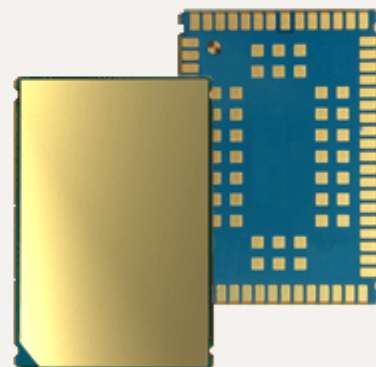


# Cinterion<sup>®</sup> ENS22-E

## AT Command Set

Version: 02.000  
DocId: ENS22-E\_ATC\_V02.000



Document Name: **Cinterion® ENS22-E AT Command Set**

Version: **02.000**

Date: **January 10, 2020**

DocId: **ENS22-E\_ATC\_V02.000**

Status **Public / Released**

**GENERAL NOTE**

THIS DOCUMENT CONTAINS INFORMATION ON GEMALTO PRODUCTS. GEMALTO M2M RESERVES THE RIGHT TO MAKE CHANGES TO THE PRODUCTS DESCRIBED HEREIN. THE SPECIFICATIONS IN THIS DOCUMENT ARE SUBJECT TO CHANGE AT THE DISCRETION OF GEMALTO M2M. THE PRODUCT AND THIS DOCUMENT ARE PROVIDED ON AN "AS IS" BASIS ONLY AND MAY CONTAIN DEFICIENCIES OR INADEQUACIES. GEMALTO M2M DOES NOT ASSUME ANY LIABILITY FOR INFORMATION PROVIDED IN THE DOCUMENT OR ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT DESCRIBED HEREIN.

GEMALTO M2M GRANTS A NON-EXCLUSIVE RIGHT TO USE THE DOCUMENT. THE RECIPIENT SHALL NOT COPY, MODIFY, DISCLOSE OR REPRODUCE THE DOCUMENT EXCEPT AS SPECIFICALLY AUTHORIZED BY GEMALTO M2M.

Copyright © 2020, Gemalto M2M GmbH, a Thales Company

**Trademark Notice**

Gemalto, the Gemalto logo, are trademarks and service marks of Gemalto and are registered in certain countries.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other registered trademarks or trademarks mentioned in this document are property of their respective owners.

## Contents

<b>1.</b>	<b>Introduction.....</b>	<b>9</b>
1.1	Scope of the document .....	9
1.2	Related documents .....	10
1.3	Document Conventions .....	12
1.3.1	Quick Reference Table .....	12
1.3.2	Superscript notation for parameters and values .....	13
1.4	AT Command Syntax .....	14
1.4.1	Using Parameters .....	14
1.5	Communication between Customer Application and ENS22-E.....	15
1.6	Supported character sets .....	16
1.6.1	GSM alphabet tables and UCS2 character values .....	18
1.6.2	UCS2 and GSM character coding and conversion .....	20
1.6.2.1	Output of SIM data (UE to TE).....	20
1.6.2.2	Input of SIM data (TE to UE) .....	21
1.7	Unsolicited Result Code Presentation.....	23
1.7.1	Common URCs.....	23
1.8	Errors and Messages .....	24
<b>2.</b>	<b>Configuration Commands.....</b>	<b>25</b>
2.1	AT&F Reset AT Command Settings to Factory Default Values .....	25
2.2	AT&V Display current configuration .....	26
2.2.1	AT&V responses.....	27
2.3	AT&W Store AT Command Settings to User Defined Profile .....	28
2.4	ATQ Result Code Presentation Mode .....	29
2.5	ATV Result code format mode .....	30
2.5.1	Verbose and numeric result codes .....	30
2.6	ATZ Restore AT Command Settings from User Defined Profile .....	31
2.7	AT+CFUN Functionality Level.....	32
2.8	AT^SMSO Switch Off ENS22-E .....	34
2.9	AT+CMEE Error Message Format .....	35
2.9.1	CME/CMS Error Code Overview .....	36
2.10	AT+CSCS Character Set .....	39
2.11	AT^SCFG Extended Configuration Settings .....	40
2.12	AT^SPOW Set UART Mode and SLEEP Mode on UART .....	46
<b>3.</b>	<b>Status Control Commands .....</b>	<b>47</b>
3.1	AT^SIND Extended Indicator Control.....	47
<b>4.</b>	<b>Serial Interface Control Commands.....</b>	<b>51</b>
4.1	ATQ Flow Control.....	51
4.2	AT&C Set Data Carrier Detect (DCD) Line Mode .....	52
4.3	AT&D Set Data Terminal Ready (DTR) Line Mode .....	53
4.4	AT&S Set Data Set Ready (DSR) Line Mode .....	54
4.5	ATE AT Command Echo .....	55
4.6	AT+ICF Character Framing.....	56
4.7	AT+IPR Bit Rate.....	58

<b>5.</b>	<b>Identification Commands</b> .....	<b>60</b>
5.1	ATI Display product identification information .....	60
5.2	AT+CGMI Request manufacturer identification.....	61
5.3	AT+CGMM Request model identification .....	62
5.4	AT+CGMR Request revision identification and software version.....	63
5.5	AT+CGSN Request International Mobile Equipment Identity (IMEI).....	64
5.6	AT+CIMI Request International Mobile Subscriber Identity (IMSI).....	66
<b>6.</b>	<b>Security Commands</b> .....	<b>67</b>
6.1	AT+CPIN PIN Authentication .....	67
6.2	AT+CLCK Facility lock .....	69
6.3	AT+CPWD Change Password .....	71
6.4	AT^SSECUC Internet Service Certificate Store Update Configuration .....	72
<b>7.</b>	<b>Network Service Commands</b> .....	<b>73</b>
7.1	AT+COPS Operator Selection .....	73
7.2	AT+CESQ Extended Signal Quality .....	76
7.3	AT+CTZR Time Zone Reporting .....	78
7.4	AT+CPSMS Enable or Disable Power Saving Mode .....	80
7.5	AT^SMONI Monitoring Serving Cell .....	82
	7.5.1 AT^SMONI Responses.....	82
	7.5.2 Service states .....	83
7.6	AT^SMONP Monitoring Neighbour Cells .....	84
	7.6.1 AT^SMONP Responses .....	84
7.7	AT+CEDRXS eDRX Setting.....	86
7.8	AT+CEDRXRDP Read dynamic eDRX parameters.....	89
7.9	AT^SNLWM2M Lwm2m Configuration Settings.....	90
7.10	AT+CIPCA Initial PDP context activation .....	93
7.11	AT+CCIOTOPT CloT Optimization Configuration .....	94
<b>8.</b>	<b>Internet Service Commands</b> .....	<b>96</b>
8.1	AT^SICS Internet Connection Setup Profile.....	100
	8.1.1 Example: GPRS connection profile .....	102
8.2	AT^SICI Internet Connection Information.....	103
	8.2.1 Checking Connection Profile Status .....	104
8.3	AT^SIPS Internet Profile Storage .....	105
8.4	AT^SISS Internet Service Setup Profile .....	107
8.5	AT^SISI Internet Service Information .....	114
8.6	AT^SISO Internet Service Open .....	116
8.7	AT^SISC Internet Service Close .....	120
8.8	AT^SISR Internet Service Read Data .....	121
	8.8.1 Example: Socket Host Reads Small Amounts of UDP Data Packets (URC Mode).....	123
8.9	AT^SISW Internet Service Write Data.....	124
	8.9.1 Usage of parameter <eodFlag> .....	126
8.10	AT^SIST Enter Transparent Mode .....	127
8.11	AT^SISD Internet Service MQTT Dynamic Setup Profile.....	129
8.12	AT^SISU MQTT Dynamic Send Request.....	132
8.13	AT^SISX Internet Service Execution.....	133
8.14	AT^SISE Internet Service Error Report.....	135
8.15	Internet Service URC "^SIS" .....	137

8.15.1	Information Elements Related to the Service Application.....	138
8.15.2	Information Elements Related to MQTT Service .....	139
8.16	Examples of how to Configure and Use Internet Service Profiles.....	141
8.16.1	Selecting URC Mode or Polling Mode .....	141
8.16.2	Non-Transparent Client Sends Data in Polling Mode .....	141
8.16.3	Socket Client Sends Data via TCP Connection with URCs.....	142
8.16.4	UDP Scenario .....	142
8.16.5	Creating Transparent TCP Socket Client .....	144
8.16.6	Opening and Closing Transparent TCP Service.....	144
8.16.7	Transparent TCP Client Receives Data While in AT Command Mode.....	145
8.16.8	Remote Server Disconnects While Transparent TCP Service is in Transparent Access Mode.....	146
8.16.9	Server Disconnects While Transparent TCP Service is in AT Command Mode and Data is Pending.....	147
8.16.10	Creating Transparent UDP Socket Client .....	148
8.16.11	Opening and Closing Transparent UDP Service .....	148
8.16.12	MQTT Scenario .....	149
8.16.13	Ping.....	154
<b>9.</b>	<b>Packet Domain Related Commands .....</b>	<b>155</b>
9.1	AT+CGACT PDP context activate or deactivate .....	155
9.2	AT+CGATT PS attach or detach.....	157
9.3	AT+CGAUTH Define PDP Context Authentication Parameters.....	158
9.4	AT+CGDCONT Define PDP Context .....	160
9.5	AT+CEREG EPS Network Registration Status .....	162
9.6	AT+CGPADDR Show PDP Address .....	165
9.7	AT+CSODCP Sending of originating data via the control plane .....	167
9.8	AT+CRTDCP Reporting of terminating data via the control plane .....	169
9.9	AT+CNMPSD No more PS data .....	171
9.10	AT+CGCONTRDP PDP context read dynamic parameters.....	172
<b>10.</b>	<b>Short Message Service (SMS) Commands.....</b>	<b>174</b>
10.1	SMS Parameters.....	174
10.2	AT+CMGC Send SMS Command.....	175
10.3	AT+CMGS Send SMS.....	176
10.4	AT+CNMA New Message Acknowledgement to UE/TE .....	177
10.5	AT+CSCA SMS Service Center Address.....	178
10.6	AT+CSMS Select Message Service.....	179
<b>11.</b>	<b>(U)SIM related Commands .....</b>	<b>181</b>
11.1	AT+CCID (U)SIM Card Identification Number.....	181
11.2	AT+CRSM Restricted SIM Access .....	182
11.3	AT+CSIM Generic (U)SIM Access .....	184
11.4	AT+CCHO Open logical channel .....	186
11.5	AT+CGLA Generic logical channel access .....	187
11.6	AT+CCHC Close logical channel .....	188
<b>12.</b>	<b>Miscellaneous Commands.....</b>	<b>189</b>
12.1	ATS3 Command Line Termination .....	189
12.2	ATS4 Response Formatting.....	190
12.3	ATS5 Command Line Editing.....	191

---

12.4	AT^SBNR Binary Read .....	192
12.5	AT^SBNW Binary Write.....	194
12.6	+++ Escape from Data Mode to AT Command Mode .....	195
12.7	AT^SNFWPUPDS Incremental Firmware Update.....	196
<b>13.</b>	<b>Hardware related Commands .....</b>	<b>198</b>
13.1	AT+CCLK Real Time Clock.....	198
13.2	AT^SBV Battery/Supply Voltage .....	199
13.3	AT^SCTM Critical Operating Temperature Monitoring.....	200
<b>14.</b>	<b>General Purpose I/O (GPIO) Pin related Commands.....</b>	<b>202</b>
14.1	AT^SPIO GPIO Driver Open/Close.....	202
14.2	AT^SCPIN Pin Configuration .....	203
14.2.1	GPIO Configuration Table .....	204
14.3	AT^SGIO Get IO State of a Specified Pin .....	205
14.4	AT^SSIO Set IO State of a Specified Pin.....	206
<b>15.</b>	<b>Appendix .....</b>	<b>207</b>
15.1	Restricted access to SIM data after SIM PIN authentication.....	207
15.2	Available AT Commands and Dependency on SIM PIN .....	208
15.3	AT Command Settings storable with AT&W.....	211
15.4	Factory Default Settings Restorable with AT&F .....	212
15.5	Summary of Unsolicited Result Codes (URC).....	213
15.6	Alphabetical List of AT Commands .....	215

---

## List of Tables

Table 1.1:	Symbols used to mark the type of parameters .....	13
Table 1.2:	Symbols used to indicate storage options or correlations with other commands .....	13
Table 1.3:	Symbols used to mark different types of default values of parameters .....	13
Table 1.4:	Types of AT commands and responses .....	14
Table 1.5:	Exemplary escape sequences generated by ENS22-E for its non-UCS2 output .....	17
Table 2.1:	General "CME ERROR" Codes (3GPP TS 27.007) .....	36
Table 2.2:	General "CME ERROR" Codes (proprietary) .....	37
Table 2.3:	EPS related "CME ERROR" Codes (3GPP TS 27.007) .....	38
Table 2.4:	SMS related "CMS ERROR" Codes (3GPP TS 27.005) .....	38
Table 8.1:	Applicability of <code>AT^SICS &lt;conParmTag&gt;</code> values .....	100
Table 8.2:	Applicability of <code>AT^SISS &lt;srvParmTag&gt;</code> values .....	107
Table 8.3:	Applicability of <code>AT^SISS &lt;srvParmTag&gt;</code> values .....	107
Table 8.4:	Applicability of <code>AT^SISD &lt;dynParamTag&gt;</code> values .....	129
Table 15.1:	Available AT Commands and Dependency on SIM PIN.....	208
Table 15.2:	Settings Stored to User Profile .....	211
Table 15.3:	Factory Default Settings Restorable with AT&F .....	212
Table 15.4:	Summary of Unsolicited Result Codes (URC).....	213
Table 15.5:	Alphabetical List of AT Commands.....	215

---

## List of Figures

Figure 1.1:	Main character table of GSM 7 bit default alphabet.....	18
Figure 1.2:	Extension character table of GSM 7 bit default alphabet.....	19



# 1. Introduction

## 1.1 Scope of the document

This document presents the AT Command Set for  
*ENS22-E LTE Engine, Release 02.000.*

Before using the ENS22-E or upgrading to a new firmware version please read the latest product information provided in "[ENS22-E Release Notes, Version 02.000](#)".

More information is available at <http://m2m.gemalto.com/>.

### DISCLAIMER:

AT commands or parameters not documented in this document are subject to change and reserved for future use. Gemalto M2M GmbH reserves the right to modify or even eliminate these options in later releases.

ENS22-E features packet switched (PS) data capability, but does not support circuit switched (CS) data transmission. However, for reasons of compatibility with other products, and for compliance with 3GPP TS specifications, some AT commands imply parameters or values related to CS data capability.

## 1.2 Related documents

---

## 1.2 Related documents

- [1] ENS22-E Release Notes, Version 02.000
- [2] ENS22-E Hardware Interface Description, Version 02.000
- [3] Application Note 16: Updating ENS22-E Firmware
- [4] Application Note 62: Transport Layer Security for Client TCP/IP Services
- [5] [3GPP TR 21.905](#) (descendant of 3GPP TR 01.04): Vocabulary for 3GPP Specifications
- [6] International Organization for Standardization (ISO): [ISO/IEC10646](#): Universal Multiple-Octet Coded Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane.  
This international standard is [closely related](#) to the [Unicode Standard](#) published by the [Unicode Consortium](#)
- [7] The [Unicode Consortium](#): [Mapping of ETSI GSM 03.38 7-bit default alphabet characters into Unicode \[TXT\]](#)
- [8] [ITU-T V.24](#) List of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE)
- [9] [ITU-T V.250](#) Serial asynchronous automatic dialling and control
- [10] [3GPP TS 11.11](#): Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
- [11] [3GPP TS 31.101](#): UICC-terminal interface; Physical and logical characteristics
- [12] [3GPP TS 31.102](#): Characteristics of the Universal Subscriber Identity Module (USIM) application
- [13] [ETSI TS 102 221](#): Smart Cards; UICC-Terminal interface; Physical and logical characteristics
- [14] [3GPP TS 11.14](#): Specification of the SIM Application Toolkit for the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface
- [15] [3GPP TS 31.111](#): Universal Subscriber Identity Module (USIM) Application Toolkit (USAT)
- [16] [ETSI TS 102 223](#): Smart Cards; Card Application Toolkit (CAT)
- [17] [3GPP TS 22.002](#) (descendant of 3GPP TS 22.02): Circuit Bearer Services (BS) supported by a Public Land Mobile Network (PLMN)
- [18] [3GPP TS 22.004](#) (descendant of 3GPP TS 02.04): General on supplementary services
- [19] [3GPP TS 22.030](#) (descendant of 3GPP TS 02.30): Man-Machine Interface (MMI) of the Mobile Station (MS)
- [20] [3GPP TS 22.060](#) (descendant of 3GPP TS 02.60): General Packet Radio Service (GPRS); Service description; Stage 1
- [21] [3GPP TS 23.060](#) (descendant of 3GPP TS 03.60): General Packet Radio Service (GPRS); Service description; Stage 2
- [22] [3GPP TS 22.081](#) (descendant of 3GPP TS 02.81): Line Identification Supplementary Services; Stage 1
- [23] [3GPP TS 22.082](#) (descendant of 3GPP TS 02.82): Call Forwarding (CF) Supplementary Services; Stage 1
- [24] [3GPP TS 22.083](#) (descendant of 3GPP TS 02.83): Call Waiting (CW) and Call Holding (HOLD); Supplementary Services; Stage 1
- [25] [3GPP TS 22.085](#) (descendant of 3GPP TS 02.85): Closed User Group (CUG) supplementary services; Stage 1
- [26] [3GPP TS 22.088](#) (descendant of 3GPP TS 02.88): Call Barring (CB) supplementary services; Stage 1
- [27] [3GPP TS 22.090](#) (descendant of 3GPP TS 02.90): Unstructured Supplementary Service Data (USSD); Stage 1
- [28] [3GPP TS 23.038](#) (descendant of 3GPP TS 03.38): Alphabets and language specific information
- [29] [3GPP TS 23.040](#) (descendant of 3GPP TS 03.40): Technical realization of the Short Message Service (SMS)
- [30] [3GPP TS 23.041](#) (descendant of 3GPP TS 03.41): Technical realization of Cell Broadcast Service (CBS)
- [31] [3GPP TS 23.107](#): Quality of Service (QoS) concept and architecture
- [32] [3GPP TS 24.011](#) (descendant of 3GPP TS 04.11): Point-to-Point (PP) Short Message Service (SMS) support on mobile radio interface
- [33] [3GPP TS 24.008](#) (descendant of 3GPP TS 04.08): Mobile radio interface Layer 3 specification; Core network protocols; Stage 3

1.2 Related documents

---

- [34] [3GPP TS 24.080](#) (descendant of 3GPP TS 04.80): Mobile radio interface layer 3 supplementary services specification; Formats and coding
- [35] [3GPP TS 24.301](#) Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS)
- [36] [3GPP TS 25.133](#) Requirements for support of radio resource management
- [37] [3GPP TS 27.005](#) (descendant of 3GPP TS 07.05): Use of Data Terminal Equipment - Data Circuit terminating Equipment (DTE - DCE) interface for Short Message Service (SMS) and Cell Broadcast Service (CBS)
- [38] [3GPP TS 27.007](#) (descendant of 3GPP TS 07.07): AT command set for User Equipment (UE)
- [39] [3GPP TS 27.060](#) (descendant of 3GPP TS 07.60): Mobile Station (MS) supporting Packet Switched Services
- [40] [3GPP TS 36.133](#) Evolved Universal Terrestrial Radio Access (E-UTRA); Requirements for support of radio resource management
- [41] [3GPP TS 36.331](#) Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification
- [42] [3GPP TS 45.008](#) (descendant of GSM 05.08): Radio subsystem link control

## 1.3 Document Conventions

Throughout this document ENS22-E is also referred to as LTE Engine or short UE, MS (Mobile Station) or Mobile Terminal (MT). In related documents the equivalent term DCE (Data Communication Equipment) may be found. AT commands are used to control the ENS22-E. The controlling device is referred to as Customer Application or short TE. Related documents may use the equivalent term DTE (Data Terminal Equipment). All abbreviations and acronyms used throughout this document are based on 3GPP specifications. For definitions please refer to 3GPP TR 21.905 [5].

### 1.3.1 Quick Reference Table

Each AT command description includes a table similar to the example shown below. The table is intended as a quick reference to indicate the following functions:

- PIN: Is the AT command PIN protected?
- + Yes
  - No
  - ± Usage is dependent on conditions specified for the command, or not all command types are PIN protected (for example write command PIN protected, read command not).
- Note: The table provided in Section 15.2, [Available AT Commands and Dependency on SIM PIN](#) uses the same symbols.
- Last: If commands are concatenated, this AT command must be the last one.
- + Yes
  - No
- Note: See also Section 1.4, [AT Command Syntax](#) for details on concatenated AT commands.

Example:

PIN	Last
-	-

## 1.3.2 Superscript notation for parameters and values

**Table 1.1:** Symbols used to mark the type of parameters

Parameter type	Meaning
<param> <sup>(num)</sup>	Parameter value must be numeric type.
<param> <sup>(str)</sup>	Parameter value must be string type.
<param> <sup>(text)</sup>	Parameter value is a string according to selected character set. Not enclosed in double quotes
<param> <sup>(u)</sup>	Unspecified, i.e. parameter value may be numeric or string type.

**Table 1.2:** Symbols used to indicate storage options or correlations with other commands

Parameter option	Meaning
<param> <sup>(+CSCS)</sup>	Parameter value has to be (is) coded according to current setting of <chset> (see <a href="#">AT+CSCS</a> for details)
<param> <sup>(&amp;W)</sup>	Parameter value is stored to user profile in non-volatile memory after executing <a href="#">AT&amp;W</a>
<param> <sup>(&amp;V)</sup>	Parameter value is displayed by <a href="#">AT&amp;V</a>
<param> <sup>(NV)</sup>	Parameter is stored in non-volatile memory.

**Table 1.3:** Symbols used to mark different types of default values of parameters

Value option	Meaning
[x]	Default value set if parameter is omitted.
x <sup>(&amp;F)</sup>	Factory value restored by <a href="#">AT&amp;F</a>
x <sup>(P)</sup>	Powerup value of a parameter not stored in non-volatile memory.
x <sup>(D)</sup>	Delivery value of a parameter which may be overridden from non-volatile setting (refer to symbol <sup>(NV)</sup> and symbol <sup>(&amp;W)</sup> above).

## 1.4 AT Command Syntax

The "AT" or "at" prefix must be set at the beginning of each command line. To terminate a command line enter <CR>. Commands are usually followed by a response that includes "<CR><LF><response><CR><LF>". Throughout this document, only the responses are presented, <CR><LF> are omitted intentionally.

The maximum length of a single AT command line is 1406 characters, including the "AT" or "at" prefix.

**Table 1.4:** Types of AT commands and responses

AT command type	Syntax	Function
Test command	AT+CXXX=?	The test response returns supported parameters and supported values. Values can be shown as a list of single values or a range, for example, (1,2,3) or (1-3).
Read command	AT+CXXX?	This command returns the currently set value of the parameter or parameters.
Write command	AT+CXXX=<...>	This command sets user-definable parameter values.
Exec(ution) command	AT+CXXX	The execution command reads non-variable parameters determined by internal processes in the UE.

### 1.4.1 Using Parameters

- Multiple parameters are separated by commas. This applies to write commands, command responses, URCs and result codes. Please note that throughout this document spaces behind commas may be added for better readability.
- Optional parameters are enclosed in square brackets. If optional parameters are omitted, the current settings are used until you change them.
- Optional parameters or subparameters can be omitted unless they are followed by other parameters. If you want to omit a parameter in the middle of a string it must be replaced by a comma.
- A parameter value enclosed in square brackets represents the value that will be used if an optional parameter is omitted.
- When the parameter is a character string the string must be enclosed in quotation marks. Symbols in quotation marks will be recognized as strings.
- All spaces will be ignored when using strings without quotation marks.
- It is possible to omit the leading zeros of strings which represent numbers.
- If an optional parameter of a ITU-T V.250 command is omitted, its value is assumed to be 0.

## 1.5 Communication between Customer Application and ENS22-E

After power-up or restart ensure that the UE is in ready state before trying to send any AT command or data. For detailed information on timing conditions, signal states and particularly the startup behavior of the ENS22-E's signal lines refer to the Hardware Interface Description [2].

Leaving hardware flow control unconsidered the Customer Application (TE) is coupled with the ENS22-E (UE) via a receive and a transmit line.

Since both lines are driven by independent devices collisions may (and will) happen. For example, if the TE issues an AT command and the ENS22-E starts sending a URC. This will probably cause the TE to misinterpret the URC being part of the AT command's response. To avoid this conflict the following measures must be taken:

- If an AT command is finished (with "OK" or "ERROR") the TE shall always wait at least 100 ms before sending the next one. This applies to bit rates of 9600 bps or higher (see [AT+IPR](#)). At bit rates below 9600 bps the delay must be longer: 300 ms at 1200 bps, and 500 ms at 300 bps. The pause between two AT commands gives the ENS22-E the opportunity to the transmission of pending URCs and get necessary service.
- The TE shall communicate with the ENS22-E using activated echo ([ATE1](#)), i.e. the ENS22-E echoes characters received from the TE. Hence, when the TE receives the echo of the first character "A" of the AT command just sent by itself it has control both over the receive and the transmit paths.

Using Backspace at command line:

- As the standard GSM alphabet does not provide a backspace functionality the ENS22-E is designed to use the character "08" (hex 0x08) as backspace for command line input. This allows the user to easily erase the last character when writing an AT command. On the other hand, this solution requires entering the escape sequence \08 for writing the "ò" character in GSM character string parameters.
- If command echo is enabled ([ATE1](#)) Backspace may cause 08 - 32 - 08 (decimal) character sequence or no echo, depending on serial interface and speed of character input.

## 1.6 Supported character sets

### 1.6 Supported character sets

ENS22-E supports two character sets: *GSM 7 bit*, also referred to as GSM alphabet or SMS alphabet (3GPP TS 23.038 [28]) and *UCS2 16 bit* (ISO-10646 [6]). See [AT+CSCS](#) for information about selecting the character set. Character tables can be found below.

#### Explanation of terms

- **Escape Character**  
There are two types of escape sequences which lead to an alternative interpretation on subsequent characters by the UE:
  - **AT command interface**  
Escape sequences starting with character value 0x5C are used for the UE's non-UCS2 input and output.
  - **GSM 7 bit default alphabet**  
If the escape sequence used within a text starts with character value 0x1B in the GSM 7 bit default alphabet, which represents the extension character and needs to be correctly interpreted by the TE, both for character input and output. To the ENS22-E, an escape sequence appears like any other byte received or sent.
- **TE Character Set**  
The character set currently used by the Customer Application is selected with [AT+CSCS](#). It is recommended to select UCS2 setting.
- **Data Coding Scheme (DCS)**  
DCS is part of a short message and is saved on the SIM.
- **International Reference Alphabet (IRA)**  
The International Reference Alphabet is equivalent to ASCII (American Standard Code for Information Interchange) and ISO 646, i.e. it defines a 7-bit coded character set. The mapping can be obtained from the character set tables below (UCS2 values 0x0000 to 0x007F).

When you enter characters that are not valid characters of the supported alphabets the behavior is undefined. If GSM alphabet is selected, all characters sent over the serial line (between TE and UE) must be in the range from 0 to 127 (7 bit range).

Note: If the UE is configured for GSM alphabet, but the Customer Application (TE) uses ASCII, bear in mind that some characters have different code values, such as the following:

- "@" character with GSM alphabet value 0 is not displayable by an ASCII terminal program, e.g. Microsoft® Hyperterminal®.
- "@" character with GSM alphabet value 0 will terminate any C string! This is because value 0 is defined as C string end tag. Therefore, the GSM Null character will cause problems on application level when using 'C'-functions, e.g. "strlen()". Using an escape sequence as shown in the table below solves the problem. By the way, this may be the reason why even network providers sometimes replace '@' with "@=" in their SIM application.
- Some other characters of the GSM alphabet may be misinterpreted by an ASCII terminal program. For example, GSM "ö" (as in "Börse") is assumed to be "|" in ASCII, thus resulting in "B|rse". This is because in both alphabets there are different characters assigned to value 7C (hexadecimal).

If the TE sends characters differently coded or undefined in ASCII or GSM (e.g. Ä, Ö, Ü) it is possible to use escape sequences. The UE's input parser translates the escape sequence to the corresponding GSM character value.

#### Note:

The UE also uses escape sequences for its non-UCS2 output: Quotation mark (") and the escape character itself (\, respectively Ö in GSM alphabet) are converted, as well as all characters with a value below 32 (hexadecimal 0x20).

Hence, the input parser of the Customer Application needs to be able to translate escape sequences back to the corresponding character of the currently used alphabet.

