NE866B1/NL865B1 AT Commands Reference Guide

80534ST10817A Rev.4– 2017-12-14

TELIT
TECHNICAL
DOCUMENTATION

MM.YYYY]

Mod. 0871

SPECIFICATIONS ARE 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 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

I. 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.

II. 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

III. 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.

IV. 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.

V. Third Party Rights

The software may include Third Party Right software. In this case you agree to comply with all terms and conditions imposed on you in respect of such separate software. In addition to Third Party Terms, the disclaimer of warranty and limitation of liability provisions in this License shall apply to the Third Party Right software.
TELIT HEREBY DISCLAIMS ANY AND ALL WARRANTIES EXPRESS OR IMPLIED FROM ANY THIRD PARTIES REGARDING ANY SEPARATE FILES, ANY THIRD PARTY MATERIALS INCLUDED IN THE SOFTWARE, ANY THIRD PARTY MATERIALS FROM WHICH THE SOFTWARE IS DERIVED (COLLECTIVELY “OTHER CODE”), AND THE USE OF ANY OR ALL THE OTHER CODE IN CONNECTION WITH THE SOFTWARE, INCLUDING (WITHOUT LIMITATION) ANY WARRANTIES OF SATISFACTORY QUALITY OR FITNESS FOR A PARTICULAR PURPOSE.
NO THIRD PARTY LICENSORS OF OTHER CODE SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND WHETHER MADE UNDER CONTRACT, TORT OR OTHER LEGAL THEORY, ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE OTHER CODE OR THE EXERCISE OF ANY RIGHTS GRANTED UNDER EITHER OR BOTH THIS LICENSE AND THE LEGAL TERMS APPLICABLE TO ANY SEPARATE FILES, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

APPLICABILITY TABLE

PRODUCTS	SW RELEASE
NE866B1-E1	29.00.011
NL865B1-E1	29.00.021

CONTENTS

1 INTRODUCTION

1.1 Scope

Purpose of this document is providing a detailed specification and a comprehensive listing as a reference for the whole set of AT command for the NE866 series (LTE cat.1 modules)

1.2 Audience

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

1.3 Contact Information, Support

For general contact, technical support services, technical questions and report documentation errors contact Telit Technical Support at:

TS-EMEA@telit.com

TS-AMERICAS@telit.com

TS-APAC@telit.com

Alternatively, use:

http://www.telit.com/support

For detailed information about where you can buy the Telit modules or for recommendations on accessories and components visit:

http://www.telit.com

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 Text Conventions

	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

- 3GPP TS 27.007 specification and rules
  http://www.3gpp.org/ftp/Specs/archive/27_series/27.007/
- 3GPP TS 27.005 specification and rules
  http://www.3gpp.org/ftp/Specs/archive/27_series/27.005/
- Hayes standard AT command set

2 OVERVIEW

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

NOTICE:

(EN) The integration of the LTE **NE866B1/NL865B1** cellular module within user application shall be done according to the design rules described in this manual.

(IT) L'integrazione del modulo cellulare LTE **NE866B1/NL865B1** all'interno dell'applicazione dell'utente dovrà rispettare le indicazioni progettuali descritte in questo manuale.

(DE) Die Integration des **NE866B1/NL865B1** LTE Mobilfunk-Moduls in ein Gerät muß gemäß der in diesem Dokument beschriebenen Kunstruktionsregeln erfolgen.

(SL) Integracija LTE **NE866B1/NL865B1** modula v uporabniški aplikaciji bo morala upoštevati projektna navodila, opisana v tem priročniku.

(SP) La utilización del modulo LTE **NE866B1/NL865B1** debe ser conforme a los usos para los cuales ha sido deseñado descritos en este manual del usuario.

(FR) L'intégration du module cellulaire LTE **NE866B1/NL865B1** dans l'application de l'utilisateur sera faite selon les règles de conception décrites dans ce manuel.

(HE) האינטגרטור מתבקש ליישם את ההנחיות המפורטות במסמך זה בתהליך האינטגרציה של המודם הסלולרי **NE866B1/NL865B1** עם המוצר.

The information presented in this document is believed to be accurate and reliable. However, no responsibility is assumed by Telit Communications S.p.A. for its use, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of Telit Communications S.p.A. other than for circuitry embodied in Telit products. This document is subject to change without notice.

3 AT COMMANDS

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

- Hayes standard AT command set, in order to maintain the compatibility with existing SW programs.
- 3GPP TS 27.007 specific AT command and LTE specific commands.

More over Telit wireless module family supports also Telit proprietary AT commands for special purposes.

The following is a description of how to use the AT commands with the Telit wireless module family.

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

3.1 Definitions

The following syntactical definitions apply:

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

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

<...> Name enclosed in angle brackets is a syntactical element. 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 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 have not a Read command, which are called *action type* commands, action should be done on the basis of the recommended default setting of the sub parameter.

3.2 AT Command Syntax

The syntax rules followed by Telit implementation of either Hayes AT commands, Modem 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 commands may be “set” (to store a value or values for later use), “read” (to determine the current value or values stored), or “tested” (to determine ranges of values supported). Each of them has a test command (trailing **=?**) to give information about the type of its 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:
if sub parameters are associated with the action, the ranges of sub parameters values that are supported;
if the command has no sub parameters, issuing the correspondent Test command (trailing **=?**) raises the result code “**ERROR**”.
Note: issuing the Read command (trailing **?**) causes the command to be executed.

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

Action commands don’t store the values of any of their possible sub parameters.

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** are optional, issuing **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 to be a valid string type parameter input. According to V25.ter space characters are ignored on the command line and may be used freely for formatting purposes, unless they are embedded 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 (e.g. typing **AT+COPS=1,0,"A1"** is the same as typing **AT+COPS=1,0,A1**; typing **AT+COPS=1,0,"A BB"** is different from typing **AT+COPS=1,0,A BB**).

A string is always case sensitive.

A small set of commands requires always to write 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/**" or **AT#/** or **at#/**.
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
- **+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

Anyway, it is always preferable to separate into different command lines the basic commands and the extended commands.
Furthermore, it is suggested to avoid placing several action commands in the same command line, because if one of them fails, then an error message is received but it is not possible to argue which one of them has failed the execution.
If command **V1** is enabled (verbose responses codes) and all commands in a command line have been performed successfully, result code **<CR><LF>OK<CR><LF>** is sent from the TA to the TE.

If 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.

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

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

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:

General Errors	
Numeric Format	**Verbose Format**
3	Operation not allowed
4	Operation not supported
5	Need to enter SIM PIN
23	Memory failure
30	No network service
50	Incorrect parameters
57	MT temporarily busy (temporarily out of service due to other MT usage)
159	Uplink busy
257	Socket busy
512	Required parameter not configured
513	Not registered
514	FOTA updating
515	CID is active
516	Radio test state error
517	CID is invalide
518	No NONIP message
519	Link security error
520	Deactive last active CID
521	CID is not defined

4 AT COMMANDS REFERENCES

4.1. Command Line General Format

4.1.1 Command Line Prefixes

4.1.1.1 Starting A Command Line - AT

AT - Starting A Command Line	**SELINT 2**
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, with the only exception of AT#/ prefix
Reference	3GPP TS 27.007

4.1.2 3GPP TS 27.007 AT Commands

4.1.2.1 General

4.1.2.1.1 Request Manufacturer Identification - +CGMI

+CGMI - Request Manufacturer Identification	**SELINT 2**
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

4.1.2.1.2 Request Model Identification - +CGMM

+CGMM - Request Model Identification	**SELINT 2**
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

4.1.2.1.3 Request Revision Identification - +CGMR

+CGMR - Request Revision Identification	**SELINT 2**
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

4.1.2.1.4 Request Product Serial Number Identification - +CGSN

+CGSN - Request Product Serial Number Identification	**SELINT 2**
AT+CGSN	Execution command returns the product serial number, identified as the IMEI of the mobile, without command echo.
AT+CGSN=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

4.1.2.1.5 Serial Number - +GSN

+GSN - Serial Number	SELINT 2
AT+GSN	Execution command returns the device board serial number. Note: The number returned is not the IMSI, it is only the board number
Reference	V.25ter

4.1.2.1.6 Request International Mobile station Equipment Identity and SW Ver- +IMEISV

+IMEISV –Request International Mobile station Equipment Identity and SW Version	SELINT 2
AT+IMEISV	Execution command returns the International Mobile station Equipment Identity and Software Version Number, identified as the IMEISV of the mobile, without command echo. The IMEISV is composed of the following elements (each element shall consist of decimal digits only): • Type Allocation Code (TAC). Its length is 8 digits; • Serial Number (SNR) is an individual serial number uniquely identifying each equipment within each TAC. Its length is 6 digits; • Software Version Number (SVN) identifies the software version number of the mobile equipment. Its length is 2 digits. •
AT+IMEISV=?	Test command returns **OK** result code.
Reference	3GPP TS 23.003

4.1.2.1.7 Request international mobile subscriber identity (IMSI) - +CIMI

+CIMI - Request International Mobile Subscriber Identify (IMSI)	SELINT 2
AT+CIMI	Execution command returns the value of the Internal Mobile Subscriber Identity stored in the SIM without command echo. Note: a SIM card must be present in the SIM card housing, otherwise the command returns **ERROR**.
AT+CIMI=?	Test command returns **OK** result code.
Reference	3GPP TS 27.007

4.1.2.1.8 Command Echo - E

E - Command Echo	SELINT 2
ATE[<n>]	Set command enables/disables the command echo. Parameter: **<n>** 0 - disables command echo 1 - enables command echo (factory default) , hence command sent to the device are echoed back to the **DTE** before the response is given. Note: if parameter is omitted, the command has the same behavior of **ATE0**
Reference	V25ter

4.1.2.1.9 Fixed DTE Interface Rate - +IPR

+IPR - Fixed DTE Interface Rate	SELINT 2
AT+IPR=<rate>	Set command specifies the DTE speed at which the device accepts commands during command mode operations; it may be used to fix the DTE-DCE interface speed. Parameter: <rate> 9600 (default value) 57600 115200 **!!Note: <rate>** - baud rate higher than the fastest speed supported by the Low Power UART, 9600 will disable Deep Sleep Low Power Operation.
AT+IPR?	Read command returns the current value of +IPR parameter.
AT+IPR=?	Test command returns the list of fixed-only <rate> values in the format: +IPR: (list of fixed-only <rate> values)

4.1.2.1.10 dentification Information – I

I - Identification Information	SELINT 2
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

4.1.2.1.11 Command Line Termination Character - S3

S3 - Command Line Termination Character	SELINT 2
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

4.1.2.1.12 Response Formatting Character - S4

S4 - Response Formatting Character	**SELINT 2**
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 **S4 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

4.1.2.1.13 Command Line Editing Character – S5

S5 - Command Line Editing Character	**SELINT 2**
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**)
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

4.1.2.1.14 Extended Error Report – +CEER

+CEER - Extended Error Report	**SELINT 2**
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) or in-call modification; - the last call release; - the last unsuccessful PDP context activation; - the last PDP context deactivation. Note: if none of the previous conditions has occurred since power up then **"Normal, unspecified"** condition is reported
AT+CEER=?	Test command returns **OK** result code.

4.1.2.1.15 setting date format – +CSDF

+CSDF – setting date format	**SELINT 2**
AT+CSDF=[<mode>[,<auxmode>]]	This command sets the date format of the date information presented to the user, which is specified by use of the **<mode>** parameter. The **<mode>** affects the date format on the phone display and doesn't affect the date format of the AT command serial interface, so it not used. The command also sets the date format of the TE-TA interface, which is specified by use of the **<auxmode>** parameter (i.e., the **<auxmode>** affects the **<time>** of AT+CCLK).

+CSDF – setting date format	SELINT 2
	Parameters: **<mode> (dummy parameter, not used)** **<auxmode>:** 1 yy/MM/dd (default) 2 yyyy/MM/dd Note: The <time> format of +CCLK and +CALA is "yy/MM/dd,hh:mm:ss+zz" when <auxmode>=1 and it is "yyyy/MM/dd,hh:mm:ss+zz" when <auxmode>=2.
AT+CSDF?	Read command reports the currently selected **<mode>** and **<auxmode>** in the format: **+CSDF: <mode>,<auxmode>**
AT+CSDF=?	Test command reports the supported range of values for parameters **<mode>** and **<auxmode>**

4.1.2.1.16 Initial PDP context activation – +CIPCA

+CIPCA – Initial PDP context activation	SELINT 2
AT+CIPCA=[<n>[,<AttachWithoutPDN>]]	The set command controls whether an initial PDP context (see subclause 10.1.0) shall be established automatically following an attach procedure when the UE is attached to GERAN or UTRAN RATs and whether the UE is attached to E-UTRAN with or without a PDN connection. For <n>≠0, deactivating the last (active) PDP context can lead to a (re)establishment of the initial PDP context. Changing setting of <n> from 0 to 1 will cause an immediate attempt to (re)establish the initial PDP context if no PDP context is active. Changing <n> from 0 to 2 will if not roaming cause an immediate attempt to (re)establish the initial PDP context if no other PDP context is active. The value of <n>=3 applies to E-UTRAN RATs and does not change the setting of PDP context activation in GERAN or UTRAN RATs. Changing <n> will never cause a PDP context deactivation. For <AttachWithoutPDN>=1, the EPS Attach is performed without a PDN connection. NOTE: For this command, the term roaming corresponds to being registered to a VPLMN which is not equivalent to HPLMN or EHPLMN. Parameters: <**n**>: integer type. Activation of PDP context upon attach. 0 - Do not activate 1 - Always activate 2 - Activate when not roaming 3 - No change in current setting <**AttachWithoutPDN**>: integer type. EPS Attach with or without PDN connection. 0 - EPS Attach with PDN connection 1 - EPS Attach without PDN connection
AT+CIPCA?	The read command returns the current setting of the command.
AT+CIPCA=?	The test command returns values supported as a compound value.

