Numeric parameter indicating the command requested to HTTP server: 0 – POST 1 – PUT **<resource>** - String parameter indicating the HTTP resource (uri), object of the request **<data_len>** - Numeric parameter indicating the data length to input in bytes **<post_param>** - Numeric/string parameter indicating the HTTP Content-type identifier, used only for POST command, optionally followed by colon character (:) and a string that extends with sub-types the identifier: **"0[:extension]"** – "application/x-www-form-urlencoded" with optional extension **"1[:extension]"** – "text/plain" with optional extension **"2[:extension]"** – "application/octet-stream" with optional extension **"3[:extension]"** – "multipart/form-data" with optional extension other content – free string corresponding to other content type and possible sub-types **<extra_header_line>** - String parameter indicating optional HTTP header line If sending ends successfully, the response is OK; otherwise an error code is reported.

#HTTPSND – send HTTP POST or PUT request	
	Note: the HTTP request header sent with #HTTPSND always contains the "Connection: close" line, and it cannot be removed. When the HTTP server answer is received, then the following URC is put on the serial port: **#HTTPRING:** <prof_id>,<http_status_code>,<content_type>,<data_size> **Where:** **<prof_id>** is defined as above **<http_status_code>** is the numeric status code, as received from the server (see <u>RFC 2616</u>) **<content_type>** is a string reporting the "Content-Type" header line, as received from the server (see RFC 2616) **<data_size>** is the byte amount of data received from the server. If the server doesn't report the "Content-Length:" header line, the parameter value is 0. **Note:** if there are no data from server or the server doesn't answer within the time interval specified in <timeout> parameter of #HTTPCFG command, then the URC #HTTPRING <http_status_code> parameter has value 0.
AT#HTTPSND=?	Test command returns the supported range of parameters <prof_id>, <command> and <data_len> and the maximum length of <resource>, <post_param> and <extra_header_line> parameters in the format: # HTTPSND: (list of supported <prof_id>s),(list of supported <command>s), <r_length>, (list of supported <data_len>s),<p_length>,<m_length> **where:** **<r_length>** - integer type value indicating the maximum length of parameter <resource>. **<p_length>** - integer type value indicating the maximum length of parameter <post_param>. **<m_length>** - integer type value indicating the maximum length of parameter <extra_header_line>
Example	*Post 100 byte without "Content-type" header* AT#HTTPSND=0,0,"/",100 >>> *Post 100 byte with "application/x-www-form-urlencoded"* AT#HTTPSND=0,0,"/",100,0 >>> *Post 100 byte with "multipart/form-data" and extension*

#HTTPSND – send HTTP POST or PUT request	
	AT#HTTPSND=0,0,"/",100,"3:boundary=----FormBoundary" >>>

3.4.7.7.4 Receive HTTP server data - #HTTPRCV

#HTTPRCV – Receive HTTP server data	
AT#HTTPRCV= <prof_id>, [,<maxByte>]	Execution command permits the user to read data from HTTP server in response to a previous HTTP module request. The module is notified of these data by the #HTTPRING URC. The device shall prompt a three character sequence <less_than><less_than><less_than> (IRA 60, 60, 60) followed by the data. If reading ends successfully, the response is OK; otherwise an error code is reported. **Parameters:** **<prof_id>** - Numeric parameter indicating the profile identifier. Range: 0-2 **< maxByte >** - Max number of bytes to read at a time Range: 0,64-1500 (default is 0 which means infinite size) **Note**: if <maxByte> is unspecified, server data will be transferred all in once. **Note**: If the data are not present or the #HTTPRING <http_status_code> parameter has value 0, an error code is reported.
AT#HTTPRCV=?	Test command reports the supported range of values for <prof_id>,<maxbyte> parameter in the format: **#HTTPRCV: (list of supported <prof_id>s,<maxbyte>)**

3.4.7.8 GPS AT Commands Set

3.4.7.8.1 *GPS Power Management - $GPSP*

$GPSP – GPS Power Management	
AT$GPSP= <status>	Set command allows to manage power-up or down of the GPS controller. **Parameter:** **<status>** 0 - GPS controller is powered down (default) 1 - GPS controller is powered up
AT$GPSP?	Read command reports the current value of the <status> parameter, in the format: **$GPSP: <status>**
AT$GPSP=?	Test command reports the range of supported values for parameter <status>
Example	AT$GPSP=0 OK
Note	- Power up clears GPS memory and then starts the GPS receiver. GPS data cleaning is performed on the base of the current value of the **<reset_type>** parameter (see $GPSR). - GPS operation mode is performed on the base of the current values of $GPSSLSR configuration (see $GPSSLSR). - **<status>** value is stored through $GPSSAV command.