Unsupported characters are shown as a space (hexadecimal 0x20).



## 1.6 Supported character sets

**Table 1.5:** Exemplary escape sequences generated by ENS22-E for its non-UCS2 output

Character Value	ASCII Character	GSM Character	UCS2 Character	Escape Sequence	Numeric Escape Sequence
0x5C	\	Ö	00D6	\5C	0x5C 0x35 0x43
0x22	"	"	0022	\22	0x5C 0x32 0x32
0x00	NULL	@	n/a	\00	0x5C 0x30 0x30

Usually terminal programs are not able to recognize escape sequences, and thus, handle them as normal characters.

To prevent misinterpretation of control characters or special characters it is recommended to always use UCS2 alphabet and PDU mode.

## 1.6 Supported character sets

## 1.6.1 GSM alphabet tables and UCS2 character values

This section provides tables for the GSM default alphabet (3GPP TS 23.038 [28]) supported by the ENS22-E. Below any GSM character find the corresponding two byte character value of the UCS2 alphabet. For details refer to "ETSI GSM 03.38 mapping into Unicode" [7].

Main character table of GSM 7 bit default alphabet				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	b3	b2	b1		0	1	2	3	4	5	6	7
0	0	0	0	0	@ 0040	Δ 0394	SP 0020	0 0030	i 00A1	P 0050	¿ 00BF	p 0070
0	0	0	1	1	£ 00A3	_ 005F	! 0021	1 0031	A 0041	Q 0051	a 0061	q 0071
0	0	1	0	2	\$ 0024	Φ 03A6	" 0022	2 0032	B 0042	R 0052	b 0062	r 0072
0	0	1	1		3	¥ 00A5	Γ 0393	# 0023	3 0033	C 0043	S 0053	c 0063
0	1	0	0	4	è 00E8	Λ 039B	* 00A4	4 0034	D 0044	T 0054	d 0064	t 0074
0	1	0	1		5	é 00E9	Ω 03A9	% 0025	5 0035	E 0045	U 0055	e 0065
0	1	1	0	6	ù 00F9	Π 03A0	& 0026	6 0036	F 0046	V 0056	f 0066	v 0076
0	1	1	1		7	ì 00EC	Ψ 03A8	' 0027	7 0037	G 0047	W 0057	g 0067
1	0	0	0	8	ò 00F2 <sup>3)</sup>	Σ 03A3	( 0028	8 0038	H 0048	X 0058	h 0068	x 0078
1	0	0	1	9	ç 00E7	Θ 0398	) 0029	9 0039	I 0049	Y 0059	i 0069	y 0079
1	0	1	0	10/A	LF [LF] <sup>2)</sup>	Ξ 039E	* 002A	: 003A	J 004A	Z 005A	j 006A	z 007A
1	0	1	1	11/B	∅ 00D8	<sup>1)</sup>	+ 002B	; 003B	K 004B	Ä 00C4	k 006B	ä 00E4
1	1	0	0	12/C	ø 00F8	Æ 00C6	, 002C	< 003C	L 004C	Ö 00D6	l 006C	ö 00F6
1	1	0	1	13/D	CR [CR] <sup>2)</sup>	æ 00E6	- 002D	= 003D	M 004D	Ñ 00D1	m 006D	ñ 00F1
1	1	1	0	14/E	À 00C5	ß 00DF	. 002E	> 003E	N 004E	Ü 00DC	n 006E	ü 00FC
1	1	1	1	15/F	á 00E5	É 00C9	/ 002F	? 003F	O 004F	Ş 00A7	o 006F	à 00E0

Figure 1.1: Main character table of GSM 7 bit default alphabet

- 1) This code is an escape to the following extension of the 7 bit default alphabet table.
- 2) This code is not a printable character and therefore not defined for the UCS2 alphabet. It shall be treated as the accompanying control character.
- 3) See Section 1.5 for further details on using backspace and "ò" character.

Extension character table of GSM 7 bit default alphabet				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	b3	b2	b1		0	1	2	3	4	5	6	7
0	0	0	0	0								
									007C			
0	0	0	1	1								
0	0	1	0	2								
0	0	1	1	3								
0	1	0	0	4		^						
						005E						
0	1	0	1	5						€ <sup>2)</sup>		
										20AC		
0	1	1	0	6								
0	1	1	1	7								
1	0	0	0	8			{					
							007B					
1	0	0	1	9			}					
							007D					
1	0	1	0	10 /A	<sup>3)</sup>	[LF]						
1	0	1	1	11 /B								
1	1	0	0	12 /C				[				
								005B				
1	1	0	1	13 /D				~				
								007E				
1	1	1	0	14 /E				]				
								005D				
1	1	1	1	15 /F			\					
							005C					

Figure 1.2: Extension character table of GSM 7 bit default alphabet

- 1) This code value is reserved for the extension to another extension table. On receipt of this code, a receiving entity shall display a space until another extension table is defined.
- 2) This code represents the EURO currency symbol. The code value is the one used for the character 'e'. Therefore a receiving entity which is incapable of displaying the EURO currency symbol will display the character 'e' instead.
- 3) This code is defined as a Page Break character and may be used for example in compressed CBS messages. Any mobile which does not understand the 7 bit default alphabet table extension mechanism will treat this character as Line Feed.

## 1.6 Supported character sets

If the Customer Application receives a code where a symbol is not represented in Figure 1.2, [Extension character table of GSM 7 bit default alphabet](#) it shall display the character shown in the main GSM 7 bit default alphabet table (see Figure 1.1, [Main character table of GSM 7 bit default alphabet](#)).

### 1.6.2 UCS2 and GSM character coding and conversion

This section provides basic information on how to handle input and output character conversion, e.g. for SMS text mode, if the character representation of UE and Customer Application differ, i.e. if the Data Coding Scheme and the TE character set use different mappings.

#### 1.6.2.1 Output of SIM data (UE to TE)

Used character set	DCS = 7 bit GSM	DCS = 8 bit Data	DCS = 16 bit UCS2
GSM	Case 1 GSM (1:1)	Case 2 8 bit to IRA (1:2)	Case 3 UCS2 to IRA (2:4)
UCS2	Case 4 GSM to IRA (1:4)	Case 5 8 bit to IRA (1:4)	Case 6 UCS2 to IRA (2:4)

Note: The ratio of SIM bytes to output bytes is given in parentheses.

##### Case 1

Every GSM character is sent to the TE as it is (8-bit value with highest bit set to zero).

Example: 47'H, 53'H, 4D'H → 47'H, 53'H, 4D'H, displayed as "GSM"

##### Case 2

Every data byte is sent to the TE as 2 IRA characters each representing a halfbyte.

Example: B8'H (184 decimal) → 42'H, 38'H, displayed as "B8"

##### Case 3

Every 16-bit UCS2 value is sent to the TE as 4 IRA characters.

Example: C4xA7'H (50343 decimal) → 43'H, 34'H, 41'H, 37'H, displayed as "C4A7"

Problem: An odd number of bytes leads to an error because there are always two bytes needed for each UCS2 character

##### Case 4

Every GSM character is sent to the TE as 4 IRA characters to show UCS2 in text mode.

Example: 41'H ("A") → 30'H, 30'H, 34'H, 31'H, displayed as "0041"

##### Case 5

Every data byte is sent to the TE as IRA representation of UCS2 (similar to case 4).

Example: B2'H → 30'H, 30'H, 42'H, 32'H, displayed as "00B2"

##### Case 6

Every 16-bit value is sent to the TE as IRA representation of it. It is assumed that number of bytes is even.

Example: C3x46'H → 43'H, 33'H, 34'H, 36'H, displayed as "C346"

## 1.6 Supported character sets

## 1.6.2.2 Input of SIM data (TE to UE)

Used character set	DCS = 7 bit GSM	DCS = 8 bit Data	DCS = 16 bit UCS2
GSM	Case 1 GSM (1:1)	Case 2 IRA to 8 bit (2:1)	Case 3 IRA to 16 bit (4:2)
UCS2	Case 4 UCS2 to GSM (4:1)	Case 5 UCS2 to 8 bit (4:1)	Case 6 UCS2 to 16 bit (4:2)

Note: The ratio between the number of input characters and bytes stored on the SIM is given in parentheses.

## Case 1

Every character is sent from TE to UE as GSM character (or ASCII with standard terminal emulation, e.g. Hyperterminal®).

Character value must be in range from 0 to 127 because of 7-bit GSM alphabet.

To reach maximum SMS text length of 160 characters in 140 bytes space characters will be compressed on SIM.

Example: "ABCDEFGH" typed is sent and stored uncompressed as → 4142434445464748'H (stored compressed as 41E19058341E91'H)

## Case 2

Every data byte is sent as 2 IRA characters.

Maximum text length is 280 IRA characters which will be converted into 140 bytes SMS binary user data

Example: "C8" typed is sent as 43'H, 38'H → stored as C8'H

## Case 3

Every 16-bit value is sent as 4 IRA characters.

Maximum text length is 280 IRA characters which will be converted into 70 UCS2 characters (16-bit each)

Number of IRA characters must be a multiple of four because always 4 half bytes are needed for a 16-bit value

Example: "D2C8" typed is sent as 44'H, 32'H, 43'H, 38'H → stored as D2C8'H

## Case 4

Every GSM character is sent as 4 IRA characters representing one UCS2 character.

Example: To store text "ABC" using UCS2 character set you have to type "004100420043".

This is sent as 30'H,30'H,34'H,31'H, 30'H,30'H,34'H,32'H, 30'H,30'H,34'H,33'H → detected as IRA representation of 3 UCS2 characters, converted to GSM character set and stored as 41'H, 42'H, 43'H.

Maximum input is 640 IRA characters representing 160 UCS2 characters when compression is active. These are converted to 160 GSM 7-bit characters.

Without compression only 140 GSM characters can be stored which are put in as 560 IRA characters.

Values of UCS2 characters must be smaller than 80'H (128 decimal) to be valid GSM characters.

Number of IRA characters must be a multiple of four. Problems:

- "41" → Error, there are four IRA characters (two bytes) needed
- "0000" → Error, not an UCS2 character
- "4142" → Error, value of UCS2 character > 7F'H
- "008B" → Error, value of UCS2 character > 7F'H

This affects the maximum input length of a string

## Case 5

Every UCS2 character is sent as 4 IRA characters and is converted into two 8-bit values. This means that the first two characters have to be '00'.

Example: UCS2 character 009F'H typed as "009F" is sent as 30'H,30'H,39'H,46'H → converted into 8-bit value 9F'H.

Maximum number of UCS2 characters is 140 which are represented by 560 IRA characters. Number of IRA characters must be a multiple of four.

## Case 6

Every UCS2 character is sent as 4 IRA characters each and is converted into a 16-bit value again.

Example: UCS2 character 9F3A'H typed as "9F3A" is sent as 39'H,46'H,33'H,41'H → converted into 9F3A'H.

Maximum number of UCS2 characters is 70 which are represented by 280 IRA characters. Number of IRA characters must be a multiple of four.

Invalid UCS2 values must be prevented.

## 1.6 Supported character sets

---

### Case 7

In the range from 0 to 32 in character table of GSM 7 bit default alphabet (Below hexadecimal 0x20), "@" and "\$" are the only two which are supported by normal escape.

The other characters will be displayed by "\" and two IRA characters.

For example, a UCS2 character "00A3" escape as a GSM character "\01", while a UCS2 character "00A5" escape as a GSM character "\03"

Invalid UCS2 values must be prevented.

### Case 8

The first two characters have to be '00' because 4 IRA characters are converted into two 8-bit values. If this is not the case, conversion is not supported.

Number of IRA characters must be a multiple of four. Problems:

- "03A6" → Error, not support
- "039E" → Error, not support

Invalid UCS2 values must be prevented.

### Case 9

Quotation mark (") and the escape character itself (\, respectively Ö in GSM alphabet) regard as illegal character.

Problems:

- "\" → Error, invalid characters in text string
- "" → Error, invalid characters in text string

The character "\" in front of the escape character itself indicates that it needs to be transferred. That means "\\\" can be transformed into an UCS2 character "00D6".

The character (") can be represented by "\22" as a GSM character. That means "\22" can be transformed into an UCS2 character "0022".

Invalid UCS2 values must be prevented.

### Case 10

Extension character table of GSM 7 bit default alphabet starting with "1B" can be correctly interpreted. Extension character table of UCS2 value can be displayed by escape sequences.

Example:

The character "^" with GSM alphabet value "1B14" can be transformed into an UCS2 character "005E".

But "005E" with UCS2 value will transformed into a GSM value "\1B\14".

## 1.7 Unsolicited Result Code Presentation

URC stands for Unsolicited Result Code and is a report message issued by the UE without being requested by the TE, i.e. an URC is issued automatically when a certain event occurs. Hence, a URC is not issued as part of the response related to an executed AT command.

Typical events leading to URCs are received short messages, changes in temperature, network registration etc. For most of these messages, the UE needs to be configured whether or not to send a URC. Descriptions of these URCs are provided with the associated AT command. A summary of all URCs can be found in Section 15.5, [Summary of Unsolicited Result Codes \(URC\)](#).

*Important:* If the interface used for URC output is reserved by an active data connection or a long running AT command, URCs are buffered internally and will be issued after the interface becomes idle again. A pending URC will be signaled on the URC output interface via RING line. This allows to systematically suspend any longer data connection (refer [+++](#)) to check for pending URCs after being signaled!

For detailed information regarding configuration of URC signaling refer to [AT^SCFG](#), "URC/Ringline", [<urcRinglineCfg>](#).

### 1.7.1 Common URCs

This section contains all URCs not associated to a certain AT command. They cannot be defined by the user and appear automatically when the voltage conditions described below occur.

*Notes for "^SBC" URCs reporting undervoltage/overvoltage conditions:*

Please refer to [\[2\]](#) for specifications regarding the minimum and maximum operating voltage limits. If the supply voltage exceeds the defined thresholds the UE switches off automatically. The automatic shutdown procedure is equivalent to the power-down initiated with the [AT^SMSO](#) command.

URC 1

`^SYSSTART`

Indicates that the UE has been restarted to Normal mode and is ready to operate.

URC 2

`^SBC: Undervoltage Warning`

Supply voltage is close to the defined undervoltage threshold.

URC 3

`^SBC: Undervoltage Shutdown`

Undervoltage threshold exceeded. UE automatically switches off.

URC 4

`^SBC: Overvoltage Warning`

Supply voltage is close to defined overvoltage threshold.

URC 5

`^SBC: Overvoltage Shutdown`

Overvoltage threshold exceeded. UE automatically switches off.

---

## 1.8 Errors and Messages

The command result codes "+CME ERROR: <err>" and "+CMS ERROR: <err>" indicate errors related to mobile equipment or network functionality.

The format of <err> can be either numeric or verbose and is selectable via [AT+CMEE](#).

A result error code terminates the execution of the command and prevents the execution of all remaining commands that may follow on the same command line. If so, neither "ERROR" nor "OK" result codes are returned for these commands. A 30 seconds timeout will deliver "ERROR" when the input of a command is not complete.

Using the wrong command syntax may result in errors: For example, using the execute command syntax although the command has no execute format, causes "ERROR" to be returned. Likewise, using the write command syntax although the command has no write format causes "+CME ERROR: <err>" to be returned.

See also:

- Section [2.9.1](#), [CME/CMS Error Code Overview](#)
- Section [2.5.1](#), [Verbose and numeric result codes](#)



## 2. Configuration Commands

The AT Commands described in this chapter allow the external application to determine the ENS22-E's behaviour under various conditions.

### 2.1 AT&F Reset AT Command Settings to Factory Default Values

[AT&F](#) resets AT command settings to their factory default values.

For a list of affected parameters refer to Section [15.4, Factory Default Settings Restorable with AT&F](#).

#### Syntax

Exec Command	
AT&F[<value>]	
Response(s)	
OK	
PIN Last	Reference(s)
- -	V.250

#### Parameter Description

<value> <sup>(num)</sup>	
[0]	Reset parameters in Section <a href="#">15.4, Factory Default Settings Restorable with AT&amp;F</a> to their factory default values.

## 2.2 AT&V Display current configuration

[AT&V](#) returns the current parameter setting. The configuration varies depending on whether or not PIN authentication has been done.

### Syntax

Exec Command	
AT&V[<value>]	
Response(s)	
ACTIVE PROFILE: ... (see Section 2.2.1, <a href="#">AT&amp;V responses</a> ) OK	
PIN Last	Reference(s)
- -	V.250

### Parameter Description

<value> <sup>(num)</sup>	
[0]	Profile number

## 2.2.1 AT&V responses

The following table shows different kinds of responses depending on whether or not the PIN is entered.

PIN authentication done	No PIN authentication
ACTIVE PROFILE: E1 Q0 V1 &C1 &D2 &S0 \Q3 S3:013 S4:010 S5:008 +CSCS: GSM +CMEE: 2 +ICF: 3 +CSMS: 0,1,1,1 +COPS: 0,0,"operator" +CEREG: 0,1 OK	ACTIVE PROFILE: E1 Q0 V1 &C1 &D2 &S0 \Q3 S3:013 S4:010 S5:008 +CSCS: GSM +CMEE: 2 +ICF: 3 OK

## 2.3 AT&W Store AT Command Settings to User Defined Profile

[AT&W](#) stores the current AT command settings to a user defined profile in non-volatile memory of ENS22-E. The AT command settings will automatically be restored from the user defined profile during power-up or if [ATZ](#) is used. [AT&F](#) restores AT command factory default settings. Hence, until first use of [AT&W](#), [ATZ](#) works as [AT&F](#). A list of parameters stored to the user profile can be found at Section [15.3, AT Command Settings storable with AT&W](#).

### Syntax

Exec Command	
AT&W[<value>]	
Response(s)	
OK	
ERROR	
+CME ERROR: <err>	
PIN Last	Reference(s)
- -	V.250

### Parameter Description

<value> <sup>(num)</sup>	
[0]	User Profile Number

## 2.4 ATQ Result Code Presentation Mode

**ATQ** controls if the ENS22-E transmits any result code to the TE. Other information text transmitted as response is not affected.

### Syntax

Exec Command	
ATQ[<n>]	
Response(s)	
If <n>=0: OK	
If <n>=1: (none)	
PIN Last	Reference(s)
- -	V.250

### Parameter Description

<n> <sup>(num)(&amp;V)(&amp;W)</sup>	
Result Code Presentation Mode. It is not recommended to change this value.	
[0] <sup>(&amp;F)(D)</sup>	UE transmits result code.
1	Result codes are suppressed and not transmitted.

## 2.5 ATV Result code format mode

This command determines the contents of header and trailer transmitted with AT command result codes and information responses. Possible responses are described in Section 2.5.1, [Verbose and numeric result codes](#).

### Syntax

Exec Command	
ATV[<value>]	
Response(s)	
OK ERROR	
PIN Last	Reference(s)
- -	ITU-T V.250 <a href="#">[9]</a>

### Parameter Description

<value> <sup>(num)(&amp;V)(&amp;W)</sup>	
[0]	Information response: <text><CR><LF> Short result code format: <numeric code><CR>
1 <sup>(&amp;F)(D)</sup>	Information response: <CR><LF><text><CR><LF> Long result code format: <CR><LF><verbose code><CR><LF>

### 2.5.1 Verbose and numeric result codes

Verbose format	Numeric format	Meaning
OK	0	command executed, no errors
CONNECT	1	link established
ERROR	4	invalid command or command line too long

## 2.6 ATZ Restore AT Command Settings from User Defined Profile

First [ATZ](#) resets the AT command settings to their factory default values, similar to [AT&F](#). Afterwards the AT command settings are restored from a user defined profile in non-volatile memory of ENS22-E, if one was stored with [AT&W](#) before. Any additional AT command on the same command line may be ignored. A delay of 300 ms is required before next AT command is sent.

### Syntax

Exec Command	
ATZ[<value>]	
Response(s)	
OK	
PIN Last	Reference(s)
- -	V.250

### Parameter Description

<value> <sup>(num)</sup>	
[0]	User Profile Number

## 2.7 AT+CFUN Functionality Level

**AT+CFUN** controls ENS22-E's functionality levels. It can also be used to reset the UE.

### Syntax

<p>Test Command</p> <p>AT+CFUN=?</p> <p>Response(s)</p> <p>+CFUN: (list of supported &lt;fun&gt;s), (list of supported &lt;rst&gt;s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Read Command</p> <p>AT+CFUN?</p> <p>Response(s)</p> <p>+CFUN: &lt;fun&gt;</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Write Command</p> <p>AT+CFUN=&lt;fun&gt;[, &lt;rst&gt;]</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p> <p>If &lt;fun&gt;= 0:</p> <p>OK</p> <p>If &lt;rst&gt;= 1:</p> <p>OK</p> <p>^SHUTDOWN</p> <p>^SYSSTART</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Parameter Description

<fun> <sup>(num)</sup>	
0	<p>Minimum functionality.</p> <p>Shuts down TX and RX RF-circuits whereby ENS22-E logs off from the network. Access to the (U)SIM is also disabled. AT commands whose execution requires a radio connection or (U)SIM access either return an error result code (e.g. "+CME ERROR: 524") or reflect the limited operating state. See Section 2.9.1, <a href="#">CME/CMS Error Code Overview</a>.</p> <p>To return to normal mode the command <b>AT+CFUN=1</b> can be used. After this, <b>AT+COPS=0</b> is necessary for registering to a network.</p>
1 <sup>(D)</sup>	<p>Full functionality level.</p>



---

`<rst>`<sup>(num)</sup>

Parameter `<rst>` applies only to `<fun>=1`.

- |     |  |
|-----|--|
| [0] | UE switches to <code>&lt;fun&gt;</code> level without reset.   |
| 1   | UE resets and restarts to full functionality level. " <code>^SHUTDOWN</code> " URC indicates that the UE has completed the reset procedure and will restart. " <code>^SYSSTART</code> " URC indicates that the UE has restarted and is ready to operate. |

## 2.8 AT^SMSO Switch Off ENS22-E

[AT^SMSO](#) initiates ENS22-E's power-off procedure. Do not send any other AT command after this.

Low level of the ME's V300 signal and the URC "[^SHUTDOWN](#)" notify that the procedure has completed and the ME has entered the POWER DOWN mode. Therefore, be sure not to disconnect the operating voltage until the V300 signal has gone low and until the URC "[^SHUTDOWN](#)" is displayed. Otherwise, you run the risk of losing data. For further details on how to turn off the ME refer to [\[2\]](#).

### Syntax

Test Command
AT^SMSO=?
Response(s)
OK
Exec Command
AT^SMSO
Response(s)
^SMSO: MS OFF
OK
ERROR
PIN Last
- +

### Unsolicited Result Code

[^SHUTDOWN](#)

Indicates that the power-off procedure is finished and the module will be switched off in less than 1 second.

## 2.9 AT+CMEE Error Message Format

**AT+CMEE** controls the format of error result codes that indicates errors related to ENS22-E functionality. Format can be selected between plain "ERROR" output, error numbers or verbose "+CME ERROR: <err>" and "+CMS ERROR: <err>" messages.

Possible error result codes are listed in Table 2.1, [General "CME ERROR" Codes \(3GPP TS 27.007\)](#), Table 2.3, [EPS related "CME ERROR" Codes \(3GPP TS 27.007\)](#) and Table 2.4, [SMS related "CMS ERROR" Codes \(3GPP TS 27.005\)](#).

### Syntax

<p>Test Command</p> <p>AT+CMEE=?</p> <p>Response(s)</p> <p>+CMEE: (list of supported&lt;errMode&gt;s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+CMEE?</p> <p>Response(s)</p> <p>+CMEE: &lt;errMode&gt;</p> <p>OK</p>	
<p>Write Command</p> <p>AT+CMEE=&lt;errMode&gt;</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38], 3GPP TS 27.005 [37]</p>

### Parameter Description

<errMode> <sup>(num)(&amp;V)(&amp;W)</sup>	
0(&F)(D)	Disable result code, i.e. only "ERROR" will be displayed.
1	Enable error result code with numeric values.
2	Enable error result code with verbose (string) values.

### Example

To obtain enhanced error messages it is recommended to choose <errMode>=2.

```
AT+CMEE=2
OK
```

## 2.9.1 CME/CMS Error Code Overview

**Table 2.1:** General "CME ERROR" Codes (3GPP TS 27.007)

<err> Code	Text (if AT+CMEE=2)
0	phone failure
1	no connection to phone
2	phone adapter link reserved
3	operation not allowed
4	operation not supported
5	PH-SIM PIN required
6	PH-FSIM PIN required
7	PH-FSIM PUK required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	memory full
21	invalid index
22	not found
23	memory failure
24	text string too long
25	invalid characters in text string
26	dial string too long
27	invalid characters in dial string
30	no network service
31	network timeout
32	network not allowed - emergency calls only
40	network personalization PIN required
41	network personalization PUK required
42	network subset personalization PIN required
43	network subset personalization PUK required
44	service provider personalization PIN required
45	service provider personalization PUK required
46	corporate personalization PIN required
47	corporate personalization PUK required
48	hidden key required
49	EAP method not supported
50	incorrect parameters

<err> Code	Text (if AT+CMEE=2)
51	command implemented but currently disabled
52	command aborted by user
53	not attached to network due to MT functionality restrictions
55	operation not allowed because of MT functionality restrictions
57	temporarily out of service due to other MT usage
58	language/alphabet not supported
59	unexpected data value
60	system failure
61	data missing
63	message waiting indication subscr
100	unknown

**Table 2.2:** General "CME ERROR" Codes (proprietary)

<err> Code	Text (if AT+CMEE=2)
150	invalid mobile class
256	operation temporary not allowed
259	user abort
260	data Uplink busy
261	SS not executed
262	SIM blocked
263	invalid block
271	write operation temporary not allowed
272	parameter out of range
273	service profile ID out of range
274	connection profile ID out of range
275	no service
276	service already in use
277	service not in UP status
278	error in address
279	operation not allowed
286	server already existed
287	PDP activation in progress
288	PDP configuration was modified
289	PDP activation failure
290	PDP no DNS received
291	DNS invalid server
292	DNS undefined error
294	operation timeout
512	Required Parameter Not Configured
513	TUP Not Registered
529	FOTA is updating
514	AT Cmd Internal Error

<err> Code	Text (if AT+CMEE=2)
515	cid is active
516	radio test state error
517	cid is invalid
518	cid is not active
519	link security error
520	Last PDN disconnection not allowed
521	cid is not defined
522	AT UART parity error
523	AT UART frame error
524	UE not power on
525	sent data sequence repeat error
526	AT command abort error
527	command interrupted
528	configuration conflicts
530	not the AT allocated socket
531	SIM PIN is blocked
532	SIM PUK is blocked
533	not mipi module
534	file not found
535	conditions of use not satisfied
536	at uart buffer error
537	back off timer is running

**Table 2.3:** EPS related "CME ERROR" Codes (3GPP TS 27.007)

<err> Code	Text (if AT+CMEE=2)
103	illegal MS (#3)
106	illegal ME (#6)
107	EPS services not allowed (#)
111	PLMN not allowed (11)
113	roaming not allowed in this tracking area (#13)
132	service option not supported
133	requested service option not subscribed
134	service option temporary out of order
181	unsupported QCI value

**Table 2.4:** SMS related "CMS ERROR" Codes (3GPP TS 27.005)

<err> Code	Text (if AT+CMEE=2)
302	operation not allowed
303	operation not supported
320	memory failure
340	no +CNMA acknowledgement expected

## 2.10 AT+CSCS Character Set

**AT+CSCS** write command informs the ENS22-E which character set is used by the TE. This enables the UE to convert character strings correctly between TE and UE character sets. Please also refer to Section 1.6, [Supported character sets](#).

Note: If UE-TE interface is set to 8-bit operation ([AT+ICF](#)) and selected character set is `<chset>="GSM"` (7-bit), the highest bit will be set to zero.