4.1.2.1.17 Clock Management – +CCLK

+CCLK - Clock Management	SELINT 2
AT+CCLK=<time>	Set command sets the real-time clock of the **ME**. Parameter:

+CCLK - Clock Management	SELINT 2
	<time> - current time as quoted string. The actual format depends on +CSDF <auxmode> parameter. For <auxmode>=0(default) - the format is: "**yy**/MM/dd,hh:mm:ss±zz" For <auxmode>=1 – the format is: "**yyyy**/MM/dd,hh:mm:ss±zz" **yy** - year, range is 00..99, which translates to years 2000-2099 **yyyy** – year, range is 1970-2105 **MM** - month (two last digits are mandatory), range is 01..12 **dd** - day (two last digits are mandatory); The range for dd(day) depends on the month and on the year it refers to. Available ranges are: (01..28) (01..29) (01..30) (01..31) Trying to enter an out of range value will raise an error **hh** - hour (two last digits are mandatory), range is 00..23 **mm** - minute (two last digits are mandatory), range is 00..59 **ss** - seconds (two last digits are mandatory), range is 00..59 **±zz** - time zone (indicates the difference, expressed in quarter of an hour, between the local time and GMT; two last digits are mandatory), range is -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=?	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).
Example	AT+CCLK="02/09/07,22:30:00+00" OK AT+CCLK? +CCLK: "02/09/07,22:30:25" OK

4.1.2.1.18 Read ICCID (Integrated Circuit Card Identification) - +CCID

+CCID - Read ICCID	SELINT 2
AT+CCID	Execution command reads on SIM the ICCID (card identification number that provides a unique identification number for the SIM)
AT+CCID=?	Test command returns the **OK** result code.

4.1.2.2 Network Service Handling

4.1.2.2.1 EPS network registration status - +CEREG

+CEREG – EPS network registration status		SELINT 2
+CEREG=[<n>]	The set command controls the presentation of an unsolicited result code **+CEREG: <stat>** when **<n>**=1 and there is a change in the MT's EPS network registration status in E-UTRAN, or unsolicited result code.	

+CEREG – EPS network registration status	SELINT 2
	+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>]] when <**n**>=2 and there is a change of the network cell in E-UTRAN. The parameters **<AcT>**, **<tac>** and **<ci>** are sent only if available. The value <**n**>=3 further extends the unsolicited result code with [,<**cause_type**>,<**reject_cause**>], when available, when the value of <**stat**> changes. **+CEREG:<n>,<stat>[,[<lac>],[<ci>],[<AcT>],[<rac>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]** when <**n**>=4,5 the URC also provides information about PSM timings. **Note:** If the EPS MT in E-UTRAN also supports circuit mode services and/or GPRS services, the +CREG command and +CREG: result codes and/or the +CGREG command and +CGREG: result codes apply to the registration status and location information for those services.
+CEREG?	The read command returns the status of result code presentation and an integer <stat> which shows whether the network has currently indicated the registration of the MT. Location information elements <tac>, <ci> and <AcT>, if available, are returned only when <n>=2 and MT is registered in the network. The parameters [,<cause_type>,<reject_cause>], if available, are returned when <n>=3. **Defined values:** **<n>:** integer type 0 - disable network registration unsolicited result code 1 - enable network registration unsolicited result code **+CEREG: <stat>** 2 - enable network registration and location information unsolicited result code **+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>]]** 3 - enable network registration, location information and EMM cause value information unsolicited result code **+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]** 4 - For a UE that wants to apply PSM, enable network registration and location information unsolicited result code **+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]** 5 - For a UE that wants to apply PSM, enable network registration, location information and EMM cause value information unsolicited result code **+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]** **<stat>:** integer type; indicates the EPS registration status 0 - not registered, MT is not currently searching an operator to register to. 1 - registered, home network. 2 - not registered, but MT is currently trying to attach or searching an operator to register to. 3 - registration denied. 4 - unknown (e.g. out of E-UTRAN coverage).

+CEREG – EPS network registration status	SELINT 2
	5 - registered, roaming. **<tac>:** string type; two byte tracking area code in hexadecimal format (e.g. "00C3" equals 195 in decimal). **<ci>:** string type; four byte E-UTRAN cell ID in hexadecimal format. **<AcT>:** integer type; indicates the access technology of the serving cell. 0 - GSM 1 - GSM Compact 2 - UTRAN 3 - GSM w/EGPRS (see NOTE 3) 4 - UTRAN w/HSDPA (see NOTE 3) 5 - UTRAN w/HSUPA (see NOTE 3) 6 - UTRAN w/HSDPA and HSUPA (see NOTE 3) 7 - E-UTRAN <**cause_type**>: integer type; indicates the type of <**reject_cause**>. 0 - Indicates that <reject_cause> contains an EMM cause value 1 - Indicates that <reject_cause> contains a manufacturer-specific cause <**reject_cause**>: integer type; contains the cause of the failed registration. The value is of type as defined by <cause_type>. <**Active-Time**>: string type; one byte in an 8 bit format. Indicates the Active Time value (T3324) allocated to the UE in E-UTRAN. The Active Time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes). For the coding and the value range, see the GPRS Timer 2 IE in 3GPP TS 24.008 Table 10.5.163/3GPP TS 24.008. See also 3GPP TS 23.682 and 3GPP TS 23.401 <**Periodic-TAU**>: string type; one byte in an 8 bit format. Indicates the extended periodic TAU value (T3412) allocated to the UE in E-UTRAN. The extended periodic TAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see the GPRS Timer 3 IE in 3GPP TS 24.008 Table 10.5.163a/3GPP TS 24.008. See also 3GPP TS 23.682 and 3GPP TS 23.401

+CEREG – EPS network registration status	SELINT 2
	Note 2: 3GPP TS 44.060 [71] specifies the System Information messages which give the information about whether the serving cell supports EGPRS. **Note 3:** 3GPP TS 25.331 [74] specifies the System Information blocks which give the information about whether the serving cell supports HSDPA or HSUPA. **Note 4:** The NE866 supports only the value 7 (E-UTRAN) on **<AcT>**
+CEREG=?	Test command returns values supported as a compound value. **+CEREG: (list of supported <n>s)**
Reference	3GPP TS 27.007

4.1.2.2.2 Signalling connection status - +CSCON

+CSCON - Signalling connection status	SELINT 2
AT+CSCON=[<n>]	This command gives details of the terminal's perceived radio connection status (i.e. to the base-station). It returns an indication of the current state. Note, however, that this state is only updated when radio events, such as send and receive, take place. This means that the current state may be out of date. The terminal may think it is "Connected" yet cannot currently use a base station due to a change in the link quality. The set command controls the presentation of an unsolicited result code +CSCON. <**n**>: integer type 0 disable unsolicited result code 1 enable unsolicited result code +CSCON: <mode> 2 enable unsolicited result code +CSCON: <mode>[,<state>] 3 enable unsolicited result code +CSCON: <mode>[,<state>[,<access>]] If <n>=1, +CSCON: <mode> is sent from the MT when the connection mode of the MT is changed. The set command controls the presentation of an unsolicited result code +CSCON. If <n>=1, *+CSCON: <mode>* is sent from the MT when the connection mode of the MT is changed. If <n>=2 and there is a state within the current mode, *+CSCON: <mode>[,<state>]* is sent from the MT. If <n>=3, *+CSCON: <mode>[,<state>[,<access>]]* is sent from the MT. If setting fails, an MT error, +CME ERROR: <err> is returned. Refer to Chapter 5: Error Values for possible <err> values. When the MT is in UTRAN or E-UTRAN, the mode of the MT refers to idle when no PS signaling connection and to connected mode when a PS signaling connection between UE and network is setup. When the UE is in GERAN, the mode refers to idle when the MT is in either the IDLE state or the STANDBY state and to connected mode when the MT is in READY state. The <state> value indicates the state of the MT when the MT is in GERAN, UTRAN connected mode or EUTRAN. **Note**: Unsolicited notifications are not currently supported. This functionality will be added in a future release. **Note**: Only <n>=0 and <n>=1 are supported. <n>=0 is the default value.
AT+CSCON?	The read command returns the status of result code presentation <n> and an integer <mode> which shows whether the MT is currently in idle mode or connected mode. State information <state> is returned only when <n>=2.

+CSCON - Signalling connection status	SELINT 2
	Radio access type information <access> is returned only when <n>=3. Format is: +CSCON:<n>,<mode>[,<state>[,access]] <**n**>: integer type; unsolicited configuration (see above for valid values) <**mode**>: integer type; indicates the signaling connection status 0 idle 1 connected 2-255 <reserved for future use> <**state**>: integer type; indicates the CS or PS state while in GERAN and the RRC state information if the MT is in connected Mode while in UTRAN and E-UTRAN. 0 UTRAN URA_PCH state 1 UTRAN Cell_PCH state 2 UTRAN Cell_FACH state 3 UTRAN Cell_DCH state 4 GERAN CS connected state 5 GERAN PS connected state 6 GERAN CS and PS connected state 7 E-UTRAN connected state <**access**>: integer type; indicates the current radio access type. 0 Indicates usage of radio access of type GERAN 1 Indicates usage of radio access of type UTRAN TDD 2 Indicates usage of radio access of type UTRAN FDD 3 Indicates usage of radio access of type E-UTRAN TDD 4 Indicates usage of radio access of type E-UTRAN FDD
AT+CSCON=?	Test command returns supported values as a compound value. +CSCON: (list of supported <n>s)
Reference	3GPP TS 27.007

4.1.2.2.3 Operator Selection - +COPS

+COPS - Operator Selection	SELINT 2
AT+COPS= [<mode> [,<format> [,<oper>[,< AcT>]]]]	Set command forces an attempt to select and register the network operator. **<mode>** parameter defines whether the operator selection is done automatically or it is forced by this command to operator **<oper>**. The operator **<oper>** shall be given in format **<format>**. Parameters: **<mode>** 0 - automatic choice (the parameter <oper> will be ignored) (factory default) 1 - manual choice (<oper> field shall be present) 2 - deregister from network; the MODULE is kept unregistered until a **+COPS** with **<mode>=0, 1** or **4** is issued 3 - set only **<format>** parameter (the parameter <oper> will be ignored) 4 - manual/automatic (**<oper>** field shall be present); if manual selection fails, automatic mode (**<mode>=0**) is entered **<format>** 0 - alphanumeric long form (max length 16 digits) 2 - Numeric 5 or 6 digits [country code (3) + network code (2 or 3)] **<oper>** network operator in format defined by **<format>** parameter. **<AcT>** access technology selected: 7 E-UTRAN Note: module supports **<AcT>** parameter value 7 only.

