AT Commands Reference Guide

For CE910 Series

80399ST10110A Rev.9 – 2016-02-18

APPLICABILITY TABLE

PRODUCT
CE910-DUAL
CE910-SC

SW Version
18.22.003 (Sprint, 2.00) 18.12.013 (Aeris.net) 18.02.023 (Verizon, 1.00) 18.12.023 (Verizon, 1.10) 18.02.031 (US Cellular) 18.02.393 (-SC)

SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE

Notice

While reasonable efforts have been made to assure the accuracy of this document, Telit assumes no liability resulting from any inaccuracies or omissions in this document, or from use of the information obtained herein. The information in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies or omissions. Telit reserves the right to make changes to any products described herein and reserves the right to revise this document and to make changes from time to time in content hereof with no obligation to notify any person of revisions or changes. Telit does not assume any liability arising out of the application or use of any product, software, or circuit described herein; neither does it convey license under its patent rights or the rights of others.

It is possible that this publication may contain references to, or information about Telit products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that Telit intends to announce such Telit products, programming, or services in your country.

Copyrights

This instruction manual and the Telit products described in this instruction manual may be, include or describe copyrighted Telit material, such as computer programs stored in semiconductor memories or other media. Laws in the Italy and other countries preserve for Telit and its licensors certain exclusive rights for copyrighted material, including the exclusive right to copy, reproduce in any form, distribute and make derivative works of the copyrighted material. Accordingly, any copyrighted material of Telit and its licensors contained herein or in the Telit products described in this instruction manual may not be copied, reproduced, distributed, merged or modified in any manner without the express written permission of Telit. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit, as arises by operation of law in the sale of a product.

Computer Software Copyrights

The Telit and 3rd Party supplied Software (SW) products described in this instruction manual may include copyrighted Telit and other 3rd Party supplied computer programs stored in semiconductor memories or other media. Laws in the Italy and other countries preserve for Telit and other 3rd Party supplied SW certain exclusive rights for copyrighted computer programs, including the exclusive right to copy or reproduce in any form the copyrighted computer program. Accordingly, any copyrighted Telit or other 3rd Party supplied SW computer programs contained in the Telit products described in this instruction manual may not be copied (reverse engineered) or reproduced in any manner without the express written permission of Telit or the 3rd Party SW supplier. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit or other 3rd Party supplied SW, except for the normal non-exclusive, royalty free license to use that arises by operation of law in the sale of a product.

Usage and Disclosure Restrictions

License Agreements

The software described in this document is the property of Telit and its licensors. It is furnished by express license agreement only and may be used only in accordance with the terms of such an agreement.

Copyrighted Materials

Software and documentation are copyrighted materials. Making unauthorized copies is prohibited by law. No part of the software or documentation may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, without prior written permission of Telit

High Risk Materials

Components, units, or third-party products used in the product described herein are NOT fault-tolerant and are NOT designed, manufactured, or intended for use as on-line control equipment in the following hazardous environments requiring fail-safe controls: the operation of Nuclear Facilities, Aircraft Navigation or Aircraft Communication Systems, Air Traffic Control, Life Support, or Weapons Systems (High Risk Activities"). Telit and its supplier(s) specifically disclaim any expressed or implied warranty of fitness for such High Risk Activities.

Trademarks

TELIT and the Stylized T Logo are registered in Trademark Office. All other product or service names are the property of their respective owners.

Contents

1.   Introduction

1.1.   Scope

This document provides a detailed specification and a comprehensive listing of all AT commands supported by the CE910-Series.

1.2.   Audience

Readers of this document should be familiar with Telit modules and the means of controlling them using 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 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 the scope for this document, target audience, contact and support information, and text conventions.

Chapter 2: "Overview" discusses the goal of this document and implementation suggestions.

Chapter 3: "AT Commands" The core of this reference guides.

1.5.　Text Conventions

STOP	**_Danger – This information MUST be followed or catastrophic equipment failure or bodily injury may occur._**
⚠	**_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

2.   Overview

2.1.   About the document

This document describes all AT commands implemented on the Telit wireless module CE910-Series.

3.   AT COMMANDS

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

1. TIA/EIA/707-A.3 AT Command.
2. Partial Hayes standard AT command set.
3. Partially 3GPP 27.005 specific AT Commands for SMS (Short Message Service).
4. Partially ETSI 3GPP 27.007 specific AT Commands for controlling voice and Phonebook.

Moreover, the Telit wireless module family supports Telit proprietary AT commands for specific purposes.
In case of Telit Test AT Commands, the Telit wireless module family should be checked individually and then should be rebooted when completed the operation.
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 recognized 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 the 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. They do not appear in the command line.

[...] Optional sub parameter 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 the sub parameter 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 sub parameters, and so do not have a Read command, called *action type*

---

[1] 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.

commands, action should be taken on the basis of the recommended default setting of the sub parameter.

3.2. AT Command Syntax

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

- **Parameter type commands**. This type of command 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 sub parameters; they also have a Read command (trailing **?**) to check the current values of sub parameters.

- **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 corresponding Test command - trailing =**?** - returns the **OK** result code), and, if sub parameters are associated with the action, the ranges of sub parameters values that are supported.

Action commands do not store the values of any of their possible sub parameters.
In the case of a Telit command, the "read" action may be used for a specific purpose.
Moreover:
   The response to the Test Command (trailing **=?**) may be changed in the future by Telit to allow the description of new values/functionalities

   If all the sub parameters 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 sub parameters to be retained.

3.2.1. String Type Parameters

A string either enclosed between quotes or not, is considered a valid string type parameter input. According to V25 space characters are ignored on the command line and may be used freely for formatting purposes unless they are embedded in 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.

A small set of commands requires always 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 sub parameter
- **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 delimited with semicolon. In the second command, the sub parameter is omitted.

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

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

These commands might be performed in a single command line as shown below:

ATCMD1 CMD2=10+CMD1;+CMD2=, ,10;+CMD1?;+CMD1=?<CR>

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, an error message is received but it is not possible to determine 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 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 **<CR><LF>ERROR<CR><LF>** is sent and no subsequent commands in the command line are processed.

If command **V0** is enabled (numeric response 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** differs from the standard commands 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.**

3.2.2.1. 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 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
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
General purpose error:	
100	unknown
560	cannot open socket
561	remote disconnected or time-out
562	connection failed
563	tx error
564	already listening
601	wrong state
602	Can not activate
606	Bad or no response from server
608	Already connected
613	Data socket yet opened in cmdmode
614	FTP CmdMode data socket closed
615	Service is not available
616	Invalid user id
617	Invalid password
618	File is not found
683	Active call state
684	RR connection Established

*(Values in parentheses are 3GPP TS 24.008 cause codes)

3GPP TS 27.007 CDMA Network Problems

Numeric Format	Meaning
148	Unspecified CDMA error
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
603	Cannot resolve name
605	Cannot connect control socket
607	Not connected

3.2.2.2. Message Service Failure Result Code - +CMS ERROR: <err>

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

Numeric Format	Meaning
0-1	ME failure
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 (RUIM only)
311	SIM PIN required (RUIM only)
312	PH-SIM PIN required (RUIM only)
313	SIM failure (RUIM only)
314	SIM busy (RUIM only)
315	SIM wrong (RUIM only)
316	SIM PUK required (RUIM only)
317	SIM PIN2 required (RUIM only)
318	SIM PUK2 required (RUIM only)
320	Memory failure
321	Invalid memory index
322	Memory full
331	No network service
332	Network time-out
340	Invalid transaction ID
500	unknown error

The following values are IS-41D SMS cause codes:

Network Problems

Numeric Format	Meaning
0	Address vacant
1	Address translation failure
2	Network resource shortage
3	Network failure
4	Invalid teleservice id
5	Other network problem
6	Other network problem more first

Terminal Problems

Numeric Format	Meaning
32	No page response
33	Destination busy
34	No Acknowledgement
35	Network failure
36	SMS delivery postponed
37	Destination out of service
38	Destination no longer at this address
39	Other Terminal problem
40	Other terminal problem more first
47	Other terminal problem more last
48	SMS delivery postponed more first
63	SMS delivery postponed more last

Radio Interface Problems

Numeric Format	Meaning
64	Radio if resource shortage
65	Radio if incompatible
66	Other radio if problem
67	Other radio if problem more first
95	Other radio if problem more last

General Problems

Numeric Format	Meaning
96	Unexpected parameter size
97	SMS origination denied
98	SMS termination denied
99	Supplemental service not supported
100	SMS not supported
101	RESERVED 101
102	Missing expected parameter
103	Missing mandatory parameter
104	Unrecognized parameter value
105	Unexpected parameter value
106	User data size error
107	Other general problems
108	Other general problems more first

3.2.3. 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

3.2.4. Command Response Time-Out

Every command issued to the Telit modules returns 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 network, and involve only internal set up settings or readings typically have quicker response times than commands that require network interaction.

In the table below are listed only the commands whose interaction with 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)
+CPBR	5 (single reading) 15 (complete reading of a 500 records full phonebook)
+CPBF	10 (string present in a 500 records full phonebook) 5 (string not present)
+CPBW	5
+VTS	5 (transmission of full "1234567890*#ABCD" string with no delay between tones, default duration)
+CSMS	5
+CMGF	5
+CSMP	5
+CNMI	5
+CMGS	180 / 5 for prompt">"
+CMSS	180
+CMGW	5 / 5 for prompt">"
+CMGD	5
+CMGR	5
+CMGL	5
D	40
A	5 (voice call)
H	2

+CHUP	2
#TONE	5 (if no duration specified)
#EMAILD	60
#EMAILACT	150
#SEMAIL	210 (context activation + DNS resolution)
#SKTD	140 (DNS resolution + timeout set with AT#SKTCT)
#SKTOP	290 (context activation + DNS resolution + timeout set with AT#SKTCT)
#QDNS	170
#FTPOPEN	120 (timeout set with AT#FTPTO, in case no response is received from server)
#FTPCLOSE	500 (timeout set with AT#FTPTO, in case no response is received from server)
#FTPTYPE	500 (timeout set with AT#FTPTO, in case no response is received from server)
#FTPDELE	500 (timeout set with AT#FTPTO, in case no response is received from server)
#FTPPWD	500 (timeout set with AT#FTPTO, in case no response is received from server)
#FTPCWD	500 (timeout set with AT#FTPTO, in case no response is received from server)
#FTPLIST	500 (timeout set with AT#FTPTO, in case no response is received from server) + time to get listing
#FTPPUT	500 (timeout set with AT#FTPTO, in case no response is received from server)
#SGACT	150
#SH	10
#SD	140 (DNS resolution + connection timeout set with AT#SCFG)

3.2.5. Command Issuing Timing

The chain Command -> Response shall always be respected and a new command must not be issued before the module has finished sending all 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 regardless to wait for at least 20ms between the end of the reception of the response and issuing 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 shall be respected.

During command mode, due to hardware limitations, under severe CPU load the serial port can lose some characters if placed in autobauding at high speeds. If this problem is encountered the baud rate should be adjusted 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 this setting to remain 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 historically the one that was saved and restored in early releases of code, and the **extended section** that includes all the remaining values.

The **&W** command is used to save the actual values of **both sections** of profiles into the NVM user profile.

Commands **&Y** and **&P** are both used to set the profile to be loaded at start up. **&Y** instructs the device to load at start up only the **base section**. **&P** instructs the device to load at start up the full profile: **base + extended sections**.

The **&F** command resets to factory profile values in the base section of profile, while the **&F1** resets to factory profile values in 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, without issuing any **&W**, while others are stored by issuing specific commands (+**CSAS**, **#SLEDSAV**, **#VAUXSAV**, **#SKTSAV**, **#ESAV**); 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
EXTENDED MESSAGES:	X
FLOW CONTROL OPTIONS:	&K, +IFC
DSR (C107) OPTIONS:	&S
DTR (C108) OPTIONS:	&D
DCD (C109) OPTIONS :	&C
RI (C125) OPTIONS :	\R
POWER SAVING:	+CFUN
DEFAULT PROFILE:	&Y0
S REGISTERS:	S0;S1;S2;S3;S4;S5;S7;S12;S25;S30;S38
CHARACTER FORMAT:	+ICF

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

+FCLASS,	+ILRR,	+DR,
+CSCS,	+CRC,	+CVHU,
+CREG,	+CLIP,	+CLIR,
+CCWA	+CPBS,	+CMEE,
+CMGF,	+CSDH,	+CNMI,
#ACAL,	#PSMRI,	#ACALEXT,

#ECAM,	#SMOV,	#SKIPESC,
#E2ESC	#CFLO	
+CALM,	+CRSL,	+CMUT,
+CLVL,	+VTD,	#PCMTXG
#PCMRXG	#DVICFG,	#CAP,
#SRS,	#SRP,	#STM,
#DVI,	#SHFEC,	#QSS,
#STIA,		
#HSMICG,	#SHFSD,	#SPKMUT,
#E2SLRI,	#DAC,	#PSEL,
#HFRECG	#HSRECG,	#SHFAGC,
#SHSAGC,	#SHSEC,	#SHSNR,
#SHFNR,	#SHSSD,	#TSVOL
#E2SMSRI	#HFMICG	
#TEMPMON (lt is partially stored in NVM, see command description)		#NITZ
#NOPT		

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

#SELINT,	#DIALMODE,	#CODEC
#SCFG,	#ICMP	#SHSFTX,
#SHSFRX	#SHFFTX,	#SHFFRX,
#SRXAGC,	#SHSAGCTX,	#SHSAGCRX,
#SHFAGCTX	#SHFAGCRX	
#DNS	#TCPMAXDAT	#TCPREASS
#SMSMOEN	#SMSSO	
#USERID	#PASSW	#SMSATRUNCFG
#SMSATRUN	#TCPATRUNCFG	#TCPATRUND
#TCPATRUNL	#ENAEVMONI	#ENAEVMONICFG
#EVMONI	#SGACTCFG	#SGACTCFGEXT
#TESTMODE	#RSSICFG	#SSLEN
#SSLCFG	#SSLSECCFG	#SSLSECDATA
#RIND		

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

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

#SLED
Stored by #SLEDSAV command.

#VAUX
Stored by #VAUXSAV command.

#PKTSZ,	#DSTO,	#SKTTO,
#SKTSET	#SKTCT	

Stored by #SKTSAV command and automatically restored at start up; factory default values are restored by #SKTRST command.

#ESMTP,	#EADDR,	#EUSER,
#EPASSW		

stored by #ESAV command and automatically restored at start up; factory default values are restored by #ERST command.

---

[3] Both commands +CSAS and +CRES deal with non-volatile memory

3.4. AT Commands Availability Table

The following table lists the AT command set and matches the availability of every single command versus the Telit wireless module family.

COMMAND	Verizon (HW 1.00)	Verizon (HW 1.10)	Sprint (HW 2.00)	Aeris	US Cellular	-SC	Function
Command Line General Format – Command Line Prefixes							
AT	•	•	•	•	•	•	Starting A Command Line
A/	•	•	•	•	•	•	Last Command Automatic Repetition Prefix
#/	•	•	•	•	•	•	Repeat Last Command
General Configuration Commands							
#NOPT	•	•	•	•	•	•	Set Notification Port
#SELINT	•	•	•	•	•	•	Select Interface Style
#MSN	•	•	•	•	•	•	Manufacturer Serial Number
#HWREV	•	•	•	•	•	•	Hardware revision
#DIAGCFG	•	•	•	•	•	•	Diagnostic Port Configuration
Hayes AT Commands – Generic Modem Control							
&F	•	•	•	•	•	•	Set To Factory-Defined Configuration
Z	•	•	•	•	•	•	Soft Reset
+FCLASS	•	•	•	•	•	•	Select Active Service Class
&Y	•	•	•	•	•	•	Designate A Default Reset Basic Profile
&P	•	•	•	•	•	•	Designate A Default Reset Full Profile
&W	•	•	•	•	•	•	Store Current Configuration
&Z	•	•	•	•	•	•	Store Telephone Number In The Module Internal Phonebook
&N	•	•	•	•	•	•	Display Internal Phonebook Stored Numbers
+GMI	•	•	•	•	•	•	Manufacturer Identification
+GMM	•	•	•	•	•	•	Model Identification
+GMR	•	•	•	•	•	•	Revision Identification
+GCAP	•	•	•	•	•	•	Capabilities List
+GSN	•	•	•	•	•	•	Serial Number
&V	•	•	•	•	•	•	Display Current Base Configuration And Profile
&V0	•	•	•	•	•	•	Display Current Configuration And Profile
&V1	•	•	•	•	•	•	S Registers Display
&V3	•	•	•	•	•	•	Extended S Registers Display

&V2	•	•	•	•	•	•	Display Last Connection Statistics
\V	•	•	•	•	•	•	Single Line Connect Message
+GCI	•	•	•	•	•	•	Country Of Installation
%L	•	•	•	•	•	•	Line Signal Level
%Q	•	•	•	•	•	•	Line Quality
L	•	•	•	•	•	•	Speaker Loudness
M	•	•	•	•	•	•	Speaker Mode
Hayes AT Commands – DTE-Modem Interface Control							
E	•	•	•	•	•	•	Command Echo
Q	•	•	•	•	•	•	Quiet Result Codes
V	•	•	•	•	•	•	Response Format
X	•	•	•	•	•	•	Extended Result Codes
I	•	•	•	•	•	•	Identification Information
&C	•	•	•	•	•	•	Data Carrier Detect (DCD) Control
&D	•	•	•	•	•	•	Data Terminal Ready (DTR) Control
\Q	•	•	•	•	•	•	Standard Flow Control
&K	•	•	•	•	•	•	Flow Control
&S	•	•	•	•	•	•	Data Set Ready (DSR) Control
\R	•	•	•	•	•	•	Ring (RI) Control
+IPR	•	•	•	•	•	•	Fixed DTE Interface Rate
+IFC	•	•	•	•	•	•	DTE-Modem Local Flow Control
+ILRR	•	•	•	•	•	•	DTE-Modem Local Rate Reporting
+ICF	•	•	•	•	•	•	DTE-Modem Character Framing
Hayes AT Commands – Call Control							
D	•	•	•	•	•	•	Dial
T	•	•	•	•	•	•	Tone Dial
P	•	•	•	•	•	•	Pulse Dial
A	•	•	•	•	•	•	Answer
H	•	•	•	•	•	•	Disconnect
O	•	•	•	•	•	•	Return To On Line Mode
&G	•	•	•	•	•	•	Guard Tone
&Q	•	•	•	•	•	•	Sync/Async Mode
Hayes AT Commands – Modulation Control							
+MS	•	•	•	•	•	•	Modulation Selection

%E	•	•	•	•	•	•	Line Quality Monitor And Auto Retrain Or Fallback/Fall forward
Hayes AT Commands – Compression Control							
+DS	•	•	•	•	•	•	Data Compression
+DR	•	•	•	•	•	•	Data Compression Reporting
Hayes AT Commands – Break Control							
\B	•	•	•	•	•	•	Transmit Break To Remote
\K	•	•	•	•	•	•	Break Handling
\N	•	•	•	•	•	•	Operating Mode
Hayes AT Commands – S Parameters							
S0	•	•	•	•	•	•	Number Of Rings To Auto Answer
S1	•	•	•	•	•	•	Ring Counter
S2	•	•	•	•	•	•	Escape Character
S3	•	•	•	•	•	•	Command Line Termination Character
S4	•	•	•	•	•	•	Response Formatting Character
S5	•	•	•	•	•	•	Command Line Editing Character
S7	•	•	•	•	•	•	Connection Completion Time-Out
S10	•	•	•	•	•	•	Carrier Off With Firm Time
S12	•	•	•	•	•	•	Escape Prompt Delay
S25	•	•	•	•	•	•	Delay To DTR Off
S30	•	•	•	•	•	•	Disconnect Inactivity Timer
S38	•	•	•	•	•	•	Delay Before Forced Hang Up
Hayes AT Commands – Error Control							
+ES	•	•	•	•	•	•	Error Control Selection
ETSI GSM 07.07.27.007 – General							
+CGMI	•	•	•	•	•	•	Request Manufacturer Identification
+CGMM	•	•	•	•	•	•	Request Model Identification
+CGMR	•	•	•	•	•	•	Request Revision Identification
+CGSN	•	•	•	•	•	•	Request Product Serial Number Identification
+CSCS	•	•	•	•	•	•	Select TE Character Set
+CIMI	•	•	•	•	•	•	Request International Mobile Subscriber Identity (IMSI)
+CMUX	•	•	•	•	•	•	Multiplexing Mode
ETSI GSM 07.07/27.007 – Call Control							
+CHUP	•	•	•	•	•	•	Hang Up Call
+CEER	•	•	•	•	•	•	Extended Error Report

+CRC	•	•	•	•	•	•	Cellular Result Codes
+CVHU	•	•	•	•	•	•	Voice Hang Up Control
ETSI GSM 07.07/27.007 – Network Service Handling							
+CNUM	•	•	•	•	•	•	Subscriber Number
+COPN	•	•	•	•	•	•	Read Operator Names
+CREG	•	•	•	•	•	•	Network Registration Report
+CLIP	•	•	•	•	•	•	Calling Line Identification Presentation
+CLIR	•	•	•	•	•	•	Calling Line Identification Restriction
+CCWA	•	•	•	•	•	•	Call Waiting
+CHLD	•	•	•	•	•	•	Call Holding Services
+CLCC	•	•	•	•	•	•	List Current Calls
ETSI GSM 07.07/27.007 – Mobile Equipment Control							
+CPAS	•	•	•	•	•	•	Phone Activity Status
+CFUN	•	•	•	•	•	•	Set Phone Functionality
+CSQ	•	•	•	•	•	•	Signal Quality
+CPBS	•	•	•	•	•	•	Select Phonebook Memory Storage
+CPBR	•	•	•	•	•	•	Read Phonebook Entries
+CPBF	•	•	•	•	•	•	Find Phonebook Entries
+CPBW	•	•	•	•	•	•	Write Phonebook Entry
+CCLK	•	•	•	•	•	•	Clock Management
+CALA	•	•	•	•	•	•	Alarm Management
+CALM	•	•	•	•	•	•	Alert Sound Mode
+CRSL	•	•	•	•	•	•	Ringer Sound Level
+CLVL	•	•	•	•	•	•	Loudspeaker Volume Level
+CMUT	•	•	•	•	•	•	Microphone Mute Control
+CLAC	•	•	•	•	•	•	Available AT commands
+CALD	•	•	•	•	•	•	Delete Alarm
ETSI GSM 07.07/27.007 – Mobile Equipment Errors							
+CMEE	•	•	•	•	•	•	Report Mobile Equipment Error
ETSI GSM 07.077/27.007 – Voice Control							
+VTS	•	•	•	•	•	•	DTMF Tones Transmission
+VTD	•	•	•	•	•	•	Tone Duration
ETSI GSM 07.077/27.007 – Commands For Battery Charger							
+CBC	•	•	•	•	•	•	Battery Charge

ETSI GSM 07.05/27.005 – General Configuration							
+CSMS	•	•	•	•	•	•	Select Message Service
+CPMS	•	•	•	•	•	•	Preferred Message Storage
+CMGF	•	•	•	•	•	•	Message Format
ETSI GSM 07.05/27.005 – Message Configuration							
+CSMP	•	•	•	•	•	•	Set Text Mode Parameters
+CSDH	•	•	•	•	•	•	Show Text Mode Parameters
+CSAS	•	•	•	•	•	•	Save Settings
+CRES	•	•	•	•	•	•	Restore Settings
ETSI GSM 07.05/27.005 – Message Receiving And Reading							
+CNMI	•	•	•	•	•	•	New Message Indications To Terminal Equipment
+CMGL	•	•	•	•	•	•	List Messages
+CMGR	•	•	•	•	•	•	Read Message
ETSI GSM 07.05/27.005 – Message Sending And Writing							
+CMGS	•	•	•	•	•	•	Send Message
+CMSS	•	•	•	•	•	•	Send Message From Storage
+CMGW	•	•	•	•	•	•	Write Message To Memory
+CMGD	•	•	•	•	•	•	Delete Message
Custom AT Commands – General Configuration							
#CGMI	•	•	•	•	•	•	Manufacturer Identification
#CGMM	•	•	•	•	•	•	Model Identification
#CGMR	•	•	•	•	•	•	Revision Identification
#CGSN	•	•	•	•	•	•	Product Serial Number Identification
#CIMI	•	•	•	•	•	•	International Mobile Subscriber Identity (IMSI)
#MEID	•	•	•	•	•	•	Mobile Equipment Identifier
#SHDN	•	•	•	•	•	•	Software Shut Down
#FASTSHDN	•	•	•	•		•	Fast power down
#Z	•	•	•	•	•	•	Extended Reset
#REBOOT	•	•	•	•	•	•	Reboot
$RESET	•	•	•	•	•	•	Reset
#WAKE	•	•	•	•	•	•	Wake From Alarm Mode
#QTEMP	•	•	•	•	•	•	Query Temperature Overflow
#TEMPMON	•	•	•	•	•	•	Temperature monitor
#GPIO	•	•	•	•	•	•	General Purpose Input/Output Pin Control

#SLED	●	●	●	●	●	●	STAT_LED GPIO Setting
#SLEDSAV	●	●	●	●	●	●	Save STAT_LED GPIO Setting
#E2SMSRI	●	●	●	●	●	●	SMS Ring Indicator
#ADC	●	●	●	●	●	●	Analog/Digital Converter Input
#DAC	●	●	●	●	●	●	Digital/Analog Converter Control
#VAUX	●	●	●	●	●	●	Auxiliary Voltage Output Control
#VAUXSAV	●	●	●	●	●	●	#VAUX Saving
#V24CFG	●	●	●	●	●	●	V24 Output Pins Configuration
#V24	●	●	●	●	●	●	V24 Output Pins Control
#CBC	●	●	●	●	●	●	Battery and Charger Status
#DIALMODE	●	●	●	●	●	●	ATD Dialing Mode
#ACAL	●	●	●	●	●	●	Automatic Call
#ACALEXT	●	●	●	●	●	●	Extended Automatic Call
#ECAM	●	●	●	●	●	●	Extended Call Monitoring
#SMOV	●	●	●	●	●	●	SMS Overflow
#CODEC	●	●	●	●	●	●	Audio Codec
#NITZ	●	●	●	●	●	●	Network Timezone
#SKIPESC	●	●	●	●	●	●	Skip Escape Sequence
#E2ESC	●	●	●	●	●	●	Escape Sequence Guard Time
#GAUTH	●	●	●	●	●	●	PPP Connection Authentication Type
#RTCSTAT	●	●	●	●	●	●	RTC Status
#PSMRI	●	●	●	●	●	●	Power Saving Mode Ring Indicator
#CFLO	●	●	●	●	●	●	Command Mode Flow Control
#FILEPWD		●	●	●			Change and insert file systm password
#GSMAD	●	●	●	●	●	●	GSM Antenna Detection
#I2CWR	●	●	●	●	●	●	I2C data via GPIO
#I2CRD	●	●	●	●	●	●	I2C data from GPIO
#CSQLED	●	●	●	●	●	●	Control GPIOs based on Signal Strength
#MONI	●	●	●	●	●	●	Cell Monitor
#CPBD	●	●	●	●	●	●	Delete All Phonebook Entries
#TESTMODE	●	●	●	●		●	Enable Test Mode command in not signaling mode
#CMUXMODE	●	●	●	●	●	●	CMUX Mode Set
#RSSICFG		●	●	●			RSSI Configuration
#RIND							Received signal strength indicaton

Custom AT Commands – Audio AT commands							
#CAP	•	•	•	•	•	•	Change Audio Path
#OAP	•	•	•	•	•	•	Open Audio Loop
#SRS	•	•	•	•	•	•	Select Ringer Sound
#SRP	•	•	•	•	•	•	Select Ringer Path
#STM	•	•	•	•	•	•	Signaling Tones Mode
#TONE	•	•	•	•	•	•	Tone Playback
#TSVOL	•	•	•	•	•	•	Tone Classes Volume
#DVI	•	•	•	•	•	•	Digital Voiceband Interface
#DVICFG	•	•	•	•	•	•	DVI configuration
#AXE	•	•	•	•	•	•	AXE Pin Reading
#SHFEC	•	•	•	•	•	•	Handsfree Echo Canceller
#HFMICG	•	•	•	•	•	•	Handsfree Microphone Gain
#HSMICG	•	•	•	•	•	•	Handset Microphone Gain
#SHFSD	•	•	•	•	•	•	Set Headset Sidetone
#SPKMUT	•	•	•	•	•	•	Speaker Mute Control
#HFRECG	•	•	•	•	•	•	Handsfree Receiver Gain
#HSRECG	•	•	•	•	•	•	Handset Receiver Gain
#PRST	•	•	•	•	•	•	Audio Profile Factory Configuration
#PSAV	•	•	•	•	•	•	Audio Profile Configuration Save
#PSEL	•	•	•	•	•	•	Audio Profile Selection
#PSET	•	•	•	•	•	•	Audio Profile Setting
#SHFAGC	•	•	•	•	•	•	Handsfree Automatic Gain Control
#SHFNR	•	•	•	•	•	•	Handsfree Noise Reduction
#SHSAGC	•	•	•	•	•	•	Handest Automatic Gain
#SHSEC	•	•	•	•	•	•	Handset Echo Canceller
#SHSNR	•	•	•	•	•	•	Handset Noise Reduction
#SHSSD	•	•	•	•	•	•	Set Handset Sidetone
#PCMTXG	•	•	•	•	•	•	PCM Tx Volume
#PCMRXG	•	•	•	•	•	•	PCM Rx Volume
#SHFAGCRX	•	•	•	•	•	•	Handsfree RX AGC Value tuning
#SHFAGCTX	•	•	•	•	•	•	Handsfree TX AGC Value tuning
#SHSAGCRX	•	•	•	•	•	•	Handset RX AGC Value tuning
#SHSAGCTX	•	•	•	•	•	•	Handset TX AGC Value tuning

#SRXAGC	•	•	•	•	•	•	RX AGC enable
#SHSFRX	•	•	•	•	•	•	Handset RX filter coefficients values
#SHSFTX	•	•	•	•	•	•	Handset TX filter coefficients values
#SHFFRX	•	•	•	•	•	•	Handsfree RX filter coefficients values
#SHFFTX	•	•	•	•	•	•	Handsfree TX filter coefficients values
#DTMF	•	•	•	•	•	•	Embedded DTMF decoder enabling
#SPCM	•	•	•	•	•	•	PCM Play and Receive
Custom AT Commands – Multisocket							
#SS	•	•	•	•	•	•	Socket Status
#SI	•	•	•	•	•	•	Socket Info
#SGACT	•	•	•	•	•	•	Context Activation
#SGACTCFG	•	•	•	•		•	Context activation and configuration
#SGACTCFGEXT	•	•	•	•		•	Context Activation and Configuration Extended
#SH	•	•	•	•	•	•	Socket Shutdown
#SCFG	•	•	•	•	•	•	Socket Configuration
#SCFGEXT	•	•	•	•	•	•	Socket Configuration Extended
#SCFGEXT2	•	•	•	•	•	•	Socket Configuration Extended2
#CGPADDR	•	•	•	•	•	•	Show Address
#SD	•	•	•	•	•	•	Socket Dial
#SA	•	•	•	•	•	•	Socket Accept
#SO	•	•	•	•	•	•	Socket Restore
#SL	•	•	•	•	•	•	Socket Listen
#SLUDP	•	•	•	•	•	•	Socket Listen UDP
#SRECV	•	•	•	•	•	•	Received Data In Command Mode
#SSEND	•	•	•	•	•	•	Send Data In Command Mode
#SSENDEXT	•	•	•	•	•	•	Send Data In Command Mode Extended
#SLASTCLOSURE	•	•	•	•	•	•	Detect the cause of a socket disconnection
Custom AT Commands - FTP							
#FTPTO	•	•	•	•	•	•	FTP Time-Out
#FTPOPEN	•	•	•	•	•	•	FTP Open
#FTPCLOSE	•	•	•	•	•	•	FTP Close
#FTPPUT	•	•	•	•	•	•	FTP Put
#FTPGET	•	•	•	•	•	•	FTP Get
#FTPGETPKT	•	•	•	•		•	FTP Get in command mode

#FTPTYPE	•	•	•	•	•	•	FTP Type
#FTPMSG	•	•	•	•	•	•	FTP Read Message
#FTPDELE	•	•	•	•	•	•	FTP Delete
#FTPPWD	•	•	•	•	•	•	FTP Print Working Directory
#FTPCWD	•	•	•	•	•	•	FTP Change Working Directory
#FTPLIST	•	•	•	•	•	•	FTP List
#FTPRECV	•	•	•	•		•	Receive data in command mode
#FTPAPP			•				FTP Append
#FTPAPPEXT			•				FTP Append Extended
#FTPCFG			•				FTP Config
#FTPFSIZE			•				Get file size
#FTPREST			•				Set restart position
Custom AT Commands – Enhanced Easy GPRS® Extension							
#USERID	•	•	•	•	•	•	Authentication User ID
#PASSW	•	•	•	•	•	•	Authentication Password
#PKTSZ	•	•	•	•	•	•	Packet Size
#DSTO	•	•	•	•	•	•	Data Sending Time-Out
#SKTTO	•	•	•	•	•	•	Socket Inactivity Time-Out
#SKTSET	•	•	•	•	•	•	Socket Definition
#SKTOP	•	•	•	•	•	•	Socket Open
#QDNS	•	•	•	•	•	•	Query DNS
#CACHEDNS	•	•	•	•	•	•	DNS Response Caching
#DNS	•	•	•	•	•	•	Manual DNS Selection
#SKTCT	•	•	•	•	•	•	Socket TCP Connection Time-Out
#SKTSAV	•	•	•	•	•	•	Socket Parameters Save
#SKTRST	•	•	•	•	•	•	Socket Parameters Reset
#CDMADC	•	•	•	•	•	•	CDMA Data Connection
#SKTD	•	•	•	•	•	•	Socket Dial
#SKTL	•	•	•	•	•	•	Socket Listen
#E2SLRI	•	•	•	•	•	•	Socket Listen Ring Indicator
#FRWL	•	•	•	•	•	•	Firewall Setup
#GDATAVOL	•	•	•	•	•	•	PPP Data Volume
#ICMP	•	•	•	•	•	•	ICMP Support
#PING	•	•	•	•	•	•	PING Request

#TCPMAXDAT	●	●	●	●	●	●	Maximum TCP Payload Size
#TCPREASS	●	●	●	●	●	●	TCP reassembly
Custom AT Commands – E-Mail Management							
#ESMTP	●	●	●	●	●	●	E-mail SMTP Server
#EADDR	●	●	●	●	●	●	E-mail Sender Address
#EUSER	●	●	●	●	●	●	E-mail Authentication User Name
#EPASSW	●	●	●	●	●	●	E-mail Authentication Password
#SEMAIL	●	●	●	●	●	●	E-mail Sending With PPP Context Activation
#EMAILACT	●	●	●	●	●	●	E-mail PPP Context Activation
#EMAILD	●	●	●	●	●	●	E-mail Sending
#ESAV	●	●	●	●	●	●	E-mail Parameters Save
#ERST	●	●	●	●	●	●	E-mail Parameters Reset
#EMAILMSG	●	●	●	●	●	●	SMTP Read Message
Custom AT Commands – HTTP							
#HTTPCFG	●	●	●	●	●	●	Configure HTTP parameters
#HTTPQRY	●	●	●	●	●	●	Send HTTP GET, HEAD or DELETE request
#HTTPSND	●	●	●	●	●	●	Send HTTP POST or PUT request
#HTTPRCV	●	●	●	●	●	●	Receive HTTP server data
Custom AT Commands – SSL AT commands							
#SSLCFG			●				Configure general parameters of a SSL socket
#SSLD			●				Opening a socket SSL to a remote server
#SSLEN			●				Enabling a SSL socket
#SSLH			●				Closing a SSL socket
#SSLO			●				Restoring a SSL socket after a +++
#SSLRECV			●				Reading data from a SSL socket
#SSLS			●				Reporting the status of a SSL socket
#SSLSECDATA			●				Managing the security data
#SSLSEND			●				Sending data through a SSL socket
#SSLSENDEXT			●				Sending data through a secure socket in Command Mode extended
#SSLSECCFG			●				Configure security parameters of a SSL socket
Custom AT Commands – AT Run Commands							
#SMSATRUN	●	●	●	●	●	●	Enable SMS AT Run service
#SMSATRUNCFG	●	●	●	●	●	●	Set SMS AT Run Parameters
#SMSATWL	●	●	●	●	●	●	SMS AT Run White List

#TCPATRUNCFG	•	•	•	•	•	•	Set TCP AT Run Service Parameters
#TCPATRUNL	•	•	•	•	•	•	Set TCP AT Run Service in listen (server) mode
#TCPATRUNFRWL	•	•	•	•	•	•	TCP AT Run Firewall List
#TCPATRUNAUTH	•	•	•	•	•	•	TCP AT Run Authentication Parameters List
#TCPATRUND	•	•	•	•	•	•	Enable TCP AT Run Service in dial (client) mode
#TCPATRUNCLOSE	•	•	•	•	•	•	Closing TCP Run AT socket
#TCPATCMDSEQ	•	•	•	•	•	•	TCP AT Run Command Sequence
#TCPATCONSER	•	•	•	•	•	•	TCP Run AT Service to a Serial Port
#ATRUNDELAY	•	•	•	•	•	•	Run AT command execution Delay
Custom AT Commands – Event Monitor Commands							
#ENAEVMONI	•	•	•	•	•	•	Enable EvMoni Service
#ENAEVMONICFG	•	•	•	•	•	•	Set EvMoni Service Parameters
#EVMONI	•	•	•	•	•	•	Event Monitoring
#CMGS	•	•	•	•	•	•	Send Message
#CMGW	•	•	•	•	•	•	Write Message to Memory
Custom AT Commands – IoT Portal Commands							
#DWCFG		•	•	•			Configure deviceWISE parameters
#DWCONN		•	•	•			Connect to M2M Service
#DWSTATUS		•	•	•			Query connection status
#DWSEND		•	•	•			Send data to M2M Service
#DWSENDR		•	•	•			Send raw data to M2M Service
#DWRCV		•	•	•			Receive data from M2M Service
#DWRCVR		•	•	•			Receive raw data from M2M Service
#DWLRCV		•	•	•			List information on message pending from M2M Service
#DWEN		•	•	•			Enable agent features
Custom AT Commands – Generic Configuration AT Commands							
#CAI	•	•	•	•	•	•	Common Air Interface parameters
#MODEM	•	•	•	•	•	•	Modem Configuration parameters
#ENG	•	•	•	•	•	•	Mobile NAM parameters
#NOTI	•	•	•	•	•	•	CDMA Notification
$MDN	•	•	•	•	•	•	Mobile Directory Number
$MSID	•	•	•	•	•	•	Mobile Station ID
+SERVICE	•	•	•	•	•	•	Notification of Service
#RTN	•	•	•	•	•	•	Reverse Logistic Support

$CELLPOS	•	•	•	•	•	•	Base Station Lat/long Data
Custom AT Commands – Air interface and call processing							
#PREFRC	•	•	•	•	•	•	Preferred Radio Configuration
#VOICEPRIV	•	•	•	•	•	•	Voice Privacy Setting
#PREFVOC	•	•	•	•	•	•	Vocoder Setting Value Reading or Writing
#OTASPEN	•	•	•	•	•	•	OTASP Setting
+CFG	•	•	•	•	•	•	Configuration String
+CRM	•	•	•	•	•	•	RM Interface Setting
#CLRMRU	•	•	•	•	•	•	Clear MRU Table
Custom AT Commands – DATA Session AT Commands							
+CTA	•	•	•	•	•	•	Data Inactivity Timer
+PZID	•	•	•	•	•	•	Packet Zone ID
$GODORMANT	•	•	•	•	•	•	Interrupt Packet Data
#TESTORI	•	•	•	•	•	•	Test Origination
Custom AT Commands – RUIM specific AT commands							
#QSS						•	Query RUIM STATUS
+CPIN						•	Enter PIN
+CLCK						•	Facility Lock/Unlock
+CPWD						•	Change Facility Password
#CCID						•	Read ICCID (Integrated Circuit Card Identification)
+CCID						•	Read ICCID (Integrated Circuit Card Identification)
#PCT						•	Display remained PIN Counter
#SPN						•	Service Provider Name
#CHVEN						•	Enable/ Disable CHV
Custom AT Commands – SIM Toolkit AT Commands							
#STIA						•	SIM Toolkit Interface Activation
#STGI						•	SIM Toolkit Get Information
#STSR						•	SIM Toolkit Send Response
Custom AT Commands – QCT Proprietary AT Commands							
$QCMIPNAI	•	•	•	•	•		Network Access Identifier
$QCMIPPHA	•	•	•	•	•		Primary Home Agent Address
$QCMIPSHA	•	•	•	•	•		Secondary Home Agent Address
$QCMIPHA	•	•	•	•	•		Home Address
$QCMIPMHSSX	•	•	•	•	•		Home Agent Shared Secret

$QCMIPMASSX	•	•	•	•	•		AAA Server Shared Secret
$QCMIPMHSPI	•	•	•	•	•		Home Agent Security Parameter Index
$QCMIPMASPI	•	•	•	•	•		AAA Server Security Parameter Index
$QCMIPRT	•	•	•	•	•		Reverse Tunneling Preference
$QCMIP	•	•	•	•	•		Enable/Disable Mobile IP
$QCMIPP	•	•	•	•	•		Active MIP Profile Selection
$QCMIPEP	•	•	•	•	•		Enable/Disable Current MIP Profile
$QCMIPGETP	•	•	•	•	•		Profile Information
$QCMIPMASS	•	•	•	•	•		MN-AAA Shared Secrets
$QCMIPMHSS	•	•	•	•	•		MN-HA Shared Secrets
$QCMDR	•	•	•	•	•		Medium Data Rate
Custom AT Commands – FOTA/OMA-DM AT commands							
#OMADMSVADDR			•				OMA-DM Server Address
#OMADMSVPORT			•				OMA-DM Server Port
#OMADMPROXY			•				OMA-DM Proxy Server Address
#OMADMSVID			•				OMA-DM Server ID
#OMADMSVPW			•				OMA-DM Server Password
#OMADMSVNON			•				OMA-DM Server Auth Data
#OMADMCUID			•				OMA-DM Client ID
#OMADMCUPW			•				OMA-DM Client Password
#OMADMCUNON			•				OMA-DM Client Auth Data
#OMADMCEN			•				OMA-DM Client Enable/Disable
+OMADM			•				OMA-DM Device Configuration
+PRL			•				OMA-DM NIPRL/CIPRL
+FUMO			•				OMA-DM NIFUMO/CIFUMO
#HFA			•				Hands Free Activation
#DCCANCEL			•				Device Configuration Cancel
#PRLCANCEL			•				Load PRL Cancel
#FUMOCANCEL			•				FUMO session cancel
#HFACANCEL			•				Hands Free Activation Cancel
Custom AT Commans – Easy Script Extension – Python Interpreter							
#WSCRIPT		•	•	•			Write Script
#ESCRIPT		•	•	•			Select Active Script

#STARTMODESCR		•	•	•			Script Execution Start Mode
#EXECSCR		•	•	•			Execute Active Script
#RSCRIPT		•	•	•			Read Script
#LSCRIPT		•	•	•			List Script Names
#LCSCRIPT		•	•	•			List Script Names with CRC16 info
#DSCRIPT		•	•	•			Delete Script
Custom AT Commands – Verizon Specific AT commands							
#MEIDESN	•	•				•	Read MEID & ESN
#ALERTSND	•	•					Alert Sound Setting
#EMERGALERT	•	•					Emergency Call Tone Setting
#NAMLOCK	•	•					NAM Lock
+VCMGR	•	•					Read Message
+VCMGL	•	•					List Message
#SMSMOEN	•	•					SMS Mobile Origination
#SMSSO	•	•					Service Option for SMS
#SMSPSIZ	•	•					Set Payload Length
#SMSAC	•	•					Select transport method to send SMS
$PRL	•	•			•		Preferred Roaming List
#BANDCLS	•	•				•	Display Current Band Class
#DEFAULTBAND	•	•			•		Set Default Band
#ERI	•	•					Enhanced Roaming Indicator
#ERIDATA	•	•					Enhanced Roaming Indicator Version
$ONECALL	•	•					Call for only one phone number
$MIPRMNAI	•	•					Tethered NAI Management for MIP
$SIPRMNAI	•	•					Tethered NAI Management for SIP
Custom AT Commands – Sprint & Aeris.Net specific AT Commands							
+E			•	•			Command Echo
+Q			•	•			Quite Result Code
+V			•	•			Response Format
$FWREV			•	•			Firmware Revision
$MIPERR			•	•			Mobile IP Error
$DIAG			•	•			Diagnostic Port Setting
Custom AT Commands – Sprint specific AT Commands							
$1XRXPWR			•				Current Receive Signal Strength Indicator for 1xRTT

$1XECIO			•				Current Ec/Io for 1xRTT
+LIST			•				List commands
$ROAM			•				Roaming Reference
$ERI			•				Current Roaming Indicator
Custom AT Commands – Aries specific AT Commands							
#CURRNAM				•			Current NAM
#PRLDATA				•			PRL data
#ESN				•			ESN data
+ESN				•			ESN data
#PRI				•			PRI version
Custom AT Commands – Telit Test AT Commands							
#MODE	•	•	•	•	•	•	Change Operational Mode of Modem

3.5. AT Commands References

3.5.1. Command Line General Format

3.5.1.1. Command Line Prefixes

3.5.1.1.1. *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

3.5.1.1.2. *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 be 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

3.5.1.1.3. *Repeat Last Command - #/*

#/ - Repeat Last Command	
AT#/	Execute command is used to execute again the last received command.

3.5.2. General Configuration Commands

3.5.2.1.1. *Select Interface Style - #SELINT*

#SELINT - Select interface style	
AT#SELINT=<v>	Set command sets the AT command interface style depending on parameter <v>. Parameter: **<v>** - AT command interface 2 - switches the AT command interface style of the product, to CE910-Series
AT#SELINT?	Read command reports the current interface style.
AT#SELINT=?	Test command reports the available range of values for parameter **<v>**.
Note	It is suggested to reboot the module after every **#SELINT** setting.

3.5.2.1.2. *Set Notification Port - #NOPT*

#NOPT - Set notification port	

AT#NOPT=<num>	Set command specifies the port print out Notification (URC) messages
	
	Parameter:
	<num> - Notification Port
	  0 – All Ports; URC messages are sent to all ports. < default value >
	  1 – Main UART Port only
	  2 – Telit USB Modem Port only
	  3 – Multiplxer DLCI1 Port only
	  4 – Multiplxer DLCI2 Port only
	  5 – Multiplxer DLCI3 Port only
	  6 – Multiplxer DLCI4 Port only
	  7 – Telit USB Diagnostic Port only
	  8 – Python MDM Port only
	  9 – Python MDM2 Port only
	  10 – ATRUN SMS Port only
	  11 – ATRUN TCP Port only
	
	Note : URC message sent out on this port only if the port is opened for AT interface and enabled as notification(URC) service.
	
	Note : If the port is closed and enabled as notification(URC) service, URC message will be discarded.

#NOPT - Set notification port	
	Note : Main UART & Telit USB Modem Ports opened for AT interface at power on time, automatically and other ports opened by the specific behaviour, as below.
	Multiplexer DLCI 1-4 Ports : Multiplexer(+CMUX) is running
	Python MDM 1-2 Ports : Python VM is running
	ATRUN SMS/TCP Ports : ATRUN is running
	Note : The notification output on Telit USB Diagnotic Port is available, only if AT#DIAGCFG setting value is 1.
AT#NOPT?	Read command reports the current notification port.
AT#NOPT=?	Test command reports the available range of values for parameter **<num>**.

3.5.2.1.3. *Manufacturer Serial Number - #MSN*

#MSN - Manufacturer serial Number	
AT#MSN	Returns the device board serial number.
	Note: the format of the numbers in output is always 7digits, left-filled with 0s
AT#MSN=?	Test command returns **OK** result code.

3.5.2.1.4. *Hardware revision - #HWREV*

#HWREV - Hardware revision	
AT#HWREV	Execution command returns the device Hardware revision identification code without command echo.
AT#HWREV=?	Test command returns the **OK** result code.

3.5.2.1.5. *Diagnostic Port Configuration - #DIAGCFG*

#DIAGCFG - Diagnostic Port Configuration	
AT#DIAGCFG=<mode>	Set command configure the mode of Telit Diagnostic Port
	Parameter:
	<**mode**>
	0 - Telit Diagnostic Port used as the diagnostic channel (default)
	1 - Telit Diagnostic Port used as AT channel
	2 - Telit Diagnostic Port used as Python script debugging channel
	Note: If mode-1 enabled, the diagnostic channel changed to UART2.
	Note: In the mode-1, AT channel doesn't support the control line (DCD/RI/DSR/DTR) behavior
	Note: mode-2 is available only if Python script supported.
	Note: If mode-2 enabled, the diagnostic channel will be unavailable and UART2 will be reserved for Python SER2 built-in module.
	Note: If this command performed, successfully, the device will be reset, automatically and new setting applied at the next boot-up.
	Note: This setting stored in NVM area.

#DIAGCFG - Diagnostic Port Configuration	
	Note: When upgrading new firmware, mode-0 MUST be enabled. (F/W available on Telit USB diagnostic port)
	Note: When debugging Python script, mode-0 or mode-2 should be enabled.
AT#DIAGCFG?	Read command reports the current diagnostic port configuration.
AT#DIAGCFG=?	Test command reports the available range of values for parameter **<mode>**.

3.5.3. Hayes Compliant AT Commands

3.5.3.1. Generic Modem Control

3.5.3.1.1. *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 the factory profile base section parameters are considered. 1 - either the factory profile base section or the extended section is considered (full factory profile). Note: if parameter **<value>** is omitted, the command has the same behaviour as **AT&F0**
Reference	V25ter.

3.5.3.1.2. *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 be terminated. Note: if parameter **<n>** is omitted, the command has the same behaviour as **ATZ0**.
Reference	V25ter.

3.5.3.1.3. *Select Active Service Class - +FCLASS*

+FCLASS - Select Active Service Class	

AT+FCLASS=<n>	Set command sets the wireless module in specified connection mode (data, fax, voice); hence, all the calls done afterwards will be data or voice.
	
	Parameter:
	<n>
	    0 - data (factory default)
	    1 - fax class 1 (only for backward compatibility)
	    2.0- fax class 2.0 (only for backward compatibility)
	    8 - voice
	  Note: CE910 doesn't support FAX

AT+FCLASS?	Read command returns the current configuration value of the parameter **<n>**.
AT+FCLASS=?	Test command returns all supported values of the parameters **<n>**.
Reference	3GPP TS 27.007

3.5.3.1.4. *Default Reset Basic Profile Designation - &Y*

&Y - Default Reset Basic Profile Designation	
AT&Y[<n>]	Execution command defines the basic profiles that will be loaded on start up. Parameter: **<n>** 0..1 - profile (default is 0): the wireless module is able to store 2 complete configurations (see **&W**). Note: differently from command **Z<n>**, which loads just once the desired profile, the one chosen through command **&Y** will be loaded on every start up. Note: if parameter is omitted, the command has the same behaviour as **AT&Y0**

3.5.3.1.5. *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 start up. 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 start up. Note: if parameter is omitted, the command has the same behaviour as **AT&P0**
Reference	Telit Specifications

3.5.3.1.6. *Store Current Configuration - &W*

&W - Store Current Configuration	
AT&W[<n>]	Execution command stores on profile **<n>** the complete configuration of the device. Parameter: **<n>** 0..1 - profile Note: if parameter is omitted, the command has the same behaviour of **AT&W0**.

3.5.3.1.7. *Store Telephone Number In The Module Internal Phonebook - &Z*

&Z - Store Telephone Number In The Wireless Module Internal Phonebook	
AT&Z<n>=<nr>	Execution command stores in the record **<n>** the telephone number **<nr>**. The records cannot be overwritten; they must be cleared before rewriting. Parameters: **<n>** - phonebook record **<nr>** - telephone number (string type) Note: **<nr>** should be inputted without the double quotation mark (""). Note: the wireless module has a built in non-volatile memory in which 10 telephone numbers of a maximum 24 digits can be stored. Note: to delete the record **<n>** the command **AT&Z<n>=<CR>** must be issued. Note: the records in the module memory can be viewed with the command **&N**, while the telephone number stored in the record *n* can be dialed by giving the command **ATDS=<n>**.

3.5.3.1.8. *Display Internal Phonebook Stored Numbers - &N*

&N - Display Internal Phonebook Stored Numbers	
AT&N[<n>]	Execution command returns the telephone number stored at the <n> position in the internal memory. Parameter: <n> - phonebook record number Note: if parameter <n> is omitted then all the internal records are shown.

3.5.3.1.9. *Manufacturer Identification - +GMI*

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

3.5.3.1.10. *Model Identification - +GMM*

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

3.5.3.1.11. *Revision Identification - +GMR*

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

3.5.3.1.12. *Capabilities List - +GCAP*

+GCAP - Capabilities List	
AT+GCAP	Execution command returns the equipment supported command set list. Where: **+CIS707-A**: IS-707-A (High Speed Packet Data Services) command set **+FCLASS**: Fax command set **+ES**: Error Control Selection command set **+DS**: Data Service common modem command set **+MS**: Mobile Specific command set Note: CE910 doesn't support FAX
Reference	V.25ter

3.5.3.1.13. *Serial Number - +GSN*

+GSN - Serial Number	
AT+GSN	Verizon&RUIM version: Execution command returns the device board serial

+GSN - Serial Number	
	number in 8-digit decimal. Note: The number returned is not the IMSI, it is only the board number Sprint&Aeris version: Execution command returns the "<ESN> " or the "<MEID>:<pseudo ESN> of the device. Execution command returns the decimal value on the first line and the hexadecimal value on the second line.
Reference	V.25ter

3.5.3.1.14.   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.

3.5.3.1.15.   Display Current Configuration And Profile – &V0

&V0 - Display Current Configuration And Profile	
AT&V0	Execution command returns all the configuration parameters settings. Note: this command is the same as **&V**, it is included only for backwards compatibility.

3.5.3.1.16.   S Registers Display – &V1

&V1 - S Registers Display	
AT&V1	Execution command returns the value of the **S** registers in decimal and hexadecimal value in the format: REG DEC HEX <reg0> <dec> <hex> <reg1> <dec> <hex> ... where **<regn>** - **S** register number 000..005 007 012 025 038 **<dec>** - current value in decimal notation **<hex>** - current value in hexadecimal notation

3.5.3.1.17. *Extended S Registers Display - &V3*

&V3 - Extended S Registers Display	

AT&V3	Execution command returns the value of the **S** registers in decimal and hexadecimal value in the format:
	
	  REG DEC HEX
	  \<reg0\> \<dec\> \<hex\>
	  \<reg1\> \<dec\> \<hex\>
	  ...
	 where
	 \<reg*n*\> - **S** register number
	  000..005
	  007
	  012
	  025
	  030
	  038
	 \<dec\> - current value in decimal notation
	 \<hex\> - current value in hexadecimal notation

3.5.3.1.18. *Display Last Connection Statistics - &V2*

&V2 - Display Last Connection Statistics	
AT&V2	Execution command returns the last connection statistics & connection failure reason.