### Syntax

Test Command

```
AT+CSCS=?
```

Response(s)

```
+CSCS: (list of supported<chset>s)
```

```
OK
```

Read Command

```
AT+CSCS?
```

Response(s)

```
+CSCS: <chset>
```

```
OK
```

Write Command

```
AT+CSCS=<chset>
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

Reference(s)

3GPP TS 27.007 [\[38\]](#)

### Parameter Description

`<chset>`<sup>(str)(&V)(&W)</sup>

"GSM"<sup>(&F)(D)</sup>

GSM 7 bit default alphabet (3GPP TS 23.038 [\[28\]](#), Subclause 6.2.1).

"UCS2"

16-bit universal multiple-octet coded character set (ISO-10646 [\[6\]](#)). UCS2 character strings are converted to hexadecimal numbers in the range 0000 to FFFF; e.g. "004100620063" equates to three 16-bit characters with decimal values 65, 98 and 99.

## 2.11 AT^SCFG Extended Configuration Settings

AT^SCFG can be used to query and configure various settings of the ENS22-E.

AT^SCFG read command returns a list of all supported parameters and their current values.

AT^SCFG write command queries a configuration parameter (if no value is entered) or sets its value(s).

Input of parameter names is always coded in GSM character set, parameter values are expected to be given as specified via AT+CSCS.

### Syntax

#### Test Command

AT^SCFG=?

Response(s)

```
^SCFG: "GPIO/mode/ASC1", (list of supported <g_mode>s)
^SCFG: "GPIO/mode/DCD0", (list of supported <g_mode>s)
^SCFG: "GPIO/mode/DSR0", (list of supported <g_mode>s)
^SCFG: "GPIO/mode/DTR0", (list of supported <g_mode>s)
^SCFG: "GPIO/mode/RING0", (list of supported <g_mode>s)
^SCFG: "MEopMode/SRPOM", (list of supported <SrpomMode>s)
^SCFG: "MEopMode/PowerMgmt/Suspend", (list of supported <Suspend>s)
^SCFG: "MESHUTDOWN/sVsup/threshold", (list of supported <vthresh>s)
^SCFG: "Radio/Band/4G", (list of supported <lte_mask>s)
^SCFG: "Tcp/WithURCs", (list of supported <tcpWithUrc>)
^SCFG: "URC/Ringline", (list of supported <urcRinglineCfg>s)
^SCFG: "URC/Ringline/ActiveTime", (list of supported <urcRinglineDuration>s)
OK
```

#### Read Command

AT^SCFG?

Response(s)

```
^SCFG: "GPIO/mode/ASC1", <g_mode>
^SCFG: "GPIO/mode/DCD0", <g_mode>
^SCFG: "GPIO/mode/DSR0", <g_mode>
^SCFG: "GPIO/mode/DTR0", <g_mode>
^SCFG: "GPIO/mode/RING0", <g_mode>
^SCFG: "MEopMode/SRPOM", <SrpomMode>
^SCFG: "MEopMode/PowerMgmt/Suspend", <Suspend>
^SCFG: "MESHUTDOWN/sVsup/threshold", <vthresh_curr>, <vthresh_request>
^SCFG: "Radio/Band/4G", <lte_mask>
^SCFG: "Tcp/WithURCs", <tcpWithUrc>
^SCFG: "URC/Ringline", <urcRinglineCfg>
^SCFG: "URC/Ringline/ActiveTime", <urcRinglineDuration>
OK
```

#### Write Command

Configure ASC1 interface lines RXD1, TXD1, RTS1, CTS1 shared with GPIO16 - GPIO19 lines

AT^SCFG="GPIO/mode/ASC1"[, <g\_mode>]

Response(s)

```
^SCFG: "GPIO/mode/ASC1", <g_mode>
OK
ERROR
+CME ERROR: <err>
```



Write Command

Configure DCD0 line of ASC0 interface shared with GPIO2

```
AT^SCFG="GPIO/mode/DCD0", <g_mode>]
```

Response(s)

```
^SCFG: "GPIO/mode/DCD0", <g_mode>
OK
ERROR
+CME ERROR: <err>
```

Write Command

Configure DSR0 line of ASC0 interface shared with GPIO3 line

```
AT^SCFG="GPIO/mode/DSR0", <g_mode>]
```

Response(s)

```
^SCFG: "GPIO/mode/DSR0", <g_mode>
OK
ERROR
+CME ERROR: <err>
```

Write Command

Configure DTR0 line of ASC0 interface shared with GPIO1 line

```
AT^SCFG="GPIO/mode/DTR0", <g_mode>]
```

Response(s)

```
^SCFG: "GPIO/mode/DTR0", <g_mode>
OK
ERROR
+CME ERROR: <err>
```

Write Command

Configure RING0 line of ASC0 interface shared with GPIO24 line

```
AT^SCFG="GPIO/mode/RING0", <g_mode>]
```

Response(s)

```
^SCFG: "GPIO/mode/RING0", <g_mode>
OK
ERROR
+CME ERROR: <err>
```

Write Command

Enable or disable RPM if inserted (U)SIM contains no RPM files.

```
AT^SCFG="MEopMode/SRPOM", <SrpomMode>]
```

Response(s)

```
^SCFG: "MEopMode/SRPOM", <SrpomMode>
OK
ERROR
+CME ERROR: <err>
```

Write Command

Enable/disable the Suspend mode.

```
AT^SCFG="MEopMode/PowerMgmt/Suspend", <Suspend>]
```

Response(s)

```
^SCFG: "MEopMode/PowerMgmt/Suspend", <Suspend>
OK
ERROR
+CME ERROR: <err>
```

Write Command

Query / configure undervoltage threshold

```
AT^SCFG="MESHUTDOWN/sVsup/threshold"[, <vthresh>]
```

Response(s)

```
^SCFG: "MESHUTDOWN/sVsup/threshold", <vthresh_curr>, <vthresh_request>
```

OK

ERROR

+CME ERROR: <err>

Write Command

Radio band selection

```
AT^SCFG="Radio/Band/4G"[, <lte_mask>]
```

Response(s)

```
^SCFG: "Radio/Band/4G", <lte_mask>
```

OK

ERROR

+CME ERROR: <err>

Write Command

Configuration of Internet Service URCS:

```
AT^SCFG="Tcp/WithURCs"[, <tcpWithUrc>]
```

Response(s)

```
^SCFG: "Tcp/WithURCs", <tcpWithUrc>
```

OK

ERROR

+CME ERROR: <err>

Write Command

Subcommand controls the behaviour of the RING line if it is selected for URC signalization.

```
AT^SCFG="URC/Ringline"[, <urcRinglineCfg>]
```

Response(s)

```
^SCFG: "URC/Ringline", <urcRinglineCfg>
```

OK

ERROR

+CME ERROR: <err>

Write Command

Subcommand controls duration of active RING line if it is selected for URC signalization.

```
AT^SCFG="URC/Ringline/ActiveTime"[, <urcRinglineDuration>]
```

Response(s)

```
^SCFG: "URC/Ringline/ActiveTime", <urcRinglineDuration>
```

OK

ERROR

+CME ERROR: <err>

PIN Last

- -

### Unsolicited Result Code

```
^SYSRESUME, <reason>
```

The URC is issued when the module is waked up from the Suspend mode.

### Parameter Description

`<g_mode>`<sup>(str)(+CSCS)(NV)</sup>

Mode of operation for signal lines

This parameter can be used to configure shared signal lines of the ENS22-E module. Depending on the connected devices the setting determines the specific function assigned to the related signal line(s). Keep in mind that the functions assigned to shared lines are mutually exclusive. When a line is reserved for one interface, then the `<g_mode>` parameter is automatically set to 'rsv' for all other interfaces that use this line.

To enable, for example, the second serial interface ASC1 select `AT^SCFG="GPIO/mode/ASC1","std"`.

The changes will take effect after restart of the UE and are non-volatile.

"std"	Signal lines are assigned to this interface.
"gpio"	Signal lines are used as GPIO.
"rsv"	Signal lines are reserved by other interface. This value cannot be set by the <code>AT^SCFG</code> write command.

`<SrpmMode>`<sup>(str)(+CSCS)(NV)</sup>

RPM Mode

This parameter enables or disables the usage of the Radio Policy Manager (RPM) according to "GSMA's IoT Device Connection Efficiency Guidelines", Version 4.0, Chapter 8. It is possible to enable or disable RPM by using the `<SrpmMode>` parameter. Note that RPM setting will be ignored on the inserted (U)SIM.

"0" <sup>(D)</sup>	RPM disabled
"1"	RPM enabled

`<vthresh>`<sup>(str)(NV)</sup>

Undervoltage threshold value

The undervoltage thresholds defined for the BB (base band) domain are calculated for max. 400 mV voltage drops. Power supply sources for ENS22-E applications shall be designed to tolerate 400 mV voltage drops without crossing the lower limits of 3.1 V. For ENS22-E applications operating at the limit of the allowed tolerance the default undervoltage warning threshold may be adapted by selecting a `<vthresh>` value.

New settings take effect after restart.

"-4"	2.90 V
"-3"	2.95 V
"-2"	3.00 V
"-1"	3.05 V
"0" <sup>(D)</sup>	3.10 V
"1"	3.15 V
"2"	3.20 V
"3"	3.25 V
"4"	3.30 V

`<vthresh_curr>`<sup>(str)</sup>

Current BB undervoltage threshold

For values see `<vthresh>`.

`<vthresh_request>`<sup>(str)</sup>

BB undervoltage threshold after next restart

For values see `<vthresh>`.

`<Suspend>`<sup>(str)(NV)</sup>

Enable Suspend mode

Enable or disable Suspend mode.

“0”<sup>(D)</sup> Suspend is disabled.

“1” Suspend is enabled.

For the module to enter Suspend mode, URCs are triggered by `AT^SIND`.

`<lte_mask>`<sup>(str)(+CSCS)(NV)</sup>

LTE radio band(s) mask

`<lte_mask>` specifies LTE frequency bands in *hexadecimal 32-bit value order*. Every bit corresponds to a dedicated band number as defined in 3GPP standard TS 36.101 (4G).

When a bit (corresponding to a dedicated band) is set, the band is enabled, when the bit is cleared, the dedicated band is disabled respectively. The `<lte_mask>` value should be a string starting with '0x', to be transferred into binary follow below pattern.

Mask in Hexadecimal:	[Hex8]	[Hex7]	.....	[Hex2]	Hex1
Mask in binary:	b32b31b30b29	b28b27b26b25	.....	b8b7b6b5	b4b3b2b1

Hex8 to Hex2 can be omit if all zero, but `<lte_mask>` should contain no more than 8 hexadecimal digitals, and also contain at least one hexadecimal digital.

ENS22-E supported band listed below:

“0x00000004”	LTE BAND III (BC3)
“0x00000010”	LTE BAND V (BC5)
“0x00000080”	LTE BAND VIII (BC8)
“0x00080000”	LTE BAND XX (BC20)
“0x08000000”	LTE BAND XXVIII (BC28)

Parameter sets the LTE frequency band(s) the UE is allowed to use. The setting can be either a single or a multiple band value. Delivery default is the combination of all bands supported by the ENS22-E. Parameters must be set when ENS22-E is in minimum functionality (refer to `AT+CFUN=0`).

A changed value takes effect immediately.

An ERROR result code will be returned when the user tries

- to set a band not supported by the ENS22-E,
- to set `<lte_mask>= "0x00000000"`

`<tcpWithUrc>`<sup>(str)(+CSCS)(NV)</sup>

URC mode or polling mode for Internet service commands

This parameter enables or disables the presentation of the following URCs related to Internet service commands: `^SISR` URC, `^SISW` URC and `^SIS` URC for parameter `<urcCause>=0` (Internet service events). `^SIS` URCs with `<urcCause>=1` or `2` used to indicate incoming Socket connections are always enabled.

“on”<sup>(D)</sup> Enable URCs related to Internet service commands. Throughout the Chapter "Internet Service AT Commands" the mode is also referred to as URC mode.

“off” Disable URCs related to Internet service commands. This requires the TE to employ polling techniques when using the Internet service AT commands: The TE is responsible to retrieve all status information needed to control an Internet session. The method is referred to as polling mode.

`<urcRinglineCfg>`<sup>(str)(+CSCS)(NV)</sup>

Parameter specifies the ring line to be used for signaling URCs both for idle interface and while interface is reserved, i.e. while busy on AT command execution or data transmission.

For details about URC presentation refer to Section 1.7, [Unsolicited Result Code Presentation](#).

For details on using the RING0 line to wake up the TE refer to [2].

Setting is local for every interface.

“off“	URCs are not indicated by a ring line. This applies also when "off" is set on the ASC0 interface.
“local“ <sup>(D)</sup>	URCs are indicated by an activated ring line on the same interface where the URC appears.
“asc0“	URCs are indicated by activated RING0 line of the ASC0 interface (low active).

`<urcRinglineDuration>`<sup>(str)(+CSCS)(NV)</sup>

This parameter determines how long the ring line is activated in order to indicate a URC. For details about URC presentation refer to Section 1.7, [Unsolicited Result Code Presentation](#). For more information on the RING0 line and its timing for different types of URCs please refer to [2]. `<urcRinglineDuration>` is applicable to all types of URCs except the RING URC. The timing of the RING URC (incoming calls) is not configurable by AT command.

The type of ring line used for URC indication depends on parameter `<urcRinglineCfg>`.

Setting is global for all interfaces.

“0“	RING line will be activated for about 5s.
“1“	RING line will be activated for about 100 ms.
“2“ <sup>(D)</sup>	RING line will be activated for about 1 s.

`<reason>`<sup>(str)</sup>

Parameter indicates the reason of being woken-up from SUSPEND mode, with following values:

“urcOrData“	Module was woken-up due to a URC generated or data received.
“wake-in“	Module was woken-up through configured wake input.

### Example

Usage of "URC/Ringline":

AT+CSCS="GSM"	Switch to GSM character set.
OK	
AT^SCFG?	Query all parameters.
...	
^SCFG:"URC/Ringline","local"	URCs on this interface will be indicated by Ring line associated to the interface (e.g. RING0 for ASC0).
...	
OK	
AT^SCFG="URC/Ringline","asc0"	URCs on <i>this</i> interface will be indicated by an activated RING0 no matter whether or not the UE-TE link is reserved.
^SCFG:"URC/Ringline","asc0"	
OK	
AT^SCFG="URC/Ringline","off"	Disable any Ring line indication for URCs on this interface.
^SCFG:"URC/Ringline","off"	
OK	

## 2.12 AT^SPOW Set UART Mode and SLEEP Mode on UART

AT^SPOW enables or disables the UE's UART interfaces ASC0 and ASC1 and controls SLEEP mode on both interfaces. Parameter settings are non-volatile and common for all interfaces.

Please refer to "ENS22-E Hardware Interface Description, Version 02.000" for more information on power saving.

### Syntax

<p>Test Command</p> <p>AT^SPOW=?</p> <p>Response(s)</p> <p>^SPOW: (list of supported &lt;mode&gt;s), (time range of &lt;timeout&gt;)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>
<p>Read Command</p> <p>AT^SPOW?</p> <p>Response(s)</p> <p>^SPOW: &lt;mode&gt;, &lt;timeout&gt;</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>
<p>Write Command</p> <p>AT^SPOW=&lt;mode&gt;, &lt;timeout&gt;</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>
<p>PIN Last</p> <p>- -</p>

### Parameter Description

<b>&lt;mode&gt;</b> <sup>(num)(NV)</sup>	
1	UART interfaces ASC0 and ASC1 are activated without sleep mode support.
2 <sup>(D)</sup>	UART interfaces ASC0 and ASC1 are activated with sleep mode support.
<b>&lt;timeout&gt;</b> <sup>(num)(NV)</sup>	
100...5000 <sup>(D)</sup> ...10000	In SLEEP mode (<mode>=2), time in milliseconds the UE remains awake after the last sent character, in case the UART is ready to go SLEEP mode (i.e., the TE set RTS0 line to logical high). Minimum value: 100 ms, recommended 5000 ms. <timeout> values below 100 are denied with ERROR.

### Note

- The change takes effect after restart of the UE, e.g. with AT+CFUN=1,1.

## 3. Status Control Commands

The AT Commands described in this chapter allow the external application to obtain various status information from the ENS22-E.

### 3.1 AT^SIND Extended Indicator Control

[AT^SIND](#) controls the presentation of indicator event URCs. You can enable or disable URCs to be issued by the ENS22-E each time the value of the related indicator changes, you can request the current status of all indicators, and you can also query the status of a single indicator.

#### Syntax

##### Test Command

```
AT^SIND=?
```

##### Response(s)

```
^SIND: (<indDescr>, list of supported <indValue>s)[, (<indDescr>, list of supported <indValue>s)[,
...]], (list of supported <mode>s)
OK
```

##### Read Command

```
AT^SIND?
```

##### Response(s)

```
^SIND: <indDescr>, <mode>[, <indValue>]
[^SIND: <indDescr>, <mode>[, <indValue>]]
...

If <indDescr>="is_cert":
^SIND: is_cert, <mode>, <srvProfileId>, <issuer>, <serialNumber>, <subject>,
<signatureAlgorithm>, <thumbprintAlgorithm>, <thumbprint>

If <indDescr>="is_psk":
^SIND: is_psk, <mode>, <srvProfileId>, <pskUrl>, <pskId>
OK
ERROR
+CME ERROR: <err>
```

##### Write Command

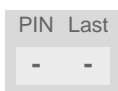
```
AT^SIND=<indDescr>, <mode>
```

##### Response(s)

```
^SIND: <indDescr>, <mode>[, <indValue>]

If <indDescr>="is_cert" and <mode>=2:
^SIND: is_cert, <mode>, <srvProfileId>, <issuer>, <serialNumber>, <subject>,
<signatureAlgorithm>, <thumbprintAlgorithm>, <thumbprint>

If <indDescr>="is_psk" and <mode>=2:
^SIND: is_psk, <mode>, <srvProfileId>, <pskUrl>, <pskId>
OK
ERROR
+CME ERROR: <err>
```



## Unsolicited Result Codes

### URC 1

Format of the standard indicator:

+CIEV: <indDescr>, <indValue>

Value of an indicator has changed.

### URC 2

Format of the "is\_cert" indicator:

+CIEV: <indDescr>, <srvProfileId>, <issuer>, <serialNumber>, <subject>, <signatureAlgorithm>, <thumbprintAlgorithm>, <thumbprint>

The URC is issued after opening a secure Internet connection with AT^SISO.

### URC 3

Format of the "is\_psk" indicator:

+CIEV: <indDescr>, <srvProfileId>, <pskUrl>, <pskId>

The URC is issued after opening a secure Internet connection with AT^SISO.

## Parameter Description

<indDescr><sup>(str)</sup>

This section describes <indDescr> values and their associated <indValue> ranges.

For command input on the AT^SIND write command line <indDescr> values are handled as string type. In responses and URCs <indDescr> values are output without quotation marks.

"celevel"	Coverage Enhancement Level: "celevel" delivers the current coverage enhancement level (CE level) of the serving cells. For "celevel", only <mode>=2 is supported. 0 Enhanced coverage level 0. 1 Enhanced coverage level 1. 2 Enhanced coverage level 2. 255 Default invalid value before any random access.
"simstatus"	(U)SIM Status Indication: The (U)SIM status indicator allows the ENS22-E to output the status of the Subscriber Identity Module via URC. The indication is non-volatile and disabled by default. 3 SIM PIN required - (U)SIM locked. SIM PIN must be entered to enable ENS22-E to register to the network. 4 Personalization info ready - SIM PIN is entered and ENS22-E is registered to the network; or SIM PIN is deactivated and not required.
"euiccid"	(U)SIM embedded identification number: "euiccid" delivers the (U)SIM embedded identification number retrieved from the (U)SIM Elementary File EF <sub>EUICCID</sub> .
"is_cert"	Report details of the server certificate used for a secure Internet connection opened with AT^SISO. The following certificate details are listed: <srvProfileId>, <issuer>, <serialNumber>, <subject>, <signatureAlgorithm>, <thumbprintAlgorithm>, <thumbprint> Detailed guidelines for managing the required certificates can be found in [4]. See also AT commands AT^SISS, AT^SBNR and AT^SBNW.



## 3.1 AT^SIND

"is_psk"	Report details of the server PSK used for a secure Internet connection opened with <a href="#">AT^SISO</a> . The following PSK details are listed: <a href="#">&lt;srvProfileId&gt;</a> , <a href="#">&lt;pskUrl&gt;</a> , <a href="#">&lt;pskId&gt;</a> Detailed guidelines for managing the required PSK can be found in [4]. See also AT commands <a href="#">AT^SISS</a> , <a href="#">AT^SSECUC</a> .
"suspendAvailable"	Suspend mode availability indication: 0 SUSPEND mode not available. The URC is triggered when: Suspend mode is disabled by <a href="#">&lt;Suspend&gt;="0"</a> , and PSM or eDRX mode is disabled 1 SUSPEND mode available. The URC is triggered when: Suspend mode is enabled by <a href="#">&lt;Suspend&gt;="1"</a> , and PSM or eDRX mode is enabled
"suspendReady"	Indication of module readiness to enter suspend mode: 0 SUSPEND mode not ready. The URC is triggered when the module is no longer ready to go to Suspend mode. The module will not go to Suspend mode when UARTs enter to Sleep mode. 1 SUSPEND mode ready. The URC is triggered when the module is ready to go to Suspend mode. The module will automatically go to Suspend mode when UARTs enter to Sleep mode.

**<indValue>**<sup>(num)</sup>Integer type value in the range stated above for the corresponding [<indDescr>](#).**<mode>**<sup>(num)</sup>Set or query URC presentation mode of a specific indicator [<indDescr>](#).

0	Disables the presentation of a specific URC. This is powerup value if <a href="#">&lt;indDescr&gt;</a> is not "suspendAvailable" or "suspendReady".
1	Enables the presentation of a specific URC. This is the powerup value if <a href="#">&lt;indDescr&gt;</a> is "suspendAvailable" or "suspendReady". The URC will be buffered in the UE when the UE-TE link is reserved (e.g. in online data mode), and flushed to the TE when the UE-TE link is free again. Otherwise, the URC will be forwarded directly to the TE.
2	Requests the presentation mode of the URC status and the current value of a single indicator type.

**<srvProfileId>**<sup>(num)</sup>Internet service profile identifier. Refer to [AT^SISS](#), [<srvProfileId>](#).**<issuer>**<sup>(str)</sup>Certificate's issuer. Refer to [<indDescr>](#).**<serialNumber>**<sup>(str)</sup>Certificate's serial number. Refer to [<indDescr>](#).**<subject>**<sup>(str)</sup>Certificate's subject. Refer to [<indDescr>](#).**<signatureAlgorithm>**<sup>(str)</sup>Certificate's signature algorithm. Refer to [<indDescr>](#).

---

`<thumbprintAlgorithm>(str)`

Certificate's thumbprint algorithm. Refer to [<indDescr>](#).

`<thumbprint>(str)`

Certificate's thumbprint. Refer to [<indDescr>](#).

`<pskUrl>(str)`

Server URL. Refer to [<indDescr>](#).

`<pskId>(str)`

PSK identity. Refer to [<indDescr>](#).

**Note**

- For "celevel", only `<mode>=2` is supported.

## 4. Serial Interface Control Commands

The AT Commands described in this chapter allow the external application to determine various settings related to the ENS22-E's serial interface.

### 4.1 AT\Q Flow Control

[AT\Q](#) allows to configure flow control on the ENS22-E's asynchronous serial interface ASC0 and ASC1. [AT\Q](#) setting is always common for all interfaces and can be changed, for compatibility reasons, on each interface, but is only applicable to ASC0 and ASC1.

#### Syntax

```
Exec Command
AT\Q[<n>]
Response(s)
OK

PIN Last
- -
```

#### Parameter Description

Parameter	Description
<n> <sup>(num)(&amp;V)(&amp;W)</sup>	
0 - 2	These values are accepted, but don't have any effect. <a href="#">AT\Q</a> setting is always reported as <n>=3 in the active profile (see <a href="#">AT&amp;V</a> ).
[3] <sup>(&amp;F)(D)</sup>	RTS/CTS hardware flow control

## 4.2 AT&C Set Data Carrier Detect (DCD) Line Mode

The [AT&C](#) command controls the behavior of the UE's DCD line.

[AT&C](#) offers full functionality only if the DCD0 line is enabled with [AT^SCFG](#)="GPIO/mode/DCD0","std". Otherwise [AT&C](#) will respond "+CME ERROR: operation not allowed".

### Syntax

Exec Command		Reference(s)	
AT&C[<value>]		V.250	
Response(s)			
OK			
PIN	Last		
-	-		

### Parameter Description

<value> <sup>(num)(&amp;V)(&amp;W)</sup>	Description
[0]	DCD line shall always be on.
1(&F)	DCD line shall be on only when data carrier signal is present.
2	DCD line shall be on when one or more Internet services defined on the related serial channel are in an active state as described below. For details on the various service states refer to <a href="#">AT^SISI</a> , parameter <srvState> or <a href="#">AT^SISO</a> , parameter <srvState>. The DCD line status for an Internet service is signaled at the interface where the service had been opened with the AT command <a href="#">AT^SISO</a> . <ul style="list-style-type: none"> <li>Transparent TCP or Transparent UDP client, SOCKET: DCD shall be on when &lt;srvState&gt;="Connecting" or "Up"</li> </ul>

## 4.3 AT&D Set Data Terminal Ready (DTR) Line Mode

**AT&D** determines how the UE responds if the DTR line is changed from ON to OFF state during data connection ( active Internet Services transparent access mode; refer to [AT^SIST](#)).

**AT&D** offers full functionality only if the DTR0 line is enabled with `AT^SCFG="GPIO/mode/DTR0", "std"`. Otherwise **AT&D** will respond "+CME ERROR: operation not allowed".

### Syntax

Exec Command	
AT&D[<value>]	
Response(s)	
OK	
PIN Last	Reference(s)
- -	V.250

### Parameter Description

<value> <sup>(num)(&amp;V)(&amp;W)</sup>	
[0]	UE ignores status of the DTR line.
1	ON->OFF on DTR: Change to command mode while retaining the data connection.
2 <sup>(&amp;F)</sup>	ON->OFF on DTR: The behaviour is like <a href="#">AT&amp;D1</a> .

## 4.4 AT&S Set Data Set Ready (DSR) Line Mode

**AT&S** determines how the UE sets the DSR line depending on its communication state.

**AT&S** offers full functionality only if the DSR0 line is enabled with `AT^SCFG="GPIO/mode/DSR0","std"`. Otherwise **AT&S** will respond "+CME ERROR: operation not allowed".

### Syntax

Exec Command	
AT&S[<value>]	
Response(s)	
OK	
PIN Last	Reference(s)
- -	ITU-T V.250 [9]

### Parameter Description

<value> <sup>(num)(&amp;V)(&amp;W)</sup>	
[0] <sup>(&amp;F)</sup>	DSR line is always ON
1	UE in command mode: DSR is OFF. UE in data mode: DSR is ON.

## 4.5 ATE AT Command Echo

ATE controls if the ENS22-E echoes characters received from TE during AT command state.

### Syntax

Exec Command	
ATE[<value>]	
Response(s)	
OK	
PIN Last	Reference(s)
- -	V.250

### Parameter Description

<value> <sup>(num)(&amp;V)(&amp;W)</sup>	
[0]	Echo mode off
1 <sup>(&amp;F)</sup>	Echo mode on

## 4.6 AT+ICF Character Framing

**AT+ICF** controls character framing and parity format used for receiving and transmitting data via ENS22-E's asynchronous serial interfaces (UART).

Following settings are supported:

- 7 bits, even parity, 1 stop bit (**AT+ICF=5,1**)
- 7 bits, odd parity, 1 stop bit (**AT+ICF=5,0**)
- 8 bits, even parity, 1 stop bit (**AT+ICF=2,1**)
- 8 bits, no parity, 1 stop bit (**AT+ICF=3**)
- 8 bits, odd parity, 1 stop bit (**AT+ICF=2,0**)
- 8 bits, no parity, 2 stop bits (**AT+ICF=1**)

If the current **<format>** setting uses no parity (e.g. **<format>=3**) and **AT+ICF** is used to switch to a format with parity (e.g. **<format>=2**), it is necessary to explicitly set parameter **<parity>**, e.g. **AT+ICF=2,1**. If not done correctly ENS22-E will respond with "+CME ERROR: incorrect parameters". This is because for **<format>=3**, the parameter **<parity>** is set to the internal value 'no parity' which is not supported by **<format>=2**.

### Syntax

<p>Test Command</p> <p>AT+ICF=?</p> <p>Response(s)</p> <p>+ICF: (list of supported <b>&lt;format&gt;</b>s), (list of supported <b>&lt;parity&gt;</b>s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+ICF?</p> <p>Response(s)</p> <p>+ICF: <b>&lt;format&gt;</b>[, <b>&lt;parity&gt;</b>]</p> <p>OK</p>	
<p>Write Command</p> <p>AT+ICF=[<b>&lt;format&gt;</b>[, <b>&lt;parity&gt;</b>]]</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: <b>&lt;err&gt;</b></p>	
<p>PIN Last</p> <p>- +</p>	<p>Reference(s)</p> <p>V.250</p>

### Parameter Description

<b>&lt;format&gt;</b> <sup>(num)(&amp;V)(&amp;W)</sup>	
Specifies the character format used for receiving and transmitting.	
1	8 data 0 parity 2 stop
2	8 data 1 parity 1 stop
3(&F)(D)	8 data 0 parity 1 stop
5	7 data 1 parity 1 stop



---

<parity><sup>(num)(&V)(&W)</sup>

Specifies the method of calculating the parity bit, if a parity bit is supported by <format>. If <format> does not support parity, this parameter has to be omitted.

0	odd
1	even

### Note

- Generally, **AT+ICF** should be used as a standalone command as specified in Section Combining AT commands on the same command line. If nevertheless combinations with other commands on the same command line cannot be avoided, there are several constraints to be considered:
  - Avoid combinations with the AT commands listed in Section Combining AT commands on the same command line.
  - Keep in mind that there shall be a minimum pause between two AT commands as specified in Section 1.5, [Communication between Customer Application and ENS22-E](#).
  - The selected setting is stored in the user defined profile with next **AT&W**. It will only be used on next start of the module. **AT+ICF** read command will always show the setting stored in the user defined profile, not a currently detected character framing.

## 4.7 AT+IPR Bit Rate

**AT+IPR** allows to query and set the bit rate of the ENS22-E's asynchronous serial interface (UART).  
The test command returns the values of supported fixed bit rates.  
The read command returns the current bit rate of the interface.  
The write command determines the bit rate to be used for the interface.

It is not recommended to set bit rates lower than 9600 bps in order to avoid timing problems (see Section 1.5, [Communication between Customer Application and ENS22-E](#) for details about timing).

It is highly recommended to use **AT+IPR** as a standalone AT command, i.e. better do not combine with other commands on the same AT command line.

The current setting of **AT+IPR** will be preserved after firmware download (i.e. a firmware update does not restore the factory setting), or in the event of power failure.

### Syntax

<p>Test Command</p> <p>AT+IPR=?</p> <p>Response(s)</p> <p>+IPR: ( ) , (list of supported selectable &lt;rate&gt;s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+IPR?</p> <p>Response(s)</p> <p>+IPR: &lt;rate&gt;</p> <p>OK</p>	
<p>Write Command</p> <p>AT+IPR=&lt;rate&gt;</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>V.250</p>

### Parameter Description

<p>&lt;rate&gt;<sup>(num)(&amp;V)(NV)</sup></p> <p>Bit rate per second (bps)</p> <p>1200</p> <p>2400</p> <p>4800</p> <p>9600</p> <p>19200</p> <p>38400</p> <p>57600</p> <p>115200<sup>(D)</sup></p> <p>230400</p>	
---	--

---

460800

921600

**Note**

- Generally, [AT+IPR](#) should be used as a standalone command as specified in Section Combining AT commands on the same command line. If nevertheless combinations with other commands on the same command line cannot be avoided, there are several constraints to be considered:
  - Avoid combinations with the AT commands listed in Section Combining AT commands on the same command line.
  - Keep in mind that there shall be a minimum pause between two AT commands as specified in Section [1.5, Communication between Customer Application and ENS22-E](#).
  - If switching to the new bit rate takes effect while a response is being transmitted, the last bytes may be sent with the new bit rate and thus, not properly transmitted. The following commands will be correctly sent at the new bit rate.  
Please consider this effect especially when local echo is active ([ATE1](#)).

## 5. Identification Commands

The AT Commands described in this chapter allow the external application to obtain various identification information related to the ENS22-E and linked entities.

### 5.1 ATI Display product identification information

The [ATI](#) execute command delivers a product information text.

#### Syntax

Exec Command