+COPS - Operator Selection	SELINT 2
	Note: **currently values not saved in NVM, and not available after reboot.**
AT+COPS?	Read command returns current value of **<mode>**,**<format>**,**<oper>** and **<AcT>** in format **<format>**; if no operator is selected, **<format>**, **<oper>** and **<AcT>** are omitted **+COPS: <mode>[, <format>, <oper>,< AcT>]** Where **<AcT>** access technology selected: 7 E-UTRAN Note: module supports **<AcT>** parameter value 7 only
AT+COPS=?	Test command returns the currently configured operator, followed by the supported <mode> and <format>s. **+COPS: [currently configured (<stat>,,, numeric <oper>[,<Act>])][,,(list of supported <mode>s), (list of supported <format>s)]** where **<stat>** - operator availability 0 - unknown 1 - available 2 - current 3 - forbidden **<AcT>** access technology selected: 7 E-UTRAN Note: module supports **<AcT>** parameter value 7 only Note: since with this command a network scan is done, this command may require some seconds before the output is given.
Reference	3GPP TS 27.007

4.1.2.2.4 eDRX setting - +CEDRXS

+CEDRXS - eDRX setting	
AT+CEDRXS=[<mode>,[,<AcT-type>[,<Requested_eDRX_value>]]]	The set command controls the setting of the UEs eDRX parameters. The command controls whether the UE wants to apply eDRX or not, as well as the requested eDRX value for each specified type of access technology. The set command also controls the presentation of an unsolicited result code +CEDRXP:<AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]] when <n>=2 and there is a change in the eDRX parameters provided by the network. A special form of the command can be given as +CEDRXS=3. In this form, eDRX will be disabled and data for all parameters in the command +CEDRXS will be removed or, if available, set to the manufacturer specific default values. **<mode>:** integer type, indicates to disable or enable the use of eDRX in the UE. This parameter is applicable to all specified types of access technology, i.e. the most recent setting of <mode> will take effect for all specified values of <AcT>. 0 Disable the use of eDRX 1 Enable the use of eDRX 2 Enable the use of eDRX and enable the unsolicited result code +CEDRXP:<AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]

	3 Disable the use of eDRX and discard all parameters for eDRX or, if available, reset to the manufacturer specific default values. **<AcT-type>:** integer type, indicates the type of access technology. This AT-command is used to specify the relationship between the type of access technology and the requested eDRX value. 0 Access technology is not using eDRX. This parameter value is only used in the unsolicited result code. 1 EC-GSM-IoT (A/Gb mode) 2 GSM (A/Gb mode) 3 UTRAN (Iu mode) 4 E-UTRAN (WB-S1 mode) 5 E-UTRAN (NB-S1 mode) **<Requested_eDRX_value>:** string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. The default value, if available, is manufacturer specific. **<NW-provided_eDRX_value>:** string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. **<Paging_time_window>:** string type; half a byte in a 4 bit format. The paging time window referes to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see the Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. **Note:** only access type <AcT-type>=5 is supported.
+CEDRXS?	The read command returns the current settings for each defined value of <AcT-type>. **[+CEDRXS:<AcT-type>,<Requested_eDRX_value> [<CR><LF>+CEDRXS:<AcT-type>,<Requested_eDRX_value> [...]]]**
+CEDRXS=?	The test command returns the supported <mode>s and the value ranges for the access technology and the requested eDRX value as compound values.
Example	at+cedrxs=1,5,"0101" OK at+cedrxs? +CEDRXS: 5,"0101" OK

4.1.2.2.5 eDRX setting - +CEDRXRDP

+CEDRXRDP - eDRX read dynamic parameters	
AT+CEDRXRDP	The execution command returns <AcT-type> and <Requested_eDRX_value>, <NW-provided_eDRX_value> and <Paging_time_window> if eDRX is used for the cell that the MS is currently registered to. If the cell that the MS is currently registered to is not using eDRX, AcT-type=0 is returned. <AcT-type>: integer type, indicates the type of access technology. This AT-command is used to specify the relationship between the type of access technology and the requested eDRX value. 0 Access technology is not using eDRX 1 EC-GSM-IoT (A/Gb mode) 2 GSM (A/Gb mode)

	3 UTRAN (Iu mode) 4 E-UTRAN (WB-S1 mode) 5 E-UTRAN (NB-S1 mode) <Requested_eDRX_value>: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. <NW-provided_eDRX_value>: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. <Paging_time_window>: string type; half a byte in a 4 bit format. The paging time window referes to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see the Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008.
+CEDRXRDP =?	Test command returns OK

4.1.2.2.6 Setting of 3GPP Rel13 PSM mode run time - +CPSMS

+CPSMS – Power saving mode setting	**SELINT 2**
+CPSMS= [<mode> [,<Requested_Periodic-RAU> [,<Requested_GPRSREADYtimer> [,<Requested_Periodic-TAU> [,<Requested_Active-Time>]]]]]	The set command controls the setting of the UEs power saving mode (PSM) parameters. The command controls whether the UE wants to apply PSM or not, as well as the requested extended periodic RAU value and the requested GPRS READY timer value in GERAN/UTRAN, the requested extended periodic TAU value in EUTRAN and the requested Active Time value. See the unsolicited result codes provided by command +CEREG for the Active Time value and the extended periodic TAU value that are allocated to the UE by the network in E-UTRAN. A special form of the command can be given as +CPSMS=2. In this form, the use of PSM will be disabled and data for all parameters in the command +CPSMS will be removed or, if available, set to the manufacturer specific default values. <**mode**>: integer type. Indication to disable or enable the use of PSM in the UE. 0 - Disable the use of PSM. 1 - Enable the use of PSM. 2 - Disable the use of PSM and discard all parameters for PSM or, if available, reset to the manufacturer specific default values. **<Requested_Periodic-RAU>**: string type; one byte in an 8 bit format. Requested extended periodic RAU value (T3312) to be allocated to the UE in GERAN/UTRAN. The requested extended periodic RAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see the GPRS Timer 3 IE in 3GPP TS 24.008 [8] Table 10.5.163a/3GPP TS 24.008. See also 3GPP TS 23.682 [149] and 3GPP TS 23.060 [47]. The default value, if available, is manufacturer specific. **<Requested_GPRS-READY-timer>**: string type; one byte in an 8 bit format. Requested GPRS READY timer value (T3314) to be allocated to the UE in GERAN/UTRAN. The requested GPRS READY timer value is coded as one byte (octet 2) of the GPRS Timer information element coded as bit format (e.g. "01000011" equals 3 decihours or 18 minutes). For the coding and the value range, see the GPRS Timer IE in 3GPP TS 24.008 [8] Table 10.5.172/3GPP TS 24.008. See also 3GPP TS 23.060 [47]. The default value, if available, is manufacturer

+CPSMS – Power saving mode setting		SELINT 2
	specific. **<Requested_Periodic-TAU>:** string type; one byte in an 8 bit format. Requested extended periodic TAU value (T3412) to be allocated to the UE in E-UTRAN. The requested extended periodic TAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see the GPRS Timer 3 IE in 3GPP TS 24.008 [8] Table 10.5.163a/3GPP TS 24.008. See also 3GPP TS 23.682 [149] and 3GPP TS 23.401 [82]. The default value, if available, is manufacturer specific. **<Requested_Active-Time>**: string type; one byte in an 8 bit format. Requested Active Time value (T3324) to be allocated to the UE. The requested Active Time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes). For the coding and the value range, see the GPRS Timer 2 IE in 3GPP TS 24.008 [8] Table 10.5.163/3GPP TS 24.008. See also 3GPP TS 23.682 [149], 3GPP TS 23.060 [47] and 3GPP TS 23.401 [82]. The default value, if available, is manufacturer specific. **Note1**: RAU and GPRS-READY is not supported by NB-IOT. So for parameters <Requested_Periodic-RAU> and <Requested_GPRSREADYtimer> no value will be output, and any input will be ignored.	
AT+CPSMS?	The read command returns the current parameter values. +CPSMS:<mode>,,,[<Requested_Periodic-TAU>],[<Requested_Active-Time>]	
AT+CPSMS=?	The test command returns the supported <mode>s and the value ranges for the requested extended periodic RAU value and the requested GPRS READY timer value in GERAN/UTRAN, the requested extended periodic TAU value in E-UTRAN and the requested Active Time value as compound values. +CPSMS:(list of supported <mode>s),,,(list of supported <Requested_Periodic-TAU>s),(list of supported <Requested_Active-Time>s)	
Example	AT+CPSMS=1,,,01000011,01000011 OK AT+CPSMS? +CPSMS:1,,,01000011,01000011 OK AT+CPSMS=? +CPSMS:(0,1,2),,,(00000000-11111111), (00000000-11111111) OK	

4.1.2.3 Mobile Equipment Control

4.1.2.3.1 Set Phone functionality - +CFUN

+CFUN - Set Phone Functionality		SELINT 2
AT+CFUN=[<fun>[,<rst>]]	Set command selects the level of functionality in the ME.	

+CFUN - Set Phone Functionality	SELINT 2
	Parameters: <fun> - is the power saving function mode 0 - minimum functionality, NON-CYCLIC SLEEP mode. The first wake-up event, or rising RTS line, stops power saving and takes the ME back to full functionality level <fun>=1. 1 - mobile full functionality with power saving disabled (factory default) 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 1 - reset the device. The device is fully functional after the reset. This value is available only for <fun> = 1 Note: Only <fun> = 0 & 1 are supported. Note:<rst> is dummy and will be ignored. Note: when <fun>=0, in order to register to the network, the user must send +CFUN=1, followed by a +COPS=0 (or 1 with desired <oper>)
AT+CFUN?	Read command reports the current setting of **<fun>.**
AT+CFUN=?	Test command returns the list of supported values for **<fun>** and **<rst>.**
Reference	3GPP TS 27.007

4.1.2.3.2 Available AT Commands - +CLAC

+CLAC - Available AT Commands	SELINT 2
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 cmdn>** - defines the AT command including the prefix **AT**
AT+CLAC=?	Test command returns the **OK** result code
Reference	3GPP TS 27.007

4.1.2.3.3 Extended Signal Quality- +CESQ

+CESQ – Extended Signal Quality	SELINT 2
AT+CESQ	Execution command reports received signal quality parameters in the form: +CESQ: <rxlev>,<ber>,<rscp>,<ecno>,<rsrq>,<rsrp> Where < rxlev > - received received signal strength level 99 - not known or not detectable <ber> - bit error rate (in percent) 99 - not known or not detectable <rscp> - received signal code power 255 - not known or not detectable <ecno> - ratio of the received energy per PN chip to the total received power spectral density 255 - not known or not detectable <rsrq> - reference signal received quality (see 3GPP TS 36.133 subclause 9.1.7). 0 - rsrq < -19.5 dB 1 - -19.5 dB < rsrq < -19 dB 2 - -19 dB < rsrq < -18.5 dB

+CESQ – Extended Signal Quality	SELINT 2
	… 32 - -4 dB < rsrq < -3.5 dB 33 - -3.5 dB < rsrq < -3 dB 34 - -3 dB < rsrq 255 - not known or not detectable <rsrp> - type, reference signal received power (see 3GPP TS 36.133 subclause 9.1.4). 0 - rsrp < -140 dBm 1 - -140 dBm < rsrp < -139 dBm 2 - -139 dBm < rsrp < -138 dBm ... 95 - -46 dBm < rsrp < -45 dBm 96 - -45 dBm < rsrp < -44 dBm 97 - -44 dBm < rsrp 255 not known or not detectable
AT+CESQ =?	Test command returns the supported range of values of the parameters <rxlev>, <ber>, <rscp>, <ecno>, <rsrq>, <rsrp>.
Reference	3GPP TS 27.007

4.1.2.3.4 Signal Quality - +CSQ

+CSQ - Signal Quality	SELINT 2
AT+CSQ	Execution command reports received signal quality indicators in the form: *+CSQ: <rssi>,<ber>* 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 <**ber**> - bit error rate (in percent) 0 - less than 0.2% 1 - 0.2% to 0.4% 2 - 0.4% to 0.8% 3 - 0.8% to 1.6% 4 - 1.6% to 3.2% 5 - 3.2% to 6.4% 6 - 6.4% to 12.8% 7 - more than 12.8% 99 - not known or not detectable **Note**: <ber> is currently not implemented, and will always be 99.
AT+CSQ=?	Test command returns the supported range of values of the parameters <rssi> and <ber>. Note: although +CSQ is an execution command without parameters, ETSI 07.07 requires the Test command to be defined.
Reference	3GPP TS 27.007

4.1.2.4 Mobile Equipment Errors

4.1.2.4.1 Report Mobile Equipment Error - +CMEE

+CMEE - Report Mobile Equipment Error	SELINT 2
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.

+CMEE - Report Mobile Equipment Error	SELINT 2
	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. 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

4.1.2.5 Commands for Packet Domain

4.1.2.5.1 PDN Connection Activate Or Deactivate - +CGACT

+CGACT - PDN Connection Activate Or Deactivate	**SELINT 2**
AT+CGACT= [<state>[,<cid> [,<cid>[,…]]]]	Execution command is used to activate or deactivate the specified PDN Connection(s) Parameters: **<state>** - indicates the state of PDN Connection activation 0 - deactivated 1 - activated **<cid>** - a numeric parameter which specifies a particular PDN Connection definition (see **+CGDCONT** command)
AT+CGACT?	Read command returns the current activation state for all the defined PDN Connections in the format: **+CGACT: <cid>,<state>[<CR><LF>+CGACT: <cid>,<state>[…]]**
AT+CGACT =?	Test command reports information on the supported PDN Connection activation states parameters in the format: **+CGACT: (0,1)**
Example	AT+CGACT=0,1 OK AT+CGACT? +CGACT: 1,0 OK
Reference	3GPP TS 27.007

4.1.2.5.2 GPRS Attach Or Detach - +CGATT

+CGATT -PS Attach Or Detach	**SELINT 2**
AT+CGATT=[<state>]	Execution command is used to attach the terminal to, or detach the terminal from, the Packet Domain service depending on the parameter **<state>**. Parameter: **<state>** - state of Packet Domain attachment 0 - detached 1 - attached
AT+CGATT?	Read command returns the current Packet Domain service state.
AT+CGATT=?	Test command requests information on the supported Packet Domain service states.
Example	AT+CGATT? +CGATT: 0 OK AT+CGATT=? +CGATT: (0,1) OK AT+CGATT=1 OK
Reference	3GPP TS 27.007

4.1.2.5.3 Define PDN connection- +CGDCONT

+CGDCONT - Define PDN connection	**SELINT 2**
+CGDCONT=[<cid> [,<PDP_type>[,<APN> [,<PDP_addr>[,<d_comp>	Set command specifies PDN connection parameter values for a PDN connection identified by the (local) context identification parameter, <cid>. Parameters:

+CGDCONT - Define PDN connection		SELINT 2
[,<h_comp>[,<IPv4AddrAlloc>[,<request_type>[,<PCSCF_discovery>[,<IM_CN_Signalling_Flag_Ind>[,<NSLPI>]]]]]]]]	<cid> - (PDN connection Identifier) numeric parameter which specifies a particular PDN connection definition. 0..max - where the value of max is returned by the Test command. <cid>=0 is initial PDP context, when it is not explicitly set and no other profiles defined it is appears as default profile with APN string provided by network when registered and disappears when another profile being defined. But can be defined explicitly and this way APN string will show the text provided by user. <PDP_type> - (Packet Data Protocol type) a string parameter which specifies the type of packet data protocol. Supports only **"IP" - Internet Protocol v4** and "NONIP" for Non IP PDN connection <APN> - (Access Point Name) a string parameter which is a logical name that is used to select the GGSN or the external packet data network. If the value is empty ("") or omitted, then the subscription value will be requested. APN name can contain up to 64 characters. <PDP_addr>: string type; identifies the MT in the address space applicable to the PDP. **Dummy variable, not used. Expects empty field (no quotation marks)** <d_comp>: integer type; controls PDP data compression. 0 off 1 on (manufacturer preferred compression) 2 V.42bis 3 V.44 <h_comp>: integer type; controls PDP header compression. 0 off 1 on (manufacturer preferred compression) 2 RFC 1144 [105] (applicable for SNDCP only) 3 RFC 2507 [107] 4 RFC 3095 [108] (applicable for PDCP only) <IPv4AddrAlloc>: integer type; controls how the MT/TA requests to get the IPv4 address information. **Dummy variable, not used.** <request_type>: integer type; indicates the type of PDP context activation request for the PDP context. **Dummy variable, not used.** <P-CSCF_discovery>: integer type; influences how the MT/TA requests to get the P-CSCF address. **Dummy variable, not used.** <IM_CN_Signalling_Flag_Ind>: integer type; indicates to the network whether the PDP context is for IMCN subsystem-related signaling only or not. **Dummy variable, not used.** <NSLPI>: integer type; indicates the NAS signaling priority requested for this PDP context: 0 - indicates that this PDP context is to be activated with the value for the low priority indicator configured in the MT. 1 – (default value) indicates that this PDP context is is to be activated with the value for the low priority indicator set to "MS is not configured for NAS signaling low priority". Note: Only <hcomp> and <dcomp> values of 0 are supported. Note: values are saved in NVM.	
AT+CGDCONT?	Read command returns the current settings for each defined context in the format: +CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[,	

+CGDCONT - Define PDN connection	SELINT 2
	<IPv4AddrAlloc>[,<request_type>[,<PCSCF_discovery>[,<IM_CN_Signalling_Flag_Ind>[,<NSLPI>]]]]] <CR><LF><PDP_type>,<APN>,<PDP_addr>[…] NOTE: dummy variables are not printed to the terminal.
AT+CGDCONT=?	Test command returns values supported as a compound value
Reference	3GPP TS 27.007

4.1.2.5.4 Show PDP Address - +CGPADDR

+CGPADDR - Show PDP Address	SELINT 2
AT+CGPADDR=[<cid>[,<cid>[,…]]]	Execution command returns a list of PDN addresses for the specified context identifiers in the format: **+CGPADDR: <cid>,<PDP_addr>[<CR><LF>+CGPADDR: <cid>, <PDP_addr>[…]]** Parameters: **<cid>** - a numeric parameter which specifies a particular PDN connection definition (see **+CGDCONT** command). If no **<cid>** is specified, the addresses for all defined contexts are returned. **<PDP_addr>** - a string that identifies the terminal in the address space applicable to the PDP. The address may be static or dynamic. For a static address, it will be the one set by the **+CGDCONT** command when the context was defined. For a dynamic address it will be the one assigned during the last PDN connection activation that used the context definition referred to by **<cid>**; if no address is available the empty string (“”) is represented as **<PDP_addr>**
AT+CGPADDR=?	Test command returns a list of defined **<cid>**s.
Example	AT+CGPADDR=1 +CGPADDR: 1,”xxx.yyy.zzz.www” OK AT+CGPADDR=? +CGPADDR: (1) OK
Reference	3GPP TS 27.007

4.1.2.5.1 APN rate control - + CGAPNRC

+CGAPNRC – APN rate control	
+CGAPNRC[=<cid>]	The set command returns the APN rate control parameters (see 3GPP TS 24.008 [8]) associated to the provided context identifier <cid>. If the parameter <cid> is omitted, the APN rate control parameters for all active PDP contexts are returned. Parameters: <**cid**>: integer type; specifies a particular PDP context definition (see the +CGDCONT and +CGDSCONT commands). <**Additional_exception_reports**>: integer type; indicates whether or not additional exception reports are allowed to be sent when the maximum uplink rate is reached. This refers to bit 4 of octet 1 of the APN rate control parameters IE as specified in 3GPP TS 24.008 [8] subclause 10.5.6.3.2. 0 Additional_exception_reports at maximum rate reached are not allowed to be sent. 1 Additional_exception_reports at maximum rate reached are allowed to be sent.

	<**Uplink_time_unit**>: integer typ; specifies the time unit to be used for the maximum uplink rate. This refers to bits 1 to 3 of octet 1 of the APN rate control parameters IE as specified in 3GPP TS 24.008 [8] subclause 10.5.6.3.2. 0 unrestricted 1 minute 2 hour 3 day 4 week <**Maximum_uplink_rate**>: integer type; specifies the maximum number of messages the UE is restricted to send per uplink time unit. This refers to octet 2 to 4 of the APN rate control parameters IE as specified in 3GPP TS 24.008 [8] subclause 10.5.6.3.2.
AT+CGAPNRC =?	The test command returns a list of <cid>s associated with secondary and non secondary active PDP contexts.

4.1.3 Custom AT Commands

4.1.3.1 General Configuration AT Commands

4.1.3.1.1 Auto-Attach Property - #AUTOATT

#AUTOATT - Auto-Attach Property	**SELINT 2**
AT#AUTOATT= [<auto>]	Set command enables/disables the TE GPRS auto-attach property. Parameter: **<auto>** 0 - disables GPRS auto-attach property 1 - enables GPRS auto-attach property (factory default): after the command **#AUTOATT=1** has been issued (and at every following startup) the terminal will automatically try to attach to the GPRS service. **Note**: the **auto** value is automatically saved to NVM.
AT#AUTOATT?	Read command reports whether the auto-attach property is currently enabled or not, in the format: **#AUTOATT: <auto>**
AT#AUTOATT=?	Test command reports available values for parameter **<auto>**

4.1.3.1.2 Set the mapping for band and power class - #PCLASS

#PCLASS - Set the mapping for band and power class	**SELINT 2**
AT#PCLASS=<band>,<power_class>	Set the mapping for band and power class. **Parameter**: <**band**> - LTE Band. <**power_class**> - Power class value for band. Supported values: 3, 5
AT#PCLASS?	The read command list all mapping of bands and power class in the format: **#PCLASS: <band1>,<power_class1>[<CR><LF>#PCLASS: <band2>,<power_class2>[…[<CR><LF>#PCLASS: <bandN>,<power_classN>]]]**
AT#PCLASS=?	Test command reports information on the supported bands and power classes in the format: **AT#PCLASS=?**

	#PCLASS: (8,20),(3,5) **OK**
Example	AT#PCLASS? #PCLASS: 8,3 #PCLASS: 20,3 OK AT#PCLASS=8,5 OK AT#PCLASS? #PCLASS: 8,5 #PCLASS: 20,3 OK

4.1.3.1.3 Lock to single BCCH ARFCN - # BCCHLOCK

BCCHLOCK – Lock to single BCCH ARFCN	**SELINT 2**
AT#BCCHLOCK= **<LockedBcch>** **[,<LockedUarfcn>** **[,<LockedPsc>** **[,<LockedEarfcn>** **[,<LockedPci>]]]]**	This command allows to set the single BCCH EARFCN the device must be locked to, selectable within those allowed for the specific product. Parameters: **<LockedBcch>: dummy variable, not used – will print 0.** **<LockedUarfcn>**: **dummy variable, not used – will print 0.** **<LockedPsc>**: **dummy variable, not used – will print 0.** **<LockedEarfcn>:** A number in the range 0-65535 representing the earfcn to search. An <earfcn> value of 0 will remove the earfcn restriction and any associated Physical Cell ID lock. **<LockedPci>:** string type; E-UTRAN physical cell ID in hexadecimal format. Valid range 0 - 1F7. Note: **AT#BCCHLOCK** setting has higher priority than PLMN selection, that is why it is not recommended to use this command together with manual PLMN selection **AT+COPS=1**,... .
AT#BCCHLOCK=?	Test command returns the **OK** result code

4.1.3.1.4 Select Band - #BND

#BND - Select Band	**SELINT 2**
AT#BND=<band>[,<UMTS band>[,<LTE band>]]	Set command selects the current LTE bands. Parameter **<band>:** 0 - (default value) **<UMTS band>:** 0 - (default value) **<LTE band>** values in the range 1 – 4294967295 as a sum of: 128 - B8 524288 - B20 **Note:** This setting is effective after power cycle. **Note:** not all products support all the values of parameter <band>: please refer to test command to find the supported range of values. **Note:** not all products support all the values of parameter **<UMTS band>**: please refer to test command to find the supported range of

	values. **Note:** not all products support all the values of parameter **<LTE band>**: please refer to test command to find the supported range of values (maximum value is the sum representation of supported bands). **Note**: the LTE_band is automatically stored in NVM.
AT#BND?	Read command returns the current selected band in the format: **#BND: <band>,<UMTS band>,<LTE band>**
AT#BND=?	Test command returns the supported range of values of parameters **<band>, <UMTS band> and <LTE band>** LTE bands shown as maximal bit mask for model in DEC.
Example	

4.1.3.1.5 Cell Monitor - #MONI

#MONI - Cell Monitor	**SELINT 2**
AT#MONI[=[<number>]]	**#MONI** is both a set and an execution command. Set command sets the cells, from which extract network related information. Parameter: **<number>** 0 – it is the serving cell (default) 1 – neighbor cells 2..7 – it is not available Execution command **(AT#MONI<CR>)** reports LTE related information for selected cell or cells: a) When extracting data for the serving cell and the network name is known the format is: **#MONI: <netname> RSRP:<rsrp> RSRQ:<rsrq> TAC:<tac> Id:<id> EARFCN:<earfcn> PWR:<dBm> DRX:<drx>** b) When the network name is unknown, the format is: **#MONI: Cc:<cc> Nc:<nc> RSRP:<rsrp> RSRQ:<rsrq> TAC:<tac> Id:<id> EARFCN:<earfcn> PWR:<dBm> DRX:<drx>** c) When extracting data for a neighbor cell, the format is: **#MONI: RSRP:<rsrp> RSRQ:<rsrq> PhysCellId:<pid> EARFCN:<earfcn> PWR:<dBm>dbm** ~~(currently neighbor cell monitoring not available, return OK)~~ where: **<netname>** - name of network operator (currently not available) **<cc>** - country code **<nc>** - network operator code **<rsrp>** - Reference Signal Received Power **<rsrq>** - Reference Signal Received Quality **<tac>** - Tracking Area Code **<id>** - cell identifier **<earfcn>** - E-UTRA Assigned Radio Channel **<dBm>** - received signal strength in dBm **<drx>** - Discontinuous reception cycle length (dummy, always 0) **<pid> -** physical cell id (for neighbor cells)
AT#MONI=?	Test command reports the maximum number of cells from which we can extract information, along with the ordinal number of the current selected cell, in the format: #MONI: (<MaxCellNo>,<CellSet>)

#MONI - Cell Monitor	SELINT 2
	where: <MaxCellNo> - maximum number of cells from which we can extract network related information. (dummy, always 0) <CellSet> - the last setting done with command #MONI.
Examples	*Set command selects the cell 0 in the network* *at#moni=0* *OK*

4.1.3.1.6 Serving Cell Information - #SERVINFO

#SERVINFO - Serving Cell Information	SELINT 2
AT#SERVINFO	Execution command reports information about serving cell, in the format: #SERVINFO: <EARFCN>,<dBM>,[<NetNameAsc>],<NetCode>,<PhysicalCellId>,<TAC>,<DRX>,<SD>,<RSRP> where: <**EARFCN**> - E-UTRA Assigned Radio Channel <**dBM**> - received signal strength in dBm <**NetNameAsc**> - operator name, quoted string type <**NetCode**> - string representing the network operator in numeric format: 5 or 6 digits [country code (3) + network code (2 or 3)] <**PhysicalCellId**> - Physical Cell ID <**TAC**> - Tracking Area Code <**DRX**> - Discontinuous reception cycle length (dummy, always 0) <**SD**> - Service Domain 0 – No Service 1 – CS Only 2 – PS Only 3 – CS & PS <**RSRP**> - Reference Signal Received Power
AT#SERVINFO=?	Test command tests for command existence.