3.5.3.1.19. *Single Line Connect Message - \V*

\V - Single Line Connect Message	

AT\V\<n\>	Execution command sets single line connect message.
	
	 Parameter:
	 \<n\>
	  0 - off
	  1 - on

3.5.3.1.20. *Country Of Installation - +GCI*

+GCI - Country Of Installation	
AT+GCI=<code>	Set command selects the installation country code according to ITU-T.35 Annex A. Note: CE910-SC doesn't support the write command. Because it's a read-only item from an inserted RUIM.
AT+GCI?	Read command reports the currently selected country code. Note: CE910-SC only support the read command when RUIM is inserted.
AT+GCI=?	Test command reports the supported country codes.
Reference	V25ter.

3.5.3.1.21. *Line Signal Level - %L*

%L - Line Signal Level	
AT%L	It has no effect and is included only for backward compatibility with landline modems

3.5.3.1.22. *Line Quality - %Q*

%Q - Line Quality	
AT%Q	It has no effect and is included only for backward compatibility with landline modems

3.5.3.1.23. *Speaker Loudness - L*

L - Speaker Loudness	
ATL<n>	It has no effect and is included only for backward compatibility with landline modems

3.5.3.1.24. *Speaker Mode - M*

M - Speaker Mode	
ATM<n>	It has no effect and is included only for backward compatibility with landline modems

3.5.3.2. DTE - Modem Interface Control

3.5.3.2.1. *Command Echo - E*

E - Command Echo	
ATE[<n>]	Set command enables/disables the command echo. Parameter:

E - Command Echo	
	<n> 0 - disables command echo 1 - enables command echo (factory default) , hence command sent to the device are echoed back to the **DTE** before the response is given. Note: if parameter is omitted, the command has the same behaviour of **ATE0**
Reference	V25ter

3.5.3.2.2. *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) 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+CREG=? **+CREG: (0-2) nothing is appended to the response**
Reference	V25ter

3.5.3.2.3. *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 3.2.3[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

3.5.3.2.4. Extended Result Codes - X

X - Extended Result Codes	

ATX[<n>]	Set command selects the result code messages subset used by the modem to inform the **DTE** of the result of the commands.
	
	Parameter:
	<n>
	  0 - send only **OK**, **CONNECT**, **RING**, **NO CARRIER**, **ERROR**, **NO ANSWER** results.
	  1..4 - reports all messages (factory default is 1).
	
	Note: If parameter is omitted, the command has the same behaviour of **ATX0**
	
	Note: Current value is returned by AT&V
	Parameter:
	<n>
	0 - EXTENDED MESSAGES : X0=NO
	1..4 - EXTENDED MESSAGES : X1=YES

Note	For complete control on **CONNECT** response message see also **+DR** command.
Reference	V25ter

3.5.3.2.5. 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

3.5.3.2.6. 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.

&C - Data Carrier Detect (DCD) Control	
	1 - **DCD** follows the Carrier detect status: if carrier is detected **DCD** is high, otherwise **DCD** is **low**. (factory default) 2 - **DCD off** while disconnecting Note: if parameter is omitted, the command has the same behaviour of **AT&C0**
Reference	V25ter

3.5.3.2.7. *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 – device ignores **DTR** transitions (factory default) 1 - when the MODULE is connected, the **High** to **Low** transition of **DTR** pin sets the device in command mode, the current connection is NOT closed 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 – device ignores **DTR** transitions 4 - **C108/1** operation is disabled; same behaviour as for **<n>**=3 5 - **C108/1** operation is enabled; same behaviour as for **<n>**=2 Note: if a connection has been set up issuing either **#SKTD** or **#SKTOP**, then **AT&D1** has the same effect as **AT&D2**. If a connection has been set up issuing AT#SD then AT&D1 and AT&D2 have different effect, as described above. Note: if **AT&D2** has been issued and the **DTR** has been tied **Low**, autoanswering is inhibited and it is possible to answer only issuing command **ATA**. Note: Recommended that AT&D2 is issued prior to dial-up network service from DTE. If DTR event is ignored, DCE could be stuck in dormant state in a situation that DCE is not able to communicate with NW(like No service) and DTE tries to disconnect dial-up service. If voice is activated with data service simultaneously, refer to AT+CVHU command guide. Note: if parameter is omitted, the command has the same behaviour of **AT&D0**
Reference	V25ter

3.5.3.2.8. *Standard Flow Control - \Q*

\Q - Standard Flow Control	
AT\Q[<n>]	Set command controls the RS232 flow control behaviour.

\Q - Standard Flow Control	

	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\Q0**
	
	 Note: Hardware flow control (**AT\Q3**) is not active in command mode.
	
	 Note: **\Q's** settings are functionally a subset of **&K**'s ones.

Reference	V25ter

3.5.3.2.9. *Flow Control - &K*

&K - Flow Control	

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.

3.5.3.2.10. *Data Set Ready (DSR) Control - &S*

&S - Data Set Ready (DSR) Control	

AT&S[<n>]	Set command controls the RS232 **DSR** pin behaviour.
	
	 Parameter:
	 <n>
	   0 - always **High**
	   1 - follows the Data traffic channel indication.
	   2 - **High** when connected
	   3 - **High** when device is ready to receive commands (factory default).
	
	 Note: if option 1 is selected then **DSR** is tied **High** when the device receives from the network the Data traffic channel indication.
	
	 Note: in power saving mode the **DSR** pin is always tied **Low** & USB_VBUS pin is always tied Low.
	
	 Note: if parameter is omitted, the command has the same behaviour of **AT&S0**

&S - Data Set Ready (DSR) Control	
	Note: If option 1 or 2 are active, **DSR** will not tie **High** in case of voice channel

3.5.3.2.11. Ring (RI) Control - \R

\R - Ring (RI) Control	

AT\R[<n>]	Set command controls the **RING** output pin behaviour.
	
	Parameter:
	<n>
	  0 - **RING** on during ringing and further connection
	  1 - **RING** on during ringing (factory default)
	  2 - **RING** follows the ring signal
	
	Note: to check the ring option status use the **&V** command.
	
	Note: if parameter is omitted, the command has the same behaviour of **AT\R0**

3.5.3.2.12. *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; it may be used to fix the **DTE-DCE** interface speed.
	
	NOTE: DTE speed of USB port is always 0. DTE speed of USB does not change.
	
	Parameter:
	\<rate\>
	 -
	  300
	  600
	  1200
	  2400
	  4800
	  9600
	  19200
	  38400
	  57600
	  115200 (default)
	  230400
	  460800
	  921600
	  3200000
	  4000000
	
	If **\<rate\>** is specified and not 0, **DTE-DCE** speed is fixed at that speed, hence no speed auto-detection (autobauding) is enabled.

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

3.5.3.2.13. *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)
	
	 The supported flow control list as follows
	
	 <table><tr><td><by_te></td><td><by_ta></td></tr><tr><td>0</td><td>0</td></tr><tr><td>2</td><td>2</td></tr></table>
	 Note: Hardware flow control (**AT+IFC=2,2**) is not active in command mode.
	 Note: This command is equivalent to **&K** command.

AT+IFC?	Read command returns active flow control settings. Note: If flow control behaviour has been set with **AT&Kn** command with the parameter that is not allowed by **AT+IFC** the read command **AT+IFC?** will return: **+IFC: 0,0**
AT+IFC=?	Test command returns all supported values of the parameters **<by_te>** and **<by_ta>**.
Reference	V25ter

3.5.3.2.14. *DTE-Modem Local Rate Reporting - +ILRR*

+ILRR - DTE-Modem Local Rate Reporting	

AT+ILRR=<n>	Set command controls whether or not the +ILRR: <rate> information text is transmitted from the modem (module) to the DTE.
	Parameter:
	<n>
	  0 - local port speed rate reporting disabled (factory default)
	  1 - local port speed rate reporting enabled
	
	Note: this information if enabled is sent upon connection.

AT+ILRR?	Read command returns active setting of **<n>**.
AT+ILRR=?	Test command returns all supported values of the parameter **<n>**
Reference	V25ter

3.5.3.2.15. *DTE-Modem Character Framing - +ICF*

+ICF - DTE-Modem Character Framing	

AT+ICF=[<format>,[<parity>]]	Set command defines the asynchronous character framing to be used when autobauding is disabled.
	
	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.
	  1 - 8 Date, 2 Stop
	  2 - 8 Data, 1 Parity, 1 Stop
	  3 - 8 Data, 1 Stop (default)
	  5 - 7 Data, 1 Parity, 1 Stop
	<parity> - determines how the parity bit is generated and checked, if present; setting this sub parameter is mandatory and has a meaning only if **<format>** subparameter is either 2 or 5 otherwise is not allowed.
	  0 - Odd (not supported)
	  1 - Even (not supported)

AT+ICF?	Read command returns current settings for sub parameters **<format>** and **<parity>**. If current setting of subparameter **<format>** is neither 2 nor 5, the current setting of subparameter **<parity>** will always be represented as 0.
AT+ICF=?	Test command returns the ranges of values for the parameters **<format>** and **<parity>**
Reference	V25ter
Example	*8N2* AT+ICF=1 OK *8O1* AT+ICF=2,0 OK

+ICF - DTE-Modem Character Framing	
	8E1 AT+ICF=2,1 OK *8N1* AT+ICF = 3 (default) OK *7O1* AT+ICF=5,0 OK *7E1* AT+ICF=5,1 OK

3.5.3.3. Call Control

3.5.3.3.1. *Dial - D*

D – Dial	
ATD<number>[;]	Execution command starts a call to the phone number given as parameter. If ";" is present, a **voice** call to the given number is performed, regardless of the current value of the connection mode set by +**FCLASS** command. Parameter: <number> - phone number to be dialed Note: type of call (**data** or **voice**) depends on last +**FCLASS** setting. Note: the numbers accepted are 0-9 and *#ABCDP,W@!$; Note: for backwards compatibility with landline modems modifiers "P", ",", "W", "!", "@", "$", ";" are accepted.
ATD><str>[;]	Issues a call to phone number which corresponding alphanumeric field is <str>; all available memories will be searched 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.
ATD><n>[;]	Issues a call to phone number in entry location <n> of the active phonebook.

D – Dial	
	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. Parameter: **\<nr>** - internal phonebook position to be called (See commands **&N** and **&Z**)
Example	*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
Note	CE910 series doesn't support CSD call. But ATD respective with phonebook without semicolon works as voice call.
Reference	V25ter.

3.5.3.3.2. *Tone Dial – T*

T - Tone Dial	
ATT	Set command has no effect is included only for backward compatibility with landline modems.
Reference	V25ter.

3.5.3.3.3. *Pulse Dial – P*

P - Pulse Dial	
ATP	Set command has no effect is included only for backward compatibility with landline modems.
Reference	V25ter.

3.5.3.3.4. *Answer – A*

A - Answer	
ATA	Execution command is used to answer to an incoming call if automatic answer is disabled. Note: This command MUST be the last in the command line and must be followed immediately by a **\<CR>** character.
Reference	V25ter.

3.5.3.3.5. *Disconnect - H*

H - Disconnect	
ATH	Execution command is used to close the current conversation (voice, data or fax). Note: this command can be issued only in command mode; when a data conversation is active the device is in on-line mode (commands are not sensed and characters are sent to the other party), hence escape sequence (see **register S2**) is required before issuing this command, otherwise if **&D1** option is active, **DTR** pin has to be tied **Low** to return in command mode.
Reference	V25ter.

3.5.3.3.6. *Return To On Line Mode - O*

O - Return To On Line Mode	
ATO	Execution command is used to return to on-line mode from command mode. If there is no active connection, it returns **NO CARRIER**. Note: After issuing this command, if the device is in conversation, to send other commands to the device you must return to command mode by issuing the escape sequence (see **register S2**) or tying low **DTR** pin if **&D1** option is active.
Reference	V25ter.

3.5.3.3.7. *Guard Tone - &G*

&G - Guard Tone	
AT&G	Set command has no effect is included only for backward compatibility with landline modems.

3.5.3.3.8. *Sync/Async Mode - &Q*

&Q - Sync/Async Mode	
AT&Q	Set command has no effect is included only for backward compatibility with landline modems.

3.5.3.4. Modulation Control

3.5.3.4.1. *Modulation Selection - +MS*

+MS - Modulation Selection	

AT+MS=
<carrier>
[,<automode>
[,<min_rate>
[,<max_rate>]]]
	Set command has no effect is included only for backward compatibility with landline modems.
	
	Parameters:
	<carrier> - a string which specifies the preferred modem carrier to use in originating or answering a connection
	  V21
	  V22
	  V22B
	  V23C
	  V32
	  V34
	<automode> - it enables/disables automatic modulation negotiation.
	  0 - disabled
	  1 - enabled. It has effect only if it is defined for the associated modulation.
	<min_rate> - it specifies the lowest value at which the **DCE** may establish a connection.
	  0 - unspecified
	<max_rate> - it specifies the highest value at which the **DCE** may establish a connection.
	  0 - unspecified
	  300..14400 - rate in bps

AT+MS?	Read command returns the current value of <carrier>, <automode>, <min_rate>, <max_rate> parameters.
AT+MS=?	Test command returns all supported values of the <carrier>, <automode>, <min_rate>, <max_rate> parameters.

3.5.3.4.2. *Line Quality Monitor And Auto Retrain Or Fallback/Fallforward - %E*

%E - Line Quality Monitor And Auto Retrain Or Fallback/Fallforward	
AT%E<n>	Execution command has no effect and is included only for backward compatibility with landline modems.

3.5.3.5. Compression Control

3.5.3.5.1. *Data Compression - +DS*

AT+DS=\<n\>	Set command sets the V42 compression parameter.
	Command has no effect, supported only for the purpose of cross-technology compatibility within products supporting Telit Unified AT-commands.
	Parameter:
	\<n\>
	0 – no compression, currently the only supported value. Returns OK.
	Note. This command has no effect. In 3G CDMA, data compression for CS data and FAX are controlled by the network, not the individual user.
AT+DS?	Returns current data compression setting.
AT+DS=?	Test command returns all supported values of the command.
Reference	V25ter
Example	AT+DS=?
	+DS: (0)
	OK
	AT+DS?
	+DS: 0
	OK
	AT+DS=0
	OK

3.5.3.5.2. *Data Compression Reporting - +DR*

+DR - Data Compression Reporting	
AT+DR=\<n\>	Set command enables/disables the data compression reporting upon connection.
	Parameter:
	\<n\>
	0 - data compression reporting disabled;(default)
	1 - data compression reporting enabled upon connection.
	Note: if enabled, the following intermediate result code is transmitted before the final result code:
	+DR: \<compression\>
	(the only supported value for \<compression\> is "NONE")

+DR - Data Compression Reporting	
AT+DR?	Read command returns current value of **<n>**.
AT+DR=?	Test command returns all supported values of the parameter **<n>**
Reference	V25ter

3.5.3.6. Break Control

3.5.3.6.1. *Transmit Break To Remote - \B*

\B - Transmit Break To Remote	
AT\B	Execution command has no effect and is included only for backward compatibility with landline modems

3.5.3.6.2. *Break Handling - \K*

\K - Break Handling	
AT\K[<n>]	Execution command has no effect and is included only for backward compatibility with landline modems Parameter: **<n>** 0..5

3.5.3.6.3. *Operating Mode - \N*

\N - Operating Mode	
AT\N	Execution command has no effect and is included only for backward compatibility with landline modems

3.5.3.7. 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 sub parameter of an **S-Parameter**, an **ERROR** result code will be issued and the stored value left unchanged.

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

1) **ATS*n*=<*value*><CR>** selects ***n*** as last selected parameter number and sets the contents of the **S*n***-parameter. 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.
2) **AT=<value><CR>** sets the contents of the selected **S-parameter**
3) **AT?** returns the current value of the last S-parameter accessed

Example:
ATS7=10<CR> establishes S7 as last selected parameter and set the contents of S7 to 10
OK
AT=40>CR> sets the content of S7 to 40
OK
AT=15<CR> sets the content of S7 to 15
OK
AT?<CR> returns the current value of S7
015
OK

Reference: V25ter and RC56D/RC336D

3.5.3.7.1. *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

3.5.3.7.2. *Ring Counter – S1*

S1 - Ring Counter	
ATS1	**S1** is incremented each time the device detects the ring signal of an incoming call. **S1** is cleared as soon as no ring occurs. Note: the form **ATS1** has no effect.
ATS1?	Read command returns the value of this parameter.

3.5.3.7.3. *Escape Character – S2*

S2 - Escape Character	
ATS2=<char>	Set command sets the ASCII character to be used as escape character. Parameter: **<char>** - escape character decimal ASCII 0..255 - factory default value is 43 (+). Note: the escape sequence consists of three escape characters preceded and followed by *n* ms of idle (see **S12** to set *n*).
ATS2?	Read command returns the current value of **S2** parameter. Note: the format of the numbers in output is always 3 digits, left-filled with 0s

3.5.3.7.4. *Command Line Termination Character - S3*

S3 - Command Line Termination Character	
ATS3=<char>	Set command sets the value of the character either 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** is 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

3.5.3.7.5. *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** is 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

3.5.3.7.6. *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: **<char>** - command line editing character (decimal ASCII) 0..127 - factory default value is 8 (ASCII **BS**)

S5 - Command Line Editing Character	
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

3.5.3.7.7. *Connection Completion Time-Out - S7*

S7 - Connection Completion Time-Out	
ATS7=<tout>	Set command sets the amount of time, in seconds, that the device shall allow between either answering a call (automatically or by **A** command) or completion of signalling of call addressing information to network (dialling), and establishment of a connection with the remote device. Parameter: **<tout>** - number of seconds 1..255 - factory default value is 60
ATS7?	Read command returns the current value of **S7 parameter**. Note: the format of the numbers in output is always 3 digits, left-filled with 0s
Reference	V25ter

3.5.3.7.8. *Carrier Off With Firm Time - S10*

S10 –Carrier Off With Firm Time	
ATS10=<time>	Set command has no effect and is included only for backward compatibility with landline modems Parameter: **<time>** - expressed in tenths of a second 1..255 - factory default value is 14.
ATS10?	Read command returns the current value of S10 parameter. Note: the format of the numbers in output is always 3 digits, left-filled with 0s

3.5.3.7.9. *Escape Prompt Delay - S12*

S12 - Escape Prompt Delay	
ATS12=\<time\>	Set command sets: 1) the minimum period, before receipt of the first character of the three escape character sequence, during which no other character has to be detected in order to accept it as valid first character; 2) the maximum period allowed between receipt of first or second character of the three escape character sequence and receipt of the next; 3) the minimum period, after receipt of the last character of the three escape character sequence, during which no other character has to be detected in order to accept the escape sequence as a valid one. Parameter: **\<time\>** - expressed in fiftieth of a second 20..255 - factory default value is 50. Note: the minimum period **S12** has to pass after **CONNECT** result code too, before a received character is accepted as valid first character of the three escape character sequence.
ATS12?	Read command returns the current value of **S12 parameter**. Note: the format of the numbers in output is always 3 digits, left-filled with 0s

3.5.3.7.10.   *Delay To DTR Off - S25*

S25 -Delay To DTR Off	
ATS25=\<time\>	Set command defines the amount of time, in hundredths of second, that the device will ignore the **DTR** for taking the action specified by command **&D**. Parameter: **\<time\>** - expressed in hundredths of a second 0..255 - factory default value is 5. Note: the delay is effective only if its value is greater than 5.
ATS25?	Read command returns the current value of **S25 parameter**. Note: the format of the numbers in output is always 3 digits, left-filled with 0s

3.5.3.7.11.   *Disconnect Inactivity Timer - S30*

S30 -Disconnect Inactivity Timer	
ATS30=\<tout\>	Execution command has no effect and is included only for backward compatibility with landline modems.
ATS30?	Read command returns the current value of **S30 parameter**. Note: the format of the numbers in output is always 3 digits, left-filled with 0s

3.5.3.7.12.   *Delay Before Forced Hang Up - S38*

S38 -Delay Before Forced Hang Up	
ATS38=<delay>	Execution command has no effect and is included only for backward compatibility with landline modems.
ATS38?	Read command returns the current value of **S38 parameter**. Note: the format of the numbers in output is always 3 digits, left-filled with 0s

3.5.3.8. Error Control

3.5.3.8.1. *Error Control Selection - +ES*

+ES – Error Control Selection	
AT+ES[=<orig_req>,<orig_fall back>,<ans_fallback>]	Set command sets the manner of operation of the V.42 protocol in the modem. Parameters: **<orig_reg>** - Specifies the initial request mode of operation when originating a call. (Default value is 3) 0 - Direct Mode 1 - Initiate call with Buffer mode only 2 - Initiate V.42 without Detection phase. If V.8 is in use, this is a request to disable V.42 Detection Phase 3 - Initiate V.42 with Detection Phase 4 - Initiate Alternative Protocol **<orig_fallback>** - Specifies the acceptable fallback mode of operation when originating a call. (Default : 0) 0 - Error Control Optional; if error control cannot be established, use Buffered mode with flow control 1 - Errol Control Optional; if error control cannot be established, change data rate to match line <carrier> rate and use Direct mode. 2 - Error Control Required; if error control cannot be established, disconnect. 3 - Error Control (LAPM) Required if LAPM cannot be established, disconnect. 4 - Error Control (Alternate (MNP)) Required if MNP cannot be established, disconnect. **<ans_fallback>** - Specifies the acceptable fallback mode of operation when answering a call. (Default : 2) 0 - Direct Mode 1 - Error Control Disabled, use Buffered mode 2 - Error Control Optional; if error control cannot be established, use Buffered mode with flow control 3 - Errol Control Optional; if error control cannot be established, change data rate to match line <carrier> rate and use Direct mode. 4 - Error Control Required; if error control cannot be established, disconnect. 5 - Error Control (LAPM) Required if LAPM cannot be established, disconnect. 6 - Error Control (Alternate (MNP)) Required if MNP cannot be established,

+ES – Error Control Selection	
	disconnect. Note: Execution command (AT+ES<CR>) return the OK result code
AT+ES?	Read command reports current V.42 error control setting value in the format +ES: **<orig_req>,<orig_fallback>,<ans_fallback>**
AT+ES=?	Test command returns all supported values of the **<orig_req>**, **<orig_fallback>**, **<ans_fallback>** parameters.

3.5.4. 3GPP TS 27.007 AT Commands

3.5.4.1. General

3.5.4.1.1. *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

3.5.4.1.2. *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

3.5.4.1.3. *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

3.5.4.1.4. *Request Product Serial Number Identification - +CGSN*

+CGSN - Request Product Serial Number Identification	
AT+CGSN	Execution command returns the device electronic serial number (ESN) or the mobile equipment identifier (MEID) without command echo. Note: The ESN(11-digit decimal) / MEID(18-digit decimal) of modem. For more information about convert a MEID from hex to decimal please see the "MEID Conversion, HEX to DEC" in the Software User Guide.
AT+CGSN=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.5.4.1.5. *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
	  "IRA" - international reference alphabet (ITU-T T.50)
	  "UCS2" - 16-bit universal multiple-octet coded character set (ISO/IEC10646)
	   (In case supporting RUIM)

AT+CSCS?	Read command returns the current value of the active character set.
AT+CSCS=?	Test command returns the supported values for parameter **<chset>**.
Reference	3GPP TS 27.007

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

+CIMI - Request International Mobile Subscriber Identify (IMSI)	
AT+CIMI	This command returns the value of the Internal Mobile Subscriber Identity stored in the device.
AT+CIMI=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

3.5.4.1.7. *Multiplexing Mode - +CMUX*

+CMUX - Multiplexing Mode	
AT+CMUX=<mode>	Set command is used to enable/disable the 3GPP 07.10 multiplexing protocol control channel Parameters: **<mode>** multiplexer transparency mechanism 0 - basic option; it is currently the only supported value. Note: after entering the *Multiplexed Mode* an inactive timer of five seconds starts. If no CMUX control channel is established before this inactivity timer expires the engine returns to *AT Command Mode* Note: all the CMUX protocol parameter are fixed as defined in GSM07.10 and cannot be changed. Note: the maximum frame size is fixed: **N1=128**
AT+CMUX=<fwd>,<rev>	Set command is used for setting the number of forward and reverse links for data calls and to indicate whether or not default service is Rate Set 1 or Rate Set 2**.** Odd multiplex (both <fwd> and <rev> are odd numbers) indicates Rate Set 1. Even multiplex (both <fwd> and <rev> are even numbers) indicates Rate Set 2. Parameters: **<fwd>** the forward MUX option specified in hexadecimal format:1~F **<rev>** the reverse MUX option specified in hexadecimal format:1~2

	Note : The channel 1 features the all functions (voice call, data call, SMS and AT commands). The channel 2 is the all function except the data call. The channel 3 is only the DM for the debugging. Note: The +CMUX command exists the QualcommTM command table and the original function is setting the multiplex option. Reference CL93-V0327-1 F Note: If <rev> is omitted, it is assumed to have the same value as <fwd>.
AT+CMUX?	Read command returns the current value of **<fwd>** and **<rev>** parameters, in the format: **+CMUX: <fwd>,<rev>**
AT+CMUX=?	Test command returns the range of supported values for parameters **<fwd>** and **<rev>**.
Reference	3GPP 27.007, 3GPP 27.010

3.5.4.2. Call Control

3.5.4.2.1. *Hang Up Call - +CHUP*

+CHUP - Hang Up Call	
AT+CHUP	Execution command cancels all active and held calls, also if a multi-party session is running.
AT+CHUP=?	Test command returns the **OK** result code
Reference	GSM 07.07

3.5.4.2.2. *Extended Error Report - +CEER*

+CEER - Extended Error Report	
AT+CEER	Execution command returns one or more lines of information text **<report>** offering the TA user an extended error report, in the format: **+CEER: <report>** This report regards some error condition that may occur: - the failure in the last unsuccessful call setup (originating or answering) - the last call release - the last unsuccessful CDMA attach or unsuccessful PDP context activation, - the last CDMA detach or PDP context deactivation. Note: if none of this condition has occurred since power up then **"No cause information available"** condition is reported
AT+CEER=?	Test command returns **OK** result code.

+CEER - Extended Error Report	
Reference	3GPP TS 27.007

3.5.4.2.3. *Cellular Result Codes - +CRC*

+CRC - Cellular Result Codes	

AT+CRC=
[<mode>]
	Set command controls whether or not the extended format of incoming call indication is used.
	
	Parameter:
	<mode>
	  0 - disables extended format reporting (factory default)
	  1 - enables extended format reporting:
	
	When enabled, an incoming call is indicated to the **TE** with unsolicited result code
	
	+CRING: <type>
	
	Instead of the normal **RING**.
	
	where
	<type> - call type:
	  VOICE - normal voice

AT+CRC?	Read command returns current value of the parameter **<mode>**.
AT+CRC=?	Test command returns supported values of the parameter **<mode>**.
Reference	3GPP TS 27.007

3.5.4.2.4. *Voice Hang Up Control - +CVHU*

+CVHU - Voice Hang Up Control	

AT+CVHU=
[<mode>]
	Set command selects whether **ATH** or "**drop DTR**" shall cause a voice connection to be disconnected or not.
	
	Parameter:
	<mode>
	  0 - "**Drop DTR**" ignored but **OK** result code given. **ATH** disconnects.
	  1 - "**Drop DTR**" and **ATH** ignored but **OK** result code given (Verizon/Sprint /Aeris.Net models factory default value).
	  2 - "**Drop DTR**" behaviour according to **&D** setting. **ATH** disconnects (Factory default value except Verizon/Sprint/Aeris.Net models) .

AT+CVHU?	Read command reports the current value of the **<mode>** parameter, in the format: **+CVHU: <mode>**
AT+CVHU=?	Test command reports the range of supported values for parameter **<mode>**

3.5.4.3. Network Service Handling

3.5.4.3.1. *Subscriber Number - +CNUM*

+CNUM - Subscriber Number	

AT+CNUM	Execution command returns the MSISDN in the format:
	
	+CNUM: <alpha>,<number>,<type>[<CR><LF>
	+CNUM: <alpha>,<number>,<type>[…]]
	
	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

3.5.4.3.2. *Read Operator Names - +COPN*

+COPN - Read Operator Names	
AT+COPN	Returns the operator's name from the **ME** in the format: **+COPN: <numeric1>,<alpha1>[<CR><LF>** **+COPN: <numeric2>,<alpha2>[…]]** Note: In case of CDMA, the network name (operator) is not sent by network. And each CDMA carrier's list of operators is confidential and not given out. Therefore, the module only supports two result codes: **+COPN: HOME**: If the value of ERI is 1. **+COPN: ROAMING**: If the value of ERI is any other value.
AT+COPN=?	Test command returns the **OK** result code
Reference	3GPP TS 27.007

3.5.4.3.3. *Network Registration Report - +CREG*

+CREG - Network Registration Report	
AT+CREG= [<mode>]	Set command enables/disables network registration reports depending on the parameter **<mode>**.

+CREG - Network Registration Report	

	Parameter:
	<mode>
	  0 - disable network registration unsolicited result code (factory default)
	  1 - enable network registration unsolicited result code
	  2 - enable network registration unsolicited result code with network system identification data
	
	If **<mode>=1**, network registration result code reports:
	
	  +CREG: <stat>
	
	  where
	  <stat>
	    0 - not registered, ME is not currently searching a new operator to register to
	    1 - registered, home network
	    2 - reserved
	    3 - registration denied
	    4 - reserved
	    5 - registered, roaming
	
	If **<mode>=2**, network registration result code reports:
	
	  +CREG: <stat>[,<SID>]
	
	  where:
	  <SID> - System identification
	
	Note: **<SID>** is reported only if **<mode>=2** and the mobile is acquired on some network cell.

AT+CREG?	Read command reports the **<mode>** and **<stat>** parameter values in the format: **+CREG: <mode>,<stat>[,<SID>]** Note: **<SID>** is reported only if **<mode>=2** and the mobile is acquired on some network cell.
AT+CREG=?	Test command returns the range of supported **<mode>**
Reference	3GPP TS 27.007

3.5.4.3.4. *Calling Line Identification Presentation – +CLIP*

+CLIP - Calling Line Identification Presentation	
AT+CLIP=[<n>]	Set command enables/disables the presentation of the CLI (Calling Line Identity) at the **TE**. This command refers to the UMTS supplementary service CLIP (Calling Line Identification Presentation) that enables a called subscriber to get the CLI of the calling party when receiving a mobile terminated call.

+CLIP - Calling Line Identification Presentation	

	Parameters:
	\<n\>
	  0 - disables CLI indication (factory default)
	  1 - enables CLI indication
	
	If enabled the device reports after each RING the response:
	
	+CLIP: \<number\>,\<type\>,"",128,\<alpha\>,\<CLI_validity\>
	
	where:
	\<number\> - string type phone number of format specified by **\<type\>**
	\<type\> - type of address octet in integer format
	  128 - both the type of number and the numbering plan are unknown
	  129 - unknown type of number and ISDN/Telephony numbering plan
	  145 - international type of number and ISDN/Telephony numbering plan (contains the character "+")
	\<alpha\> - string type; alphanumeric representation of **\<number\>** corresponding to the entry found in phonebook; used character set should be the one selected with command Select **TE** character set +**CSCS**.
	\<CLI_validity\>
	  0 - CLI Presentation allowed.
	  1 - CLI Presentation restricted.
	  2 - CLI is not available.

AT+CLIP?	Read command returns the presentation status of the CLI in the format:
	
	+CLIP: \<n\>,\<m\>
	where:
	\<n\>
	  0 - CLI presentation disabled
	  1 - CLI presentation enabled
	\<m\> - status of the CLIP service on the UMTS network
	  2 - unknown (e.g. no network is present)
	
	Note: For compatibility with DE910, the value of \<m\> is returned

AT+CLIP=?	Test command returns the supported values of parameter **\<n\>**
Reference	3GPP TS 27.007

3.5.4.3.5. *Calling Line Identification Restriction - +CLIR*

+CLIR - Calling Line Identification Restriction	
AT+CLIR=[\<n\>]	Execution command has no effect and is included only for backward compatibility with WCDMA products. For compatibility with WCDMA products, Parameter \<n\> is available only 0, 1 and 2.

+CLIR - Calling Line Identification Restriction	
	Execution command returns the OK result code
AT+CLIR?	For compatibility with WCDMA products, Read command returns +CLIR: 0,2
AT+CLIR=?	For compatibility with WCDMA products, Test command returns +CLIR: (0-2)
Reference	3GPP TS 27.007

3.5.4.3.6. *Call Waiting – +CCWA*

+CCWA - Call Waiting	

AT+CCWA=[<n>]	Sets the presentation of an unsolicited result code of the call waiting supplementary service
	
	Parameters:
	<n> - Enables/disables the presentation of an unsolicited result code:
	   0 – disable (factory default)
	   1 – enable
	
	Note: the unsolicited result code enabled by parameter <n> is in the format:
	
	  +CCWA: <number>,<type>,"",1,<alpha>,<cli_validity>
	
	<number> - Phone number of format specified by **<type>**
	<type> - Address in Integer format
	<alpha> - Alphanumeric representation of **<number>** corresponding to the entry found in phonebook; used character set should be the one selected with +CSCS
	<CLI_validity>
	  0 - CLI valid
	  1 - CLI has been withheld by the originator
	  2 - CLI is not available due to interworking problems or limitations of originating network

AT+CCWA?	Reports the current value of the parameter **<n>**.
AT+CCWA=?	Reports the supported values for the parameter **<n>**.
Reference	3GPP TS 27.007

3.5.4.3.7. *Call Holding Service – +CHLD*

+CHLD - Call Holding Service	
AT+CHLD=<n>	Controls the network call hold service Parameters: **<n>** 2 – places all active calls (if any exist) on hold and accepts the other (waiting) call.

+CHLD - Call Holding Service	
	Note: If no call is active then only **OK** message is sent.
AT+CHLD=?	Reports the supported values for the parameter **<n>**.
Reference	3GPP TS 27.007

3.5.4.3.8. *List Current Calls - +CLCC*

+CLCC - List Current Calls	

AT+CLCC	Execution command returns the list of current calls and their characteristics in the format:
	
	[+CLCC:<id1>,<dir>,<stat>,<mode>,<mpty>,<number>,<type> ,<alpha>[<CR><LF>+CLCC:<id2>,<dir>,<stat>,<mode>, <mpty>,<number>,<type>,<alpha>[…]]]
	
	where:
	<id*n*> - call identification number
	<dir> - call direction
	  0 - mobile originated call
	  1 - mobile terminated call
	<stat> - state of the call
	  0 - active
	  1 - held
	  2 - dialing (**MO** call)
	  3 - alerting (**MO** call)
	  4 - incoming (**MT** call)
	  5 - waiting (**MT** call)
	
	  Note: 1(held), 3(alerting) and 5(waiting) are not supported for CE910 Series
	
	<mode> - call type
	  0 - voice
	  1 - data
	  9 - unknown
	<mpty> - multiparty call flag
	  0 - call is not one of multiparty (conference) call parties
	<number> - string type phone number in format specified by **<type>**
	<type> - type of phone number octet in integer format
	  129 - national numbering scheme
	  145 - international numbering scheme (contains the character "+")
	<alpha> - string type; alphanumeric representation of **<number>** corresponding to the entry found in phonebook; used character set should be the one selected with **+CSCS**.

AT+CLCC=?	Test command returns the **OK** result code
Reference	3GPP TS 27.007

3.5.4.4. Mobile Equipment Control

3.5.4.4.1. *Phone Activity Status - +CPAS*

+CPAS - Phone Activity Status	

AT+CPAS	Execution command reports the device status in the form:
	
	+CPAS: <pas>
	
	Where:
	<pas> - phone activity status
	  0 - ready (device allows commands from **TA/TE**)
	  1 - unavailable (device does not allow commands from **TA/TE**)
	  2 - unknown (device is not guaranteed to respond to instructions)
	  3 - ringing (device is ready for commands from **TA/TE**, but the ringer is active)
	  4 - call in progress (device is ready for commands from **TA/TE**, but a call is in progress)

AT+CPAS=?	Test command reports the supported range of values for **<pas>**. Note: although +**CPAS** is an execution command, 3gpp TS 27.007 requires the Test command to be defined.
Example	ATD03282131321; OK AT+CPAS +CPAS: 4 *the called phone has answered to your call* OK ATH OK
Reference	3GPP TS 27.007

3.5.4.4.2. *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. 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 disabled (factory default)
	  2 - disable TX
	  4 - disable both TX and RX
	  5 - mobile full functionality with power saving enabled
	
	<rst> - reset flag
	  0 - do not reset the ME before setting it to **<fun>** functionality level
	
	Note: URCs and network behavior (incoming calls or SMS with +CNMI activated) can wake up from CFUN=0 only (there is no way to wake up by TE - RTS not supportted).
	
	Note: issuing **AT+CFUN=4[,0]** actually causes the module to perform a network deregistration.
	
	Note: if power saving enabled, 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, set the **<fun>** parameter at value = 5 and the line **DTR** (RS232) must be set 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, the **DTR** must be enabled and it must be waited for the **CTS** (RS232) line to go in **ON** status.
	Until 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
	
	Note: If AT+CFUN=2[,0] mode is activating, Current call is disconnected.

+CFUN - Set Phone Functionality	
	Note: If AT+CFUN=2[,0] mode is activating, Current activated PDP Context and socket connection is disconnected.
eAT+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

3.5.4.4.3. *Signal Quality - +CSQ*

+CSQ - Signal Quality	

AT+CSQ	Execution command reports received signal quality indicators in the form:
	
	 +CSQ: <rssi>,<fer>
	 where
	 <rssi> - received signal strength indication
	  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
	 <fer> - frame error rate (in percent)
	  0 - less than 0.01%
	  1 - 0.01% to 0.1%
	  2 - 0.1% to 0.5%
	  3 - 0.5% to 1.0%
	  4 - 1.0% to 2.0%
	  5 - 2.0% to 4.0%
	  6 - 4.0% to 8.0%
	  7 - more than 8.0%
	  99 - not known or not detectable
	
	 Note: AT+CSQ? Command is only allowed for Sprint and Aeris.

AT+CSQ=?	Test command returns the supported range of values of the parameters **<rssi>** and **<fer>**.
Reference	3GPP TS 27.007

3.5.4.4.4. *Select Phonebook Memory Storage - +CPBS*

+CPBS - Select Phonebook Memory Storage	
AT+CPBS= <storage> [,<password>]	Set command selects phonebook memory storage **<storage>**, which will be used by other phonebook commands. Parameter: **<storage>** "ME" - EFS phonebook(Factory default)

+CPBS - Select Phonebook Memory Storage	
	"SM" - RUIM phonebook(Facroty default for RUIM)(RUIM only) "FD" - RUIM fixed dialing-phonebook (FDN)(RUIM only) "LD" - RUIM last dialing phonebook (RUIM only) "MC" - device missed (unanswered received) calls list (+CPBF is not applicable for this storage) "RC" - ME received calls list (+CPBF is not applicable for this storage) "DC" - MT dialled calls list (+CPBF is not applicable for this storage) "EN" - RUIM (or MT) emergency number (+CPBW is not be applicable for this st orage) <password>: string type value representing the PIN2-code required when selecting PIN2-code locked <storage> above "FD" Note: If **"SM"** is the currently selected phonebook, selecting**"FD"** phonebook with "**AT+CPBS="FD""**' command simply selects the FDN as the phonebook upon which all subsequent **+CPBW, +CPBF** and **+CPBR** commands act; the command does not deactivate "SM" phonebook, and does not activate FDN Note: if **<password>** parameter is given, PIN2 will be verified, even if it is not required, i.e. it has already been inserted and verified during current session
AT+CPBS?	Read command returns the actual values of the parameter **<storage>**, the number of occupied records **<used>** and the maximum index number **<total>**, in the format: **+CPBS: <storage>,<used>,<total>** Note:If **<storage>** is "**ME**", then an initial value of **<used>** is 1 because module's own phone number always occupies index 1 of records. Note: For **<storage>**="**MC**": if there are more than one missed calls from the same number the read command will return only the last call.
AT+CPBS=?	Test command returns the supported range of values for the parameters **<storage>**.
Example	AT+CPBS="ME"　　　　*current phonebook storage is NV* AT+CPBR=1 +CPBR: 1,"0105872928",129,"James","example@telit.com" OK
Reference	3GPP TS 27.007

3.5.4.4.5. *Read Phonebook Entries – +CPBR*

+CPBR - Read Phonebook Entries	
AT+CPBR= **<index1>** **[,<index2>]**	Execution command returns phonebook entries in location number range **<index1>**..**<index2>** from the current phonebook memory storage selected with **+CPBS**. If **<index2>** is omitted, only location **<index1>** is returned. Parameters:

+CPBR - Read Phonebook Entries	
	<index1> - integer type, value in the range of location numbers of the currently selected phonebook memory storage (see **+CPBS**).
	<index2> - integer type, value in the range of location numbers of the currently selected phonebook memory storage (see **+CPBS**).
	If the storage is "ME" then the response format is:
	[+CPBR: <index1>,<number>,<type>,<text>,<e_text> [<CR><LF> **+CPBR: <index2>,<number>,<type>,<text>,<e_text> […]]]**
	If the storage is "DC" and "RC" then the response format is:
	[+CPBR: <index1>,<number>,<type>,<text>,<time>,<duration>[<CR><LF> **+CPBR: <index2>,<number>,<type>,<text>,<time>,<duration>[…]]]**
	If the storage is "MC" then the response format is:
	[+CPBR: <index1>,<number>,<type>,<text>,<time>[<CR><LF> **+CPBR: <index2>,<number>,<type>,<text>,<time>[…]]]**
	where: **<index_n>** - the location number of the phonebook entry **<number>** - string type phone number of format **<type>** **<type>** - type of phone number octet in integer format 129 - national numbering scheme 145 - international numbering scheme (contains the character "+") **<text>** - the alphanumeric text associated to the number; used character set should be the one selected with command **+CSCS.** **<e_text >** - Email alphanumeric text; used character set should be the one selected with command **+CSCS** **<time>** - Date and time in clock seconds **<duration>** - Duration of the call
	Note: if "MC" is the currently selected phonebook memory storage, a sequence of missed calls coming from the same number will be saved as one missed call and **+CPBR** will show just one line of information.
	Note: If all queried locations are empty (but available), no information text lines will be returned, while if listing fails in an **ME** error, **+CME ERROR: <err>** is returned.
AT+CPBR=?	Test command returns the supported range of values for parameters **<index_n>** and the maximum lengths of **<number>** and **<text>** fields, in the format:
	+CPBR: (<minIndex> - <maxIndex>),<nlength>,<tlength>
	where: **<minIndex>** - the minimum **<index>** number, integer type **<maxIndex>** - the maximum **<index>** number, integer type

+CPBR - Read Phonebook Entries	
	<nlength> - maximum **<number>** field length, integer type **<tlength>** - maximum **<name>** field length, integer type
Note	Remember to select the PB storage with +**CPBS** command before issuing PB commands.
Example	AT+CPBS="ME" OK AT+CPBS? +CPBS: "ME",1,50 OK AT+CPBR=? +CPBR: (1-50),40,20 OK AT+CPBR=1 +CPBR: 1,"01048771234",129,"James","example@telit.com" OK
Reference	3GPP TS 27.007

3.5.4.4.6. *Find Phonebook Entries – +CPBF*

+CPBF - Find Phonebook Entries	

**AT+CPBF=
<findtext>**
	Execution command returns phonebook entries (from the current phonebook memory storage selected with +**CPBS**) which alphanumeric field start with string **<findtext>**.
	
	Parameter:
	<findtext> - string type; used character set should be the one selected with command +**CSCS.**
	
	The command returns a report in the form:
	
	[+CPBF: <index1>,<number>,<type>,<text>,<e_text> [<CR><LF>
	+CPBF: <index2>,<number>,<type>,<text>,<e_text> […]]]
	
	where:
	<index*n*> - the location number of the phonebook entry
	<number> - string type phone number of format **<type>**
	<type> - type of phone number octet in integer format
	  129 - national numbering scheme
	  145 - international numbering scheme (contains the character "+")
	<text> - the alphanumeric text associated to the number; used character set should be the one selected with command +**CSCS.**
	<e_text > - Email alphanumeric text; used character set should be the one selected with command +**CSCS**

+CPBF - Find Phonebook Entries	
	Note: +CPBF is not applicable if the current selected storage (see +CPBS) is either "MC", either "RC" or "DC". Note: if **<findtext>=""** the command returns all the phonebook records. Note: if no PB records satisfy the search criteria then an **ERROR** message is reported. Note: Remember to select the PB storage with +CPBS command before issuing PB commands.
AT+CPBF=?	Test command reports the maximum lengths of **<number>** and **<text>** fields, in the format: **+CPBF: [<nlength>],[<tlength>]** where: **<nlength>** - maximum length of field **<number>**, integer type **<tlength>** - maximum length of field **<text>**, integer type
Note	Remember to select the PB storage with **+CPBS** command before issuing PB commands.
Example	AT+CPBS="ME" *Selecting phonebook* OK AT+CPBF="J" *Searching for string "J"* +CPBF: 1,"01048771234",129,"James","example@telit.com" +CPBF: 2,"0169998888",129,"Jane","" OK *Searching for everything in phone book, and finding all entries* AT+CPBF="" +CPBF: 1,"01048771234",129,"James","example@telit.com" +CPBF: 2,"0169998888",129,"Jane","" +CPBF: 7,"0115556666",129,"Juliet","" +CPBF: 5,"0181111234",129,"Kevin","" OK
Reference	3GPP TS 27.007

3.5.4.4.7. *Write Phonebook Entry - +CPBW*

+CPBW - Write Phonebook Entry	

AT+CPBW= [<index>] [,<number> [,<type> [,<text> [,<e_text>]]]]	Execution command writes phonebook entry in location number **<index>** in the current phonebook memory storage selected with **+CPBS**.
	
	Parameters:
	<index> - integer type, value in the range of location numbers of the currently selected phonebook memory storage (see **+CPBS**).
	<number> - string type, phone number in the format **<type>**
	<type> - the type of number
	  129 - national numbering scheme
	  145 - international numbering scheme (contains the character "+")
	<text> - the text associated to the number, string type; used character set should be the one selected with command **+CSCS**.
	<e_text > - Email alphanumeric text; used character set should be the one selected with command **+CSCS**
	
	Note: If record number **<index>** already exists, it will be overwritten.
	
	Note: if either **<number>**, **<type>** , **<text>** and **<e_text>** are omitted, the phonebook entry in location **<index>** is deleted.
	
	Note: In CE910-DUAL Sprint & Aeris case, the index 1 has been always occupied by own phone number. So to change index 1 you have to change the own phone number. If AT+CPBW=1 is executed, the module will return "ERROR" result code.
	
	Note: if **<index>** is omitted or **<index>**=0, the number **<number>** is stored in the first free phonebook location.
	
	Note: if either "DC", "MC" or "RC" memory storage has been selected (see **+CPBS**) it is possible just to delete the phonebook entry in location **<index>**, therefore parameters **<number>**, **<type>** and **<text>** must be omitted.
	
	Note: Remember to select the PB storage with **+CPBS** command before issuing PB commands.

AT+CPBW=?	Test command returns location range supported by the current storage as a compound value, the maximum length of **<number>** field, supported number format of the storage and maximum length of **<text>** field. The format is: **+CPBW: (list of supported <index>s),<nlength>,** **(list of supported <type>s),<tlength>,**`<elength>` where: **<nlength>** - integer type value indicating the maximum length of field **<number>**. **<tlength>** - integer type value indicating the maximum length of field **<text>**

+CPBW - Write Phonebook Entry	
	<elength>: integer type value indicating the maximum length of field <email>
Reference	3GPP TS 27.007
Example	AT+CPBS="ME" OK AT+CPBW=? +CPBW: (1-50),40,(129,145),20,20 OK AT+CPBW=6,"18651896699",129,"John","ex@telit.com" OK
Note	Remember to select the PB storage with **+CPBS** command before issuing PB commands.

3.5.4.4.8. *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), range is (00..23)
	  mm - minute (two last digits are mandatory), range is (00..59)
	  ss - second (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 -47..+48

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

3.5.4.4.9. *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.
	
	When the RTC time reaches the alarm time then the alarm starts, the behaviour 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.
	  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 pin GPIO6 high, provided its **\<direction>** has been set to alarm output, 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.

+CALA - Alarm Management	
	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>]**
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

3.5.4.4.10. *Alert Sound Mode - +CALM*

+CALM - Alert Sound Mode	
AT+CALM=	Set command is used to select the general alert sound mode of the device.

+CALM - Alert Sound Mode	

<mode>	Parameter:
	<mode>
	  0 - normal mode (factory default)
	  1 - silent mode; no sound will be generated by the device, except for alarm sound
	  2 - stealth mode; no sound will be generated by the device
	
	Note: if silent mode is selected then incoming calls will not produce alerting sounds but only the unsolicited messages **RING** or **+CRING**.

AT+CALM?	Read command returns the current value of parameter **<mode>**.
AT+CALM=?	Test command returns the supported values for the parameter **<mode>** as compound value. **+CALM: (0-2)**
Reference	3GPP TS 27.007

3.5.4.4.11. *Ringer Sound Level - +CRSL*

+CRSL - Ringer Sound Level	

AT+CRSL=<level>	Set command is used to select the incoming call ringer sound level of the device.
	
	Parameter:
	<level> - ringer sound level
	  0 - Off
	  1 - low
	  2 - middle
	  3 - high (factory default)
	  4 - progressive

AT+CRSL?	Read command reports the current **<level>** setting of the call ringer in the format: **+CRSL: <level>**
AT+CRSL=?	Test command reports **<level>** supported values as compound value. **+CRSL: (0-4)**
Reference	3GPP TS 27.007

3.5.4.4.12. *Loudspeaker Volume Level - +CLVL*

+CLVL - Loudspeaker Volume Level	

AT+CLVL=<level>	Set command is used to select the volume of the internal loudspeaker audio output of the device.
	
	Parameter:
	<level> - loudspeaker volume
	  0..*max* - the value of *max* can be read by issuing the Test command **AT+CLVL=?**
	  10 - factory default

AT+CLVL?	Read command reports the current **<level>** setting of the loudspeaker volume in the format: **+CLVL: <level>**
AT+CLVL=?	Test command reports **<level>** supported values range in the format: **+CLVL: (0-*max*)**
Reference	3GPP TS 27.007

3.5.4.4.13.   *Microphone Mute Control - +CMUT*

+CMUT - Microphone Mute Control	

AT+CMUT=**<n>**	Set command enables/disables the muting of the microphone audio line during a voice call.
	
	Parameter:
	<n>
	  0 - mute off, microphone active (factory default)
	  1 - mute on, microphone muted.
	
	Note: this command mutes/activates both microphone audio paths, internal mic and external mic.

AT+CMUT?	Read command reports whether the muting of the microphone audio line during a voice call is enabled or not, in the format: **+CMUT: <n>**
AT+CMUT=?	Test command reports the supported values for **<n>** parameter.
Reference	3GPP TS 27.007

3.5.4.4.14.   *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

3.5.4.4.15.   *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

3.5.4.5. Mobile Equipment Errors

3.5.4.5.1. *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>** final result code instead of the default **ERROR** final result code. **ERROR** is anyway returned normally when the error message is related to syntax, invalid parameters, or **DTE** functionality.
	
	 Parameter:
	 <n> - enable flag
	  0 - disable +**CME ERROR:<err>** reports, use only **ERROR** report.(factory default)
	  1 - enable +**CME ERROR:<err>** reports, with **<err>** in numeric format
	  2 - enable +**CME ERROR: <err>** reports, with **<err>** in verbose format

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

3.5.4.6. Voice Control

3.5.4.6.1. *DTMF Tones Transmission - +VTS*

+VTS - DTMF Tones Transmission	
AT+VTS= <dtmfstring> [,duration]	Execution command allows the transmission of DTMF tones. Parameters: **<dtmfstring>** - String of **<dtmf>s,** i.e. ASCII characters in the set **(0-9), #,*** the string can be a **<dtmf>s** long; it allows the user to send a sequence of DTMF tones, each of them with a duration that was defined through +**VTD** command. **<duration>** - Can be specified only if the length of first parameter is just one ASCII character 0..5 - a single DTMF tone will be transmitted for a duration depending on the network, no matter what the current +**VTD** setting is. Note: this commands operates in voice mode only (see +**FCLASS**). Note: **<dtmfstring>**should be inputed without the double quotation mark(""").
AT+VTS=?	Test command provides the list of supported **<dtmf>s** and the list of supported **<duration>s** in the format: **(list of supported <dtmf>s)[,(list of supported <duration>s)]**
Reference	3GPP TS 27.007 and TIA IS-101

3.5.4.6.2. *Tone Duration - +VTD*

+VTD - Tone Duration	

AT+VTD= <duration>	Set command sets the length of tones transmitted with +**VTS** command.
	
	Parameter:
	<duration> - duration of a tone
	
	  0 – 95ms (factory default)
	  1 – 150ms
	  2 – 200ms
	  3 – 250ms
	  4 – 300ms
	  5 – 350ms

AT+VTD?	Read command reports the current Tone Duration, in the format: **<duration>**
AT+VTD=?	Test command provides the list of supported **<duration>s** in the format: **(list of supported <duration>s)**
Reference	3GPP TS 27.007 and TIA IS-101

3.5.4.7. Commands For Battery Charger

3.5.4.7.1. *Battery Charge - +CBC*

+CBC - Battery Charge	

AT+CBC	Execution command returns the current Battery Charge status in the format:
	
	+CBC: <bcs>,<bcl>
	
	where:
	<bcs> - battery status
	  0 - **ME** is powered by the battery
	  1 - **ME** has a battery connected, and charger pin is being powered
	  2 - **ME** does not have a battery connected
	  3 - Recognized power fault, calls inhibited
	<bcl> - battery charge level
	  0 - battery is exhausted, or **ME** does not have a battery connected
	  25 - battery charge remained is estimated to be 25%
	  50 - battery charge remained is estimated to be 50%
	  75 - battery charge remained is estimated to be 75%
	  100 - battery is fully charged.
	
	Note: There is not charger pin. So, **<bcs>=1** will never appear.
	
	Note: without battery/power connected on **VBATT** pins or during a power fault the unit is not working, therefore values **<bcs>=2** and **<bcs>=3** will never appear.

AT+CBC=?	Test command returns parameter values supported as a compound value. **+CBC: (0-3),(0-100)** Note: although **+CBC** is an execution command, 3gpp TS 27.007 requires the Test command to be defined.
Example	AT+CBC +CBC: 0,75 OK
Note	The **ME** does not make differences between being powered by a battery or by a power supply on the **VBATT** pins, so it is not possible to distinguish between these two cases.
Reference	3GPP TS 27.007

3.5.5. Partially 3GPP TS 27.005 AT Commands for SMS and CBS

3.5.5.1. General Configuration

3.5.5.1.1. *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:
	
	For compatibility with WCDMA products, Parameter <service> is available only 2.
	