```
ATI
```

Response(s)

```
Cinterion  
ENS22-E  
REVISION <revUE>  
OK
```

Exec Command

```
ATI1
```

Response(s)

```
Cinterion  
ENS22-E  
REVISION <revUE>  
A-REVISION <revApp>  
OK
```

PIN Last

```
- -
```

Reference(s)

ITU-T V.250 [9]

#### Parameter Description

<revUE><sup>(num)</sup>

Version xx and variant yyy of software release.

<revApp><sup>(num)</sup>

Revision of Application software in the format XX.YYY.ZZ, where X, Y and Z are numbers.

## 5.2 AT+CGMI Request manufacturer identification

[AT+CGMI](#) returns a manufacturer identification text.

### Syntax

Test Command	
AT+CGMI=?	
Response(s)	
OK	
Exec Command	
AT+CGMI	
Response(s)	
Cinterion	
OK	
PIN Last	Reference(s)
- -	3GPP TS 27.007 <a href="#">[38]</a>

## 5.3 AT+CGMM Request model identification

[AT+CGMM](#) returns a product model identification text.

### Syntax

Test Command	
AT+CGMM=?	
Response(s)	
OK	
Exec Command	
AT+CGMM	
Response(s)	
ENS22-E	
OK	
PIN Last	Reference(s)
- -	3GPP TS 27.007 [38]

## 5.4 AT+CGMR Request revision identification and software version

AT+CGMR delivers product firmware version identification.

### Syntax

Test Command

AT+CGMR=?

Response(s)

OK

Exec Command

AT+CGMR

Response(s)

REVISION <number>

OK

PIN Last

- -

### Parameter Description

<number><sup>(str)</sup>

Version xx and variant yyy of software release.

## 5.5 AT+CGSN Request International Mobile Equipment Identity (IMEI)

AT+CGSN delivers the International Mobile Equipment Identity (IMEI).

### Syntax

#### Test Command

AT+CGSN=?

Response(s)

+CGSN: (list of supported <snt>s)

OK

#### Exec Command

AT+CGSN

Response(s)

<sn>

OK

#### Write Command

AT+CGSN=<snt>

Response(s)

If <snt>=0

<sn>

If <snt>=1

+CGSN: <sn>

If <snt>=2

+CGSN: <imeisv\_number>

If <snt>=3

+CGSN: <SVN>

OK

ERROR

+CME ERROR: <err>

PIN Last

- -

Reference(s)

3GPP TS 27.007 [38]

### Parameter Description

<sn><sup>(str)</sup>

International Mobile Equipment Identity (IMEI) identifying a mobile equipment used in the mobile network. IMEI is composed of Type Allocation Code (TAC) (8 digits), Serial Number (SNR) (6 digits) and the Check Digit (CD) (1 digit). For information on IMEI refer to 3GPP TS 23.003, subclause 6.2.1 and 3GPP TS 27.007 [38].

<snt><sup>(num)</sup>

Integer type indicating the serial number type that has been requested.



---

<imeisv\_number><sup>(num)</sup>

IMEISV (International Mobile station Equipment Identity and Software Version number).

The 16 digits of IMEISV are composed of Type Allocation Code (TAC) (8 digits), Serial Number (SNR) (6 digits) and the software version (SVN) (2 digits). For information on IMEISV refer to 3GPP TS 23.003, subclause 6.2.2 and 3GPP TS 27.007 [38].

<SVN><sup>(num)</sup>

Current SVN which is a part of IMEISV; refer 3GPP TS 23.003, subclause 6.2.2 and 3GPP TS 27.007 [38]. This allows identifying different software versions of a given mobile.

## 5.6 AT+CIMI Request International Mobile Subscriber Identity (IMSI)

[AT+CIMI](#) delivers the International Mobile Subscriber Identity (IMSI). The IMSI permits the TE to identify the individual SIM attached to the UE.

### Syntax

Test Command

```
AT+CIMI=?
```

Response(s)

```
OK
```

Exec Command

```
AT+CIMI
```

Response(s)

```
<imsi>
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
+ -
```

Reference(s)

3GPP TS 27.007 [38]

### Parameter Description

<imsi><sup>(str)</sup>

International Mobile Subscriber Identity (string without quotes).

## 6. Security Commands

The AT Commands described in this chapter allow the external application to determine various security related settings.

### 6.1 AT+CPIN PIN Authentication

The [AT+CPIN](#) write command can be used to enter one of the passwords listed below. The read command checks whether or not the UE is waiting for a password, or which type of password is required.

Each time a password is entered with [AT+CPIN](#) the module starts reading data from the SIM. The duration of reading varies with the SIM card. This may cause a delay of several seconds before all commands which need access to SIM data are effective. See Section [15.1, Restricted access to SIM data after SIM PIN authentication](#) for further detail.

#### Syntax

Test Command AT+CPIN=? Response(s) OK	
Read Command AT+CPIN? Response(s) +CPIN: <code> OK ERROR +CME ERROR: <err>	
Write Command AT+CPIN=<pin>[, <new pin>] Response(s) OK ERROR +CME ERROR: <err>	
PIN Last - -	Reference(s) 3GPP TS 27.007 <a href="#">[38]</a>

#### Parameter Description

<pin> <sup>(str)</sup> Password (string type), e.g. SIM PIN1. If the requested password was a PUK then <pin> must be followed by <new pin>.
<new pin> <sup>(str)</sup> If the requested code was a PUK: specify a new password or restore the former disabled password.

---

<code><sup>(text)</sup>

SIM PIN authentication

READY	PIN has already been entered. No further entry needed.
SIM PIN	ME is waiting for SIM PIN1.
SIM PUK	ME is waiting for SIM PUK1 if PIN1 was disabled after three failed attempts to enter PIN1.

**Notes**

- Successful PIN authentication only confirms that the entered PIN was recognized and correct. The output of the result code OK does not necessarily imply that the mobile is registered to the desired network. Typical example: PIN was entered and accepted with OK, but the ME fails to register to the network. This may be due to missing network coverage, denied network access with currently used SIM card, no valid roaming agreement between home network and currently available operators etc. ENS22-E offers various options to verify the present status of network registration: For example, the [AT+COPS](#) command indicates the currently used network. With [AT+CREG](#) you can also check the current status and activate an unsolicited result code which appears whenever the status of the network registration changes (e.g. when the ME is powered up, or when the network cell changes).
- See [AT+CPWD](#) for information on passwords.
- See [AT+CLCK](#) for information on lock types.

## 6.2 AT+CLCK Facility lock

[AT+CLCK](#) can be used to lock, unlock or interrogate a network or UE [<facility>](#). The command can be aborted when network facilities are being set or interrogated.

### Syntax

Test Command	
AT+CLCK=?	
Response(s)	
+CLCK: list of supported <a href="#">&lt;facility&gt;s</a>	
OK	
Write Command	
AT+CLCK= <a href="#">&lt;facility&gt;</a> , <a href="#">&lt;mode&gt;</a> [, <a href="#">&lt;password&gt;</a> ]	
Response(s)	
OK	
ERROR	
+CME ERROR: <a href="#">&lt;err&gt;</a>	
PIN Last	Reference(s)
- -	3GPP TS 27.007 <a href="#">[38]</a> , 3GPP TS 22.004 <a href="#">[18]</a> , 3GPP TS 22.088 <a href="#">[26]</a> , 3GPP TS 23.088, 3GPP TS 24.088

### Parameter Description

<a href="#">&lt;facility&gt;</a> <sup>(str)</sup>	
"SC"	SIM (lock (U)SIM cards) (U)SIM requests password upon UE power-up and when this lock command is issued. <a href="#">&lt;password&gt;</a> : SIM PIN1.
<a href="#">&lt;mode&gt;</a> <sup>(num)</sup>	
0	Unlock
1	Lock
<a href="#">&lt;password&gt;</a> <sup>(str)</sup>	
Password string used to lock and to unlock a <a href="#">&lt;facility&gt;</a> . Length and authority for passwords depend on the <a href="#">&lt;facility&gt;</a> in question and are therefore listed in the section on parameter <a href="#">&lt;facility&gt;</a> . Passwords can be modified with <a href="#">AT+CPWD</a> .	

### Example

Lock (U)SIM card ([<facility>](#)= "SC")

AT+CLCK="SC", 1, "9999"	The "SC" parameter enables or disables the SIM PIN authentication (PIN 1) when you power up the UE.
OK	(U)SIM card locked. As a result, SIM PIN 1 must be entered to enable UE to register to the network.

---

```
AT+CLCK="SC",0,"9999"  
OK
```

Unlocks SIM card.

When powered up, UE registers to the network without requesting SIM PIN1.

Note: Depending on the services offered by the provider, this feature is not supported by all SIM card types. If so, the command returns ERROR when you attempt to unlock the card.

## 6.3 AT+CPWD Change Password

**AT+CPWD** allows to define a new password for a password protected `<facility>` lock function. Each password is a string of digits, the length of which varies with the associated `<facility>`. The test command returns a list of pairs which represent the available facilities and the maximum length of the associated password. See **AT+CLCK** for more information on the various lock features.

### Syntax

<p>Test Command</p> <pre>AT+CPWD=?</pre> <p>Response(s)</p> <pre>+CPWD : list of supported (&lt;facility&gt;, &lt;password length&gt;) OK</pre>	
<p>Write Command</p> <pre>AT+CPWD=&lt;facility&gt;, &lt;old password&gt;, &lt;new password&gt;</pre> <p>Response(s)</p> <pre>New password has been registered for the facility lock function. OK If parameter &lt;old password&gt; was not correct: +CME ERROR: 16 (+CME ERROR: incorrect password) If the password for the selected &lt;facility&gt; has been invalidated due to too many failed attempts: +CME ERROR: ... If error is related to ME functionality: +CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>+ -</pre>	<p>Reference(s)</p> <pre>3GPP TS 27.007 [38]</pre>

### Parameter Description

<code>&lt;facility&gt;</code> <sup>(str)</sup>	<p>“SC“</p> <p>SIM PIN. (U)SIM requests password upon ME power-up and when this lock command is issued.</p> <p>If incorrectly entered three times, the SIM PUK is required to perform authentication. Input of the SIM PUK password is possible only with AT command <b>AT+CPIN</b>. <code>&lt;password length&gt;</code>: 4 to 8 digits.</p>
<code>&lt;password length&gt;</code> <sup>(num)</sup>	<p>4...8</p> <p>Length of password. The range of permitted length for a password depends on the associated <code>&lt;facility&gt;</code>. It is available from the test command response, or in the description of parameter <code>&lt;facility&gt;</code>.</p>
<code>&lt;old password&gt;</code> <sup>(str)</sup>	<p>Password specified for the facility.</p>
<code>&lt;new password&gt;</code> <sup>(str)</sup>	<p>New password.</p>

## 6.4 AT^SSECUC Internet Service Certificate Store Update Configuration

[AT^SSECUC](#) can be used to configure and query the pre-shared keys (PSKs) for Internet Service of the ENS22-E.

### Syntax

Write Command

Write/Read Pre-Shared Key information

```
AT^SSECUC="Cred/TLS/PSK/<slotID>"[, <PSK_identity>, <PSK_key>, [<PSK_url>]]
```

Response(s)

```
^SSECUC: "Cred/TLS/PSK/<slotID>", <PSK_identity>, , <PSK_url>  
OK  
ERROR  
+CME ERROR: <err>
```

Write Command

Remove Pre-Shared Key

```
AT^SSECUC="Cred/TLS/PSK/<slotID>", "", ""
```

Response(s)

```
^SSECUC: "Cred/TLS/PSK/<slotID>"  
OK  
ERROR  
+CME ERROR: <err>
```

PIN Last

- -

### Parameter Description

<slotID><sup>(num)</sup>

PSK slot ID

1...2

<PSK\_identity><sup>(str)(NV)</sup>

PSK identity in ASCII strings

Up to 16 printable characters.

<PSK\_key><sup>(str)(NV)</sup>

PSK key as ASCII string in hexadecimal encoding

Up to 16 octets in length, 32 characters.

<PSK\_url><sup>(str)(NV)</sup>

PSK URL for PSK auto selection

Up to 64 printable characters.

### Note

- [AT^SSECUC](#)="Cred/TLS/PSK/<slotID>" without any parameters queries current PSK slot value.



## 7. Network Service Commands

The AT Commands described in this chapter are related to various network services.

### 7.1 AT+COPS Operator Selection

**AT+COPS** queries the present status of the ENS22-E's network registration and allows to determine whether automatic or manual network selection shall be used.

Two operator selection modes are available with **AT+COPS**:

- **Automatic**  
ENS22-E searches for the home operator automatically. If successful the ENS22-E registers to the home network. If the home network is not found, ENS22-E goes on searching. If a permitted operator is found, ENS22-E registers to this operator.  
If no operator is found the ENS22-E remains unregistered.
- **Manual**  
Desired operator can be determined using the **AT+COPS** write command. If the operator is found, ENS22-E registers to it immediately. If the selected operator is forbidden, the ENS22-E remains unregistered.

The **AT+COPS** test command lists sets of five parameters, each representing an operator present in the network. A set consists of

- an integer indicating the availability of the operator,
- numeric format representation of the operator and
- an integer indicating the access technology of the operator.

Any of the parameters may be unavailable and will then be an empty field (,). The list of operators comes in the following order: Home network, networks referenced in SIM and other networks.

The operator list is followed by a list of the supported **<mode>**s and **<format>**s. These lists are delimited from the operator list by two commas.

The response to the **AT+COPS** test command is independent on the settings made with **<mode>** and **<rat>**. The **AT+COPS** test command will return a list of operators with the supported **<rat>**s.

The response to the **AT+COPS** read command depends on the registration status. If the UE is not registered, the read command returns only the current **<mode>**. If the UE is registered the response returns the currently selected operator, the currently set format and the currently used **<rat>**.

The **AT+COPS** write command forces an attempt to select and register to a network operator. If the selected operator is not available, no other operator will be selected. The selected operator name **<format>** will apply to further read commands, too.

#### Syntax

Test Command

AT+COPS=?

Response(s)

+COPS: [list of supported (**<opStatus>**, , , numeric **<opName>**[, **<rat>**])s ], , (list of supported **<mode>**s), (list of supported **<format>**s)

OK

ERROR

+CME ERROR: **<err>**

Read Command

AT+COPS?

Response(s)

+COPS: <mode>[, <format>, <opName>[, <rat>]]  
OK  
ERROR  
+CME ERROR: <err>

Write Command

AT+COPS=<mode>[, <format>[, <opName>[, <rat>]]]

Response(s)

OK  
ERROR  
+CME ERROR: <err>

PIN Last

± -

Reference(s)

3GPP TS 27.007 [38]

**Parameter Description**

<opStatus><sup>(num)</sup>

Operator Status

0	Unknown
1	Operator available
2	Current operator
3	Operator forbidden

<opName><sup>(str)&(V)</sup>

Operator Name

Operator name in numeric format.

<mode><sup>(num)&(V)</sup>

0 <sup>(D)</sup>	Automatic mode; <opName> field is ignored.
1	Manual operator selection After restarting the UE the network will be selected according to the priority order specified in 3GPP TS 23.122: "Last Registered PLMN", "Home PLMN", "Preferred PLMN" (related (U)SIM elementary files are EF_LOCI, EF_IMSI, EF_PLMNwAcT). The same priority order applies when swapping the (U)SIM during operation. In manual mode, only <format>=2 is supported.
2	Manually deregister from network and remain unregistered until <mode>=0 or 1 is selected.
3	Set only <format> (for AT+COPS read command).

<rat><sup>(num)</sup>

Radio Access Technology (RAT)

9	E-UTRAN (NB-S1 mode)
---	----------------------

---

<format><sup>(num)(&V)(&W)</sup>

2<sup>(&F)(D)</sup>

Numeric format of <opName>. This is the Location Area Identification (LAI) number, which consists of the 3-digit Mobile Country Code (MCC) plus the 2- or 3-digit Mobile Network Code (MNC).

## 7.2 AT+CESQ Extended Signal Quality

The `AT+CESQ` command returns received signal quality parameters. If the current serving cell is not a GERAN cell, `<rxlev>` and `<ber>` are set to value 99. If the current serving cell is not a UTRA FDD or UTRA TDD cell, `<rscp>` is set to 255. If the current serving cell is not a UTRA FDD cell, `<ecno>` is set to 255. If the current serving cell is not an E-UTRA cell, `<rsrq>` and `<rsrp>` are set to 255.

### Syntax

<p>Test Command</p> <p>AT+CESQ=?</p> <p>Response(s)</p> <p>+CESQ: (list of supported <code>&lt;rxlev&gt;</code>s), (list of supported <code>&lt;ber&gt;</code>s), (list of supported <code>&lt;rscp&gt;</code>s), (list of supported <code>&lt;ecno&gt;</code>s), (list of supported <code>&lt;rsrq&gt;</code>s), (list of supported <code>&lt;rsrp&gt;</code>s)</p> <p>OK</p>	
<p>Exec Command</p> <p>AT+CESQ</p> <p>Response(s)</p> <p>+CESQ: <code>&lt;rxlev&gt;</code>, <code>&lt;ber&gt;</code>, <code>&lt;rscp&gt;</code>, <code>&lt;ecno&gt;</code>, <code>&lt;rsrq&gt;</code>, <code>&lt;rsrp&gt;</code></p> <p>OK</p> <p>ERROR</p>	
<p>PIN Last</p> <p>+ -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Parameter Description

<code>&lt;rxlev&gt;</code> <sup>(num)</sup>	Received signal strength level (see 3GPP TS 45.008 [42] subclause 8.1.4). For 2G networks only
99	not known or not detectable
<code>&lt;ber&gt;</code> <sup>(num)</sup>	Channel bit error rate (in percent). For 2G networks only
99	not known or not detectable
<code>&lt;rscp&gt;</code> <sup>(num)</sup>	Received signal code power (see 3GPP TS 25.133 [36] subclauses 9.1.1.3 and 9.1.1.1.3) For 3G networks only
255	not known or not detectable
<code>&lt;ecno&gt;</code> <sup>(num)</sup>	Ratio of the received energy per PN chip to the total received power spectral density (see 3GPP TS 25.133 [36] subclause 9.1.2.3) For 3G networks only
255	not known or not detectable
<code>&lt;rsrq&gt;</code> <sup>(num)</sup>	Reference signal received quality (see 3GPP TS 36.133 [40] subclause 9.1.7). For 4G networks only
0	rsrq < -19.5 dB
1	-19.5 dB ≤ rsrq < -19 dB
2	-19 dB ≤ rsrq < -18.5 dB

---

...	
32	$-4 \text{ dB} \leq \text{rsrq} < -3.5 \text{ dB}$
33	$-3.5 \text{ dB} \leq \text{rsrq} < -3 \text{ dB}$
34	$-3 \text{ dB} \leq \text{rsrq}$
255	not known or not detectable

<rsrp><sup>(num)</sup>

Reference signal received power (see 3GPP TS 36.133 [40] subclause 9.1.4). For 4G networks only

0	$\text{rsrp} < -140 \text{ dBm}$
1	$-140 \text{ dBm} \leq \text{rsrp} < -139 \text{ dBm}$
2	$-139 \text{ dBm} \leq \text{rsrp} < -138 \text{ dBm}$
...	
95	$-46 \text{ dBm} \leq \text{rsrp} < -45 \text{ dBm}$
96	$-45 \text{ dBm} \leq \text{rsrp} < -44 \text{ dBm}$
97	$-44 \text{ dBm} \leq \text{rsrp}$
255	not known or not detectable

## 7.3 AT+CTZR Time Zone Reporting

The **AT+CTZR** command enables and disables event reporting for changes of time zone and daylight saving time. If reporting is enabled the ENS22-E returns unsolicited result codes whenever the time zone changes.

### Syntax

<p>Test Command</p> <p>AT+CTZR=?</p> <p>Response(s)</p> <p>+CTZR: (list of supported &lt;mode&gt;s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+CTZR?</p> <p>Response(s)</p> <p>+CTZR: &lt;mode&gt;</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Write Command</p> <p>AT+CTZR=&lt;mode&gt;</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Unsolicited Result Codes

URC 1

+CTZV: <tz>

The URC indicates that the time zone has changed.

URC 2

+CTZE: <tz>, <dst>[, <time>]

The URC provides information on local time and daylight saving time.

URC 3

+CTZEU: <tz>, <dst>[, <utime>]

The URC provides information on universal time and daylight saving time.

### Parameter Description

<mode> <sup>(num)</sup>	
0 <sup>(P)</sup>	Disable event reporting by URCs
1	Enable time event reporting by "+CTZV"
2	Enable time event reporting by "+CTZE"
3	Enable time event reporting by "+CTZEU"

---

<tz><sup>(str)</sup>

String type value representing the sum of the local time zone (difference between the local time and GMT expressed in quarters of an hour) plus daylight saving time.

<time><sup>(str)</sup>

Time in format "yyyy/mm/dd, hh:mm:ss", wherein characters indicates year, month, day, hour, minutes, seconds.

<utime><sup>(str)</sup>

Time in format "yyyy/mm/dd, hh:mm:ss", wherein characters indicates year, month, day, hour, minutes, seconds.

<dst><sup>(num)</sup>

Parameter indicates daylight saving time as follows:

- |   |   |
|---|---|
| 0 | No adjustment for daylight daving time      |
| 1 | +1 hour adjustment for daylight saving time |
| 2 | +2 hour adjustment for daylight saving time |

## 7.4 AT+CPSMS Enable or Disable Power Saving Mode

**AT+CPSMS** controls settings of the UE's power saving mode (PSM) parameters. It controls whether the UE wants to apply PSM or not, as well as the requested extended periodic TAU value in E-UTRAN and the requested Active Time value.

### Syntax

#### Test Command

AT+CPSMS=?

#### Response(s)

+CPSMS: (list of supported <mode>s),,, (list of supported <RequestedPeriodicTAU>s), (list of supported <RequestedActiveTime>s)  
OK  
ERROR  
+CME ERROR:<err>

#### Read Command

AT+CPSMS?

#### Response(s)

+CPSMS:<mode>,,[, <RequestedPeriodicTAU>][, <RequestedActiveTime>]  
OK  
ERROR  
+CME ERROR:<err>

#### Write Command

AT+CPSMS=[<mode>,,[, <RequestedPeriodicTAU>[, <RequestedActiveTime>]]]

#### Response(s)

OK  
ERROR  
+CME ERROR:<err>

#### PIN Last

- -

#### Reference(s)

3GPP TS 27.007 [38]

### Parameter Description

<mode><sup>(num)(NV)</sup>

Enable/disable the use of power saving mode (PSM) in the UE.

0	Disable the use of PSM
1 <sup>(D)</sup>	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

<RequestedPeriodicTAU><sup>(str)(NV)</sup>

One byte in an 8 bit format.

Requested extended periodic TAU value 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 more information on coding and value range, see 3GPP TS 24.008 [33], table 10.5.163a).



---

`<RequestedActiveTime>`<sup>(str)(NV)</sup>

One byte in an 8 bit format.

Requested Active time value 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 more information on coding and value range, see 3GPP TS 24.008 [33], table 10.5.163).

## 7.5 AT^SMONI Monitoring Serving Cell

The `AT^SMONI` command supplies information of the serving cell.

### Syntax

Test Command

```
AT^SMONI=?
```

Response(s)

```
OK
```

Exec Command

```
AT^SMONI
```

Response(s)

See: Section 7.5.1, [AT^SMONI Responses](#)

```
OK
```

PIN Last

```
- -
```

### 7.5.1 AT^SMONI Responses

*UE is not connected:*

- UE is camping on a LTE (NB-IoT) cell and registered to the network:

*Syntax:*

```
^SMONI: ACT,EARFCN,Band,Deployment type,PLMN code,TAC,Cell ID,RSRP,RSRQ,RSSI,TX  
power,Connection state,SNR
```

*Example:*

```
^SMONI: NB-IoT,3736,8,4,46000,3d45,3632849,-805,-117,-720,210,NOCONN,83
```

- UE is searching and could not (yet) find a suitable LTE (NB-IoT) cell:

*Syntax:*

```
^SMONI: ACT,EARFCN,Band,Deployment type,PLMN code,TAC,Cell ID,RSRP,RSRQ,RSSI,TX  
power,Connection state,SNR
```

*Example:*

```
^SMONI: NB-IoT,,,,,,,,,SEARCH,
```

- UE is camping on a LTE (NB-IoT) cell and registered to the network in PSM:

*Syntax:*

```
^SMONI: ACT,EARFCN,Band,Deployment type,PLMN code,TAC,Cell ID,RSRP,RSRQ,RSSI,TX  
power,Connection state,SNR
```

*Example:*

```
^SMONI: NB-IoT,3736,8,4,46000,3d45,3635748,-838,-113,-734,210,SLEEP,132
```

*UE is connected:*

- LTE (NB-IoT) cell:

*Syntax:*

```
^SMONI: ACT,EARFCN,Band,Deployment type,PLMN code,TAC,Cell ID,RSRP,RSRQ,RSSI,TX  
power,Connection state,SNR
```

*Example:*