4.1.3.1.7 Read current network status - #RFSTS

#RFSTS – Read current network status	SELINT 2
AT#RFSTS	Execution command reads current network status, in the format: **#RFSTS:** **<PLMN>,<EARFCN>,<RSRP>,<RSSI>,<RSRQ>,<TAC>,<RAC>,[<TXPWR>],<DRX>,<MM>,<RRC>,<CID>,<IMSI>,[<NetNameAsc>],<SD>,<ABND>,<SNR>,<ECL>** **Where:** <**PLMN**> - Country code and operator code(MCC, MNC) <**EARFCN**> **-** E-UTRA Assigned Radio Channel <**RSRP**> **-** Reference Signal Received Power <**RSSI**> **-** Received Signal Strength Indication <**RSRQ** **-** Reference Signal Received Quality <**TAC**> **-** Tracking Area Code **<RAC>** - Routing Area Code (dummy, always FFFF) <**TXPWR**> **-** Tx Power (In traffic only) <**DRX**> - Discontinuous reception cycle Length (cycle length in ms) (dummy, always 0) <**MM**> **-** Mobility Management state (dummy) <RRC> - Radio Resource state <CID> - Cell ID <**IMSI**> - International Mobile Station ID **<NetNameAsc>** - Operator name, quoted string type (currently not available) **<SD>** - Service Domain 0 - No Service 1 - CS only 2 - PS only

#RFSTS – Read current network status	SELINT 2
	3 - CS+PS **<ABND>** - Active Band 1..63according to 3GPP TS 36.101 (dummy, always 0) <**SNR**> - signal to noise ratio <**ECL**> - Current Enhanced Coverage Level (0..2)
AT#RFSTS=?	Test command tests for command existence.

4.1.3.1.8 Query user equipment statistics - #UESTATS

#RFSTS – Query user equipment statistics	SELINT 2
	SET command is used to query UE statistics according to <stat> parameter. Parameters: <stat> - indicates the type of statistics to query: **0 - all statistics in form of:** #UESTATS:<rlcul_data_rate>,<rlc_dl_data_rate>,<mac_ul_data_rate>,<mac_dl_data_rate>, <rlc_ul_bler>,<rlc_dl_bler>,<mac_ul_bler>,<mac_dl_bler>,<ll1_transmitted_bytes>, <ll1_received_bytes>,<total_tb_tx>,<total_tb_rx>,<total_tb_retx>,<total_harq_nack_received>, <allocated>,<free>,<max_free>,<num_allocs>,<num_frees> **1 - throughput statistics in form of:** #UESTATS:<rlcul_data_rate>,<rlc_dl_data_rate>,<mac_ul_data_rate>,<mac_dl_data_rate> **2 - bler statistics in form of:** #UESTATS:<rlc_ul_bler>,<rlc_dl_bler>,<mac_ul_bler>,<mac_dl_bler>,<ll1_transmitted_bytes>, <ll1_received_bytes>,<total_tb_tx>,<total_tb_rx>,<total_tb_retx>,<total_harq_nack_received> **3 - apps memory statistics in form of:** #UESTATS:<allocated>,<free>,<max_free>,<num_allocs>,<num_frees>
AT#UESTATS	Execute command queries all UE statistics in the format: #UESTATS:<rlcul_data_rate>,<rlc_dl_data_rate>,<mac_ul_data_rate>,<mac_dl_data_rate>, <rlc_ul_bler>,<rlc_dl_bler>,<mac_ul_bler>,<mac_dl_bler>,<ll1_transmitted_bytes>, <ll1_received_bytes>,<total_tb_tx>,<total_tb_rx>,<total_tb_retx>,<total_harq_nack_received>, <allocated>,<free>,<max_free>,<num_allocs>,<num_frees>
AT#UESTATS=?	Test command tests for command existence.

4.1.3.1.9 Paging time window value and eDRX setting - #CEDRXS

#CEDRXS - Paging time window value and eDRX setting	SELINT 2
AT#CEDRXS=[<mode>,[,<AcT_type>[,<Requested_Paging	The set command controls the setting of the UEs paging time window value and eDRX parameters.

#CEDRXS - Paging time window value and eDRX setting		SELINT 2
_time_window> [,<Requested_eDRX _value>]]]]	The command controls whether the UE wants to apply paging time window and eDRX or not, as well as the requested paging time window and eDRX value for each specified type of access technology. The set command also controls the presentation of an unsolicited result code: **#CEDRXP:<AcT_type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>** **[,<Paging_time_window>]]]** when <n>=2 and there is a change in the paging time window and eDRX parameters provided by the network. A special form of the command can be given as #CEDRXS=3. In this form, paging time window and eDRX will be disabled and data for all parameters in the command #CEDRXS will be removed or, if available, set to the manufacturer specific default values. **Parameters:** <**mode**>: integer type, indicates to disable or enable the use of paging time window and eDRX in the UE. This parameter is applicable to all specified types of access technology, i.e. the most recent setting of <mode> will take effect for all specified values of <AcT>. 0 - Disable the use of paging time window and eDRX 1 - Enable the use of paging time window and eDRX 2 - Enable the use of paging time window and eDRX and enable the unsolicited result code: **#CEDRXP:<Act_type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]** 3 - Disable the use of paging time window and eDRX and discard all parameters for paging time window and eDRX or, if available, reset to the manufacturer specific default values. <**Act_type**>: integer type, indicates the type of access technology. This AT-command is used to specify the relationship between the type of access technology and the requested paging time window and paging time window and eDRX value. 0 - Access technology is not using paging time window and eDRX. This parameter value is only used in the unsolicited result code. 1 - EC-GSM-IoT (A/Gb mode) 2 - GSM (A/Gb mode) 3 - UTRAN (Iu mode) 4 - E-UTRAN (WB-S1 mode) 5 - E-UTRAN (NB-S1 mode) <**Requested_eDRX_value**>: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. The default value, if available, is manufacturer specific. <**NW-provided_eDRX_value**>: string type; half a byte in a 4 bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see Extended DRX parameters information element in 3GPP TS 24.008 [8]	

#CEDRXS - Paging time window value and eDRX setting	SELINT 2
	Table 10.5.5.32/3GPP TS 24.008. <**Paging_time_window**>: string type; half a byte in a 4 bit format. The paging time window referes to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 of 3GPP TS 24.008 [8]). For the coding and the value range, see the Extended DRX parameters information element in 3GPP TS 24.008 [8] Table 10.5.5.32/3GPP TS 24.008. **Note:** only access type <Act_type>=5 is supported.
#CEDRXS=?	The read command returns the current settings for each defined value of <AcT_type>: **[#CEDRXS:<AcT_type>,<Requested_Paging_time_window>,<Requested_eDRX_value>** **[<CR><LF>#CEDRXS:<Act_type>,<Requested_Paging_time_window>,<Requested_eDRX_value> [...]]]**
#CEDRXS=?	The read command returns the current settings for each defined value of <AcT_type>: **[#CEDRXS:<AcT_type>,<Requested_Paging_time_window>,<Requested_eDRX_value>** **[<CR><LF>#CEDRXS:<Act_type>,<Requested_Paging_time_window>,<Requested_eDRX_value> [...]]]**
Example	at#CEDRXS=1,5,1110,0101 OK at#CEDRXS? #CEDRXS: 5,1110,0101 OK

4.1.3.1.10 Power saving mode status report - #PSMR

#PSMR - Power saving mode status report	SELINT 2
AT#PSMR=<n>	The read command returns the code presentation <**n**> and current power mode <**mode**> of M: power saving mode or normal mode in the format: **#PSMR: <n>,<mode>** Where <**mode**> is integer type; indicates the power mode of MT. **0 -** normal mode **1 -** power saving mode
AT#PSMR=?	Test command reports supported values in the format: **AT#PSMR=?** **#PSMR: (0,1)** **OK**

4.1.3.1.11 Temperature monitor configuration - #TEMPCFG

#TEMPCFG – Temperature monitor configuration	SELINT 2
AT#TEMPCFG=<TempExLowBound>[,<TempOpLowBound>[,<TempOpUpBound>[,<TempExUpBound>]]]	This parameter command manages the temperature range used by the TEMPMON command Parameters: **<TempExLowBound> -** the extreme temperature lower limit **<TempOpLowBound> -** the operating temperature lower limit **<TempOpUpBound> -** the operating temperature upper limit **<TempExUpBound> -** the extreme temperature upper limit

#TEMPCFG – Temperature monitor configuration	SELINT 2
	Note 1: The extreme temperature lower limit must not be lower than lower limit (see TEMPMON for temperature limits); Note 2: the operating temperature lower limit must be bigger than the extreme temperature lower limit, and not lower than its minimum admitted value (see TEMPMON for temperature limits); Note 3: the operating temperature upper limit must be bigger than the operating temperature lower limit, and not lower than its minimum admitted value (see TEMPMON for temperature limits); Note 4: the extreme temperature upper limit must be bigger than the operating temperature upper limit Note 5: The extreme temperature upper limit must be lower than its upper limit (see TEMPMON for temperature limits). Note 5: a factory reset restores the factory default values. (currently not supported)
AT#TEMPCFG?	read the currently active temperature range : **#TEMPCFG: <TempExLowBound>,** **<TempOpLowBound>,** **<TempOpUpBound>,** **<TempExUpBound>**
AT#TEMPCFG =?	Test command returns the supported range of **<TempExLowBound>,** **<TempOpLowBound>,** **<TempOpUpBound>,** **<TempExUpBound>** parameters**.**

Example	//test the currently set values
	AT#TEMPCFG?
	#TEMPCFG: -30,-10,55,80
	OK
	//set a new temperature range
	AT#TEMPCFG=-40,-15,55,85
	OK
	//read the currently set values
	AT#TEMPCFG?
	#TEMPCFG: -40,-15,55,85
	OK

4.1.3.1.12 Temperature Monitor – #TEMPMON

#TEMPMON - Temperature Monitor	SELINT 2
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. (**not supported**) 1 - triggers the measurement of the module internal temperature, reporting the result in the format: **#TEMPMEAS: <level>,<value>** where: **<level>** - threshold level (see Note) -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. (see Note) 0 - it disables the presentation of the temperature monitor URC

#TEMPMON - Temperature Monitor	SELINT 2
	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 an 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..7 - as a sum of: 0 - no action 1 - automatic shut-down when the temperature is beyond the extreme bounds 2 - RF RX and TX circuits automatically disabled (using **+CFUN=4**) when operating temperature bounds are reached. When the temperature is back to normal the module is brought back to the previous state, before RF RX and TX disabled. 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. **<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 **<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 required. **Note**: currently <**urcmode**>, <**level**>, <**action**>, <**hyst_time**> and <**GPIO**> are dummy variables and their values are ignored.
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>**, **<= 0>**, **<action>**, **<hyst_time>** and **<GPIO>**
Note	The following table is describing the default temperature levels. Extreme Temperature Lower Bound: -30°C Operating Temperature Lower Bound: -10°C Operating Temperature: Operating Temperature Upper Bound: 55°C Extreme Temperature Upper Bound: 80°C

4.1.3.1.13 General Purpose Input/Output Pin Control – #GPIO

#GPIO - General Purpose Input/Output Pin Control	SELINT 2
AT#GPIO=[<pin>, <mode>[,<dir>[,<save]]]	Execution command sets the value of the general purpose output pin **GPIO<pin>** according to **<dir>** and **<mode>** parameter. Not all configurations for the three parameters are valid. Parameters: **<pin>** - GPIO pin number; supported range is from 1 to a value that depends on the hardware. **<mode>** - its meaning depends on **<dir>** setting: 0 - if **<dir>=0** – INPUT, remove any Pull-up/Pull-down - output pin cleared to 0 (**Low**) if **<dir>=1** - OUTPUT - no meaning if **<dir>=2** - ALTERNATE FUNCTION - no meaning if **<dir>=3** – ALTERNATE FUNCTION 1 - if **<dir>=0** – INPUT, if **<dir>=0** – INPUT, remove any Pull-up/Pull-down - output pin set to 1 (**High**) if **<dir>=1** - OUTPUT