	Parameter:
	<service>
	0 - The syntax of SMS AT commands is compatible with 3GPP TS 27.005 Phase 2 version 4.7.0
	1 - The syntax of SMS AT commands is compatible with 3GPP TS 27.005 Phase 2+ version.
	2 - The syntax of SMS AT commands is compatible partially with 3GPP TS 27.005 Phase 2 version 4.7.0. (reflected partially IS-637A, B in CDMA network) (factory default)
	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>**
AT+CSMS=?	Test command reports the supported value of the parameter <service>.
Example	AT+CSMS=? +CSMS: (2) OK AT+CSMS=2

+CSMS - Select Message Service	
	+CSMS: 1,1,0 OK AT+CSMS? +CSMS: 2,1,1,0 OK

3.5.5.1.2. *Preferred Message Storage - +CPMS*

+CPMS - Preferred Message Storage	

AT+CPMS=
<memr>[,<memw>]
	Set command selects memory storages **<memr>**, **<memw>** to be used for reading, writing, sending and storing SMs.
	
	Parameters:
	<memr> - memory from which messages are read and deleted
	  "ME" – SMS memory storage into module (default)
	  "SM" – SIM SMS memory storage (RUIM only, default in case supporting RUIM)
	
	<memw> - memory to which writing and sending operations are made
	  "ME" – SMS memory storage into module (default)
	  "SM" – SIM SMS memory storage (RUIM only, default in case supporting RUIM)
	
	The command returns the memory storage status in the format:
	
	+CPMS: <usedr>,<totalr>,<usedw>,<totalw>
	
	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

AT+CPMS?	Read command reports the message storage status in the format: **+CPMS: <memr>,<usedr>,<totalr>,<memw>,<usedw>,<totalw>** where **<memr>**, **<memw>** are the selected storage memories for reading, writing and storing respectively.
AT+CPMS=?	Test command reports the supported values for parameters **<memr>**, **<memw>**
Example	AT+CPMS=? +CPMS: ("ME"),("ME")

+CPMS - Preferred Message Storage	
	OK at+cpms? +CPMS: "ME",5,99,"ME",5,99 OK AT+CPMS="ME","ME" +CPMS: 5,99,5,99 OK AT+CPMS? +CPMS: "ME",5,99,"ME",5,99 OK In case supporting RUIM, AT+CPMS=? +CPMS: ("ME","SM"),("ME","SM") OK AT+CPMS? +CPMS: "SM",2,10,"ME",15,99 OK AT+CPMS="SM","SM" +CPMS: 2,10,2,10 OK AT+CPMS? +CPMS: "SM",2,10,"SM",2,10 OK

3.5.5.1.3. *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 (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

3.5.5.2. Message Configuration

3.5.5.2.1. *Set Text Mode Parameters - +CSMP*

+CSMP - Set Text Mode Parameters	

AT+CSMP= [<callback_addr> [,<tele_id > [,<priority> [,<enc_type >]]]]	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:
	 <callback_addr> - Callback address.
	   Note: The maximum length is different with every carrier.
	   In case of Sprint ,Aeris.Net and US Cellular:
	     Maximum length is 32 characters
	   In case of Verizon:
	     Maximum length is 20 characters
	
	 Note: Initially, this parameter is null. Some carrier networks discard SMS's without a callback number. So we recommend that customer setup callback number using AT+CSMP command.
	 Note : The **<callback_addr>** isn't used and saved for only Aeris.Net
	
	 <tele_id> - Teleservice ID
	   4097 - page
	   4098 - SMS message (factory default)
	
	 <priority> - Priority
	   Note: The priority is different with every carrier.
	   In case of Sprint and Aeris.Net:
	     0 - Normal (factory default)
	     1 - Interactive
	     2 - Urgent
	     3 - Emergency
	   In case of Verizon:
	     0 - Normal (factory default)
	     1 - High
	
	 <enc_type> - data coding scheme:
	   0 - 8-bit Octet (Aeris.Net factory default)
	   2 - 7-bit ASCII (Verizon/Sprint factory default)
	   4 - 16-bit Unicode (Sprint/Aeris.Net does not support)
	
	 Note: the current settings are stored through **+CSAS**

AT+CSMP?	Read command reports the current setting in the format: **+CSMP: < callback_addr >,<tele_id >,< priority >,< enc_type >**
AT+CSMP=?	Test command returns the **OK** result code.

+CSMP - Set Text Mode Parameters	
Example	AT+CSMP=? OK AT+CSMP? +CSMP: "",4098,0,0 OK AT+CSMP="1234567890",4097,1,2 OK AT+CSMP? +CSMP: "1234567890",4097,1,2 OK

3.5.5.2.2. *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 (**<tooa>, <tele_id>, <priority>, <enc_type>,
	    <length>**) in **+CMT, +CMGL, +CMGR** result codes for **SMS-DELIVERs**
	    and **SMS-SUBMITs** in text mode. (factory default)
	  1 - show the values in result codes

AT+CSDH?	Read command reports the current setting in the format: **+CSDH: <show>**
AT+CSDH=?	Test command reports the supported range of values for parameter **<show>**
Example	AT+CSDH=1 OK AT+CMGL="ALL" +CMGL: 0,"STO UNSENT","","0114933460",,,,4097,0,0,12 Test message +CMGL: 1,"STO SENT","01085718504","0114933460",,129,4097,0,0,4 test +CMGL: 2,"REC READ","0114933460","0114933460",20140708103914,129,4098,0,2,12 test message +CMGL: 3,"REC READ","0114933460","0114933460",20140708103932,129,4098,0,2,4 test +CMGL: 4,"STO UNSENT","0114933460","0114933460",,129,4098,0,2,4 test +CMGL: 5,"REC READ","0114933460","0114933460",20140708104012,129,4098,0,0,8

+CSDH - Show Text Mode Parameters	
	test SMS OK AT+CSDH? +CSDH: 1 OK AT+CSDH=0 OK AT+CMGL="ALL" +CMGL: 0,"STO UNSENT","","0114933460", Test message +CMGL: 1,"STO SENT","01085718504","0114933460", test +CMGL: 2,"REC READ","0114933460","0114933460",20140708103914 test message +CMGL: 3,"REC READ","0114933460","0114933460",20140708103932 test +CMGL: 4,"STO UNSENT","0114933460","0114933460", test +CMGL: 5,"REC READ","0114933460","0114933460",20140708104012 test SMS OK

3.5.5.2.3. *Save Settings - +CSAS*

+CSAS - Save Settings	
AT+CSAS [=<profile>]	Execution command saves settings made by, +**CSMP** command in local non volatile memory Parameter: **<profile>** 0,1 - it saves the settings to NVM. Note: If parameter is omitted the settings are saved to profile 0 in the non volatile memory.
AT+CSAS=?	Test command returns the possible range of values for the parameter **<profile>**.
Example	AT+CSAS=? +CSAS: (0,1) OK AT+CSAS OK AT+CSAS=1 OK AT+CSAS=0

+CSAS - Save Settings	
	OK

3.5.5.2.4. *Restore Settings - +CRES*

+CRES - Restore Settings	
AT+CRES [=<profile>]	Execution command restores message service settings saved by **+CSAS** command from NVM. Parameter: **<profile>** 0,1 - it restores message service settings from NVM. Note: If parameter is omitted the command restores message service settings from Profile 0 in the non volatile memory.
AT+CRES=?	Test command returns the possible range of values for the parameter **<profile>**.
Example	AT+CRES=? +CRES: (0,1) OK AT+CRES=0 OK AT+CRES=1 OK

3.5.5.3. Message Receiving And Reading

3.5.5.3.1. *New Message Indications To Terminal Equipment - +CNMI*

+CNMI - New Message Indications To Terminal Equipment	

AT+CNMI=[<mt>]	Set command selects the behaviour of the device on how the receiving of new messages from the network is indicated to the **DTE**.
	
	 Parameter:
	 <mt> - The information written in italics will be present depending on **+CSDH** last setting.
	
	 Unsolicited result codes buffering option
	   0 - No Indication (factory default)
	
	   1 - Indicate like below
	     +CMTI: <memr>,<index>
	       <memr> - memory storage where the new message is stored
	         "ME"
	       <index> - location on the memory where SMS is stored.

+CNMI - New Message Indications To Terminal Equipment	

	2 - Indicate like below
	(**PDU Mode**)
	  +CMT: ,<length><CR><LF><pdu>
	    <length> - PDU length
	    <pdu> - PDU Message
	
	<pdu>: <orig_num><date><tele_id><priority><enc_type><length><data>
	
	where:
	
	<orig_num> : <addr_len><tooa><address>
	
	    <addr_len> : Octets length of address field(1 Octet : <tooa> and <address>).
	
	    <tooa> : Type of address(1 Octet).
	
	    <addres> : Address digits with representation of semi-octets.
	
	<date> : Serivce center time stamp (6 Octets : YYMMDDHHMMSS).
	
	<tele_id> : Teleservice ID (2 Octets).
	
	<priority> : Priority(1 Octet).
	
	<enc_type> : Encoding type(1 Octet).
	
	<length> : Octets length of user data(1 Octet).
	
	<data> : User data of message.
	
	(**TEXT Mode**)
	  +CMT:
	  <orig_num>,<callback>,<date>[,<tooa>,<tele_id>,<priority>,<enc_type>,<length>]<CR><LF><data>
	    <orig_num> - Origination number.
	    <callback> - Callback number.
	    <date> - Received date in form as "YYYYMMDDHHMMSS".
	    <tooa> - Type of <orig_num>.
	    <tele_id> - Teleservice ID.
	      4097 - page
	      4098 - SMS message
	      4099 - voice mail notification
	      262144 - voice mail notification
	    <priority> - Priority.
	      Note: The priority is different with every carrier.
	      In case of Sprint and Aeris.Net:
	        0 - Normal (factory default)
	        1 - Interactive
	        2 - Urgent
	        3 - Emergency
	      In case of Verizon:
	        0 - Normal (factory default)
	        1 - High
	    <enc_type> - Encoding type of message.

+CNMI - New Message Indications To Terminal Equipment	
	0 - 8-bit Octet 2 - 7-bit ASCII 4 - 16-bit Unicode **\<length\>** - Length of message. **\<data\>** - Message data. (Indicates the new voice mail count, if \<tele_id\> is voice mail notification) Note : Regardless of \<mt\>, a message is saved in SMS memory storage.
AT+CNMI?	Read command returns the current parameter settings for **+CNMI** command in the form: **+CNMI: \<mt\>**
AT+CNMI=?	Test command reports the supported range of values for the **+CNMI** command parameters.
Example	AT+CNMI=? +CNMI: (0-2) OK AT+CNMI=1 OK AT+CNMI? +CNMI: 1 OK +CMTI: "ME",98 AT+CNMI=2 OK AT+CNMI? +CNMI: 2 OK +CMT: "0114933460","0114933460",20140109182224,129,4098,0,0,8 TEST SMS#SMSFULL

3.5.5.3.2. *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:

+CMGL - List Messages	

	\<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>,"",\<length>\<CR>\<LF>\<pdu>
	
	 Case of received message from base station :
	   \<PDU>: \<orig_num>\<date>\<tele_id>\<priority>\<enc_type>\<length>\<data>
	 Case of sending message to base station:
	   \<PDU>: \<da>\<callback>\<tele_id>\<priority>\<enc_type>\<length>\<data>
	   where:
	   \<orig_num> : \<addr_len>\<tooa>\<address>
	      \<addr_len> : Octets length of address field(1 Octet : \<tooa> and \<address>).
	      \<tooa> : Type of address(1 Octet).
	      \<addres> : Address digits with representation of semi-octets.
	   \<da> : \<addr_len>\<toda>\<address>
	      \<addr_len> : Octets length of address field(1 Octet : \<toda> and \<address>).
	      \<toda> : Type of address(1 Octet).
	      \<addres> : Address digits with representation of semi-octets.
	   \<callback> : \<addr_len>\<toca>\<address>
	      \<addr_len> : Octets length of address field(1 Octet : \<toca> and \<address>).
	      \<toca> : Type of address(1 Octet).
	      \<addres> : Address digits with representation of semi-octets.
	   \<date> : Serivce center time stamp (6 Octets : YYMMDDHHMMSS).
	   \<tele_id> : Teleservice ID (2 Octets).
	   \<priority> : Priority(1 Octet).
	   \<enc_type> : Encoding type(1 Octet).
	   \<length> : Octets length of user data(1 Octet).
	   \<data> : User data of message.
	 where:
	 \<index> - message position in the memory storage list.
	 \<stat> - status of the message
	 \<length> - length of the PDU in bytes

+CMGL - List Messages	
	<pdu> - message in PDU format
	(Text Mode)
	Parameter:
	<stat>
	"REC UNREAD" - new message
	"REC READ" - read message
	"STO UNSENT" - stored message not yet sent
	"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):
	If there is at least a **Received** message to be listed the representation format is:
	+CMGL:
	<index>,<stat>,<orig_num>,<callback>,<date>*[,<tooa>,<tele_id>,<priority>,<enc_type>,<length>]*<CR><LF> <data>
	If there is at least a **Sent** or an **Unsent** message to be listed the representation format is:
	+CMGL:
	<index>,<stat>,<da>,<callback>,*[,<toda>,<tele_id>,<priority>,<enc_type>,<length>]*<CR><LF><data>
	Where
	<orig_num> - Origination number.
	<da> - Destination number.
	<callback> - Callback number.
	<date> - Received date in form as "YYYYMMDDHHMMSS".
	<tooa> - Type of <orig_num>.
	<toda> - Type of <da>.
	<tele_id> - Teleservice ID.
	4097 - page
	4098 - SMS message
	4099 - voice mail notification
	262144 - voice mail notification
	<priority> - Priority.
	Note: The priority is different with every carrier.
	In case of Sprint and Aeris.Net:
	0 - Normal (factory default)
	1 - Interactive
	2 - Urgent
	3 - Emergency
	In case of Verizon:
	0 - Normal (factory default)
	1 - High

+CMGL - List Messages	

	<enc_type> - Encoding type of message.
	   0 - 8-bit Octet
	   2 - 7-bit ASCII
	   4 - 16-bit Unicode
	<length> - Length of message.
	<data> - Message data. (Indicates the new voice mail count, if <tele_id> is voice mail notification)
	
	Note: If a message is present when +CMGL="ALL" is used it will be changed status from **REC UNREAD** to **REC READ**.

AT+CMGL=?	Test command returns a list of supported **<stat>**s
Example	<PDU Mode> Case of received message from base station: AT+CMGL=1 +CMGL: 29,1,"",52 07812811495346350808040947271002020221C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C20 OK 07 <addr_len: 7byte> 81 <type_addr: 129> 281149534635 <Origination number: 821194356453> 080804094727 <Date: 08/08/04,09:47:27> 1002 <Teleservice_id: 4098(decimal)> 02 <priority: urgent > 02 <encoding_type: 7-bit ASCII > 21 <data_len: 33> C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C20 <user_data: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa> Else: AT+CMGL=2 +CMGL: 31,2,"",23 07811091346554F307811091346554F3100200000a61616161616161616161 OK 07 <addr_len: 7byte> 81 <type_addr:129> 1091346554F3 <Destination_addr: 01194356453> 07 <addr_len: 7byte> 81 <type_addr:129> 1096224658F1 <Callback_Number: 01692264851> 1002 <Teleservice_id: 4098(decimal)> 00 <priority: normal > 00 <encoding_type: 8-bit Octet >

+CMGL - List Messages	
	0A <data_len: 10> 61616161616161616161 <data: aaaaaaaaaa> <PDU Mode> AT+CMGF=0 OK AT+CMGF? +CMGF: 0 OK AT+CMGL=? (0-4) OK AT+CMGL=4 +CMGL: 0,2,"",16 068119499397771002000000A5465737420534D532031 +CMGL: 1,2,"",17 0681194993977700100200000B5465737420534D5320320D +CMGL: 2,2,"",16 068119499397770010020000 0A5465737420534D532033 +CMGL: 3,2,"",16 068119499397670010020000 0A5465737420534D532034 +CMGL: 4,2,"",16 068119499397670010020000 0A5465737420534D532035 OK <Text Mode> AT+CMGF=1 OK AT+CMGF? +CMGF: 1 OK AT+CMGL=? ("REC UNREAD","REC READ","STO UNSENT","STO SENT","ALL") OK AT+CMGL="ALL" +CMGL: 0,"STO UNSENT","9194397977","", +CMGL: 1,"STO UNSENT","9194397977","", +CMGL: 2,"STO UNSENT","9194397977","", +CMGL: 3,"STO UNSENT","9194397976","", +CMGL: 4,"STO UNSENT","9194397976","", OK

3.5.5.3.3. *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)
	If there is at least one message to be listed the representation format is: **+CMGR:<stat>,"",<length><CR><LF><PDU>**
	Case of received message from base station : **<PDU>:** **<orig_num><date><tele_id><priority><enc_type><length><data>**
	Case of sending message to base station: **<PDU>:** **<da><callback><tele_id><priority><enc_type><length><data>** where:
	<orig_num> : <addr_len><tooa><address>
	<addr_len> : Octets length of address field(1 Octet : <tooa> and <address>).
	<tooa> : Type of address(1 Octet).
	<addres> : Address digits with representation of semi-octets.
	<da> : <addr_len><toda><address>
	<addr_len> : Octets length of address field(1 Octet : <toda> and <address>).
	<toda> : Type of address(1 Octet).
	<addres> : Address digits with representation of semi-octets.
	<callback> : <addr_len><toca><address>
	<addr_len> : Octets length of address field(1 Octet : <toca> and <address>).
	<toca> : Type of address(1 Octet).
	<addres> : Address digits with representation of semi-octets.
	<date> : Serivce center time stamp (6 Octets : YYMMDDHHMMSS).
	<tele_id> : Teleservice ID (2 Octets).
	<priority> : Priority(1 Octet).

+CMGR - Read Message	
	<enc_type> : Encoding type(1 Octet).
	<length> : Octets length of user data(1 Octet).
	<data> : User data of message.
	where
	<stat> - status of the message
	0 - new message
	1 - read message
	2 - stored message not yet sent
	3 - stored message already sent
	<length> - length of the PDU in bytes.
	<pdu> - message in PDU format
	(Text Mode)
	Output format for received messages (the information written in *italics* will be present depending on **+CSDH** last setting)*:*
	Output format for message delivery confirm:
	+CMGR:
	<stat>,<orig_num>,<callback>,<date>*[,<tooa>,<tele_id>,<priority>,<enc_type>,<length>]*<CR><LF><data>
	If there is either a **Sent** or an **Unsent** message in location <index> the output format is:
	+CMGR:
	<stat>,<da>,<callback>,*[,<toda>,<tele_id>,<priority>,<enc_type>,<length>]*<CR><LF><data>
	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
	<orig_num> - Origination number.
	<da> - Destination number.
	<callback> - Callback number.
	<date> - Received date in form as "YYYYMMDDHHMMSS".
	<tooa> - Type of <**orig_num**>.
	<toda> - Type of <**da**>.
	<tele_id> - Teleservice ID.
	4097 - page
	4098 - SMS message
	4099 - voice mail notification
	262144 - voice mail notification

+CMGR - Read Message	

	\<priority> - Priority.
	   Note: The priority is different with every carrier.
	   In case of Sprint and Aeris.Net:
	     0 - Normal (factory default)
	     1 - Interactive
	     2 - Urgent
	     3 - Emergency
	   In case of Verizon:
	     0 - Normal (factory default)
	     1 - High
	\<enc_type> - Encoding type of message.
	   0 - 8-bit Octet
	   2 - 7-bit ASCII
	   4 - 16-bit Unicode
	\<length> - Length of message.
	\<data> - Message data. (Indicates the new voice mail count, if \<tele_id> is voice mail notification)

AT+CMGR=?	Test command returns the **OK** result code
Example	\<PDU Mode> Case of received message from base station: AT+CMGR=29 +CMGR: 1,"",52 07812811495346350808040947271002020221C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C20 OK 07 \<addr_len: 7byte> 81 \<type_addr: 129> 281149534635 \<Origination number: 821194356453> 080804094727 \<Date: 08/08/04,09:47:27> 1002 \<Teleservice_id: 4098(decimal)> 02 \<priority: urgent > 02 \<encoding_type: 7-bit ASCII > 21 \<data_len: 33> C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C3870E1C20 \<user_data: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa> Else: at+cmgr=31 +CMGR: 2,"",23 07801091346554F307801091346554F3100200000A61616161616161616161 OK

+CMGR - Read Message	
	07 \<addr_len: 7byte\> 81 \<type_addr:129\> 1091346554F3 \<Origination number: 01193645534 \> 07 \<addr_len: 7byte\> 81 \<type_addr:129\> 1091346554F3 \< Callback number: 01193645534 \> 1002 \<Teleservice_id: 4098(decimal)\> 00 \<priority: Normal \> 00 \<encoding_type: 8-bit Octet \> 0A \<data_len: 10\> 61616161616161616161 \<usr data: aaaaaaaaaa\> \<Text Mode\> AT+CSDH=1 OK AT+CMGR=1 +CMGR: "REC READ","0114933460","01149334690",20140109180259,129,4098,0,2,12 TEST MESSAGE OK AT+CMGR=2 +CMGR: "STO UNSENT","0114933460","0114933460",,129,4098,0,0,12 TEST MESSAGE OK

3.5.5.4. Message Sending And Writing

3.5.5.4.1. *Send Message - +CMGS*

+CMGS - Send Message	
(PDU Mode) **AT+CMGS=** **<length>**	**(PDU Mode)** Execution command sends to the network a 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)** and waits for the specified number of bytes. Parameter: **<length>** - length of the PDU to be sent in bytes. (excluding the Destination Address octets) 5..183 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. 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: **+CMGS: <mr>** where **<mr>** - message reference number. Note: if message sending fails for some reason, an error code is reported. Note: The limit of user data is 160 characters.
Example – PDU mode	AT+CMGF=0 OK AT+CMGS=35 > 07811091346554F307811096224658F1100200000166262626262626262626262626 2626262626262626262

+CMGS - Send Message	

	+CMGS: 4
	
	OK
	
	07 \<addr_len: 7byte>
	81 \<type_addr: 129>
	1091346554F3 \<Destination_address:01194356453>
	07 \<addr_len: 7byte>
	81 \<type_addr: 129>
	1096224658F1 \<callback_address:01692264851>
	1002 \<Teleservice_id: 4098(decimal)>
	00 \<priority: normal >
	00 \<encoding_type: 8-bit Octet >
	16 \<data_len: 22>
	6262626262626262626262626262626262626262
	        \<user_data: bbbbbbbbbbbbbbbbbbbbbb>
	
	AT+CMGR=20
	+CMGR: 2,"",31
	07811091346554F307811091346554F31002020212C3870E1C3870E1C387162C5
	8B162C58B1620
	
	OK
	07 \<addr_len: 7byte>
	81 \<type_addr: 129>
	1091346554F3 \<Destination_address:01194356453>
	07 \<addr_len: 7byte>
	81 \<type_addr: 129>
	1091346554F3 \<callback_address: 01194356453>
	1002 \<Teleservice_id: 4098(decimal)>
	02 \<priority: ungent >
	02 \<encoding_type: 7-bit ASCII >
	12 \<data_len: 18>
	C3870E1C3870E1C387162C58B162C58B1620
	        \<user_data: aaaaaaaaaabbbbbbbbbb>

(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);
	  ASCII characters in the set (0 9), #,*;
	  Note: The maximum length is different with every carrier.
	  In case of Sprint and Aeris.Net:
	    Maximum length is 32 characters

+CMGS - Send Message	

	In case of Verizon:
	    Maximum length is 20 characters
	
	<toda> - type of destination address
	    129 - number in national format
	    145 - number in international format (contains the "+")
	
	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:
	
	+CMGS: <mr>
	
	where
	<mr> - message reference number.
	
	Note: if message sending fails for some reason, an error code is reported.
	
	Note: The limit of user data is 160 characters.
	
	Note: To discard SMS, press the "ESC" key, an "OK" response will be returned.

AT+CMGS=?	Test command returns 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 – TEXT mode	AT+CMGF=1 OK AT+CMGS="9194547830" > Test SMS +CMGS: 1 OK

3.5.5.4.2. *Send Message From Storage - +CMSS*

+CMSS - Send Message From Storage	

AT+CMSS= <index>[,<da> [,<toda>]]	Execution command sends to the network a message which is already stored in the **<memw>** storage (see **+CPMS**) at the location **<index>**.
	
	Parameters:
	<index> - location value in the message storage **<memw>** of the message to send
	<da> - destination address, string type represented in the currently selected character set (see **+CSCS**); if it is given it shall be used instead of the one stored with the message.
	<toda> - type of destination address
	  129 - number in national format
	  145 - number in international format (contains the "+")
	
	If message is successfully sent to the network then the result is sent in the format:
	
	+CMSS: <mr>
	
	where:
	<mr> - message reference number.
	
	If message sending fails for some reason, an error code is reported:
	
	+CMS ERROR:<err>
	
	Note: to store a message in the **<memw>** storage see command **+CMGW.**

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: 5 OK AT+CMSS=5 +CMSS: 136 OK

3.5.5.4.3. *Write Message To Memory - +CMGW*

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

+CMGW - Write Message To Memory	
\<length\> **[,\<stat\>]**	Parameter: **\<length\>** - length in bytes of the PDU to be written. (excluding the Destination Address octets) 5..183 **\<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 is reported. Note: The limit of user data is 160 characters.
Example – PDU mode	AT+CMGF=0 OK AT+CMGW=35 > 07811091346554F307811096224658F110020000016626262626262626262626262626 2626262626262626262 +CMGW: 4 OK 07 \<addr_len: 7byte\> 81 \<type_addr: 129\> 1091346554F3 \<Destination_address:01194356453\> 07 \<addr_len: 7byte\> 81 \<type_addr: 129\> 1096224658F1 \<callback_address:01692264851\> 1002 \<Teleservice_id: 4098(decimal)\> 00 \<priority: normal \>

+CMGW - Write Message To Memory	
	00 <encoding_type: 8-bit Octet > 16 <data_len: 22> 62626262626262626262626262626262626262626262 <user_data: bbbbbbbbbbbbbbbbbbbbbb>

(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);
	  ASCII characters in the set (0 9), #,*;
	  Note: The maximum length is different with every carrier.
	  In case of Sprint and Aeris.Net:
	    Maximum length is 32 characters
	  In case of Verizon:
	    Maximum length is 20 characters
	
	<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)
	
	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 is reported.

+CMGW - Write Message To Memory	
	Note: To discard SMS, press the "ESC" key, an "OK" response will be returned. Note: The limit of user data is 160 characters.
AT+CMGW=?	Test command returns the **OK** result code.
Example – TEXT mode	AT+CMGW=? OK AT+CMGF=1 OK AT+CMGW > Test message > Ctrl+Z must be used to write message +CMGW: 1 OK AT+CMGW="9194397977" > Test SMS +CMGW: 2 OK AT+CMGW="9194397977",129 > Test SMS +CMGW: 3 OK
Note	To avoid malfunctions is suggested to wait for the **+CMGW: <index>** or **+CMS ERROR: <err>** response before issuing further commands.

3.5.5.4.4. *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 4 - delete all messages from **<memr>** storage.

+CMGD - Delete Message	
	Note: if **<delflag>** is present and not set to 0 then **<index>** is ignored and ME shall follow the rules for **<delflag>** shown above. Note: if the location to be deleted is empty, an error message is reported.
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),(0-4) OK AT+CMGD=11 *Delete message in 10th record* OK AT+CMGD=1,4 *Delete all messages* OK

3.5.6. Telit Custom AT Commands

3.5.6.1. General Configuration AT Commands

3.5.6.1.1. *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

3.5.6.1.2. *Model Identification - #CGMM*

#CGMM - Model Identification	
AT#CGMM	Execution command returns the device model identification code with command echo. Note: In case of HW2.00 model, returns CE910B-DUAL instead of CE910-DUAL.
AT#CGMM=?	Test command returns the **OK** result code.
Example	AT#CGMM #CGMM: CE910-DUAL OK

3.5.6.1.3. *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

3.5.6.1.4. *Product Serial Number Identification - #CGSN*

#CGSN - Product Serial Number Identification	
AT#CGSN	Execution command returns the device electronic serial number (ESN) or the mobile equipment identifier (MEID) with command echo. Note: This command returns 11-digit decimal of ESN. But, if MEID was entered to modem, this command returns 18-digit decimal of MEID. For more information about convert a MEID from hex to decimal please see the "MEID Conversion, HEX to DEC" in the Software User Guide.
AT#CGSN=?	Test command returns the **OK** result code.
Example	<ESN module> AT#CGSN #CGSN: 09210437158 OK <MEID module> AT#CGSN #CGSN: 268435456000000001 OK

3.5.6.1.5. *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

3.5.6.1.6. *Mobile Equipment Identifier - #MEID*

#MEID – Set Mobile equipment identifier	
AT#MEID?	Returns the current MEID Note: the MEID is hex values. For more information about convert a MEID from hex to decimal please see the "MEID Conversion, HEX to DEC" in the Software User Guide.
AT#MEID=?	Returns the OK result code.
Example	AT#MEID?

#MEID – Set Mobile equipment identifier	
	#MEID: A1234512345678 OK

3.5.6.1.7. *Software Shut Down – #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: after the issuing of this command any previous activity is 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. Note: after the issuing of this command any previous activity is terminated and the device will not respond to any further command.
AT#SHDN=?	Test command returns the OK result code.

3.5.6.1.8. *Fast power down – #FASTSHDN*

#FASTSHDN – Configure fast power down

AT#FASTSHDN=
[<enable>[,<pin>]]
	Set command configure fast power down
	
	Parameter:
	<enable> - enables/disables fast power down.
	  0 - disables (factory default)
	  1 - enables on GPIO event
	
	 <pin> - GPIO number is used for event monitoring.
	          Valid range is "any input pin" (see "Hardware User's Guide").
	          Default value is 0.
	
	If GPIO is set, (default setting is Pull-Up, GPIO is Low active) if GPIO goes to low then module enter power off autonomously without waiting for all EFS transaction is finished.
	
	Note: the values are stored in NVM and available on following reboot.
	Note: This command is available after NVM initialization at the boot-up.
	Note: When GPIO is used as **#FASTSHDN**, **that is priority is first of the other function. Customer should never be used for other functions.**
	Note: If the <enable> is 0, it does not use the <pin>.

AT# FASTSHDN	Execution command immediately fast power down, regardless the GPIO status and is not waiting for All EFS transaction will be finished.
AT# FASTSHDN?	Read command returns the #FASTSHDN current setting, in the format: **#FASTSHDN: <enable>,<pin>**
AT#FASTSHDN =?	Test command reports the range for the parameters **<enable>** and **<pin>**.

3.5.6.1.9. *Reset - $RESET*

$RESET – Reset The Modem	
AT$RESET	Immediately resets the modem.
AT$RESET=?	Test command returns the **OK** result code.

3.5.6.1.10.   *Reboot - #REBOOT*

#REBOOT - Reboot	
AT#REBOOT	Execution command reboots the module.
AT#REBOOT=?	Test command returns the OK result code.
Example	AT#REBOOT=? OK AT#REBOOT OK

3.5.6.1.11.   Extended Reset - #Z

#Z - Extended reset	
AT#Z=<profile>	Set command loads both base section and extended section of the specified user profile stored with AT&P. Parameter **<profile>** 0 – user profile 0 1 – user profile 1
AT#Z=?	Test command tests for command existence.

3.5.6.1.12.   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 **OK** result code.

3.5.6.1.13.   Query Temperature Overflow - #QTEMP

#QTEMP - Query Temperature Overflow	
AT#QTEMP=	Set command has currently no effect. The interpretation of parameter

#QTEMP - Query Temperature Overflow	
[<mode>]	<mode> is currently not implemented: any value assigned to it will simply have no effect.

AT#QTEMP?	Read command queries the device internal temperature sensor for over temperature and reports the result in the format:
	
	#QTEMP: <temp>
	
	where:
	<temp> - over temperature indicator
	  0 - the device temperature is in the working range
	  1 - the device temperature is out of the working range
	
	Note: typical *temperature working range* is (-40℃..+85℃); anyway you are strongly recommended to consult the "Hardware User Guide" to verify the real temperature working range of your module

AT#QTEMP=?	Test command reports supported range of values for parameter **<mode>**.
Note	The device should not be operated out of its working temperature range, elsewhere proper functioning of the device is not ensured.

3.5.6.1.14.   *Temperature Monitor – #TEMPMON*

#TEMPMON - Temperature Monitor	

AT#TEMPMON=
<mod>
[,<urcmode>
[,<action>
[,<hyst_time>
[,<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 (see Note)
	     -1 - operating temperature lower bound (see Note)
	      0 - normal temperature
	      1 - operating temperature upper bound (see Note)
	      2 - extreme temperature upper bound (see Note)
	
	   <value>
	   actual temperature expressed in Celsius degrees

Setting of the following optional parameters has meaning only if <mod>=0:

<urcmode> - URC presentation mode. **(Default 0)**
  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 0)**. If **<action>** is not zero, it is mandatory to set the **<hyst_time>** parameter too.
  0 - no action (00)
  1 - automatic shut-down when the temperature is beyond the extreme bounds (01)
  2 - RF TX circuits automatically disabled (using +**CFUN=2**) when operating temperature bounds are reached. When the temperature is back to normal the module is brought back to the previous state, before RF TX disabled. (10)
  4 - the 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. (100)
Note: Possible values for the parameter **<action>** are form 0 to 7 (000, 001, 010, 011, 100, 101, 110 and 111)

<hyst_time> - hysteresis time: all the actions happen only if the extreme or operating bounds are maintained at least for this period. This parameter is needed and required if **<action>** is not zero.
    0..255 - time in seconds
Note: **<action>** can assume values from 1-7

<GPIO> - GPIO number. Valid range is "any output pin" (see "Hardware User's Guide"). This parameter is needed and required only if **<action>=4** is enabled.

	Note: **if the <GPIO> is specified <action> shall** assume values from 4-7. Note: last **<urcmode>** settings are saved as extended profile parameters. Note: last **<action>**, **<hyst_time>** and **<GPIO>** settings are global parameters saved in NVM
AT#TEMPMON?	Read command reports the current parameter settings for **#TEMPMON** command in the format: **#TEMPMON: <urcmode>,<action>[,<hyst_time>[,<GPIO>]]**
AT#TEMPMON=?	Test command reports the supported range of values for parameters **<mod>**, **<urcmode>**, **<action>**, **<hyst_time>** and **<GPIO>**
Note	**CDMA Limits** <table><tr><td>**Extreme Temperature Lower Bound**(*)</td><td>**-40°C**</td></tr><tr><td>**Operating Temperature Lower Bound**(*)</td><td>**-40°C**</td></tr><tr><td>**Operating Temperature**</td><td></td></tr><tr><td>**Operating Temperature Upper Bound**(*)</td><td>**+85°C**</td></tr><tr><td>**Extreme Temperature Upper Bound**(*)</td><td>**+85°C**</td></tr></table> (*) Due to temperature measurement uncertainty there is a tolerance of +/-2°C The automatic power off is deferred in case of an Emergency Call

3.5.6.1.15. *General Purpose Input/Output Pin Control – #GPIO*

#GPIO - General Purpose Input/Output Pin Control	

AT#GPIO=[<pin>, <mode>[,<dir>]]	Execution command sets the value of the general purpose output pin **GPIO<pin>** according to **<dir>** and **<mode>** parameter.
	Not all configuration for the three parameters are valid.
	
	Parameters:
	<pin> - GPIO pin number; supported range is from 1 to a value that depends on the hardware.
	<mode> - its meaning depends on **<dir>** setting:
	  0 - no meaning if **<dir>=0** - INPUT
	    - output pin cleared to 0 (**Low**) if **<dir>=1** - OUTPUT
	    - no meaning if **<dir>=2** - ALTERNATE FUNCTION
	  1 - no meaning if **<dir>=0** - INPUT
	    - output pin set to 1 (**High**) if **<dir>=1** - OUTPUT
	    - no meaning if **<dir>=2** - ALTERNATE FUNCTION
	  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
	    - Reports a no meaning value if **<dir>=2** - ALTERNATE FUNCTION

#GPIO - General Purpose Input/Output Pin Control	

	<dir> - GPIO pin direction
	  0 - pin direction is INPUT
	  1 - pin direction is OUTPUT
	2 - pin direction is ALTERNATE FUNCTION (see Note).
	3 - pin direction is "Fast power down". It is only possible to set by #FASTSHDN (see #FASTSHDN)
	
	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.
	
	Note: "ALTERNATE FUNCTION" value is valid only for following pins:
	• **GPIO1** - alternate function is "STAT_LED" (see **#SLED**)
	• **GPIO6** - alternate function is "Alarm Output" (see **+CALA**)
	
	Note: while using the pins in the alternate function, the GPIO read/write access to that pin is not accessible and shall be avoided.

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
AT#GPIO=?	Test command reports the supported range of values of the command parameters **<pin>**, **<mode>** and **<dir>**.
Example	AT#GPIO=3,0,1 OK AT#GPIO=3,2 #GPIO: 1,0 OK AT#GPIO=4,1,1 OK AT#GPIO=5,0,0 OK AT#GPIO=6,2

#GPIO - General Purpose Input/Output Pin Control	
	#GPIO: 0,1 OK

3.5.6.1.16. *STAT_LED GPIO Setting - #SLED*

#SLED - STAT_LED GPIO Setting	

AT#SLED=<mode>
[,<on_duration>
[,<off_duration>]]
	Set command sets the behaviour of the **STAT_LED** GPIO
	
	Parameters:
	<mode> - defines how the **STAT_LED** GPIO is handled
	  0 - GPIO tied **Low**
	  1 - GPIO tied **High**
	  2 - GPIO handled by Module Software (factory default)
	  3 - GPIO is turned on and off alternatively, with period defined by the sum
	    <on_duration> + <off_duration>
	<**on_duration**> - duration of period in which **STAT_LED** GPIO is tied **High** while
	          <mode>=3
	  1..100 - in tenth of seconds (default is 10)
	<**off_duration**> - duration of period in which **STAT_LED** GPIO is tied **Low** while
	          <mode>=3
	  1..100 - in tenth of seconds (default is 10)
	
	Note: values are saved in NVM by command **#SLEDSAV**
	
	Note: at module boot the **STAT_LED** GPIO is always tied **High** and holds this value until the first NVM reading.
	
	Note: Set **AT#GPIO=1,0,2** to enable LED on the EVK.

AT#SLED?	Read command returns the **STAT_LED** GPIO current setting, in t he format: **#SLED: <mode>,<on_duration>,<off_duration>**
AT#SLED=?	Test command returns the range of available values for parameters **<mode>**, **<on_duration>** and **<off_duration>**.
Example	AT#SLED=? #SLED: (0-3),(1-100),(1-100) OK AT#SLED? #SLED: 2,10,10 OK AT#SLED=0 OK AT#SLED=0 OK AT#SLED=1 OK AT#SLED=2

#SLED - STAT_LED GPIO Setting	
	OK AT#SLED=3,50,50 OK AT#SLED? #SLED: 3,50,50 OK AT#SLED=3,5,5 OK AT#SLED? #SLED: 3,5,5 OK

3.5.6.1.17. *Save STAT_LED GPIO Setting - #SLEDSAV*

#SLEDSAV - Save STAT_LED GPIO Setting	
AT#SLEDSAV	Execution command saves **STAT_LED** setting in NVM.
AT#SLED=?	Test command returns **OK** result code.

3.5.6.1.18. *SMS Ring Indicator - #E2SMSRI*

#E2SMSRI - SMS Ring Indicator	

AT#E2SMSRI=
[**<n>**]
	Set command enables/disables the Ring Indicator pin response to an incoming SMS message. If enabled, a negative going pulse is generated on receipt of an incoming SMS message. The duration of this pulse is determined by the value of **<n>**.
	
	Parameter:
	<n> - **RI** enabling
	  0 - disables **RI** pin response for incoming SMS messages (factory default)
	  50..1150 - enables **RI** pin response for incoming SMS messages. The value of **<n>** is the duration in ms of the pulse generated on receipt of an incoming SM.

AT#E2SMSRI?	Read command reports the duration in ms of the pulse generated on receipt of an incoming SM, in the format: **#E2SMSRI: <n>** Note: as seen before, the value **<n>=0** means that the **RI** pin response to an incoming SM is disabled.
AT#E2SMSRI=?	Reports the range of supported values for parameter **<n>**
Example	AT#E2SMSRI=? #E2SMSRI: (0,50-1150)) OK AT#E2SMSRI?

#E2SMSRI - SMS Ring Indicator	
	#E2SMSRI: 0 OK AT#E2SMSRI=50 OK AT#E2SMSRI? #E2SMSRI: 50 OK

3.5.6.1.19. *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 CE910-Series
	<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>**.

3.5.6.1.20. *Digital/Analog Converter Control - #DAC*

#DAC - Digital/Analog Converter Control	

AT#DAC=
[<enable>
[,<value>]]
	It has no effect and is included only for backward compatibility.
	
	Parameters:
	<enable> - enables/disables DAC output.
	  0 - disables pin; it is in high impedance status (factory default)
	  1 - enables pin; the corresponding output is driven
	<value> - scale factor of the integrated output voltage; it must be present if
	        <enable>=1

#DAC - Digital/Analog Converter Control	
	0..1023 - 10 bit precision Note: **integrated output voltage = MAX_VOLTAGE * value / 1023**
AT#DAC?	Read command reports whether the **DAC_OUT** pin is currently enabled or not, along with the integrated output voltage scale factor, in the format: **#DAC: <enable>,<value>**
AT#DAC=?	Test command reports the range for the parameters **<enable>** and **<value>**.
Example	*Enable the DAC out and set its integrated output to the 50% of the max value:* AT#DAC=1,511 OK *Disable the DAC out:* AT#DAC=0 OK
Note	With this command the DAC frequency is selected internally. D/A converter must not be used during POWERSAVING. **DAC_OUT** line must be integrated (for example with a low band pass filter) in order to obtain an analog voltage. For a more in depth description of the integration filter refer to the hardware user guide.

3.5.6.1.21. *Auxiliary Voltage Output Control – #VAUX*

#VAUX- Auxiliary Voltage Output Control	

AT#VAUX=
[<n>,<stat>]
	Set command enables/disables the Auxiliary Voltage pins output.
	
	Parameters:
	<n> - **VAUX** pin index
	  1 - there is currently just one **VAUX** pin
	<stat>
	  0 - output off
	  1 - output on
	  2 - query current value of **VAUX** pin
	
	Note: when **<stat>=2** and command is successful, it returns:
	
	  #VAUX: <value>
	
	  where:
	  <value> - power output status
	    0 - output off
	    1 - output on
	
	Note: the current setting is stored through **#VAUXSAV**

AT#VAUX?	Read command reports whether the Auxiliary Voltage pin output is currently

#VAUX- Auxiliary Voltage Output Control	
	enabled or not, in the format: #VAUX: <value>
AT#VAUX=?	Test command reports the supported range of values for parameters <n>, <stat>.

3.5.6.1.22. Auxiliary Voltage Output Save - #VAUXSAV

#VAUXSAV - Auxiliary Voltage Output Save	
AT#VAUXSAV	Execution command saves the actual state of #VAUX pin to NVM. The state will be reload at power-up.
AT#VAUXSAV=?	Test command returns the OK result code.

3.5.6.1.23. V24 Output Pins Configuration - #V24CFG

#V24CFG - V24 Output Pins Configuration	

AT#V24CFG=<pin>, <mode>	Set command sets the AT commands serial port (UART) 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)
	    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"
	    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"
	
	<mode> - AT commands serial port interface hardware pins mode:
	  0 - AT commands serial port mode: output pins are controlled by serial port device driver. (default)
	  1 - GPIO mode: output pins are directly controlled by #V24 command only.

AT#V24CFG?	Read command returns actual mode for all the pins in the format: #V24CFG: <pin1>,<mode1>[<CR><LF><CR><LF> #V24CFG: <pin2>,<mode2>[...]] Where: <pinn> - AT command serial port interface HW pin <moden> - AT commands serial port interface hardware pin mode
AT#V24CFG=?	Test command reports supported range of values for parameters <pin> and <mode>.

3.5.6.1.24.   *V24 Output Pins Control – #V24*

#V24 - V24 Output Pins Control	

AT#V24=<pin>[,<state>]	Set command sets the AT commands serial port (UART) 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"
	   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 state of the pin.

AT#V24?	Read command returns actual state for all the pins 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>**.

3.5.6.1.25.   *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 millivolt: it is the real battery voltage only if charger is not connected; if the charger is connected this value depends on the charger voltage.

#CBC- Battery And Charger Status	
	NOTE: '1' and '2' at <ChargerState> is not supported.
AT#CBC=?	Test command returns the **OK** result code.

3.5.6.1.26.   *Dialling Mode - #DIALMODE*

#DIALMODE - Dialling Mode	

AT#DIALMODE= [<mode>]	Set command sets dialling modality.
	
	Parameter:
	<mode>
	  0 - (voice call only) **OK** result code is received as soon as it starts remotely ringing (factory default)
	  1 - (voice call only) **OK** result code is received after the called party answers or entered traffic state (CDMA models only). Any character typed aborts the call and **OK** result code is received.
	  2 - (voice call and circuit data call) the following custom result codes are received, monitoring step by step the call status:
	    DIALING (MO in progress)
	    RINGING (remote ring, not supported CDMA models)
	    CONNECTED (remote call accepted or traffic state entered on CDMA models)
	    RELEASED (after **ATH**)
	    DISCONNECTED (remote hang-up)
	
	Note: The setting is saved in NVM and available on following reboot.
	Note: "RINGING" doesn't work in CDMA models because it is working in receiving "call origination progress indication". But CDMA is not supported "call origination progress indication"
	Note: Mode of 2 is not working on packet data call. Currently circuit data call is not supporting in CDMA networks. So mode of 2 is not working on data call.

AT#DIALMODE?	Read command returns current **ATD** dialing mode in the format: **#DIALMODE: <mode>**
AT#DIALMODE=?	Test command returns the range of values for parameter **<mode>**

3.5.6.1.27.   *Automatic Call - #ACAL*

#ACAL - Automatic Call	

AT#ACAL= [<mode>]	Set command enables/disables the automatic call function.
	
	Parameter:
	<mode>
	  0 - disables the automatic call function (factory default)
	  1 - enables the automatic call function.

#ACAL - Automatic Call	

AT#ACAL?	Read command reports whether the automatic call function is currently enabled or not, in the format:
	
	#ACAL: <mode>
	
	where
	
	<mode>
	  0 - automatic call function disabled
	  1 - automatic call function from internal phonebook enabled

AT#ACAL=?	Test command returns the supported range of values for parameter **<mode>**.
Note	See **&Z** to write and **&N** to read the number on module internal phonebook.

3.5.6.1.28. *Extended Automatic Call - #ACALEXT*

#ACALEXT - Extended Automatic Call	

AT#ACALEXT=
<mode>,<index>
	Set command enables/disables the extended automatic call function.
	
	Parameters:
	<mode>
	  0 - disables the automatic call function (factory default)
	  1 - enables the automatic call function from internal phonebook.
	<index> - it indicates a position in the currently selected phonebook.
	  (default value is 0)
	
	If the extended automatic call function is enabled and **&D2** has been issued, the transition **OFF/ON** of **DTR** causes an automatic call to the number stored in position **<index>** in the selected phonebook.

AT#ACALEXT?	Read command reports either whether the automatic call function is currently enabled or not, and the last **<index>** setting in the format: **#ACALEXT: <mode>,<index>**
AT#ACALEXT=?	Test command returns the range of available values for parameter **<mode>** and **<index>**
Note	Issuing #ACALEXT causes the #ACAL <mode> to be changed. Issuing AT#ACAL=1 causes the #ACALEXT <index> to be set to default. It is recommended to NOT use contemporaneously either #ACALEXT and #ACAL
Note	See **&Z** to write and **&N** to read the number on module internal phonebook.

3.5.6.1.29. *Extended Call Monitoring - #ECAM*

#ECAM - Extended Call Monitoring	
AT#ECAM=	This command enables/disables the call monitoring function in the ME.

#ECAM - Extended Call Monitoring	

[<onoff>]	Parameter:
	<onoff>
	  0 - disables call monitoring function (factory default)
	  1 - enables call monitoring function; the ME informs about call events, such as incoming call, connected, hang up etc. using the following unsolicited indication:
	
	    #ECAM: <ccid>,<ccstatus>,<calltype>,,,[<number>,<type>]
	
	    where
	    <ccid> - call ID
	    <ccstatus> - call status
	      0 - idle
	      1 - calling (MO)
	      2 - connecting (MO)
	      3 - active
	      4 - hold
	      5 - waiting (MT)
	      6 - alerting (MT)
	      7 – busy
	      8 – retrieved
	      9 – CNAP (Calling Name Presentation) information (MT)
	
	    Note: **2 - connecting (MO)**, **4 – hold**, **5 - waiting (MT)**, **7 – busy** and **8 – retrieved** are not supported for CE910-Series.
	    <calltype> - call type
	      1 - voice
	      2 – circuit switched data
	    <number> - called number (valid only for **<ccstatus>=1**)
	    <type> - type of **<number>**
	      129 - national number
	      145 - international number
	
	Note: the unsolicited indication is sent along with usual codes (**OK**, **NO CARRIER**, **BUSY**...).

AT#ECAM?	Read command reports whether the extended call monitoring function is currently enabled or not, in the format: **#ECAM: <onoff>**
AT#ECAM=?	Test command returns the list of supported values for **<onoff>**

3.5.6.1.30.   *SMS Overflow - #SMOV*

#SMOV - SMS Overflow	
AT#SMOV= [<mode>]	Set command enables/disables the SMS overflow signalling function.

#SMOV - SMS Overflow	

	Parameter:
	 <mode>
	   0 - disables SMS overflow signaling function (factory default)
	   1 - enables SMS overflow signalling function; when the maximum storage capacity has reached, the following network initiated notification is send:
	
	 #SMOV: <memo>
	 < memo >
	   "ME" – SMS memory storage into module
	   "SM" – SIM SMS memory storage (In case supporting RUIM)

AT#SMOV?	Read command reports whether the SMS overflow signalling function is currently enabled or not, in the format: **#SMOV: <mode>**
AT#SMOV=?	Test command returns the supported range of values of parameter **<mode>**.
Example	AT+CPMS? +CPMS: "ME",99,99,"ME",99,99 OK AT+CMGD=1 OK AT#SMOV=1 OK AT+CMGF=1 OK AT+CMGW="1111111111" > aaaaaaaa +CMGW: 1 OK #SMOV: "ME"

3.5.6.1.31. *Audio Codec – #CODEC*

#CODEC - Audio Codec	
AT#CODEC= [<codec>]	Set command sets the audio codec mode. Parameter: **<codec>** 0 - SO3 : EVRC (factory default) 1 - SO32768 : QCELP 2 - SO17 : High Rate Voice Service (13 kbps) IS-733 3 - SO68 : 4GV Narrow Band Service Option 4 - SO73 : Enhanced Variable Rate Voice Service EVRC

#CODEC - Audio Codec	
AT#CODEC?	Read command returns current audio codec mode in the format: **#CODEC: \<codec>**
AT#CODEC=?	Test command returns the range of available values for parameter **\<codec>**
Example	AT#CODEC=? #CODEC: (0-4) OK AT#CODEC? #CODEC: 1 OK AT#CODEC=0 OK

3.5.6.1.32. *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 receiving the SYNC message.
	
	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)
	  1..15 - as a sum of:
	    1 - enables automatic date/time updating
	    2 - enables Full Network Name applying (Not Supported)
	    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)
	    (default: 7)
	\<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)

#NITZ - Network Timezone	
	"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 – On/Off indicator for Daylight Saving Time; range is 0-1.
	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>**.

3.5.6.1.33. *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 - doesn't skip the escape sequence; its transmission is enabled (factory default). 1 - skips the escape sequence; its transmission is not enabled. Note: in case of an FTP connection, the escape sequence is 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>**.

3.5.6.1.34. *Escape Sequence Guard Time – #E2ESC*

#E2ESC - Escape Sequence Guard Time	

#E2ESC - Escape Sequence Guard Time	

AT#E2ESC= [<gt>]	Set command sets a guard time in seconds for the escape sequence in CDMA to be considered a valid one (and return to on-line command mode).
	
	Parameter:
	<gt>
	  0 - no guard time (factory default)
	  1..10 - guard time in seconds
	
	Note: if the Escape Sequence Guard Time is set to a value different from zero, it overrides the one set with **S12**.

AT#E2ESC?	Read command returns current value of the escape sequence guard time, in the format: **#E2ESC: <gt>**
AT#E2ESC=?	Test command returns the **OK** result code.

3.5.6.1.35.   *PPP Connection Authentication Type - #GAUTH*

#GAUTH – PPP Connection Authentication Type	
AT#GAUTH= [<type>]	Set command sets the PPP connection authentication type. Parameter **<type>** 3 – AUTO authentication (PAP and CHAP , factory default)
AT#GAUTH?	Read command reports the current PPP connection authentication type, in the format: **#GAUTH: <type>**
AT#GAUTH=?	Test command returns the range of supported values for parameter **<type>**.

3.5.6.1.36.   *RTC Status - #RTCSTAT*

#RTCSTAT - RTC Status	
AT#RTCSTAT= [<status>]	Set command resets the RTC status flag. Parameter: **<status>** 0 - Set RTC Status to **RTC HW OK** Note: the initial value of RTC status flag is **RTC HW Error** and it doesn't change until a command **AT#RTCSTAT=0** is issued. Note: if a power failure occurs and the buffer battery is down the RTC status flag is set to **1**. It doesn't change until command **AT#RTCSTAT=0** is issued.
AT#RTCSTAT?	Read command reports the current value of RTC status flag, in the format:

#RTCSTAT - RTC Status	
	#RTCSTAT: <status>
AT#RTCSTAT=?	Test command returns the range of supported values for parameter **<status>**

3.5.6.1.37.   *GSM Antenna Detection - #GSMAD*

#GSMAD - GSM Antenna Detection	

AT#GSMAD= **<mod>,** **[<urcmode>** **[,<interval>** **[,<detGPIO>** **[,<repGPIO>]]]]**	Set command sets the behaviour of antenna detection algorithm
	
	Parameters:
	<mod>
	 0 - detection algorithm not active
	 1 - periodic activation of the antenna detection algorithm; detection is started every **<interval>** period, using **<detGPIO>** for detection; if the algorithm detects a change in the antenna status the module is notified by URC **#GSMAD** (see format below)
	 2 - instantaneous activation of the antenna detection algorithm; if the algorithm detects a change in the antenna status the module is notified by URC (see format below); this instantaneous activation doesn't affect a periodic activation eventually started before.
	
	   URC format:
	
	   #GSMAD: <presence>
	
	   where:
	   <presence>
	     0 - antenna connected.
	     1 - antenna connector short circuited to ground.
	     2 - antenna connector short circuited to power.
	     3 - antenna not detected (open).
	
	<urcmode> - URC presentation mode. It has meaning only if **<mod>** is 1.
	 0 - it disables the presentation of the antenna detection URC
	 1 - it enables the presentation of the antenna detection URC, whenever the antenna detection algorithm detects a change in the antenna status; the unsolicited message is in the format:
	
	   #GSMAD: <presence>
	
	   where:
	   <presence> is as before
	
	<interval> - duration in seconds of the interval between two consecutive antenna detection algorithm runs (default is 120). It has meaning only if **<mod>**

	is 1.
	  1..3600 - seconds
	
	\<detGPIO\> - defines which GPIO shall be used as input by the Antenna Detection
	         algorithm. For the \<detGPIO\> actual range see Test Command
	
	\<repGPIO\> - defines which GPIO shall be used by the Antenna Detection
	         algorithm to report antenna condition. It has meaning only if **\<mod\>** is
	         1. For the **\<repGPIO\>** actual range see Test Command.
	  0 - no report is made using GPIO
	
	Note: last **\<urcmode\>** settings are saved as extended profile parameters.
	