```
^SMONI: NB-IoT,3736,8,4,46000,3d45,3635749,-874,-123,-738,210,CONN,37
```

Columns for LTE (NB-IoT) Serving Channel parameters:

Column	Description
ACT	Access Technology
EARFCN	E-UTRA Absolute Radio Frequency Channel Number
Band	E-UTRA frequency band (see 3GPP 36.101)
Deployment type	Unknown mode (0), in-band different PCI mode (1), in-band same PCI mode (2), guard band (3) or stand-alone (4)
PLMN code	Public Land Mobile Network code
TAC	Tracking Area Code (see 3GPP 23.003 Section 19.4.2.3)
Cell ID	Cell ID
RSRP	Reference Signal Received Power as measured by L1 in cBm (see 3GPP 36.214 Section 5.1.1.)
RSRQ	Reference Signal Received Quality as measured by L1 in cB (see 3GPP 36.214 Section 5.1.2.)
RSSI	Received signal strength indicator in cBm
TX power	Used Uplink Power in cBm
Conn_state	Connection state. Can be one of following: CONN, SLEEP, NOCONN or SEARCH
SNR	Signal-to-noise ratio

## 7.5.2 Service states

Depending on the service state, an additional textual output is generated (refer also to the response examples):

- "SEARCH" (Searching) - The MS is searching, but could not (yet) find a suitable cell. This output appears after restart of the MS or after loss of coverage.
- "NOCONN" (No connection) - The MS is camping on a cell and registered to the network. The service state is 'idle', i.e. there is no connection established or a dedicated channel in use.
- "SLEEP" (Sleeping) - The MS is in LTE PSM (Power Saving Mode).

## 7.6 AT^SMONP Monitoring Neighbour Cells

The `AT^SMONP` supplies information of active cells and all neighbour cells.

### Syntax

```

Test Command
AT^SMONP=?
Response(s)
OK

Exec Command
AT^SMONP
Response(s)
See: Section 7.6.1, AT^SMONP Responses
OK

PIN Last
- -
    
```

### Notes

- Due to the fact that not all necessary information of the neighbour cells can be decoded during a connection, only neighbour cells that have already been visible in IDLE mode will be further updated, as long as they are still included in the list.
- The neighbour cells have the same PLMN as the serving cell. In case of searching for a network the serving cell can change and the UE shows different sets of neighbour cells depending on the PLMN of the serving cell.

### 7.6.1 AT^SMONP Responses

- In case of a NB-IoT serving cell:

```

Syntax:
NB-IoT:
EARFCN1, PCI1, RSRP1, RSRQ1, RSSI1, SNR1
EARFCN2, PCI2, RSRP2, RSRQ2, RSSI2, SNR2
...
EARFCNn, PCIn, RSRPn, RSRQn, RSSIn, SNRn

Example:
NB-IoT:
3736,294,-793,-108,-676,-106
OK
    
```

Columns for NB-IoT parameters:

Column	Description
EARFCN	E-UTRA Absolute Radio Frequency Channel Number
PCI	Physical Cell ID

---

Column	Description
RSRP	Reference Signal Received Power as measured by L1 in dBm (see ETSI TS 136 214 version 10.1.0 Section 5.1.1.)
RSRQ	Reference Signal Received Quality as measured by L1 in dB (see ETSI TS 136 214 version 10.1.0 Section 5.1.3.)
RSSI	Received Signal Strength Indication as measured by L1 in dBm
SNR	Signal to Noise Ratio

## 7.7 AT+CEDRXS eDRX Setting

The [AT+CEDRXS](#) command controls the settings of eDRX parameters of the UE. 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 [AT+CEDRXS](#) test command displays the supported [<mode>](#)s and the value ranges for the access technology and the requested eDRX value as compound values.

The [AT+CEDRXS](#) read command displays current settings for each defined value of [<AcT-type>](#).

### Syntax

#### Test Command

```
AT+CEDRXS=?
```

Response(s)

```
+CEDRXS: (list of supported <mode>s), (list of supported <AcT-type>s), (list of supported <Requested_eDRX_value>s)
```

```
OK
```

#### Read Command

```
AT+CEDRXS?
```

Response(s)

```
[+CEDRXS: <AcT-type>, <Requested_eDRX_value>]
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

#### Write Command

```
AT+CEDRXS=[<mode>[, <AcT-type>[, <Requested_eDRX_value>]]]
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Unsolicited Result Code

When [<mode>](#)=2:

```
+CEDRXP: <AcT-type> [, <Requested_eDRX_value> [, <NW_provided_eDRX_value> [, <Paging_time_window>]]]
```

eDRX is enabled and URC code is enabled when the eDRX values provided by network has changed.

### Parameter Description

[<mode>](#)<sup>(num)</sup>

Enable or disable 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-type>](#).

[0] Disable the use of eDRX.

1 Enable the use of eDRX.

- 2 Enable the use of eDRX and enable the `<+CEDRXP>` unsolicited result code.
- 3 Disable the use of eDRX and reset the `<Requested_eDRX_value>` to default.

`<AcT-type>`<sup>(num)(NV)</sup>

Type of access technology. This parameter is used to specify the relationship between the type of the access technology and requested eDRX value.

- 0 Access technology is not using eDRX. This parameter value is only used in the unsolicited result code
- 5 E-UTRAN (NB-S1 mode)

`<Requested_eDRX_value>`<sup>(str)(NV)</sup>

Half a byte in a 4 bit format. The values are coded as follows:

- "0010" 20,48 seconds
- "0011" 40,96 seconds
- "0101" 81,92 seconds
- "1001" 163,84 seconds
- "1010" 327,68 seconds
- "1011" 655,36 seconds
- "1100" 1310,72 seconds
- "1101" 2621,44 seconds
- "1110" 5242,88 seconds
- "1111" 10485,76 seconds

`<NW_provided_eDRX_value>`<sup>(str)</sup>

Half a byte in a 4 bit format. The values are coded in the same way as for `<Requested_eDRX_value>`

`<Paging_time_window>`<sup>(str)</sup>

Half a byte in a 4 bit format. The requested values are applied immediately and are stored non-volatile. In case the module is attached to the network the change of the values will trigger tracking area update. The stored configuration is used during the network attach after restart in case the auto-attach is enabled.

The values are coded as follows:

- "0000" 2,56 seconds
- "0001" 5,12 seconds
- "0010" 7,68 seconds
- "0011" 10,24 seconds
- "0100" 12,8 seconds
- "0101" 15,36 seconds
- "0110" 17,92 seconds
- "0111" 20,48 seconds
- "1000" 23,04 seconds
- "1001" 25,6 seconds
- "1010" 28,16 seconds
- "1011" 30,72 seconds
- "1100" 33,28 seconds

---

"1101"	35,84 seconds
"1110"	38,4 seconds
"1111"	40,96 seconds



## 7.8 AT+CEDRXRDP Read dynamic eDRX parameters

The `AT+CEDRXRDP` command reads the current dynamic eDRX parameters applied in the cell. The 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, the `<AcT-type>=0` is returned.

### Syntax

Test Command	
AT+CEDRXRDP=?	
Response(s)	
OK	
Exec Command	
AT+CEDRXRDP	
Response(s)	
+CEDRXRDP: <AcT-type>[, <Requested_eDRX_value>[, <NW_provided_eDRX_value>[, <Paging_time_window>]]]	
OK	
ERROR	
+CME ERROR: <err>	
PIN Last	Reference(s)
- -	3GPP TS 27.007 [38]

## 7.9 AT^SNLWM2M Lwm2m Configuration Settings

[AT^SNLWM2M](#) can be used to query and configure Lwm2m settings of the ENS22-E.  
[AT^SNLWM2M](#) write command queries a configuration parameter (if no value is entered) or sets its value(s).  
[AT^SNLWM2M](#) read command returns a list of parameters and their current values.

### Syntax

#### Test Command

```
AT^SNLWM2M=?
```

#### Response(s)

```
^SNLWM2M: "cfg", "dft", "/0/0/2", ("2", "3")  
^SNLWM2M: "cfg", "dft", "/0/0/0", ("IP:Port")  
^SNLWM2M: "URC/srv", "dft", "enabled", ("0", "1")  
^SNLWM2M: "URC/procedure", "dft", "fwdownload", "enabled", ("0", "1")  
^SNLWM2M: "URC/procedure", "dft", "fwupdate", "enabled", ("0", "1")  
^SNLWM2M: "act", "dft", ("start", "stop")  
OK
```

#### Read Command

```
AT^SNLWM2M?
```

#### Response(s)

```
^SNLWM2M: "cfg", "dft", "/0/0/2", <rsc value>  
^SNLWM2M: "cfg", "dft", "/0/0/0", <rsc value>  
^SNLWM2M: "URC/srv", "dft", "enabled", <urc value>  
^SNLWM2M: "URC/procedure", "dft", "fwdownload", "enabled", <urc value>  
^SNLWM2M: "URC/procedure", "dft", "fwupdate", "enabled", <urc value>  
OK
```

#### Write Command

Resource configure and Query [<rsc value>](#) on AT^SNLWM2M interface.

```
AT^SNLWM2M="cfg", "dft", <mgmt rsc>[, <rsc value>]
```

#### Response(s)

```
^SNLWM2M: "cfg", "dft", <mgmt rsc>, <rsc value>  
OK  
ERROR  
+CME ERROR: <err>
```

#### Write Command

Enable LwM2M client

```
AT^SNLWM2M="act", "dft", <action>
```

#### Response(s)

```
^SNLWM2M: "act", "dft", <action>  
OK  
ERROR  
+CME ERROR: <err>
```

#### Write Command

Configure FOTA URC switch to inform about the server connection status

```
AT^SNLWM2M="URC/srv", "dft", <urc setting>[, <urc value>]
```

#### Response(s)

```
^SNLWM2M: "URC/srv", "dft", <urc setting>, <urc value>  
OK  
ERROR  
+CME ERROR: <err>
```

Write Command

Configure FOTA URC switch to inform about the client automatic processing

AT^SNLWM2M="URC/procedure","dft", <procedure>, <urc setting>[, <urc value>]

Response(s)

^SNLWM2M: "URC/procedure","dft", <procedure>,<urc setting>,<urc value>

OK

ERROR

+CME ERROR: <err>

PIN Last

- -

### Unsolicited Result Codes

#### URC 1

URC for status change of server connection:

^SNLWM2M: "srv","dft",<srv id>,"<srv status>"[, <status parameter>]

#### URC 2

URC for status change of server connection:

^SNLWM2M: "procedure","dft"[, <srv id>],<procedure>,<procedure status>[, <indication text>]

### Parameter Description

<mgmt rsc><sup>(str)</sup>

The following resource is configurable on AT^SNLWM2M interface. Where:

"/0/0/0" Lwm2m Server URI Resource. The <rsc value> is corresponding to <Lwm2m Server>.

"/0/0/2" Security Mode.The <rsc value> is corresponding <DTLS Switch>.

<rsc value><sup>(str)(NV)</sup>

Resource value is depending to <mgmt rsc>.

<Lwm2m Server><sup>(str)(NV)</sup>

This parameter is used for configure Lwm2m server when <mgmt rsc> is "/0/0/0". And format will be like this: "IP:Port". Where:

"IP" IPv4 address in dot-separated numeric (0-255) parameter of the form: a1.a2.a3.a4.

"Port" 0~65535. If the port value is omitted, the port is 5683 by default.

"NOTE" Only IPv4 is supported.  
The changes will take effect after reboot.

<DTLS Switch><sup>(str)(NV)</sup>

This parameter is used for enable/disable DTLS function when <mgmt rsc> is "/0/0/2". Where:

"2" Enable DTLS Function.

"3" Disable DTLS Function.

`<action>`<sup>(str)</sup>

This parameter is designed to execute an action which shall start/stop LWM2M client, where:

"start"	Start the service
"stop"	Stop the service

`<procedure>`<sup>(str)</sup>

This parameter is used for enabling URCs to inform about the client automatic processing, where:

"fwdownload"
"fwupdate"

`<urc setting>`<sup>(str)</sup>

Configures the urc property as "enabled" which means the URC is sent.

`<urc value>`<sup>(str)(NV)</sup>

Configures the urc property if the URC is sent or not, where:

"0"	The URC is not sent
"1"	The URC is sent

`<srv status>`<sup>(str)</sup>

This parameter is used for URC status when there is a change of server connection. Registered is the default value that are uniquely supported.

`<srv id>`<sup>(num)</sup>

1...x - short server id that initiated the action, procedure or status change.

1	Default values that are uniquely supported.
---	---

`<procedure status>`<sup>(str)</sup>

This parameter is used for URC in case of client internal procedure, where:

"init"
"finished"

`<status parameter>`<sup>(str)</sup>

IP Address Registered to Server. e.g. "35.221.220.95:5684"

`<indication text>`<sup>(str)</sup>

Downloaded IP address. e.g. "coaps://35.221.220.95"

## 7.10 AT+CIPCA Initial PDP context activation

**AT+CIPCA** controls whether an initial PDP context 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.

The **AT+CIPCA** test command returns values supported as a compound value.

The **AT+CIPCA** read command returns the current setting of the command.

The **AT+CIPCA** write command controls whether the UE is attached to E-UTRAN with or without a PDN connection.

### Syntax

<p>Test Command</p> <p>AT+CIPCA=?</p> <p>Response(s)</p> <p>+CIPCA: (list of supported&lt;n&gt;s), (list of supported&lt;AttachWithoutPDN&gt;s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+CIPCA?</p> <p>Response(s)</p> <p>[+CIPCA: &lt;n&gt;, &lt;AttachWithoutPDN&gt;]</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Write Command</p> <p>AT+CIPCA=[&lt;n&gt;[, &lt;AttachWithoutPDN&gt;]]</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Parameter Description

<n> <sup>(num)</sup>	
Activation of PDP context upon attach.	
3	No change in current setting.
<AttachWithoutPDN> <sup>(num)(NV)</sup>	
EPS Attach with or without PDN connection.	
[0]	EPS Attach with PDN connection.
1	EPS Attach without PDN connection.

## 7.11 AT+CCIOTOPT CloT Optimization Configuration

The [AT+CCIOTOPT](#) write command controls which CloT EPS optimizations the UE indicates as supported and preferred in the ATTACH REQUEST and TRACKING AREA UPDATE REQUEST messages.

The [AT+CCIOTOPT](#) write command is also used to control the URC "[+CCIOTOPTI](#)". The URC "[+CCIOTOPTI](#)" is used to indicate the supported CloT EPS optimizations by the network.

The read command returns the current settings for supported and preferred CloT EPS optimizations and the current status of the URC "[+CCIOTOPTI](#)".

### Syntax

<p>Test Command</p> <p>AT+CCIOTOPT=?</p> <p>Response(s)</p> <p>+CCIOTOPT: (list of supported &lt;n&gt;s), (list of supported &lt;supported_UE_opt&gt;s), (list of supported &lt;preferred_UE_opt&gt;s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Read Command</p> <p>AT+CCIOTOPT?</p> <p>Response(s)</p> <p>+CCIOTOPT: &lt;n&gt;, &lt;supported_UE_opt&gt;, &lt;preferred_UE_opt&gt;</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Write Command</p> <p>AT+CCIOTOPT=[&lt;n&gt;[, &lt;supported_UE_opt&gt;[, &lt;preferred_UE_opt&gt;]]]</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Unsolicited Result Code

+CCIOTOPTI: <supported\_Network\_opt>

Indicate the supported CloT EPS optimizations by the network

### Parameter Description

<n> <sup>(num)</sup>	
Enable/disable reporting of unsolicited code +CCIOTOPTI.	
0 <sup>(D)</sup>	Disable reporting
1	Enable reporting
3	Disable reporting and reset the parameters for CloT EPS optimization to the default values

<supported\_UE\_opt><sup>(num)(NV)</sup>

Indicate the UE's support for CloT EPS optimizations.

- |                  |   |
|------------------|---|
| 1 <sup>(D)</sup> | Support for control plane CloT EPS optimization   |
| 3                | Support for both control plane CloT EPS optimization and user plane CloT EPS optimization |

<preferred\_UE\_opt><sup>(num)(NV)</sup>

Indicate the UE's preference for CloT EPS optimizations.

- |                  |  |
|------------------|--|
| 1 <sup>(D)</sup> | Preference for control plane CloT EPS optimization   |
| 2                | Preference for user plane CloT EPS optimization. This can be set only when <supported_UE_opt>=3. |

<supported\_Network\_opt><sup>(num)</sup>

Indicate the Network support for CloT EPS optimizations.

- |   |   |
|---|---|
| 0 | No support  |
| 1 | Support for control plane CloT EPS optimization   |
| 2 | Support for user plane CloT EPS optimization  |
| 3 | Support for both control plane CloT EPS optimization and user plane CloT EPS optimization |

## 8. Internet Service Commands

ENS22-E has an embedded TCP/IP stack that is driven by AT commands and enables the host application to easily access the Internet. The advantage of this solution is that it eliminates the need for the application manufacturer to implement own TCP/IP, thus minimizing cost and time to integrate Internet connectivity into a new or existing host application. This chapter is a reference guide to all the AT commands and responses defined for use with the TCP/IP stack.

*The embedded TCP/IP stack provides the following services:*

1. Socket Services:
  - TCP Client in Non-Transparent or Transparent Mode
  - UDP Client in Non-Transparent or Transparent Mode
  - Non-Transparent UDP Endpoint
2. MQTT Client

Please note that a total number of 5 sockets is available for usage.

*Step-by-step overview of how to configure and use TCP/IP communications with ENS22-E:*

- Select URC mode or polling mode as described below.
- First of all, create a GPRS connection profile with `AT^SICS`. The connection profile is a set of basic parameters which determines the type of connection to use for Internet service. The connection type is also referred to as bearer. Up to 5 connection profiles can be defined, each identified by the `<conProfileId>`.
- Secondly, use `AT^SISS` to create a service profile based on one of the connection profiles. Up to 5 service profiles can be defined, each identified by the `<srvProfileId>`. The service profile specifies the type of Internet service to use, e.g., Socket, MQTT. To assign a connection profile to a service profile, the `<conProfileId>` of `AT^SICS` must be entered as "conId" value of the `AT^SISS` parameter `<srvParmTag>`. This offers flexibility to combine connection profiles and service profiles.
- Once the connection profile and the service profile are created, an Internet session can be opened by entering the `AT^SISO` write command and the desired `<srvProfileId>`. A URC indicates how to proceed. The "`^SISW`" URC means that data can be sent. The "`^SISR`" URC means that received data are available for reading. If an error occurs, the "`^SIS`" URC is delivered instead.
- The next steps differ depending on the service type and the transfer mode (Non-Transparent or Transparent).
- Once a service is opened, the `AT^SISI` command shall be used to monitor the progress of the session. The command reports the service state of the used service profile and indicates the number of bytes received, the number of bytes sent and, in the case of sending, the number of bytes acknowledged or unacknowledged at TCP layer.
- If an error occurs during a session, you can enter the `AT^SISE` command and the `<srvProfileId>` to identify the reason. This is especially important in the polling mode.
- The `AT^SICI` command can be used at any time to query the current status of one or all connection profile(s).
- Finally, to close an open service, enter the `AT^SISC` write command and the `<srvProfileId>`.
- All profile settings of `AT^SISS` and `AT^SICS` are volatile. To store the settings to non-volatile memory and load the stored profiles, `AT^SIPS` can be used.

*Transparent Mode or Non-Transparent Mode:*

- *Transparent Mode* is stream-oriented, i.e. a single data stream can be sent, and another single data stream can be received. Packetizing of data is done automatically by the embedded TCP/IP stack. Transparent Mode is activated with `AT^SIST` and then indicated by the "CONNECT" result code. The mode eliminates the need for `AT^SISW` and `AT^SISR` read and write sequences.
- *Non-Transparent Mode* is packet-oriented, i.e. data is transferred in separate packets. Maximum packet size is 1358 bytes for TCP and UDP. To read and write data, `AT^SISW` and `AT^SISR` sequences shall be used. To send or receive more than 1358 bytes, the read or write action shall be repeated until the data transfer has completed. Each read or write action requires that the command response (of `AT^SISR` or `AT^SISW`) confirms that the service is ready to send or receive data. The read/write cycles can be controlled by URCs or



polling. For details, see section "URC mode and polling mode" below.

To end an upload data stream, set the `<eodFlag>` in the last `AT^SISW` command.

- UDP Endpoint employs Non-Transparent Mode.  
TCP Client, UDP client can employ both modes, and may even mix both modes in the same connection.

#### *Socket service used with UDP protocol*

The significant differences between the TCP and UDP protocols imply that UDP sometimes requires particular procedures or even specific parameters. Details on how to handle UDP services can be found in extra notes or are included in the general parameter descriptions.

The ENS22-E offers two kinds of Socket service with UDP protocol.

- *UDP Client:* Intended for connections to a given remote host. In this case, the IP address and the UDP port of the remote host are set as a fixed parameter in the service profile. The UDP Client can use Transparent Mode and Non-Transparent Mode.
- *Non-Transparent UDP Endpoint:* IP address and UDP port of the remote hosts are handled in each read (`AT^SISR`) and write (`AT^SISW`) request. This enables the host application to communicate with different remote hosts.  
See example in Section 8.16.4, [UDP Scenario](#).

#### *MQTT client service*

The MQTT client provides basic MQTT functionality according to specification OASIS MQTT Version 3.1.1.

Supported transport protocols:

- TCP/IP - defined in URI as "mqtt://"
- TLS - defined in URI as "mqqtts://"

The MQTT client support sending MQTT control packets CONNECT, DISCONNECT, PUBLISH, SUBSCRIBE and UNSUBSCRIBE.

The MQTT client handle automatically MQTT control packets CONNACK, PUBACK, PUBREC, PUBCOMP, SUBACK, UNSUBACK, PINGREQ and PINGRESP.

#### *MQTT client control*

*Setup:* there are 3 types to set up and handle MQTT connection by AT commands interface from host device:

- Static parameters request - parameters are set up together with connection parameters by command `AT^SISS` before opening a connection with `AT^SISO`. Only one request is possible to be sent during active connection which is triggered by `AT^SISO`.
- Dynamic parameters - request parameters are set up by `AT^SISD` only after opening connection by `AT^SISO`. Multiple request are allowed to be send during active connection. Trigger for sending request is done by `AT^SISU`.
- Dynamic request - request parameters are set up and request itself is triggered by one `AT^SISU` only after opening connection by `AT^SISO`. Multiple requests are allowed to send during active connection.

*Triggers:* the host can trigger sending request to server by:

- `AT^SISO` command - request will be send automatically after successful connection. Request parameters needs to be defined statically by `AT^SISS`. Payload could be defined by hcContent parameter or by `AT^SISW`.
- `AT^SISU` command - request will be send during already active connection. Request parameters need to be defined dynamically by `AT^SISD` or `AT^SISU`. Payload could be defined by hcContent parameter or by `AT^SISW`.
- `AT^SISC` command - used for sending DISCONNECT control packet for correct end of connection with server.

Return of OK response from module signal accepting data for sending by MQTT client but not acknowledgement from MQTT server.

Getting information about the current status of the MQTT client is possible by URCs OR by polling from the host device.

#### Secure connection (TLS/DTLS)

All services except Listener services support server and client authentication for limited Transport Layer Security (TLS 1.2 / DTLS 1.2).

- To set TLS, add "s" to the address type, such as socktcp<sub>s</sub>, sockudp<sub>s</sub>, and mqtt<sub>s</sub>. For details see [AT^SISS](#) <srvParmTag> parameter "address".
- To create, load, delete and read customized TLS certificates, use [AT^SBNW](#) and [AT^SBNR](#).
- To create, delete and read PSKs, use [AT^SSECUC](#).
- The [AT^SISS](#) session profile parameter "secConf" switches between PKI mode and PSK mode.
- The [AT^SISS](#) session profile parameter "secOpt" specifies whether or not to check certificates received from the server against the local certificate store (if at least one local certificate is set).
- The used certificates can be shown in text format (issuer, serial, subject, signatureAlg, thumbprint). For this purpose, it is recommended to enable URC "+CIEV: "is\_cert". For details see [AT^SIND](#).
- Due to memory limitations, only one secure connection can be supported at a time.
- Due to connection I/O buffer size limitation, the maximum client certificate's size is 1536 bytes, and the maximum peer certificate chain's size is 2048 bytes.

#### URC mode or polling mode (for non-Transparent Mode only)

The ENS22-E offers two modes of controlling a Non-Transparent Mode session opened with [AT^SISO](#). To select the mode, use the [AT^SCFG](#) command, parameter "Tcp/WithURCs" (refer to <tcpWithUrc>).

- URC mode (delivery default):  
The progress of a Non-Transparent Mode session is URC driven. The URCs notify the host whether data can be sent or received, whether data transfer has completed, whether the service can be closed or whether an error has occurred. This mechanism eliminates the need to poll the service until the necessary progress information is received.  
To enable the URC mode, select: [AT^SCFG](#)="Tcp/WithURCs",on.
- Polling mode:  
In polling mode, the presentation of URCs related to Internet Services is disabled. The host is responsible to retrieve all the status information needed for controlling the Internet session. This is done by polling, where the host application keeps sending the commands [AT^SISR](#), [AT^SISW](#), [AT^SISI](#).  
To enable the polling mode select: [AT^SCFG](#)="Tcp/WithURCs",off.  
The disabled URCs are the following: "[^SISR](#)" URC, "[^SISW](#)" URC and "[^SIS](#)" URC for parameter <urcCause>=0 (Internet service events), but not for all other <urcCause>s (needed Endpoint and therefore always enabled).

#### Additional AT commands designed for controlling and monitoring the connection profiles (bearers):

- The [AT^SICI](#) command can be used at any time to query the current status of one or all connection profile(s).
- The [AT^SISX](#) command sends ICMP (Internet Control Message Protocol) Echo Requests to a target IP address or host name to verify the TCP/IP network connection.

#### Address notation

Server addresses shall be provided as IPv4 addresses in standard dot format, e.g. "192.168.1.2", or as IPv6 addresses in standard colon format enclosed in square brackets, e.g. "[FE80::2]", or as server address names resolvable by a DNS server.

#### Timeouts

Timeouts are not part of the Internet AT command functionality implemented in ENS22-E and, if desired, are the responsibility of the host application. It is recommended that the host application validates URCs and AT command responses and reacts adequately, for example by sending a close message or starting a timer.

*Using the DCD line to detect the connection status of Internet services*

With **AT&C** you can configure the DCD line of the used serial interface to indicate whether an Internet service is active. For Socket, the states "Up" or "Connecting" are indicated.

*Usage of IP Service commands on different interfaces*

The UE allows to create an Internet service profile on one interface with **AT^SISS** and to use it on another interface with **AT^SISO**, **AT^SISC**, **AT^SISR**, **AT^SISW**, **AT^SIST**. Nevertheless, to avoid conflicts a profile should be handled on the same interface where it was created with **AT^SISS**, at least where it was opened with **AT^SISO**.

## 8.1 AT^SICS Internet Connection Setup Profile

AT^SICS serves to create and edit Internet connection profiles. A connection profile can be assigned to one or more service profiles defined with AT^SISS, and thus, determines which type of connection is to be established when opening a service profile with AT^SISO.

The AT^SICS read command requests the current settings of all Internet connection profiles.

The AT^SICS write command specifies all parameters of a connection profile identified by <conProfileId>. To add or change the settings the write command needs to be executed for each single <conParmTag>. All profile parameters set with AT^SICS are volatile. To save profile settings in non-volatile memory please refer to AT^SIPS.

ENS22-E is designed to use PAP authentication (Password Authentication Protocol) when parameters "user" and "passwd" are given within the connection profile. If credentials are omitted, no authentication is performed.

The list below shows which <conParmTag> parameters are mandatory (= m) or optional (= o).

**Table 8.1:** Applicability of AT^SICS <conParmTag> values

<conParmTag>	<conParmValue-conType> GPRS0 or GPRS6
"conType"	m
"user"	o
"passwd"	o
"apn"	o
"dns1"	o
"alphabet"	o

### Syntax

<p>Test Command</p> <p>AT^SICS=?</p> <p>Response(s)</p> <p>OK</p>
<p>Read Command</p> <p>AT^SICS?</p> <p>Response(s)</p> <p>^SICS: &lt;conProfileId&gt;, &lt;conParmTag&gt;, &lt;conParmValue&gt;</p> <p>[^SICS: ...]</p> <p>OK</p>
<p>Write Command</p> <p>AT^SICS=&lt;conProfileId&gt;, &lt;conParmTag&gt;, &lt;conParmValue&gt;</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>
<p>PIN Last</p> <p>- -</p>

## Parameter Description

`<conProfileId>`<sup>(num)</sup>

0...4  
Internet connection profile identifier.  
The `<conProfileId>` identifies all parameters of a connection profile, and, when a service profile is created with `AT^SISS` the `<conProfileId>` needs to be set as "conId" value of the `AT^SISS` parameter `<srvParmTag>`.

`<conParmTag>`<sup>(str)</sup>

Internet connection parameter.

"conType"  
Type of Internet connection.  
For supported values of `<conParmValue>` refer to `<conParmValue-conType>`.

"alphabet"  
Selects the character set for input and output of string parameters within a profile.  
The selected value is bound to the specific profile. This means that different profiles may use different alphabets.  
It is recommended that "alphabet" should be set after `<conParmValue-conType>` and before other parameters.  
For supported values of `<conParmValue>`, refer to `<conParmValue-alphabet>`.

"user"  
User name string: maximum 31 characters (where "" is default).

"passwd"  
Password string: maximum 31 characters (where "\*\*\*\*" is default).

"apn"  
Access point name (APN) string value: maximum 63 characters (where "" is default).

"dns1"  
Primary DNS server address (IPv4 address in dotted-four-byte format or IPv6 address in colon format enclosed in square brackets).  
This value determines whether to use the DNS server addresses dynamically assigned by the network or a specific DNS server address given by the user.  
"dns1" = "0.0.0.0" (default) means that the connection profile uses dynamic DNS assignment. Any other address means that the Primary DNS is manually set.  
The default value applies automatically if no other address is set. The `AT^SICS` read command only returns a manually configured IP address.  
See also note below.

`<conParmValue>`<sup>(str)(+CSCS)</sup>

Parameter value; type and supported content depend on related `<conParmTag>`.

`<conParmValue-conType>`<sup>(str)(+CSCS)</sup>

Supported connection type values in `<conParmValue>` for `<conParmTag>` value "conType".

"GPRS0"  
Packet switched connection type IPv4.

"GPRS6"  
Packet switched connection type IPv6.

"none"  
Clears the connection profile.

`<conParmValue-alphabet>`<sup>(str)(+CSCS)</sup>

Supported string parameter character set selections in `<conParmValue>` for `<conParmTag>` value "alphabet".

["0"  
Applicable character set is determined by current setting of `AT+CSCS`.

"1"  
International Reference Alphabet (IRA, seven bit ASCII).

### Notes

- After entering `AT^SISO`, ENS22-E automatically tries to find and attach to an existed Packet Domain with same `<conParmValue-conType>` and APN, if not found, ENS22-E automatically tries to define and activate a new Packet Domain with `<conParmValue-conType>` and APN. After entering `AT^SISC`, the automatically defined Packet Domain will be deleted if no opening service is based on it. Yet, the only exception is `AT+CGATT` which can be used any time to detach from the Packet Domain and thus disconnect the bearer opened with `AT^SISO`.
- If the network does not support or is not correctly configured for automatic DNS address assignment, the TCP/IP stack cannot resolve fully qualified domain names. In this case, a warning message will be returned when trying to open an Internet service configured for automatic DNS address assignment.

### 8.1.1 Example: GPRS connection profile

<code>at^sics=0,"conType","GPRS0"</code>	Select connection type GPRS0.
OK	
<code>AT^SICS=0,dns1,"193.254.160.1"</code>	IP address of Primary DNS server.
OK	
<code>AT^SICS=0,"alphabet","1"</code>	The character set of string parameters(ASCII).
OK	
<code>AT^SICS=0,"passwd","t-d1"</code>	Password for GPRS services provided by the German operator T-D1.
OK	
<code>AT^SICS=0,"apn","internet.t-d1.de"</code>	APN to access the GPRS services provided by the German operator T-D1.
OK	

## 8.2 AT^SICI Internet Connection Information

The `AT^SICI` read command requests the current status of the Internet connection profiles currently defined with `AT^SICS`.

The `AT^SICI` write command displays the status of the specified Internet connection profile. If the connection profile is not defined yet, the command returns a "+CME ERROR" response.

### Syntax

Test Command

```
AT^SICI=?
```

Response(s)

```
^SICI: (list of supported <conProfileId>s)
```

```
OK
```

Read Command

```
AT^SICI?
```

Response(s)

```
[^SICI: <conProfileId>, <conState>, <numServices>, <conAddr>]
```

```
[^SICI: ...]
```

```
OK
```

Write Command

```
AT^SICI=<conProfileId>
```

Response(s)

```
^SICI: <conProfileId>, <conState>, <numServices>, <conAddr>
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

`<conProfileId>`<sup>(num)</sup>

Internet connection profile. Internet connection profile identifier as defined by `AT^SICS (<conProfileId>)`.