#GPIO - General Purpose Input/Output Pin Control	SELINT 2
	- no meaning if **<dir>=2** - ALTERNATE FUNCTION - no meaning if **<dir>=3** – ALTERNATE FUNCTION 2 - Reports the read value from the pin if **<dir>=0** - INPUT - Reports the read value from the pin if **<dir>=1** - OUTPUT - Reports the read value from the pin if **<dir>=2,3,4** – ALT1, ALT2, ALT3 3 - if <**dir**>=**0** – INPUT, enable Pull-Up (see Note) 4 - if <**dir**>=**0** – INPUT, enable Pull-Down **<dir>** - GPIO pin direction 0 - pin direction is INPUT 1 - pin direction is OUTPUT 2,3,4 – alternate functions ALT1, ALT2, ALT3 (see notes) **<save>** - GPIO pin save configuration to NVM 0 – pin configuration is not saved 1 – pin configuration is saved **NE866 Note**: #GPIO command is only effective when <state> parameter in #ALTFUN command is 0 (GPIO+LOG), 1 (I2C+GPIO) or 3 (GPIO only). **NE866 Note**: PULL-UP is not supported, and will return an error. **NL865 Note**: Before any ALT functionality is set, LPWAKE<cond> parameter must be active (<cond>≠0 and <cond>≠1) ALT1 turns the GPIO <pin> into an PULL UP LPWAKE input interrupt pin. ALT2 turns the GPIO <pin> into an PULL DOWN LPWAKE input interrupt pin. ALT3 turns the GPIO <pin> into an NO-PULL LPWAKE input interrupt pin. An interrupt will occur for every edge in input voltage level (rising and falling) See #LPWAKE command for additional info. **Note**: default state for GPIOs (unless previously saved by user) is INPUT NO-PULL. **Note**: when <save> is omitted the configuration is stored only if user set or reset ALTx function on <dir> parameter. **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>[,mode]** 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; - logic value read from pin **GPIO<pin>**in the case the pin **<dir>** is set to alternate function. <**mode**> - will be displayed only for INPUT PULL UP/DOWN modes. Will display “3” for pull up and “4” for pull down.
AT#GPIO?	Read command reports the read direction and value of all **GPIO** pins, in the format: **#GPIO: <dir>,<stat>[,<mode>][<CR><LF>#GPIO: <dir>,<stat>[,<mode>][…]]** where **<dir>,<stat>,**<**mode**> - as seen before

#GPIO - General Purpose Input/Output Pin Control	SELINT 2
AT#GPIO=?	Test command reports the supported range of values of the command parameters **<pin>**, **<mode>** , **<dir> and <save>**.
Example	AT#GPIO=1,1,1 OK AT#GPIO=1,2 #GPIO: 1,1 OK AT#GPIO=1,0,1 OK AT#GPIO=1,2 #GPIO: 1,0 OK AT#GPIO=1,4,0 OK AT#GPIO=1,2 #GPIO: 0,0,4 OK

4.1.3.1.14 Read Analog/Digital Converter input - #ADC

#ADC - Read Analog/Digital Converter input	SELINT 2
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 – default pin **<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 values of the command parameters **<adc>**, **<mode>** and **<dir>**.

4.1.3.1.15 Low Power Wake Configuration - #LPWAKE

#LPWAKE – Low Power Wake Configuration	SELINT 2
#LPWAKE=<cond>[,<action>]	LP_WAKE input line can be configured to trigger an interrupt with certain **conditions**. When an interrupt occurs, an **action** will be carried out. Set command applies configuration of low power wake function. **<cond>:** condition for interrupt to occur. 0 – Interrupt will never occur (disable LP_WAKE). This is the default state. 1 – interrupt will occur on a rising edge 2 – interrupt will occur on a falling edge

#LPWAKE – Low Power Wake Configuration	SELINT 2
	3 – interrupt will occur on a both edges **<action>:** an action to be carried out when an interrupt is triggered. This parameter is mandatory if <cond> is not 0. 1 – send a URC over the UART. **NE866:** The URC format is: **#LPWAKE** **NL865:** The URC format is: **#LPWAKE[: <pin>]** <pin> is an 8 bit hexadecimal integer, and is printed if the source of trigger known. The 7 LSB represent the current value of #GPIO<pin>s. the 8th bit is the value of LPWAKE line, and is updated only if no interrupt occurred through the GPIOs. **NL865 Note**: It is recommended to use LPWAKE + GPIO interrupts with only a single edge chosen (i.e. <cond>=1/2), since an interrupt is toggled (fall + rise) for every edge in the input GPIO line. **Note**: new settings are automatically saved to NVM storage.
#LPWAKE?	Read command reports the current configuration of low power wake function. The format is: **#LPWAKE: <cond>,<action>** Where <cond> is as above. <action>: 0 – no action will be done 1 – URC over UART
#LPWAKE=?	Test command reports the range for parameters <cond> and <action>
Example	AT#LPWAKE=1,1 OK AT#LPWAKE=0 OK

4.1.3.1.16 Alternate Functionality Control - #ALTFUN

#ALTFUN – Alternate Functionality Control – NE866 only	SELINT 2
AT#ALTFUN=<state>	Set command configures the functionality of HW pins between 5 different states:

	HW PAD	B1	C1	C2
<state>	SW-TGPIO# / MODE	1	2	3
0	LOG+GPIO	GPIO	LOG	GPIO
1	I2C+GPIO	I2C SDA	GPIO	I2C SCL
2	I2C+LOG	I2C SDA	LOG	I2C SCL
3	GPIO only	GPIO	GPIO	GPIO
4	SPI	SPI_CS	SPI_MOSI	SPI_MISO

NOTE: by default, <state>=0
NOTE: For SPI functionality, SPI_CLK is connected to pad A1 (not configurable).

#ALTFUN – Alternate Functionality Control – NE866 only	SELINT 2
	NOTE: The commands #SPIWRITE, #SPIREAD, #SPIOPEN and #SPICLOSE are available only when <state>=4. If <state>=4 and an SPI port is enabled, the user will not be allowed to change <state> until all SPI ports are closed. NOTE: The commands #I2CWR and #I2CRD are available only when <state>=1 or <state>=2 NOTE: the #GPIO command is available (with different <pin>s available) for <state>=(0,1,3) only. The test command #GPIO=? Can be consulted. NOTE: the <state> parameter is saved to NVM
AT#ALTFUN?	Read command reports the current <state>
AT#ALTFUN=?	Test command reports the supported range of <state> parameter
Example	AT#ALTFUN=1 OK AT#ALTFUN=0 OK

4.1.3.1.17 Read from I2C- # I2CRD

#I2CRD – Read from I2C	SELINT 2
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. **<sclPin>:** GPIO number to be used for SCL. **<deviceId>:** address of the I2C device, with the LSB, used for read\write command. It doesn't matter if the LSB is set to 0 or to 1. 10 bit addressing supported. Value has to be written in hexadecimal form (without 0x). **<registerId>:** Register to write data to, range 0..255. Value has to be written in hexadecimal form (without 0x). **<len>:** number of data bytes to send. Valid range is 1-254. Data Read from I2C will be dumped in Hex. If data are successfully sent, then the response is **OK**. If data sending fails for some reason, an error code is reported. Example if CheckAck is set and no Ack signal was received on the I2C bus. NOTE: If data requested are more than data available in the device, dummy data (normally 0x00 or 0xff) will be dumped. NOTE: device address and register address where to read from\ write to have to be written in hexadecimal form without 0x. NE866 NOTE: This command depends on the <state> of #ALTFUN command being <state>=1 or <state>=2. NL865 NOTE: Currently this command is not available. NL865 NOTE: Reading and writing to device ID 0x40 is forbidden, and will return an error.
AT#I2CRD=?	Test command shows the range of parameters
Example	AT#ALTFUN? #ALTFUN: 0 << No I2C functionality OK

#I2CRD – Read from I2C	SELINT 2
	AT#I2CRD=1,3,40,02,1 ERROR AT#ALTFUN=1 << I2C now available OK AT#I2CRD=1,3,40,02,1 #I2CRD: 00 OK

4.1.3.1.18 Write to I2C - #I2CWR

#I2CWR – Write to I2C	SELINT 2
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. **<sclPin>:** GPIO number to be used for SCL. **<deviceId>:** address of the I2C device, with the LSB, used for read\write command. It doesn't matter if the LSB is set to 0 or to 1. 10 bit addressing supported. Value has to be written in hexadecimal form (without 0x). **<registerId>:** Register to write data to, range 0..255. Value has to be written in hexadecimal form (without 0x). **<len>:** number of data bytes to send. Valid range is 1-254. The module responds to the command with the prompt '**> **' (<CR><LF><grater_than><space>) and awaits for the data to send. To complete the operation before <len> bytes where inputted, send **Ctrl-Z** char (**0x1A** hex); to exit without writing the message send **ESC** char (**0x1B** hex). If after 2 minutes less then bytes than <bytestosend> are send by the user, an ERROR will occur and the command will be canceled. If data are successfully sent, then the response is **OK**. If data sending fails for some reason, an error code is reported. Example if CheckAck is set and no Ack signal was received on the I2C bus. NOTE: device address, register address where to read from\ write to, and data bytes have to be written in hexadecimal form without 0x. NE866 NOTE: This command depends on the <state> of #ALTFUN command being <state>=1 or <state>=2. NL865 NOTE: Currenty this command is not available. NL865 NOTE: Reading and writing to device ID 0x40 is forbidden, and will return an error.
AT#I2CWR=?	Test command shows the range of parameters
Example	AT#ALTFUN? #ALTFUN: 0 << No I2C functionallity OK AT#I2CWR=1,3,40,20,10 ERROR AT#ALTFUN=1 << I2C now available OK AT#I2CWR=1,3,40,02,10 > 00112233445566778899 OK AT#I2CWR=1,3,40,02,10 > 0011223344<CTRL-Z> OK

#I2CWR – Write to I2C	SELINT 2
	AT#I2CWR=1,3,40,02,10 > 00112233<ESC> ERROR

4.1.3.1.19 Initializes GPIOs with SPI protocol - #SPIOPEN

#SPIOPEN – Initializes GPIOs with SPI protocol - NE866 ONLY	SELINT 2
AT#SPIOPEN=<ID>,<speed>,<mode>[,<data_size>]	This command initializes the GPIO port for SPI protocol: <ID> - supported value is 1 <speed> - supported clock frequency 1 – 1 MHz 2 – 2 MHz 3 – 6 MHz <mode> - clock mode configuration: 0 – Clock signal is active **high** and data is sampled at **rising edge** 1 – Clock signal is active **low** and data is sampled at **falling edge – is 3** 2 – Clock signal is active **high** and data is sampled at **falling edge – is 1** 3 – Clock signal is active **low** and data is sampled at **rising edge – is 2** <data_size> - word size of spi communication. Default is 8 bit per word. NOTE: This command depends on the <state> of #ALTFUN command being <state>=4.
AT#SPIOPEN?	Read command returned configured parameters <ID>,<speed>,<mode>,<data_size>. If SPI is opened. If it is not enabled – it returns #SPIOPEN: 1,0,0,0
AT#SPIOPEN=?	Test command reports available values for <ID>,<speed>,<mode> and <data_size>.
Example	AT#ALTFUN? #ALTFUN: 0 << No SPI functionality OK AT#SPIOPEN=1,3,3 ERROR AT#ALTFUN=4 << SPI now available OK AT#SPIOPEN=1,3,3 OK AT#SPIOPEN? #SPICONF: 1,3,3,8 OK

4.1.3.1.20 De-Initializes GPIOs with SPI protocol - #SPICLOSE

#SPICLOSE – De-Initializes GPIOs with SPI protocol - NE866 ONLY	SELINT 2
AT#SPICLOSE=<ID>	This command de-initializes the SPI protocol on the GPIO ports. <ID> - supported value is 1 NOTE: This command depends on the <state> of #ALTFUN command being <state>=4.
AT#SPICLOSE?	Read command returned current configured <**ID**> (0 as default)

#SPICLOSE – De-Initializes GPIOs with SPI protocol - NE866 ONLY	SELINT 2
AT#SPICLOSE=?	Test command reports available values for parameter <ID>.