3.4.7.8.2 *GPS Module Reset - $GPSR*

$GPSR – GPS module Reset	
AT$GPSR= <reset_type>	Execution command allows to reset the GPS controller. **Parameter:** **<reset_type>** **0** – **Factory Reset**: this option clears all GPS memory including clock drift (See note). **1** - **Coldstart** (No Almanac, No Ephemeris): this option clears all data that is currently stored in the internal memory of the GPS receiver including position, almanac, ephemeris, and time. The stored clock drift however, is retained. **2** - **Warmstart** (No ephemeris): this option clears all initialization data in the GPS receiver and subsequently reloads the data that is currently displayed in the Receiver Initialization Setup screen. The almanac is retained but the ephemeris is cleared.

$GPSR – GPS module Reset	
	3 - **Hotstart** (with stored Almanac and Ephemeris): the GPS receiver restarts by using the values stored in the internal memory of the GPS receiver; validated ephemeris and almanac.
AT$GPSR?	Read command displays the current <**reset_type**> value (see note).
AT$GPSR=?	Test command reports the range of supported values for parameter <reset_type>
Example	AT$GPSR=0 OK
Note	- '**Factory Reset**' performs the same operation as '**Coldstart**'. - The current setting is stored through $GPSSAV command. - <**reset_type**> sets the kind of start when GPS is activated through $GPSP or $GPSSLSR commands. - Default value of <**reset_type**> is '3', 'hot start'.

3.4.7.8.3 *Get Acquired Position - $GPSACP*

$GPSACP - Get Acquired Position	
AT$GPSACP	Execution command returns information about the last GPS position in the format: **$GPSACP:** <UTC>,<latitude>,<longitude>,<hdop>,<altitude>, <fix>,<cog>,<spkm>,<spkn>,<date>,<nsat> **where:** **<UTC>** - UTC time (hhmmss.sss) referred to GGA sentence **<latitude>** - format is ddmm.mmmm N/S (referred to GGA sentence) **where:** dd – degrees - 00..90 mm.mmmm - minutes - 00.0000..59.9999 N/S: North / South **<longitude>** - format is dddmm.mmmm E/W (referred to GGA sentence) **where:** ddd - degrees - 000..180 mm.mmmm - minutes - 00.0000..59.9999 E/W: East / West **<hdop>** - x.x - Horizontal Diluition of Precision (referred to GGA sentence) **<altitude>** - xxxx.x Altitude - mean-sea-level (geoid) in meters (referred to GGA sentence) **<fix> -** 0 or 1 -Invalid Fix

$GPSACP - Get Acquired Position	
	2 - 2D fix
	3 - 3D fix
	<cog> - ddd.mm - Course over Ground (degrees, True) (referred to VTG sentence)
	where:
	ddd - degrees - 000..360
	mm – minutes - 00..59
	<spkm> - xxxx.x Speed over ground (Km/hr) (referred to VTG sentence)
	<spkn> - xxxx.x- Speed over ground (knots) (referred to VTG sentence)
	<date> - ddmmyy Date of Fix (referred to RMC sentence)
	where:
	dd - day - 01..31
	mm – month - 01..12
	yy – year - 00..99 - 2000 to 2099
	<nsat> - nn - Total number of satellites in use (referred to GGA sentence) - 00..12
AT$GPSACP?	Read command has the same meaning as the Execution command.
AT$GPSACP=?	Test command returns the OK result code.
Example	at$gpsp? $gpsp: 0 <when module is down there no aquired position> at$gpsacp $GPSACP: 000000.000,,,,,0,,,,000000,00 OK at$gpsp=1 OK <Until first fix is received the command will display initial GPS position> at$gpsacp $GPSACP: 3124.6000N,03504.2000E,0.0,-18.0,0,0.0,0.0,0.0,060180,00 <Once fix has been received the command will display actual GPS position> OK at$gpsacp $GPSACP: 3206.4020N,03450.2678E,1.1,3.3,0,0.0,0.0,0.0,030613,06 OK

$GPSACP - Get Acquired Position	
Reference	NMEA 01803 Specifications.