	Note: GPIO is set to LOW when antenna is connected. Set to HIGH otherwise
	
	Note: **#GSMAD** parameters, excluding **\<urcmode\>**, are saved in NVM.

AT#GSMAD=?	Test command reports the supported range of values for parameters **\<mod\>,** **\<urcmode\>, \<interval\>, \<detGPIO\>** and **\<repGPIO\>**.
AT#GSMAD?	Read command returns the current parameter settings for **#GSMAD** command in the format: **#GSMAD: \<mod\>,\<urcmode\>,\<interval\>,\<detGPIO\>,\<repGPIO\>**

3.5.6.1.38.   *Power Saving Mode Ring Indicator - #PSMRI*

#PSMRI – Power Saving Mode Ring Indicator	
AT#PSMRI=\<n\>	Set command enables/disables the Ring Indicator pin response to an URC message while modem is in power saving mode. If enabled, a negative going pulse is generated, when URC message for specific event is invoked. The duration of this pulse is determined by the value of **\<n\>**. Parameter: **\<n\>** - **RI** enabling 0 - disables **RI** pin response for URC message(factory default) 50-1150 - enables **RI** pin response for URC messages. Note: the behavior for #PSMRI is invoked only when modem is in sleep mode (AT+CFUN=5 and DTR Off on Main UART)
AT#PSMRI?	Read command reports the duration in ms of the pulse generated, in the format: **#PSMRI: \<n\>**
AT#PSMRI=?	Reports the range of supported values for parameter **\<n\>**
Note	When RING signal for incoming call/SMS/socket listen is enabled, the behavior for #PSMRI will be ignored.

3.5.6.1.39.   Command Mode Flow Control – #CFLO

#CFLO – Command Mode Flow Control	

AT#CFLO=
 <mode>
	Set command enables/disables the flow control in command mode. If enabled, current flow control is applied to both command mode and data mode.
	
	 Parameter:
	 <mode>
	   0 – Disable flow control set in command mode (factory default)
	   1- Enable flow control set in command mode
	
	 Note: This behavior is valid only for Main UART port.

AT#CFLO?	Read command reports current setting value , in the format: **#CFLO: <mode>**
AT#CFLO=?	Test command reports the range of supported values for parameter **<mode>**

3.5.6.1.40.   Cell Monitor – #MONI

#MONI - Cell Monitor	

AT#MONI[=
 [**<number>**]]
	Set command to select one of three pilot set, Active/Candidate/
	 Neighbour set, from which extract CDMA-related informations.
	
	 Parameter:
	 <number>
	
	 <CDMA network>
	   0 – it is the active set (factory default)
	   1 – it is the candidate set
	   2 – it is the neighbour set
	   3..7 – it is not available
	
	 Note: Candidate set (number = 1) display in traffic state only. That is CDMA specifications (refer to 2.6.6.1.2 Pilot Sets of C.S0005). If mobile stays in Idle state, pilot set and strength are displayed to 0.
	
	   a) When number is set to 0 (active set), extracting information format is:
	
	     #MONI: A_PN:<PNn>,A_PN_STR:<PNn_str>[<CR><LF>
	     #MONI: A_PN:<PNn>,A_PN_STR:<PNn_str>[…]]
	
	   b) When number is set to 1 (candidate set), extracting information format is:
	
	     #MONI: C_PN:<PNn>,C_PN_STR:<PNn_str>[<CR><LF>
	     #MONI: C_PN:<PNn>,C_PN_STR:<PNn_str>[…]]
	
	   c) When number is set to 2 (neighbour set), extracting information format is:

#MONI - Cell Monitor	
	#MONI: N_PN:\<PNn>,N_PN_STR:\<PNn_str>[\<CR>\<LF> #MONI: N_PN:\<PNn>,N_PN_STR:\<PNn_str>[…]] where: **\<PNn>** - Value of **n**th (active/candidate/neighbour)pilot sets. **\<PNn_Str>** - Pilot strength of **n**th (active/candidate/neighbour)pilot sets.
AT#MONI=?	Test command returns the **OK** result code**.**
Note	Maximum value of parameter **n** is 3. Top 3 PNs of Active/Candidate/Neighbour set are displayed in the signal strength order.
Example	AT#MONI=0 OK AT#MONI A_PN:80,A_PN_STR:-10 OK AT#MONI=? OK

3.5.6.1.41. *I2C data via GPIO - #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, without the LSB used for read\write command, 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. To complete the operation send **Ctrl-Z** char (**0x1A** hex); to exit without writing the message send **ESC** char (**0x1B** hex).

#I2CWR – Write to I2C	
	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 **E.g.** AT#I2CWR=2,3,20,10,14 > 00112233445566778899AABBCCDD<ctrl-z> OK Set GPIO2 as SDA, GPIO3 as SCL; Device I2C address is 0x20; 0x10 is the address of the first register where to write I2C data; 14 data bytes will be written since register 0x10 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#I2CWR=?	Test command returns the range of each parameter**.**

3.5.6.1.42.   *I2C data from GPIO - #I2CRD*

#I2CRD – Read from I2C	
AT#I2CRD= <sdaPin>, <sclPin>, <deviceId>, <registerId>, <len>	This command is used to Read Data from 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, without the LSB used for read\write command, 10 bit addressing supported. Value has to be written in hexadecimal form (without 0x). <registerId>: Register to read data from , range 0..255. Value has to be written in hexadecimal form (without 0x). <len>: number of data to receive. Valid range is 1-254. Data Read from I2C will be dumped in Hex:

#I2CRD – Read from I2C	
	E.g. AT#I2CRD=2,3,20,10,14 #I2CRD: 00112233445566778899AABBCCDD OK 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 returns the range of each parameter.

3.5.6.1.43.   *Control GPIOs based on Signal Strength - #CSQLED*

#CSQLED-LED control by Signal strength	

**AT#CSQLED=
<enable>[,<led1Pin>,
<led2Pin>,<led3Pin>]**
	Set command control LEDs based on Signal strength.
	
	Parameter:
	<enable>
	  Control LEDs based on signal strength :
	  0 : disable (default)
	  1 : enable
	
	<led1Pin>: GPIO number for led1. Valid range is "any output pin" (see "Hardware User's Guide").
	Default value of led1Pin is 2.
	
	<led2Pin>: GPIO number for led2. Valid range is "any output pin" (see "Hardware User's Guide").
	Default value of led1Pin is 3.
	
	<led3Pin>: GPIO number for led3. Valid range is "any output pin" (see "Hardware User's Guide").
	Default value of led1Pin is 4.
	
	Note: This value stored in NVM region.
	Note: LED table base on Signal strength.
	
	LED table:
	
	(see table below)

AT+CSQ response = +CSQ: xx, 99 where xx value is below	LED 1 bar : high = on, low = off	LED 2 bars : high = on, low = off	LED 3 bars : high = on, low = off

#CSQLED-LED control by Signal strength					
	0	low	low	low	
	1	low	low	low	
	2	low	low	low	
	3	low	low	low	
	4	low	low	low	
	5	low	low	low	
	6	low	low	low	
	7	high	low	low	
	8	high	low	low	
	9	high	low	low	
	10	high	low	low	
	11	high	low	low	
	12	high	low	low	
	13	high	low	low	
	14	high	low	low	
	15	high	high	low	
	16	high	high	low	
	17	high	high	low	
	18	high	high	low	
	19	high	high	low	
	20	high	high	low	
	21	high	high	low	
	22	high	high	low	
	23	high	high	low	
	24	high	high	high	
	25	high	high	high	
	26	high	high	high	
	27	high	high	high	
	28	high	high	high	
	29	high	high	high	
	30	high	high	high	
	31	high	high	high	
AT#CSQLED?	Read command reports the current setting values in the format: #CSQLED: **<enable>,<led1Pin>,<led2Pin>,<led3Pin>**				
AT#CSQLED=?	Test command returns **OK**.				

3.5.6.1.44.   *Change and insert file system password – #FILEPWD*

#FILEPWD – Change and insert file system password	

AT#FILEPWD=
 <Mode>,<Pwd>
 [,<NewPwd>]
	This command changes and inserts file system password.
	
	 Parameters:
	 <Mode>:
	  1 – insert file system password;
	  2 – change file system password.
	 <Pwd>:

#FILEPWD – Change and insert file system password	
	current password when inserting password, old password when changing password, string type (factory default is the empty string ""). **<NewPwd>:** new password when changing password, string type (only allowed if **<Mode>** parameter is 2). Note: maximum password length is 12 characters. Note: password is saved in NVM. Note: password value doesn't depend on the specific CMUX instance. Note: in default configuration current password is equal to the empty string "" and password will be always considered inserted. Note: if current password is different from the empty string "", password will be always not inserted at power on. Note: if current password is different from the empty string "", after successful password insertion (<Mode> 1) password will remain inserted until power off. Note: after successful password change (<Mode> 2) password will be not inserted. Note: if current password is different from the empty string "" and password is not inserted then AT commands that make use of the file system (SCRIPT) will have either ERROR or +CME ERROR: 16 or +CME ERROR: incorrect password response depending on AT+CMEE setting.
AT#FILEPWD=?	Test command reports the supported range of values for parameters.
Example	AT#FILEPWD=2,"","mynewpwd" OK AT#FILEPWD=1,"mynewpwd" OK

3.5.6.1.45.   *Delete All Phonebook Entries - #CPBD*

#CPBD - Delete All Phonebook Entries	
AT#CPBD	Execution command deletes all phonebook entries in the current phonebook memory storage selected with **+CPBS**. Note: in case of SM or ME, it takes some time to delete all its entries.
AT#CPBD=?	Test command returns the OK result code.

3.5.6.1.46.   *Enable Test Mode command in not signaling mode - #TESTMODE*

#TESTMODE – Enable Test Mode command in not signalling mode	
AT#TESTMODE= <command>	The command allows setting module in not signalling 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 :
This string corresponds to a command. To be accepted by AT#TESTMODE, the command has to belong to the following list of command enabled for this use:
Commands quoted string type.

- "*TM*": forces the module in Test Mode.
- "*SETCDMABAND <band>*": sets the CDMA band (BC0 or BC1).

band	Band
0	BC0(Support)
1	BC1(Support)

*BC0(Cellular), BC1(PCS)

- "*CH <CDMA_CH>*": sets the Channel.

CDMA_CH	Band
1 ~ 799	BC0
991 ~ 1023	
1 ~ 1199	BC1

- "*TCH*": starts the non-stop module transmission.
        It enables TX power.

- "TXPDM <TXPDM>": set the value of desired
        TX power strength.

MODEL	BAND	TXPDM			Power Range
		Low CH	Mid CH	High CH	
CE910	BC0	80	79	80	23~25dBm
	BC1	82	81	83	23~25dBm

- "*OM*": forces the module in Operative Mode.

- "ESC": exits the current non-stop sequence.
  It must use to stop TX transmission.

.Note
- Bands support varies depending on the product.

	- In case of RUIM supported model, this command operate when RUIM is inserted. Note 1: This command should be checked individually Note 2: In Test Mode, the other AT commands doesn't use. Note 3: In Test Mode, DTE speed recommend 115200(see +IPR) Note 4: In Test Mode, the multiplexing protocol control channel recommend don't enable(see +CMUX) Note 5: After issuing AT#TESTMODE="TM" or "OM", the module have to reboots. Note 6: The Test Mode Status is stored in NVM. Note 7: "TM" command just set on the Online or FTM mode status. Note 8: "TCH" command have to set after the "SETCDMABAND" command set. Note 9: After "TCH"command set, "SETCDMABAND"command can't set. If you want to re-setting the cdmaband, you have to set 'ESC" command prior to "SETCDMABAND"command set. Note 10: Before "CH" command set, you have to set the "SETCDMABAND" Command. Note 11: After "TCH" command set, you have to set "TXPDM" command. Note 12: Using by "TXPDM" command, you can get the desired TX power.
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
EXAMPLE	Recommend test sequence is below : - AT#TESTMODE="TM" - AT#TESTMODE="SETCDMABAND 0' - AT#TESTMODE="CH 384" - AT#TESTMODE="TCH' - AT#TESTMODE="TXPDM 100" - AT#TESTMODE="ESC" - AT#TESTMODE="OM"

3.5.6.1.47. *CMUX Mode Set - #CMUXMODE*

#CMUXMODE – CMUX Mode Set

AT#CMUXMODE= <mode>	Set command specifices the CMUX mode Parameters: **<mode>** multiplexer transparency mechanism 0 - Ignore DTR feature is disabled, a transmission of the physical DTR line instructs the DCE to disable the CMUX and switches to the normal command mode**. (default)** 1 – Ignore DTR feature is disabled, a transmission of the physical DTR line instructs the DCE to disable the CMUX and switches to the normal command mode 5 – Ignore DTR feature is enabled, the DCE doesn't care the physical DTR line transitions Note : DLC establishment on Virtual Channel between mode 0 and mode 1 is different. See HE920 Serial Interface User Guide. Note : a software or hardware reset restores the default value. Note : during cmux session the set command will return **ERROR,** only the read and test command can be used.
AT#CMUXMODE?	Read command returns the current value of **<mode> parameter.** **+CMUXMODE: <mode>**
AT#CMUXMODE=?	Test command returns the range of supported values for parameter **<mode>**

3.5.6.1.48.   *RSSI Configuration - #RSSICFG*

#RSSICFG – RSSI configuration	

AT#RSSICFG=
<method>
	Set command sets RSSI configuration.
	
	Parameter:
	<method>
	  0 – RSSI calculated as Rx power (i.e. received signal strength).
	      Default for CE910-DUAL Verizon, Sprint and US Cellular.
	  1 – RSSI calculated as Rx power + Ec/Io, where "Ec" represents
	      the average energy per pseudonoise chip for the pilot channel
	      and "Io" represents the total received power spectral density,
	      Including signal and interference.
	      Default for CE910-SC and for CE910-DUAL Aeris.
	
	Note: The value is stored in NVM.

AT#RSSICFG?	Read command returns current **RSSICFG** in the format: **#RSSICFG: <method>**

AT#RSSICFG=?	Test command returns the supported range of values for parameter **<method>**.
Example	If Call box setting is below - Cell power : -50dBm , Pilot strength:-7dBm AT#RSSICFG=0 OK AT#CAI? #CAI: 4376,30,522,30,350,330,6,6,6,**-50**,-5,0,2,0,0,0,2,0,0,1 AT+CSQ +CSQ: 31,99 AT#RSSICFG=1 OK AT#CAI? #CAI: 4376,30,522,30,350,330,6,6,6,**-57**,-7,0,2,0,0,0,2,0,0,1 AT+CSQ +CSQ: 28,99

3.5.6.1.49. *Report RSSI - #RIND*

#RIND – Report RSSI(Received Signal Strength Indicaton)	

AT#RIND=
<mode>
	Set command enables / disables RSSI reports depending on the parameter **<mode>.**
	
	Parameter:
	< mode >
	  0 – Disable RSSI unsolicited report. (Factory default).
	  1 – Enable RSSI unsolicited report whenever the value of **<rssi_ind>**
	    changes.
	
	  #RIND: <rssi_ind >
	  Where:
	   < rssi_ind>- Received signal strength indication.
	    0 - signal strength ≤ (-112) dBm
	    1..4 - signal strength in 15 dBm steps
	    5 - signal strength ≥ (-51) dBm
	    99 - not measurable
	
	<table><tr><td>**Received signal strength (Rx power + Ec/Io) (dBm)**</td><td>**<rssi_ind>**</td></tr><tr><td>-112dBm or less</td><td>0</td></tr><tr><td>-111dBm ~ -97dBm</td><td>1</td></tr><tr><td>-96dBm ~ -82dBm</td><td>2</td></tr></table>

	-81dBm ~ -67dBm	3
	-66dBm ~ -52dBm	4
	-51dBm or greater	5
	Note: The value is stored in NVM. Note: Received signal strength (dBm) = Rx power + Ec/Io.	
AT#RIND?	Read command returns current **RIND** in the format: # **RIND: < mode >**	
AT#RIND =?	Test command returns the supported range of values for parameter **<mode>**.	
Example	AT#RIND=1 OK AT#RIND? #RIND: 1 AT#CAI? #CAI: 4376,30,522,30,350,330,6,6,6,**-50**,-5,0,2,0,0,0,2,0,0,1 ->Change the RX level AT#CAI? #CAI: 4376,30,522,30,350,330,6,6,6,**-110**,-5,0,2,0,0,0,2,0,0,1 # RIND: 1 (report whenever the value of **<rssi_ind>** changes)	

3.5.6.2. Audio AT Commands

3.5.6.2.1. *Change Audio Path – #CAP*

#CAP - Change Audio Path	
AT#CAP=<n>	It has no effect and is included only for backward compatibility. Parameter: **<n>**: (0-2)
AT#CAP?	Read command reports the active audio path in the format: **#CAP: <n>**.
AT#CAP=?	Test command reports the supported values for the parameter **<n>**.

3.5.6.2.2. *Open Audio Loop – #OAP*

#OAP – Open Audio Loop

AT#OAP= \<mode>	Set command sets Open Audio Path. Parameter: 0 - disables Open Audio Path (factory default) 1 - enables Open Audio Path *Note: This parameter is not saved in NVM*
AT#OAP?	Read command returns the current Open Audio Path, in the format: **#OAP: \<mode>**
AT#OAP =?	Test command returns the supported range of values of parameter **\<mode>**.

3.5.6.2.3. *Select Ringer Sound – #SRS*

#SRS - Select Ringer Sound	

AT#SRS=
[\<n>,\<tout>]
	Set command sets the ringer sound.
	
	Parameters:
	\<n> - ringing tone
	  0 - current ringing tone
	  1..*max* - ringing tone number, where *max* can be read by issuing the Test
	      command **AT#SRS=?**.
	  10 - factory default
	\<tout> - ringing tone playing time-out in seconds.
	 0 - ringer is stopped (if present) and current ringer sound is set.(factory default)
	 1..60 - ringer sound playing for **\<tout>** seconds and, if **\<n> > 0,** ringer sound **\<n>**
	      is set as default ringer sound.
	
	Note: when the command is issued with **\<n> > 0** and **\<tout> > 0**, the **\<n>** ringing
	tone is played for **\<tout>** seconds and stored as default ringing tone.
	
	Note: if command is issued with **\<n> > 0** and **\<tout> = 0**, the playing of the ringing
	is stopped (if present) and **\<n>** ringing tone is set as current.
	
	Note: if command is issued with **\<n> = 0** and **\<tout> > 0** then the current ringing
	tone is played.
	
	Note: if both **\<n>** and **\<tout>** are **0** then the default ringing tone is set as current
	and ringing is stopped.
	
	Note: If all parameters are omitted then the behaviour of Set command is the same
	as Read command

AT#SRS?	Read command reports current selected ringing and its status in the form: **#SRS: \<n>,\<status>** where: **\<n>** - ringing tone number

#SRS - Select Ringer Sound	

	1..*max*
	 <status> - ringing status
	   0 - selected but not playing
	   1 - currently playing

AT#SRS=?	Test command reports the supported values for the parameters **<n>** and **<tout>**

3.5.6.2.4. *Select Ringer Path - #SRP*

#SRP - Select Ringer Path	

AT#SRP=<n>	Set command selects the ringer path towards whom sending ringer sounds and all signalling tones.
	
	 Parameter:
	 <n> - ringer path number
	   0 - sound output towards current selected audio path (see command
	     #CAP) (factory default)
	   1 - sound output towards handsfree
	   2 - sound output towards handset

AT#SRP?	Read command reports the set value of the parameter <n> in the format: **#SRP: <n>**.
AT#SRP=?	Test command reports the supported values for the parameter **<n>**.
Example	AT#SRP=? #SRP: (0-3) OK AT#SRP=3 OK

3.5.6.2.5. *Signalling Tones Mode - #STM*

#STM - Signalling Tones Mode	

AT#STM=
 <mode>
	Set command enables/disables the signalling tones output on the audio path selected with **#SRP** command
	
	 Parameter:
	 <mode> - signalling tones status
	   0 - signalling tones disabled
	   1 - signalling tones enabled (factory default)
	   2 – all tones disabled
	
	 Note: **AT#STM=0** has the same effect as **AT+CALM=2**; **AT#STM=1** has the same effect as AT+CALM=**0**.

AT#STM?	Read command reports whether the current signaling tones status is enabled or not,

#STM - Signalling Tones Mode	
	in the format: **#STM: <mode>**
AT#STM=?	Test command reports supported range of values for parameter **<mode>**.

3.5.6.2.6. *Tone Playback – #TONE*

#TONE - Tone Playback	

AT#TONE=<tone>
[,<duration>]
	Execution command allows the playback of either a single DTMF tone or a dial tone for a specified period of time.
	
	Parameters:
	<tone> - tone to be reproduced
	  (0-9), #, *, (A-D) - dtmf tone
	  (G-L) - user defined tones
	  Y - free tone
	  Z - busy tone
	<duration> - playback duration in 1/10 sec.
	  1..300 - tenth of seconds (default is 30)

AT#TONE=?	Test command returns the supported range of values for parameters **<tone>** and **<duration>**.

3.5.6.2.7. *Tone Classes Volume – #TSVOL*

#TSVOL – Tone Classes Volume	
AT#TSVOL= <class>, **<mode>** **[,<volume>]**	Set command is used to select the volume mode for one or more tone classes. Parameters: **<class>** -sum of integers each representing a class of tones which the command refers to 1 - CDMA tones 2 - ringer tones 4 - reserved 8 - reserved 16 - DTMF tones 64 - user defined tones 128 - Dial tones 255 - all classes **<mode>** - it indicates which volume e're using for the classes of tones represented by **<class>** 0 - we're using default volume 1 - we're using the volume **<volume>**. **<volume>** - volume to be applied to the set of classes of tones represented by **<class>**; it is mandatory if **<mode>** is **1**. 0..*max* - the value of *max* can be read issuing the Test command **AT#TSVOL=?**

#TSVOL – Tone Classes Volume	
AT#TSVOL?	Read command returns for each class of tones the last setting of **<mode>** and, if **<mode>** is not **0**, of **<volume>** too, in the format: **#TSVOL:1,<mode1>[,<volume1>]<CR><LF>** **…** **#TSVOL:128,<mode128>[,<volume128>]**
AT#TSVOL=?	Test command returns the supported range of values of parameters **<class>**, **<mode>** and **<volume>**.
Example	at#tsvol=84,1,5 OK at#tsvol? #TSVOL:1,0 #TSVOL:2,0 #TSVOL:4,1,5 #TSVOL:8,0 #TSVOL:16,1,5 #TSVOL:32,0 #TSVOL:64,1,5 #TSVOL:128,0 OK

3.5.6.2.8. *Embedded DTMF decoder enabling - #DTMF*

#DTMF – Embedded DTMF decoder enabling	SELINT 2
AT#DTMF=<mode>	Set command enables/disables the embedded DTMF decoder. Parameters: **<mode>:** 0 – disable DTMF decoder (default) 1 – enables DTMF decoder 2 – enables DTMF decoder without URC notify Note: if **<mode>**=1, the receiving of a DTMF tone is pointed out with an unsolicited message through AT interface in the following format: #DTMFEV: x with x as the DTMF digit Note: the duration of a tone should be not less than 50ms. Note: the value set by command is not saved and a software or hardware reset restores the default value. The value can be stored in NVM using profiles. Note: When DTMF decoder is enabled, PCM playing and recording are automatically disabled (AT#SPCM will return error).

AT#DTMF?	Read command reports the currently selected **<mode>** in the format: **#DTMF: <mode>**
AT#DTMF =?	Test command reports supported range of values for all parameters.

3.5.6.2.9. *Digital Voiceband Interface - #DVI*

#DVI - Digital Voiceband Interface	
AT#DVI=<mode> [,<dviport>, <clockmode>]	Set command enables/disables the Digital Voiceband Interface. Parameters: **<mode>** - enables/disables the DVI. 0 - disable DVI; audio is forwarded to the analog line; DVI pins can be used for other purposes, like GPIO, etc. (factory default) 1 - enable DVI; audio is forwarded to the DVI block **<dviport>** 2 - DVI port 2 will be used(factory default) **<clockmode>** 0 - DVI slave 1 - DVI master (factory default) Note: **#DVI** parameters are saved in the extended profile
AT#DVI?	Read command reports last setting, in the format: **#DVI: <mode>,<dviport>,<clockmode>**
AT#DVI=?	Test command reports the range of supported values for parameters **<mode>,<dviport>** and **<clockmode>**
Example	AT#DVI=1,2,1 OK *DVI activated for Digital audio.* *DVI is configured as master providing on DVI Port #2*

3.5.6.2.10. *Digital Voiceband Interface Configuration - #DVICFG*

#DVICFG – DVI CONFIGURATION	
AT#DVICFG=[<clock>[,<decoder pad>[,<decoder format>[, <encoder pad>[,<encoder format>]]]]]	Set command sets the DVI configuration Parameter: **<clock>**: Clock speed for master mode 　　0 : normal mode 　　1 : high speed mode(factory default) **<decoder pad>**: PCM padding enable in decoder path

#DVICFG – DVI CONFIGURATION	

	0 : disable
	1 : enable(factory default)
	<decoder format>: PCM format in decoder path
	  0 : u-Law(factory default)
	  1 : A-Law
	  2 : linear
	<encoder pad>: PCM padding enable in encoder path
	  0 : disable
	  1 : enable(factory default)
	<encoder format>: PCM format in encoder path
	  0 : u-Law(factory default)
	  1 : A-Law
	  2 : linear
	
	Note: #DVICFG parameters are saved in the extended profile.
	Note: Normal mode in <clock> is supported in DVI master

AT#DVICFG?	Read command reports the value of parameter in the format: **#DVICFG: <clock>,<decoder pad>,<decoder format>, <encoder pad>,<encoder format>**
AT#DVICFG=?	Test command returns the supported range of values of parameter **<clock>,<decoder pad>,<decoder format>, <encoder pad>,<encoder format>**.

3.5.6.2.11. *AXE Pin Reading - #AXE*

#AXE - AXE Pin Reading	
AT#AXE	It has no effect and is included only for backward compatibility.
AT#AXE=?	Test command returns the **OK** result code.

3.5.6.2.12. *Handsfree Echo Canceller - #SHFEC*

#SHFEC - Handsfree Echo Canceller	
AT#SHFEC= <mode>	Set command enables/disables the echo canceller function on audio handsfree output. Parameter: **<mode>** 0 - disables echo canceller for handsfree mode (factory default) 1 - enables echo canceller for handsfree mode *Note: This parameter is saved in NVM issuing AT&W command.*
AT#SHFEC?	Read command reports the value of parameter <mode>, in the format:

#SHFEC - Handsfree Echo Canceller	
	#SHFEC: <mode>
AT#SHFEC=?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.13. *Handsfree Microphone Gain - #HFMICG*

#HFMICG - Handsfree Microphone Gain	
AT#HFMICG= <level>	Set command sets the handsfree microphone input gain Parameter: **<level>**: handsfree microphone input gain (factory default : 4) 0..7 - handsfree microphone gain (+6dB/step)
AT#HFMICG?	Read command returns the current handsfree microphone input gain, in the format: **#HFMICG: <level>**
AT#HFMICG=?	Test command returns the supported range of values of parameter **<level>**.

3.5.6.2.14. *Handset Microphone Gain - #HSMICG*

#HSMICG - Handset Microphone Gain	
AT#HSMICG= <level>	Set command sets the handset microphone input gain Parameter: **<level>**: handset microphone input gain (factory default : 4) 0..7 - handset microphone gain (+6dB/step)
AT#HSMICG?	Read command returns the current handset microphone input gain, in the format: **#HSMICG: <level>**
AT#HSMICG=?	Test command returns the supported range of values of parameter **<level>**.

3.5.6.2.15. *Set Headset Sidetone - #SHFSD*

#SHFSD - Set Headset Sidetone	

AT#SHFSD=
<mode>
	Set command enables/disables the sidetone on handsfree audio output.
	
	Parameter:
	<mode>
	  0 - disables the handsfree sidetone (factory default)
	  1 - enables the handsfree sidetone
	
	Note: This parameter is saved in NVM issuing AT&W command.

AT#SHFSD?	Read command reports whether the handsfree sidetone is currently enabled or not, in the format:

#SHFSD - Set Headset Sidetone	
	#SHFSD: <mode>
AT#SHFSD=?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.16.   *Speaker Mute Control - #SPKMUT*

#SPKMUT - Speaker Mute Control	

AT#SPKMUT=<n>	Set command enables/disables the global muting of the speaker audio line, for every audio output (ring, incoming sms, voice, Network coverage)
	
	Parameter:
	<n>
	  0 - mute off, speaker active (factory default)
	  1 - mute on, speaker muted.
	
	Note: this command mutes/activates both speaker audio paths, internal speaker and external speaker.

AT#SPKMUT?	Read command reports whether the muting of the speaker audio line during a voice call is enabled or not, in the format: **#SPKMUT: <n>**
AT#SPKMUT=?	Test command reports the supported values for **<n>** parameter.

3.5.6.2.17.   *Handsfree Receiver Gain - #HFRECG*

#HFRECG - Handsfree Receiver Gain	
AT#HFRECG= **<level>**	Set command sets the handsfree analogue output gain Parameter: **<level>**: handsfree analogue output gain (factory default : 0) 0..6 - handsfree analogue output (-3dB/step) *Note: This parameter is saved in NVM issuing AT&W command.*
AT#HFRECG?	Read command returns the current value of parameter <level>, in the format: **#HFRECG: <level>**
AT#HFRECG =?	Test command returns the supported range of values of parameter **<level>**.

3.5.6.2.18.   *Handset Receiver Gain - #HSRECG*

#HSRECG - Handset Receiver Gain	
AT#HSRECG= **<level>**	Set command sets the handset analogue output gain Parameter: **<level>**: handset analogue output gain (factory default : 0)

	0..6 - handset analogue output (-3dB/step) *Note: This parameter is saved in NVM issuing AT&W command.*
AT#HSRECG?	Read command returns the current handset analog output gain, in the format: **#HSRECG: <level>**
AT#HSRECG =?	Test command returns the supported range of values of parameter **<level>**.

3.5.6.2.19. *Audio Profile Factory Configuration - #PRST*

#PRST - Audio Profile Factory Configuration	

AT#PRST	Execution command resets the actual audio parameters in the NVM of the device to the default set. It is not allowed if active audio profile is 0.
	The audio parameters to reset are:
	
	    - microphone line gain
	    - earpiece line gain
	    - side tone gain
	    - LMS adaptation speed (step size)
	    - LMS filter length (number of coefficients)
	    - speaker to micro signal power relation
	    - noise reduction max attenuation
	    - noise reduction weighting factor (band 300-500Hz)
	    - noise reduction weighting factor (band 500-4000Hz)
	    - AGC Additional attenuation
	    - AGC minimal attenuation
	    - AGC maximal attenuation

AT#PRST=?	Test command returns the **OK** result code.
Example	AT#PRST OK *Current audio profile is reset*

3.5.6.2.20. *Audio Profile Configuration Save – #PSAV*

#PSAV - Audio Profile Configuration Save	

AT#PSAV	Execution command saves the actual audio parameters in the NVM of the device.
	It is not allowed if active audio profile is 0.
	
	The audio parameters to store are:
	
	    - microphone line gain
	    - earpiece line gain
	    - side tone gain
	    - LMS adaptation speed
	    - LMS filter length (number of coefficients)
	    - speaker to micro signal power relation
	    - noise reduction max attenuation

#PSAV - Audio Profile Configuration Save	
	- noise reduction weighting factor (band 300-500Hz) - noise reduction weighting factor (band 500-4000Hz) - AGC Additional attenuation - AGC minimal attenuation - AGC maximal attenuation
AT#PSAV=?	Test command returns the **OK** result code.
Example	AT#PSAV OK *Current audio profile is saved in NVM*

3.5.6.2.21. *Audio Profile Selection – #PSEL*

#PSEL - Audio Profile Selection	

AT#PSEL=**\<prof\>**	Set command selects the active audio profile
	
	Parameter:
	\<prof\>: current profile
	  0 - standard profile(factory default)
	  1..3 - extended profile, modifiable.
	
	Note: This parameter is saved in NVM issuing AT&W command.

AT#PSEL?	The read command returns the active profile in the format: **#PSEL:\<prof\>**
AT#PSEL=?	Test command returns the supported range of values of parameter **\<prof\>**.

3.5.6.2.22. *Audio Profile Setting – #PSET*

#PSET - Audio Profile Setting	
AT#PSET= **\<scal _in\>** [,**\<scal _out\>** [,**\<side_tone_atten\>** [,**\<adaption_speed\>** [,**\<filter_length\>** [,**\<rxtxrelation\>** [,**\<nr_atten\>** [,**\<nr_w_0\>** [,**\<nr_w_1\>** [,**\<add_atten\>** [,**\<min_atten\>** [,**\<max_atten\>**]]]]]]]]]]]]	Set command sets parameters for the active audio profile. It is not allowed if active audio profile is 0. Parameters: **\<scal_in\>** - microphone line digital gain **\<scal_out\>** - earpiece line digital gain **\<side_tone_atten\>** - side tone attenuation **\<adaption_speed\>** - LMS adaptation speed **\<filter_length\>** - LMS filter length (number of coefficients) **\<rxtxrelation\>** - speaker to micro signal power relation**(unused)** **\<nr_ atten \>** - noise reduction max attenuation**(unused)** **\<nr_w_0\>** - noise reduction weighting factor (band 300-500Hz) **(unused)** **\<nr_w_1\>** - noise reduction weighting factor (band 500-4000Hz) **(unused)** **\<add_atten\>** - AGC Additional attenuation**(unused)** **\<min_atten\>** - AGC minimal attenuation**(unused)** **\<max_atten\>** - AGC maximal attenuation**(unused)**

#PSET - Audio Profile Setting	
AT#PSET?	Read command returns the parameters for the active profile in the format: **#PSET:\<scal_in\>,\<scal_out\>,\<side_tone_atten\>,\<adaption_speed\>, \<filter_length\>,\<rxtxrelation\>,\<nr_atten\>,\<nr_w_0\>,\<nr_w_1\>, \<add_atten\>,\<min_atten\>,\<max_atten\>** It is not allowed if active audio profile is 0.
AT#PSET=?	Test command returns the supported range of values for the audio parameters.

3.5.6.2.23. *Handsfree Automatic Gain Control - #SHFAGC*

#SHFAGC - Handsfree Automatic Gain Control	
AT#SHFAGC = \<mode\>	Set command enables/disables the automatic gain control function on audio handsfree input. Parameter: **\<mode\>** 0 - disables automatic gain control for handsfree mode (factory default) 1 - enables automatic gain control for handsfree mode *Note: This parameter is saved in NVM issuing AT&W command.*
AT#SHFAGC?	Read command reports whether the automatic gain control function on audio handsfree input is currently enabled or not, in the format: **#SHFAGC: \<mode\>**
AT#SHFAGC =?	Test command returns the supported range of values of parameter **\<mode\>**.

3.5.6.2.24. *Handsfree Noise Reduction - #SHFNR*

#SHFNR - Handsfree Noise Reduction	
AT#SHFNR = \<mode\>	Set command enables/disables the noise reduction function on audio handsfree input. Parameter: **\<mode\>** 0 - disables noise reduction for handsfree mode (factory default) 1 - enables noise reduction for handsfree mode *Note: This parameter is saved in NVM issuing AT&W command.*
AT#SHFNR?	Read command reports whether the noise reduction function on audio Handsfree input is currently enabled or not, in the format: **#SHFNR: \<mode\>**
AT#SHFNR =?	Test command returns the supported range of values of parameter **\<mode\>**.

3.5.6.2.25. *Handset Automatic Gain Control - #SHSAGC*

#SHSAGC - Handset Automatic Gain Control	
AT#SHSAGC = <mode>	Set command enables/disables the automatic gain control function on audio handset input. Parameter: **<mode>** 0 - disables automatic gain control for handset mode (factory default) 1 - enables automatic gain control for handset mode *Note: This parameter is saved in NVM issuing AT&W command.*
AT#SHSAGC?	Read command reports whether the automatic gain control function on audio handset input is currently enabled or not, in the format: **#SHSAGC: <mode>**
AT#SHSAGC =?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.26. *Handset Echo Canceller - #SHSEC*

#SHSEC - Handset Echo Canceller	
AT#SHSEC = <mode>	Set command enables/disables the echo canceller function on audio handset output. Parameter: **<mode>** 0 - disables echo canceller for handset mode (factory default) 1 - enables echo canceller for handset mode *Note: This parameter is saved in NVM issuing AT&W command.*
AT#SHSEC?	Read command reports whether the echo canceller function on audio handset output is currently enabled or not, in the format: **#SHSEC: <mode>**
AT#SHSEC =?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.27. *Handset Noise Reduction - #SHSNR*

#SHSNR - Handset Noise Reduction	
AT#SHSNR = <mode>	Set command enables/disables the noise reduction function on audio handset input. Parameter: **<mode>** 0 - disables noise reduction for handset mode (factory default) 1 - enables noise reduction for handset mode

#SHSNR - Handset Noise Reduction	
	Note: This parameter is saved in NVM issuing AT&W command.
AT#SHSNR?	Read command reports whether the noise reduction function on audio handset input is currently enabled or not, in the format: **#SHSNR: <mode>**
AT#SHSNR =?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.28.   *Set Handset Sidetone - #SHSSD*

#SHSSD - Set Handset Sidetone	

AT#SHSSD=
<mode>
	Set command enables/disables the sidetone on handset audio output.
	
	Parameter:
	<mode>
	  0 - disables the handset sidetone (factory default)
	  1 - enables the handset sidetone
	Note: This parameter is saved in NVM issuing AT&W command.

AT#SHSSD?	Read command reports whether the headset sidetone is currently enabled or not, in the format: **#SHSSD: <mode>**
AT#SHSSD=?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.29.   *DVI Microphone Gain - #PCMTXG*

#PCMTXG – DVI Microphone Gain	
AT#PCMTXG=<TX_VOL>	Set command sets the DVI (PCM) Audio TX gain Parameter: **<TX_VOL>** : PCM TX volume in TX path (factory default : 0) TX VOL RANGE : -5000(-50 dB) ~ 1200(+12 dB) Note: meaning of a TX_VOL is 1/100 dB step. Note: meaning of -50 dB is mute
AT#PCMTXG?	Read command returns the current PCM Audio TX value: **#PCMTXG: <TX_VOL>**
AT#PCMTXG=?	Test command returns the supported range of values of parameter **<TX_VOL>**

3.5.6.2.30.   *DVI Speaker Volume Level - #PCMRXG*

#PCMRXG – DVI Speaker Volume Level

#PCMRXG – DVI Speaker Volume Level	
AT#PCMRXG=<RX_VOL>	Set command sets the PCM Audio RX value Parameter: **<RX_VOL>** : PCM RX volume in RX path (factory default : 0) RX_VOL RANGE : -5000(-50 dB) ~ 1200(+12 dB) Note: meaning of a RX_VOL is 1/100 dB step. Note: meaning of -50 dB is mute
AT#PCMRXG?	Read command returns the current PCM Audio RX value: **#PCMRXG: <RX VOL>**
AT#PCMRXG=?	Test command returns the supported range of values of parameter **<RX VOL>**

3.5.6.2.31.   *Handsfree RX AGC Value tuning - #SHFAGCRX*

#SHFAGCRX – Handsfree RX AGC Value tuning	

AT#SHFAGCRX= <agc_static_gain>,<agc_aig>, <agc_exp_thres>,<agc_exp_slope>, <agc_compr_thres>, <agc_compr_slope>	Set command sets the handsfree RX AGC value tuning
	
	Parameter:
	<agc_static_gain>
	  precompressor static gain. This is the gain applied to the input samples when static gain is enabled. Meaningful Range is 0x2000 to 0xFFFF.
	  Value(agc_static_gain) = $8192 * 10^{\wedge}(X/20)$: X range is 0 to 18 dB.
	
	<agc_aig>
	  pre-compressor gain selection flag. Write 0xFFFF to enable adaptive gain (static gain disabled). Write 0x0000 to enable static gain (adaptive gain disabled). Meaningful value is just 0x0000 or 0xFFFF.
	
	<agc_exp_thres>
	  expansion threshold. This is the energy threshold of the input under which expansion is applied. This parameter must be less than agc_compr_thres. Meaningful range is 0x0 to 0x2580. Write 0 to disable the expander.
	  Value(agc_exp_thres) = $128 * (X+75)$: X range is -75 to 0 dBm0mu
	
	<agc_exp_slope>
	  expansion slope. This is the slope of the expander gain when expansion is applied. Meaningful range is 0xFF01 to 0xFFF6.
	  Value(agc_exp_slope) = $256 * X$: X range is -0.04 to -0.996.
	
	<agc_compr_thres>
	  compression threshold. This is the energy threshold of the input above which compression is applied. Meaningful range is 0x0 to 0x2580. This parameter must be greater than agc_exp_thres.
	  Value(agc_compr_thres) = $128 * (X+75)$: X range is -75 to 0 dBm0mu
	
	<agc_compr_slope>
	  compression slope. This is the slope of the compressor gain when compression is applied. Meaningful range is 0x8000 to 0xFFFF.
	  Value(agc_compr_sloop) = $65536 * X$: X range is 0.50001 to 0.99999
	
	Note: these values are automatically saved in NVM.

AT#SHFAGCRX?	Read command returns the current values **#SHFAGCRX:** **<agc_static_gain>,<agc_aig>,<agc_exp_thres>,<agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**
AT#SHFAGCRX =?	Test command returns the supported range of values of parameter **<agc_static_gain>,<agc_aig>,<agc_exp_thres>,<agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**

3.5.6.2.32.   *Handsfree TX AGC Value tuning - #SHFAGCTX*

#SHFAGCTX – Handsfree TX AGC Value tuning	

AT#SHFAGCTX=<agc_static_gain>,<agc_aig>, <agc_exp_thres>,<agc_exp_slope>, <agc_compr_thres>, <agc_compr_slope>	Set command sets the handsfree TX AGC value tuning
	
	Parameter:
	<agc_static_gain>
	   precompressor static gain. This is the gain applied to the input samples when static gain is enabled. Meaningful Range is 0x2000 to 0xFFFF.
	   Value(agc_static_gain) = 8192 ∗ 10^(X/20) : X range is 0 to 18 dB.
	
	<agc_aig>
	   pre-compressor gain selection flag. Write 0xFFFF to enable adaptive gain (static gain disabled). Write 0x0000 to enable static gain (adaptive gain disabled). Meaningful value is just 0x0000 or 0xFFFF.
	
	<agc_exp_thres>
	   expansion threshold. This is the energy threshold of the input under which expansion is applied. This parameter must be less than agc_compr_thres.
	   Meaningful range is 0x0 to 0x2580. Write 0 to disable the expander.
	   Value(agc_exp_thres) = 128 ∗ (X+75) : X range is -75 to 0 dBm0mu
	
	<agc_exp_slope>
	   expansion slope. This is the slope of the expander gain when expansion is applied. Meaningful range is 0xFF01 to 0xFFF6.
	   Value(agc_exp_slope) = 256 * X : X range is -0.04 to -0.996.
	
	<agc_compr_thres>
	   compression threshold. This is the energy threshold of the input above which compression is applied. Meaningful range is 0x0 to 0x2580. This parameter must be greater than agc_exp_thres.
	   Value(agc_compr_thres) = 128 ∗ (X+75) : X range is -75 to 0 dBm0mu
	
	<agc_compr_slope>
	   compression slope. This is the slope of the compressor gain when compression is applied. Meaningful range is 0x8000 to 0xFFFF.
	   Value(agc_compr_sloop) = 65536 * X : X range is 0.50001 to 0.99999
	
	Note: these values are automatically saved in NVM.

AT#SHFAGCTX?	Read command returns the current values **#SHFAGCTX: <agc_static_gain>,<agc_aig>,<agc_exp_thres>,<agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**
AT#SHFAGCTX =?	Test command returns the supported range of values of parameter **<agc_static_gain>,<agc_aig>,<agc_exp_thres>,<agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**

3.5.6.2.33.   *Handset RX AGC Value tuning – #SHSAGCRX*

#SHSAGCRX – Handset RX AGC Value tuning	
AT#SHSAGCRX= <agc_static_gain>,<agc_aig>, <agc_exp_thres>,<agc_exp_slope>, <agc_compr_thres>, <agc_compr_slope>	Set command sets the handset RX AGC value tuning Parameter: **<agc_static_gain>** precompressor static gain. This is the gain applied to the input samples when static gain is enabled. Meaningful Range is 0x2000 to 0xFFFF. Value(agc_static_gain) = $8192 * 10^{(X/20)}$: X range is 0 to 18 dB. **<agc_aig>** pre-compressor gain selection flag. Write 0xFFFF to enable adaptive gain (static gain disabled). Write 0x0000 to enable static gain (adaptive gain disabled). Meaningful value is just 0x0000 or 0xFFFF. **<agc_exp_thres>** expansion threshold. This is the energy threshold of the input under which expansion is applied. This parameter must be less than agc_compr_thres. Meaningful range is 0x0 to 0x2580. Write 0 to disable the expander. Value(agc_exp_thres) = $128 * (X+75)$: X range is -75 to 0 dBm0mu **<agc_exp_slope>** expansion slope. This is the slope of the expander gain when expansion is applied. Meaningful range is 0xFF01 to 0xFFF6. Value(agc_exp_slope) = 256 * X : X range is -0.04 to -0.996. **<agc_compr_thres>** compression threshold. This is the energy threshold of the input above which compression is applied. Meaningful range is 0x0 to 0x2580. This parameter must be greater than agc_exp_thres. Value(agc_compr_thres) = $128 * (X+75)$: X range is -75 to 0 dBm0mu **<agc_compr_slope>** compression slope. This is the slope of the compressor gain when compression is applied. Meaningful range is 0x8000 to 0xFFFF. Value(agc_compr_sloop) = 65536 * X : X range is 0.50001 to 0.99999 Note: these values are automatically saved in NVM.
AT#SHSAGCRX?	Read command returns the current handset RX AGC values **#SHSAGCTX: <agc_static_gain>,<agc_aig>,<agc_exp_thres>, <agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**
AT#SHSAGCRX =?	Test command returns the supported range of values of parameter **<agc_static_gain>,<agc_aig>,<agc_exp_thres>,<agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**

3.5.6.2.34. *Handset TX AGC Value tuning - #SHSAGCTX*

#SHSAGCTX – Handset TX AGC Value tuning	

AT#SHSAGCTX= <agc_static_gain>,<agc_aig>, <agc_exp_thres>,<agc_exp_slope>, <agc_compr_thres>, <agc_compr_slope>	Set command sets the handset TX AGC value tuning
	
	Parameter:
	<agc_static_gain>
	    precompressor static gain. This is the gain applied to the input samples when static gain is enabled. Meaningful Range is 0x2000 to 0xFFFF.
	    Value(agc_static_gain) = 8192 ∗ 10^(X/20) : X range is 0 to 18 dB.
	
	<agc_aig>
	    pre-compressor gain selection flag. Write 0xFFFF to enable adaptive gain (static gain disabled). Write 0x0000 to enable static gain (adaptive gain disabled). Meaningful value is just 0x0000 or 0xFFFF.
	
	<agc_exp_thres>
	    expansion threshold. This is the energy threshold of the input under which expansion is applied. This parameter must be less than agc_compr_thres. Meaningful range is 0x0 to 0x2580. Write 0 to disable the expander.
	    Value(agc_exp_thres) = 128 ∗ (X+75) : X range is -75 to 0 dBm0mu
	
	<agc_exp_slope>
	    expansion slope. This is the slope of the expander gain when expansion is applied. Meaningful range is 0xFF01 to 0xFFF6.
	    Value(agc_exp_slope) = 256 * X : X range is -0.04 to -0.996.
	
	<agc_compr_thres>
	    compression threshold. This is the energy threshold of the input above which compression is applied. Meaningful range is 0x0 to 0x2580. This parameter must be greater than agc_exp_thres.
	    Value(agc_compr_thres) = 128 ∗ (X+75) : X range is -75 to 0 dBm0mu
	
	<agc_compr_slope>
	    compression slope. This is the slope of the compressor gain when compression is applied. Meaningful range is 0x8000 to 0xFFFF.
	    Value(agc_compr_sloop) = 65536 * X : X range is 0.50001 to 0.99999
	
	Note: these values are automatically saved in NVM.

AT#SHSAGCTX?	Read command returns the current handset TX AGC values **#SHSAGCTX: <agc_static_gain>,<agc_aig>,<agc_exp_thres>, <agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**
AT#SHSAGCTX =?	Test command returns the supported range of values of parameter **<agc_static_gain>,<agc_aig>,<agc_exp_thres>,<agc_exp_slope>,<agc_compr_thres>,<agc_compr_slope>**

3.5.6.2.35. **RX AGC enable - #SRXAGC**

#SRXAGC – RX AGC Enable	
AT#SRXAGC= **<mode>**	Set command sets the RX AGC enabling Parameter: **<mode>** 0 - disables rx agc (factory default) 1 - enables rx agc Note: RX AGC enabling makes RX level decreasing Note: these values are automatically saved in NVM.
AT#SRXAGC?	Read command returns the current RX AGC values: **#SRXAGC: <mode>**
AT#SRXAGC=?	Test command returns the supported range of values of parameter **<mode>**.

3.5.6.2.36. **Handset RX filter coefficients values - #SHSFRX**

#SHSFRX - Handset RX filter coefficients values	
AT#SHSFRX= **<tap0>,<tap1>,<tap2>,<tap3>,** **<tap4>,<tap5>,<tap6>**	It has no effect and is included only for backward compatibility. Parameter: **<tap0>** **<tap1>** **<tap2>** **<tap3>** **<tap4>** **<tap5>** **<tap6>** Note: these values are automatically saved in NVM.
AT#SHSFRX?	Read command returns the current values: **#SHSFRX: <tap0>,<tap1>,<tap2>,<tap3>,<tap4>,<tap5>,<tap6>**
AT#SHSFRX=?	Test command returns the supported range of values of parameter **<tap0>,<** **tap1>,<tap2>,<tap3>,<tap4>,<tap5>,<tap6>**.

3.5.6.2.37. **Handset TX filter coefficients values - #SHSFTX**

#SHSFTX - Handset TX filter coefficients values	
AT#SHSFTX= **<tap0>,<tap1>,<tap2>,<tap3>,** **<tap4>,<tap5>,<tap6>**	Set command sets the handset TX filter coefficients values Parameter: **<tap0>**: Filter Tap, h[0] and h[12] **<tap1>**: Filter Tap, h[1] and h[11] **<tap2>**: Filter Tap, h[2] and h[10]

#SHSFTX - Handset TX filter coefficients values	
	\<tap3\>: Filter Tap, h[3] and h[9] **\<tap4\>**: Filter Tap, h[4] and h[8] **\<tap5\>**: Filter Tap, h[5] and h[7] **\<tap6\>**: Filter Tap, h[6] Note: these values are automatically saved in NVM.
AT#SHSFTX?	Read command returns the current handset TX filter coefficients values: **#SHSFTX: \<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>,\<tap4\>,\<tap5\>,\<tap6\>**
AT#SHSFTX=?	Test command returns the supported range of values of parameter **\<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>,\<tap4\>,\<tap5\>,\<tap6\>**.

3.5.6.2.38.   *Handsfree RX filter coefficients values - #SHFFRX*

#SHFFRX - Handsfree RX filter coefficients values	
AT#SHFFRX= \<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>, \<tap4\>,\<tap5\>,\<tap6\>	It has no effect and is included only for backward compatibility. Parameter: **\<tap0\>** **\<tap1\>** **\<tap2\>** **\<tap3\>** **\<tap4\>** **\<tap5\>** **\<tap6\>** Note: these values are automatically saved in NVM.
AT#SHFFRX?	Read command returns the current values: **#SHFFRX: \<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>,\<tap4\>,\<tap5\>,\<tap6\>**
AT#SHFFRX=?	Test command returns the supported range of values of parameter **\<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>,\<tap4\>,\<tap5\>,\<tap6\>**.

3.5.6.2.39.   *Handsfree TX filter coefficients values - #SHFFTX*

#SHFFTX - Handsfree TX filter coefficients values	
AT#SHFFTX= \<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>, \<tap4\>,\<tap5\>,\<tap6\>	Set command sets the handsfree TX filter coefficients values Parameter: **\<tap0\>**: Filter Tap, h[0] and h[12] **\<tap1\>**: Filter Tap, h[1] and h[11] **\<tap2\>**: Filter Tap, h[2] and h[10] **\<tap3\>**: Filter Tap, h[3] and h[9]

#SHFFTX - Handsfree TX filter coefficients values	
	\<tap4\>: Filter Tap, h[4] and h[8] **\<tap5\>**: Filter Tap, h[5] and h[7] **\<tap6\>**: Filter Tap, h[6] Note: these values are automatically saved in NVM.
AT#SHFFTX?	Read command returns the current handsfree TX filter coefficients values: **#SHFFTX: \<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>,\<tap4\>,\<tap5\>,\<tap6\>**
AT#SHFFTX=?	Test command returns the supported range of values of parameter **\<tap0\>,\<tap1\>,\<tap2\>,\<tap3\>,\<tap4\>,\<tap5\>,\<tap6\>**.

3.5.6.2.40.   *PCM Play and Receive - #SPCM*

#SPCM - PCM Play and Receive	

AT#SPCM=\<mode\>[, dir,[format]]	Execution command allows user either to send speech sample coming from microphone and/or downlink audio channel to serial port, or to reproduce a PCM coming from serial port to speaker and/or uplink audio channel; both modes are also available during speech calls.
	
	 Parameters:
	 \<mode\>: action to be execute;
	   1 - reproduce PCM stream from serial to selected path.
	   2 - send speech from selected path to serial.
	
	 \<dir\>: Select the audio path.
	   0 - send/receive to/from analog front end
	   1 - send/receive to/from audio channel
	   2 - reserved
	
	 \< format \>: PCM bits format
	   0 - 8 bit
	   1 - 16 bit
	
	 Note: 0 in \<format\> has no effect and is included only for backward compatibility and it works with Linear DVI configuration
	
	 Note: Execution command switches module in online mode. Module moves back to command mode either after entering the escape sequence +++ or as a consequence of a **DTR transition**.
	
	 Note: it is mandatory to set +IPR at least to 230400.
	