4.1.3.1.21 Write Data To An Active SPI Port - #SPIWRITE

#SPIWRITE – Write Data To An Active SPI Port - NE866 ONLY	SELINT 2
AT#SPIWRITE=<cmd>,<data>	This command is used to send data to a previously activated SPI port (via #SPIOPEN). <cmd> - command is received in hexadecimal form (without 0x) inside double quotes. Full bytes must be sent (zero pad if needed), even for #SPIOPEN <data_size> smaller than 8. A command can be comprised of up to 2 words. A command must be sent. <data> - data is receive in hexadecimal form (without 0x) inside double quotes. Full bytes must be sent (zero pad if needed), even for #SPIOPEN <data_size> smaller than 8. Up to 256 bytes of data can be sent with a single command. The data section can be left empty (if only a command is required). NOTE: this command must be sent after an SPI port is enabled via #SPIOPEN NOTE: This command depends on the <state> of #ALTFUN command being <state>=4.
AT#SPIWRITE=?	Test command shows the syntax of the #SPIWRITE command
Example	AT#SPIOPEN=1,1,3 OK AT#SPIWRITE="AF","0102030405" OK AT#SPIWRITE="AF","01020304050" ERROR AT#SPIWRITE="AFBF","" OK

4.1.3.1.22 Read Data From An Active SPI Port - #SPIREAD

#SPIREAD – Read Data From An Active SPI Port – NE866 ONLY	SELINT 2
AT#SPIREAD=<cmd>,<dataLen>	This command is used to send a command to a previously activated SPI port (via #SPIOPEN) and then read <dataLen> amount of words from it. The received data is printed onto the terminal in hexadecimal form. <cmd> - command is received in hexadecimal form (without 0x) inside double quotes. Full bytes must be sent (zero pad if needed), even for #SPIOPEN <data_size> smaller than 8. A command can be comprised of up to 2 words. A command must be sent. <dataLen> - number of bytes to read from the SPI port. Supports up to 256 bytes in a single transfer. NOTE: this command must be sent after an SPI port is enabled via #SPIOPEN NOTE: This command depends on the <state> of #ALTFUN command being <state>=4.
AT#SPIREAD=?	Test command shows the syntax of the #SPIREAD command
Example	AT#SPIOPEN=1,1,3 OK at#spird="0f",10 #SPIRD: 00000000000000000000 OK at#spird="0e",10 #SPIRD: FFFFFFFFFFFFFFFFFFFF

#SPIREAD – Read Data From An Active SPI Port – NE866 ONLY	SELINT 2
	OK

4.1.3.1.23 Reboot - #REBOOT

#REBOOT - Reboot	SELINT 2
AT#REBOOT	Execution command reboots immediately the unit. It can be used to reboot the system after a remote update of the script in order to have the new one running. Note: if AT#REBOOT follows an AT command that stores some parameters in NVM, it is recommended to insert a delay of at least 5 seconds before to issue AT#REBOOT, to permit the complete NVM storing
AT#REBOOT=?	Test command returns **OK** result code.
Example	AT#REBOOT OK … Module Reboots …

4.1.3.1.24 Clock Mode - #CCLKMODE

#CCLKMODE - Clock Mode	SELINT 2
AT#CCLKMODE=<mode>	Set command changes to format used in AT+CCLK to the local time or the UTC time format. **Parameter:** **<mode>** - time and date mode 0 - Local time + local time zone offset (default) 1 – UTC time + local time zone offset **Note:** the setting is saved automatically in NVM. (currently not supported)
AT#CCLKMODE?	Read command reports whether the local time or the UTC time is enabled, in the format: **#CCLKMODE: <mode>** (<mode> described above).
AT#CCLKMODE=?	Test command reports the supported range of values for parameter <mode>.
Example	at#cclkmode? #CCLKMODE: 0 OK

4.1.3.1.25 Network Time zone - #NITZ

#NITZ - Network Time zone	SELINT 2
AT#NITZ=[<val>[,<mode>]]	Set command enables/disables (a) automatic date/time updating, (b) Full Network Name applying (not supported) and (c) **#NITZ** URC; moreover it permits to change the **#NITZ** URC format.

#NITZ - Network Time zone	SELINT 2
	Date and time information can be sent by the network after network registration or after attachment. Parameters: **<val> -** integer (default: 5) 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 – dummy, not supported (will always remain 0) 4 - it sets the **#NITZ** URC *'extended'* format (see **<datetime>** below) 8 – dummy, not supported (will always remain 0) **<mode>** 0 - disables **#NITZ** URC (factory default) 1 - enables **#NITZ** URC; after date and time updating the following unsolicited indication is sent: **#NITZ: <datetime>** where: **<datetime>** - string whose format depends on subparameter **<val>**: **"yy/MM/dd,hh:mm:ss"** - *'basic'* format, if **<val>** is in (0..3) **"yy/MM/dd,hh:mm:ss±zz"** - *'extended'* format, if **<val>** is in (4..7) **"yy/MM/dd,hh:mm:ss±zz,d"** - *'extended'* format with DST support, if **<val>** is in (8..15) where: **yy** - year **MM** - month (in digits) **dd** - day **hh** - hour **mm** - minute **ss** - second **zz** - time zone (indicates the difference, expressed in quarter of an hour, between the local time and GMT; two last digits are mandatory, range is -47..+48)
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>**
Example	AT#NITZ? #NITZ: 5,0 OK AT#NITZ=15,1 OK AT#NITZ? #NITZ: 5,1 OK

4.1.3.2 Multisocket AT Commands

4.1.3.2.1 Context Activation - #SGACT

#SGACT - Context Activation	**SELINT 2**
AT#SGACT=<cid>, <stat>[,<userId>, <pwd>]	Execution command is used to activate or deactivate the specified PDN connection. Parameters: **<cid>** - PDN connection identifier 1..5 - numeric parameter which specifies a particular PDN connection 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: context activation/deactivation returns **ERROR** if there is not any socket associated to it (see **AT#SCFG**). **Note:** In LTE network, default PDN connection(cid 1) is activated by piggybacking on LTE attach procedure and maintained until detached from NW. This command with cid 1 is just binding or unbinding application to the default PDN connection. NOTE: userID and password are currently not supported
AT#SGACT?	Returns the state of all the contexts that have been defined **#SGACT: <cid1>,<Stat1><CR><LF>** **…** **#SGACT: <cid5>,<Stat5>** where: **<cid*n*>** - as **<cid>** before **<stat*n*>** - context status 0 - context deactivated 1 - context activated
AT#SGACT=?	Test command reports the range for the parameters **<cid>** and **<stat>**
Note	It is strongly recommended to use the same command (e.g. **#SGACT**) to activate the context, deactivate it and interrogate about its status.

4.1.3.2.2 Socket Shutdown - #SH

#SH - Socket Shutdown	**SELINT 2**
AT#SH=<connId>	This command is used to close a socket. Parameter: **<connId>** - socket connection identifier 1..6 Note: socket cannot be closed in states “resolving DNS” and “connecting”
AT#SH=?	Test command reports the range for parameter **<connId>**.

4.1.3.2.3 Socket Configuration - #SCFG

#SCFG - Socket Configuration	**SELINT 2**
AT#SCFG= <connId>,<cid> [,<pktSz>[,<maxTo>[,<connTo>[,<txTo>]]]]	Set command sets the socket configuration parameters. Parameters: **<connId>** - socket connection identifier 1..6 **<cid>** - PDN connection identifier 0..5 - numeric parameter which specifies a particular PDN connection definition **<pktSz>** - packet size to be used by the TCP/UDP/IP stack for data sending. **Dummy** parameter, not used. 0 - select automatically default value(300). 1..512 - packet size in bytes.

#SCFG - Socket Configuration	SELINT 2
	<maxTo> - exchange timeout (or socket inactivity timeout); if there's no data exchange within this timeout period the connection is closed. **Dummy** parameter, not used. 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. **Dummy** parameter, not used. 10..1200 - timeout value in hundreds of milliseconds (default 600) **<txTo>** - data sending timeout; after this period data are sent also if they're less than max packet size. **Dummy** parameter, not used. 0 - no timeout 1..255 - timeout value in hundreds of milliseconds (default 50) 256 – set timeout value in 10 milliseconds 257 – set timeout value in 20 milliseconds 258 – set timeout value in 30 milliseconds 259 – set timeout value in 40 milliseconds 260 – set timeout value in 50 milliseconds 261 – set timeout value in 60 milliseconds 262 – set timeout value in 70 milliseconds 263 – set timeout value in 80 milliseconds 264 – set timeout value in 90 milliseconds Note: if DNS resolution is required, max DNS resolution time(20 sec) has to be considered in addition to **<connTo>** **Note: connection timeout is not supported (only relevant to TCP connections)** **Note: max (exchange), connection and tx timeout are not supported** **Note: <cid> parameter is saved to NVM.**
AT#SCFG?	Read command returns the current socket configuration parameters values for all the six sockets, in the format: **#SCFG: <connId1>,<cid1>,<pktsz1>,<maxTo1>,<connTo1>,<txTo1> <CR><LF>** **. . .** **#SCFG: <connId6>,<cid6>,<pktsz6>,<maxTo6>,<connTo6>,<txTo6> <CR><LF>**
AT#SCFG=?	Test command returns the range of supported values for all the subparameters.
Example	at#scfg? #SCFG: 1,1,300,90,600,50 #SCFG: 2,2,300,90,600,50 #SCFG: 3,2,250,90,600,50 #SCFG: 4,1,300,90,600,50 #SCFG: 5,1,300,90,600,50 #SCFG: 6,1,300,90,600,50 OK

4.1.3.2.4 Socket Configuration Extended - #SCFGEXT

#SCFGEXT - Socket Configuration Extended	**SELINT 2**
AT#SCFGEXT= <conned>,<*srMode*>, <*recvDataMode*> [,<*keepalive*>[, <ListenAutoRsp> [,<sendDataMode>]]]	Set command sets the socket configuration extended parameters. Parameters: **<connId>** - socket connection identifier 1..6 **<srMode>** - SRing unsolicited mode 0 - Normal (default): SRING : <connId> where <connId> is the socket connection identifier 1 – Data amount: SRING : <connId>,<recData> where <recData> is the amount of data received on the socket connection number <connId> 2 - Data view: SRING : <connId>,<recData>,<data> same as before and <data> is data received displayed following <dataMode> value 3 – Data view with UDP datagram informations: 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 **<recvDataMode>** - data view mode for received data in command mode(AT#SRECV or <srMode> = 2) 0- text mode (default) 1- hexadecimal mode ***<keepalive>* -** Set the TCP Keepalive value in minutes. **Dummy** parameter – not used, and will always return 0. 0 – Deactivated (default) 1 – 240 – Keepalive time in minutes ***<ListenAutoRsp>* -** Set the listen auto-response mode, that affects the command AT#SLUDP. **Dummy** parameter – not used, and will always return 0. 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 is available only on TCP connections. 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. Note: the **SRING** indication are only indicative on the first package in the queue. The next packages will cause **SRING** indications only after the preceding packages were read using **#SRECV** (or with **<srMode>** 2 and 3). **NOTE: keepalive timer is not supported (relevant only to TCP).** **NOTE2: listen auto response is currently not supported (since ODM is not supported).** **NOTE3: Supported parameters are saved to NVM.**
AT#SCFGEXT?	Read command returns the current socket extended configuration parameters values for all the six sockets, in the format: **#SCFGEXT: <connId1>,<srMode1>,<dataMode1>,<keepalive1>,** <**ListenAutoRsp1**>,<**sendDataMode1**>,<**CR**><**LF**> **...** **#SCFGEXT: <connId6>,<srMode6>,<dataMode6>,<keepalive6>** <**ListenAutoRsp6**>,<**sendDataMode6**>,
AT#SCFGEXT=?	Test command returns the range of supported values for all the subparameters.
Example	Socket 1 set with data amount sring, hex receive data mode, 1 min keepalive (ignored), no auto response and hex send data mode.

	at#scfgext? #SCFGEXT: 1,1,1,1,0,1 #SCFGEXT: 2,0,0,0,0,0 #SCFGEXT: 3,0,0,0,0,0 #SCFGEXT: 4,0,0,0,0,0 #SCFGEXT: 5,0,0,0,0,0 #SCFGEXT: 6,0,0,0,0,0 OK

4.1.3.2.5 Socket configuration Extended 2 - #SCFGEXT2

#SCFGEXT2 - Socket Configuration Extended	**SELINT 2**
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 – used for online mode only (not supported). **Dummy** parameter – not used, and will always return 0. (<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) abort before CONNECT(online mode (not supported)) or OK(command mode). **Dummy** parameter – not used. 0 – Not possible to interrupt connection attempt 1 – It is possible to interrupt the connection attempt 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. **<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 Note: 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. Possible **<cause>** values are: 0 – not available(socket has not yet been closed) 1 - remote host TCP connection close due to FIN/END: normal remote disconnection decided by the remote application 2 - remote host TCP connection close due to RST, all others cases in which the socket is aborted without indication from peer (for instance because

	peer doesn't send ack after maximum number of retransmissions/peer is no more alive). All these cases include all the "FATAL" errors after recv or send on the TCP socket(named as different from EWOULDBLOCK) 3 - socket inactivity timeout 4 - network deactivation(PDN connection deactivation from network) **NOTE: bufferstart is not used since ODM is not supported.** **NOTE2: abortConnAttempt is not used since TCP and DNS are not supported** **NOTE3: <noCarrierMode> is saved to NVM.**
AT#SCFGEXT2?	Read command returns the current socket extended configuration parameters values for all the six sockets, in the format: **#SCFGEXT2:<connId1>,<bufferStart1>,0,0,0,0<CR><LF>** . . . **#SCFGEXT2:<connId6>,<bufferStart6>,0,0,0,0<CR><LF>**
AT#SCFGEXT=?	Test command returns the range of supported values for all the subparameters.
Example	at#scfgext2=2,0,0,0,0,1 OK at#scfgext2? #SCFGEXT2: 1,0,0,0,0,0 #SCFGEXT2: 2,0,0,0,0,1 #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

4.1.3.2.6 Socket info - #SI

#SI – socket info	**SELINT 2**
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 (not supported on LE866) Note: not yet acknowledged data are available only for TCP connections; the value **<ack_waiting>** is always 0 for UDP connections. Note: issuing **#SI<CR>** causes getting information about data traffic of all the sockets; the response format is: **#SI: <connId1>,<sent1>,<received1>,<buff_in1>,<ack_waiting1>** **<CR><LF>** **…** **#SI: <connId6>,<sent6>,<received6>,<buff_in6>,<ack_waiting6>**
AT#SI=?	Test command reports the range for parameter **<connId>**.
Example	AT#SI #SI: 1,123,400,10,50 #SI: 2,0,100,0,0 #SI: 3,589,100,10,100 #SI: 4,0,0,0,0 #SI: 5,0,0,0,0 #SI: 6,0,98,60,0 OK *Sockets 1,2,3,6 are opened with some data traffic.* *For example socket 1 has 123 bytes sent, 400 bytes received, 10 byte waiting to be read and 50 bytes waiting to be acknowledged from the remote side.* AT#SI=1 #SI: 1,123,400,10,50 OK *We have information only about socket number 1*

4.1.3.2.7 Socket Status - #SS

#SS – socket Status	**SELINT 2**
AT#SS[=<connId>]	Execution command reports the current status of the socket: **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: 0 - Socket Closed. 1 - Socket with an active data transfer connection.