3.4.7.8.4 *Unsolicited NMEA Data Configuration - $GPSNMUN*

$GPSNMUN - Unsolicited NMEA Data Configuration	
AT$GPSNMUN= **<enable>** **[,<GGA>,<GLL>,** **<GSA>,<GSV>,** **<RMC>,<VTG >]**	Set command permits to activate an Unsolicited streaming of GPS data (in NMEA format) through the standard GSM serial port and defines which NMEA sentences will be available **Parameters**: **<enable>** 0 - NMEA data stream de-activated (default). 1 - NMEA data stream activated with the following unsolicited response syntax: **$GPSNMUN: <NMEA SENTENCE><CR>** 2 - NMEA data stream activated with the following unsolicited response syntax: **<NMEA SENTENCE><CR>** 3 - Dedicated NMEA data stream. it is not possible to send AT commands; With the escape sequence "+++" the user can return to command mode. NMEA syntax is: **<NMEA SENTENCE><CR>** **<GGA>** - Global Positioning System Fix Data 0 - disable (default) 1 - enable **<GLL>** - Geographical Position - Latitude/Longitude 0 - disable (default) 1 - enable **<GSA>** - GPS DOP and Active Satellites 0 - disable (default) 1 - enable **<GSV>** - GPS Satellites in View 0 - disable (default) 1 - enable **<RMC>** - recommended Minimum Specific GPS Data 0 - disable (default) 1 - enable **<VTG>** - Course Over Ground and Ground Speed 0 - disable (default) 1 - enable
AT$GPSNMUN?	Read command returns whether the unsolicited GPS NMEA data streaming is currently enabled or not, along with the NMEA sentences availability status, in the format: **$GPSNMUN:<enable>,<GGA>,<GLL>,<GSA>,<GSV>,<RMC>,<VTG>**

AT$GPSNMUN=?	Test command returns the supported range of values for parameters **<enable>, <GGA>, <GLL>, <GSA>, <GSV>, <RMC>, <VTG>**
Example	AT$GPSNMUN=1,0,0,1,0,0,0 OK These sets the GSA as available sentence in the unsolicited message AT$GPSNMUN=0 OK Turn-off the unsolicited mode AT$GPSNMUN? $GPSNMUN: 1,0,0,1,0,0,0 OK Give the current frame selected (GSA) The unsolicited message will be: $GPSNMUN: $GPGSA,A,3,23,20,24,07,13,04,02,,,,,,2.4,1.6,1.8*3C
Reference	NMEA 01803 Specifications

3.4.7.8.5 *Save GPS Parameters Configuration - $GPSSAV*

$GPSSAV - Save GPS Parameters Configuration	
AT$GPSSAV	Execution command stores the current GPS parameters in the NVM of the device.
AT$GPSSAV=?	Test command returns the 'OK' result code
Example	AT$GPSSAV OK
Note	The saved parameters are those of : $GPSP, $GPSR, $GPSNMUN, $GPSNMUNEX, $GPSQOS, $GPSSLSR, $GPSSTOP and $GPSAT commands.

3.4.7.8.6 *Restore to Default GPS Parameters - $GPSRST*

$GPSRST - Restore To Default GPS Parameters	
AT$GPSRST	Execution command resets the GPS parameters to "Factory Default" configuration and stores them in the NVM of the device.
AT$GPSRST=?	Test command returns the 'OK' result code.
Example	AT$GPSRST OK
Note	The restored parameters are those of : $GPSP, $GPSR, $GPSNMUN, $GPSNMUNEX, $GPSQOS, $GPSSLSR, $GPSSTOP and $GPSAT commands (see their default value at each command description)

	The module must be restarted to use the new configuration

3.4.7.8.7 *GPS Antenna Supply Voltage Readout - $GPSAV*

$GPSAV - GPS Antenna Supply Voltage Readout	
AT$GPSAV	Execution command returns the measured GPS antenna's supply voltage in Mv.
AT$GPSAV?	Read command has the same meaning as the Execution command
AT$GPSAV=?	Test command returns the OK result code
Note	In LE9x0, gps antenna is passive, so this command has no real meaning. It exists for backward compatibility.

3.4.7.8.8 *GPS Antenna Type Definition - $GPSAT*

$GPSAT - GPS Antenna Type Definition	
AT$GPSAT=\<type>	Set command selects the GPS antenna used. **Parameter \<type> :** 0 - GPS Antenna not power supplied by the module 1 - GPS Antenna power supplied by the module (default)
AT$GPSAT?	Read command returns the currently used antenna, in the format: **$gpsat: \<type>**
AT$GPSAT=?	Test command returns the valid range values of \<type> parameter. **at$gpsat=?** **$gpsat: (0,1)**
Note	The current setting is stored through $GPSSAV. This command has no real meaning. It exists for backward compatibility.