	 The following table summarizes the status of audio path during a speech call for different configurations and with sidetone disabled:
	
	 <table><tr><td></td><td>**mode = 1**</td><td>**mode = 2**</td></tr></table>

#SPCM - PCM Play and Receive			
	dir = 0	Uplink off / Downlink on PCM stream on speaker	Uplink off / Downlink off PCM stream from microphone
	dir = 1	Uplink on / Downlink off PCM stream on Uplink	Uplink off / Downlink on PCM stream from Downlink
	Sidetone is active for default.		
AT#SPCM=?	Test command returns the supported range of values for parameters **<mode>, <dir>** and **<format>**. **#SPCM: <mode>,<dir>,<format>**		
Example	AT#SPCM=1,0,0 CONNECT +++ NO CARRIER Note: after the CONNECT, 8Khz 8bit PCM stream has to be sent to serial port AT#SPCM=2,0,0 CONNECT +++ NO CARRIER Note: after the CONNECT, 8Khz 8bit PCM stream can be read from serial port		

3.5.6.3. Multisocket AT Commands

3.5.6.3.1. *Socket Status - #SS*

#SS - Socket Status	
AT#SS[=<connId>]	Execution command reports the current status of the sockets in the format: Parameters: <connId> - socket connection identifier 1..6 The response format is: **#SS: <connId>,<state>,<locIP>,<locPort>,<remIP>,<remPort>** where: **<connId>** - socket connection identifier, as before **<state>** - actual state of the socket:

#SS - Socket Status	

	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.
	<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.
	
	Note: issuing **#SS<CR>** causes getting information about status of all the sockets; the response format is:
	#SS: <connId1>,<state1>,<locIP1>,<locPort1>,<remIP1>,<remPort1>
	<CR><LF>
	…
	#SS: <connId6>,<state6>,<locIP6>,<locPort6>,<remIP6>,<remPort6>

AT#SS=?	Test command reports the range for parameter **<connId>**.
Example	AT#SS #SS: 1,3,91.80.90.162,61119,88.37.127.146,10510 #SS: 2,4,91.80.90.162,1000 #SS: 3,0 #SS: 4,0 #SS: 5,3,91.80.73.70,61120,88.37.127.146,10509 #SS: 6,0 OK Socket 1: opened from local IP 91.80.90.162/local port 61119 to remote IP 88.37.127.146/remote port 10510 is suspended with pending data Socket 2: listening on local IP 91.80.90.162/local port 1000 Socket 5: opened from local IP 91.80.73.70/local port 61120 to remote IP 88.37.127.146/remote port 10509 is suspended with pending data AT#SS=2 #SS: 2,4,91.80.90.162,1000

#SS - Socket Status	
	OK
	We have information only about socket number 2

3.5.6.3.2. *Socket Info – #SI*

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

#SI - Socket Info	
	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*

3.5.6.3.3. *Context Activation - #SGACT*

#SGACT - Context Activation	

AT#SGACT=<cid>, <stat>[,<userId>[, <pwd>]]	Execution command is used to activate or deactivate the specified PDP context.
	
	Parameters:
	<cid> - PDP context identifier
	  1 - numeric parameter which specifies a particular PDP context definition
	<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
	
	Note: In CDMA PDP context activation, Only one context ID(1) is supported.
	
	Note : <userId> and <pwd> are Don't Care parameters in North America carriers such as Verizon, Sprint and so on. (Because authentication information is automatically populated in a device based on the their specification and updated by a network through OTA or carrier's specific method.) **For more detail information, refer to #USERID and #PASSW command usage.**

AT#SGACT?	Returns the state of the contexts, in the format:
	
	#SGACT: <cid>,<Stat>
	
	where:
	<cid> - as **<cid>** before
	<stat> - context status
	  0 - context deactivated
	  1 - context activated

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

3.5.6.3.4. *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 **Registration** or after a **NW PDP CONTEXT deactivation** if al least one IPEasy socket is configured to this context (see AT#SCFG). Parameters: <**cid**> - PDP context identifier 1 – numeric parameter which specifies a particular PDP context definition <**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 aer 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 six contexts, in the format: **#SGACTCFG: <cid1>,<retry1>,<delay1>,<urcmode><CR><LF>**

#SGACTCFG – Context Activation and Configuration	
AT#SGACTCFG=?	Test command returns the range of supported values for parameters <**cid**>,<**retry**>,<**delay**> and <**urcmode**>

3.5.6.3.5. *Context Activation and Configuration Extended – #SGACTCFGEXT*

#SGACTCFGEXT – Context Activation and Configuration	
AT#SGACTCFGEXT= <**cid**>, <**abortAttemptEnable**>, [,<**unused**> [,<**unused**> [,<**unused**>]]]	Execution command is used to enable new features related to context activation. Parameters: <**cid**> - PDP context identifier 1 – numeric parameter which specifies a particular PDP context definition <**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 CDMA context activation attempt through #SGACT command in the following manner. While waiting for AT#SGACT=<cid>,1 response (up to 150s), it is possible to abort attempt by sending a byte and get back AT interface control (NO CARRIER indication). Note: values are automatically saved in NVM.
AT#SGACTCFGEXT?	Read command reports the state of all the six contexts, in the format: **#SGACTCFGEXT: <cid>,<abortAttemptEnable>,,0,0,0<CR><LF>**
AT#SGACTCFGEXT=?	Test command returns the range of supported values for parameters

3.5.6.3.6. *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) and incoming connection mode. Trying to close an active socket connection produces an error and to close a closed socket or a listening socket produces **OK** response without any action.

#SH - Socket Shutdown	
AT#SH=?	Test command reports the range for parameter **<connId>**.

3.5.6.3.7. *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> - PDP context identifier
	  1 - numeric parameter which specifies a particular PDP context definition
	<pktSz> - packet size to be used by the TCP/UDP/IP stack for data sending.
	  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
	  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.
	  *1*0..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.
	  0 - no timeout
	  1..255 - timeout value in hundreds of milliseconds (default 50)
	
	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: (1-6),(1),(0-1500),(0-65535),(10-1200),(0-255) OK AT#SCFG? #SCFG: 1,1,300,90,600,50

#SCFG - Socket Configuration	
	#SCFG: 2,1,300,90,600,50
	#SCFG: 3,1,300,90,600,50
	#SCFG: 4,1,300,90,600,50
	#SCFG: 5,1,300,90,600,50
	#SCFG: 6,1,300,90,600,50
	OK
	AT#SCFG=6,1,500,100,700,60
	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: 4,1,300,90,600,50
	#SCFG: 5,1,300,90,600,50
	#SCFG: 6,1,500,100,700,60
	OK

3.5.6.3.8. *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>
	   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
	3 – Data view with UDP datagram informations:

#SCFGEXT - Socket Configuration Extended	
	SRING : <sourceIP>,<sourcePort><connId>,<recData>, <dataLeft>,<data> same as before with <sourceIP>,<sourcePort> and <dataLeft> that means the number of bytes left in the UDP datagram <dataMode> - "data view mode" presentation format 0 - data represented as text (default) 1 - data represented as sequence of hexadecimal numbers (from 00 to FF) <keepalive> - TCP keepalive timer timeout 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> has effect only on TCP connections. Note: these values are automatically saved in NVM 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>, <unused_A1>,<unused_B1><CR><LF> ... #SCFGEXT: <connId6>,<srMode6>,<dataMode6>,<keepalive6> <unused_A6>,<unused_B6>
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 and a keepalive time of 30 minutes. Socket 3 set with data amount sring, hex data mode and no keepalive. AT#SCFGEXT? #SCFGEXT: 1,2,0,30,0,0 #SCFGEXT: 2,0,0,0,0,0 #SCFGEXT: 3,1,1,0,0,0

#SCFGEXT - Socket Configuration Extended	
	#SCFGEXT: 4,0,0,0,0,0 #SCFGEXT: 5,0,0,0,0,0 #SCFGEXT: 6,0,0,0,0,0 OK

3.5.6.3.9. *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 will be done when new data are 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 Note : is necessary to avoid overlapping of the two methods. Enabling new method, the old method for transmission timer (#SCFG) is 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 / #SKTD) 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 has been given to the AT interface, the ERROR message will be received on the interface itself.

#SCFGEXT2 - Socket Configuration Extended	
	<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>** are added For possible **<cause>** values, see also **#SLASTCLOSURE**

Note: like **#SLASTCLOSURE**, in case of subsequent consecutive closure causes are received, the original disconnection cause is 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.

Note : values are automatically saved in NVM. |
| **AT#SCFGEXT2?** | Read command returns the current socket extended configuration 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#SCFGEXT2=?** | Test command returns the range of supported values for all the subparameters |
| Example | AT#SCFGEXT2=1,1 OK

AT#SCFGEXT2=2,1 OK

AT#SCFGEXT2? #SCFGEXT2: 1,1,0,0,0,0 #SCFGEXT2: 2,1,0,0,0,0 |

#SCFGEXT2 - Socket Configuration Extended	
	#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.*

3.5.6.3.10.   *Show Address – #CGPADDR*

#CGPADDR – Show Address	
AT#CGPADDR=\<cid\>	Execution command returns the IP address for the current activated CDMA PDP context **\<cid\>** - context identifier Note : Only one context ID(1) is supported.
AT#CGPADDR=?	Returns \<cid\> when modem gets the IP address, otherwise returns only OK result
Example	AT#SGACT=1,1 +IP: xxx.yyy.zzz.www OK AT#CGPADDR=? #CGPADDR: (1) OK AT#CGPADDR=1

#CGPADDR – Show Address	
	#CGPADDR: 1,"xxx.yyy.zzz.www" OK

3.5.6.3.11.   *Socket Dial - #SD*

#SD - Socket Dial	
AT#SD=<connId>, <txProt>,<rPort>, <IPaddr> [,<closureType> [,<lPort> [,<connMode>]]]	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 **<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 (+++) and #SH **<lPort>** - UDP connections local port 1..65535 **<connMode>** - Connection mode 0 - online mode connection (default) 1 - command mode connection Note: **<closureType>** parameter is valid for TCP connections only and has no effect (if used) for UDP connections. If the socket connection is opened in CMD mode, <closureType> 255 does not take effect. 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>**.

#SD - Socket Dial	
	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**
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

3.5.6.3.12. *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.

3.5.6.3.13. *Socket Restore - #SO*

#SO - Socket Restore	

#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.

3.5.6.3.14. *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
	  1..65535
	<lingerT> - linger time
	  0 - immediate closure after remote closure
	  255 - local host closes only after an escape sequence (+++) and #SH
	
	Note : If the socket connection is opened in CMD mode, <closureType> 255 does not take effect.
	
	Note: if successful, commands returns a final result code **OK** . Then, when there's an incoming connection on the local port and if the sender is not filtered by internal firewall (see **#FRWL**), an URC is received:
	
	SRING : <connId>
	
	Note: the command **#SCFGEXT** doesn't influence the presentation format of the URC **SRING**
	
	Afterwards we can use **#SA** to accept the connection or **#SH** to refuse it.
	
	If the socket is closed by the network the following URC is received:
	
	#SL: ABORTED
	
	Note: when closing the listening socket <listenPort> is a Don't Care parameter.

AT#SL?	Read command returns all the actual listening sockets.

#SL - Socket Listen	
AT#SL=?	Test command returns the range of supported values for all the subparameters.
Example	AT#SL=? #SL: (1-6),(0,1),(1-65535),(0,255) OK Next command opens a socket listening on port 3500 AT#SL=1,1,3500 OK

3.5.6.3.15. *UDP SocketListen - #SLUDP*

#SLUDP – UDP Socket Listen	

AT#SLUDP=
<connId>,
<listenState>,
<listenPort>
	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
	  1..65535
	
	Note: if successful, commands returns a final result code **OK** . Then, when there's an incoming connection on the local port and if the sender is not filtered by internal firewall (see **#FRWL**), an URC is received:
	
	SRING : <connId>
	
	Afterwards it is possible to use **#SA** to accept the connection or **#SH** to refuse it. 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 sockets.
AT#SLUDP=?	Test command returns the range of supported values for all the subparameters.
Example	AT#SLUDP=? #SLUDP: (1-6),(0,1),(1-65535) OK

#SLUDP – UDP Socket Listen	
	Next command opens a socket listening on port 860 AT#SLUDP=1,1,860 OK SRING: 1 AT#SA=1 OK CONNECT Test

3.5.6.3.16. 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: <sourceIP>,<sourcePort><connId>,<recData>,
	<dataLeft>
	data
	
	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*

#SRECV – Received Data in Command Mode

	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 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 Or: *if the received datagram, received from <IPaddr and <IPport> is of 60 bytes* AT#SRECV=2,15 #SRECV: <IPaddr>,<IPport>,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

3.5.6.3.17.   *Send Data In Command Mode - #SSEND*

#SSEND – Send Data in Command Mode

#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 bytes. Trial to send data more than 1500 return ERROR 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 the range of supported values for parameter **<connId>**
Example	*Send data through socket number 2* AT#SSEND=2 >Test<CTRL-Z> OK

3.5.6.3.18.   *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 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.

#SSENDEXT – Send Data in Command Mode extended	
	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: Back Space key don't behave like a BS, i.e. previous character is not deleted, sent a generic byte(0x08) through the socket instead. ESC key don't work like an escape sequence, sent a generic byte(0x1B) through the socket instead.)
AT#SSENDEXT=?	Test command returns the range of supported values for parameters **<connId>** and **<bytestosend>**

Example	*Open the socket in command mode:*
	AT#SD=1,0,<port>,"IP address",0,0,1
	OK
	
	Give the command specifiying 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.

3.5.6.3.19. *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

#SLASTCLOSURE – Detect the cause of a socket disconnection	
	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 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) disconnection. 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>**

3.5.6.4. Single Socket AT Commands

3.5.6.4.1. *Authentication User ID - #USERID*

#USERID - Authentication User ID	
AT#USERID= [<user>]	Set command sets the user identification string to be used during the authentication step. Parameter: <user> - string type, it's the authentication User Id; the max length for this value is the output of Test command, **AT#USERID=?** (factory default is the specific value based on carrier's specification). Note : this set command is only for an authentication information of Simple IP system. Note : if a wireless service provider supports only Simple IP data network system such as SK telecom in Korea you need to set this information for data connection. Note : if a wireless service provider supports Mobile IP preferred(like Verizon or Verizon MVNO) or Mobile IP only data network system(like Sprint or Sprint MVNO) you do not need to set this information because this information is automatically populated by a device itself based on carrier's specification. In case of using Mobile IP system, a specific profile is used and its information is set by device itself. Note : although the case of a wireless service provider supporting Mobile IP preferred does also support Simple IP(Simple IP fallback due to authentication fail), a user id for Simple IP does not need to be set(auto population by device itself)
AT#USERID?	Read command reports the current user identification string, in the format: **#USERID: <user>**
AT#USERID=?	Test command returns the maximum allowed length of the string parameter **<user>**.
Example	AT#USERID="myName" OK AT#USERID? #USERID: "myName" OK

3.5.6.4.2. *Authentication Password – #PASSW*

#PASSW - Authentication Password	
AT#PASSW= [<pwd>]	Set command sets the user password string to be used during the authentication step.

#PASSW - Authentication Password	
	Parameter: **<pwd>** - string type, it's the authentication password; the max length for this value is the output of Test command, **AT#PASSW=?** (factory default is the specific value based on carrier's specification). Note : this set command is only for an authentication information of Simple IP system. Note : if a wireless service provider supports only Simple IP data network system such as SK telecom in Korea you need to set this information for data connection. Note : if a wireless service provider supports Mobile IP preferred(like Verizon or Verizon MVNO) or Mobile IP only data network system(like Sprint or Sprint MVNO) you do not need to set this information because this information is set by a network via OTA or other method based on carrier's specification at an initial data connection. In case of using Mobile IP system, a specific profile is used and its information is set by a network. Note : although the case of a wireless service provider supporting Mobile IP preferred does also support Simple IP(Simple IP fallback due to authentication fail), a password for Simple IP does not need to be set(set by a network)
AT#PASSW=?	Test command returns the maximum allowed length of the string parameter **<pwd>**.
Example	AT#PASSW="myPassword" OK

3.5.6.4.3. *Packet Size - #PKTSZ*

#PKTSZ - Packet Size	

AT#PKTSZ=
[<size>]
	Set command sets the default packet size to be used by the TCP/UDP/IP stack for data sending.
	
	Parameter:
	<size> - packet size in bytes
	  0 - automatically chosen by the device
	  1..1500 - packet size in bytes (factory default is 300)

AT#PKTSZ?	Read command reports the current packet size value. Note: after issuing command **AT#PKTSZ=0**, the Read command reports the value automatically chosen by the device.
AT#PKTSZ=?	Test command returns the allowed values for the parameter **<size>**.
Example	AT#PKTSZ=100 OK

#PKTSZ - Packet Size	
	AT#PKTSZ? #PKTSZ: 100 OK AT#PKTSZ=0 OK AT#PKTSZ? #PKTSZ: 300 OK ->*value automatically chosen by device*

3.5.6.4.4. *Data Sending Time-Out - #DSTO*

#DSTO -Data Sending Time-Out	

AT#DSTO=
[<tout>]
	Set command sets the maximum time that the module awaits before sending anyway a packet whose size is less than the default one.
	
	Parameter:
	<tout> - packet sending time-out in 100ms units (factory default is 50)
	  0 - no time-out, wait forever for packets to be completed before send.
	  1..255 hundreds of ms
	
	Note: In order to avoid low performance issues, it is suggested to set the data sending time-out to a value greater than 5.
	
	Note: this time-out applies to data whose size is less than packet size and whose sending would have been delayed for an undefined time until new data to be sent had been received and full packet size reached.

AT#DSTO?	Read command reports the current data sending time-out value.
AT#DSTO=?	Test command returns the allowed values for the parameter **<tout>**.
Example	AT#DSTO=10 ->*1 sec. time-out* OK AT#DSTO? #DSTO: 10 OK

3.5.6.4.5. *Socket Inactivity Time-Out - #SKTTO*

#SKTTO - Socket Inactivity Time-Out	
AT#SKTTO= [<tout>]	Set command sets the maximum time with no data exchanging on the socket that the module awaits before closing the socket and deactivating the CDMA context. Parameter: **<tout>** - socket inactivity time-out in seconds units

#SKTTO - Socket Inactivity Time-Out	
	0 - no time-out. 1..65535 - time-out in sec. units (factory default is 90). Note: this time-out applies when no data is exchanged in the socket for a long time and therefore the socket connection has to be automatically closed. Note: In case CDMA context activated by #SKTOP, both the socket connection and CDMA context closed.
AT#SKTTO?	Read command reports the current socket inactivity time-out value.
AT#SKTTO=?	Test command returns the allowed values for parameter **<tout>**.
Example	AT#SKTTO=30 OK *->(30 sec. time-out)* AT#SKTTO? #SKTTO: 30 OK

3.5.6.4.6. *Socket Definition - #SKTSET*

#SKTSET - Socket Definition	

AT#SKTSET=
 [<socket type>,
 <remote port>,
 <remote addr>,
 [<closure type>],
 [<local port>]]
	Set command sets the socket parameters values.
	 Parameters:
	
	 <socket type> - socket protocol type
	   0 - TCP (factory default)
	   1 - UDP
	 <remote port> - remote host port to be opened
	   1..65535 - port number (factory default is 3333)
	 <remote addr> - 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 in the format: **<host name>**
	     (factory default is the empty string "")
	 <closure type> - socket closure behaviour for TCP
	   0 - local host closes immediately when remote host has closed (default)
	   255 - local host closes after an escape sequence (+++)
	 <local port> - local host port to be used on UDP socket
	   1..65535 - port number (factory default is 0)
	
	 Note: **<closure type>** parameter is valid only for TCP socket type, for UDP sockets shall be left unused.
	
	 Note: **<local port>** parameter is valid only for UDP socket type, for TCP sockets shall be left unused.
	
	 Note: The resolution of the host name is done when opening the socket, therefore if

#SKTSET - Socket Definition	
	an invalid host name is given to the **#SKTSET** command, then an error message will be issued. Note: the DNS Query to be successful requests that: - the authentication parameters are set (**#USERID**, **#PASSW**) - the CDMA coverage is enough to permit a connection.
AT#SKTSET?	Read command reports the socket parameters values, in the format: **AT#SKTSET: <socket type>,<remote port>,<remote addr>,<closure type>,<local port>**
AT#SKTSET=?	Test command returns the allowed values for the parameters.
Example	AT#SKTSET=0,1024,"www.telit.net" OK
Note	Issuing command #QDNS will overwrite <remote addr> setting.

3.5.6.4.7. *Socket Open – #SKTOP*

#SKTOP - Socket Open	
AT#SKTOP	Execution command activates the context number 1, proceeds with the authentication with the user ID and password previously set by #USERID and #PASSW commands, and opens a socket connection with the host specified in the #SKTSET command. Eventually, before opening the socket connection, it issues automatically a DNS query to solve the IP address of the host name. If the connection succeeds a **CONNECT** indication is sent, otherwise a **NO CARRIER** indication is sent.
AT#SKTOP=?	Test command returns the **OK** result code.
Example	AT#SKTOP *..CDMA context activation, authentication and socket open..* CONNECT
Note	This command is obsolete. It's suggested to use the couple #SGACT and #SO instead of it.

3.5.6.4.8. *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: **#QDNS: "<host name>","<IP address>"**

#QDNS - Query DNS	
	Note: the command has to activate the CDMA context if it was not previously activated. In this case the context is deactivated after the DNS query. Note: **<IP address>** is in the format: **xxx.xxx.xxx.xxx**
AT#QDNS=?	Test command returns the **OK** result code.
Note	This command requires that the authentication parameters are correctly set and that the CDMA network is present.

3.5.6.4.9. *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** has 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: [<hostn*1*>,<IPaddr*1*>,[…,[<hostn*n*>,<IPaddr*n*>,]]](0,1)** where: **<hostn*n*>** - hostname, string type **<IPaddr*n*>** - IP address, string type, in the format **"xxx.xxx.xxx.xxx"**

3.5.6.4.10. *Manual DNS Selection - #DNS*

#DNS – Manual DNS Selection	
AT#DNS=<cid>, <primary>, <secondary>	Set command allows to manually set primary and secondary DNS servers Parameters: **<cid>** - context identifier 1 - numeric parameter which specifies a particular PDP context definition **<primary>** - **manual primary DNS server**, string type, in the format

#DNS – Manual DNS Selection	
	"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") **<secondary>** - **manual secondary DNS server**, string type, in the format "xxx.xxx.xxx.xxx" used for the specified cid; we're using this value instead of the **secondary DNS server** come from the network (default is "0.0.0.0"). 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"** we're 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 we're 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 settings in the format: **#DNS: <cid>,<primary>,<secondary>**
AT#DNS=?	Test command reports the supported range of values for the **<cid>** parameter.only, in the format: **#DNS: (1)**

3.5.6.4.11.   *Socket TCP Connection Time-Out - #SKTCT*

#SKTCT - Socket TCP Connection Time-Out	
AT#SKTCT= [<tout>]	Set command sets the TCP connection time-out for the first **CONNECT** answer from the TCP peer to be received. Parameter: **<tout>** - TCP first **CONNECT** answer time-out in 100ms units 10..1200 - hundreds of ms (factory default value is 600). Note: this time-out applies only to the time that the TCP stack waits for the **CONNECT** answer to its connection request. Note: The time for activating the CDMA and resolving the name with the DNS query (if the peer was specified by name and not by address) is not counted in this time-out.
AT#SKTCT?	Read command reports the current TCP connection time-out.
AT#SKTCT=?	Test command returns the allowed values for parameter **<tout>**.
Example	AT#SKTCT=600 OK *socket first connection answer time-out has been set to 60 s.*

3.5.6.4.12. *Socket Parameters Save - #SKTSAV*

#SKTSAV - Socket Parameters Save	
AT#SKTSAV	Execution command saves the actual socket parameters in the NVM of the device. The socket parameters to store are: - User ID - Password - Packet Size - Socket Inactivity Time-Out - Data Sending Time-Out - Socket Type (UDP/TCP) - Remote Port - Remote Address - TCP Connection Time-Out Note : User ID and Password will not be affected by this command execution.
AT#SKTSAV=?	Test command returns the **OK** result code.
Example	AT#SKTSAV OK *socket parameters have been saved in NVM*
Note	If some parameters have not been previously specified then a default value will be stored.

3.5.6.4.13. *Socket Parameters Reset - #SKTRST*

#SKTRST - Socket Parameters Reset	
AT#SKTRST	Execution command resets the actual socket parameters in the NVM of the device to the default ones. The socket parameters to reset are: - User ID - Password - Packet Size - Socket Inactivity Time-Out - Data Sending Time-Out - Socket Type - Remote Port - Remote Address - TCP Connection Time-Out Note : User ID and Password will not be affected by this command execution. It means that they are not set to the default values, just keeping the previous value.
AT#SKTRST=?	Test command returns the **OK** result code.
Example	AT#SKTRST

#SKTRST - Socket Parameters Reset	
	OK *socket parameters have been reset*

3.5.6.4.14. CDMA Data Connection - #CDMADC

#CDMADC – CDMA Data Connection	

AT#CDMADC= [<mode>]	Execution command deactivates/activates CDMA data connection(CDMA PDP context), eventually proceeding with the authentication with the parameters given with #PASSW and #USERID.
	
	Parameter:
	<mode> - CDMA PDP context activation mode
	  0 – CDMA PDP context deactivation request
	  1 – CDMA PDP context activation request
	
	In the case that the CDMA PDP context has been activated, the result code **OK** is preceded by the intermediate result code:
	
	+IP: <ip_address_obtained>
	
	reporting the local IP address obtained from the network.

AT#CDMADC?	Read command reports the current status of the CDMA PDP context, in the format:
	
	#CDMADC: <status>
	
	where:
	<status>
	  0 - CDMA PDP context deactivated
	  1 - CDMA PDP context activated
	  2 - CDMA PDP context activation pending.

AT#CDMADC=?	Test command returns the allowed values for parameter **<mode>**.
Example	AT#CDMADC=1 +IP: 129.137.1.1 OK *Now CDMA PDP Context has been activated and our IP is 129.137.1.1* AT#CDMADC=0 OK *Now CDMA PDP context has been deactivated, IP is lost.*

3.5.6.4.15. Socket Dial - #SKTD

#SKTD - Socket Dial	

AT#SKTD=
[<socket type>,
<remote port>,
<remote addr>,
[<closure type>],
[<local port>]]
	Set command opens the socket towards the peer specified in the parameters.
	
	Parameters:
	<socket type> - socket protocol type
	  0 - TCP (factory default)
	  1 - UDP
	<remote port> - remote host port to be opened
	  1..65535 - port number (factory default is 3333)
	<remote addr> - 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 in the format: **<host name>** (factory default is the empty string "")
	<closure type> - socket closure behaviour for TCP
	  0 - local host closes immediately when remote host has closed (default)
	  255 - local host closes after an escape sequence (+++)
	<local port> - local host port to be used on UDP socket
	  1..65535 - port number
	
	Note: **<closure type>** parameter is valid only for TCP socket type, for UDP sockets shall be left unused.
	
	Note: **<local port>** parameter is valid only for UDP socket type, for TCP sockets shall be left unused.
	
	Note: the resolution of the host name is done when opening the socket, therefore if an invalid host name is given to the **#SKTD** command, then an error message will be issued.
	
	Note: the command to be successful requests that:
	  - the authentication parameters are set (#USERID, #PASSW) the CDMA coverage is enough to permit a connection
	  - the CDMA data connection has been activated with AT#SGACT or AT#CDMADC

AT#SKTD?	Read command reports the socket dial parameters values, in the format: **AT#SKTD: <socket type>,<remote port>,<remote addr>, <closure type>,<local port>**
AT#SKTD=?	Test command returns the allowed values for the parameters.
Example	AT#SKTD=0,1024,"123.255.020.001",255 CONNECT AT#SKTD=1,1024,"123.255.020.001", ,1025 CONNECT *In this way my local port 1025 is opened to the remote port 1024*

#SKTD - Socket Dial	
	AT#SKTD=0,1024,"www.telit.net", 255 CONNECT
Note	The main difference between this command and #SKTOP is that this command does not interact with the CDMA context status, leaving it **ON** or **OFF** according to the #CDMADC setting, therefore when the connection made with **#SKTD** is closed the context (and hence the local IP address) is maintained.

3.5.6.4.16. *Socket Listen - #SKTL*

#SKTL - Socket Listen	

**AT#SKTL
=[<mode>,
<socket type>,
<input port>,
[<closure type>]]**
	Execution command opens/closes the socket listening for connection requests.
	
	Parameters:
	<mode> - socket mode
	  0 - closes socket listening
	  1 - starts socket listening
	<socket type> - socket protocol type
	  0 - TCP
	<input port> - local host input port to be listened
	   1..65535 - port number
	<closure type> - socket closure behaviour for TCP
	  0 - local host closes immediately when remote host has closed (default)
	  255 - local host closes after an escape sequence (+++)
	
	Command returns the **OK** result code if successful.
	
	Note: the command to be successful requests that:
	- the authentication parameters are set (#USERID, #PASSW)
	- the CDMA coverage is enough to permit a connection
	- the CDMA data connection has been activated with AT#SGACT or AT#CDMADC
	
	When a connection request comes on the input port, if the sender is not filtered by the internal firewall (see command #FRWL), an unsolicited code is reported:
	
	   +CONN FROM: <remote addr>
	
	   Where:
	   <remote addr> - host address of the remote machine that contacted the device.
	
	When the connection is established the **CONNECT** indication is given and the modem goes into data transfer mode.
	
	On connection close or when context is closed with AT#SGACT or AT#CDMADC the socket is closed and no listen is anymore active.

#SKTL - Socket Listen	
	If the context is closed by the network while in listening, the socket is closed, no listen is anymore active and an unsolicited code is reported: **#SKTL: ABORTED** Note: when closing the listening socket <listenPort> is a Don't Care parameter.

AT#SKTL?	Read command returns the current socket listening **status** and the last settings of parameters **<socket type>**,**<input port>** and **<closure type>**, in the format:
	
	#SKTL: <status>,<socket type>,<input port>,<closure type>
	Where
	<status> - socket listening status
	  0 - socket not listening
	  1 - socket listening

AT#SKTL=?	Test command returns the allowed values for parameters **<mode>**, **<socket type>**, **<input port>** and **<closure type>**.
Example	*Activate CDMA* AT#CDMADC=1 +IP: ###.###.###.### OK *Start listening* AT#SKTL=1,0,1024 OK Or AT#SKTL=1,0,1024,255 OK *Receive connection requests* +CONN FROM: 192.164.2.1 CONNECT *exchange data with the remote host* *send escape sequence* +++ NO CARRIER *Now listen is not anymore active* *to stop listening* AT#SKTL=0,0,1024, 255

#SKTL - Socket Listen	
	OK
Note	The main difference between this command and **#SKTD** is that **#SKTL** does not contact any peer, nor does any interaction with the CDMA context status, leaving it **ON** or **OFF** according to the **#CDMADC** setting, therefore when the connection made with **#SKTL** is closed the context (and hence the local IP address) is maintained.

3.5.6.4.17. *Socket Listen Ring Indicator - #E2SLRI*

#E2SLRI - Socket Listen Ring Indicator	

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>.**

3.5.6.4.18. *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

#FRWL - Firewall Setup	
	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>** If criteria is matched, then the packet is accepted and the rule scan is finished; if criteria is not matched for any chain the packet is 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
Note	For outgoing connections made with #SKTOP and #SKTD the remote host is dynamically inserted into the **ACCEPT** chain for all the connection duration. Therefore the **#FRWL** command shall be used only for defining the **#SKTL** behaviour, deciding which hosts are allowed to connect to the local device. Rules are not saved in NVM, at startup the rules list will be empty. It will return ERROR if executed using SMSATRUN digest mode or TCPATRUN server mode

3.5.6.4.19. *Data Volume – #GDATAVOL*

#GDATAVOL - Data Volume	

AT#GDATAVOL=
[<mode>]
	Execution command reports, for the active PDP context, the amount of data the last data session received and transmitted, or it will report the total amount of data received and transmitted during the data session, since last reset.
	
	Parameter:
	<mode>
	  0 - it resets the data counter for the all the available PDP context(1).
	  1 - it reports the last data session data counter for the set PDP context ,in the format:

#GDATAVOL - Data Volume	
	#GDATAVOL: <cid>,<tot>,<sent>,<received>
	where:
	<cid> - PDP context identifier
	<tot> - number of bytes either received or transmitted in the last data session
	<sent> - number of bytes transmitted in the last data
	<received> - number of bytes received in the last data session
	2 - it reports the total data counter, since last reset, for the set PDP context, in the format:
	#GDATAVOL: <cid>,<tot>,<sent>,<received>
	where:
	<cid> - PDP context identifier
	<tot> - number of bytes either received or transmitted, in every data session since last reset
	<sent> - number of bytes transmitted, in every data session since last reset
	<receivedn> - number of bytes received, in every data session since last reset
	Note: last data session counters are not saved in NVM so they are lost at power off.
	Note: total data session counters are saved on NVM.
AT#GDATAVOL=?	Test command returns the range of supported values for parameter <mode>.
Note	Internal use only

3.5.6.4.20.   ICMP Ping Support – #ICMP

#ICMP – ICMP Ping Support	
AT#ICMP=<mode>	Set command enables/disables the ICMP Ping support. Parameter: **<mode>** 0 - disable ICMP Ping support (default) 1 - enable firewalled ICMP Ping support: the module is sending a proper ECHO_REPLY only to a subset of IP Addresses pinging it; this subset of IP Addresses has been previously specified through **#FRWL** (see) 2 - enable free ICMP Ping support; the module is sending a proper ECHO_REPLY to every IP Address pinging it.
AT#ICMP?	Read command returns whether the ICMP Ping support is currently enabled or not, in the format: **#ICMP: <mode>**
AT#ICMP=?	Test command reports the supported range of values for the **<mode>** parameter.

3.5.6.4.21.   Ping Request – #PING

#PING – Ping Request	

AT#PING=<IPaddr> [,<retryNum>[,<len> [,<timeout> [,<ttl>]]]]	Set command sends a Ping Echo Request messages and to receive the corresponding Echo Reply.
	
	Once the single Echo Reply is received a string like that this is displayed:
	#PING:<replyId>,<IpAddress>,<replyTime><ttl>
	
	<replyId> - Echo Reply number
	<IpAddress> - IP address of the remote host
	<replyTime> - Time, in 100ms units, required to receive the response
	<ttl> - Time to live of the Echo Reply message.
	
	Parameter:
	<IPaddr> - Address of the remote host. 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
	<retryNum> - Number of Ping Echo Request to be sent:
	1-64 (default 4)
	<len> - Length of Ping Echo Request message
	32-1460 (default 32)
	<timeout> - The timeout, in 100 ms units, waiting a single Echo Reply:
	1-600 (default 50)
	<ttl> - Time to live:
	1-255 (default 128)

AT#PING=?	Test command reports the supported range of values for the **#PING** command parameters
Example	AT#PING=www.telit.com #PING: 01,"81.201.117.177",6,50 #PING: 02,"81.201.117.177",5,50 #PING: 03,"81.201.117.177",6,50 #PING: 04,"81.201.117.177",5,50 OK
Note	When the Echo Request timeout expires (no reply received on time) the response will contain **<replyTime>** set to 600 and **<ttl>** set to 255. To receive the corresponding Echo Reply is not required to enable separately AT#ICMP Before sending PING request the CDMA context must have been activated by AT#SGACT or AT#CDMADC

#PING – Ping Request	
	When in dormant state, the modem retruns ERROR as the response of AT#PING at the first time so you need to re-execute it until the traffic is open.

3.5.6.4.22. Maximum TCP Payload Size - #TCPMAXDAT

#TCPMAXDAT – Maximum TCP Payload Size	
AT#TCPMAXDAT= <size>	Set command allows to set the maximum TCP payload size in TCP header options. Parameter: **<size>** - maximum TCP payload size accepted in one single TCP/IP datagram; it is sent in TCP header options in SYN packet. 0 - the maximum TCP payload size is automatically handled by module (default). 496..1420 - maximum TCP payload size
AT#TCPMAXDAT?	Read command reports the current maximum TCP payload size, in the format: **#TCPMAXDAT: <size>**
AT#TCPMAXDAT=?	Test command reports the supported range of values for parameter **<size>**

3.5.6.4.23. TCP Reassembly - #TCPREASS

#TCPREASS – TCP Reassembly	
AT#TCPREASS= <n>	Set command enables/disables the **TCP reassembly feature**, in order to handle fragmented TCP packets. Parameter: **<n>** 1 - enable TCP reassembly feature(default)
AT#TCPREASS?	Read command returns whether the TCP reassembly feature is enabled or not, in the format: **#TCPREASS: <n>**
AT#TCPREASS=?	Test command returns the supported range of values for parameter **<n>**.

3.5.6.5. FTP AT Commands

3.5.6.5.1. 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 for parameter <tout>

3.5.6.5.2. *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).
	<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 exceeds 1200 is considered as 1200.
	Note: Before opening FTP connection the CDMA must been activated with AT#SGACT or AT#CDMADC

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

3.5.6.5.3. *FTP Close - #FTPCLOSE*

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

3.5.6.5.4. *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.

#FTPPUT - FTP Put	
	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**>

3.5.6.5.5. *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 is 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.

3.5.6.5.6. *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

#FTPGETPKT - FTP Get in command mode	
	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.

3.5.6.5.7. *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 parameter **<type>**:
	#FTPTYPE: (0,1)

3.5.6.5.8. *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.

3.5.6.5.9. *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.

3.5.6.5.10. *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.

3.5.6.5.11. *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.

3.5.6.5.12. *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.

#FTPLIST - FTP List	
	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.

3.5.6.5.13.　*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 a 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 returns the range of supported values for <blocksize> parameter.
Example	AT#FTPRECV? #FTPRECV: 3000 OK *Read required part of the buffered data:* AT#FTPRECV=400 #FTPRECV:400

#FTPRECV – Receive Data In Command Mode	
	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)*

3.5.6.5.14.   *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.

#FTPAPP – FTP Append	

	<**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**>

3.5.6.5.15. *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**>

#FTPAPPEXT - FTP Append Extended	
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)*

3.5.6.5.16.   FTP Config – #FTPCFG

#FTPCFG - FTP Config	
AT#FTPCFG= \<tout\>, \<IPPignoring\> [,\<FTPSEn\>]	**\<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. s Note: The parameter is not saved in NVM.

#FTPCFG - FTP Config	
	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 IPV4 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.
AT#FTPCFG?	Read command reports the currently selected parameters in the format: **AT#FTPCFG=<tout>,<IPPignoring>,<FTPSEn>**
AT#FTPCFG=?	Test command reports the sypported range of values for parameter(s) **<tout>,<IPPignoring>,<FTPSEn>**

3.5.6.5.17. Get file size - #FTPFSIZE

#FTPFSIZE – Get file size from FTP server	
AT#FTPFSIZE=<filename>	Excution command, issued during an FTP connection, permits to get file size of <filename> file.
AT#FTPFSIZE=?	Test command returns the OK result code.

3.5.6.5.18. Set restart position - #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**).

#FTPREST – Set restart position for FTP GET	
	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.

3.5.6.6. E-mail Management AT Commands

3.5.6.6.1. *E-mail SMTP Server - #ESMTP*

#ESMTP - E-mail SMTP Server	

AT#ESMTP=
[<**smtp**>]
	Set command sets the SMTP server address, used for E-mail sending.
	SMTP server can be specified as IP address or as nick name.
	
	Parameter:
	<**smtp**> - SMTP server address, 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 in the format: <**host name**>
	   (factory default is the empty string "")
	
	Note: the max length for <**smtp**> is the output of Test command.

AT#ESMTP?	Read Command reports the current SMTP server address, in the format: #ESMTP: <smtp>
AT#ESMTP=?	Test command returns the max length for the parameter <**smtp**>.
Example	AT#ESMTP="smtp.mydomain.com" OK
Note	The SMTP server used shall be inside the APN space (the smtp server provided by the network operator) or it must allow the Relay, otherwise it will refuse to send the e-mail.

3.5.6.6.2. *E-mail Sender Address - #EADDR*

#EADDR - E-mail Sender Address	

AT#EADDR=
[<**e-add**>]
	Set command sets the sender address string to be used for sending the e-mail.
	
	Parameter:
	<**e-addr**> - sender address, string type.
	   - any string value up to max length reported in the Test command.
	   (factory default is the empty string "")

AT#EADDR?	Read command reports the current sender address, in the format: #EADDR: <e-addr>
AT#EADDR=?	Test command returns the maximum allowed length of the string parameter <e-

#EADDR - E-mail Sender Address	
	addr>.
Example	AT#EADDR="me@email.box.com" OK AT#EADDR? #EADDR: "me@email.box.com" OK

3.5.6.6.3. *E-mail Authentication User Name - #EUSER*

#EUSER - E-mail Authentication User Name	
AT#EUSER= [<e-user>]	Set command sets the user identification string to be used during the authentication step of the SMTP. Parameter: **<e-user>** - e-mail authentication User ID, string type. - any string value up to max length reported in the Test command. (factory default is the empty string "") Note: if no authentication is required then the **<e-user>** parameter shall be empty "".
AT#EUSER?	Read command reports the current user identification string, in the format: **#EUSER: <e-user>**
AT#EUSER=?	Test command returns the maximum allowed length of the string parameter **<e-user>**.
Example	AT#EUSER="myE-Name" OK AT#EUSER? #EUSER: "myE-Name" OK
Note	It is a different user field than the one used for CDMA authentication (see #USERID).

3.5.6.6.4. *E-mail Authentication Password - #EPASSW*

#EPASSW - E-mail Authentication Password	
AT#EPASSW= [<e-pwd>]	Set command sets the password string to be used during the authentication step of the SMTP. Parameter: **<e-pwd>** - e-mail authentication password, string type. - any string value up to max length reported in the Test command. (factory default is the empty string "") Note: if no authentication is required then the **<e-pwd>** parameter shall be empty

#EPASSW - E-mail Authentication Password	
	"".
AT#EPASSW=?	Test command returns the maximum allowed length of the string parameter **<e-pwd>**.
Example	AT#EPASSW="myPassword" OK
Note	It is a different password field than the one used for CDMA authentication (see #PASSW).

3.5.6.6.5. *E-mail Sending With CDMA Context Activation – #SEMAIL*

#SEMAIL - E-mail Sending With CDMA Context Activation	

AT#SEMAIL=
[<da>[,<subj>[,
<att>]]]
	Execution command sends an e-mail message. If not previously activated by #EMAILACT, activates a CDMA context. The CDMA context activated by #SEMAIL is deactivated when the e-mail is sent.
	
	Parameters:
	<da> - destination address, string type.
	<subj> - subject of the message, string type.
	<att> - attached image flag(attaching image is not supported)
	  0 - don't attach any image
	  1 - attach the last snapshot taken
	
	The device responds to the command with the prompt '**>**' and awaits for the message body text.
	
	To complete the operation send **Ctrl-Z** char (**0x1A** hex); to exit without writing the message send **ESC** char (**0x1B** hex).
	
	If e-mail message is successfully sent, then the response is **OK**.
	If message sending fails for some reason, an error code is reported
	
	Note: Care must be taken to ensure that during the command execution, no other commands are issued.
	
	To avoid malfunctions is suggested to wait for the **OK** or **ERROR / +CMS ERROR:<err>** response before issuing further commands.
	
	Note: Maximum length for message body is 1024 bytes. Trying to send more data will cause the surplus to be discarded and lost.

AT#SEMAIL=?	Test command returns the **OK** result code.
Example	AT#SEMAIL="me@myaddress.com","subject of the mail",0 >message body… this is the text of the mail message… CTRL-Z ..*wait*.. OK

#SEMAIL - E-mail Sending With CDMA Context Activation	
	Message has been sent.
Note	This command is for backward compatibility. It's suggested to use the couple #EMAILACT and #EMAILD instead of it.

3.5.6.6.6. *E-mail CDMA Context Activation - #EMAILACT*

#EMAILACT - E-mail CDMA Context Ativation	

AT#EMAILACT= [<mode>]	Execution command deactivates/activates the CDMA context, eventually proceeding with the authentication with the parameters given with #PASSW and #USERID.
	
	Parameter:
	<mode> - CDMA context activation mode
	  0 - CDMA context deactivation request
	  1 - CDMA context activation request

AT#EMAILACT?	Read command reports the current status of the CDMA context for the e-mail, in the format:
	
	#EMAILACT: <status>
	
	where:
	<status>
	  0 - CDMA context deactivated
	  1 - CDMA context activated

AT#EMAILACT=?	Test command returns the allowed values for parameter **<mode>**.
Example	AT#EMAILACT=1 OK *Now* CDMA *Context has been activated* AT#EMAILACT=0 OK *Now* CDMA *context has been deactivated.*

3.5.6.6.7. *E-mail Sending - #EMAILD*

#EMAILD - E-mail Sending	

AT#EMAILD= [<da>[, <subj>[,<att>]]]	Execution command sends an e-mail message if CDMA context has already been activated with **AT#SGACT=1,1** or **AT#EMAILACT=1**
	
	Parameters:
	<da> - destination address, string type.
	<subj> - subject of the message, string type
	<att> - attached image flag(attaching image is not supported)
	  0 - don't attach any image
	    1 - attach the last snapshot taken

#EMAILD - E-mail Sending	
	The device responds to the command with the prompt '**>**' and awaits for the message body text. To complete the operation send **Ctrl-Z** char (**0x1A** hex); to exit without writing the message send **ESC** char (**0x1B** hex). If e-mail message is successfully sent, then the response is **OK**. If message sending fails for some reason, an error code is reported Note: Care must be taken to ensure that during the command execution, no other commands are issued. To avoid malfunctions is suggested to wait for the **OK** or **ERROR / +CMS ERROR:<err>** response before issuing further commands. Note: Maximum length for message body is 1024 bytes. Trying to send more data will cause the surplus to be discarded and lost.
AT#EMAILD=?	Test command returns the **OK** result code.
Example	AT#EMAILD="me@myaddress.com","subject of the mail",0 >message body… this is the text of the mail message… CTRL-Z *..wait..* OK *Message has been sent.*
Note	The only difference between this command and the #SEMAIL is that this command does not interact with the CDMA context status, leaving it **ON** or **OFF** according to the #EMAILACT setting, thus, when the connection made with **#EMAILD** is closed, the context status is maintained.

3.5.6.6.8. *E-mail Parameters Save - #ESAV*

#ESAV - E-mail Parameters Save	
AT#ESAV	Execution command saves the actual e-mail parameters in the NVM of the device. The values stored are: - E-mail User Name - E-mail Password - E-mail Sender Address - E-mail SMTP server
AT#ESAV=?	Test command returns the **OK** result code.
Note	If some parameters have not been previously specified then a default value will be taken.

3.5.6.6.9. *E-mail Parameters Reset - #ERST*

#ERST - E-mail Parameters Reset	

AT#ERST	Execution command resets the actual e-mail parameters in the NVM of the device to the default ones.
	
	The values reset are:
	  - E-mail User Name
	  - E-mail Password
	  - E-mail Sender Address
	  - E-mail SMTP server

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

3.5.6.6.10.   *SMTP Read Message - #EMAILMSG*

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

3.5.6.7. HTTP AT Commands

3.5.6.7.1. *Configure HTTP parameters - #HTTPCFG*

#HTTPCFG – configure HTTP parameters	

AT#HTTPCFG=<prof_id>[,<server_address>[,<server_port>[,<auth_type>[,<username>[,<password>[,<ssl_enabled>[,<timeout> [,<cid>[,<pkt_size>][,<UNUSED_1>[,<UNUSED_2>]]]]]]]]]]]	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 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.

#HTTPCFG – configure HTTP parameters	

	<password> - String parameter indicating authentication password for HTTP.
	
	<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> - Numeric parameter indicating the PDP Context Identifier. Range: (1). Default: 1
	
	<pkt_size> - send(#HTTPSND) or recv(#HTTPRCV) size for data sending or receiving.
	0 – select automatically default value(300).
	1..1500 – send or recv size in bytes.
	
	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>,<pkt_size>,0,0<CR><LF>[<CR><LF>#HTTPCFG: <prof_id>,<server_address>,<server_port>,<auth_type>,<username>,<password>,<ssl_enabled>,<timeout>,<cid>>,<pkt_size>,0,0]<CR><LF>[…]]**
AT#HTTPCFG=?	Test command returns the supported range of parameters **<prof_id>,** **<server_port>, <auth_type>, <ssl_enabled>, <timeout>, <cid>** and **<pkt_size>** 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) ,(list of supported <pkt_size>s),(<UNUSED_1>),(<UNUSED_2>)** where: **<s_length>** - integer type value indicating the maximum length of parameter **<server_address>**.

#HTTPCFG – configure HTTP parameters	
	<u_length> - integer type value indicating the maximum length of parameter **<username>**. **<p_length>** - integer type value indicating the maximum length of parameter **<password>**

3.5.6.7.2. *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 can not 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

#HTTPQRY – send HTTP GET, HEAD or DELETE request	
	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: **<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>**.

3.5.6.7.3. *Send HTTP POST or PUT request - #HTTPSND*

#HTTPSND – send HTTP POST or PUT request	
AT#HTTPSND=<prof_id>,<command>,<resource>,<data_len>[,<post_param>[,<extra_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

#HTTPSND – send HTTP POST or PUT request	
	"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. Note: the HTTP request header sent with #HTTPSND always contains the "Connection: close" line, and it can not 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#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"*

#HTTPSND – send HTTP POST or PUT request	
	AT#HTTPSND=0,0,”/”,100,0 >>> *Post 100 byte with "multipart/form-data" and extension* AT#HTTPSND=0,0,”/”,100,”3:boundary=----FormBoundary” >>>

3.5.6.7.4. *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 unspecified for <maxByte>, server data will be transferred until it completes with once AT#HTTPRCV execution. 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>)**

3.5.6.8. SSL AT Commands(SSL is supported only on big memory, 128*64)

3.5.6.8.1. *Configure general parameters of a SSL socket - #SSLCFG*

#SSLCFG – Configure general parameters of a SSL socket	

#SSLCFG – Configure general parameters of a SSL socket	
AT#SSLCFG= <SSId>, <cid>, <pktSz>, <maxTo>, <defTo>, <txTo> [,<sslSRINGMode> [,<UNUSED_1> [,<UNUSED_2> [,<UNUSED_3>]]]]	This command allows configuration SSL connection parameters. Parameters: **<SSId>** - Secure Socket Identifier 1 – Until now SSL block manages only one socket **<cid>** - PDP Context Identifier 1 – Until now only context one is supported **<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). <**sslSRINGMode**> -the presentation mode of the SSLSRING unsolicited indication, which informs the user about new incoming data that can be read in command mode. It can be disabled using value 0. 0 – disable 1 – enable(default) Note) if user set sslSRINGMode 1, will be shown the following format. SSLSRING: <SSId>,<dataLen> 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#SSLCFG?	Read command reports the currently selected parameters in the format: **#SSLCFG:** **<SSId1>,<cid>,<pktSz>,<maxTo>,<defTo>,<txTo>,<sslSRINGMode>,0,0,0**
AT#SSLCFG=?	Test command returns the range of supported values for all the parameters. **#SSLCFG: (1),(1),(0-1500),(0-65535),(10-5000),(0-255),(0,1),(0),(0),(0)**

3.5.6.8.2. *Opening 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. Both command and online modes can be used. In the first case 'OK' is printed on success, and data exchange can be performed by means of #SSLSEND and #SSLRECV commands. In online mode 'CONNECT' message is printed, and data can be sent/received directly to/by the serial port. Communication can be suspended by issuing the escape sequence(by default +++) and restored with #SSLO command. 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 – only value 0 is supported <connMode> - connection mode 0 – online mode connection 1 – command mode connection(factory default). <Timeout> - time-out in 100 ms units. It represents the TCP inter-packet delay. Note: it DOES NOT represent the total handshake timeout. 10..5000 – hundreds of ms(factory default is 100) 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: 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: Before opening a SSL connection the PPP context must have been activated by AT#SGACT=x,1. Note: in online mode data are transmitted as soon as the data packet size is reached or as after a transmission timeout. Both these parameters are configurable by using

#SSLD – Opens a socket SSL to a remote server	
	#SSLCFG. Note: Before opening a SSL connection, make sure to have stored the needed secure data(CA certificate), using **AT#SSLSECDATA**.
AT#SSLD=?	Test command returns the range of supported values for all the parameters: **#SSLD: (1),(1-65535),,(0),(0,1),(10-5000)**

3.5.6.8.3. *Enabling 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: an error is raised if #SSLEN=X,1 is issued when the socket 'X' is already enabled and if #SSLEN=X,0 is issued when the socket 'X' is already disabled. Note: a SSL socket cannot be disabled by issuing #SSLEN=1 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: (1),(0,1)**

3.5.6.8.4. *Closing a SSL socket - #SSLH*

#SSLH – Close a SSL socket	
AT#SSLH=	This command allows closing the SSL connection.

#SSLH – Close a SSL socket	
<SSId>[, <ClosureType>]	Parameters: **<SSId>** - Secure Socket Identifier 1 – Until now SSL block manager only one socket. **<ClosureType>** - how to close SSL socket 0 – only value 0 is supported Note: if secure socket is not enabled using **AT#SSLEN** only test requests can be made. Note: in client side if <**ClosureType**> is not set the value set into **AT#SSLD** is used.
AT#SSLH=?	Test command returns the range of supported values for all the parameters: **#SSLH: (1),(0)**

3.5.6.8.5. *Restoring 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: if an error occur during reconnection the socket can not 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)**

3.5.6.8.6. *Reading 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

#SSLRECV – Read data from a SSL socket	
	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 1..5000 – hundreds of ms(factory default is 100) If no data are received the device respondes: **#SSLRECV: 0<CR><LF>** **TIMEOUT<CR><LF>** **<CR><LF>** **OK** If the remote host closes the connection the device respondes: **#SSLRECV: 0<CR><LF>** **DISCONNECTED<CR><LF>** **<CR><LF>** **OK** If data are received the device respondes: **#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),(1-5000)**

3.5.6.8.7. *Reporting 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

#SSLS – Report the status of a SSL socket	
	If secure socket is connected, the device responds to the command: **#SSLS:<SSId>,2,<CipherSuite>** < **CipherSuite** > available values are: 0 - unknown 1 - TLS_RSA_WITH_RC4_128_MD5 2 - TLS_RSA_WITH_RC4_128_SHA 3 - TLS_RSA_WITH_AES_128_CBC_SHA 4 - TLS_RSA_WITH_NULL_MD5(Not supported, Dummy) Otherwise: **#SSLS: <SSId>,<ConnectionStatus>** **<ConnectionStatus>** available values are: 0 – Socket Disabled 1 – Connection closed 2 – Connection open 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)**

3.5.6.8.8. *Managing 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. Parmaters: **<SSId>** - Secure Socket Identifier 1 – Until now SSL block manages only one socket. **<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..4000

#SSLSECDATA – Manage the security data	
	If the <Action> parameter si 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 delete) the response has the following format: **#SSLSECDATA: <connId>,<DataType>** **No data sotred** **OK** Note: Private keys with password ARE NOT supported. 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>, <PrivKeyIsSet> 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),(1-4000)**

3.5.6.8.9. *Sending data through a SSL socket - #SSLSEND*

#SSLSEND – Send data to SSL Socket	
AT#SSLSEND=	This command allows sending data throught a secure socket.