0...4

`<conState>`<sup>(num)</sup>

State of the Internet connection profile.

- |   |   |
|---|---|
| 0 | Down - Internet connection is defined but not connected.                              |
| 1 | Connecting - A service has been opened and so the Internet connection is initiated.   |
| 2 | Up - Internet connection is established and usable by one or more services.           |
| 3 | Limited Up - Internet connection is established, but temporarily no network coverage. |

`<numServices>`<sup>(num)</sup>

Number of services using this Internet connection profile.

0...5

<conAddr><sup>(str)</sup>

Local IPv4 address of the Internet connection profile ("0.0.0.0" if no address is associated yet), or local IPv6 address ("::" if no address is associated yet).

## 8.2.1 Checking Connection Profile Status

```
at^sici?  
^SICI: 1,2,1,"10.10.0.161"
```

Query the current status of the connection profile. One connection profile has been created at <conProfileId>=1, and one service is opened using this connection profile. The Internet connection is in <conState>=2 ("Up").

OK

Assume the network connection is temporarily not available. (If URC presentation mode of network registration is enabled the state is shown by the URC "+CEREG: 2").

```
at^sici?  
^SICI: 1,3,1,"10.10.0.161"
```

Query once again the current status of the connection profile. Parameter <conState> has changed to state "Limited up".

OK

The network connection is temporarily not available but the Internet connection is still established, so the host may decide to stop the data transfer to minimize the risk of losing data due to memory limitations.

```
at^sici?  
^SICI: 1,2,1,"10.10.0.161"
```

Query once again the current status of the connection profile. After re-establishing the network connection, <conState> reverts to state "Up".

OK

Another service has been opened with AT^SISO:

```
at^sici?  
^SICI: 1,2,2,"10.10.0.161"
```

Query once again the current status of the connection profile. After opening another service, parameter <numServices> has increased.

OK

After closing one service with AT^SISC:

```
at^sici?  
^SICI: 1,2,1,"10.10.0.161"  
OK
```

Query once again the current status of the connection profile. Parameter <numServices> has decreased.

After closing the last service with AT^SISC:

```
at^sici?  
^SICI: 1,0,0,"0.0.0.0"
```

Query the current status. The bearer is closed now. Parameter <numServices> has decreased.

OK



## 8.3 AT^SIPS Internet Profile Storage

AT^SIPS saves and loads the settings of the Internet connection and service profiles in the non-volatile memory. AT^SIPS can also be used to reset the settings to their default values without saving.

### Syntax

Test Command

AT^SIPS=?

Response(s)

^SIPS: (list of supported <type>s), (list of supported <action>s), (list of supported <ProfileId>s)  
OK

Write Command

AT^SIPS=<type>, <action>[, <ProfileId>]

Response(s)

OK  
ERROR  
+CME ERROR: <err>

PIN Last

- -

### Parameter Description

<type><sup>(str)</sup>

Internet profile storage type parameter.

“connection”	Handle AT^SICS parameter set.
“service”	Handle AT^SISS parameter set.
“all”	Handle AT^SICS parameters and AT^SISS parameters listed above.

<action><sup>(str)</sup>

Internet profile storage action parameter

“reset”	Reset current profile(s) for given <type> to default settings without saving. If <ProfileId> is given only specific profile will be reset. Only the connections which are in "Down" status and the services which are in "Allocated" status could be affected.
“save”	Stores current profile(s) for given <type>. If <ProfileId> is given only specific profile will be stored.
“load”	Loads stored profile(s) for given <type>. If <ProfileId> is given only specific profile will be loaded.

<ProfileId><sup>(num)</sup>

Internet connection or service profile storage identifier

0...4	In case of connection profile, max = 4, please see <conProfileId>. In case of a service profile, max = 4, please see <srvProfileId>.
	If parameter is omitted, AT^SIPS applies to all possible profile identifiers of used <type>.

**Note**

- All stored settings of IP service cannot be loaded automatically during startup. They can only be loaded from non-volatile memory by executing `AT^SIPS=<type>, load[,<ProfileId>]`.

## 8.4 AT^SISS Internet Service Setup Profile

AT^SISS serves to set up the necessary parameters in the Internet service profiles. Service profiles can then be used to control a data link in conjunction with AT^SISI, AT^SISO, AT^SISC, AT^SISR, AT^SISW, AT^SIST, and AT^SISE.

The AT^SISS read command requests the current settings of all Internet service profiles. One line is issued for every possible parameter of a given <srvParmTag> "srvType" value.

The AT^SISS write command specifies the parameters for a service profile identified by <srvProfileId>. At first the type of Internet service needs to be selected via <srvParmTag> value "srvType". This determines the applicability of all other <srvParmTag> values related to this "srvType" and sets their defaults. Changing the <srvParmTag> "srvType" of an existing service profile will reset the <srvParmTag> default values to suit the new "srvType".

To change the settings the write command needs to be executed for each single <srvParmTag>. All profile parameters set with AT^SISS are volatile. To save profile settings in non-volatile memory please refer to AT^SIPS.

The list below shows which <srvParmTag> parameters include socket and MQTT connection setup parameters which of them are mandatory (= m) or optional (= o). These parameters will save to non-volatile memory by AT^SIPS.

**Table 8.2:** Applicability of AT^SISS <srvParmTag> values

<srvParmTag>	Socket	MQTT
SrvType	m	m
conId	m	m
alphabet	o	o
address	m	m
secOpt	o	o
secConf	o	o
protocolVersion		o
user		o
passwd		o
clientId		o
keepAlive		o
lastWillTopic		o
lastWillMessage		o
lastWillQos		o
lastWillRetain		o
cleanSession		o

If MQTT service connection will open with AT^SISO parameter <optParam>=0:

The list below shows which <srvParmTag> parameters apply to MQTT service <mqttRequestParameter> and which of them are mandatory (= m) or optional (= o).

**Table 8.3:** Applicability of AT^SISS <srvParmTag> values

<srvParmTag>	MQTT
<i>Cmd = Publish</i>	
Cmd	m
Qos	o

<srvParmTag>	MQTT
Topic	m
Retain	o
hcContent	o
hcContentLen	o
<i>Cmd = Subscribe</i>	
Cmd	m
topicFilter	m
topicQos	o
<i>Cmd = Unsubscribe</i>	
Cmd	m
topicFilter	m

If MQTT service connection will open with AT^SISO parameter <optParam>=2: <mqttRequestParameter> be omitted.

### Syntax

```
Test Command
AT^SISS=?
Response(s)
OK
```

```
Read Command
AT^SISS?
Response(s)
^SISS: <srvProfileId>, <srvParmTag>, <srvParmValue>
[^SISS: ...]
OK
ERROR
+CME ERROR: <err>
```

```
Write Command
AT^SISS=<srvProfileId>, <srvParmTag>, <srvParmValue>
Response(s)
OK
ERROR
+CME ERROR: <err>
```

```
PIN Last
- -
```

### Parameter Description

<srvProfileId> <sup>(num)</sup>	
0...4	Internet service profile identifier. The <srvProfileId> is used to reference all parameters related to the same service profile. Furthermore, when using the AT commands AT^SISO, AT^SISR, AT^SISW, AT^SIST AT^SISD, AT^SISU and AT^SISC the <srvProfileId> is needed to select a specific service profile.

<srvParmTag><sup>(u)</sup>

Internet service profile parameter.

srvType	Type of Internet service to be configured with consecutive usage of AT^SISS. For supported values of <srvParmValue> refer to <srvParmValue-srv-Type>.
alphabet	Selects the character set for input and output of string parameters within a profile. The selected value is bound to the specific profile. This means that different profiles may use different alphabets. It is recommended that "alphabet" should be set after <srvParmValue-srv-Type> and before other parameters. For supported values of <srvParmValue> refer to <srvParmValue-alphabet>.
conId	Internet connection profile to be used, for details refer AT^SICS.
address	String value containing the URL for the specific service: If address contains a semicolon, quotes are mandatory. When GSM alphabet is enabled (see AT+CSCS), keep in mind that escape sequences are required for some special characters: For example, underscore "_" shall be escaped with \1. See also Section 1.6, Supported character sets for more information on character set and escape sequences.

1. Non-transparent Socket
  - TCP client URL  
"socktcp[s]://<host>:<remotePort>" "
  - UDP endpoint URL  
"sockudp://:<localPort>"
  - UDP client URL  
"sockudp[s]://<host>:<remotePort>[;port=<localPort>]"
2. Transparent Socket
  - Transparent TCP client  
"socktcp[s]://<host>:<remotePort>;etx[=<etxChar>][;timer=<value>]" "
  - Transparent UDP client  
"sockudp[s]://<host>:<remotePort>[;port=<localPort>];etx[=<etxChar>][;timer=<value>]"

Supported parameters for Socket services:

- <host>: element can be an FQDN, or IPv4 address in standard dot format, e.g. "192.168.1.2", or IPv6 address in standard colon format enclosed in square brackets, e.g. "[FE80::2]".
- <remotePort>: 1 ... 2<sup>16</sup>-1 for the remote port number.
- <localPort>: 1 ... 2<sup>16</sup>-1 for the local port number.
- "port": Local port of UDP client:  
0: Port number will be assigned from service (default).  
1...2<sup>16</sup>-1: defines the local port number for the UDP client.
- "etx": Parameter is mandatory for Transparent Socket. Otherwise Socket service is non-transparent.
- "<etxChar>": Specifies the character used to change from transparent access mode to AT command mode.  
range: 1...15,17...255 (16 is not allowed because it is used as DLE (0x10))  
If parameter is not set no escaping is configured, thus requiring either +++ or DTR ON-OFF transition for changing to AT command mode. If value is set, the transmitted bytes are parsed for the DLE (0x10) character followed by the specified <etxChar> value. If both characters are found the service returns to AT command mode without transmitting these two bytes (same as +++).

If you wish to send DLE characters as normal text string within your payload data the characters shall be doubled (DLE DLE).

- "timer": The parameter configures the Nagle algorithm, which is used in transparent access mode.  
range: 20...[100]...500 milliseconds in steps of 20.

### 3. MQTT

MQTT client URL

"mqtt[s]://[<user>:<passwd>@]<host>[:<port>];connackTime-out=<value>];retryMsgTime=<value>]"

<host>: element can be an FQDN, or an IPv4 address, or an IPv6 address enclosed in square brackets.

<user>: username for authentication, 12 octets.

<passwd>: password for authentication, 12 octets.

<port>: If parameter is omitted the service connects to default port 1883 and for TLS default port 8883.

"connackTimeout": Definition of max time in seconds waiting for CONNACK packet from server.

range: 5...[20]...60 seconds.

"retryMsgTime": Definition of number of seconds to wait before retrying messages with Qos > 0.

range: 3...[10]...60 seconds.

secopt

Security Option of IP service.  
(numeric)

- 0(D) Do not check received certificates from server against the local store.
- 1 Check received certificates from server against local store, if local store has at least one certificate.

NOTE: Local store is created with AT^SBNW,is\_cert

secConf

Configure secure connection mode.

(string) "[mode=<mode>];slotID=<slotID>]", where <mode>: (numeric)

- 0(D) - PKI, the module will use PKI, slotID will be ignored.
- 1 - PSK, the module will use PSK. If slotID is absent, automatic selection will be used, the module will try to search PSK\_URL and PSK\_server\_hint stored in all of the PSK information slot.

PSK Slot is created with AT^SSECUC, "Cred/TLS/PSK/<slotID>"

NOTE: if "secConf" value is empty, it means "mode=0"

protocolVersion

Define MQTT protocol specification version.

- 0: Specification version 3.1
- 1: Specification version 3.1.1

If this parameter is not set, default protocolVersion 1 is used.

user

User name string for MQTT authentication.

Parameter is optional: The user name can be set by <srvParmTag> parameter "user" and/or as part of "address" parameter. If set in both parameters at the same time, only the one given in the "address" parameter will be used, ignoring the one given in <srvParmTag> parameter "user".

For MQTT only simple authentication is supported.

Length: 12 octets

passwd

Password string for MQTT authentication.

Parameter is optional: The password can be set by <srvParmTag> parameter "passwd" and/or as part of the "address" parameter. If set in both parameters at the same time, only the one given in the "address" parameter will be used, ignoring the one given in <srvParmTag> parameter "passwd".

Length: 12 octets

clientId	Unique client identification of the connection to the MQTT server. ClientId are between 1 and 23 UTF-8 encoded bytes in length, and that contain only the following characters "0123456789abcdefghijklmnopqrstuvwxyABCDEFGHIJKLMNOPQRSTUVWXYZWXYZ"
keepAlive	Define keep alive time interval defined in seconds. range: [0]...65535 seconds
lastWillTopic	Definition of last will topic. Length: 0...64 octets
lastWillMessage	Definition of last will message. Length: 0...128 octets
lastWillQos	Definition of Qos level of publishing last will message. <ul style="list-style-type: none"> <li>• 0: At most once delivery</li> <li>• 1: At least once delivery</li> <li>• 2: Exactly once delivery</li> </ul> If this parameter is not set, default Qos 0 is used.
lastWillRetain	Definition if last will message is to be retained by server. <ul style="list-style-type: none"> <li>• 0: Retained message MUST NOT be stored as a retained message on the Server</li> <li>• 1: The Server MUST store the Application Message</li> </ul> If this parameter is not set, default Retain 0 is used.
cleanSession	Definition of session type to be initiated with server. <ul style="list-style-type: none"> <li>• 0: Server will try to resume current session with client</li> <li>• 1: Server will start new clean session</li> </ul> If this parameter is not set, default value 1 is used.
cmd	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
Qos	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
Topic	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
Retain	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
hcContent	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
hcContentLen	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
topicFilter	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
topicQos	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .

[<srvParmValue>](#)<sup>(str)(+CSCS)</sup>

Parameter value; type and supported content depend on related [<srvParmTag>](#).

[<srvParmValue-srvType>](#)<sup>(str)(+CSCS)</sup>

Supported Internet service type values in [<srvParmValue>](#) for [<srvParmTag>](#) value "srvType". Before changing the "srvType" of an existing service profile be sure that the profile is closed. To verify the connection state of the service profile enter the read command AT^SISI. Only when [<srvState>](#)=2 is returned for this specific service profile you can change its service type.

“Socket“ ENS22-E acting as client or server for TCP or UDP. "etx" subparameter in URL configures Transparent mode for Socket service.

"Mqtt"	ENS22-E acting as MQTT client. The MQTT client support sending MQTT control packets CONNECT, DISCONNECT, PUBLISH, SUBSCRIBE and UNSUBSCRIBE. The MQTT client handle automatically MQTT control packets CONNACK, PUBACK, PUBREC, PUBCOMP, SUBACK, UNSUBACK, PINGREQ and PINGRESP. The maximum number of configurable MQTT client is 1.
"none"	Reset Internet service profile settings. Operation is not allowed if profile is in use, i.e. it was activated via <a href="#">AT^SISO</a> .

`<mqttRequestParameter>(str)(+CSCS)`

Supported MQTT service type values in `<srvParmValue>` for `<srvParmTag>` MQTT request setup parameters.

"cmd"	Type of MQTT request. <ul style="list-style-type: none"> <li>• "publish": The MQTT client sends data by PUBLISH control packet request to server</li> <li>• "subscribe": The MQTT client that wants to receive data must subscribe to it by SUBSCRIBE control packet</li> <li>• "unsubscribe": The MQTT client that doesn't want to receive subscribed data anymore must unsubscribe specified topic filter by sending UNSUBSCRIBE control packet</li> </ul>
"Qos"	Definition of Qos level of publishing application message. <ul style="list-style-type: none"> <li>• 0: At most once delivery</li> <li>• 1: At least once delivery</li> <li>• 2: Exactly once delivery</li> </ul> <p>If this parameter is not set, default Qos 0 is used.</p>
"Topic"	Definition topic of publishing application message. Length: 0...64
"Retain"	Flag define if application message will be retained on server. <ul style="list-style-type: none"> <li>• 0: Retained message MUST NOT be stored as a retained message on the Server</li> <li>• 1: The Server MUST store the Application Message</li> </ul> <p>If this parameter is not set, default Retain 0 is used.</p>
"hcContent"	Definition of payload in case parameter "hcContentLen" = 0. Length: 0...128 Can be used to transfer a small amount of data without the need for <a href="#">AT^SISW</a> . The content of the "hcContent" string will only be sent if "hcContentLen" = 0. To transmit a larger amount of data, the "hcContentLen" string shall set to a non-zero value. In this case, the "hcContent" string will be ignored, and data transmission from the client to the server is done with <a href="#">AT^SISW</a> .
"hcContentLen"	Definition of payload length. If this parameter is non-zero value the payload is then defined by the <a href="#">AT^SISW</a> command. Length: 0...128 If "hcContentLen" = 0 then the data given in the "hcContent" string will be posted. If "hcContentLen" > 0 then the <a href="#">AT^SISW</a> command will be used to send data from the client to the server. If this parameter is not set, default hcContentLen 0 is used.
"topicFilter"	Definition of topic filter list for subscribing or unsubscribing data. Each topicFilter separated by a semicolon, the max number is 5.
"topicQos"	Definition of maximum Qos level at which the server can send application messages to the client for each topic filter from topicFilter list. Each topicQos separated by a semicolon, the max number is 5. If all topics should have same Qos then only one needs to be defined.



---

`<srvParmValue-alphabet>(str)(+CSCS)`

Supported string parameter character set selections in `<srvParmValue>` for `<srvParmTag>` value "alphabet".

[“0”]                                   The applicable character set is determined by the current setting of [AT+CSCS](#).

“1”                                    International Reference Alphabet (IRA, seven bit ASCII).

## 8.5 AT^SISI Internet Service Information

AT^SISI serves to monitor the status of Internet service profiles defined with AT^SISS.

The AT^SISI read command requests the current status of all defined Internet service profiles.

The AT^SISI write command requests the status of the specified Internet service profile. If the specified service profile is not defined yet, "+CME ERROR" is returned.

### Syntax

Test Command

```
AT^SISI=?
```

Response(s)

```
[^SISI: (list of defined <srvProfileId>s)]  
OK
```

Read Command

```
AT^SISI?
```

Response(s)

If response is related to service profile(s) created / opened on this interface:

```
[^SISI: <srvProfileId>, <srvState>, <rxCount>, <txCount>, <ackData>, <unackData>]  
[^SISI: ...]
```

If response is related to service profile(s) created / opened on other interface(s):

```
[^SISI: <srvProfileId>, <srvParmTag>value "srvType"]  
[^SISI: ...]
```

OK

ERROR

Write Command

```
AT^SISI=<srvProfileId>
```

Response(s)

```
^SISI: <srvProfileId>, <srvState>, <rxCount>, <txCount>, <ackData>, <unackData>
```

OK

ERROR

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

`<srvProfileId>`<sup>(num)</sup>

Internet service profile identifier as defined by `AT^SISS (<srvProfileId>)`.

0...4

`<srvState>`<sup>(num)</sup>

Internet service state (for details see `AT^SISO` parameter `<srvState>`).

2	Allocated
3	Connecting
4	Up
5	Closing
6	Down

`<rxCount>`<sup>(num)</sup>

Number of bytes received via `AT^SISR` or `AT^SIST` since last successful `AT^SISO` write command. This is the same value as displayed by the `AT^SISO` read command with the parameter `<rxCount>`.

`<txCount>`<sup>(num)</sup>

Number of bytes sent via `AT^SISW` or `AT^SIST` since last successful `AT^SISO` write command. This is the same value as displayed by the `AT^SISO` read command with the parameter `<txCount>`.

`<ackData>`<sup>(num)</sup>

Number of data bytes already sent and acknowledged at TCP layer. Value 0 indicates that no sent data is acknowledged yet.

Parameter is not applicable to Socket with UDP. For these services the counter is always set to 0.

`<unackData>`<sup>(num)</sup>

Number of data bytes already sent but not yet acknowledged at TCP layer. A value 0 indicates that all sent data is already acknowledged.

This is the same value as displayed in the response of the `AT^SISW` write command with the parameter `<unackData>`.

Parameter is not applicable to Socket with UDP. For these services, the counter is always set to 0.

### Note

- If a service is in state "Down", the responses for `<rxCount>`, `<txCount>`, `<ackData>` and `<unackData>` are the last known values for the service in the states "Connecting" and "Up".

## 8.6 AT^SISO Internet Service Open

The `AT^SISO` write command starts the Internet session configured by the service profile. All further steps needed to control the session depend on whether you are using URC mode or polling mode. The `AT^SISO` read command returns the current status of all Internet services.

URC mode:

If the service opens successfully, the URCs "`^SISW`" and "`^SISR`" will trigger the action to follow, either writing data with `AT^SISW` or reading data with `AT^SISR`. To terminate an upload data stream set the `<eodFlag>` within the last `AT^SISW` command. If the URCs notify that a data transfer has been successful ("`^SISR: x, 2`" or "`^SISW: x, 2`"), the service can be closed with `AT^SISC`. If a special event occurs, e.g. an error or a warning, after opening or while using a service then the URC type "`^SIS`" will be delivered.

Polling mode:

After opening the service, all progress information needed to control the session shall be explicitly requested by the host application. The particular sequence of AT commands varies depending on the service used. The following list summarizes, by way of an example, the steps normally involved in managing an upload or download job.

- Upload (Socket):
  - Enter the `AT^SISO` command, e.g. `AT^SISO=4`.
  - Enter `AT^SISW`, specify `<reqWriteLength>`, e.g. `AT^SISW=4,20`. Check resulting response for `<cnfWriteLength>`. Optionally, check error with `AT^SISE`, e.g. `AT^SISE=4`. If necessary, repeat the sequence several times.
  - Enter last `AT^SISW` command and enable `<eodFlag>`, e.g. `AT^SISW=4,0,1`.
  - For Socket service: Query available data with `AT^SISR`, e.g. `AT^SISR=4,1358`.
  - Check service state with `AT^SISI`, e.g. `AT^SISI=4`, if necessary several times until `<srvState>=6` ("Down").
  - Check error with `AT^SISE`, e.g. `AT^SISE=4`.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.
- Download (Socket):
  - Enter the `AT^SISO` command, e.g. `AT^SISO=4`.
  - Enter `AT^SISR`, specify `<reqReadLength>`, e.g. `AT^SISR=4,1000`. Check resulting response for `<cnfReadLength>`. If necessary, repeat the sequence several times until `<cnfReadLength>=-2` (end of data) or ERROR.
  - Check error with `AT^SISE`, e.g. `AT^SISE=4`.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.
- Socket service (upload and download possible in one session):
  - Recommended: Set `AT+CMEE=2` to enable extended error text.
  - Enter the `AT^SISO` command, e.g. `AT^SISO=4`.
  - Enter `AT^SISR` or `AT^SISW`, specifying `<reqReadLength>` or `<reqWriteLength>`, e.g. `AT^SISR=4,20` or `AT^SISW=4,20`. Check resulting response for `<cnfReadLength>` or `<cnfWriteLength>`. If necessary, repeat the sequence several times.
  - Check error with `AT^SISE`, e.g. `AT^SISE=4`.
  - For write action: Enter last `AT^SISW` command and enable `<eodFlag>`, e.g. `AT^SISW=4,0,1`.
  - For Socket service: Query available data with `AT^SISR`, e.g. `AT^SISR=4,1358`.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.
- Transparent TCP client or Transparent UDP client:
  - Enter the `AT^SISO` command, e.g. `AT^SISO=4`.
  - Enter `AT^SIST` to enter transparent data mode. After communication has finished, return to AT command mode via `+++` escape sequence.
  - Check error with `AT^SISE`, e.g. `AT^SISE=4`.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.

## 8.6 AT^SISO

- MQTT SUBSCRIBE(Using static setup):
  - Connect and use static setup defined by `AT^SISS`, enter the `AT^SISO` command, e.g. `AT^SISO=4`.
  - Enter `AT^SISE`, polling for the oldest unread event (URC), e.g. `AT^SISE=4,1`. Check resulting response, if necessary periodic read to monitor event (URC).
  - Read available data with `AT^SISR`, e.g. `AT^SISR=4,1358`. when received PUBLISH request from server with subscribed topic.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.
- MQTT PUBLISH(Using static setup):
  - Connect and use static setup defined by `AT^SISS`, enter the `AT^SISO` command, e.g. `AT^SISO=4`.
  - Enter `AT^SISE`, polling for the oldest unread event (URC), e.g. `AT^SISE=4,1`. Check resulting response, if necessary periodic read to monitor event (URC).
  - Write data to be send in payload of PUBLISH request, enter `AT^SISW`, specify `<reqWriteLength>`, e.g. `AT^SISW=4,20`. Check resulting response for `<cnfWriteLength>`. If necessary, repeat the sequence several times.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.
- MQTT SUBSCRIBE or PUBLISH(Using dynamic setup, SUBSCRIBE and PUBLISH possible in one session):
  - Connect and use static connection setup defined by `AT^SISS`, enter the `AT^SISO` command, e.g. `AT^SISO=4,2`.
  - Enter `AT^SISE`, polling for the oldest unread event (URC), e.g. `AT^SISE=4,1`. Check resulting response, if necessary periodic read to monitor event (URC).
  - To setup dynamic request parameters:  
request parameters are defined by `AT^SISD`, e.g. `AT^SISD=4,"setparam","cmd","subscribe"`, and trigger for sending request is done by `AT^SISU`, e.g. `AT^SISU=4`.  
request parameters are defined and request itself is triggered by one `AT^SISU` command, e.g. `AT^SISU=4,"publish","2:Test:0:hello:0"`.
  - Enter `AT^SISR` or `AT^SISW`, specifying `<reqReadLength>` or `<reqWriteLength>`, e.g. `AT^SISR=4,20` or `AT^SISW=4,20`. Check resulting response for `<cnfReadLength>` or `<cnfWriteLength>`. If necessary, repeat the sequence several times.
  - Close service with `AT^SISC`, e.g. `AT^SISC=4`.