3.4.7.8.9 *Set the GNSS (or GLONASS) Capability - $GPSGLO*

$GPSGLO – Set the GNSS (or GLONASS) Capability	
AT$GPSGLO=\<type>	Set command selects the GNSS (or GLONASS)capability used. **Parameter:** **\<type>** 0 – Disable GNSS(or GLONASS) 1 – Enable GNSS(or GLONASS) (default)
AT$GPSGLO?	Read command returns the currently used GNSS(or GLONASS), in the format: $GPSGLO: **\<type>**

$GPSGLO – Set the GNSS (or GLONASS) Capability	
AT$GPSGLO=?	Test command reports the range of supported values for parameter **<type>**
Note	This command saved in NVM and has effect only at the next power cycle.
Example	AT$GPSGLO=1 OK

3.4.7.8.10   *Unsolicited NMEA Extended Data Configuration - $GPSNMUNEX*

$GPSNMUNEX - Unsolicited NMEA Extended Data Configuration	
AT$GPSNMUNEX= <GNGNS>,<GNGSA>, <GLGSV>,<GPGRS>	Set command permits to activate an Unsolicited streaming of GNSS(or GLONASS) data (in NMEA extended format) through the NMEA port and defines which NMEA extended sentences will be available **Parameters**: <**GNGNS**> - Fix data of GNSS (or GLONASS) receivers. 0 - disable (default) 1 – enable <**GNGSA**> - DOP and active satellites of GNSS(or GLONASS) 0 - disable (default) 1 – enable <**GLGSV**> - GLONASS satellites in view 0 - disable (default) 1 – enable <**GPGRS**> - GPS Range Residuals 0 - disable (default) 1 – enable
AT$GPSNMUNEX?	Read command returns the NMEA extended sentences availability status, in the format: $GPSNMUNEX: <**GNGNS**>,<**GNGSA**>, <**GLGSV**>, <**GPGRS**>
AT$GPSNMUNEX=?	Test command returns the supported range of values for parameters: **<GNGNS>,<GNGSA>, <GLGSV>, <GPGRS>**
Note	NMEA extended data is displayed on NMEA port depending on $**GPSNMUN <enable>** parameter setting.
Example	AT$GPSNMUN=1 OK AT$GPSNMUNEX=1,0,0,0 OK These sets the GNGNS as available sentence in the unsolicited nmea sentences. AT$GPSNMUNEX?

$GPSNMUNEX - Unsolicited NMEA Extended Data Configuration	
	$GPSNMUNEX: 1,0,0,0 OK Give the current frame selected (GNGNS) The unsolicited message will be: $GNGNS,080558.0,3731.306144,N,12655.784429,E,AN,09,1.0,68.0,18.0,,*5B

3.4.7.8.11 *GPS Quality of Service - $GPSQOS*

$GPSQOS – GPS Quality Of Service	
AT$GPSQOS=[<horiz_accuracy>[,<vertic_accuracy>[,<rsp_time>[,<age_of_location_info>[,<location_type>[,<nav_profile>[,<velocity_request>]]]]]]]	Command used to set the location's quality of service (QoS). **Parameter**: **<horiz_accuracy>** (horizontal accuracy): 0 – 1800000, where 0 is highest accuracy and 1800000 is lowest accuracy in meters. Default value is 1800000 in meters **<vertic_accuracy>** (vertical accuracy): 0 – 990, where 0 is highest accuracy and 990 is lowest accuracy in meters. Default is 990 in meters **<rsp_time>** (response time): 0-14400, where 0 is the low delay and 14400 is the highest delay in seconds. Default value is 14400 in seconds. **<age_of_location_info>** (Maximum age of location): 0-1966020: Value 0 means that stored location information should not be used. Value 1966020 indicates the maximum tolerable age of the stored location information. The valid range of interval for SUPL (Transport protocol) is [0 - 65535] seconds & [0 - 1966020] seconds for C-plane (Transport protocol). **<location_type>** (type of location required): Used only in case of C-Plane. 0: Current Location (default) 1: Current or Last known location 2: Invalid Location, indicates that this parameter shall not be used **<nav_profile>** (navigation profile): 0: Car navigation profile (default) 1: Personal profile 2: Low speed profile 3: Invalid profile, indicates that this parameter shall not be used **< velocity_request>** (velocity information is needed): 0 FALSE 1 TRUE (default)
AT$GPSQOS?	Read command returns the current QoS values, in the format: AT$GPSQOS: **<horiz_accuracy>,<vertic_accuracy>,<rsp_time>** **,<age_of_location_i**

	nfo>,<location_type>,< nav_profile>,< velocity_request>
AT$GPSQOS=?	Returns the list of supported QoS values for each field. $GPSQOS: (0-1800000),(0-990),(0-14400),(0-1966020),(0-2),(0-3),(0,1)
Example	AT$GPSQOS=1800000,990,150,0,0,0 OK
Note	The current setting is stored through $GPSSAV