#SSLSEND – Send data to SSL Socket	
<SSId>[,<Timeout>]	Parameters: **<SSId>** - Secure Socket Identifier 1 – Until now SSL block manage only one socket. **<Timeout>** - socket send timeout, in 100 ms units 1..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 1500. 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),(1-5000)**

3.5.6.8.10. *Configure security parameters of a SSL socket – #SSLSECCFG*

#SSLSECCFG – Configure security parameters of a SSL socket	
AT#SSLSECCFG= <SSId>, <CipherSuite>, <auth_mode> [,<cert_format>]	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_128_CBC_SHA 4 - TLS_RSA_WITH_NULL_SHA(Not supported)

#SSLSECCFG – Configure security parameters of a SSL socket	
	<auth_mode> 0 – SSL Verify None[default] 1 – Manage server authentication 2 – Server/Client authentication: CA Certificate(server), Certificate(client) and Private Key(client) are needed **<cert_format>** is an optional parameter. It selects the format of the certificate to be stored via **#SSLSECDATA** command 0 – DER format 1 – PEM format[default] Note – it is supposed that the module is just powered on and the **AT#SSLSECCFG** command is entered without **<cert_format>** parameter, the default format is PEM. In this case the **AT#SSLSECCFG?** Read command doesn't return the setting of the format in order to meet retro compatibility with other families. Now, let's assume that **AT#SSLSECCFG** command is entered again, but using the **<cert_format>** parameter for the first time: if the read command is entered, it reports the parameter value just used. If subsequently the **<cert_format>** is omitted, the **AT#SSLSECCFG?** Read command reports the parameter value entered the last time. Note: Server Cacertificate has to be stored through **AT#SSLSECDATA**. 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>,<auth_mode>[,<cert_format>]**
AT#SSLSECCFG=?	Test command returns the range of supported values for all the parameters.

3.5.6.8.11. *Sending data through a secure socket in Command Mode extended - #SSLSENDEXT*

#SSLSENDEXT – Sending data through a secure socket in Command Mode extended	
AT#SSLSENDEXT=<SSId>,<bytestosend>[,<Timeout>]	This command allows sending data through a secure socket. Parameters: **<SSId>** - Secure Socket Identifier 1 – Until now SSL block manage only one socket. **<bytestosend>** - number of bytes to be sent Please refer to test command for range **<Timeout>** - time-out in 100 ms units 1..5000 – hundreds of ms (factory default is 100)

#SSLSENDEXT – Sending data through a secure socket in Command Mode extended	
	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: 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. 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#SSLSENDEXT=?	Test command returns the range of supported values for parameters <SSId> , <bytestosend> and <Timeout>. **#SSLSENDEXT: (1),(1-1500),(1-5000)**

3.5.6.9. Easy Script® Extension - Python[4] Interpreter, AT Commands

3.5.6.9.1. *Write Script - #WSCRIPT*

#WSCRIPT - Write Script	

AT#WSCRIPT=
[<script_name>,
<size>,
[,<hidden>]]
	Execution command causes the MODULE to store a file in the Easy Script® related NVM, naming it **<script_name>**
	
	    The file should be sent using RAW ASCII file transfer.
	    It is important to set properly the port settings. In particular:
	    Flow control: hardware.
	    Baud rate: 115200 bps
	
	Parameters:
	<script_name> - name of the file in NVM, string type (max 127 chars, case sensitive).
	<size> - file size in bytes
	<hidden> - file hidden attribute
	  0 - file content is readable with **#RSCRIPT** (default).

---

[4] PYTHON is a registered trademark of the Python Software Foundation.

#WSCRIPT - Write Script	
	1 - file content is hidden, **#RSCRIPT** command will report empty file. The device shall prompt a five character sequence **<CR><LF><greater_than><greater_than><greater_than>** **(IRA 13, 10, 62, 62, 62)** after command line is terminated with **<CR>**; after that a file can be entered from TE, sized **<size>** bytes. The operations completes when all the bytes are received. If writing ends successfully, the response is **OK**; otherwise an error code is reported. Note: the file name should be passed between quotes; every textual script file must have **.py** extension, whilst every pre-compiled executable script file must have **.pyo** extension; file names are case sensitive. Note: when sending the script be sure that the line terminator is **<CR><LF>** and that your terminal program does not change it. Note: with the hidden attribute it is possible to protect your files from being viewed and copied, only the file name can be viewed, its content is hidden even if the file is still being run correctly. It's your care to maintain knowledge on what the file contains.
AT#WSCRIPT=?	Test command returns **OK** result code.
Example	AT#WSCRIPT="First.py ",54,0 *>>> here receive the prompt; then type or send the textual script, sized 54 bytes* OK *Textual script has been stored*
Note	It's recommended to use the extension **.py** only for textual script files and the extension **.pyo** only for pre-compiled executable script files.

3.5.6.9.2. *Select Active Script - #ESCRIPT*

#ESCRIPT - Select Active Script	

AT#ESCRIPT=
[<script_name>]
	Set command selects either
	   a) the name of the textual script file that will be compiled and executed by the Easy Script® compiler at startup according to last **#STARTMODESCR** setting, or
	   b) the name of the pre-compiled executable file that will be executed at startup according to last **#STARTMODESCR** setting.
	
	We call this file (either textual or pre-compiled) the **current script**.
	
	Parameter:

#ESCRIPT - Select Active Script	
	<script_name> - file name, string type (max 127 chars, case sensitive). Note: all textual script files must have **.py** extension; all pre-compiled executable files must have **.pyo** extension. Note: **<script_name>** must match to the name of a file written by **#WSCRIPT** in order to have it run. Note: the command does not check whether a textual script named **<script_name>** does exist or not in the Easy Script® related NVM. If the file **<script_name>** is not present at startup then the compiler will not execute.
AT#ESCRIPT?	Read command reports as a quoted string the file name of the **current script**.
AT#ESCRIPT=?	Test command returns **OK** result code.

3.5.6.9.3. *Script Execution Start Mode - #STARTMODESCR*

#STARTMODESCR - Script Execution Start Mode	
AT#STARTMODESCR= <script_start_mode> [,<script_start_to>]	Set command sets the **current script** (see **#ESCRIPT**) execution start mode. Parameter: **<script_start_mode>** - **currente script** execution start mode 0 - **current script** will be executed at startup only if the **DTR** line is found **Low** (that is: COM is not open on a PC), otherwise the Easy Script® interpreter will not execute and the MODULE will behave normally answering only to AT commands on the serial port (factory default). 1 - **current script** will be executed at startup only if the user does not send any AT command on the serial port for the time interval specified in **<script_start_to>** parameter, otherwise the Easy Script® interpreter will not execute and the MODULE will behave normally answering only to AT commands on the serial port. The **DTR** line is not tested. **<script_start_to>** - **current script** start time-out; 10..60 - time interval in seconds; this parameter is used only if parameter **<script_start_mode>** is set to 1; it is the waiting time for an AT command on the serial port to disable active script execution start. If the user does not send any AT command on the serial port for the time specified in this parameter active script will not be executed (default is 10).
AT#STARTMODESCR?	Read command reports the **current script** start mode and the **current script** start time-out, in the format: **#STARTMODESCR= <script_start_mode>,<script_start_timeout>**
AT#STARTMODESCR=?	Test command returns the range of available values for parameters **<script_start_mode>** and **<script_start_timeout>**, in the format: **#STARTMODESCR: (0,1),(10-60)**

3.5.6.9.4. *Execute Active Script - #EXECSCR*

#EXECSCR - Execute Active Script	
AT#EXECSCR	Execution command causes the **current script** (see **#ESCRIPT**) execution not at startup. This command is useful when the execution at startup has been blocked deliberately and the user wants to control execution start.
AT#EXECSCR=?	Test command returns **OK** result code.

3.5.6.9.5. *Read Script - #RSCRIPT*

#RSCRIPT - Read Script	
AT#RSCRIPT= [<script_name>]	Execution command reports the content of file **<script_name>**. Parameter: **<script_name>** - file name, string type (max 127 chars, case sensitive). The device shall prompt a five character sequence **<CR><LF><less_than><less_than><less_than>** **(IRA 13, 10, 60, 60, 60)** followed by the file content. Note: if the file **<script_name>** was saved with the hidden attribute, then an empty file is reported with the **OK** result code. Note: If the file **<script_name>** is not present an error code is reported.
AT#RSCRIPT=?	Test command returns **OK** result code.
Example	AT#RSCRIPT="First.py " *hereafter receive the prompt; then the script is displayed, immediately after the prompt* <<<import MDM MDM.send('AT\r',10) Ans=MDM.receive(20) OK

3.5.6.9.6. *List Script Names - #LSCRIPT*

#LSCRIPT - List Script Names	
AT#LSCRIPT	Execution command reports either the list of file names for the files currently stored in the Easy Script® related NVM and the available free NVM memory in the format: **[#LSCRIPT: <script_name1>,<size1>…** **[<CR><LF>#LSCRIPT: <script_name*n*>,<size*n*>]]** **<CR><LF>#LSCRIPT: free bytes: <free_NVM>**

#LSCRIPT - List Script Names	
	where: **<script-name*n*>** - file name, quoted string type (max 127 chars, case sensitive) **<size*n*>** - size of script in bytes **<free_NVM>** - size of available NVM memory in bytes
AT#LSCRIPT=?	Test command returns **OK** result code.
Example	AT#LSCRIPT #LSCRIPT: "First.py",51 #LSCRIPT: "Second.py",178 #LSCRIPT: "Third.py",95 #LSCRIPT: free bytes: 20000 OK

3.5.6.9.7. *List Script Names adding CRC16 information – #LCSCRIPT*

#LCSCRIPT - List Script Names adding CRC16 information	
AT#LCSCRIPT	Execution command reports either the list of file names for the files currently stored in the Easy Script® related NVM, adding CRC16 information, and the available free NVM memory in the format: **[#LCSCRIPT: <script_name1>,<size1>[,<crc1>]…** **[<CR><LF>#LCSCRIPT: <script_name*n*>,<size*n*>[,<crc*n*>]]]** **<CR><LF>#LCSCRIPT: free bytes: <free_NVM>** where: **<script-name*n*>** - file name, quoted string type (max 127 chars, case sensitive) **<size*n*>** - size of script in bytes **<crc*n*>** - CRC16 poly (x^16+x^12+x^5+1) of script in hex format **<free_NVM>** - size of available NVM memory in bytes Note: CRC16 is calculated using the standard reversed CRC16-CCITT x^16+x^12+x^5+1 polynomial (0x1021 representation, reversed) with initial value FFFF. Note: if one file currently stored in NVM is in use than CRC16 cannot be calculated and execution command does not report **<crc*n*>** for that file. This is always true if command is executed by a Python script because at least the file pointed by #ESCRIPT is in use.
AT#LCSCRIPT=<script_name>	Execution command reports size and CRC16 information of file **<script_name>** in the format: **[#LCSCRIPT: <script_name>,<size>[,<crc>]]** where: **<script-name>** - file name, quoted string type (max 127 chars, case sensitive) **<size>** - size of script in bytes

#LCSCRIPT - List Script Names adding CRC16 information	
	<crc> - CRC16 poly (x^16+x^12+x^5+1) of script in hex format Parameter: **<script_name>** - file name, string type (max 127 chars, case sensitive). Note: CRC16 is calculated using the standard reversed CRC16-CCITT x^16+x^12+x^5+1 polynomial (0x1021 representation, reversed) with initial value FFFF. Note: if file **<script_name>** is in use than CRC16 cannot be calculated and execution command does not report **<crc>**. Note: if file **<script_name>** is not in the list of files stored in NVM execution command exits with error message.
AT#LCSCRIPT=?	Test command returns **OK** result code.
Example	AT#LCSCRIPT #LCSCRIPT: "First.py",51,8FD6 #LCSCRIPT: "Second.py",178,A034 #LCSCRIPT: "Third.py",120,7C48 #LCSCRIPT: free bytes: 20000 OK AT#LCSCRIPT="Second.py" #LCSCRIPT: "Second.py",178,A034 OK If file Third.py is already in use. AT#LCSCRIPT #LCSCRIPT: "First.py",51,8FD6 #LCSCRIPT: "Second.py",178,A034 #LCSCRIPT: "Third.py",120 #LCSCRIPT: free bytes: 20000 OK

3.5.6.9.8. *Delete Script - #DSCRIPT*

#DSCRIPT - Delete Script	
AT#DSCRIPT= **[<script_name>]**	Execution command deletes a file from Easy Script® related NVM memory. Parameter: **<script_name>** - name of the file to delete, string type (max 127 chars, case sensitive)

#DSCRIPT - Delete Script	
	Note: if the file **<script_name>** is not present an error code is reported.
AT#DSCRIPT=?	Test command returns **OK** result code.
Example	AT#DSCRIPT="Third.py" OK

3.5.6.10. AT Run Commands

3.5.6.10.1. *Enable SMS Run AT Service - #SMSATRUN*

#SMSATRUN – Enable SMS AT Run service	

**AT#SMSATRUN=
<mod>**
	Set command enables/disables the SMS AT RUN service.
	
	Parameter:
	< mod >
	      0: Service Disabled
	      1: Service Enabled
	
	Note: the current settings are stored in NVM.
	
	Note: while the SMS Run AT service executes a command that takes long time to get the response, a new command will be pending until the module has finished sending all of its response result code.

AT#SMSATRUN?	Read command returns the current settings of <mode> and the value of <stat> in the format:
	
	#SMSATRUN: <mod>,<stat>
	
	   where:
	   <stat> - service status
	    0 – not active
	    1 - active

AT#SMSATRUN =?	Test command returns the supported values for the SMSATRUN parameters
Notes:	• By default the SMS ATRUN service is disabled It can be activated either by the command AT#SMSATRUN or receiving a special SMS that can be sent from a Telit server.

3.5.6.10.2. *Set SMS Run AT Service parameters -*
   #SMSATRUNCFG

#SMSATRUNCFG – Set SMS AT Run Parameters	
AT#SMSATRUNCFG=	Set command configures the SMS AT RUN service.

#SMSATRUNCFG – Set SMS AT Run Parameters	
<instance> [,<urcmod> [,<timeout>]]	Parameter: **<instance>:** AT instance that will be used by the service to run the AT Command. Range 1- 3, default 3. Note: In CE910 family, <**instance**> parameter is not supported and SMS Run AT service has its independent channel. This parameter is dummy for unified policy. **<urcmod>:** 0 – disable unsolicited message 1 - enable an unsolicited message when an AT command is requested via SMS (default). When unsolicited is enabled, the AT Command requested via SMS is indicated to TE with unsolicited result code: #SMSATRUN: <Text> e.g.: #SMSATRUN: AT+CGMR;+CGSN;+GSN;+CCLK Unsolicited is dumped on the instance that requested the service activation. **<timeout>:** It defines in minutes the maximum time for a command execution. If timeout expires the module will be rebooted. Range 1 – 60, default 5. Note 1: the current settings are stored in NVM. Note 2: SMS Run AT service and EvMoni service share the same channel. For the unified policy, when the #SMSATRUNCFG sets the <**instance**> parameter, the change is reflected also in the <**instance**> parameter of the #ENAEVMONICFG command, and viceversa. Note 3: the set command returns **ERROR** if the command AT#ENAEVMONI? returns 1 as <**mod**> parameter or the command AT#SMSATRUN? returns 1 as <**mod**> parameter
AT#SMSATRUNCFG?	Read command returns the current settings of parameters in the format: **#SMSATRUNCFG:<instance>,<urcmod>,<timeout>**
AT#SMSATRUNCFG=?	Test command returns the supported values for the SMSATRUNCFG parameters

3.5.6.10.3.   *SMS AT Run White List - #SMSATWL*

#SMSATWL – SMS AT Run White List	
AT#SMSATWL= <action> ,<index> [,<entryType> [,<string>]]	Set command to handle the white list. **<action >:** 0 – Add an element to the WhiteList 1 – Delete an element from the WhiteList 2 – Print and element of the WhiteList **< index >:** Index of the WhiteList. Range 1-8 **< entryType >:** 0 – Phone Number 1 – Password NOTE: A maximum of two Password Entry can be present at same time in the white List **<string>:** string parameter enclosed between double quotes containing or the phone number or the password Phone number shall contain numerical characters and/or the character "+" at the beginning of the string and/or the character "*" at the end of the string. Password shall be 16 characters length NOTE: When the character "*" is used, it means that all the numbers that begin with the defined digit are part of the white list. E.g. "+39*" All Italian users can ask to run AT Command via SMS "+39349*" All vodafone users can ask to run AT Command via SMS.
AT#SMSATWL?	Read command returns the list elements in the format: **#SMSATWL: [<entryType>,<string>]**
AT#SMSATWL=?	Test command returns the supported values for the parameter **<action>, <index>** and **<entryType>**
Note	It will return ERROR if executed using SMSATRUN digest mode or TCPATRUN server mode

3.5.6.10.4. *Set TCP Run AT Service parameter – #TCPATRUNCFG*

#TCPATRUNCFG – Set TCP AT Run Service Parameters	
AT#TCPATRUNCFG= <connId> ,<instance> ,<tcpPort> ,<tcpHostPort>	Set command configures the TCP AT RUN service. Parameters: **<connId>** Socket connection identifier. Default 1.

#TCPATRUNCFG – Set TCP AT Run Service Parameters	

,<tcpHost>
[,<urcmod>
[,<timeout>
[,<authMode>
[,<retryCnt>
[,<retryDelay>]]]]]
	Range 1..6. This parameter is mandatory.
	
	<instance>
	AT instance that will be used by the service to run the AT Command. Default 2. Range 1 – 3. This parameter is mandatory.
	
	Note: In CE910 family, <instance> parameter is not supported and TCP Run AT service has its independent channel. This parameter is dummy for unified policy.
	
	<tcpPort>
	TCP listen port for the connection to the service in server mode. Default 1024. Range 1..65535. This parameter is mandatory.
	
	<tcpHostPort>
	TCP remote port of the Host to connect to, in client mode. Default 1024. Range 1..65535. This parameter is mandatory.
	
	<tcpHost>
	IP address of the 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
	This paramteter is mandatory. Default "".
	
	<urcmod>
	        0 – disable unsolicited messages
	        1 – enale an unsolicitied message when the TCP socket is connected or disconnect (default).
	
	When unsolicited is enabled, an asynchronous TCP Socket connection is indicated to TE with unsolicited result code:
	
	#TCPATRUN: <iphostaddress>
	
	When unsolicited is enabled, the TCP socket disconnection is indicated to TE with unsolicited result code:
	
	#TCPATRUN: <DISCONNECT>
	
	Unsolicited is dumped on the instance that requested the service activation.
	
	<timeout>
	Define in minutes the maximum time for a command execution. If timeout expires the module will be rebooted. The default value is 5 minutes. Range 1…5.
	
	<authMode>
	Determines the authentication procedure in server mode:
	        0 – when connection is up, username and password (in this order and

#TCPATRUNCFG – Set TCP AT Run Service Parameters	
	each of them followed by a Carriage Return) have to be sent to the module before the first AT command. (default) 1 – when connection is up, the user receives a request for username and, if username is correct, a request for password. Then a message of "Login successful" will close authentication phase. Note: if username and/or password are not allowed (see **AT#TCPATRUNAUTH**) the connection will close immediately. **<retryCnt>** in client mode, at boot or after a socket disconnection, this parameter represents the number of attempts that are made in order to re-connect to the Host. Default: 0. Range 0…5. **<retryDelay>** in client mode, delay between one attempt and the other. In minutes. Default: 2. Range 1…3600. Note: the current settings are stored in NVM. Note: to start automatically the service when the module is powered-on, the automatic PDP context activation has to be set (see **AT#SGACTCFG** command). Note : the set command returns **ERROR** if the command **AT#TCPATRUNL**? returns 1 as **<mod>** parameter or the command **AT#TCPATRUND**? returns 1 as **<mod>** parameter
AT#TCPATRUNCFG?	Read command returns the current settings of parameters in the format: **#TCPATRUNCFG:** **<connId>,<instance>,<tcpPort>,<tcpHostPort>,<tcpHost>,<urcmod>,<timeout>,<authMode>,<retryCnt>,<retryDelay>**
AT#TCPATRUNCFG=?	Test command returns the supported values for the TCPATRUNCFG parameters.

3.5.6.10.5. *TCP Run AT Service in listen (server) mode – #TCPATRUNL*

#TCPATRUNL – Enables TCP AT Run Service in listen (server) mode	

AT#TCPATRUNL= <mod>	Set command enables/disables the TCP AT RUN service in server mode. When this service is enabled, the module tries to put itself in TCP listen state.
	
	Parameter:
	<mod >
	    0 – Service Disabled
	    1 – Service Enabled

#TCPATRUNL – Enables TCP AT Run Service in listen (server) mode	
	Note: the current settings are stored in NVM. Note: to start automatically the service when the module is powered-on, the automatic PDP context activation has to be set (see **AT#SGACTCFG** command). Note: while the TCP Run AT service executes a command that takes long time to get the response, a new command will be pending until the module has finished sending all of its response result code.

AT#TCPATRUNL?	Read command returns the current settings of <mode> and the value of <stat> in the format:
	
	#TCPATRUNL: <mod>,<stat>
	
	   where:
	   <stat> - connection status
	    0 – not in listen
	    1 – in listen or active

AT#TCPATRUNL=?	Test command returns the supported values for the **TCPATRUNL** parameters

3.5.6.10.6.   *TCP AT Run Firewall List - #TCPATRUNFRWL*

#TCPATRUNFRWL – TCP AT Run Firewall List	

AT#TCPATRUNFRWL=
<action>,
<ip_addr>,
<net_mask>
	Set command controls the internal firewall settings for the TCPATRUN connection.
	
	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 tp 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.
	
	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

#TCPATRUNFRWL – TCP AT Run Firewall List	
	rules will be scanned for matching with the following criteria: **incoming_IP & <net_mask> = <ip_addr> & <net_mask>** If criteria is matched, then the packet is accepted and the rule scan is finished; if criteria is not matched for any chain the packet is silently dropped. Note: A maximum of 5 firewall can be present at same time in the List. Note: the firewall list is saved in NVM
AT#TCPATRUNFRWL?	Read command reports the list of all ACCEPT chain rules registered in the firewall setting in the format: **#TCPATRUNFRWL: <ip_addr>,<net_mask>** **#TCPATRUNFRWL: <ip_addr>,<net_mask>** … **OK**
AT#TCPATRUNFRWL=?	Test command returns the allowed values for parameter <**action**>.
Note	It will return ERROR if executed using SMSATRUN digest mode or TCPATRUN server mode

3.5.6.10.7. *TCP AT Run Authentication Parameters List – #TCPATRUNAUTH*

#TCPATRUNAUTH – TCP AT Run Authentication Parameters List	

AT#TCPATRUNAUTH=
<action>,
<userid>,
<passw>
	Execution command controls the authentication parameters for the TCPATRUN connection
	
	Parameters:
	<action>
	Command action
	      0 – remove selected chain
	      1 – add an **ACCEPT** chain
	      2 – remove all chains (**DROP** everything);
	          <**userid** > and <**passw**> has no meaning in this case.
	
	<userid>
	User to be added into the **ACCEPT** chain;
	string type, maximum length 50
	
	<passw>
	Password of the user on the <**userid**>;
	string type, maximum length 50
	
	Command returns **OK** result code if successful.
	
	Note: A maximum of 3 entry (password and userid) can be present at same

#TCPATRUNAUTH – TCP AT Run Authentication Parameters List	
	time in the List. Note: The Authentication Parameters List is saved in NVM.
AT#TCPATRUNAUTH?	Read command reports the list of all ACCEPT chain rules registered in the firewall setting in the format: **#TCPATRUNAUTH: <userid>,<passw>** **#TCPATRUNAUTH: <userid>,<passw>** … **OK**
AT#TCPATRUNAUTH=?	Test command returns the allowed values for parameter <**action**>.

3.5.6.10.8.   *TCP AT Run in dial (client) mode - #TCPATRUND*

#TCPATRUND – Enable TCP AT Run Service in dial (client) mode	

**AT#TCPATRUND=
<mod>**
	Set command enables/disables the TCP AT RUN service in client mode.
	When this service is enabled, the module tries to open a connection to the Host (the Host is specified in **AT#TCPATRUNCFG**).
	
	Parameter:
	< mod >
	      0: Service Disabled
	      1: Service Enabled
	
	Note: The current setting are stored in NVM
	
	Note: to start automatically the service when the module is powered-on, the automatic PDP context activation has to be set (see **AT#SGACTCFG** command).
	
	Note: If the connection closes or at boot, if service is enabled and context is active, the module will try to reconnect for the number of attempts specified in **AT#TCPATRUNCFG** also the delay between one attempt and the other will be the one specified in **AT#TCPATRUNCFG**.
	
	Note: while the TCP Run AT service executes a command that takes long time to get the response, a new command will be pending until the module has finished sending all of its response result code.

AT#TCPATRUND?	Read command returns the current settings of <**mode**> and the value of <**stat**> in the format:
	
	#TCPATRUND: <mod>,<stat>
	
	    where:
	    <stat> - connection status
	      0 – not connected

#TCPATRUND – Enable TCP AT Run Service in dial (client) mode	
	1 – connected or connecting at socket level
	2 – not connected but still trying to connect, attempting every delay time (specified in **AT#TCPATRUNCFG**)
AT#TCPATRUND =?	Test command returns the supported values for the **TCPATRUND** parameters

3.5.6.10.9.   *Closing TCP Run AT Socket - #TCPATRUNCLOSE*

#TCPATRUNCLOSE – Closes TCP Run AT Socket	
AT#TCPATRUNCLOSE	Closes the socket used by TCP ATRUN service. Note: TCP ATRUN status is still enabled after this command, so the service re-starts automatically.
AT#TCPATRUNCLOSE=?	Test command returns **OK**

3.5.6.10.10.   *TCP AT Run Command Sequence - #TCPATCMDSEQ*

#TCPATCMDSEQ – For TCP Run AT Service, allows the user to give AT commands in sequence	

AT#TCPATCMDSEQ= <mod>	Set command enable/dsable, for TCP Run AT service, a feature that allows giving more than one AT command without waiting for responses.
	It does not work with commands that uses the prompt '>' to receive the message body text (e.g. "**AT+CMGS**")
	
	Parameter:
	< mod >
	     0 - Service Disabled (default)
	     1 - Service Enabled

AT#TCPATCMDSEQ?	Read command returns the current settings of parameters in the format: **#TCPATCMDSEQ: <mod>**
AT#TCPATCMDSEQ=?	Test command returns the supported values for the **TCPATCMDSEQ** parameters.

3.5.6.10.11.   *TCP Run AT service to a serial port – #TCPATCONSER*

#TCPATCONSER – Connects the TCP Run AT service to a serial port	

AT#TCPATCONSER= <port>, <rate>	Set command sets the TCP Run AT in transparent mode, in order to have direct access to the hardware port specified. Data will be transferred directly, without being elaborated, between the TCP Run AT service and the hardware port specified.
	
	Parameters:
	< port >
	     0 – UART Data Port
	     1 – Telit USB Modem Port
	
	Not all of these ports will be available at the same time.
	The port available will be displayed by the test command.

#TCPATCONSER – Connects the TCP Run AT service to a serial port	
	<**rate**> Baud rate for data transfer. Allowed values are 300,1200,2400,4800,9600,19200,38400,57600,115200. Note: The command has to be issued from the TCP ATRUN instance Note: After this command has been issued, if no error has occurred, then a "**CONNECT**" will be returned by the module to advise that the TCP ATRUN instance is in **online mode** and connected to the port specified. Note: To exit from **online mode** and close the connection, the escape sequence (+++) has to be sent on the TCP ATRUN instance. The escape sequence needs to be sent in one single packet. The use of Telnet for Windows sending every single byte in a TCP packet is not appropriate to perform this connection.
AT#TCPATCONSER=?	Test command returns the supported values for the #**TCPATCONSER** parameters

3.5.6.10.12.   *Run AT command execution - #ATRUNDELAY*

#ATRUNDELAY – Set the delay on Run AT command execution	
AT#ATRUNDELAY= <**srv**>, <**delay**>	It has no effect and is included only for backward compatibility. Parameters: **< srv >** 0 – TCP Run AT service 1 – SMS Run AT service <**delay**> Value of the delay, in seconds. Range 0..30. Default value 0 for both services (TCP and SMS).
AT#ATRUNDELAY?	Read command returns the current settings of parameters in the format: #**ATRUNDELAY: 0, <delayTCP>** #**ATRUNDELAY: 1, <delaySMS>** **OK**
AT#ATRUNDELAY=?	Test command returns the supported values for the **ATRUNDELAY** parameters

3.5.6.11. Event Monitor Commands

3.5.6.11.1.   *Enable EvMoni Service - #ENAEVMONI*

#ENAEVMONI – Enable EvMoni Service	

#ENAEVMONI – Enable EvMoni Service	

AT#ENAEVMONI=
<mod>
	Set command enables/disables the EvMoni service.
	
	Parameter:
	<mod>
	      0 – Service Disabled (default)
	      1 – Service Enabled
	
	Note: The current settings are stored in NVM.
	
	Note: while the Event Monitor service executes a command that takes long time to get the response, a new command will be pending until the module has finished sending all of its response result code.

AT#ENAEVMONI?	Read command returns the current settings of <mode> and the value of <stat> in the format:
	
	#ENAEVMONI: <mod>,<stat>
	
	  where:
	  <stat> - service status
	    0 – not active (default)
	    1 – active

AT#ENAEVMONI=?	Test command returns the supported values for the **ENAEVMONI** parameters

3.5.6.11.2. *EvMoni Service parameter - #ENAEVMONICFG*

#ENAEVMONICFG – Set EvMoni Service Parameters	

AT#ENAEVMONICFG=
<instance>
[,<urcmod>
[,<timeout>]]
	Set command configures the EvMoni service.
	
	Parameters:
	
	<**instance**>
	AT instance that will be used by the service to run the AT Command.
	Range 1-3. (Default: 3)
	
	Note: In CE910 family, <**instance**> parameter is not supported and EvMoni service share the same channel with SMS Run AT service. This parameter is dummy for unified policy.
	
	<urcmod>
	      0 – disable unsolicited message
	      1 – enable an unsolicited message when an AT command is executed
	         after an event is occurred (default)
	
	When unsolicited is enabled, the AT Command is indicated to TE with unsolicited result code:
	
	#EVMONI: <TEXT>

#ENAEVMONICFG – Set EvMoni Service Parameters	
	e.g.: **#EVMONI: AT+CGMR;+CGSN;+GSN;+CCLK** Unsolicited is dumped on the instance that requested the service activation. <**timeout**> It defines in minutes the maximum time for a command execution. If timeout Expires the module will be rebooted. (Default: 5) Note: The current settings are stored in NVM. Note 2: EvMoni service and SMS Run AT service share the same channel. For the unified policy, when the **#ENAEVMONICFG** sets the <**instance**> parameter, the change is reflected also in the <**instance**> parameter of the **#SMSATRUNCFG** command, and viceversa. Note: The set command returns ERROR if the command **AT#ENAEVMONI?** Retirms 1 as <**mod**> parameter or the command **AT#SMSATRUN?** Returns 1 as <**mod**> parameter.
AT#ENAEVMONICFG?	Read command returns the current settings of parameters in the format: **#ENAEVMONICFG: <instance>,<urcmode>,<timeout>**
AT#ENAEVMONICFG=?	Test command returns the supported values for the **ENAEVMONICFG** parameters

3.5.6.11.3.   *Event Monitoring - #EVMONI*

#EVMONI – Set the single Event Monitoring	
AT#EVMONI= <label> ,<mode> [,<paramType> ,<param>]	Set command enables/disables the single event monitoring, configures the related parameter and associates the AT command <label> String parameter (that has to be enclosed between double quotes) indicating the event under monitoring. It can assume the following values: • VBATT - battery voltage monitoring • DTR - DTR monitoring • CONTDEACT - context deactivation monitoring • RING - call ringing monitoring • STARTUP – module start-up monitoring • REGISTERED – network registration monitoring • GPIO1 – monitoring on a selected GPIO in the GPIO range • GPIO2 – monitoring on a selected GPIO in the GPIO range • GPIO3 – monitoring on a selected GPIO in the GPIO range • GPIO4 – monitoring on a selected GPIO in the GPIO range • GPIO5 – monitoring on a selected GPIO in the GPIO range • ADCH1 – ADC High Voltage monitoring • ADCL1 – ADC Low Voltage monitoring • DTMF1 –monitoring on user defined DTMF string

#EVMONI – Set the single Event Monitoring	

	• DTMF2 –monitoring on user defined DTMF string
	• DTMF3 –monitoring on user defined DTMF string
	• DTMF4 –monitoring on user defined DTMF string
	• SMSIN – monitoring on incoming SMS
	
	<mode>
	      0 – disable the single event monitoring (default)
	      1 – enable the single event monitoring
	
	< paramType >
	Numeric parameter indicating the type of parameter contained in **<param>**. The 0 value indicates that **<param>** contains the AT command string to execute when the related event has occurred. Other values depend from the type of event.
	
	<param>
	It can be a numeric or string value depending on the value of **<paramType>** and on the type of event.
	If **<paramType>** is 0, then **<param>** is a string containing the AT command:
	
	  • It has to be enclosed between double quotes
	  • It has to start with the 2 chars AT (or at)
	  • If the string contains the character ", then it has to be replaced with the 3 characters \22
	  • the max string length is 96 characters
	  • if it is an empty string, then the AT command is erased
	
	  • If **<label>** is VBATT, **<paramType>** can assume values in the range 0 - 2.
	     o if **<paramType>** = 1, **<param>** indicates the battery voltage threshold in the range 0 – 500, where one unit corresponds to 10 mV (therefore 500 corresponds to 5 V). (Default: 0)
	     o if **<paramType>** = 2, **<param>** indicates the time interval in seconds after that the voltage battery under the value specified with **<paramType>** = 1 causes the event. The range is 0 – 255. (Default: 0)
	  • If **<label>** is DTR, **<paramType>** can assume values in the range 0 - 2.
	     o if **<paramType>** = 1, **<param>** indicates the status high or low under monitoring. The values are 0 (low) and 1 (high). (Default: 0)
	     o if **<paramType>** = 2, **<param>** indicates the time interval in seconds after that the DTR in the status specified with **<paramType>** = 1 causes the event. The range is 0 – 255. (Default: 0)
	  • If **<label>** is CONTDEACT, **<paramType>** can assume only the value 0. The event under monitoring is the context deactivation.
	  • If **<label>** is RING, **<paramType>** can assume values in the range 0 - 1.

#EVMONI – Set the single Event Monitoring	

	o if **\<paramType\>** = 1, **\<param\>** indicates the numbers of call rings after that the event occurs. The range is 1-50. (Default: 1)
	• If **\<label\>** is STARTUP, **\<paramType\>** can assume only the value 0. The event under monitoring is the module start-up.
	• If **\<label\>** is REGISTERED, **\<paramType\>** can assume only the value 0. The event under monitoring is the network registration (to home network or in roaming) after the start-up and the SMS ordening.
	• If **\<label\>** is GPIOX, **\<paramType\>** can assume values in the range 0 - 3.
	  o if **\<paramType\>** = 1, **\<param\>** indicates the GPIO pin number; supported range is from 1 to a value that depends on the hardware. (Default: 1)
	  o if **\<paramType\>** = 2, **\<param\>** indicates the status high or low under monitoring. The values are 0 (low) and 1 (high) . (Default: 0)
	  o if **\<paramType\>** = 3, **\<param\>** indicates the time interval in seconds after that the selected GPIO pin in the status specified with **\<paramType\>** = 1 causes the event. The range is 0 – 255. (Default: 0)
	• If **\<label\>** is ADCH1, **\<paramType\>** can assume values in the range 0 - 3.
	  o if **\<paramType\>** = 1, **\<param\>** indicates the ADC pin number; supported range is from 1 to a value that depends on the hardware. (Default: 1)
	  o if **\<paramType\>** = 2, **\<param\>** indicates the ADC High voltage threshold in the range 0 – 2000 mV. (Default: 0)
	  o if **\<paramType\>** = 3, **\<param\>** indicates the time interval in seconds after that the selected ADC pin above the value specified with **\<paramType\>** = 1 causes the event. The range is 0 – 255. (Default: 0)
	• If **\<label\>** is ADCL1, **\<paramType\>** can assume values in the range 0 - 3.
	  o if **\<paramType\>** = 1, **\<param\>** indicates the ADC pin number; supported range is from 1 to a value that depends on the hardware. (Default: 1)
	  o if **\<paramType\>** = 2, **\<param\>** indicates the ADC Low voltage threshold in the range 0 – 2000 mV. (Default: 0)
	  o if **\<paramType\>** = 3, **\<param\>** indicates the time interval in seconds after that the selected ADC pin under the value specified with **\<paramType\>** = 1 causes the event. The range is 0 – 255. (Default: 0)
	• If **\<label\>** is DTMFX, **\<paramType\>** can assume values in the range 0 - 2.
	  o if **\<paramType\>** = 1, **\<param\>** indicates the DTMF string; the single DTMF characters have to belong to the range ((0-9),#,*,(A-D)); the maximum number of characters in the string is 15
	  o if **\<paramType\>** = 2, **\<param\>** indicates the timeout in milliseconds. It is the maximum time interval within which a DTMF tone must be detected after detecting the previous one, to be

#EVMONI – Set the single Event Monitoring	
	considered as belonging to the DTMF string. The range is (500 – 5000). (Default: 1000) • If **\<label\>** is SMSIN, **\<paramType\>** can assume values in the range 0 - 1. o if **\<paramType\>** = 1, **\<param\>** indicates the text that must be received in incoming SMS to trigger AT command execution rings after that the event occurs; the maximum number of characters in the SMS text string is 15 Note: the DTMF string monitoring is available only if the DTMF decode has been enabled (see **#DTMF** command)
AT#EVMONI?	Read command returns the current settings for each event in the format: **#EVMONI:** **\<label\>,\<mode\>,\<param0\>[,\<param1\>[,\<param2\>[,\<param3\>]]]** Where **\<param0\>,\<param1\>,\<param2\>** and **\<param3\>** are defined as before for \<param\> depending on \<label\> value
AT#EVMONI=?	Test command returns values supported as a compound value

3.5.6.11.4. *Send Message - #CMGS*

#CMGS - Send Message	
(PDU Mode) **AT#CMGS=** **\<length\>,\<pdu\>**	**(PDU Mode)** Execution command sends to the network a message. Parameter: **\<length\>** - length of the PDU to be sent in bytes (excluding the Destination Address octets) 5..183 **\<pdu\>** - PDU in hexadecimal format (each octet of the PDU is given as two IRA character long hexadecimal number) and given in one line. If message is successfully sent to the network, then the result is sent in the format: **#CMGS: \<mr\>** where **\<mr\>** - message reference number Note: if message sending fails for some reason, an error code is reported. Note: The limit of user data is 160 characters.
(Text Mode) **AT#CMGS=\<da\>**	**(Text Mode)** Execution command sends to the network a message.

#CMGS - Send Message	
,<text>	Parameters: **<da>** - destination address, string type represented in the currently selected character set (see **+CSCS**). **<text>** - text to send If message is successfully sent to the network, then the result is sent in the format: **#CMGS: <mr>** where **<mr>** - message reference number Note: if message sending fails for some reason, an error code is reported. Note: The limit of user data is 160 characters.
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. If the EvMoni service is enabled, **<label> of #EVMONI** is **RING**, **<param>** is **#CMGS** and a RING event occurs in call incoming state, the RING event message can't be reported with **#CMGS** because of having the limitation of 3GPP2 SMS.

3.5.6.11.5.   *Write Message To Memory - #CMGW*

#CMGW - Write Message To Memory	
(PDU Mode) **AT#CMGW=** **<length>,<pdu>**	**(PDU Mode)** Execution command writes in the **<memw>** memory storage a new message. Parameter: **<length>** - length in bytes of the PDU to be written. (excluding the Destination Address octets) 5..183 **<pdu>** - PDU in hexadecimal format (each octet of the PDU is given as two IRA character long hexadecimal number) and given in one line. 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>**.

#CMGW - Write Message To Memory	
	If message storing fails for some reason, an error code is reported. Note: The limit of user data is 160 characters
(Text Mode) **AT#CMGW=<da>,<text>**	**(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**). **<text>** - text to write 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 is reported. Note: The limit of user data is 160 characters.
AT#CMGW=?	Test command returns the **OK** result code.
Note	To avoid malfunctions is suggested to wait for the **#CMGW: <index>** or **+CMS ERROR: <err>** response before issuing further commands.

3.5.6.12. Telit IoT Portal Commands
The following AT commands regard the deviceWISE functionality.

Here is a basic interaction diagram:
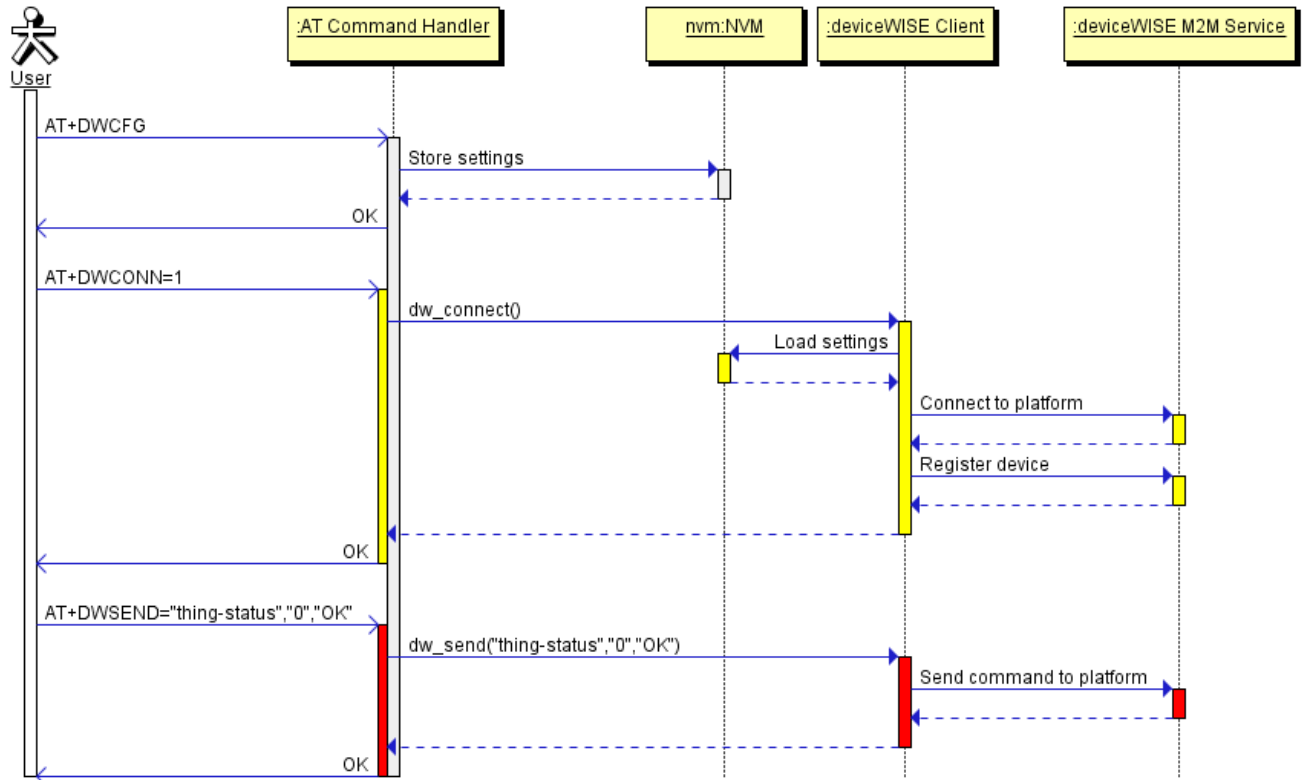
3.5.6.12.1. *Configure deviceWISE parameters - #DWCFG*

#DWCFG – Configure deviceWISE parameters	

AT#DWCFG=[<serverUrl>[,<deviceIDSelector>[,<appToken>[,<security>[,<heartBeat>[,<autoReconnect>[,<overflowHandling>[,<atrunInstanceId>[,<serviceTimeout>[,<contextID>[,<unused_2>[,<unused_3>]]]]]]]]]]]]]	This command sets the parameters related to the deviceWISE functionality
	
	Parameters:
	<serverUrl> - String parameter indicating the URL of the M2M Service instance in address:port form.
	
	<deviceIDSelector> 0 – 1 (0=IMEI 1=CCID/ESN), basically 0 if not SIM card or CDMA ID installed
	
	<appToken> - The secure application token provided in the Management Portal, typically a string of 16 characters..
	
	<security> - Flag indicating if the SSL encryption is enabled.
	  0 – SSL encryption disabled (default)
	  1 – SSL encryption enabled (not yet implemented and not available for setting)
	
	If SSL encryption enabling is required, some initial settings have to be done as follows. For further details, refer to "SSL/TLS User Guide".
	
	SSL channel has to be enabled as follows:

#DWCFG – Configure deviceWISE parameters	

	AT#SSLEN=1,1
	OK
	
	If server authentication is needed, **#SSLSECCFG** has to be set
	as follows:
	
	AT#SSLSECCFG=1,0,1,0
	OK
	
	Then, CA Certificate(DER format) has to be stored as follows:
	
	AT#SSLSECDATA=1,1,1,<size>
	>
	…………………. // store CA Certificate
	OK
	
	Note: Only the configuration SSL commands listed above are admitted. DW connection in secure mode cannot be used contemporarily to any command starting an SSL connection (including SSL sockets, FTPS, secure SMTP and HTTPS).
	
	<heartBeat> - If no packets are received in the number of seconds specified in the heartbeat field, a heartbeat message will be sent to keep the connection alive.
	  Default: 60
	  Range: 10 - 86400
	
	<autoReconnect> - Flag indicating if the connection manager should automatically reconnect to the service.
	0 – auto-reconnect disabled
	1 – auto-reconnect lazy - reconnect on next send and every 3600 seconds.
	2 – auto-reconnect moderate (default) - reconnect 120 seconds, then every 3600 seconds after the first day.
	3 – auto-reconnect aggressive - reconnect every 120 seconds.
	
	<overflowHandling> - Flag indicating if the way to handle overflows in data management.
	0 – FIFO (default)
	1 – LIFO
	
	<atrunInstanceId> - AT instance that will be used by the service to run the AT Command.
	Default 2
	Range 1 – 3
	
	Note: In CE910 family, **<atrunInstanceId>** parameter is not supported and

#DWCFG – Configure deviceWISE parameters	
	deviceWISE service has its independent channel. This parameter is dummy for unified policy. **<serviceTimeout>** - It defines in seconds the maximum time interval for a service request to the server. Default 5 Range 1 – 120 **<contextID>** - PDP context identifier 1 – numeric parameter which specifies a particular PDP context definition
AT#DWCFG?	Read command returns the current settings in the format: **#DWCFG: <serverUrl>,<deviceIDSelector>,<appToken>,<security>,<heartBeat>,<autoReconnect>,<overflowHandling>,<atrunInstanceId>,<serviceTimeout>,<contextID>,0,0**
AT#DWCFG=?	Test command returns the supported range of parameters **<deviceIDSelector>, <security>, <heartBeat>, <AutoReconnect>,<overflowHandling>,<atrunInstanceId>, <serviceTimeout>** and **<contextID>,** and the maximum length of **<serverUrl> and <appToken>** parameters.

3.5.6.12.2. *Connect to M2M Service - #DWCONN*

#DWCONN – Connect to M2M Service	

AT#DWCONN=<connect>	Set command connects/disconnects to the M2M Service.
	
	Parameters:
	<connect> - flag to connect/disconnect to the M2M Service
	  0 – disconnect (default)
	  1 – connect
	
	Note: **AT#DWCONN=1** performs the socket connection and the MQTT connection. **AT#DWCONN=0** performs the socket disconnection.
	
	Note: the PDP Context used for the network connection is the one set by parameter **<contextID>** of **AT#DWCFG** command. It has to be previously activated with **AT#SGACT** command.
	
	Note: if the secure mode connection has been enabled, it cannot be used contemporarily to any command starting an SSL connection (including SSL sockets, FTPS, secure SMTP and HTPS).

AT#DWCONN?	Read command returns the current settings for all parameters in the format:

#DWCONN – Connect to M2M Service

	#DWCONN: <connect>>,<status> Where: **<connect>** is defined as above **<status>** is the real connection status. Values: 0 = disconnected 1 = trying to connect 2 = connected 3 = waiting to connect
AT#DWCONN=?	Test command reports the supported range of values for all parameters

3.5.6.12.3. *Query connection status - #DWSTATUS*

#DWSTATUS – query connection status

AT#DWSTATUS	Execution command returns the status of the connection, including some runtime statistics. Note, all statistics should be stored in RAM, not NVM. The Cloud will return a generic structure **#DWSTATUS:** **<connected><lastErrorCode>,<latency>,<pktsIn>,<pktsOut>,<bytesIn>, <bytesOut>** **<connected>** : 3 = waiting to connect, 2 = connected, 1 = trying to connect, 0 = disconnected **<lastErrorCode**>: last error code encountered by the client **<latency> :** milliseconds measured between last request and reply. **<pktsIn> :** number of packets received, tracked by the server **<pktsOut> :** number of packets sent. **<bytesIn>** : number of bytes received, TCP/IP payload **<bytesOut>** : number of bytes sent.
AT#DWSTATUS=?	Test command reports **OK** result code

3.5.6.12.4. *Send data to M2M Service - #DWSEND*

#DWSEND - send data to M2M Service

AT#DWSEND=<type>,<param_>[,<param_2>[,…[<param_n>]]]	Execution command permits to send formatted data to the M2M Service. Parameters: **<type> -** type code for the type of message to send. (0 for normal request; 1 for method request; 2 for method update; 3 for method ack)

#DWSEND - send data to M2M Service	
	Type 0 message format: **\<param_1\>** - command – the api command to execute. **\<param_i\>** - string parameter indicating the i-th parameter, with i=1,…,24. **Type 1 message format:** **\<param_1\>** - "thingKey" – the key of a thing to execute. **\<param_2\>** - timeout – time to wait in seconds before returning an error for the request. **\<param_3\>** - method – the method key of a thing to execute. **\<param_4\>** - is singleton – 0 or 1. 1 if no more than one of these instances can exist. **\<param_5+\>** - parameters for the method. String parameter indicating the i-th parameter, with i=1,…,20. **Type 2 message format:** **\<param_1\>** - id – the identification of the method instance. **\<param_2\>** - message – a message represents the current status of the method. **Type 3 message format:** **\<param_1\>** - id – the identification of the method instance. **\<param_2\>** - status – the integer result status for the execution. 0 is reserved for OK. \<**param_3 when status is set to non-zero**\> - error message associated with the status. \<**param_3 when status is set to zero**\> - return parameters for the method. Key value pair should be used. **param_i** should be the name of the element and **param_i+1** should be the value of the element. Note: there is no limit on the length of the single **\<param_i\>**, but there is a limit in the total length of the AT command string, that cannot exceed 400 characters. If this threshold is exceeded, then an ERROR is raised. There is also a limit of 20 messages on the receive queue. If the queue is full, the consequent send will still succeed but the response for that particular request will be dropped until an item is removed from this queue (See command AT#DWRCV and AT#DWRCVR). Note: the response to the **AT#DWSEND** command reports the **\<msgId\>** value that identifies the sending. Note: 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 **AT#DWSEND** only if the connection has been

#DWSEND - send data to M2M Service	
	opened with **AT#DWCONN**
AT#DWSEND=?	Test command reports the maximum length of **\<type\>** parameter.

3.5.6.12.5. *Send raw data to M2M Service - #DWSENDR*

#DWSENDR – send raw data to M2M Service	
AT#DWSENDR=\<dataLen\>	Execution command permits to send raw data to the M2M Service. Content must be valid JSON. Parameters: **\<dataLen\>** - number of bytes to be sent Range: 1 - 1500 The module responds to the command with the prompt \<greater_than\>\<space\> and waits for the data to send. When **\<dataLen\>** 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: the response to the **AT#DWSENDR** command reports the **\<msgId\>** value that identifies the sending. There is also a limit of 20 messages on the receive queue. If the queue is full, the consequent send will still succeed but the response for that particular request will be dropped until an item is removed from this queue (See command AT#DWRCV and AT#DWRCVR). Note: it's possible to use **AT#DWSENDR** only if the connection has been opened with **AT#DWCONN**
AT#DWSENDR=?	Test command reports the supported range of values for **\<dataLen\>** parameter

3.5.6.12.6. *Receive data from M2M Service - #DWRCV*

#DWRCV – Receive data from M2M Service	
AT#DWRCV=\<msgId\>	Execution command permits the user to read formatted data arriving from M2M Service; the module is notified of these data by the URC **#DWRING**. Parameters: **\<msgId\>** - index of the data message to receive, as indicated in the URC **#DWRING** Range: \>=1 If the received data are the consequence of a previous data sending issued by **AT#DWSEND**, then the **\<msgId\>** value is the same of the **\<msgId\>** value reported in the answer of **AT#DWSEND**.

#DWRCV – Receive data from M2M Service	
	The incoming Server data are notified by the URC **#DWRING** with the following format: **#DWRING: <type>,<msgId>,<len>** where: **<type> -** type of message to receive **<msgId>** - index of the data message to receive **<len>** - length of data message to receive If the incoming data are accepted with **AT#DWRCV**, then the formatted data are received and showed with the following URC: **#DWDATA:** **<msgId>,<error>,<len>,<param_1>[,<param_2>[,…[,<param_n>]]]** where: **<msgId>** - defined as above <*error*> - error code of the message to receive, 0 if there is no error. **<len>** - defined as above **<param_i>** - string parameter indicating the i-th parameter associated to the type specified Note: it is possible to use **AT#DWRCV** only if the connection has been opened with **AT#DWCONN**, else the ME is raising an error. If the data received are the consequence of a previous data sending issued by **AT#DWSEND**, then they can be read only using **AT#DWRCV** command and not **AT#DWRCVR** command (i.e.: **AT#DWRCV** and **AT#DWRCVR** are not interchangeable).
AT#DWRCV=?	Test command reports the supported range of values for all parameters.

3.5.6.12.7.   *Receive raw data from M2M Service - #DWRCVR*

#DWRCVR – Receive raw data from M2M Service	
AT#DWRCVR=<msgId>	Execution command permits the user to read raw data arriving from M2M Service; the module is notified of these data by the URC **#DWRING**. Parameters: **<msgId>** - index of the data message to receive, as indicated in the URC **#DWRING** Range: >=1 If the data received are the consequence of a previous data sending (issued by **AT#DWSENDR**), then the **<msgId>** value is the same of the **<msgId>** value

#DWRCVR – Receive raw data from M2M Service	
	reported in the answer of **AT#DWSENDR**.
	The incoming Server data are notified by the URC **#DWRING** with the following format:
	#DWRING: <type>,<msgId>,<len>
	where: **<type>** - type of the data message to receive **<msgId>** - index of the data message to receive **<len>** - length of data message to receive
	If the incoming data are accepted with **AT#DWRCVR**, then the data are received and showed with the following URC:
	#DWRDATA: <msgId>,<error>,<len>,<data>
	where: **<msgId>** - defined as above **<error>** - error code of the message to receive, 0 if there is no error. **<len>** - defined as above **<data>** - M2M Service data
	Note: it is possible to use **AT#DWRCVR** only if the connection has been opened with **AT#DWCONN**, else the ME is raising an error.
	If the data received are the consequence of a previous data sending issued by **AT#DWSENDR**, then they can be read only using **AT#DWRCVR** command and not **AT#DWRCV** command (i.e.: **AT#DWRCV** and **AT#DWRCVR** are not interchangeable).
AT#DWRCVR=?	Test command reports the supported range of values for all parameters.