**Syntax**

Test Command

`AT^SISO=?`

Response(s)

OK

Read Command

`AT^SISO?`

Response(s)

```
^SISO: <srvProfileId>, <srvType>[, <srvState>, <socketState>, <rxCount>, <txCount>,
<locAddr>, <remAddr>]
```

[`^SISO: ...`]

OK

Write Command

`AT^SISO=<srvProfileId>[, <optParam>]`

Response(s)

OK

ERROR

+CME ERROR: `<err>`

PIN Last

± -

### Parameter Description

<srvProfileId><sup>(num)</sup>

<srvProfileId> 0 ... 4 specified with AT^SISS.

<srvType><sup>(str)</sup>

The type of the internet service as specified by the AT^SISS parameter <srvParmValue-srvType>.

<srvState><sup>(num)</sup>

Internet service state.

Please note that the state of an Internet service may influence the state of the serial DCD line. For details see description of value 2 for parameter <value> of command AT&C.

2	Allocated Service profile resources are allocated, i.e. at least the service type has been set (parameter <srvParmTag>, value "srvType" of AT^SISS). The service is not opened, but ready for configuration.
3	Connecting State after opening a service with AT^SISO where the connection is being established. If connection setup is successful, the service proceeds to the state "4" (Up) and one of the URCs "^SISW" and "^SISR" may follow. If connection setup is not successful, the "^SIS" URC may appear and the service enters <srvState> 6 (Down).
4	Up The service performs its purpose. The data transfer process is the major function at this state.
5	Closing Internet Service is closing the network connection.
6	Down This state is entered if - the service has successfully finished its session (see note on Socket), - the remote peer has reset the connection or - the IP connection has been closed because of an error (see note below on service or network errors).  If a service is in this state, make sure to close it with AT^SISC before reopening it.

<socketState><sup>(num)</sup>

Socket state identifier.

1	Socket not assigned, i.e. no TCP/UDP connection active.
2	Socket assigned as CLIENT.
3	Socket assigned as UDP Endpoint.

<rxCount><sup>(num)</sup>

Number of bytes received via AT^SISR or AT^SIST since the last successful AT^SISO write command.

<txCount><sup>(num)</sup>

Number of bytes sent via AT^SISW or AT^SIST since the last successful AT^SISO write command.

`<locAddr>`<sup>(str)</sup>

Recently used local IPv4 address in a dotted-four-byte format and TCP port separated by colon, e.g. "192.60.10.10:80", or recently used local IPv6 address in colon format enclosed in square brackets, e.g. "[FE80::2]" and TCP port separated by colon, e.g. "[FE80::2]:80".

`<remAddr>`<sup>(str)</sup>

Remote IPv4 address in the dotted-four-byte format and TCP port, separated by colon, e.g. "192.60.10.10:80", or remote IPv6 address in the colon format enclosed in square brackets, e.g. "[FE80::2]") and TCP port separated by colon, e.g. "[FE80::2]:80".

If the Socket service is configured to UDP Endpoint, a remote IP address cannot be assigned, and a default value will be shown instead.

`<optParam>`<sup>(num)</sup>

Distinguish between static and dynamic setup, only applicable to MQTT.

[0]	or none Connect and use static setup defined by <a href="#">AT^SISS</a> .
2	Connect and use dynamic setup defined by <a href="#">AT^SISD</a> .

### Notes

- If a service or network error occurs during an IP session and the "[^SIS](#)" URC or the [AT^SISE](#) command display an error message, the service enters `<srvState>=6`, i.e. the IP connection is closed. In this case, the service still allows reading the data stored in the buffer, but writing data is denied. After reading, close the service and open it again.
- As in polling mode, no error URCs are available, you are advised to integrate the commands [AT^SISI](#) and [AT^SISE](#) into the command sequences for upload and download jobs. So, the [AT^SISO](#) command may be followed by [AT^SISI](#) to check that the service has entered the `<srvState>=4` before sending [AT^SISR](#), [AT^SISW](#) or [AT^SIST](#). This helps you to detect errors even though the "[^SIS](#)" URCs are disabled. A typical example is that the service fails to be opened because the service profile contains a wrong destination address. In such a case, after opening with [AT^SISO](#) and responses OK, the service quickly enters the states `<srvState>=2, 4` and `6`, with an error being returned no earlier than after sending the first [AT^SISR](#) or [AT^SISW](#) command. In URC mode, an error URC, such as "Host not found" would be received in this situation, but in polling mode, the only way to detect the state is using [AT^SISI](#) or, alternatively, [AT^SISO](#).

## 8.7 AT^SISC Internet Service Close

The `AT^SISC` write command closes the TCP/IP connection to the remote peer which was opened with `AT^SISO`. All resources are released; all status information parameters, e.g. `<srvState>`, `<unackData>`, `<ackData>`, `<rxCount>` and `<txCount>` counters are reset. Only the initial configuration settings are preserved. The service can be restarted any time, using the same configuration or a new one.

### Syntax

Test Command

```
AT^SISC=?
```

Response(s)

```
OK
```

Write Command

```
AT^SISC=<srvProfileId>
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
± -
```

### Parameter Description

`<srvProfileId>`<sup>(num)</sup>

0...4

`<srvProfileId>` specified with `AT^SISO`.



## 8.8 AT^SISR Internet Service Read Data

The `AT^SISR` command either triggers a read operation (download) or queries the number of bytes in the internal buffer.

The "`^SISR: x, 1`" URC is also used to report status changes of the Transparent TCP or Transparent UDP service while the client is in AT command mode.

The `AT^SISR` write command may return the following errors:

- "+CME ERROR: operation temporary not allowed" e.g. if the service has no network resources allocated.
- "+CME ERROR: operation not allowed" e.g. if the service is not configured.

### Syntax

Test Command
<code>AT^SISR=?</code>
Response(s)
OK
Write Command
<code>AT^SISR=&lt;srvProfileId&gt;, &lt;reqReadLength&gt;</code>
Response(s)
<code>^SISR: &lt;srvProfileId&gt;, &lt;cnfReadLength&gt;[, &lt;remainUdpPacketLength&gt;[, &lt;Udp_RemClient&gt;]]</code>
Number of data bytes are sent as specified by <code>&lt;cnfReadLength&gt;</code> . If peek operator was used, no data bytes are sent.
OK
ERROR
+CME ERROR: <code>&lt;err&gt;</code>
PIN Last
± +

### Unsolicited Result Code

`^SISR: <srvProfileId>, <urcCauseId>`

Data availability status of the Internet service configured with `AT^SISS` has changed. The URC is issued when:

- data is available after opening an Internet service or
- less data was confirmed (in `<cnfReadLength>`) than requested (with `<reqReadLength>`) during the last "Read Data" operation and new data are available.

The URC is disabled in polling mode. See `AT^SCFG`, parameter "Tcp/WithURCs", `<tcpWithUrc>`.

The maximum data that can be cached is less than 4096 bytes, depending on the remaining memory size.

### Parameter Description

<code>&lt;srvProfileId&gt;</code> <sup>(num)</sup>	
<code>&lt;srvProfileId&gt;</code> 0 ... 4 specified with <code>AT^SISS</code> .	
<code>&lt;reqReadLength&gt;</code> <sup>(num)</sup>	
0	Peek Operator: Query number of received bytes within internal buffers. The behaviour of the peek operator depends on the selected Internet service, i.e. it may not be supported by all IP Services. For "Socket" service configured for UDP the size of the next available UDP packet is returned.
1...1358	Requested number of data bytes to be read via the Internet service specified in <code>&lt;srvProfileId&gt;</code> .

**<cnfReadLength><sup>(num)</sup>**

- 2 Indicates the end of data. Data transfer has been finished (all data have been read) and the service can be closed with [AT^SISC](#).
- 0 Indicates that no further data is available at the moment.
- >0 Number of available data bytes. The range is determined by [<reqReadLength>](#):  
 If [<reqReadLength>](#) was greater than 0, then [<cnfReadLength>](#) may be less or equal to the value requested with [<reqReadLength>](#).  
 If [<reqReadLength>](#) equals 0 (peek operator), the value indicated by [<cnfReadLength>](#) may be greater than 1358.

**<urcCauseId><sup>(num)</sup>**

Indicates whether or not the data is available for reading with [AT^SISR](#) and [AT^SIST](#).

- 1 *Meaning of value "1" for all services except Transparent TCP or Transparent UDP service:*  
 Data is available and can be read by sending the [AT^SISR](#) command. The URC appears when fewer data was confirmed (in [<cnfReadLength>](#)) than requested (with [<reqReadLength>](#)) during the last "Read Data" operation and new data is available.  
  
*Meaning of value "1" for Transparent TCP or Transparent UDP service:*  
 If the Transparent TCP or Transparent UDP service is in AT command mode the "[^SISR: x, 1](#)" URC notifies the client that the server has sent new data. In this case the client shall enter the [AT^SIST](#) command to go to transparent access mode and read the data.
- 2 End of data. Data transfer has completed (all data read). The service can be closed with [AT^SISC](#).

**<remainUdpPacketLength><sup>(num)</sup>**

Optional third parameter of the [AT^SISR](#) write command response displayed only if the Socket service uses the UDP protocol.

The reception of each datagram must be completed before the next datagram can be received. This may be a problem if the buffer of the host application is limited and not designed to handle the maximum packet size of 1358 bytes. To compensate this, the host is advised to request, via [<reqReadLength>](#), an amount less or equal its maximum buffer capacity and wait for the resulting [AT^SISR](#) write command response with parameter [<remainUdpPacketLength>](#). The benefit of this approach is that the host may properly receive all parts of a UDP datagram, as after each received data part the number of remaining bytes is synchronized, until reading the datagram is finished.

If the currently read datagram is smaller than the number of bytes requested by the host, the [<remainUdpPacketLength>](#) parameter is omitted.

- .
- 0 Indicates that all bytes of the current UDP datagram are read.
- (max. data size)-1...1 Indicates that the currently read UDP datagram is not yet complete. The displayed value is the remaining number of bytes.  
[<remainUdpPacketLength>](#) is unequal 0 until reading all parts of the current datagram is finished.

**<Udp\_RemClient><sup>(str)</sup>**

Service specific parameter of the [AT^SISR](#) write command response. If the Socket service is configured as UDP endpoint, this parameter shows the IP address and the UDP port of the remote host that has sent the current UDP data. See example in Section [8.16.4, UDP Scenario](#).

## Notes

- The URC "[^SISR: x, 1](#)" notifies that a package data is available for reading. If this package data is not read out, the notification of the next package data will not be displayed. 4096 bytes data can be cached at most. If the newly received data is larger than the remaining capacity, it will be discarded.
- ENS22-E can't handle packet after IP fragment. For this reason, the data sent to ENS22-E needs to be less than the Path MTU.
- In the secure connection, the maximum received data in a single packet is 2560 bytes.
- In MQTT service, if the package data is larger than 4096 bytes, the client will be disconnected.

### 8.8.1 Example: Socket Host Reads Small Amounts of UDP Data Packets (URC Mode)

This section applies only to the Socket service using the UDP protocol. The example shows how to read UDP packets if the buffer of the host application can handle only a few bytes. There are two datagrams available for reading.

Buffer size: 6 bytes

Datagram A = 18 bytes (content "THIS\_IS\_DATAGRAM\_A")

Datagram B = 3 bytes (content "NEW")

<code>^SISR: 0, 1</code>	Service profile 0 is configured for Socket service and UDP. The URC notifies the host that data is available for reading.
<code>at^sizr=0,6</code>	The host requests to read 6 bytes.
<code>^SISR: 0, 6, 12</code>	The first 6 bytes of datagram A are confirmed and transmitted. Another 12 bytes are still available.
<code>THIS_I</code>	
<code>OK</code>	
<code>at^sizr=0,6</code>	The host requests to read the next part of 6 bytes.
<code>^SISR: 0, 6, 6</code>	The next 6 bytes of datagram A are confirmed and transmitted. Another 6 bytes are still available.
<code>S_DATA</code>	
<code>OK</code>	
<code>at^sizr=0,6</code>	The host requests to read the next part of 6 bytes.
<code>^SISR: 0, 6, 0</code>	Last part of datagram A was received, no remainder is left.
<code>GRAM_A</code>	
<code>OK</code>	
<code>at^sizr=0,6</code>	Again, the host requests to read 6 bytes. This time, the request refers to datagram B which has only 3 bytes. As the read datagram is smaller than the size requested by the host, the response does not include the <a href="#">&lt;remainUdpPacketLength&gt;</a> .
<code>^SISR: 0, 3</code>	3 bytes are confirmed and transmitted. Datagram B was properly received.
<code>NEW</code>	
<code>OK</code>	

## 8.9 AT^SISW Internet Service Write Data

`AT^SISW` triggers a write operation (upload) and queries the amount of data already sent, but not acknowledged at the TCP layer.

`AT^SISW` write command may return the following errors:

- "+CME ERROR: operation temporary not allowed" e.g. if the service has no network resources allocated.
- "+CME ERROR: operation not allowed" e.g. if the service is not configured.

### Syntax

Test Command
<code>AT^SISW=?</code>
Response(s)
OK
Write Command
<code>AT^SISW=&lt;srvProfileId&gt;, &lt;reqWriteLength&gt;[, &lt;eodFlag&gt;[, &lt;Udp_RemClient&gt;]]</code>
Response(s)
<code>^SISW: &lt;srvProfileId&gt;, &lt;cnfWriteLength&gt;, &lt;unackData&gt;</code> Number of data bytes as specified by <code>&lt;cnfWriteLength&gt;</code> .
OK
ERROR
+CME ERROR: <code>&lt;err&gt;</code>
PIN Last
± +

### Unsolicited Result Code

`^SISW: <srvProfileId>, <urcCauseId>`

The URC indicates the following status changes:

- After opening a TCP or UDP client or a MQTT client with `AT^SISO`, the "`^SISW`" URC indicates that the service is ready to accept data. After this initial "`^SISW`" URC, the TCP or UDP Client can proceed in Transparent Mode or Non-Transparent Mode, or the MQTT client can proceed. The initial "`^SISW`" URC is not issued after a UDP Endpoint was opened, because the "`^SIS`" URC will be issued.
- During `AT^SISW` write cycles the "`^SISW`" URC indicates that the service is ready to accept new data, or that data transfer has finished successfully.

The URC is disabled in polling mode. See `AT^SCFG`, parameter "Tcp/WithURCs", `<tcpWithUrc>`.

### Parameter Description

<code>&lt;srvProfileId&gt;</code> <sup>(num)</sup>
<code>&lt;srvProfileId&gt;</code> 0 ... 4 specified with <code>AT^SISS</code> .
<code>&lt;reqWriteLength&gt;</code> <sup>(num)</sup>
0...1358
Specifies the number of bytes to be sent with <code>AT^SISW</code> or, if set to 0, requests the amount of data already sent with <code>AT^SISW</code> but not yet acknowledged:
<ul style="list-style-type: none"> <li>• Parameter <code>&lt;reqWriteLength&gt;</code> may be 1...1358 bytes to specify the amount of data to be sent with <code>AT^SISW</code>.</li> </ul>

## 8.9 AT^SISW

- If parameter `<reqWriteLength>` equals 0, `AT^SISW` does not expect any data, but a normal query is performed. This allows the application to explicitly request, in particular at the end of an upload job, the amount of unacknowledged data at the TCP layer indicated by `<unackData>`.
- If Socket service is selected with UDP protocol
  - `<reqWriteLength>=0` can be used to send an empty UDP packet;
  - it is recommended that the size of each data packet be limited to 1358 bytes.
- If MQTT service is selected, the size of each data packet is limited to 512 bytes.

`<eodFlag>`<sup>(num)</sup>

End of data indication flag.

Note for UDP Endpoint service: Value 0 must be set before `<Udp_RemClient>` parameter although `<eodFlag>` is not effective for UDP Endpoint.

[0]	No end of data and no release assistance indication. Other data may follow to be transmitted via the Internet service.
1	End of data is signalled to the Internet Service. Further <code>AT^SISW</code> write commands return an error response. However, reading data may be possible. The <code>&lt;eodFlag&gt;</code> is effective only if the <code>&lt;reqWriteLength&gt;</code> equals <code>&lt;cnfWriteLength&gt;</code> , in particular if the <code>&lt;reqWriteLength&gt;</code> equals 0. This value is effective for Socket with TCP protocol.
2	Represents release assistance indication: no further uplink or downlink data transmission is expected. This value is effective for Socket with UDP protocol.
4	Represents release assistance indication: only a single downlink data transmission (e.g. acknowledgement or response to uplink data) and no further uplink data transmission subsequent to the uplink data transmission is expected. This value is effective for Socket with UDP protocol.

`<Udp_RemClient>`<sup>(str)</sup>

Service-specific parameter of the `AT^SISW` write command. If the Socket service is configured as UDP endpoint, this parameter is mandatory, and specifies the IP address (in dotted format) and the UDP port of the remote host. For each write request another IP address and UDP port may be given. See example in Section 8.16.4, [UDP Scenario](#).

`<cnfWriteLength>`<sup>(num)</sup>

0...1358	Confirmed number of data bytes which can be transmitted via the Internet service configured in <code>&lt;srvProfileId&gt;</code> . This number may be less or equal to the value requested with <code>&lt;reqWriteLength&gt;</code> . The application has to deliver exactly the number of bytes indicated by <code>&lt;cnfWriteLength&gt;</code> . A 0 value means that no data can be written at this time, i.e. it serves as a flow control mechanism.
----------	---

`<unackData>`<sup>(num)</sup>

Number of data bytes already sent but not yet acknowledged at the TCP layer.

The value is constantly changing until the entire upload job has completed. If the value equals 0 all data sent so far is acknowledged.

`<unackData>` includes the `<cnfWriteLength>` value of the pending write operation. Therefore, the very first write operation of an upload job returns identical values for `<cnfWriteLength>` and `<unackData>`.

This mechanism allows the host application to easily verify whether the remote host has successfully received the data.

Parameter is not applicable to Socket with UDP protocol. For these services the counter is always set to 0.

<urcCauseId><sup>(num)</sup>

- |   |   |
|---|---|
| 1 | The service is ready to accept new user data.   |
| 2 | Data transfer has been finished successfully and Internet service may be closed without loss of data. |

### Notes

- Do not send any characters after `AT^SISW` command line termination(`ATS3`) until "`^SISW`" URC is received. This is necessary to avoid that any characters related to the AT command can be interpreted as being part of the data stream to be transferred. Also refer to Section 1.4, [AT Command Syntax](#).
- Enter `AT^SISW`, specify `<reqWriteLength>`, e.g. `AT^SISW=4,20`, then received response that 20 bytes can be sent(indicated by `<cnfWriteLength>`), at this time, the data will be sent when your entered data size is reached 20, and the excess input will be truncated.
- If you want to terminate an upload data stream, after the last `AT^SISW` command, you will receive "`^SISW:x, 1`", then the service can be closed with `AT^SISC`.

## 8.9.1 Usage of parameter <eodFlag>

Service type	
Socket	Parameter is optional for Socket services.  This parameter <code>&lt;eodFlag&gt;</code> is set value 1, after the service accepts the end-of-data flag, the service state changes to "Closing". No further user data is accepted. After releasing the TCP link, the service remains in state "Down".

## 8.10 AT^SIST

## 8.10 AT^SIST Enter Transparent Mode

AT^SIST activates / deactivates Transparent Mode.

After entering the AT^SIST command, the CONNECT response notifies that the service has entered Transparent Mode and is ready to transparently send and receive payload data over the AT command interface.

There are several ways to quit the Transparent Mode and revert to AT Command Mode:

- +++ escape sequence
- DTR ON-OFF transitions (see AT&D for configuring DTR line)
- "etxChar" parameter set within the server address (see AT^SISS, <srvParmTag> "address" for Socket service URLs).

Switching back and forth between both modes does not affect the underlying socket and has no influence on the service <srvState>.

If the service is in AT Command Mode the "^SISR: x, 1" URC appears once to indicate that the remote side has sent new data. In this case, the client shall enter AT^SIST to go to Transparent Mode and read out the new data.

If the connection is released by the remote side or the network is unavailable, the following applies:

- If the service is in Transparent Mode (CONNECT mode): UE returns to AT Command Mode, issues a specific "NO CARRIER" result code and goes to <srvState> 6 ("Down") in case of UDP Client. The "NO CARRIER" result code starts with 0x10 (DLE) and 0x04 (EOT) and ends with the command line termination character (ATS3), followed by the response formatting character (ATS4).
- If the service is in AT Command Mode the "^SISR: x, 1" URC appears to indicate that the state of the service has changed. In this case, the client shall go to Transparent Mode with AT^SIST in order to read remaining data (if any) and get the "NO CARRIER" result code. After this, the service changes to <srvState> 6 ("Down"). Otherwise, if the client does not go to Transparent Mode after receiving "^SISR: x, 1" URC the service remains in <srvState> 4 ("Up") even though the remote side has shut down. In some cases, depending on the remote side and the network, it is possible that "^SIS" URCs may be reported, for example, "^SIS" URCs with <urcInfoId> 15 ("Remote host has reset the connection") and 19 ("Socket has been shut down"). See Section 8.15, Internet Service URC "^SIS" for a list of possible URCs.
- In all the above scenarios, the Transparent service shall be closed with AT^SISC. Trying to return to Transparent Mode with AT^SIST would only end up with "NO CARRIER" once again and is not recommended.
- Use AT^SISO and AT^SISI to check the <srvState>.

### Syntax

Test Command

```
AT^SIST=?
```

Response(s)

```
OK
```

Write Command

```
AT^SIST=<srvProfileId>
```

Response(s)

```
CONNECT
```

(indicates that UE has entered Transparent Mode)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- +
```

---

### **Parameter Description**

<srvProfileId><sup>(num)</sup>

<srvProfileId> 0 ... 4 specified with AT^SIST.

### **Notes**

- There can be only one Socket service in Transparent Mode at the same time.
- AT^SIST is only usable on the ASC0 interface.
- Transparent mode will block module goes into sleep mode and suspend mode.



## 8.11 AT^SISD Internet Service MQTT Dynamic Setup Profile

AT^SISD serves to set up the necessary parameters in the MQTT service profiles. Service profiles can then be used to control a data link in conjunction with AT^SISU and AT^SISE.

Command could be used only after connection is opened with AT^SISO parameter <optParam>=2. Each request parameter can be changed multiple time during active connection.

The AT^SISD write command specifies the parameters for MQTT service profile identified by <srvProfileId>.

To change the settings the write command needs to be executed for each single <dynParamTag>. All profile parameters set with AT^SISD are volatile.

The list below shows which <dynParamTag> parameters apply to MQTT service and which of them are mandatory (= m) or optional (= o).

**Table 8.4:** Applicability of AT^SISD <dynParamTag> values

<dynParamTag>	Mandatory or optional
<i>Publish</i>	
"Cmd"	m
"Qos"	o
"Topic"	m
"Retain"	o
"hcContent"	o
"hcContLen"	o
<i>Subscribe</i>	
"Cmd"	m
"topicFilter"	m
"topicQos"	o
<i>Unsubscribe</i>	
"Cmd"	m
"topicFilter"	m

### Syntax

Test Command

AT^SISD=?

Response(s)

OK

Write Command

Set parameters defined by <dynParamTag> to value <dynParamValue>

AT^SISD=<srvProfileId>, "setParam", <dynParamTag>, <dynParamValue>

Response(s)

OK

ERROR

+CME ERROR: <err>

Write Command

Get current value of dynamic parameters defined by `<dynParamTag>`

AT^SISD=`<srvProfileId>`, "getParam", `<dynParamTag>`

Response(s)

^SISD: `<srvProfileId>`, `<dynParamTag>`, `<dynParamValue>`

OK

ERROR

+CME ERROR: `<err>`

Write Command

Set all dynamic parameters for service to default value

AT^SISD=`<srvProfileId>`, "cleanParam"

Response(s)

OK

ERROR

+CME ERROR: `<err>`

Write Command

Get all dynamic parameters of service

AT^SISD=`<srvProfileId>`

Response(s)

^SISD: `<srvProfileId>`, `<dynParamTag>`, `<dynParamValue>`

^SISD: ...

OK

ERROR

+CME ERROR: `<err>`

PIN Last

- -

### Parameter Description

`<srvProfileId>`<sup>(num)</sup>

Internet service profile identifier as defined by AT^SISS `<srvProfileId>`.

0...4

`<dynOption>`<sup>(str)</sup>

Parameter `<dynOption>` define action in dynamic setup of request parameters.

“setParam” Set parameter defined by `<dynParamTag>` to value `<dynParamValue>`.

“getParam” Get current value of parameter defined by `<dynParamTag>`.

“cleanParam” Set all request parameters for service to its default value.

“none” If `<dynOption>` not be set, get all dynamic parameters of service.

`<dynParamTag>`<sup>(u)</sup>

MQTT service profile parameter.

cmd refer to `<mqttRequestParameter>`.

Qos refer to `<mqttRequestParameter>`.

Topic refer to `<mqttRequestParameter>`.

Retain refer to `<mqttRequestParameter>`.

---

hcContent	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
hcContLen	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
topicFilter	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .
topicQos	refer to <a href="#">&lt;mqttRequestParameter&gt;</a> .

`<dynParamValue>`<sup>(str)(+CSCS)</sup>

Parameter value; type and supported content depend on related [<dynParamTag>](#).

## 8.12 AT^SISU MQTT Dynamic Send Request

AT^SISU serves to setup and send MQTT request.

Command could be used only after connection is opened with AT^SISO parameter <optParam>=2. Each request parameter can be changed multiple time during active connection.

The AT^SISU write command with only first parameter <srvProfileId> is used to trigger send of service request that was defined by dynamic parameters with command AT^SISD.

The AT^SISU write command with second parameter is used to define dynamic request.

All profile parameters set with AT^SISU are volatile.

### Syntax

<p>Test Command</p> <pre>AT^SISU=?</pre> <p>Response(s)</p> <pre>OK</pre>
<p>Write Command</p> <pre>AT^SISU=&lt;srvProfileId&gt;[, &lt;dynReqType&gt;[, &lt;dynReqValue&gt;]]</pre> <p>Response(s)</p> <pre>OK ERROR +CME ERROR: &lt;err&gt;</pre>
<p>PIN Last</p> <pre>- -</pre>

### Parameter Description

<srvProfileId><sup>(num)</sup>

Internet service profile identifier as defined by AT^SISS <srvProfileId>.

0...4

<dynReqType><sup>(str)</sup>

Parameter <dynReqType> define type of service specific request.

“publish“	The MQTT client sends data by PUBLISH control packet request to server.
“subscribe“	The MQTT client that wants to receive data must subscribe to it by SUBSCRIBE control packet.
“unsubscribe“	The MQTT client that doesn't want to receive subscribed data anymore must unsubscribe specified topic filter by sending UNSUBSCRIBE control packet.

<dynReqValue><sup>(str)(+CSCS)</sup>

Parameter <dynReqValue> define all request parameters separated by ":" character.

- PUBLISH: Qos:Topic:Retain:hcContent:hcContLen
- SUBSCRIBE: topicFilter:topicQos
- UNSUBSCRIBE: topicFilter

Parameter value and supported content depend on related <dynParamTag>.

## 8.13 AT^SISX Internet Service Execution

The `AT^SISX` command can be used to send ICMP (Internet Control Message Protocol) Echo Requests to a target IP address or DNS host name (also referred to as Ping command).

If the `AT^SISX` write command is executed for Ping, then the command returns one row for every ICMP Echo Request. The last Ping response line is followed by two lines to report a statistic calculated from the Echo Request responses. After starting the ping command, the AT channel is blocked until all ICMP requests are executed. To stop the periodic ping early, send any character (user break).

### Syntax

```

Test Command
AT^SISX=?
Response(s)
^SISX: "Ping", (list of supported <conProfileId>s), (range of supported <request>s), (range of supported <timelimit>s)
OK

Write Command
AT^SISX=<service>, <conProfileId>, <address>[, <request>[, <timelimit>]]
Response(s)
In case of <pingInfoType>= 1: ICMP Echo Requests (1 - 30 rows)
^SISX: "Ping", <pingInfoType>, <conProfileId>, <ip-address>, <roundTripTime>
[...]