3.4.7.8.12 *GPS Start Location Service Request-$GPSSLSR*

$GPSSLSR – GPS Start Location Service Request	
$GPSSLSR=<transport_protocol>[,<pos_mode>[,<client_id>,<clientid_type>[,<mlc_number>,<mlcnumber_type>[,<interval>[,<service_type_id>[,<pseudonym_indicator>],<error_mask>]]]]]]	Command used to start the Receiver in Autonomous or A-GPS mode. **Parameter**s: **<transport_protocol>:** 0 - CPlane 1 - SUPL 2 – Invalid **Note**: If <**pos_mode** > is Autonomous the <**transport_protocol**> should be invalid. **Note**: If **<transport_protocol>** is CPlane and <**pos_mode** > is Pure MS Assisted, then **<interval>** should be 0 (or omitted). **<pos_mode>:** 0: Pure MS Assisted - Location estimate from the network (MS Assisted mode). 1: MS Based - Assistance Data from the network (MS Based mode). 2: Not Supported. 3: Autonomous – Autonomous GPS mode of operation. **Note**: If **<pos_mode>** is Autonomous the **<transport_protocol>** should be invalid. **<client_id>:** String parameter containing the ID of the LCS-Client to which the location estimate is to be transferred. **Note**: Max length is 64 bytes. **Note**: **<client_id>** is mandatory in case of A-GPS and the **<transport_protocol>** should be Cplane. **Note**: LE9x0 don't support <client_id> setting because of not supporting the 3rd Part Location Transfer. **<clientid_type>:** 0 – MSISDN 1 – Invalid (default)

	Note: **<client_id>** and **<clientid_type>** are mandatory for A-GPS mode. **<mlc_number>:** String parameter containing the address of the GMLC through which the location estimate is to be sent to the LCS-Client. **Note**: **<mlc_number>** is mandatory in case of A-GPS. **<mlcnumber_type>:** 0 – MSISDN 1 – Invalid (default) **Note**: **<mlc_number>** and **<mlcnumber_type>** are mandatory for A-GPS mode. **<interval>:** 0 - 7200: GPS reporting period in seconds (will be sent unsolicited). if the value is 0 then a single shot NMEA Message will be provided. Any value different from 0 sets the period (in seconds) between each NMEA Sentence. **Note**: If this value is not set, it is assumed to be 0. **Note**: The Unsolicited NMEA sentences have to be enabled with the commands AT$GPSNMUN. **<service_type_id>:** 0 - 255 where 255 indicates that this parameter shall not be used. **Note**: <service_type_id> is mandatory in case of A-GPS. **<pseudonym_indicator>:** 0 FALSE (default) : display user name at the external client. 1 TRUE : display user name as anonymous at the external client **<error_mask>:** 0 - 4294967295: If certain bit is set, respective error code becomes non-abortable. **Note**: If this value is not set, it is assumed to be 0. If C-plane or Supl session is not successfully completed, it will be stopped and unsolicited indication reports the error cause in the following formats: **$GPSSLSR: SUPL/C-PLANE ERROR,<error_code>** where **<error_code>** 0 - Phone Offline 1 - No servcie 2 - No connection with PDE 3 - No data available 4 - Session Manager Busy 5 - Phone is CDMA locked 6 - Phone is GPS locked 7 - Connection failure with PDE