#SS – socket Status	SELINT 2
	2 - Socket suspended. 3 - Socket suspended with pending data. 4 - Socket listening. 5 - Socket with an incoming connection. Waiting for the user accept or shutdown command. 6 - Socket resolving DNS. 7 - Socket connecting. **<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>** **NOTE**: currently only states 0, 1 and 4 are supported.
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 OK We have information only about socket number 2

4.1.3.2.8 Socket Dial - #SD

#SD - Socket Dial	SELINT 2
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 when remote host has closed 0 - local host closes immediately (default) 255 - local host closes after an AT#SH or immediately in case of an abortive disconnect from remote. **<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. Note: **<lPort>** parameter is valid for UDP connections only and has no effect (if used) for TCP connections. Note: if we set **<connMode>** to **online mode connection** and the command is successful we enter in **online data mode** and we see the intermediate result code **CONNECT**. After the **CONNECT** we can suspend the direct interface to the socket connection (nb the socket stays open) using the escape sequence (**+++**): the module moves back to **command mode** and we receive the final result code **OK** after the suspension. After such a suspension, it’s possible to resume it in every moment (unless the socket inactivity timer timeouts, see **#SCFG**) by using the **#SO** command with the corresponding **<connId>**. Note: if we set **<connMode>** to **command mode connection** and the command is successful, the socket is opened and we remain in **command mode** and we see the result code **OK**. Note: if there are input data arrived through a connected socket and not yet read because the module entered **command mode** before reading them (after an escape sequence or after **#SD** has been issued with **<connMode>** set to **command mode connection)**, these data are buffered and we receive the **SRING** URC (**SRING** presentation format depends on the last **#SCFGEXT** setting); it’s possible to read these data afterwards issuing **#SRECV**. Under the same hypotheses it’s possible to send data while in **command mode** issuing **#SSEND** Note: resume of the socket(#SO) after suspension or closure(#SH) has to be done on the same instance on which the socket was opened through #SD. In fact, suspension has been done on the instance itself. Note: **<closureType>** 255 takes effect on a command mode connection(**<connMode>** set to 1 or online mode connection suspended with +++) only if **#SCFGEXT3 <closureTypeCmdModeEnabling>** parameter has been previously enabled. Note: if PDN connection has not properly opened then +CME ERROR: 556 (context not opened) will be given. **NOTE: TCP protocol is not supported (So txProt only accepts 1 and closureType only accepts 0 (and is ignored)).** **NOTE2: ODM is not supported, so connMode only accepts 1.**

#SD - Socket Dial	SELINT 2
	NOTE3: No DNS resolution is available, only IP addresses are supported for IPaddr parameter.
AT#SD=?	Test command reports the range of values for all the parameters.
Example	Open socket 1 in online mode at#sd=1,1,9060,"1.1.1.1",0,9061,1 OK

4.1.3.2.9 Socket Listen UDP - #SLUDP

#SLUDP - Socket Listen UDP	SELINT 2
AT#SLUDP=<connId>, <listenState>, <listenPort>	This command opens/closes a socket listening for an incoming UDP connection on a specified port. Parameters: **<connId>** - socket connection identifier 1..6 **<listenState> -** 0 - closes socket listening 1 - starts socket listening **<listenPort>** - local listening port 1..65535 Note: if successful, the command returns a final result code OK. If the ListenAutoRsp flag has not been set through the command AT#SCFGEXT (for the specific connId), then, when an UDP connection request comes on the input port, if the sender is not filtered by internal firewall (see **#FRWL**), an URC is received: **+SRING : <connId>** Afterwards we can use **#SA** to accept the connection or **#SH** to refuse it. If the ListenAutoRsp flag has been set, then, when an UDP connection request comes on the input port, if the sender is not filtered by the internal firewall (see command **#FRWL**), the connection is automatically accepted: the **CONNECT** indication is given and the modem goes into **online data mode**. If the socket is closed by the network the following URC is received: **#SLUDP: ABORTED** Note: when closing the listening socket <listenPort> is a don't care parameter **NOTE: ListenAutoRsp option is not supported (since ODM is not supported)**
AT#SLUDP?	Read command returns all the actual listening UDP sockets.
AT#SLUDP=?	Test command returns the range of supported values for all the subparameters.
Example	*Next command opens a socket listening for UDP on port 3500.* AT#SLUDP=1,1,3500 OK

4.1.3.2.10 Socket Accept - #SA

#SA - Socket Accept	SELINT 2
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.

#SA - Socket Accept	SELINT 2
	Note: setting the command before to having received a SRING will result in an ERROR indication, giving the information that a connection request has not yet been received NOTE: ODM is not supported
AT#SA=?	Test command reports the range of values for all the parameters.

4.1.3.2.11 Receive Data In Command Mode - #SRECV

#SRECV - Receive Data In Command Mode	SELINT 2
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..1024 **<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	

4.1.3.2.12 Send UDP data to a specific remote host extended #SSENDUDPEXT

#SSENDUDPEXT – send UDP data to a specific remote host extended	SELINT 2
AT#SSENDUDPEXT= <connId>, <bytestosend>, <remoteIP>, <remotePort> [,<RRCFlag>]	This command permits, while the module is in command mode, to send data over UDP to a specific remote host including all possible octets(from 0x00 to 0xFF) As indicated about #SSENDUDP: UDP socket has to be previously opened through #SLUDP / #SA, then we are able to send data to different remote hosts. Like #SSENDEXT, the device responds with the prompt '> ' (<CR><LF>< greater_than><spacebar>) and waits for the data to send, operation is automatically completed when <bytestosend> have been sent. If after 2 minutes less then bytes than <bytestosend> are send by the user, an ERROR will occur and the command will be canceled. **Parameters:** **<connId>** - socket connection identifier 1..6 **<bytestosend>** - number of bytes to be sent 1-1500

	<remoteIP> - IP address of the remote host in dotted decimal notation, string type: “xxx.xxx.xxx.xxx” **<remotePort>** - remote host port 1..65535 **<RRCFlag>** - Specifies the type of message transmission. Values of these argument are formed by logically OR-ing zero or more of the following flags: 1 – Exception Messgae – Send message with high priority 2 – Release Indicator – Indicate release after next message 4 – Release Indicator – Indicate release after next message has been replied to. By default, no flags are used (value 0) Flag values 2 and 4 cannot be used together. Note: it’s possible to use **#SSENDEXT** only if the connection was opened by **#SD**, else the ME is raising an error. Note: all special characters are sent like a generic byte. (For instance: 0x08 is simply sent through the socket and don’t behave like a BS, e.g. previous character is not deleted) Note: <RRCFlag> that contain bit 0 (Exception message) could work only when sim support exception data too.
AT#SSENDUDPEXT=?	Test command reports the supported range of values for parameters **<connId>,<bytestosend>,<remoteIP> and <remotePort>**

4.1.3.2.13 Send PING request - #PING

#PING – Send PING request	**SELINT 2**
AT#PING=<IPaddr>[,<retryNum>[,<len>[,<timeout>[,<ttl>[,<pdpId>]]]]]	This command is used to send Ping Echo Request messages and to receive the corresponding Echo Reply. Ping replies are receive asynchronously, and additional PING commands are denied (with an ERROR response) until the last Echo Reply is received or timed out. **Parameters:** **<IPaddr>** - address of the remote host, string type. This parameter can be any valid IP address in the format: “xxx.xxx.xxx.xxx” **<retryNum>** - the number of Ping Echo Request to send 1-64 (default 4) **<len>** - the 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). **Dummy** parameter – not used. **<pdpId>** PDP context identifier 0..5 - numeric parameter which specifies a particular PDP context definition. **Dummy** parameter – not used. Once the single Echo Reply message is receive a string like that is displayed: #PING: <replyId>,<Ip Address>,<replyTime>,<ttl> **Where:** **<replyId>** - Echo Reply number **<Ip Address>** - IP address of the remote host **<replyTime>** - time, in 100 ms units, required to receive the response

#PING – Send PING request	SELINT 2
	<ttl> - time to live of the Echo Reply message **Note 1**: when the Echo Request timeout expires (no reply received on time) or the second and further echo request could not be sent, the response will contain <replyTime> set to 600 and <ttl> set to 255 **Note 2:** Before send PING Request the GPRS context must have been activated by AT#SGACT=0,1 **Note 3:** Only a single ping request (along with the specified retries) can be issued at a time. **NOTE: ttl is currently currently not supported.** **NOTE2: pdpId assignement is currently not supported.** **NOTE3: String addresses are not supported for IPaddr parameter (only valid IP addresses, since DNS is not supported).**
AT#PING=?	Test command reports the supported range of values for the #PING command parameters
Example	at#ping=8.8.8.8 OK #PING: 01,8.8.8.8,33,41 #PING: 02,8.8.8.8,17,41 #PING: 03,8.8.8.8,14,41 #PING: 04,8.8.8.8,10,41

4.1.3.2.14 Send data in Command Mode extended - #SSENDEXT

#SSENDEXT - Send Data In Command Mode extended	SELINT 2
AT#SSENDEXT= <connId>, <bytestosend>, [<rrcflag>]	Execution command permits, while the module is in **command mode,** to send data through a connected socket including all possible octets (from 0x00 to 0xFF). Parameters: **<connId>** - socket connection identifier 1..6 **<bytestosend>** - number of bytes to be sent Please refer to test command for range **<RRCFlag>** - Specifies the type of message transmission. Values of these argument are formed by logically OR-ing zero or more of the following flags: 1 – Exception Messgae – Send message with high priority 2 – Release Indicator – Indicate release after next message 4 – Release Indicator – Indicate release after next message has been replied to. By default, no flags are used (value 0) Flag values 2 and 4 cannot be used together. The device responds to the command with the prompt '> ' (<CR><LF><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. If after 2 minutes less then bytes than <bytestosend> are send by the user, an ERROR will occur and the command will be canceled.

#SSENDEXT - Send Data In Command Mode extended		SELINT 2
	Note: it's possible to use **#SSENDEXT** only if the connection was opened by **#SD**, else the ME is raising an error. Note: all special characters are sent like a generic byte. (For instance: 0x08 is simply sent through the socket and don't behave like a BS, e.g. previous character is not deleted) Note: <RRCFlag> that contain bit 0 (Exception message) could work only when sim support exception data too.	
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 specifying total number of bytes as second **Parameter:**
	at#ssendext=1,256
	> ..... ; // Terminal echo of bytes sent is displayed here
	OK
	All possible bytes (from 0x00 to 0xFF) are sent on the socket as generic bytes.	

5 DOCUMENT HISTORY

5.1 Revisions

Revision	Date	Changes
0	2017-04-10	First issue
1	2017-06-26	SW 29.00.010-B016 Updated command: #BND, #AUTOATT, #PING, #RFSTS, #SCFG, #SS New commands: #SCFGEXT, #SCFGEXT2, #SCFGEXT3, +IMEISV, +CCID, +CESQ, #MONI, #SERVINFO, +GSN, E, S3, S4, S5, #SI, #SS, #TEMPMON, I, #GPIO, #ADC
2	2017-08-16	SW 29.00.0x0-B024 Updated commands: +CGDCONT, #TEMPMON, +CME ERROR, +CSCON, +COPS, +CEREG, #RFSTS, #BCCHLOCK New commands: +CCLK, +CEDRXS, +CEDRXRDP, +CEER, +CGACT, #TEMPCFG, #LPWAKE
3	2017-10-16	SW 29.00.0x0-B028 Updated commands: #LPWAKE, #GPIO, #BCCHLOCK, +CEREG+MONI, +CFUN, +COPS, +CGDCONT, +CPSMS New commands: #ALTFUN, #I2CRD, #I2CWR, #SPIOPEN, #SPICLOSE, #SPIWRITE, #SPIREAD, +IPR, +CGAPNRC, +CIPCA, +CSDF
4	2017-10-31	Updated commands: ME Error Result Code - +CME ERROR, #ALTFUN, #GPIO, #SPIREAD, #SPIWRITE, #SCFGEXT, #SCFGEXT2, #SCFG, #LPWAKE, #TEMPCFG, #I2CWR, #SSENDEXT, #SSENDUDPEXT, #RFSTS, #TEMPMON, +CGDCONT New commands: #PCLASS, #PSMR, #CEDRXS, AT#UESTATS, #CCLKMODE, #NITZ


Design Review
120°
180°
Process
Validation
240°
RF/EMC Pre-Certification
Module Selection
60°
300°
Approvals
Support
360°
Value Added Services & Connectivity
Enterprise Integration &
Application Enablement
m2mair