3.5.6.12.8. List information on message pending from M2M Service - #DWLRCV

#DWLRCV – List information on message pending from M2M Service	
AT#DWLRCV	Execution command permits the user to obtain information regarding the messages pending from M2M Service in the following format: **#DWLRCV:** **<msg_number>[,<msgId_1>,<msg_1_len>[,<msgId_2>,<msg_2_len>[,… …<msgId_n>,<msg_n_len>]]]** where: **<msg_number>** - number of messages pending from M2M Service Range: >=0

#DWLRCV – List information on message pending from M2M Service	
	<msgId_i> - index of the i-th data message to receive **<msg_i_len>** - length of the i-th data message to receive Note: it is possible to use **AT#DWLRCV** only if the connection has been opened with **AT#DWCONN**, else the ME is raising an error.
AT#DWLRCV=?	Test command reports **OK** result code

3.5.6.12.9. *Enable agent features - #DWEN*

#DWEN – enable agent features	

AT#DWEN=<feat>,<en>[,<option1>[,<option2>[,<option3>[,<option4>[,<option5>]]]]]	Set command permits to enable/disable up to 8 different deviceWISE features.
	
	 Parameters:
	 <feat> - feature to enable or disable; range (0-7)
	   0 – remote at commands
	   1 … 7 – reserved for future use.
	
	 <en> - enable or disable the features
	   0 – disable the feature
	   1 – enable the feature
	
	 <optionX> where X=1,..,5 - optional parameters depending on the feature (string)
	
	 Note: feature 0 (Remote AT commands) has no option.
	 Note: if a connection has been set up issuing either #TCPATRUND or #TCPATRUNL, then DW Remote AT Commands can not execute. Because DW Remote AT Commands can not use with TCPATRUN simultaneously.
	 Note: the **<en>** value is considered only at the very first connection to M2M Service (**AT#DWCONN=1**) after a device power on or reboot

AT#DWEN?	Read command returns the current settings for each feature in the format: **#DWEN:** **<feat>,<en>,<option1>,<option2>,<option3>,<option4>,<option5>**
AT#DWEN=?	Test command reports the supported range of values for parameters **<feat>** and **<en>** and the maximum length of **<optionX>** (where X=1,..,5) parameters

3.5.7. Telit CDMA Custom AT Commands

3.5.7.1. General Configuration AT Commands

3.5.7.1.1. *Common Air Interface parameters - #CAI*

#CAI – Common Air Interface parameters	
AT#CAI?	Read command returns the current common air interface parameters of the module. **#CAI: <sid>,<nid>,<bsid>,<packetid>,<channel>,<pilot_pn>,<mb_prev>,<bs_prev>,<in_use_prev>,<rssi>,<ecio>,<tx_adj>,<rx_state>,<rx_rate>,<tx_rate>,<service_opt>,<slot_index>,<fer>,<voice_priv>,<band>** Parameter: **<sid>** - Integer value of current system ID **<nid>** - Integer value of current network ID **<bsid>** - Integer value of current base station ID **<packetid>** - Integer value of current packet zone ID **<channel>** - Integer value of current channel number **<pilot_pn>** - Integer value of current pilot PN number **<mb_prev>** - Integer value of current mobile station protocol revision 3 - IS95A 4 - IS95B 6 - IS2000 7 - IS2000 Rel A **<bs_prev>** - Integer value of current base station protocol revision Refer to the described above **<mb_prev>** **<in_use_prev>** - Integer value of current in use protocol revision Refer to the described above **<mb_prev>** **<rssi>** - Integer value of current RSSI **<ecio>** - Integer value of current ECIO **<tx_adj>** - Integer value of current TX gain **<rx_state>** - Integer value of current Rx state 0 - CDMA state 1 - Process Sync Channel data 2 - Process Paging Channel data 3 - Process Traffic Channel initialization 4 - Process Traffic Channel data 5 - Monitor the BCCH 6 - Monitor the FCCCH 7 - Monitor both the BCCH and FCCCH 8 - Exit state **<rx_rate>** - Integer value of current Rx rate **<tx_rate>** - Integer value of current Tx rate

#CAI – Common Air Interface parameters	
	<service_opt> - Integer value of current service option <slot_index> - Integer value of current slot cycle index <fer> - Integer value of current frame error rate <voice_priv> - Integer value of current voice privacy mode 0 - disable 1 - enable <band> - Integer value of current band
AT#CAI=?	Test command returns the OK result code.
Example	AT#CAI? #CAI: 4376,30,522,30,350,330,6,6,6,-85,-5,0,2,0,0,0,2,0,0,1 OK AT#CAI=? OK

3.5.7.1.2. Modem Configure parameters - #MODEM

#MODEM – Modem Configure parameters	

AT#MODEM [=<index>]?	Read command returns the modem configuration paramters of the module.
	
	Parameter:
	<index>
	0 .. 13 - To get specific modem configiuration parameter value of the module
	
	#MODEM: <mdn>,<msin>,<vbatt>,<temp>,<systemtime>,<calltime>,<totalcalltime>,<modemstatus>,<fwver>,<model>,<namname>,<lock>,<prlver>,<deepsleep>
	
	Where:
	<mdn> - Mobile directory number
	<msin> - Mobile Subscriber Identifier Number
	<vbatt> - Current Battery Voltage Level
	<temp> - Current Temperature
	<systemtime> - Current System Time (received from the network)
	<calltime> - Latest Call Time
	<totalcalltime> - Total Call Time
	<modemstatus> - Current Modem Status
	  0: IDLE State
	  1: Origination State
	  2: Alerting State
	  3: Conversation State
	  4: Call End State
	  5: Dormant Mode State

#MODEM – Modem Configure parameters	

	<fwver> - Firmware Version, solution Patch release version
	<model> - Model Name
	<namname> - Current Nam Name
	  Note: Not all service providers use NAM name, some providers use a string to display service provider's name. If service provider does not use this, then "UNKNOWN" will be displayed.
	  In case of VERIZON, Nam Name is blank.
	
	<lock> - Current Lock Status
	  0: Not Locked
	  1: Registration Lock
	
	<prlver> - Current PRL Version
	<deepsleep> - Current Deep Sleep Status
	  - 0: Wake Up
	  - 1: Deep Sleep

AT#MODEM=?	Test command returns the **OK** result code
Example	AT#MODEM? #MODEM: 9194547049,9194547049,3.9,0,20080923152338TUE,000000,00000000103,0,SC AUTHZ31340118,CE910-DUAL,UNKNOWN ,0,10030,0 OK AT#MODEM=0? #MODEM: 1234567890 OK AT#MODEM=9? #MODEM: CE910-DUAL OK

3.5.7.1.3. *Mobile NAM parameters - #ENG*

#ENG – Mobile NAM parameters	

AT#ENG=
<index>:<value>[,
<index>:<value>...]
	Set command sets to mobile NAM parameters according to **<index>** parameter.
	
	Parameter:
	<index> - integer type; Index of mobile NAM parameter.
	  0 – Mobile Protocol Revision
	  1 – Mobile Country Code
	  2 – Mobile Network Code
	  3 – Access Overload Control
	  4 – MOB_TERM_HOME registration flag
	  5 – MOB_TERM_FOR_SID registration flag
	  6 – MOB_TERM_FOR_NID registration flag

#ENG – Mobile NAM parameters	
	7 – Station Class Mark 8 – Slot Cycle Index 9 – Mobile Directory Number 10 – Mobile Subscriber Identifier Number 11 – CDMA Preferred Serving System(A/B) 12 – Digital/Analog Mode Preference 13 – CDMA Primary Channel(A) 14 – CDMA Primary Channel(B) 15 – CDMA Secondary Channel(A) 16 – CDMA Secondary Channel(B) 17 – SID-NID pair 18 – The Preferred Forward & Reverse RC value 19 – Slot Mode Note : MDN (Mobile Directory Number) and MSIN(Mobile Subscriber Identifier Number) are blocked set command on Sprint model only.
AT#ENG [=<index>[, <index>...]]?	Read command returns the current mobile NAM parameters in format: **#ENG: <mobprev>,<mcc>,<mnc>,<accolc>,<homereg>,<termforsid>,<termfornid>,<scm>,<sci>,<mdn>,<msin>,<prefserv>,<prefmode>,<primch_a>, <primch_b>,<scch_a>,<scch_b>,(<sid>,<nid>[,<sid>,<nid>…]),(<prefrc>,<prerrc>),<slotmode>** Where: **<mobprev>** – Mobile Protocol Revision (read-only) **<mcc>** – Mobile Country Code **<mnc>** – Mobile Network Code **<accolc>** – Access Overload Control **<homereg>** – MOB_TERM_HOME registration flag **<termforsid>** – MOB_TERM_FOR_SID registration flag **<termfornid>** – MOB_TERM_FOR_NID registration flag **<scm>** – Station Class Mark **<sci>** – Slot Cycle Index **<mdn>** – Mobile Directory Number **<msin>** – Mobile Subscriber Identifier Number **<prefserv>** – CDMA Preferred Serving System(A/B) **<prefmode>** – Digital/Analog Mode Preference **<primch_a>** – CDMA Primary Channel(A) **<primch_b>** – CDMA Primary Channel(B) **<scch_a>** – CDMA Secondary Channel(A) **<scch_b>** – CDMA Secondary Channel(B) **<sid>,<nid>** – SID-NID pair **<prefrc>,<prerrc>** – The Preferred Forward & Reverse RC value **<slotmode>** – Slot Mode **Note: In RUIM version, most parameters are read-only.**

#ENG – Mobile NAM parameters	
AT#ENG=?	Test command returns the **OK** result code
Example	AT#ENG? #ENG: 6,310,00,9,1,1,1,42,2,1234567890,9135069409,5,4,283,384,691,777,(4139,65535),(0,0),0 OK AT#ENG=9? #ENG: 1234567890 OK AT#ENG=1:400,2:06 OK AT#ENG=1,2? #ENG: 400,06 OK

3.5.7.1.4. *CDMA Notification - #NOTI*

#NOTI – CDMA Notification	

AT#NOTI=
<index>,<onoff>
	Set command sets to enable or disable related CDMA notification.
	
	Parameter:
	<index> - CDMA notification selection
	 0 – All notification messages (1~18)
	 1 – "#CNIP"
	   the output when the module receives a Calling Number Identification Presentation from the network.
	 2 – "#CNAP"
	   the output when the module receives a Calling Naming Presentation from the network.
	 3 – "#DISREC"
	   the output when the module receives a Display Record from the network.
	 4 – "#LOCK"
	   the output when the module receives a LOCK from the network during registering state.
	 5 – "#UNLOCK"
	   the output when the module receive a UNLOCK from the network during locked state.
	 6 – "#SMSFULL"
	   the output when SMS are FULL.

#NOTI – CDMA Notification	
	7 – "#ENTERDEEP"
	the output when the module enters Power save mode.
	8 – "#EXITDEEP"
	the output when the module exits Power save mode.
	9 – "#ENTERDRM"
	the output when the module enters Dormant state.
	10 – "#EXITDRM"
	the output when the module exits Dormant state into Activate state.
	11 – "#DREL"
	the output when the module releases Data call.
	12 – "#ROAM"
	the RI (roaming indicator) output matching with PRL when system is changed.
	13 – "#ERR_CODE"
	the output when MIP ERROR is occurred.
	14 – "#ROAMGUARD"
	the output when the module moves between Domestic area and International area regarding data roaming.
	15 – "#N11"
	the output when N11 digits dialed by user
	16 – "#SERVICE"
	the output when the service state of module changed.
	Service State Messages
	"#SERVICE: 0" – No Service State
	"#SERVICE: 2" – Normal Service State
	"#SERVICE: 4" – Power save or Deep sleep state
	17 – "#EMERGENCY CALL"
	the output when the module tries to make an emergency call.
	18 – "#SERVICE_HDR" (Reserved)
	the output when the HDR service state of module changed.
	Service State Messages
	"#SERVICE_HDR: 0" – No Service State
	"#SERVICE_HDR: 2" – Normal Service State
	"#SERVICE_HDR: 4" – Power save or Deep sleep state.
	<onoff> - Device configuration message status
	0 – disable (default)
	1 – enable
	Note: "#EMERGENCY CALL" message is displayed on Verizon/Sprint version only.
	Note: "#EMERGENCY CALL" message is displayed always on Sprint version, even though "onoff" value is disable. (SGS requests.)
AT#NOTI?	Read command returns the current status flag of **<onoff>**.
	#NOTI: <onoff (for index 1)>,<onoff (for index 2)>, … ,<onoff (for index 18)>

#NOTI – CDMA Notification	
AT#NOTI=?	Test command reports the range of the parameter **<index>,<onoff>**
Example	AT#NOTI=? #NOTI: (0-18),(0,1) OK AT#NOTI? #NOTI: 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 OK AT#NOTI=0,1 OK AT#NOTI? #NOTI: 1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1 OK AT#NOTI=7,0 OK AT#NOTI? #NOTI: 1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1 OK

3.5.7.1.5. *Mobile Diretory Number – $MDN*

$MDN– Change Operational Mode of Modem	
AT$MDN=<mdn> or AT$MDN= <msl or otksl>, <mdn>	This command manipulates the Mobile Directory Number of the module. Parameter: **<msl>** - Master Subsidy Lock value. (See Note) **<otksl> -** One-Time Keypad Subsidy Lock **<mdn>** - The mobile directory number expressed as a 10 digit decimal phone-number. Note: Command format for each operator are as follow: Verizon : AT$MDN=<mdn> Aeris: AT$MDN=<msl>, <mdn> Sprint : AT$MDN=<msl or otksl>,<mdn>
AT$MDN?	Read command returns the mobile directory number with command echo. **$MDN: <mdn>**
AT$MDN=?	Test command returns the **OK** result code
Example	AT$MDN=? OK AT$MDN? $MDN: 1234567890

3.5.7.1.6. *Mobile Station ID - $MSID*

$MSID– Change Operational Mode of Modem	
AT$MSID=<msid> or AT$MSID= <msl or otksl>, <msid>	This command manipulates the Mobile Station ID of the module. Parameter: **<msl>** - Master Subsidy Lock value. (See Note) **<otksl> -** One-Time Keypad Subsidy Lock **<msid>** - The Mobile Station ID expressed as a 10 digit decimal phone-number Note: Command format for each operator are as follow: Verizon : AT$MSID =< msid > Aeris: AT$MSID =<msl>, < msid > Sprint : AT$MSID=<msl or otksl>,<msid> Note: This command is Read-only at RUIM specific version
AT$MSID?	Read command returns the Mobile Station ID with command echo. **$MSID: <msid>**
AT$MSID=?	Test command returns the **OK** result code
Example	AT$MSID=? OK AT$MSID? $MSID: 0000000000

3.5.7.1.7. *Notification of Service - +SERVICE*

+SERVICE – Notification of Service	
AT+SERVICE?	Read command returns the Mobile Station ID with command echo. **+SERVICE: <serv>** Parameter: <serv> 0 – No Service 1 – 1xRTT Service 2 – EVDO Release 0 (Not Support) 3 – EVDO Release A (Not Support) 4 – GPRS(Not Support)
AT+SERVICE=?	Test command returns the **OK** result code

3.5.7.1.8. *Reverse Logistic Support - #RTN*

#RTN – Reverse Logistic Support	
AT#RTN=<n>	The execute command will reset the selected parameter back to its factory value. Parameter: **<n>** - Parameter for reset. 0 – MDN

#RTN – Reverse Logistic Support	
	1 – MSID 2 – Last Call Time 3 – Total Call Time 4 – MIP Porfile – This deletes only MIP profile 1 Note: Case of Verizon & Sprint , after reset module MDN and MSID are changed as follow: The MDN and MSID default values should be set to the last four digits of the ESN or pESN for MEID. Case of Aeris, after reset module MDN and MSID are changed as follow: - MDN: The default values should be set to the last four digits of the ESN or pESN for MEID - MSID: The digits "407380" followed by the last four digits of the decimal ESN or pESN for MEID.
AT#RTN=?	Test command returns the **OK** result code

3.5.7.1.9. *Base Station Lat/long Data - $CELLPOS*

$CELLPOS –Get a latitude and longitude of Base Station	
AT$CELLPOS	Gets a Latitude and Longitude Data of Base Station in CDMA network $CELLPOS: **<latitude>,<longitude >** Parameter: NONE
AT$CELLPOS?	Read command returns the currently used values, in the format: $CELLPOS: **<latitude>,<longitude >**
AT$CELLPOS=?	Test command returns the OK result code
Example	AT$CELLPOS $CELLPOS: 37.5197,126.9311 OK AT$CELLPOS? $CELLPOS: 37.5197,126.9311 OK AT$CELLPOS=? OK AT$CELLPOS= ERROR

3.5.7.2. Air interface and call processing

3.5.7.2.1. *Preferred Radio Configuration - #PREFRC*

#PREFRC – Preferred Radio Configuration	
AT#PREFRC= **<for_rc>,<rev_rc>**	Set command sets the preferred radio configuration. Parameter: **<for_rc>** - integer forward radio configuration **<rev_rc>** - integer reverse radio configuration Note: This command is used to set the preferred RC for the forward and reverse channel. If you want to get the cached pref RC from NV, set parameter value to (1,2,3,4,5), otherwise both "for_rc and "rev_rc" must be set to '0'.
AT#PREFRC?	Read command returns the radio configurations in format: **#PREFRC: <for_rc>,<rev_rc>**
AT#PREFRC=?	Test command reports the range of **<for_rc>,<rev_rc>** parameters: **AT#PREFRC: (0-5),(0-5)**

3.5.7.2.2. *Voice Privacy Setting - #VOICEPRIV*

#VOICEPRIV – Voice Privacy Setting	

AT#VOICEPRIV=
<v_privacy>
	Set command sets voice privacy mode according to **< v_privacy >** parameter.
	
	Parameter:
	< v_privacy > - Value of the voice privacy setting value
	  0 – OFF (USC factory default)
	   1 – ON (factory default except USC)

AT#VOICEPRIV?	Read command returns the current voice privacy setting value: **#VOICEPRIV: <v_ privacy>**
AT#VOICEPRIV=?	Test command reports the range of **< v_privacy >** parameters: **#VOICEPRIV: (0,1)**
Example	AT#VOICEPRIV=? #VOICEPRIV: (0,1) OK AT#VOICEPRIV? #VOICEPRIV: 0 OK AT#VOICEPRIV=1

#VOICEPRIV – Voice Privacy Setting	
	OK AT#VOICEPRIV? #VOICEPRIV: 1 OK

3.5.7.2.3. *Vocoder Setting Value Reading or Writing - #PREFVOC*

#PREFVOC – Vocoder Setting Value Reading or Writing	

AT#PREFVOC=
[<evrc>,<so1>,
<so2>,<so3>]
	Set command sets vocoder setting value.
	
	Parameter:
	<evrc> - The mode of EVRC
	  0 – disable EVRC (factory default value)
	  1 – enable EVRC (factory default value for Sprint and US cellular only)
	
	<so1> - page voice service option in home network
	  3 – for EVRC (factory default value)
	  32768 – for QCELP
	
	<so2> - originate voice service option in home network
	  3 – for EVRC (factory default value)
	  32768 – for QCELP
	
	<so3> - originate voice service option in roam network
	  3 – for EVRC (factory default value)
	  32768 – for QCELP
	
	  Note: If <evrc> is set to 0, voice service option will be discard.
	  Note: For models supporting the 4GV, the supporting service option will be changed as follows.
	      <so1>,<so2>,<so3>
	        3 - EVRC
	        32768 - QCELP
	        68 - 4GV NB
	        70 - 4GV WB

AT#PREFVOC?	Read command returns the vocoder setting values in format: **#PREFVOC: <evrc>,<so1>,<so2>,<so3>**
AT#PREFVOC=?	Test command reports the range of the parameters
Example	AT#PREFVOC? #PREFVOC: 0,3,3,3 OK AT#PREFVOC=1,3,3,3 OK

#PREFVOC – Vocoder Setting Value Reading or Writing	
	AT#PREFVOC? #PREFVOC: 1,3,3,3 OK AT#PREFVOC=0,32768,32768,32768 OK AT#PREFVOC? #PREFVOC: 0,32768,32768,32768 OK

3.5.7.2.4. *OTASP Setting - #OTASPEN*

#OTASPEN – OTASP Setting	

**AT#OTASPEN=
< mode >**
	Set command enables or disables the OTASP function.
	
	Parameter:
	<mode>
	  0 - disables OTASP
	  1 - enables OTASP
	
	Note: This is operator specific, and is not supported by the Sprint Network.
	Note: The Unsolicited indication message is displayed as following.
	  #OTASP: <n>
	
	where:
	<n> :
	  0 : Origination for OTASP
	  1 : Start OTASP/OTAPA commit
	  2 : End OTASP/OTAPA commit(success)
	  5 : Failed

AT#OTASPEN?	Read command returns the OTASP setting in format: **#OTASPEN: < mode >**
AT#OTASPEN=?	Test command returns the values for the **< mode >** parameter.

3.5.7.2.5. *Configuration String - +CFG*

+CFG – Configuration String	
AT+CFG = <string>	Set command sets a module configuration string. The string will be stored by the module and sent to the base station prior to dialing. Each transmission of an AT+CFG command from Host replaces the contents of the previous string. Parameter: **<string> -** Configuration string may be up to 248 character.

AT+CFG?	Read command returns the configuration string in format: **+CFG: <string>**
AT+CFG =?	Test command returns the **OK** result code.
Example	AT+CFG=? OK AT+CFG? +CFG: "" OK AT+CFG="data" OK AT+CFG? +CFG: "data" OK

3.5.7.2.6. *RM Interface Setting – +CRM*

+CRM – RM Interface Setting	
AT+CRM=<value>	Set command changes the RM interface protocol. Note: When the AT$QCMIP value is changed to "1" or "2", this modifies the value of AT+CRM to 2. When AT+CRM has a value of "2", it enables network mode operation. Changing the value of AT$QCMIP to "0" will reset the AT+CRM to its original value. Parameter: **<value>** - RM Interface protocol: 0 – Circuit Data 1 – Packet Data (Relay layer packet data) 2 – Packet Data (Network layer packet data)
AT+CRM?	Read command returns the RM interface setting in format: **+CRM: <value>**
AT+CRM=?	Test command reports the range of the **<value>** parameter.
Example	AT+CRM=? +CRM: (0-2) OK AT+CRM? +CRM: 2 OK AT+CRM=0 ERROR AT$QCMIP?

	$QCMIP: 2 OK AT$QCMIP=0 OK AT+CRM=0 OK AT+CRM? +CRM: 0 OK AT$QCMIP=2 OK AT+CRM? +CRM: 2 OK

3.5.7.2.7. *Clear MRU Table – #CLRMRU*

#CLRMRU – Clear MRU Table	
AT#CLRMRU	This command is used to clear the Most Recently Used(MRU)table.
Example	AT#CLRMRU OK

3.5.7.3. DATA Session AT commands

3.5.7.3.1. *Data Inactivity Timer – +CTA*

+CTA – Data Inactivity Timer	
AT+CTA=<n>	Set command sets Um packet data inactivity timer Parameter: <n> - Um packet data inactivity timer: 0 - Traffic Channel not released during inactivity periods. 1-255 - Release the Traffic Channel after <value> 1-second intervals have elapsed since last sending or receiving RLP data frames on the Um interface. (Aeris Default: 60 seconds)

+CTA – Data Inactivity Timer	
	(Verizon/Sprint/US Cellular/SC Default: 30 Seconds)
AT+CTA?	Read command returns the data inactivity timer in format: **+CTA: <n>**
AT+CTA=?	Test command reports the range of the **<n>** parameter.
Example	AT+CTA=? +CTA: (0-255) OK AT+CTA? +CTA: 60 OK AT+CTA=30 OK AT+CTA? +CTA: 30 OK

3.5.7.3.2. *Packet Zone ID – +PZID*

+PZID – Packet Zone ID	
AT+PZID?	Displays the current **<packet_zone_id>** in the Extended System Parameters Message or the In-Traffic System Parameters Message.
AT+PZID=?	Returns the **OK** result code.
Example	AT+PZID=? OK AT+PZID? +PZID: 30 OK

3.5.7.3.3. *Interrupt Packet Data – $GODORMANT*

$GODORMANT – Interrupt Packet Data	
AT$GODORMANT	Returns the OK result code. Executed immediately, not time critical. Although running this AT-Command, The device would emerge from DORMANT state then become ACTIVE state as long as the device has any packets to send or receive Note : The device should be in Packet Data Active Session to get result "OK", In case of QNC call, Result must be "ERROR" since QNC doesn't support

$GODORMANT – Interrupt Packet Data	
	DORMANT.
AT$GODORMANT=?	Returns the **OK** result code.
Example	AT$GODORMANT OK

3.5.7.3.4. *Test Origination – #TESTORI*

#TESTORI – Test Origination	
AT#TESTORI= <svc_opt>[,<num>]	Set command originates a (loopback) test call according to **<idx >** parameter. Parameter: **< svc_opt >** Service option for test call: 0 – Rate Set 1 Loopback Service Option(Service Option: 0x02) 1 – Rate Set 2 Loopback Service Option(Service Option: 0x09) 2 – Loopback service Option 55(Service Option: 0x37) 3 – Markov Service Option(Service Option: 0x8002) 4 – Markov Service Option (13K) (Service Option: 0x801C) 5 – Rate Set 2 Markov Service Option(Service Option: 0x801F) 6 – Rate Set 1 Markov Service Option(Service Option: 0x801E) 7 – Markov Service Option 54(Service Option: 0x36) 8 – Service option for Simple TDSO(Service Option: 0x8008) 9 – Service option for FULL TDSO(Service Option: 0x20) **< num >** Destination number for test calls
Example	AT#TESTORI=0 OK AT#TESTORI=0 ,12345678 OK

3.5.8. RUIM Specific AT Commands

3.5.8.1. General Commands

3.5.8.1.1. *Query RUIM Status - #QSS*

#QSS - Query RUIM Status	

AT#QSS= [<mode>]	Set command enables/disables the Query RUIM Status unsolicited indication in the ME.
	
	 Parameter:
	 <mode> - type of notification
	   0 - disabled (factory default); it's possible only to query the current RUIM status through Read command **AT#QSS?**
	   1 - enabled; the ME informs at every RUIM status change through the following unsolicited indication:
	
	     #QSS: <status>
	
	     where:
	     <status> - current RUIM status
	       0 - RUIM NOT INSERTED
	       1 - RUIM INSERTED
	
	   2 - enabled; the ME informs at every RUIM status change through the following unsolicited indication:
	
	     #QSS: <status>
	
	     where:
	     <status> - current RUIM status
	       0 - RUIM NOT INSERTED
	       1 - RUIM INSERTED
	       2 - RUIM INSERTED and PIN UNLOCKED
	       3 - RUIM INSERTED and READY (SMS and Phonebook access are possible).
	
	 Note: the command reports the SIM status change after the <mode> has been set to 2. We strongly suggest to set <mode>=2 and save the value in the user profile, then power off the module. The proper SIM status will be available at the next power on.

AT#QSS?	Read command reports whether the unsolicited indication **#QSS** is currently enabled or not, along with the RUIM status, in the format: **#QSS: <mode>,<status>** (**<mode>** and **<status>** are described above)

#QSS - Query RUIM Status	
AT#QSS=?	Test command returns the supported range of values for parameter **<mode>**.
Example	AT#QSS? #QSS:0,1 OK

3.5.8.1.2. *Enter PIN – +CPIN*

+CPIN - Enter PIN	
AT+CPIN=<pin> [,<newpin>]	Set command sends to the device a password which is necessary before it can be operated RUIM PIN, RUIM PUK. If the PIN required is RUIM PUK or RUIM PUK2, the **<newpin>** is required. This second pin, **<newpin>** will replace the old pin in the SIM. To change the PIN at no PIN request status, 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?**

AT+CPIN?	Read command reports the PIN/PUK/PIN2/PUK2 request status of the device in the form:
	+CPIN: <code>
	where:
	<code> - PIN/PUK/PIN2/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
	  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 given; this **<code>** is returned only when the last executed command resulted in PUK2 authentication failure (i.e. **+CME ERROR: 18**)
	
	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>

AT+CPIN=?	Test command returns **OK** result code.
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*

+CPIN - Enter PIN	
	OK
Reference	3GPP TS 27.007

3.5.8.1.3. *Facility Lock/Unlock - +CLCK*

+CLCK - Facility Lock/Unlock	

AT+CLCK=
<fac>,<mode>
[,<passwd>]
	Execution command is used to lock or unlock a **ME** or a network facility.
	
	Parameters:
	<fac> - facility
	  "SC" - RUIM (PIN request) (device asks RUIM password at power-up and when this lock command issued)
	"FD" - RUIM fixed dialing memory feature (if PIN2 authentication has not been done during the current session, PIN2 is required as <passwd>)
	<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**
	
	Note: when **<mode>=2** and command successful, it returns:
	+CLCK: <status>
	
	where
	<status> - the current status of the facility
	  0 - not active
	  1 - active

AT+CLCK=?	Test command reports all the facilities supported by the device.
Reference	3GPP TS 27.007
Example	*Query RUIM Lock facility* AT+CLCK ="SC",2 +CLCK: <status> OK
Note	It will return ERROR if executed using SMSATRUN digest mode or TCPATRUN server mode

3.5.8.1.4. *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**.

+CPWD - Change Facility Password	

	Parameters:
	\<fac\> - facility
	  "SC" - RUIM (PIN request)
	  "P2" - RUIM PIN2
	
	\<oldpwd\> - string type, it shall be the same as password specified for the facility from the ME user interface or with command **+CPWD**.
	\<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: ("SC",8), ("P2",8) OK
Reference	3GPP TS 27.007

3.5.8.1.5. *Read ICCID (Integrated Circuit Card Identification) - +CCID*

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

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

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

3.5.8.1.7. *Service Provider Name - #SPN*

#SPN - Service Provider Name	

#SPN - Service Provider Name	
AT#SPN	Execution command returns the service provider string contained in the RUIM field **SPN**, in the format: **#SPN: <spn>** where: **<spn>** - service provider string contained in the RUIM field **SPN**, represented in the currently selected character set (see **+CSCS**). Note: if the RUIM field SPN is empty, the command returns just the **OK** result code
AT#SPN=?	Test command returns the **OK** result code.

3.5.8.1.8. *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

3.5.8.1.9. *Enable/Disable CHV - #CHVEN*

#CHVEN – Enable/Disable CHV	
AT#CHVEN=<mode>, <password>	Execution command is used to enable or disable CHV(PIN) on **RUIM**. **<mode>** - defines the operation to be done on the RUIM

#CHVEN – Enable/Disable CHV	
	0 - Disable PIN 1 - Enable PIN **<passwd>** - PIN code of RUIM Note : This command is the same operation with +CLCK. It's only keeping for backward compalitibilty.
AT#CHVEN?	Read command query status of PIN in the format: **#CHVEN: <n>** where : **<n> -** status of PIN 0 – PIN disabled 1 – PIN enabled
AT#CHVEN=?	Test command returns the OK result code.
Example	AT#CHVEN=1, 1111 *Enable PIN* OK

3.5.9. SIM Toolkit AT Commands(For Only RUIM version)

3.5.9.1. SIM Tookit Interface Activation - #STIA

#STIA - SIM Tookit Interface Activation	

AT#STIA= [<mode> [,<timeout>]]	Set command is used to activate the SAT sending of unsolicited indications when a **proactive command** is received from SIM.
	
	Parameters:
	<mode>
	  0 - disable SAT (no **<timeout>** required, if given will be ignored)
	  1 - enable SAT without unsolicited indication **#STN** (default)
	  2 - enable SAT and extended unsolicited indication **#STN** (see #STGI)
	  3 - enable SAT and reduced unsolicited indication **#STN** (see #STGI)
	  17 - enable SAT without unsolicited indication **#STN** and 3GPP TS 23.038 alphabet used
	  18 - enable SAT and extended unsolicited indication **#STN** (see #STGI) . only GSM default alphaber is supported
	  19 - enable SAT and reduced unsolicited indication **#STN** (see #STGI). only GSM default alphabet is supported
	  33 - enable SAT without unsolicited indication **#STN** and UCS2 alphabet used
	  34 - enable SAT with extended unsolicited indication **#STN** (see #STGI). only UCS2 character set is supported
	  35 - enable SAT with reduced unsolicited indication **#STN** (see #STGI). only UCS2 character set is supported
	<timeout> - time-out for user responses
	  1-2 - time-out in minutes (default 2). Any ongoing (but unanswered) **proactive command** will be aborted automatically after **<timeout>** minutes. In this case, the terminal response is either "ME currently unable to process command", or if applicable, "No response from user". In addition an unsolicited indication will be sent to the external application:
	
	      #STN: <cmdTerminateValue>
	
	      where:
	        <cmdTerminateValue> is defined as **<cmdType>** + **terminate offset**; the terminate offset equals 100.
	
	Note: every time the SIM application issues a **proactive command** that requires user interaction an unsolicited code will be sent, if enabled with **#STIA** command, as follows:
	
	  • if **<mode>** parameter of **#STIA** command has been set to 3 (reduced unsolicited indication) an unsolicited indication will be sent, indicating the type of **proactive command** issued by the SIM:

#STIA - SIM Tookit Interface Activation	
	#STN: <cmdType>
	• if **<mode>** parameter of **#STIA** command has been set to 2 (extended unsolicited indication) the format of the unsolicited indication depends on the specific command:
	if <cmdType>=1 (REFRESH)
	an unsolicited notification will be sent to the user:
	#STN: <cmdType>,<refresh type>
	where: **<refresh type>** 0 - SIM Initialization and Full File Change Notification; 1 - File Change Notification; 2 - SIM Initialization and File Change Notification; 3 - SIM Initialization; 4 - SIM Reset
	In this case neither #STGI nor #STSR commands are required: • **AT#STGI** is accepted anyway. • **AT#STSR=<cmdType>,0** will answer **OK** but do nothing.
	if <cmdType>=17 (SEND SS) *if <cmdType>=19 (SEND SHORT MESSAGE)* *if <cmdType>=20 (SEND DTMF)* *if <cmdType>=32 (PLAY TONE)*
	an unsolicited notification will be sent if allowed by SIM (see 3GPP TS 31.111):
	#STN: <cmdType>[,<text>]
	where: **<text>** - (optional) text to be displayed to user
	In these cases neither #STGI nor #STSR commands are required: • **AT#STGI** is accepted anyway. • **AT#STSR=<cmdType>,0** will answer **OK** but do nothing.
	In case of SEND SHORT MESSAGE (**<cmdType>**=19) command if sending to network fails an unsolicited notification will be sent

#STIA - SIM Tookit Interface Activation	
	#STN: 119
	if <cmdType>=33 (DISPLAY TEXT)
	an unsolicited notification will be sent if allowed by SIM (see 3GPP TS 31.111):
	#STN: <cmdType>,<cmdDetails>[,<text>]
	where:
	<cmdDetails> - unsigned Integer used as a bit field.
	0..255 - used as a bit field:
	bit 1:
	0 - normal priority
	1 - high priority
	bits 2 to 7: reserved for future use
	bit 8:
	0 - clear message after a delay
	1 - wait for user to clear message
	<text> - (optional) text to be displayed to user

	In this case:
	1. if **<cmdDetails>/bit8** is **0** neither #STGI nor #STSR commands are required:
	  • **AT#STGI** is accepted anyway.
	  • **AT#STSR=<cmdType>,0** will answer **OK** but do nothing.
	2. If **<cmdDetails>/bit8** is 1 #STSR command is required

	if <cmdType>=18 (SEND USSD)
	an unsolicited notification will be sent to the user:
	#STN: <cmdType>[,<text>]
	where: **<text>** - optional text string sent by SIM

	In this case:
	  • **AT#STSR=18,20** can be sent to end USSD transaction.
	  • **AT#STGI** is accepted anyway.
	  • **AT#STSR=<cmdType>,0** will answer **OK** but do nothing.

	if <cmdType>=5 (SET UP EVENT LIST)

#STIA - SIM Tookit Interface Activation	

	an unsolicited notification will be sent:
	#STN: <cmdType>[,<event list mask>]
	where:
	<event list mask> - (optional)hexadecimal number representing the list of events to monitor (see 3GPP TS 31.111)
	- '00' = MT call
	- '01' = Call connected
	- '02' = Call disconnected
	- '03' = Location status
	- '04' = User activity
	- '05' = Idle screen available
	- '06' = Card reader status (if class "a" is supported)
	- '07' = Language selection
	- '08' = Browser Termination (if class "c" is supported)
	- '09' = Data available (if class "e" is supported)
	- '0A' = Channel status (if class "e" is supported)
	
	The hexadecimal number is actually a bit mask, where each bit, when set, indicates that the corresponding event has to be monitored (e.g., if <event list mask> is 0x0001, it means that MT call has to be monitored).
	
	In these cases neither #STGI nor #STSR commands are required:
	• **AT#STGI** is accepted anyway.
	• **AT#STSR=<cmdType>,0** will answer **OK** but do nothing.
	
	All other commands:
	
	the unsolicited indication will report just the proactive command type:
	
	#STN: <cmdType>
	
	Note: if the **call control** or **SMS control facility in the SIM** is activated, when the customer application makes an outgoing call, or sends an SS or USSD, or an SMS, the following **#STN** unsolicited indication could be sent, according to 3GPP TS 31.111, to indicate whether the outgoing call has been accepted, rejected or modified by the SIM, or if the SMS service centre address or destination has been changed:
	
	#STN: <cmdTerminateValue>,<Result>[,<TextInfo>[,<Number>[,<MODestAddr>]]]
	
	where
	<cmdTerminateValue>
	  150 - SMS control response
	  160 - call/SS/USSD response
	<Result>
	  0 - Call/SMS not allowed

#STIA - SIM Tookit Interface Activation	
	1 - Call/SMS allowed 2 - Call/SMS allowed with modification **<Number>** - Called number, Service Center Address or SS String in ASCII format. **<MODestAddr>** - MO destination address in ASCII format. **<TextInfo>** - alpha identifier provided by the SIM in ASCII format. Note: when the SIM Application enters its main menu again (i.e. not at startup) an unsolicited result code **#STN: 254** is sent. The TA does not need to respond directly, i.e. **AT#STSR** is not required. It is possible to restart the SAT session from the main menu again with the command **AT#STGI=37**. Note: The settings are saved on user profile and available on following reboot. SIM Toolkit activation/deactivation is only performed at power on.

AT#STIA?	Read command can be used to get information about the SAT interface in the format:
	
	#STIA: <state>,<mode>,<timeout>,<SatProfile>
	
	where:
	<state> - the device is in one of the following state:
	  0 - SIM has not started its application yet
	  1 - SIM has started its application (SAT main menu ready)
	<mode> - SAT and unsolicited indications enabling status (see above)
	<timeout> - time-out for user responses (see above)
	<SatProfile> - SAT Terminal Profile according to 3GPP TS 31.111, i. e. the list of SIM Application Toolkit facilities that are supported by the ME. The profile cannot be changed by the TA.
	
	Note: In SAT applications usually an SMS message is sent to the network provider containing service requests, e.g. to send the latest news. The provider returns a message with the requested information.
	Before activating SAT it is recommended to set the SMS text mode with command **AT+CMGF=1** and to enable unsolicited indications for incoming SMS messages with command +CNMI.

AT#STIA=?	Test command returns the range of available values for the parameters **<mode>** and **<timeout>**.
Note	Just one instance at a time, the one which first issued **AT#STIA=n** (with **n** different from zero), is allowed to issue SAT commands, and this is valid till the same

#STIA - SIM Tookit Interface Activation	
	instance issues **AT#STIA=0**. After power cycle another instance can enable SAT.
Note	A typical SAT session on AT interface starts after an **#STN: 37** unsolicited code is received, if enabled(see above). At that point usually an **AT#STGI=37** command is issued (see **#STGI)** and after the SAT main menu has been displayed on TE an **AT#STSR=37,0,x** command is issued to select an item in the menu (see **#STSR)**

3.5.9.2. SIM Tookit Get Information - #STGI

#STGI - SIM Tookit Get Information	

AT#STGI= [<cmdType>]	**#STGI** set command is used to request the parameters of a **proactive command** from the ME.
	
	 Parameter:
	 <cmdType> - **proactive command** ID according to 3GPP TS 102.223 (decimal); these are only those command types that use the AT interface; SAT commands which are not using the AT interface (not MMI related SAT commands, e.g. PROVIDE LOCAL INFORMATION) are executed without sending any indication to the user
	  1 - REFRESH
	  5 - SET UP ENENT LIST
	  16 - SET UP CALL
	  17 - SEND SS
	  18 - SEND USSD
	  19 - SEND SHORT MESSAGE
	  20 - SEND DTMF
	  32 - PLAY TONE
	  33 - DISPLAY TEXT
	  34 - GET INKEY
	  35 - GET INPUT
	  36 - SELECT ITEM
	  37 - SET UP MENU
	  40 - SET UP IDLE MODE TEXT
	
	 Requested command parameters are sent using an **#STGI** indication:
	
	 #STGI: <parameters>
	
	 where **<parameters>** depends upon the ongoing **proactive command** as follows:
	
	 if <cmdType>=1 (REFRESH)
	
	 #STGI: <cmdType>,<refresh type>
	 where:
	 <refresh type>
	  0 - SIM Initialization and Full File Change Notification;

#STGI - SIM Tookit Get Information	
	1 - File Change Notification; 2 - SIM Initialization and File Change Notification; 3 - SIM Initialization; 4 - SIM Reset *if <cmdType>=5 (SET UP EVENT LIST)* **#STGI: <cmdType>,<event list mask>** where: **<event list mask>** - hexadecimal number representing the list of events to monitor (see 3GPP TS 31.111): - '00' = MT call - '01' = Call connected - '02' = Call disconnected - '03' = Location status - '04' = User activity - '05' = Idle screen available - '06' = Card reader status (if class "a" is supported) - '07' = Language selection - '08' = Browser Termination (if class "c" is supported) - '09' = Data available (if class "e" is supported) - '0A' = Channel status (if class "e" is supported) The hexadecimal number is actually a bit mask, where each bit, when set, indicates that the corresponding event has to be monitored (e.g., *if <cmdType>=16 (SET UP CALL)* **#STGI: <cmdType>,<cmdDetails>,[<confirmationText>],** **<calledNumber>** where: **<cmdDetails>** - unsigned integer, used as an enumeration 0 - Set up call, but only if not currently busy on another call 1 - Set up call, but only if not currently busy on another call, with redial 2 - Set up call, putting all other calls (if any) on hold 3 - Set up call, putting all other calls (if any) on hold, with redial 4 - Set up call, disconnecting all other calls (if any) 5 - Set up call, disconnecting all other calls (if any), with redial **<confirmationText>** - string for user confirmation stage **<calledNumber>** - string containing called number *if <cmdType>=17 (SEND SS)* *if <cmdType>=18 (SEND USSD)* *if <cmdType>=19 (SEND SHORT MESSAGE)* *if <cmdType>=20 (SEND DTMF)*

#STGI - SIM Tookit Get Information	

	if <cmdType>=32 (PLAY TONE)
	if <cmdType>=40 (SET UP IDLE MODE TEXT)
	
	#STGI: <cmdType>[,<text>]
	
	where:
	<text> - text to be displayed to user
	
	if <cmdType>=33 (DISPLAY TEXT)
	
	#STGI: <cmdType>,<cmdDetails>[,<text>]
	
	where:
	<cmdDetails> - unsigned Integer used as a bit field.
	  0..255 - used as a bit field:
	    bit 1:
	      0 - normal priority
	      1 - high priority
	    bits 2 to 7: reserved for future use
	    bit 8:
	      0 - clear message after a delay
	      1 - wait for user to clear message
	<text> - text to be displayed to user
	
	if <cmdType>=34 (GET INKEY)
	
	#STGI: <cmdType>,<cmdDetails>,<text>
	
	where:
	<cmdDetails> - unsigned Integer used as a bit field.
	  0..255 - used as a bit field:
	    bit 1:
	      0 - Digits only (0-9, *, # and +)
	      1 - Alphabet set;
	    bit 2:
	      0 - SMS default alphabet (GSM character set)
	      1 - UCS2 alphabet
	    bit 3:
	      0 - Character sets defined by bit 1 and bit 2 are enabled
	      1 - Character sets defined by bit 1 and bit 2 are disabled and the "Yes/No"
	        response is requested
	    bits 4 to 7:
	      0
	    bit 8:
	      0 - No help information available
	      1 - Help information available

#STGI - SIM Tookit Get Information	

	<text> - String as prompt for text.
	
	if **<cmdType>=35** *(GET INPUT)*
	
	#STGI: <cmdType>,<commandDetails>,<text>,<responseMin>, <responseMax>[,<defaultText>]
	
	where:
	<commandDetails> - unsigned Integer used as a bit field.
	  0..255 - used as a bit field:
	    bit 1:
	     0 - Digits only (0-9, *, #, and +)
	     1 - Alphabet set
	    bit 2:
	     0 - SMS default alphabet (GSM character set)
	     1 - UCS2 alphabet
	    bit 3:
	     0 - ME may echo user input on the display
	     1 - User input shall not be revealed in any way. Hidden entry mode (see 3GPP TS 31.111) is only available when using digit input. In hidden entry mode only characters ('0'-'9', '*' and '#') are allowed.
	    bit 4:
	     0 - User input to be in unpacked format
	     1 - User input to be in SMS packed format
	    bits 5 to 7:
	     0
	    bit 8:
	     0 - No help information available
	     1 - Help information available
	<text> - string as prompt for text
	<responseMin> - minimum length of user input
	 0..255
	<responseMax> - maximum length of user input
	 0..255
	<defaultText> - string supplied as default response text
	
	if **<cmdType>=36** *(SELECT ITEM)*
	
	The first line of output is:
	
	#STGI: <cmdType>,<commandDetails>,<numOfItems>[,<titleText>] <CR><LF>
	
	One line follows for every item, repeated for **<numOfItems>**:
	
	#STGI: <cmdType>,<itemId>,<itemText>[,<nextActionId>]

#STGI - SIM Tookit Get Information	
	where: **<commandDetails>** - unsigned Integer used as a bitfield 0..255 - used as a bit field: **bit 1**: 0 - Presentation type is not specified 1 - Presentation type is specified in bit 2 **bit 2**: 0 - Presentation as a choice of data values if bit 1 = '1' 1 - Presentation as a choice of navigation options if bit 1 is '1' **bit 3**: 0 - No selection preference 1 - Selection using soft key preferred **bits 4 to 7**: 0 **bit 8**: 0 - No help information available 1 - Help information available **<numOfItems>** - number of items in the list **<titleText>** - string giving menu title **<itemId>** - item identifier 1..**<numOfItems>** **<itemText>** - title of item **<nextActionId>** - the next proactive command type to be issued upon execution of the menu item. 0 - no next action information available. *if **<cmdType>=37** (SET UP MENU)* The first line of output is: **#STGI: <cmdType>,<commandDetails>,<numOfItems>,<titleText>** **<CR><LF>** One line follows for every item, repeated for **<numOfItems>**: **#STGI: <cmdType>,<itemId>,<itemText>[,<nextActionId>]** where: **<commandDetails>** - unsigned Integer used as a bitfield 0..255 - used as a bit field: **bit 1**: 0 - no selection preference 1 - selection using soft key preferred **bit 2 to 7**: 0 **bit 8**:

#STGI - SIM Tookit Get Information	
	0 - no help information available
	1 - help information available
	<numOfItems> - number of items in the list
	<titleText> - string giving menu title
	<itemId> - item identifier
	1..**<numOfItems>**
	<itemText> - title of item
	<nextActionId> - the next proactive command type to be issued upon execution of the menu item.
	0 - no next action information available.
	Note: upon receiving the **#STGI** response, the TA must send **#STSR** command (see below) to confirm the execution of the proactive command and provide any required user response, e.g. selected menu item.
AT#STGI?	The read command can be used to request the currently ongoing **proactive command** and the SAT state in the format
	#STGI: <state>,cmdType>
	where:
	<state> - SAT interface state (see **#STIA)**
	<cmdType> - ongoing proactive command
	An error message will be returned if there is no pending command.
AT#STGI=?	Test command returns the range for the parameters **<state>** and **<cmdType>**.
Note	The unsolicited notification sent to the user:
	#STN: 37
	is an indication that the main menu of the SIM Application has been sent to the TA. It will be stored by the TA so that it can be displayed later at any time by issuing an **AT#STGI=37** command.
	A typical SAT session on AT interface starts after an **#STN: 37** unsolicited code is received, if enabled. At that point usually an **AT#STGI=37** command is issued, and after the SAT main menu has been displayed on TE an **AT#STSR=37,0,x** command is issued to select an item in the menu (see below). The session usually ends with a SIM action like sending an SMS, or starting a call. After this, to restart the session from the beginning going back to SAT main menu it is usually required an **AT#STSR=37,16** command.
	The unsolicited notification sent to the user:
	#STN:237
	is an indication that the main menu of the SIM Application has been removed from the TA, and it is no longer available. In this case **AT#STGI=37** command response

#STGI - SIM Tookit Get Information	
	will be always **ERROR**.

3.5.9.3. SIM Tookit Send Response - #STSR

#STSR - SIM Tookit Send Response	

AT#STSR= [<cmdType>, <userResponse> [,<data>]]	The write command is used to provide to SIM user response to a command and any required user information, e.g. a selected menu item.
	
	Parameters:
	<cmdType> - integer type; **proactive command** ID according to 3GPP TS 31.111 (see **#STGI**)
	<userResponse> - action performed by the user
	  0 - command performed successfully (call accepted in case of call setup)
	  16 - proactive SIM session terminated by user
	  17 - backward move in the proactive SIM session requested by the user
	  18 - no response from user
	  19 - help information required by the user
	  20 - USSD/SS Transaction terminated by user
	  32 - TA currently unable to process command
	  34 - user has denied SIM call setup request
	  35 - user cleared down SIM call before connection or network release
	<data> - data entered by user, depending on **<cmdType>**, only required if **<Result>** is 0:
	<p align=center>***Get Inkey***</p>**<data>** contains the key pressed by the user; used character set should be the one selected with **+CSCS**
	Note: if, as a user response, a binary choice (Yes/No) is requested by the SIM application using bit 3 of the **<commandDetails>** parameter the valid content of the **<inputString>** is:
	a) "IRA", "8859-1", "PCCP437" charsets: "Y" or "y" (positive answer) and "N" or "n" (negative answer)
	b) UCS2 alphabet "0079" or "0059" (positive answer) and "006E" or "004E" (negative answer)
	
	<p align=center>***Get Input***</p>**<data>** - contains the string of characters entered by the user (see above)
	
	<p align=center>***Select Item***</p>**<data>** - contains the item identifier selected by the user
	
	Note:
	Use of icons is not supported. All icon related actions will respond with no icon available.

AT#STSR?	The read command can be used to request the currently ongoing **proactive command** and the SAT state in the format

#STSR - SIM Tookit Send Response	
	#STSR: <state>,<cmdType> where: **<state>** - SAT interface state (see **#STIA)** **<cmdType>** - ongoing proactive command An error message will be returned if there is no pending command.
AT#STSR=?	Test command returns the range for the parameters **<state>** and **<cmdType>**.