	8 - PDSM Ended session because of Error condition
	9 - User ended the session
	10 - End key pressed from UI
	11 - Network Session was ended
	12 - Timeout (viz., for GPS Search)
	13 - Conflicting request for session and level of privacy
	14 - Could not connect to the Network
	15 - Error in Fix
	16 - Reject from PDE
	17 - Ending session due to TC exit
	18 - Ending session due to E911 call
	19 - Added protocol specific error type
	20 - Ending because BS info is stale
	21 - VX lcs agent auth fail
	22 - Unknown System Error
	23 - Unsupported Service
	24 - Subscription Violation
	25 - The desired fix method failed
	26 - Antenna switch
	27 - No fix reported due to no tx confirmation rcvd
	28 - Network indicated a Normal ending of the session
	29 - No error specified by the network
	30 - No resources left on the network
	31 - Position server not available
	32 - Network reported an unsupported version of protocol
	33 - MOLR System failure
	34 - MOLR Unexpected data value
	35 - MOLR Data missing
	36 - MOLR Facility Not Supported
	37 - MOLR Subscription Violation
	38 - MOLR Position Method Failure
	39 - MOLR Undefined
	Note: Errors [1-32] can be marked as non – abortable, using **<error_mask>,** so session will continue until stopped manually by user. **EXCEPTIONS:** Errors [9-12] are non – abortable by default, and error mask does not affect them.
AT$GPSSLSR?	Read command returns the current settings, in the format: **$GPSSLSR: <transport_protocol>[,<pos_mode>[,<client_id>,<clientid_type>[,<mlc_number>,<mlcnumber_type>[,<interval>[,<service_type_id>[,<pseudonym_indicator>][,<error_mask>]]]]]]**
AT$GPSSLSR=?	**$GPSSLSR: (0-2),(0-3),(64),(0,1),(64),(0,1),(0-7200),(0-255),(0,1),(0-4294967295)**

Example	AT$GPSSLSR= 2,3,,,,,1 OK
Note	- The current setting is stored through $GPSSAV - Power up clears GPS memory and then starts the GPS receiver. GPS data cleaning is performed on the base of the current value of the <**reset_type**> parameter (see $GPSR).

3.4.7.8.13 *GPS Stop Location Service Request - $GPSSTOP*

$GPSSTOP – GPS Stop Location Service Request	
$GPSSTOP= [<abort_cause>]	Command used to stop the Receiver in Autonomous or A-GPS mode initiated through $GPSSLSR set command. **Parameter**: <**abort_cause**> 0: User denies the request 1: Unspecified cause for abort 2: Cause Invalid
AT$GPSSTOP?	Read command returns the current value of parameter <**abort_cause**>.
$GPSSTOP=?	OK
Example	AT$GPSSTOP=1 OK
Note	The current setting is stored through $GPSSAV. The default factory value is '1', it can be recovered by $GPSRST.

3.4.7.8.14 *Update SLP address - $SLP*

$SLP - Update SLP address	
AT$SLP=<slp_address_type> [,<slp_address:slp_port>]	Set command allows updating the SLP address and SLP port number. **Parameters**: <**slp_address_type**>: SLP address type 0 - IPv4 1 - FQDN 3 – IPv6 <**slp_address**>: SLP address in FQDN format, IPv4 or IPv6 format <**slp_port_number**> : Slp Port number integer parameter. Default value is 7275.

	Note: If <slp_address> is omitted, chosen <slp_address_type> will be deleted. Note: The current setting is stored in NVM. Note: IPv6 is passed in the following format (example): AT$SLP=3,"[2001:db8:255::8:7]:7275"
AT$SLP?	Read command returns the current SLP address.
AT$SLP=?	Test command returns the range of values for parameter $SLP: (0-1,3),("IP,URL,IPv6")

3.4.7.8.15 Update SLP address - $LCSSLP

$LCSSLP - Update SLP address	
AT$LCSSLP=<slp_address_type>[,<slp_address>[,<slp_port_number>]]	Set command allows updating the SLP address and SLP port number. Parameters: <slp_address_type>: SLP address type 0 - IPv4 1 - FQDN 2 – Delete SLP address 3 – IPv6 <slp_address>: SLP address in FQDN format or IPv4 format <slp_port_number> : Slp Port number integer parameter. Default value is 7275. Note: If <slp_address_type> is 0,1 or 3, then <slp_address> is a mandatory parameter. Note: Other types of address are erased during set command. Note: The current setting is stored in NVM.
AT$LCSSLP?	Read command returns the current SLP address.
AT$LCSSLP=?	Test command returns the range of values for parameter <slp_address_type> (0-3)

3.4.7.8.16 Update SLP address type - $SLPTYPE

$SLPTYPE - Update SLP address	
AT$SLPTYPE=<slp_address_type>	Set command allows updating the SLP address type to be chosen. <slp_address_type>: SLP address type 0 - IPv4 1 - FQDN 3 – IPv6

	<slp_address>: SLP address in FQDN format, IPv4 or IPv6 format **Note**: The current setting is stored in NVM. **Note:** This parameter also update during AT$SLP set command
AT$SLPTYPE?	Read command returns the current SLP address type.
AT$SLPTYPE=?	Test command returns the range of values for parameter **$SLPTYPE: (0-1, 3)**

3.4.7.8.17   *Set the User Plane Secure Transport - $SUPLSEC*

$SUPLSEC – Set the User Plane Secure Transport	

AT$SUPLSEC=
<option>
	Set command configures the User Plane Secure Transport
	
	Parameter:
	<option >: Integer type
	  0 - Disable User Plane Secure Transport(UPL)
	  1 - Enable User Plane Secure Transport(SUPL)

AT$SUPLSEC?	Read command returns the currently used values, in the format: $SUPLSEC: **< option >**
AT$SUPLSEC=?	Test command returns the supported range of values of parameters **< option >** **$SUPLSEC: (0-1)**
Example	AT$SUPLSEC =1 OK AT$SUPLSEC? $SUPLSEC:1 OK

3.4.7.8.18   *Configure SUPL TLS and Hash - $SUPLCFG*

$SUPLCFG – Configure SUPL TLS and Hash	

AT$SUPLCFG
=[<tls>],[<hash
>]
	This command permits to configure the SUPL TLS and Hash algorithm version.
	Parameters:
	
	<tls >:
	  0 – Use TLS v.1.0 (default)
	  1 – Use TLS v.1.1
	
	<hash >:
	  0 – Use SHA-1 (default)
	  1 – Use SHA-256

$SUPLCFG – Configure SUPL TLS and Hash	
AT$SUPLCFG ?	Read command reports the currently selected <tls> and <hash> in the format: **$SUPLCFG: <tls>,<hash>**
AT$SUPLCFG =?	Test command reports the supported range of values for parameters **<tls>** and **<hash>**

3.4.7.8.19 *Set the version of supported SUPL - $SUPLV*

$SUPLV – Set the version of supported SUPL.	

AT$SUPLV= <version>	Set command configures the version of supported SUPL.
	
	 Parameter:
	 <version>: Integer type
	 0 : N/S SUPL
	  1 : SUPL 1.0
	  2 : SUPL 2.0

AT$SUPLV?	Read command returns the currently used values, in the format: $SUPLV: **< version >**
AT$SUPLV=?	Test command returns the supported range of values of parameters **< version>** $SUPLV: (0-2)
Example	AT$SUPLV =1 OK AT$SUPLV? $SUPLV:1.0 OK

3.4.7.8.20 *Update location information - $LCSLUI*

$LCSLUI - Update location information	
AT$LCSLUI=<update_type>	Set command allows updating the Location information. **Parameters**: **<update_type>** : the current access technology 0 - GSM 1 - WCDMA **Note**: the current access technology can be read with AT+COPS?
AT$LCSLUI=?	Test command returns the range of values for parameter **<update_type>**.

3.4.7.8.21 *Update terminal information - $LCSTER*

$LCSTER - Update terminal information

AT$LCSTER=<id_type>[,<id_value>[,<pref_pos_mode>[,<tls_mode>]]]	Set command updates the terminal information like IMSI, MSISDN or IPv4 address. **Parameters**: **<id_type>** : is a number which can have any of the following values 0 - MSIDSN 1 - IMSI (default value) 2 - IPv4 address 3 - Invalid **<id_value>** : is a string , as defined in **<id_type>** **<pref_pos_mode>** : preferred position mode, 0 – default position mode 1 – none preferred position mode **<tls_mode>** : indicates if TLS mode should/should not be used by the SET 0 - non-TLS mode 1 - TLS mode (default value) **Note**: If **<id_type>** is MSISDN or IPv4 address then **<id_value>** shall be entered

3.4.7.8.22   *Enable/Disable unsolicited response - $LICLS*

$LICLS – Enable/Disable unsolicited response	
AT$LICLS =<mode>	Set command is used to enable/disable unsolicited $LICLS response. **Parameter**: **<mode>** 0 – disable unsolicited 1 – enable unsolicited (default value) The unsolicited result code is in the format: $LICLS: **<request_type>[,<cid>]** Where **<request_type>** 0 – Setup Request to setup the control link 1 – Release Request to release the control link **<cid>** : id associated to the context that shall be deactivated (see +CGDCONT) If the <**request_type**> is a setup request, the unsolicited indication is sent/used to request the client to define, setup, activate and prepare the pdp-context. If <**request_type**> is a release request, the unsolicited indication is sent/used to inform the client that the pdp-context (associated with this command type) including the associated terminal is not used any more, and shall be deactivated. **Note**: The current setting is stored in NVM.
AT$LICLS?	Read command returns the current value of parameter <**mode**>.

AT$LICLS=?	Test command returns the range of values for parameter <**mode**>.