3.5.10. Qualcomm Proprietary AT Commands

3.5.10.1. AT Commands for Mobile IP (Not applicable for RUIM version)

3.5.10.1.1. Network Access Identifier - $QCMIPNAI

$QCMIPNAI – Network Access Identifier	
AT$QCMIPNAI= <nai>,<store_nv>	This command sets the network access identifier. Parameter: **<nai>** - Network access identifier (20,21,23-7E) which is the range of printable ASCII characthers. **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPNAI?	Read command returns the current status in format: **$QCMIPNAI: <nai>,<store_nv>**
AT$QCMIPNAI=?	Returns the range of parameters. $QCMIPNAI: (20,21,23-7E),(0,1) Note: 1st parameter of $QCMIPNAI always returns (20,21,23-7E) which is the range of printable ASCII characters. The maximum size is 70bytes.
Example	AT$QCMIPNAI=? $QCMIPNAI: (20,21,23-7E),(0,1) OK AT$QCMIPNAI? $QCMIPNAI: Unset OK AT$QCMIPNAI=5C9F421F@hcm.sprintpcs.com,1 OK AT$QCMIPNAI? 5C9F421F@hcm.sprintpcs.com,1 OK

3.5.10.1.2. Primary Home Agent Address - $QCMIPPHA

$QCMIPPHA – Primary Home Agent Address

$QCMIPPHA – Primary Home Agent Address	
AT$QCMIPPHA= <address>,<store_nv>	This command sets the primary home agent address. Parameter: **<address>** - IP address IP address of primary home agent address. **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPPHA?	Read command returns the current status in format: **$QCMIPPHA: <address>,<store_nv>**
AT$QCMIPPHA=?	Returns the range of parameters. $QCMIPPHA: ((0-255).(0-255).(0-255).(0-255)),(0,1)
Example	AT$QCMIPPHA=? $QCMIPPHA: ((0-255).(0-255).(0-255).(0-255)),(0,1) OK AT$QCMIPPHA? $QCMIPPHA: 63.168.238.41,1 OK AT$QCMIPPHA=255.255.255.255,0 OK AT$QCMIPPHA? $QCMIPPHA: 255.255.255.255,0 OK AT$QCMIPPHA=63.168.238.41,1 OK AT$QCMIPPHA? $QCMIPPHA: 63.168.238.41,1 OK

3.5.10.1.3. *Secondary Home Agent Address - $QCMIPSHA*

$QCMIPSHA – Secondary Home Agent Address	
AT$QCMIPSHA= <address>,<store_nv>	This command sets the secondary home agent address. Parameter: **<address>** - IP address IP address of secondary home agent address. **<store_nv>** - Data store option 0: store in RAM 1: store in NV

$QCMIPSHA – Secondary Home Agent Address	
AT$QCMIPSHA?	Read command returns the current status in format: **$QCMIPSHA: <address>,<store_nv>**
AT$QCMIPSHA=?	Returns the range of parameters. $QCMIPSHA: ((0-255).(0-255).(0-255).(0-255)),(0,1)
Example	AT$QCMIPSHA=? $QCMIPSHA: ((0-255).(0-255).(0-255).(0-255)),(0,1) OK AT$QCMIPSHA? $QCMIPSHA: 63.168.238.41,1 OK AT$QCMIPSHA=255.255.255.255,0 OK AT$QCMIPSHA? $QCMIPSHA: 255.255.255.255,0 OK AT$QCMIPSHA=63.168.238.41,1 OK AT$QCMIPSHA? $QCMIPSHA: 63.168.238.41,1

3.5.10.1.4. *Home Address - $QCMIPHA*

$QCMIPHA – Home Address	
AT$QCMIPHA= <address>,<store_nv>	This command sets the home address. Parameter: **<address>** - IP address IP address of home address. **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPHA?	Read command returns the current status in format: **$QCMIPHA: <address>,<store_nv>**
AT$QCMIPHA=?	Returns the range of parameters. $QCMIPHA: ((0-255).(0-255).(0-255).(0-255)),(0,1)
Example	AT$QCMIPHA=? $QCMIPHA: ((0-255).(0-255).(0-255).(0-255)),(0,1) OK

$QCMIPHA – Home Address	
	AT$QCMIPHA? $QCMIPHA: 0.0.0.0,1 OK AT$QCMIPHA=255.255.255.255,0 OK AT$QCMIPHA? $QCMIPHA: 255.255.255.255,0 OK AT$QCMIPHA=0.0.0.0,1 OK AT$QCMIPHA? $QCMIPHA: 0.0.0.0,1 OK

3.5.10.1.5.   *Home Agent Shared Secret - $QCMIPMHSSX*

$QCMIPMHSSX – set the MIP password	
AT$QCMIPMHSSX= <password>,<store_nv>	This command sets the MIP password. Parameter: **<password>** - Password **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPMHSSX?	Read command returns the current status in format: **$QCMIPMHSSX: <set>** **<set>** - setting status **Set** – parameter is set **Unset** – parameter is not set Note : the value is not displayed.
AT$QCMIPMHSSX=?	Returns the range of parameters. $QCMIPMHSSX: [0x00-0xFF] (max 16 bytes),(0,1)
Example	AT$QCMIPMHSSX=? $QCMIPMHSSX: [0x00-0xFF] (max 16 bytes),(0,1) OK AT$QCMIPMHSSX? $QCMIPMHSSX: Unset OK AT$QCMIPMHSSX=00,1

$QCMIPMHSSX – set the MIP password	
	OK AT$QCMIPMHSSX? $QCMIPMHSSX: Set OK

3.5.10.1.6. *AAA Server Shared Secret - $QCMIPMASSX*

$QCMIPMASSX – AAA server shared secret	
AT$QCMIPMASSX= <password>,<store_nv>	This command sets the MIP password Parameter: **<password>** - Password **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPMASSX?	Read command returns the current status in format: **$QCMIPMASSX: <set>** **<set>** - setting status Set – parameter is set Unset – parameter is not set Note : the value is not displayed.
AT$QCMIPMHSSX=?	Returns the range of parameters. $QCMIPMHSSX: [0x00-0xFF] (max 16 bytes),(0,1)
Example	AT$QCMIPMASSX=? $QCMIPMASSX: [0x00-0xFF] (max 16 bytes),(0,1) OK AT$QCMIPMASSX? $QCMIPMASSX: Unset OK AT$QCMIPMASSX=00,1 OK AT$QCMIPMASSX? $QCMIPMASSX: Set OK

3.5.10.1.7. *Home Agent Security Parameter Index - $QCMIPMHSPI*

$QCMIPMHSPI – set the MIP security parameter index	

$QCMIPMHSPI – set the MIP security parameter index	
AT$QCMIPMHSPI= <index>,<store_nv>	This command sets the MIP security parameter index. Parameter: **<index>** - Security parameter index 0-4294967295 **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPMHSPI?	Read command returns the current status in format: $QCMIPMHSPI: <index>,<store_nv>
AT$QCMIPMHSPI=?	Returns the range of parameters. $QCMIPMHSPI: (0-4294967295),(0,1)
Example	AT$QCMIPMHSPI=? $QCMIPMHSPI: (0-4294967295),(0,1) OK AT$QCMIPMHSPI? $QCMIPMHSPI: 3,1 OK AT$QCMIPMHSPI=4,0 OK AT$QCMIPMHSPI? $QCMIPMHSPI: 4,0 OK

3.5.10.1.8.   *AAA Server Security Parameter Index - $QCMIPMASPI*

$QCMIPMASPI – set the MIP AAA server security parameter index	
AT$QCMIPMASPI= <index>,<store_nv>	This command sets the MIP AAA server security parameter index. Parameter: **<index>** - Security parameter index 0-4294967295 **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPMASPI?	Read command returns the current status in format: $QCMIPMASPI: <index>,<store_nv>
AT$QCMIPMASPI=?	Returns the range of parameters. $QCMIPMASPI: (0-4294967295),(0,1)

$QCMIPMASPI – set the MIP AAA server security parameter index	
Example	AT$QCMIPMASPI=? $QCMIPMASPI: (0-4294967295),(0,1) OK AT$QCMIPMASPI? $QCMIPMASPI: 3,1 OK AT$QCMIPMASPI=4,0 OK AT$QCMIPMASPI? $QCMIPMASPI: 4,0 OK

3.5.10.1.9. *Reverse Tunneling Preference - $QCMIPRT*

$QCMIPRT – Reverse tunneling preference	
AT$QCMIPRT= <rev_tun>,<store_nv>	This command sets the reverse tunneling preference. Parameter: **<rev_tun>** - Reverse tunneling preference 0 : disable 1 : enable **<store_nv>** - Data store option 0: store in RAM 1: store in NV
AT$QCMIPRT?	Read command returns the current status in format: $QCMIPRT: <rev_tun>,<store_nv>
AT$QCMIPRT=?	Returns the range of parameters. $QCMIPRT: (0,1),(0,1)
Example	AT$QCMIPRT=? $QCMIPRT: (0,1),(0,1) OK AT$QCMIPRT? $QCMIPRT: 0,0 OK AT$QCMIPRT=1,1 OK AT$QCMIPRT? $QCMIPRT: 1,1

$QCMIPRT – Reverse tunneling preference	
	OK

3.5.10.1.10. Enable/Disable Mobile IP - $QCMIP

$QCMIP – Enable/Disable mobile IP	
AT$QCMIP=<n>	This command enables/disables mobile IP.

Parameter:
<n>
0 : Mobile IP disable, simple IP only.
1 : Mobile IP preferred. In the initial MIPregistration, if the network does not support Mobile IP, then the mobile automatically reverts to Simple IP (force a PPP Renegotiation by sending a LCP C-Req).
However, if a Mobile IP session is registered, and then the mobile enters a network that does not support Mobile IP, the mobile will drop the session and inform the upper layers of the failure (for example, by dropping DCD to a laptop).

2 : Mobile IP only. The mobile will make data calls only when Mobile IP is supported in the network. During a MIP session, if the mobile hands off to a network that does not support MIP, then the mobile will drop the session and inform the upper layers of the failure (for example, by dropping DCD to a laptop). This value is stored in NV.
Note: If module is provisioned ,the default value is 2 for Sprint and Aeris.Net which supports only Mobile IP and the default value is 1 for Verizon which supports both Simple IP and Mobile IP. |
| AT$QCMIP? | Read command returns the current status in format:

$QCMIP: <n> |
| AT$QCMIP=? | Returns the range of parameters.

$QCMIP: (0-2) |
| Example | AT$QCMIP=?
$QCMIP: (0-2)

OK
AT$QCMIP?
$QCMIP: 2

OK
AT$QCMIP=0
OK
AT$QCMIP?
$QCMIP: 0

OK
AT$QCMIP=1 |

$QCMIP – Enable/Disable mobile IP	
	OK AT$QCMIP? $QCMIP: 1 OK

3.5.10.1.11.   *Active MIP Profile Selection - $QCMIPP*

$QCMIPP – active MIP user profile selection	
AT$QCMIPP= <index>	This command selects the active MIP user profile. Parameter: <index> - User profile number, 0-5 Note: This value is stored in NV memory. This AT command is expected to be used by users to configure Dial-Up Networking
AT$QCMIPP?	Read command returns the current status in format: $QCMIPP: <index>
AT$QCMIPP=?	Returns the range of parameters. $QCMIPP: (0-5)
Example	AT$QCMIPP=? $QCMIPP: (0-5) OK AT$QCMIPP? $QCMIPP: 2 OK

3.5.10.1.12.   *Enable / Disable Current MIP Profile - $QCMIPEP*

$QCMIPEP – current MIP profile	
AT$QCMIPEP=<n>	This command enables/disables the currently active MIP profile. Parameter: <n> 0: Disable the currently active profile (profile is unavailable until it is re-enabled). 1: Enable the currently active profile.
AT$QCMIPEP?	Read command returns the current status in format: $QCMIPEP: <n>
AT$QCMIPEP=?	Returns the range of parameters.

$QCMIPEP – current MIP profile	
	$QCMIPEP: (0,1)
Example	AT$QCMIPEP=? $QCMIPEP: (0,1) OK AT$QCMIPEP? $QCMIPEP: 1 OK AT$QCMIPEP=0 OK AT$QCMIPEP? $QCMIPEP: 0 OK

3.5.10.1.13.   *Profile Information – $QCMIPGETP*

$QCMIPGETP – profile information	
AT$QCMIPGETP=<n>	This command returns all information corresponding to the specified profile number. Parameter: <n> Profile number, 0-5. Note: If no profile number is entered, all information corresponding to the currently active profile is returned. If there is no profile associated with the specified number, an error is returned
AT$QCMIPGETP?	Read command returns the current status in format: $QCMIPGETP: <n>
AT$QCMIPGETP=?	Returns the range of parameters. $QCMIPGETP: (0-5)
Example	AT$QCMIPGETP=? $QCMIPGETP: (0-5) OK AT$QCMIPGETP? $QCMIPGETP: 0 OK AT$QCMIPGETP=0 Profile:0 Disabled NAI:Unset Home Addr:0.0.0.0

$QCMIPGETP – profile information	
	Primary HA:255.255.255.255 Secondary HA:0.0.0.0 MN-AAA SPI:2 MN-HA SPI:3 Rev Tun:0 MN-AAA SS:Set MN-HA SS:Set OK

3.5.10.1.14.   MN-AAA Shared Secrets - $QCMIPMASS

$QCMIPMASS – MN-AAA shared secrets	
AT$QCMIPMASS= <val>,<store_nv>	This command sets the MN-AAA shared secrets for the currently active MIP profile. Parameter: **<val> -** Shared secret data (Max size is 16bytes) Note: Double quotes are only required if the string contains a comma. **<store_nv>** - Data store option 0: store in RAM 1: store in NV Note: If the value provisioned is not committed to NV, the temporary values will be deleted at the end of the following call or if $QCMIPP is called.
AT$QCMIPMASS?	Displays the current setting
AT$QCMIPMASS=?	Returns the range of parameters. $QCMIPMASS: (20,21,23-7E),(0,1)
Example	AT$QCMIPMASS=secret data OK AT$QCMIPMASS? $QCMIPMASS: Set OK AT$QCMIPMASS=? $QCMIPMASS: (20,21,23-7E),(0,1) OK

3.5.10.1.15.   MN-HA Shared Secrets - $QCMIPMHSS

$QCMIPMHSS – MN-HA shared secrets	

$QCMIPMHSS – MN-HA shared secrets	
AT$QCMIPMHSS= <val>,<store_nv>	This command sets the MN-HA shared secrets for the currently active MIP profile. Parameter: **<val>** - Shared secret data (Max size is 16bytes) Note: Double quotes are only required if the string contains a comma. **<store_nv>** - Data store option 0: store in RAM 1: store in NV Note: If the value provisioned is not committed to NV, the temporary values will be deleted at the end of the following call or if $QCMIPP is called.
AT$QCMIPMHSS?	Displays the current setting
AT$QCMIPMHSS=?	Returns the range of parameters. $QCMIPMHSS: (20,21,23-7E),(0,1)
Example	AT$QCMIPMHSS? $QCMIPMHSS: Unset OK AT$QCMIPMHSS=secret data OK AT$QCMIPMHSS? $QCMIPMHSS: Set OK AT$QCMIPMHSS=? $QCMIPMHSS: (20,21,23-7E),(0,1) OK

3.5.10.2. AT Commands for device service option

3.5.10.2.1.   *Medium Data Rate - $QCMDR*

$QCMDR – the medium data rate setting	
AT$QCMDR = <value>	This command changes the medium data rate settings. Parameter: **<value>** - Set medium data rate 0 : MDR service only 1 : MDR service if available 2 : LSPD only 3 : SO 33, if available Note: When the AT$QCMIP=1 or 2, AT$QCMDR is always fixed to '3' and not changeable to other values. It is necessary to change $QCMIP=0 first to change

$QCMDR – the medium data rate setting	
	$QCMDR to 0~3 and it also means not using Mobile IP but Simple IP only.
AT$QCMDR?	Returns the current setting of Medium Data Rate: $QCMDR: <value>
AT$QCMDR=?	Returns the range of parameters. $QCMDR: (0-3)
Example	AT$QCMDR=? $QCMDR: (0-3) OK AT$QCMDR? $QCMDR: 3 OK AT$QCMDR=3 OK

3.5.11. FOTA/OMA-DM for the Sprint Network

3.5.11.1. Configuration Commands

3.5.11.1.1. *OMA-DM Server Address - #OMADMSVADDR*

#OMADMSVADDR – OMA-DM Server Address	
AT#OMADMSVADDR=\<url\>	This command sets OMA-DM server address. Parameter: **\<url\>** - OMA-DM server address Factory default server address for Sprint OMA-DM server is: https://oma.ssprov.sprint.com/oma Note: URL should be started with "https://" or "http://" string
AT#OMADMSVADDR?	Reports the current OMA-DM server address: #OMADMSVADDR: \<url\>
AT#OMADMSVADDR=?	Test command returns the OK result code
Example	AT#OMADMSVADDR=? OK AT#OMADMSVADDR= https://oma.ssprov.sprint.com/oma OK AT#OMADMSVADDR? #OMADMSVADDR : https://oma.ssprov.sprint.com/oma OK

3.5.11.1.2. *OMA-DM Server Port - #OMADMSVPORT*

#OMADMSVPORT – OMA-DM Server Port	
AT#OMADMSVPORT=\<port_num\>	This command sets OMA-DM server port. Parameter: **\<port_num\>** - OMA-DM server port Factory default server port for Sprint OMA-DM server is: 443
AT#OMADMSVPORT?	Reports the current OMA-DM server port: #OMADMSVPORT: \<port_num\>
AT#OMADMSVPO	Test command returns the OK result code

#OMADMSVPORT – OMA-DM Server Port	
RT=?	
Example	AT#OMADMSVPORT=? OK AT#OMADMSVPORT? #OMADMSVPORT : 443 OK AT#OMADMSVPORT=550 OK AT#OMADMSVPORT? #OMADMSVPORT: 550 OK

3.5.11.1.3. *OMA-DM Proxy Server Address - #OMADMPROXY*

#OMADMPROXY – OMA-DM Proxy Server Address	
AT#OMADMPROXY=<port_num>,<url>	This command sets the OMA-DM proxy server address. Parameter: **<port_num>** - OMA-DM proxy server port number. Factory default for Sprint is: 80 **<url >** - URL OMA-DM proxy server URL. Factory default URL for Sprint is http://oma.ssprov.sprint.com
AT#OMADMPROXY?	Reports the current OMA-DM proxy server address: #OMADMSVPORT: <url>:<port_num>
AT#OMADMPROXY =?	Test command returns the OK result code
Example	AT#OMADMPROXY=? OK AT#OMADMPROXY? #OMADMPROXY : http://oma.ssprov.sprint.com:80 OK AT#OMADMPROXY=120,http://www.telit.com OK AT#OMADMPROXY? #OMADMPROXY: http://www.telit.com:120 OK AT#OMADMPROXY=80,http://68.31.28.1 OK

3.5.11.1.4. *OMA-DM Server ID - #OMADMSVID*

#OMADMSVID – OMA-DM server ID	
AT#OMADMSVID?	This command is only read for the OMA-DM server ID. According to the "Sprint OMA-DM Requirements v2.54" the server id is "sprint". Reports the current OMA-DM server ID: #OMADMSVID: \<id>
AT#OMADMSVID = ?	Test command returns the OK result code
Example	AT#OMADMSVID=? OK AT#OMADMSVID? #OMADMSVID: sprint OK

3.5.11.1.5. *OMA-DM Server Password - #OMADMSVPW*

#OMADMSVPW – OMA-DM server password	
AT#OMADMSVPW?	This command is only read for the OMA-DM server authentication secret. The server password is calculated with algorithm as according to the "Sprint OMA-DM Requirements v2.54" Reports the current OMA-DM server auth secret: #OMADMSVPW: \<pw>
AT#OMADMSVPW =?	Test command returns the OK result code
Example	AT#OMADMSVPW=? OK AT#OMADMSVPW? #OMADMSVPW : yMIiklJdGhj57vwr07SpHP OK

3.5.11.1.6. *OMA-DM Server Auth Data - #OMADMSVNON*

#OMADMSVNON – OMA-DM server auth data	
AT#OMADMSVNON?	This command is only read for the OMA-DM server authentication data. According to the "Sprint OMA-DM Requirements v2.54", the server authentication data is server nonce. Reports the current OMA-DM server authentication data:

#OMADMSVNON – OMA-DM server auth data	
	#OMADMSVID: <nonce>
AT#OMADMSVNON =?	Test command returns the OK result code
Example	AT#OMADMSVNON=? OK AT#OMADMSVNON? #OMADMSVNON: yQOaxLLRhibE8hLBBzhUWA== OK

3.5.11.1.7. *OMA-DM Client ID - #OMADMCUID*

#OMADMCUID – OMA-DM client ID	
AT#OMADMCUID?	This command is only read for the OMA-DM client ID. According to the "Sprint OMA-DM Requirements v2.54" the client id is the individual MEID. Reports the current OMA-DM client ID: #OMADMCUID: <id>
AT#OMADMCUID =?	Test command returns the OK result code
Example	AT#OMADMCUID=? OK AT#OMADMCUID? #OMADMCUID: MEID:A1000009DF0004 OK

3.5.11.1.8. *OMA-DM Client Password - #OMADMCUPW*

#OMADMCUPW – OMA-DM client password	
AT#OMADMCUPW ?	This command is only read for the OMA-DM client authentication secret. The client password is calculated with algorithm as according to the "Sprint OMA-DM Requirements v2.54" Reports the current OMA-DM client password: #OMADMCUPW: <pw>
AT#OMADMCUPW =?	Test command returns the OK result code
Example	AT#OMADMCUPW=? OK AT#OMADMCUPW?

#OMADMCUPW – OMA-DM client password	
	#OMADMCUPW : EsLIH173IYk04BMiOttgpq OK

3.5.11.1.9. *OMA-DM Client Auth Data - #OMADMCUNON*

#OMADMCUNON – OMA-DM client auth data	
AT#OMADMCUNON?	This command is only read for the OMA-DM client authentication data. According to the "Sprint OMA-DM Requirements v2.54", the client authentication data is client nonce. Reports the current OMA-DM client authentication data: #OMADMCUNON: <nonce>
AT#OMADMCUNON =?	Test command returns the OK result code
Example	AT#OMADMCUNON=? OK AT#OMADMCUNON? #OMADMCUNON: eWhHQlJTR3M3cHRnVHhDSg== OK

3.5.11.2. Session Control Commands

3.5.11.2.1. *OMA-DM Client Enable/Disable- #OMADMCEN*

#OMADMCEN – OMA-DM Client Enable/Disable	
AT#OMADMCEN=<onoff>	This command enables/disabled the OMA-DM Client feature. Parameter: **<onoff>** - OMA-DM Client Status 0: disable 1: enable (default)
AT#OMADMCEN?	Reports the current OMA-DM client status: #OMADMCEN: <onoff>
AT#OMADMCEN=?	Test command returns the OK result code
Example	AT#OMADMCEN=? #OMADMCEN: (0,1)

#OMADMCEN – OMA-DM Client Enable/Disable	
	OK AT#OMADMCEN? #OMADMCEN: 1 OK AT#OMADMCEN=0 OK AT#OMADMCEN? #OMADMCEN: 0 OK AT#OMADMCEN=1 OK OMA-DM service ready #900

3.5.11.2.2. *OMA-DM Device Configuration - +OMADM*

+OMADM– OMA-DM Device Configuration	
AT+OMADM=<onoff>	This command initiates an OMA-DM client initiated device configuration (CIDC). Parameter: **<onoff>** - Device configuration function status 2: initiate CIDC Note: This AT+OMADM command is Sprint requirement and it follows the format defined by Sprint document "AT Command Requirements v1.9.2".
Example	// CIDC (OMA-DM client device configuration) initiation. AT+OMADM=2 OK

3.5.11.2.3. *OMA-DM NIPRL/CIPRL - +PRL*

+PRL – OMA-DM NIPRL / CIPRL	
AT+PRL=<onoff>	This command initiates an OMA-DM CIPRL session, i.e. the downloading of a new /updated PRL. Parameter: **<onoff>** - PRL configuration function status 2: check now (initiate CIPRL) Note: This AT+OMADM command is Sprint requirement and it follows the format defined by Sprint document "AT Command Requirements v1.9.2".

+PRL – OMA-DM NIPRL / CIPRL	
Example	// Perform a client initiated PRL update AT+PRL=2 OK

3.5.11.2.4. *OMA-DM NIFUMO/CIFUMO - +FUMO*

+FUMO – OMA-DM NIFUMO / CIFUMO	
AT+FUMO=<onoff>	This command sets OMA-DM NIFUMO/CIFUMO enable parameter. Parameter: <onoff> - FUMO configuration function status 2: check now (check and initiate CIFUMO) Note: This AT+OMADM command is Sprint requirement and it follows the format defined by Sprint document "AT Command Requirements v1.9.2".
Example	// Perform a client initiated FUMO session AT+FUMO=2 OK

3.5.11.2.5. *Hands Free Activation - #HFA*

#HFA – Initiates the Sprint Hands Free Activation (HFA) session	
AT#HFA	This command initiates the "Sprint Hands Free Activation" (HFA) session.
AT#HFA=?	Test command returns the OK result code
Example	AT#HFA=? OK AT#HFA OK

3.5.11.2.6. *Device Configuration Cancel - #DCCANCEL*

#DCCANCEL – Cancels the current device configuration DM session	
AT#DCCANCEL	This command cancels the current device configuration DM session.
AT#DCCANCEL=?	Test command returns the OK result code
Example	AT#DCCANCEL=? OK AT#DCCANCEL OK

3.5.11.2.7. *Load PRL Cancel - #PRLCANCEL*

#PRLCANCEL – Cancels the new PRL load session	
AT#PRLCANCEL	This command cancels the current PRL update DM session.
AT#PRLCANCEL=?	Test command returns the OK result code
Example	AT#PRLCANCEL=? OK AT#PRLCANCEL OK

3.5.11.2.8. *Cancel current FUMO DM session - #FUMOCANCEL*

#FUMOCANCEL – Cancels the current FUMO DM session	
AT#FUMOCANCEL	This command cancels the current FUMO DM session.
AT#FUMOCANCEL=?	Test command returns the OK result code
Example	AT#FUMOCANCEL=? OK AT#FUMOCANCEL OK

3.5.11.2.9. *Hands Free Activation Cancel - #HFACANCEL*

#HFACANCEL – Cancels the current HFA DM session	
AT#HFACANCEL	This command cancels the current HFA DM session.
AT#HFACANCEL=?	Test command returns the OK result code
Example	AT#HFACANCEL=? OK AT#HFACANCEL OK

3.5.12. Verizon Specific AT commands

3.5.12.1. General Commands

3.5.12.1.1. *MEID & ESN - #MEIDESN*

#MEIDESN – This command reports the MEID or the ESN of the module.	
AT#MEIDESN?	Read command returns the MEID or the ESN of the module in format: **#MEIDESN: <meid>,<esn_dec>,<esn_hex>** Parameter: **<meid>** - string 14-digit decimal of MEID **<esn_dec>** - string 11-digit decimal of ESN **<esn_hex>** - string 8-digit hexadecimal of ESN Note: If modem is MEID equipped, values of <esn_dec> and <esn_hex> field are all '0'. If modem is ESN equipped, values of <meid> field are all '0'.
AT#MEIDESN=?	Test command returns the **OK** result code.
Example	at#meidesn? #MEIDESN: A1000009D11111,000000000000,00000000 OK at#meidesn=? OK

3.5.12.1.2.   *Alert Sound Setting - #ALERTSND*

#ALERTSND – Alert Sound Setting	

AT#ALERTSND= [<index>,<onoff>…]	This command enables/disables the alert sounds for the device. Enables or disables the modem's alert sounds.
	
	Parameter:
	<index>
	  0: All alert sound
	  1: Ready sound (not available) (default: 0)
	  2: SMS alert sound. (default: 1)
	  3: Emergency call alert sound. (default: 0)
	  4: Roaming alert sound. (default: 0)
	  5: No service alert sound. (default: 1)
	
	<onoff>
	  0: Off
	  1: On
	
	Note: Number of Index can be increased later

AT#ALERTSND?	Read command returns current alert sound setting in the format: **#ALERTSND: <onoff(for index 1)>,<onoff(for index 2)>, …**
AT#ALERTSND=?	Reports the range of supported values for parameter **< index >,< onoff >**
Example	AT#ALERTSND? #ALERTSND: 0,1,0,0,1 OK AT#ALERTSND=2,0 OK AT#ALERTSND? #ALERTSND: 0,0,0,0,1 OK AT#ALERTSND=0,1 <- All alert sound on. OK AT#ALERTSND? #ALERTSND: 1,1,1,1,1 OK AT#ALERTSND=0,0 <- All alert sound off. OK AT#ALERTSND? #ALERTSND: 0,0,0,0,0

	OK AT#ALERTSND=2,1 OK AT#ALERTSND=5,1 OK AT#ALERTSND? #ALERTSND: 0,1,0,0,1 OK AT#ALERTSND=? #ALERTSND: (0-5),(0,1) OK

3.5.12.1.3.   *Emergency Call Tone Setting - #EMERGALERT*

#EMERGALERT – Emergency Call Tone Setting	
AT#EMERGALERT= <mode>	This command sets the Emergency Call tone. Sets the Emergency Call tone. Parameter: **<mode>** 0: Disable the alert tone for emergency dialing. (default) 1: Enable the alert tone for emergency dialing
AT#EMERGALERT?	Read command reports current Emergency call tone setting in the format: **#EMERGALERT: <mode>**
AT#EMERGALERT=?	Reports the range of supported values for parameter **< mode >**
Example	AT#EMERGALERT=? #EMERGALERT: (0,1) OK AT#EMERGALERT? #EMERGALERT: 0 OK AT#EMERGALERT=1 OK

3.5.12.1.4.   *NAM Lock - #NAMLOCK*

#NAMLOCK – Lock NAM	
AT#NAMLOCK=	This command enables/disables the current NAM Lock of the device.

<n>	

	Parameter:
	<n>
	  0: Disable NAM LOCK (default)
	  1: Enable NAM LOCK

AT#NAMLOCK?	Read command returns the current NAM LOCK setting in the format: **#NAMLOCK: <n>**
AT#NAMLOCK=?	Reports the range of supported values for parameter **< n >**
Example	AT#NAMLOCK=? #NAMLOCK: (0,1) OK AT#NAMLOCK? #NAMLOCK: 0 OK AT#NAMLOCK=1 OK

3.5.12.1.5. *Read Message – +VCMGR*

+VCMGR - Read Message	
AT+VCMGR= <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) Output format for received messages (the information written in *italics* will be present depending on **+CSDH** last setting)*:* **+VCMGR: <stat>,<orig_num>,<callback>,<date>[,<tooa>,<tele_id>,<priority>,<enc_type>,<length>]<CR><LF><data>** Output format for Sent or Unsent messages: **+VCMGR: <stat>,<da>,<callback>,<date>[,<toda>,<tele_id>,<priority>,<enc_type>,<length>]<CR><LF><data>** where: **<stat>** - status of the message "REC UNREAD" - new received message unread

+VCMGR - Read Message	

	"REC READ" - received message read
	"STO UNSENT" - message stored not yet sent
	"STO SENT" - message stored already sent
	<orig_num> - Origination number.
	<da> - Destination number.
	<callback> - Callback number.
	<date> - Received/Sent date in form as "YYYYMMDDHHMMSS".
	<tooa> - Type of <**orig_num**>.
	<toda> - Type of <**da**>.
	<tele_id> - Teleservice ID.
	   4097 - page
	   4098 - SMS message
	   4099 - voice mail notification
	   262144 - voice mail notification
	<priority> - Priority.
	   Note: The priority is different with every carrier.
	   In case of Sprint and Aeris.Net:
	      0 - Normal (factory default)
	      1 - Interactive
	      2 - Urgent
	      3 - Emergency
	   In case of Verizon:
	      0 - Normal (factory default)
	      1 - High
	<enc_type> - Encoding type of message.
	   0 - 8-bit Octet
	   2 - 7-bit ASCII
	   4 - 16-bit Unicode
	<length> - Length of message.
	<data> - Message data.

Miscellaneous	Unsolicited Result Codes - Not applicable Execution Time - Executes immediately. Reference – Verizon Note : Available only under text mode (AT+CMGF=1) . Also, this included sent date as against AT+CMGR
AT+VCMGR=?	Test command returns the **OK** result code
Example	AT+CMGF=1 OK AT+VCMGR=2 +VCMGR: "REC READ","",0111234567",20071221160610,,4098,,16,9 TEST MESSAGE2 OK

+VCMGR - Read Message	
	AT+VCMGR=3 +VCMGR: "STO SENT","01191775982","01096529157",20071221160610,,4098,,16,9 TEST MESSAGE3 OK

3.5.12.1.6.   *List Message - +VCMGL*

+VCMGL - List Messages	

AT+VCMGL
[=<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 SMSs 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)
	
	Parameter:
	<stat>
	  "REC UNREAD" - new message
	  "REC READ" - read message
	  "STO UNSENT" - stored message not yet sent
	  "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):
	
	If there is at least a **Sent/Unsent** message to be listed the representation format is:
	+VCMGL:
	<index>,<stat>,<da>,<callback>,<date>*[,<toda>,<tele_id>,<priority>,<enc_type>,<length>]***<CR><LF> <data>**
	
	If there is at least a **Received** message to be listed the representation format is:
	+VCMGL:
	<index>,<stat>,<orig_num>,<callback>,<date>*[,<tooa>,<tele_id>,<priority>,<enc_type>,<length>]***<CR><LF> <data>**
	
	Where
	<orig_num> - Origination number.
	<da> - Destination number.
	<callback> - Callback number.
	<date> - Received/Sent date in form as "YYYYMMDDHHMMSS".
	<tooa> - Type of <orig_num>.
	<toda> - Type of <da>.
	<tele_id> - Teleservice ID.

+VCMGL - List Messages	
	4097 – page
	4098 - SMS message
	4099 - voice mail notification
	262144 - voice mail notification
	<priority> - Priority.
	Note: The priority is different with every carrier.
	In case of Sprint and Aeris.Net:
	0 - Normal (factory default)
	1 - Interactive
	2 - Urgent
	3 - Emergency
	In case of Verizon:
	0 - Normal (factory default)
	1 - High
	<enc_type> - Encoding type of message.
	0 - 8-bit Octet
	2 - 7-bit ASCII
	4 - 16-bit Unicode
	<length> - Length of message.
	<data> - Message data.
	Note: If a message is present when +CMGL="ALL" is used it will be changed status from **REC UNREAD** to **REC READ**.
Miscellaneous	Unsolicited Result Codes - Not applicable
	Execution Time - Executes immediately.
	Reference – Verizon
	Note : Available only under text mode (AT+CMGF=1) . Also, this included sent date as against AT+CMGL
AT+VCMGL=?	Test command returns a list of supported **<stat>**s
Example	

3.5.12.1.7.   *SMS Mobile Origination - #SMSMOEN*

#SMSMOEN – SMS Mobile Origination	

AT#SMSMOEN **=<n>**	This command sets which SMS MO is available or not.
	
	 Parameter:
	 <n> - Enable or disable SMS MO
	   0 - Disable SMS MO
	   1 - Enable SMS MO (default)

AT#SMSMOEN?	Read command reports the current value of the parameter **<n>**.
AT#SMSMOEN=?	Test command reports the supported value of **<n>** parameter.

#SMSMOEN – SMS Mobile Origination	
Example	AT#SMSMOEN=? #SMSMOEN: (0,1) OK AT#SMSMOEN? #SMSMOEN: 1 OK AT#SMSMOEN=0 OK

3.5.12.1.8. *Service Option for SMS - #SMSSO*

#SMSSO – Service Option for SMS	

AT#SMSSO
=**<n>**
	This command sets service option for SMS.
	
	Parameter:
	<n> - Service Option
	   0 - Service option by default value from NV. This is selected by service option
	   set from NV(6 or 14)
	   6 - Short Message Services (IS-637) (default)
	   14: Short Message Services using MUX Option 2 (TSB-79)

AT#SMSSO?	Read command reports the current value of the parameter **<n>**.
AT#SMSSO=?	Test command reports the supported value of **<n>** parameter.
Example	AT#SMSSO=? #SMSSO: (0,6,14) OK AT#SMSSO? #SMSSO: 6 OK AT#SMSSO=14 OK

3.5.12.1.9. *Set Payload Length - #SMSPSIZ*

#SMSPSIZ – Set Payload Length	
AT#SMSPSIZ	This command set max payload length of SMS.

#SMSPSIZ – Set Payload Length	
=<length>	Parameter: **<length>** - Max payload length of SMS 0-220 (default is 170)
AT#SMSPSIZ?	Read command reports the current value of the parameter **< length >**.
AT#SMSPSIZ =?	Test command reports the supported value of **< length >** parameter.
Example	AT#SMSPSIZ=? #SMSPSIZ: (0-220) OK AT#SMSPSIZ? #SMSPSIZ: 170 OK AT#SMSPSIZ=100 OK AT#SMSPSIZ? #SMSPSIZ: 100

3.5.12.1.10.   *Select transport method to send SMS - #SMSAC*

#SMSAC – Select transport method to send SMS	

AT#SMSAC =<method>	This command is for selecting transport method to send SMS.
	
	Parameter:
	<method> - Transport method
	   0 - Traffic Channel (default)
	   1 - Access Channel

AT#SMSAC?	Read command reports the current value of the parameter **< method >**.
AT#SMSAC=?	Test command reports the supported value of **< method >** parameter.
Example	AT#SMSAC? #SMSAC: 0 OK AT#SMSAC =? #SMSAC: (0-1) OK AT#SMSAC =1 OK

3.5.12.1.11. *Preferred Roaming List - $PRL*

$PRL – Preferred Roaming List	
AT$PRL?	Read command returns the current device PRL id **<id>** in format: **$PRL: <id>**
AT$PRL=?	Test command returns the **OK** result code.
Example	AT$PRL=? OK AT$PRL? $PRL: 10052 OK

3.5.12.1.12. *Display Current Band Class - #BANDCLS*

#BANDCLS – Display Current Band Class	
AT#BANDCLS?	Read command returns the current band class in format: **#BANDCLS: <Current BC>,<Supported BC>**
AT#BANDCLS=?	Test command returns the **OK** result code.
Example	AT#BANDCLS? #BANDCLS: BC0,(BC0,BC1) OK

3.5.12.1.13. *Set Default Band - #DEFAULTBAND*

#DEFAULTBAND – Set Default Band	

AT#DEFAULTBAND=<Band>	This command sets the Band to determine system selection
	Parameter:
	<Band>
	   0 – Home Only
	   1 - Automatic
	   2 – Automatic-A
	   3 – Automatic-B
	
	Note: The Default Band mode is made available when the
	PRL has a PREF ONLY setting set to FALSE,When it is set to FALSE, the mobile station's System select setting shall provide the options of Home Only, Automatic-A, and
	Automatic-B.
	When the PRL is set to TURE, the mobile station's System
	Select shall only provide Home Only and Automatic.

AT#DEFAULTBAND?	Read command reports the current value of the parameters: #DEFAULTBAND:<Band>,<PRL_Enable> Note:PRL_Enable represents the PRL_enable of PRL included in CE910

3.5.12.1.14. *Enhanced Roaming Indicator - #ERI*

#ERI – Enhanced Roaming Indicator	

AT#ERI?	This command returns the Enhanced Roaming Indicator Information.
	
	#ERI:<ind_id>,<icn_img_id>,<icn_mode>,<call_prmt_id>,<alert_id>,
	<eng_type>,<text_data_len>,<text_data>
	
	Note : If ERI file not include or invalid ERI file in the current device and roaming indicator value of PRL is 64~93, mobile set to <ind_id>=2.
	
	Where
	<ind_id> - Indicator ID.
	  0 ~ 2 – Roaming Indicator ID (That means not ERI ID).
	        If <ind_id>=0~2, Roaming Indicator Icon display refers to below.
	          0 - Roaming Icon On.
	          1 - Roaming Icon Off.
	         2 - Roaming Icon Flash.
	  64 ~ 93 – ERI Indicator ID.
	        If <ind_id>=64~93, Roaming Indicator Icon display refers to
	         <inc_img_id>.
	<icn_img_id> - Icon Image ID.
	 0 - Roaming Icon On.
	 1 - Roaming Icon Off.
	 2 - Roaming Icon Flash.
	    If <ind_id>=0~2, <icn_img_ind>=0.
	<icn_mode> - Icon Mode.
	 If <ind_id>=0~2, <icn_mode>=0.
	<call_prmt_id> - Call Prompt ID.
	  If <ind_id>=0~2, <call_prmt_id>=0.
	<alert_id> - Alert ID.
	 0 - Verizon Wireless.
	 1 - Network Extender.
	 2 – None.
	 3 – None.
	 4 - Extended Network.
	 5 – Roaming.
	 6 – None.
	 7 - Loss of Service.
	    If <ind_id>=0~2, <alert_id>=2.
	    If Mobile status is No Serivce, <alert_id>=7.
	<eng_type> - Character Encoding Type.
	 0 - Octet, unspecified.
	 1 - IS91 Extended Protocol Message.
	 2 - 7-bit ASII.
	 3 - IA5(Table 11 of ITU-T T.50).
	 4 - UNICODE (ISO/IEC 10646-1:1993).
	 5 - Shift-JIS.

#ERI – Enhanced Roaming Indicator	

	6 - Korean (KS x 1001:1998).
	7 - Latin/Hebrew (ISO 8859-8:1988).
	8 - Latin (ISO 8859-8:998).
	9 - GSM 7-bit default alphabet.
	   If <ind_id>=0~2, <eng_type>=2.
	<text_data_len> - Amount of Text Data.
	<text_data> - Text Data.
	 If Mobile status is No Serivce (AT+SERVICE? / +SERVICE: 0) ,
	<text_data>=”No Service”.
	  If <ind_id>=0~2, Text Data is None.

AT#ERI=?	Test command returns the **OK** result code.
Example	AT#ERI? #ERI: 71,1,0,0,4,2,16,Extended Network OK AT#ERI? #ERI: 1,0,0,0,2,0,0, OK AT#ERI? #ERI: 1,0,0,0,7,0,10,No Service OK AT#ERI=? OK

3.5.12.1.15. *Enhanced Roaming Indicator Version - #ERIDATA*

#ERIDATA – Enhanced Roaming Indicator Version	
AT#ERIDATA?	Read command returns the current device ERI Version **<eri_data_ver>** in format: **#ERIDATA: <eri_data_ver>** Note : ERI file not include or invalid ERI file in the current device, <eri_data_ver>=”None”. Note : ERI file located in EFS area that you can load ERI file by EFS Explorer of QPST.
AT#ERIDATA=?	Test command returns the **OK** result code.
Example	AT#ERIDATA? #ERIDATA: 5 OK AT#ERIDATA? #ERIDATA: None OK AT#ERIDATA=?

#ERIDATA – Enhanced Roaming Indicator Version	
	OK

3.5.12.1.16.   Call for only one phone number - $ONECALL

$ONECALL – Call for only one phone number	
AT$ONECALL= <n>[,<number>]	This command enables/disables call for only one phone number. Parameter: **<n>** 0: Disable (default) 1: Enable call for only one phone **<number>** **<number>** - string type, phone number Note: **<number>** accepted are 0-9, +*#ABCDPW@!$ and ()- but, the input "(", ")","-" for a format divide are ignored. Note: for backwards compatibility with landline modems modifiers "P", "W", "@", "!", "$" are accepted excepting for ";", ";".
AT$ONECALL?	Read command returns the current status in format: **$ONECALL: <n>,<number>** where: **<n>** - as seen before **<number>** - as seen before
AT$ONECALL=?	Reports the range of supported values for parameter **<n>** and integer type value indicating the maximum length of **<number>**
Example	AT$ONECALL=? $ONECALL: (0,1),20 OK AT$ONECALL? $ONECALL: 0, OK AT$ONECALL=1,0123456789 OK
Reference	V25ter.

3.5.12.1.17.   Tethered NAI Management for MIP- $MIPRMNAI

$MIPRMNAI – Tethered NAI Management for MIP

$MIPRMNAI – Tethered NAI Management for MIP	
AT$MIPRMNAI= <nai_string>	This command sets the tethered NAI for mobile IP. Parameter: <nai_string>
AT$MIPRMNAI?	Read command returns the currently used NAI, in the format: $MIPRMNAI: < nai_string >
Note	The maximum length of NAI is 72(bytes).
Example	AT$MIPRMNAI =1234567890@vzw3g.com OK

3.5.12.1.18. Tethered NAI Management for SIP- $SIPRMNAI

$SIPRMNAI – Tethered NAI Management for SIP	
AT$SIPRMNAI= <nai_string>	This command sets the tethered NAI for simple IP. Parameter: <nai_string>
AT$SIPRMNAI?	Read command returns the currently used NAI, in the format: $SIPRMNAI: < nai_string >
Note	The maximum length of NAI is 72(bytes).
Example	AT$SIPRMNAI =1234567890@vzw3g.com OK

3.5.13. Sprint & Aeris.Net Specific AT commands

3.5.13.1. General Commands

3.5.13.1.1. *Command Echo - +E*

+E – Command Echo	
AT+E<n>	This command enable/disable the command echo Parameter: <n> - integer 0 – disables command echo 1 – enables command echo (factory default), therefore command sent to the device are echoed back to the DTE before the response is given
Miscellaneous	Unsolicited Result Codes - Not applicable Execution Time - Executes immediately. Note : If parameter is omitted, the command has the same behavior of AT+E0 Note : The parameter <n> can be saved in a profile setting, thus command echo can be defaulted on or off based on the profile settings upon power up
AT+E=?	Test command returns the **OK** result code.
Example	AT+E=? OK AT+E1 OK AT+E0 OK <--- "AT" entered here OK <--- "AT+E1" entered here OK AT OK

3.5.13.1.2. *Quite Result Code - +Q*

+Q – Quite Result Code	
AT+Q[0]	This command enables the result codes. Returns the OK result code Note: if parameter is omitted, the command has the same behaviour of AT+Q0
Miscellaneous	Unsolicited Result Codes - Not applicable Execution Time - Executed immediately, not time critical.

+Q – Quite Result Code	
AT+Q?	Returns the OK result code
AT+Q=?	Returns the OK result code
Example	AT+Q=? OK AT+Q? OK AT+Q OK AT+Q0 OK AT+Q1 ERROR AT+Q10 ERROR

3.5.13.1.3.   *Response Format – +V*

+V – Response Format	
AT+V[1]	This command set the response format to full headers and trailers and verbose format of result codes. Returns the OK result code Note: if parameter is omitted, the command has the same behaviour of AT+V1
Miscellaneous	Unsolicited Result Codes - Not applicable Execution Time - Executed immediately, not time critical.
AT+V?	Returns the OK result code
AT+V=?	Returns the OK result code
Example	AT+V=? OK AT+V? OK AT+V OK AT+V1 OK AT+V0 ERROR

+V – Response Format	
	AT+V2 ERROR
	AT+V10 ERROR

3.5.13.1.4. *Firmware Revision - $FWREV*

$FWREV – firmware revision	
AT$FWREV?	Return the current firmware revision $FWREV: xx.xx.xxx-xxxx OK
AT$FWREV=?	Return OK

3.5.13.1.5. *Mobile IP Error Code - $MIPERR*

$MIPERR – Mobile IP error	
AT$MIPERR?	Read command returns the Mobile IP Error code with command echo. Parameter: **<value>** - Returns the last mobile IP registration error code Note: Use in place of $SPMIPERR Error Code Description 0 No error condition. 64 Unknown error. 65 Unknown error. 66 Network error. 67 Registration failure. 68 Network error. 69 Your device requested a session lifetime that is too long. 70 System error. 71 System error. 72 System error. 73 System error. 74 System error. 75 System error. 76 System error. 79 System error. 80 Home Agent Failure. Home Agent addresses may be incorrect or the Home Agent may not be responding. 81 Home Agent Failure. Home Agent addresses may be incorrect or the Home Agent may not be responding. 82 Home Agent Failure. Home Agent addresses may be incorrect or the Home Agent may not be responding. 88 Home Agent Failure. Home Agent addresses may be incorrect or the Home Agent may not be responding.

$MIPERR – Mobile IP error	
	96 System error. 97 Network error. 98 Network error. 99 Network error. 100 Network error. 101 Network error. 104 System error. 105 System error. 106 System error. 128 Unknown error. 129 Unknown error. 130 Network error. 131 Username and/or password may be incorrect. 132 Network error. 133 Username and/or password may be incorrect. 134 System error. 135 System error. 136 Unrecognizable Home Agent Addresses. 137 System error. 138 System error. 139 System error. 140 Network error. 141 Network error.
AT$MIPERR=?	Returns OK
Example	AT$MIPERR=? OK AT$MIPERR? $MIPERR: 0 OK

3.5.13.1.6. *Diagnostic Port Setting - $DIAG*

$DIAG – Diagnostic Port Setting	
AT$DIAG=<value>	This command set the current diagnostic port setting to enabled or disabled Parameter: <value> - Set the diagnostic port settings 0 : Disable diagnostic port 1 : Enable diagnostic port (factory default)
AT$DIAG?	Returns the current diagnostic port setting: $DIAG: <value>
AT$DIAG=?	Returns the range of parameters
Example	AT$DIAG=0

$DIAG – Diagnostic Port Setting	
	OK AT$DIAG=? $DIAG: (0,1) OK

3.5.14. Sprint Specific AT commands

3.5.14.1. General Commands

3.5.14.1.1. *Current Receive Signal Strength Indicator for 1xRTT - $1XRXPWR*

$1XRXPWR – Current Receive Signal Strength Indicator for 1xRTT	
AT$1XRXPWR?	Read command returns the current channel number and corresponding received power in format: **<antenna>,<ch>,<pn>,<rssi>** Parameter: <antenna> - Antenna number <ch> - Channel <pn> - Pilot offset <rssi> - Received power Note: If the device does not support multiple antennas, only one value is returned. If the device supports multiple antennas, the primary antenna is listed first followed by additional antennas.
AT$1XRXPWR=?	Test command returns the **OK** result code.
Example	AT$1XRXPWR? 0,70,86,-52.0 OK AT$1XRXPWR=? OK

3.5.14.1.2. *Current Ec/Io for 1xRTT - $1XECIO*

$1XECIO – Current Ec/Io	
AT$1XECIO?	Read command returns the current PN offset and corresponding pilot strength in format: **<antenna>,<ch>,<pn>,<ecio>** Parameter: <antenna> - Antenna number <ch> - Channel <pn> - Pilot offset <ecio> - Ec/Io Note: If the device does not support multiple antennas, only one value is returned. If the device supports multiple antennas, the primary antenna is

$1XECIO – Current Ec/Io	
	listed first followed by additional antennas.
AT$1XECIO=?	Test command returns the **OK** result code.
Example	AT$1XECIO? 0,70,86,-5.0 OK AT$1XECIO=? OK

3.5.14.1.3. *List commands - +LIST*

+LIST – List commands	
AT+LIST	Execution command causes the ME to return the AT commands that are available for the user, in the following format: <AT cmd>[<CR><LF><AT cmd2>[...]]
AT+LIST=?	Test command returns the **OK** result code.

3.5.14.1.4. *Roaming Reference - $ROAM*

$ROAM – roaming setting	
AT$ROAM= <value>	This command manipulates the roaming settings of the module. Parameter: **<value>** - Set the roaming settings 0 : Sprint only 1 : Automatic (factory default) 2 : Roam Only (It is able to set Aeris.Net only) Note: Use in place of $SPROAM
AT$ROAM?	Returns the current roaming setting: $ROAM: <value>
AT$ROAM=?	Returns the range of parameters. $ROAM: (0,1) or $ROAM: (0-2) (In case of Aeris.Net)
Example	AT$ROAM=? $ROAM: (0,1) OK AT$ROAM? $ROAM: 1 OK AT$ROAM=1 OK

$ROAM – roaming setting	

3.5.14.1.5.   *Current Roaming Indicator - $ERI*

$ERI – Current Roaming Indicator	
AT$ERI?	Read command returns the current roaming indicator value with command echo.
	Returns the current enhanced roaming indicator value.
	$ERI <roam_ind>
	Note: If you see the valid ERI value, ERI supporting PRL of SPRINT must include in the CE910-DUAL Ex) PRL_50509_for_1X_devices_supporting_ERI.prl
AT$ERI=?	Returns OK
Example	AT$ERI? $ERI: 1 OK AT$ERI=? OK

3.5.15. Aeris.NET Specific AT commands

3.5.15.1. General Commands

3.5.15.1.1.   *Current NAM - #CURRNAM*

#CURRNAM – Current NAM	
AT#CURRNAM=<value>	This command sets the NAM to be used. Parameter: <value> - NAM number (0-based digit), 0-1 Note: It is suggested to reboot the module after every #CURRNAM setting.
AT#CURRNAM?	Read command returns the current nam number: #CURRNAM: <value>
AT#CURRNAM=?	Test command returns the **OK** result code.
Example	AT#CURRNAM=? OK AT#CURRNAM? #CURRNAM: 0 OK AT#CURRNAM=1 OK AT#CURRNAM? #CURRNAM: 1 OK

3.5.15.1.2.   *PRL data - #PRLDATA*

#PRLDATA – Write PRL data	
AT#PRLDATA=<nam>	This command allows the PRL data to be changed. Parameter: <nam> - NAM number (0-based digit) Note: PRL data string is hexadecimal. "Ctrl+Z" finishes the PRL data string. If PRL data is successfully written to the modem, then modem will be rebooted automatically.
AT#PRLDATA=?	Test command returns the **OK** result code.

#PRLDATA – Write PRL data	
Example	AT#PRLDATA=? OK AT#PRLDATA=0 > 002f27128080800063181d0de304c4c2d7108bc20000445f18000222b880101115860080 88ad300404459180202786d ← "Ctrl+Z" used to enter information OK ← modem resets

3.5.15.1.3.   *Pseudo Electronic Serial Number - #ESN*

#ESN – Read ESN	
AT#ESN?	This command reports Pseudo electronic serial nimber in the format. **#ESN:<p_esn>** **<p_esn>** : Pseudo electronic serial number(8-digit hexa decimal) Note: This command is only available in MEID equipped. If modem is ESN epuipped return ERROR.
AT#ESN=?	Test command returns the **OK** result code.
Example	**<ESN module>** AT#ESN? ERROR AT#ESN=? OK **<MEID module>** AT#MEID? #MEID: A00000,00000001 OK AT#ESN? #ESN:801D0FC7 OK AT#ESN=? OK

3.5.15.1.4.   *Pseudo Electronic Serial Number - +ESN*

+ESN – Read ESN	
AT+ESN?	This command reports Pseudo electronic serial nimber in the format. **+ESN:<p_esn>** **<p_esn>** : Pseudo electronic serial number(11-digit decimal) Note: This command is only available in MEID equipped. If modem is ESN epuipped return ERROR.
AT+ESN=?	Test command returns the **OK** result code.
Example	**<ESN module>** AT+ESN? ERROR AT+ESN=? OK **<MEID module>** AT#MEID? #MEID: A00000,00000001 OK AT+ESN? 12801904583 OK AT+ESN=? OK

3.5.15.1.5.   *PRI version – #PRI*

#PRI – Read PRI version	
AT#PRI	OK
AT#PRI?	This command reports PRI(Product Release Instruction) version in the format. **#PRI: <v_pri>** **<v_pri>** : Product Release Instruction version
AT#PRI=?	OK

#PRI – Read PRI version	
Example	AT#PRI? #PRI: 1.14.1 OK AT#PRI OK AT#PRI=? OK

3.5.16. Telit Test AT commands

3.5.16.1.1.   *Change Operational Mode of Modem – #MODE*

#MODE – Change Operational Mode of Modem	

AT#MODE=<mode>	This command changes the operational mode of the modem
	
	Parameter:
	<mode> - operational mode selection
	
	  OFFLINE – Offline Mode – All(or most)of tasks are in offline.
	   Similar to power-off, but power is still here.
	  RESET – Resets the module
	  PWROFF – Powers off the module
	  LPM – Low Power Mode – RX/TX turned off, unable to receive network
	  FTM – Factory Test Mode – For RF Tests
	  ONLINE – Online Mode – Returns to normal operation
	
	Note : Some mode change is not possible such as LPM mode to FTM mode.
	
	Note : Please reboot, if a module needs to check an operation of other AT Commands.

AT#MODE?	Returns the current mode of the modem in the format **<mode>**. **#MODE: <mode>**
AT#MODE=?	Test command reports the range of the parameter **<mode>**
Example	AT#MODE=OFFLINE OK AT#MODE=LPM OK AT#MODE=FTM OK AT#MODE=ONLINE OK

#MODE – Change Operational Mode of Modem	
	AT#MODE=RESET
	OK
	AT#MODE=PWROFF
	OK

3.6. AT parser abort

The following AT Command list can be aborted, while executing the AT Command
ATD
ATA
+FRS
+FRH
+FRM
+CLCC
+COPN
+CLIP
+CLIR
NOTE: If DTE transmits any character before receiving the response to the issued AT Command, this make 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	General 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
SCA	Service Center Address
SMS	Short Message Service
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

5. Document History

Revision	Date	Changes
0	2012-07-10	Initial release for MKT/ENG sample
1	2012-11-02	For Verizon Official version - Added Custom AT Commands – Generic Configuration AT Commands - Added Custom AT Commands – Authentication - Added Custom AT Commands – DATA Session AT Commands - Added Custom AT Commands – Verizon Specific AT commands - Added Custom AT Commands – QCT Proprietary AT Commands
2	2013-06-03	For Sprint, Aeris.net and CE910-SC official official version
3	2013-10-14	Updated +CME ERROR codes
4	2013-12-13	For Verizon, Sprint and CE910-SC MR version
5	2014-07-23	For Verizon, Aeris.net MR version For USCelluar official version
6	2015-02-09	For Verizon, Aeris.net and Sprint MR version For CE910-SC MR version
7	2015-06-01	New: #MEIDESN Updated : +CNMI,+CMGL,+CMGR, +CLCK, +CSQ, +GCI, #ATRUNDELAY, #CAP, #DWLRCV, #DWCFG, #FASTSHDN, #GSMAD, #GPIO, #HTTPCFG, #MEIDENS, #MODE, #MODEM, #SMOV, #SMSATRUN, #SMSATRUNCFG, #SMSATWL, #QTEMP, #TCPATCMDSEQ, #TCPATCONSER,#TCPATRUNAUTH, #TCPATRUNCFG, #TCPATRUNCLOSE, #TCPATRUND,#TCPATRUNFRWL, #TCPATRUNL, #FRWL, #TESTMODE
8	2015-07-07	Updated : +CSQ, #CAI
9	2016-02-18	Add CE910-DUAL HW 2.00 New: #RSSICFG, #SSLCFG, #SSLD, #SSLEN, #SSLH,

		#SSLO, #SSLRECV, #SSLS, #SSLSECDATA, #SSLSEND, #SSLSECCFG, #SSLSENDEXT, #RIND, #FTPAPP, #FTPAPPEXT, #FTPCFG, #FTPSIZE, #FTPREST Updated : #TESTMODE, +CPBW, +CPBR, #GPIO, #OMADMPROXY, #VOICEPRIV, ATX, #STIA #PRI, #OMADMSVPORT, $ONECALL, +CPIN, #CGMM, +CPBS,+CPMS