3.4.7.8.23   *LCS certificate - $LTC*

$LTC – LCS certificate	
AT$LTC=<string>,<total_message_length>,<seq_no>,<Security_Object_Type>	Set command is used to pass the security objects (e.g. certificate, key) to the Transport Layer Security Protocol (binary string). The certificate shall be in hexadecimal format (each octet of the certificate is given as two IRA character long hexadecimal number). **Parameter**: **<string>** - the string certificate segment (max 300 characters per segment) **<total_message_length>** - The total size of the certificate to be received 1-4096 **<seq_no>** - The sequence number of the segment. 1-13 **<Security_Object_Type>** 0: Root Certificate **Note**: The last two certificates are stored in NVM.
AT$LTC	Execution command deletes the certificates stored in NVM.
AT$LTC?	Read command provides the first 300 characters of each valid certificate stored in NVM in the format: **$LTC: <string>,<total_message_length>,1, <Security_Object_Type>** If no certificate is stored the read command provides: **$LTC: "",0,1 ,<Security_Object_Type>**
AT$LTC=?	Test command returns the range of values for parameters **<total_message_length>**,**<seq_no>** and **<Security_Object_Type>**

3.4.7.8.24   *Lock context for LCS use - $LCSLK*

$LCSLK – Lock context for LCS use	
AT$LCSLK=<mode>[,<cid>]	Set command is used to reserve a cid for LCS. **Parameters**: **<mode>** 0 – unlock the current cid available for LCS use **Note**: No CID should be specified. Locked one will be released automatically. 1 – lock the specified cid in order to setup a control link for LCS use only

	<cid> - a numeric parameter which specifies a particular PDP context definition (see +**CGDCONT** command) **Note**: **<cid>** is mandatory if **<mode>** is set to lock, otherwise shall be omitted. **Note**: the set command returns ERROR if the current cid and/or the previously set are in use. **Note**: The current setting is stored in NVM.
AT$LCSLK?	Read command returns the current value of parameters **<mode>** and **<cid>** (if **<mode>** is lock).
AT$LCSLK=?	Test command returns the range of values for parameters **<mode>** and **<cid>**

3.4.7.8.25 *Clear GPS Data - $GPSCLRX*

$GPSCLRX – Clear GPS Data	
AT$GPSCLRX	This command resets all of the parameters related with GPS Resets all parameters related with GPS as following: -GPS Almanac Data -GPS Ephemeris Data -LBS User Plane PDE IP Address -LBS User Plane PDE IP Port **Note**: This command is global and cannot clear individual pieces of data.
AT$GPSCLRX =?	Test command returns the OK result code
Example	AT$GPSCLRX=? OK AT$GPSCLRX OK

3.5 AT parser abort

The following listed AT Commands can be aborted during execution:
ATD
ATA
+CLCK
+CLCC
+COPN
+CPOL
+CLIP
+CLIR

Note: If DTE transmit any character before receiving the response to the issued AT Command, this causes current AT Command to be aborted.

4 List of acronyms

ARFCN	Absolute Radio Frequency Channel Number
AT	Attention command
BA	BCCH Allocation
BCCH	Broadcast Control Channel
CA	Cell Allocation
CBM	Cell Broadcast Message
CBS	Cell Broadcast Service
CCM	Current Call Meter
CLIR	Calling Line Identification Restriction
CTS	Clear To Send
CUG	Closed User Group
DCD	Data Carrier Detect
DCE	Data Communication Equipment
DCS	Digital Cellular System
DNS	Domain Name System Server
DSR	Data Set Ready
DTE	Data Terminal Equipment
DTMF	Dual Tone Multi Fraquency
DTR	Data Terminal Ready
GPRS	Global Packet Radio Service
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
IRA	International Reference Alphabet
IWF	Interworking Function
MO	Mobile Originated
MT	Mobile Terminal
NVM	Non Volatile Memory
PCS	Personal Communication Service
PDP	Packet Data Protocol
PDU	Packet Data Unit
PIN	Personal Identification Number
PPP	Point to Point Protocol
PUK	Pin Unblocking Code
RLP	Radio Link Protocol
RMC	Recommended minimum Specific data
RTS	Request To Send
SAP	SIM Access Profile
SCA	Service Center Address
SMS	Short Message Service
SMSC	Short Message Service Center
SMTP	Simple Mail Transport Protocol

TA	Terminal Adapter
TCP	Transmission Control Protocol
TE	Terminal Equipment
UDP	User Datagram Protocol
USSD	Unstructured Supplementary Service Data
UTC	Coordinated Universal Time
VDOP	Vertical dilution of precision
VTG	Course over ground and ground speed