In case of <pingInfoType>= 2: Packet statistic (1 row)
^SISX: "Ping", <pingInfoType>, <conProfileId>, <sent>, <received>, <lost>, <lostPercent>

In case of <pingInfoType>= 3: Time statistic (1 row)
^SISX: "Ping", <pingInfoType>, <conProfileId>, <minRTT>, <maxRTT>, <meanRTT>

OK
ERROR
+CME ERROR: <err>
    
```

PIN Last  
 + -

### Parameter Description

`<service>`<sup>(str)</sup>

"Ping" Ping host.

`<conProfileId>`<sup>(num)</sup>

The PDP Context Identifier as defined with `AT+CGDCONT` for the PDP context definition to be used (see `<cid>`). For `AT^SISX`, only PDP Context Identifier 0 can be used. If the specified PDP context is not yet activated, then `AT^SISX` returns ERROR.

`<address>`<sup>(str)</sup>

In case of "Ping" this is the target host as FQDN or IP address (i.e. "192.168.1.3"),

`<ip-address>`<sup>(str)</sup>

The IP address of the host (i.e. "192.168.1.3").

<request><sup>(num)</sup>

[1]...30                      Number of "Ping" ICMP Echo Requests to be sent.

<timelimit><sup>(num)</sup>

200...[5000]...10000                      Amount of time, in milliseconds, to wait for an ICMP Echo Response message related to a previously sent Echo Request message.

<pingInfoType><sup>(num)</sup>

Indicate the type and format of the information given in the command responses (see syntax of the AT^SISX write command above).

- 1                      Ping status received for the ICMP Echo Request(s). Depending on parameter <request> 1 - 30 rows. Address and Round Trip Time are returned for each request.
- 2                      one row packet statistic calculated from the ping response(s) received for <pingInfoType>=1.
- 3                      one row time statistics calculated from the ping response(s) received for <pingInfoType>=1 (displayed only if at least one Echo response message was received).

<roundTripTime><sup>(num)</sup>

Round Trip Time in milliseconds. If the Echo Response message is not received within the time specified by <timelimit>, the value -1 is returned (e.g. when the bearer shuts down during command execution).

<sent><sup>(num)</sup>

Number of sent packets.

<received><sup>(num)</sup>

Number of received packets.

<lost><sup>(num)</sup>

Number of lost packets.

<lostPercent><sup>(num)</sup>

Lost packets in percent.

<minRTT><sup>(num)</sup>

Minimum Round Trip Time in milliseconds.

<maxRTT><sup>(num)</sup>

Maximum Round Trip Time in milliseconds.

<meanRTT><sup>(num)</sup>

Average Round Trip Time in milliseconds.

## 8.14 AT^SISE Internet Service Error Report

The `AT^SISE` write command returns the current error status of the specified Internet service profile. If the service profile is not defined or the value is invalid, "+CME ERROR" response is returned.

The error status is identical to the content of the "`^SIS`" URC last issued.

If a service fails, the commands `AT^SISR`, `AT^SISW` or `AT^SIST` will return an error response.

If MQTT service is in polling mode, `AT^SISE` can be used to query responses of the commands `AT^SISS`, `AT^SISO` `AT^SISD`, and `AT^SISU`. The `AT^SISE` command returns `<infoID>` and `<info>`.

### Syntax

Test Command
<code>AT^SISE=?</code>
Response(s)
OK
Write Command
<code>AT^SISE=&lt;srvProfileId&gt;[, &lt;infoMode&gt;]</code>
Response(s)
<code>^SISE: &lt;srvProfileId&gt;, &lt;infoID&gt;[, &lt;info&gt;]</code>
OK
ERROR
+CME ERROR: <code>&lt;err&gt;</code>
PIN Last
- -

### Parameter Description

<code>&lt;srvProfileId&gt;</code> <sup>(num)</sup>	Internet service profile identifier as defined by <code>AT^SISS &lt;srvProfileId&gt;</code> . 0...4
<code>&lt;infoMode&gt;</code> <sup>(num)</sup>	This parameter extends use of command to allow polling functionality for MQTT. [0] <sup>(&amp;F)(P)</sup> Return current error status in response parameter <code>&lt;infoID&gt;</code> and <code>&lt;info&gt;</code> . 1 Return oldest unread event in response parameter <code>&lt;infoID&gt;</code> and <code>&lt;info&gt;</code> , only used for polling mode.
<code>&lt;infoID&gt;</code> <sup>(num)</sup>	Reason for Internet service error. This is a subset of <code>&lt;urcInfoId&gt;</code> which can be found in Section 8.15, <a href="#">Internet Service URC "ASIS"</a> . The value 0 indicates there is no special Internet service error.
<code>&lt;info&gt;</code> <sup>(str)</sup>	Information text related to <code>&lt;infoID&gt;</code> . This is a subset of <code>&lt;urcInfoText&gt;</code> which can be found in Section 8.15, <a href="#">Internet Service URC "ASIS"</a> .

**Note**

- In particular cases, the `AT^SISE` command may deliver information and warnings. This applies the following services:  
withIServices\_FTP\_CLIENT MQTT: Informations  
If an error occurs, the information and warning messages are overwritten by error messages.



## 8.15 Internet Service URC "^SIS"

## 8.15 Internet Service URC "^SIS"

This section first describes the syntax and the parameters of the "^SIS" URC. The second part consists of tables listing the information elements delivered within the "^SIS" URC and the command response of AT^SISE.

The presentation of the "^SIS" URC containing the parameter <urcCause>=0 or <urcCause>=5 can be disabled or enabled depending on whether polling mode or URC mode is preferred. To do so, use the AT^SCFG command, parameter "Tcp/WithURCs" (refer to <tcpWithUrc>). However, if related to <urcCause>=1 or 2, the "^SIS" URC will always be delivered regardless of the setting made with AT^SCFG.

**Unsolicited Result Code**

Indicates an event, an error or an information element. The same information can be obtained via AT^SISE.

^SIS: <srvProfileId>, <urcCause>[, [<urcInfoId>][, <urcInfoText>]]

The URC may appear if an event or an error occurs, for example, after opening an Internet service with AT^SISO or any time during operation. The URC also indicates a request for a mobile terminated Internet service client connection, or a failure if a mobile terminated request is rejected. Furthermore, the URC may deliver an information element resulting from a specific command given in the service profile.

A list of possible causes can be found in the tables below. The tables contain the <urcInfoId> and, if applicable, a specific <urcInfoText>. If marked with \*) the precise text related to a <urcInfoId> varies greatly depending on the scenario and the implementation of the remote server. In these cases, only a brief explanation of the scenario can be found in the table column "Description".

**Parameter Description**

<urcCause><sup>(num)</sup>

URC cause identifier.

0	An event has occurred after opening or while using an Internet service. The event number is presented via <urcInfoId> and could be an error, a warning, an information element or a note. Optionally additional information may be supplied via <urcInfoText>.
1	Indicates that an opened Non-Transparent Listener service is receiving a connection request from a remote client. The incoming socket connection is dynamically assigned to the next free Internet service profile. In this case, the parameter <urcInfoId> inside the "^SIS" URC equals the <srvProfileId> of the dynamically assigned service profile. This ID shall be used to accept the connection request with AT^SISO or to reject it with AT^SISC.
2	Incoming Socket service client connection has failed. The client request was rejected automatically because no free Internet service profile was available.
5	Only for Non-Transparent UDP endpoint. The service is ready to use.

<urcInfoId><sup>(num)</sup>

Information identifier related to <urcCause>. See tables below for further detail. The <urcInfoId> number ranges indicate the type of information element:

0	Service is working properly.
2...504	Error, service is aborted and enters <srvState>=6 (Down).
2500...4500	Information related to progress of service.

<urcInfoText><sup>(str)</sup>

Information text related to <urcCause>. See tables below for further detail. The maximum length of an information text is 255 bytes. Any longer messages will be truncated.

## 8.15.1 Information Elements Related to the Service Application

The following table lists the information elements which may be returned by all supported services within the "^^SIS" URC and the command response of AT^SISE. It should be noted that TCP/IP socket problems may occur in all Internet service connections (Socket, Transparent TCP or Transparent UDP).

<urcInfoId>	<urcInfoText>	Description
<i>Information Elements Returned by the TCP/IP socket</i>		
2	Invalid socket descriptor	Socket error
3	Bad address specified	Socket error
4	Invalid operation	Socket error
5	No free socket descriptors	Socket error
6	The operation would block	Socket error
7	A previous attempt at this operation is still ongoing	Socket error
8	Socket is not bound	Socket error
11	Operation not supported	Socket error
12	The address is already in use	Socket error
13	The network is unavailable	Socket error
14	An established connection was aborted (transmission time-out or protocol error)	Socket error
15	Remote host has reset the connection	Socket error
16	No buffer space available	Socket error
17	The socket is already connected	Socket error
18	For TCP/IP sockets, the socket is not connected	Socket error
20	Connection timed out	Socket error
22	Remote host is unreachable	Socket error
23	An unexpected error occurred	Socket error
27	The port is already in use	Socket error
<i>General Information Elements</i>		
48	Remote peer has closed the connection	Remote peer has closed the connection unexpectedly, and no data are available. The host shall close the service. Note that the "^^SIS" URC only indicates the <urcInfoId>, but the AT^SISE command indicates both the <urcInfoId> and the <urcInfoText>.
49	Fatal: No memory is available for service action	
56	Syntax error in URL	
62	SSL unexpected Error	SSL error
63	SSL data Error	SSL error
64	SSL CTR DRBG seed failed	SSL error
65	SSL configuration default failed	SSL error

## 8.15 Internet Service URC "^^SIS"

<urcInfoId>	<urcInfoText>	Description
66	SSL setup failed	SSL error
67	SSL handshake timeout	SSL error
68	SSL handshake failed	SSL error
69	SSL peer closed	SSL error
70	SSL certificate error	SSL error
71	SSL private key error	SSL error
72	SSL PSK error	SSL error
73	SSL fatal alert	SSL error
90	PDP: internal error 1	
91	PDP: internal error 2	
94	PDP: connection failed	
96	PDP: no IP assigned	
119	PDP: no DNS received	
120	DNS: no data	
121	DNS: timeout	
125	DNS: invalid server	
126	DNS: undefined error	

## 8.15.2 Information Elements Related to MQTT Service

<urcInfoId>	<urcInfoText>	Description
<i>Error URCs</i>		
140	MQTT: publish received cannot be processed	Received publish packet payload over 4k
<i>Info URCs</i>		
500	Connection refused, version rejected	
501	Connection refused, identifier rejected	
502	Connection refused, bad user name or password	
503	Connection refused, not authorized	
504	Connection accepted, MQTT service is unavailable	
2500	Connection accepted on clean session	
2501	Connection accepted on retain session	
2505	Disconnected by the client	
2506	Disconnected by the server OR unexpected TCP client disconnection	
2510	Unsubscribed request acknowledged	
2520	Subscribed topic name	Filter topic subscribed with maximum QoS=0
2521	Subscribed topic name	Filter topic subscribed with maximum QoS=1
2522	Subscribed topic name	Filter topic subscribed with maximum QoS=2

<urcInfoId>	<urcInfoText>	Description
3488	Publish received topic name	PUBLISH received with DUP=0 and Retain=0 and Payload
3489	Publish received topic name	PUBLISH received with DUP=0 and Retain=1 and Payload
3490	Publish received topic name	PUBLISH received with DUP=1 and Retain=0 and Payload
3491	Publish received topic name	PUBLISH received with DUP=1 and Retain=1 and Payload
3492	Publish received topic name	PUBLISH received with DUP=0 and Retain=0 and NO payload
3493	Publish received topic name	PUBLISH received with DUP=0 and Retain=1 and NO payload
3494	Publish received topic name	PUBLISH received with DUP=1 and Retain=0 and NO payload
3495	Publish received topic name	PUBLISH received with DUP=1 and Retain=1 and NO payload
3520	Publish topic name	Publish acknowledged
4500	Filter topic subscribe not accepted	

## 8.16 Examples of how to Configure and Use Internet Service Profiles

Below you can find selective examples of how to configure and use Internet service profiles. To visualize the difference between the URC mode and the polling mode, the operation of services (after opening with `AT^SISO`) is explained in separate examples for either mode.

For all examples it is assumed that a connection profile has been created as explained in Section 8.1.1.

When GSM alphabet is enabled (see `AT+CSCS`), keep in mind that escape sequences are required for some special characters: For example, underscore "\_" shall be escaped with `\11`. See also Section 1.6, [Supported character sets](#) for more information on character set and escape sequences.

### 8.16.1 Selecting URC Mode or Polling Mode

To enable or disable the presentation of URCs for the Internet services, use the `AT^SCFG` command, type "Tcp/WithURCs" and select "on" or "off" for parameter `<tcpWithUrc>`.

<code>at^scfg=tcp/withurcs</code>	Query the current setting.
<code>^SCFG: "Tcp/WithURCs", "on"</code>	URC mode is enabled (delivery default).
OK	
<code>at^scfg=tcp/withurcs,off</code>	Select polling mode (by switching off URC mode).
<code>^SCFG: "Tcp/WithURCs", "off"</code>	Polling mode is accepted.
OK	

### 8.16.2 Non-Transparent Client Sends Data in Polling Mode

The example uses service profile 0 configured for Socket service and TCP.

<code>AT^SISO=0</code>	The host opens service profile 0.
OK	Note: URC <code>^SISW: 0, 1</code> does not appear! The host must poll the service state by means of <code>AT^SISI</code> until the service is in <code>&lt;srvState&gt;</code> "Up".
<code>AT^SISI=0</code>	
<code>^SISI: 0,3,0,0,0,0</code>	The service is still in <code>&lt;srvState&gt;</code> "Connecting".
OK	
<code>AT^SISI=0</code>	
<code>^SISI: 0,4,0,0,0,0</code>	The service is now in <code>&lt;srvState&gt;</code> "Up" and is ready to accept data.
OK	
<code>AT^SISW=0,10</code>	The host requests to write 10 bytes.
<code>^SISW: 0, 10, 10</code>	Response that 10 bytes can be sent and are unacknowledged.
0123456789	
OK	
<code>AT^SISW=0,10,1</code>	The host requests to write next 10 bytes and enables the <code>&lt;eodFlag&gt;</code> .
<code>^SISW: 0,0,10</code>	But the service refuses the request, even the end-of-data flag remains without effect.
OK	
<code>AT^SISW=0,10</code>	Next try (polling).
<code>^SISW: 0, 10, 20</code>	Response that the service accepts 10 bytes and this data and the previously sent data are unacknowledged.
0123456789	
OK	

## 8.16 Examples of how to Configure and Use Internet Service Profiles

AT^SISW=0,0,0	Polling to request unacknowledged data.
^SISW: 0, 0, 10	There are 10 bytes still unacknowledged.
OK	
AT^SISI=0	Polling for <code>&lt;srvState&gt;</code> change ("Closing" -> "Down").
^SISI: 0,5,0,20,10,10	20 bytes are transferred via serial interface, 10 bytes are acknowledged by remote peer, 10 bytes are on the way.
OK	
AT^SISI=0	Polling for <code>&lt;srvState&gt;</code> change ("Closing" -> "Down").
^SISI: 0,6,0,20,20,0	The Service is in state "Down" and all data is confirmed by the remote peer.
OK	
AT^SISC=0	Closing the Socket service.
OK	

## 8.16.3 Socket Client Sends Data via TCP Connection with URCs

The example uses service profile 0 configured for Socket service and TCP.

AT^SISO=0	The host opens service profile 0.
OK	
^SISW: 0, 1	Socket service is ready for upload.
AT^SISW=0,10	The host requests to write 10 bytes.
^SISW: 0, 10, 10	Response that 10 bytes can be sent and are unacknowledged.
0123456789	
OK	
^SISW: 0, 1	The URC indicates that the Socket service is ready to transfer more data. Data of the last AT^SISW command are transferred to the remote host.
AT^SISW=0,10,1	The host requests to write next 10 bytes and enables the <code>&lt;eodFlag&gt;</code> to notify that these are the last 10 bytes of the session.
^SISW: 0, 10, 20	Response that the service accepts 10 bytes and this data and the previously sent data are unacknowledged.
0123456789	
OK	
-----	Waiting for URC.
^SISW: 0,2	Data transfer has been finished.
AT^SISC=0	Closing the Socket service.
OK	

## 8.16.4 UDP Scenario

The following example shows communication between a UDP endpoint (host 1) and a UDP client (host 2). Part 1 of the example refers to the UDP endpoint and shows the configuration which allows the UDP endpoint to communicate with any UDP client.

Part 2 shows the settings of a UDP client.

Part 3 and 4 are running in parallel: Part 3 shows the procedures on the UDP endpoint side done when communicating with a UDP client. Part 4 shows the equivalent steps done by the UDP client when communicating with the UDP endpoint.

Part 1 - host 1 is configured for use as UDP endpoint:

## 8.16 Examples of how to Configure and Use Internet Service Profiles

AT^SISS=0,"srvType","Socket"	Select service type Socket.
OK	
AT^SISS=0,"alphabet","1"	The character set of string parameters(ASCII).
OK	
AT^SISS=0,"conId",1	Select connection profile 1.
OK	
AT^SISS=0,"address","sockudp://:6666"	Specify the local UDP port.
OK	

Part 2 - host 2 is configured for use as UDP client:

AT^SISS=1,"srvType","Socket"	Select service type Socket.
OK	
AT^SISS=1,"alphabet","1"	The character set of string parameters(ASCII).
OK	
AT^SISS=1,"conId",1	Select connection profile 1.
OK	
AT^SISS=1,"address","sockudp:// 10.10.0.219:6666"	Specify the address of host 1.
OK	

Part 3 - host 1 opens the service and communicates with host 2:

AT^SISO=0	Open the service.
OK	
^SIS: 0, 5	URC indicates that host 1 is waiting (on UDP port 6666).
AT^SISO?	Request connection status.
^SISO:	
0,"Socket",4,3,0,0,"10.10.0.219:6666","0.0.0.0:0"	
^SISO: 1,""	
^SISO: 2,""	
^SISO: 3,""	
^SISO: 4,""	
OK	
AT^SISR=0,100	Host 1 requests to read 100 bytes.
OK	
^SISR: 0, 22, "10.10.0.222:6561"	Host 1 is reading the text string received from host 2 (see part 4 of the example) and gets information about the remote client, i.e. the address of host 2.
Hello, I'm the client!	
OK	
AT^SISW=0,25,0,"10.10.0.222:6561"	Host 1 starts sending 25 bytes to host 2.
^SISW: 0, 25, 0	Host 1 is sending the message "Hi, I'm the UDP endpoint!" (25 bytes) to host 2. As a result, host 2 is receiving the URC "^SISR: 1, 1" shown below in part 4 of the example.
OK	
^SISW: 0, 1	URC confirms that host 1 can send data again.
AT^SISC=0	Close the service.
OK	

Part 4 - host 2 opens the service and communicates with host 1:

AT^SISO=1	Open the service.
OK	
^SISW: 1, 1	URC indicates that host 2 can send data.

## 8.16 Examples of how to Configure and Use Internet Service Profiles

AT^SISO?	Request connection status.
^SISO: 0, ""	
^SISO:	
1, "Socket", 4, 2, 0, 0, "10.10.0.222:6561", "10.10.0.219:6666"	
^SISO: 2, ""	
^SISO: 3, ""	
^SISO: 4, ""	
OK	
AT^SISW=1,22	Host 2 starts to write 22 bytes.
^SISW: 1, 22, 0	Host 2 is writing the message "Hello, I'm the client!" (22 bytes). As a result, host 1 is receiving the URC "^SISR: 0, 1" shown above in part 3 of this example.
OK	
^SISW: 1, 1	URC confirms that host 2 could send data.
^SISR: 1, 1	URC indicates that host 2 has received data from host 1.
AT^SISR=1,100	Host 2 requests to read 100 bytes.
^SISR: 1, 25	Response indicates that host 2 has received 25 characters from host 1.
Hi, I'm the UDP endpoint!	
OK	
AT^SISC=1	Close the service.
OK	

## 8.16.5 Creating Transparent TCP Socket Client

AT^SISS=1, "srvType", "Socket"	Select service type "Socket".
OK	
AT^SISS=1, "conId", 1	Select connection profile 1.
OK	
AT^SISS=1, "address", "socktcp://10.10.0.110:9996;etx=26;timer=200"	Specify server address. Add "etx" parameter to set Transparent mode.
OK	

## 8.16.6 Opening and Closing Transparent TCP Service

AT^SISO=1	Open Transparent TCP service.
OK	
^SISW: 1, 1	URC indicates that Transparent TCP service is ready.
at^sist=1	Open transparent access mode.
CONNECT	Service is ready to send or receive data. Client is sending data, e.g Good Morning (not visible in example). Client types +++ to return to AT command mode (+++ not visible in example).
OK	
at^sisi=1	Check the service state of service profile 1.
^SISI: 1, 4, 0, 16, 16, 0	Service is in state "Up" (4). 0 bytes received, 16 bytes sent and acknowledged.
OK	
AT^SISO?	Check the service state of service profile 1.
^SISO: 0, ""	



## 8.16 Examples of how to Configure and Use Internet Service Profiles

^SISO: 1, "Socket", 4, 2, 0, 16, "10.10.0.200:1024", "10.10.0.108:65532"	Service is in state "Up" (4). Socket is assigned. 0 bytes received, 16 bytes sent. Local and remote IP address are assigned.
^SISO: 2, ""	
^SISO: 3, ""	
^SISO: 4, ""	
OK	
AT^SISC=1	Close the service.
OK	
at^sisi=1	Check the service state of service profile 1.
^SISI: 1, 2, 0, 0, 0, 0	Service profile 1 is in state "Allocated" (2).
OK	
AT^SISO?	Check the service state of service profile 1.
^SISO: 0, ""	
^SISO:	Service is in state "Allocated" (2). Socket not assigned.
1, "Socket", 2, 1, 0, 0, "0.0.0.0:0", "10.10.0.108:65532"	
^SISO: 2, ""	
^SISO: 3, ""	
^SISO: 4, ""	
OK	

## 8.16.7 Transparent TCP Client Receives Data While in AT Command Mode

AT^SISO=1	Open Transparent TCP service.
OK	
^SISW: 1, 1	URC indicates that Transparent TCP service is ready.
at^sist=1	Open transparent access mode.
CONNECT	Service is ready to send or receive data.
Hello	Client is reading received data.
	Client types +++ to return to AT command mode (+++ not visible in example).
OK	
^SISR: 1, 1	URC indicates that new data is available for the client.
at^sist=1	Return to transparent access mode.
CONNECT	Service is ready to send or receive data.
Goodbye	Client is reading received data.
	Client types +++ to return to AT command mode (not visible in example).
OK	
at^sisi=1	Check the service state of service profile 1.
^SISI: 1, 4, 12, 6, 6, 0	Service is in state "Up" (4). 12 bytes received, 6 bytes sent and acknowledged.
OK	
AT^SISO?	Check the service state of service profile 1.
^SISO: 0, ""	
^SISO:	Service is in state "Up" (4). Socket is assigned. 12 bytes received, 6 bytes sent. Local and remote IP address are assigned.
1, "Socket", 4, 2, 12, 6, "10.10.0.200:1025", "10.10.0.108:65532"	
^SISO: 2, ""	
^SISO: 3, ""	
^SISO: 4, ""	

## 8.16 Examples of how to Configure and Use Internet Service Profiles

OK at^sici? ^SICI: 0,2,1,"10.10.0.200"	Check the state of the Internet connection profile. Internet connection profile is in the state "Up". Local IP address is assigned.
OK AT^SISC=1 OK	Close the service.
at^sisi=1 ^SISI: 1,2,0,0,0,0 OK	Check the service state of service profile 1. Service profile 1 is in state "Allocated" (2).
AT^SISO? ^SISO: 0,"" ^SISO: 1,"Socket",2,1,0,0,"0.0.0.0:0","0.0.0.0:0" ^SISO: 2,"" ^SISO: 3,"" ^SISO: 4,"" OK	Check the service state of service profile 1.  Service is in state "Allocated" (2). Socket not assigned.
at^sici? ^SICI: 0,0,0,"0.0.0.0" OK	Check the state of the Internet connection profile. No Internet connection profile currently used.

### 8.16.8 Remote Server Disconnects While Transparent TCP Service is in Transparent Access Mode

AT^SISO=1 OK ^SISW: 1, 1	Open Transparent TCP service.  URC indicates that Transparent TCP service is ready.
at^sist=1 CONNECT Good Morning NO CARRIER	Open transparent access mode. Service is ready to send or receive data. Remote server has sent data and disconnected. Client reads data. Then client gets NO CARRIER in AT command mode.
at^sisi=1 ^SISI: 1,6,12,0,0,0 OK	Check the service state of service profile 1. Service is in state "Down" (6). 12 bytes received.
AT^SISO? ^SISO: 0,"" ^SISO: 1,"Socket",6,2,12,0,"10.10.0.200:1026","10.10.0.108:65532" ^SISO: 2,"" ^SISO: 3,"" ^SISO: 4,"" OK	Check the service state of service profile 1.  Service is in state "Down" (6). Socket is assigned. 6 bytes received. 12 bytes sent. Local and remote IP address are assigned.
at^sici? ^SICI: 0,2,1,"10.10.0.200"	Check the state of the Internet connection profile. Internet connection profile is in state "Up". Local IP address is assigned.
OK AT^SISC=1 OK	Close the service.
at^sisi=1 ^SISI: 1,2,0,0,0,0 OK	Check the service state of service profile 1. Service profile 1 is in state "Allocated" (2).

## 8.16 Examples of how to Configure and Use Internet Service Profiles

## 8.16.9 Server Disconnects While Transparent TCP Service is in AT Command Mode and Data is Pending

<pre>AT^SISO=1 OK ^SISW: 1, 1  at^sist=1 CONNECT Good Morning  OK ^SISR: 1, 1  at^sisi=1 OK ^SISI: 1,4,12,3,3,0  OK AT^SISO? ^SISO: 0, "" ^SISO: 1, "Socket", 4, 2, 12, 3, "10.10.0.200:1027", "10. 10.0.108:65532" ^SISO: 2, "" ^SISO: 3, "" ^SISO: 4, "" OK at^sist=1 CONNECT Good bye NO CARRIER at^sisi? ^SISI: 1,6,20,3,3,0  OK AT^SISO? ^SISO: 0, "" ^SISO: 1, "Socket", 20, 3, "10.10.0.200:1027", "10.10.0. 108:65532" ^SISO: 2, "" ^SISO: 3, "" ^SISO: 4, "" OK AT^SISC=1 OK at^sisi=1 ^SISI: 1,2,0,0,0,0 OK AT^SISO? ^SISO: 0, "" ^SISO: 1, "Socket", 2, 1, 0, 0, "0.0.0.0:0", "0.0.0.0:0"</pre>	<p>Open Transparent TCP service.</p> <p>URC indicates that Transparent TCP service is ready.</p> <p>Open transparent access mode. Service is ready to send or receive data. Client is reading data. Client types +++ to return to AT command mode (not visible in example).</p> <p>Remote server has sent data and closed the connection (not visualized in this example). URC indicates that remaining data is available for reading. Check service state of service profile 1.</p> <p>Service profile 1 is in state "Up" (4). 12 bytes received. 3 bytes sent and acknowledged.</p> <p>Check the service state of service profile 1.</p> <p>Service is in state "Up" (4). Socket is assigned. 12 bytes received. 3 bytes sent. Local and remote IP address are assigned.</p> <p>Return to transparent access mode. Service is ready to send or receive data. Client reads remaining data. Then client gets NO CARRIER in AT command mode. Check service state of service profile 1. Service is in service "Down" state (6). 20 bytes received. 3 bytes sent and acknowledged.</p> <p>Check the service state of service profile 1.</p> <p>Service is in state "Down" (6). Socket is assigned. 20 bytes received. 3 bytes sent. Local and remote IP address are assigned.</p> <p>Close the service.</p> <p>Check service state of service profile 1. Service profile 1 is in state "Allocated" (2).</p> <p>Check the service state of service profile 1.</p> <p>Service is in state "Allocated" (2). Socket is not assigned. Local and remote IP address not assigned.</p>
--	---

## 8.16 Examples of how to Configure and Use Internet Service Profiles

```

^SISO: 2, ""
^SISO: 3, ""
^SISO: 4, ""
OK

```

## 8.16.10 Creating Transparent UDP Socket Client

AT^SISS=1,"srvType","Socket"	Select service type "Socket".
OK	
AT^SISS=1,"alphabet","1"	The character set of string parameters(ASCII).
OK	
AT^SISS=1,"conId",1	Select connection profile 1.
OK	
AT^SISS=1,"address","sockudp:// 10.10.0.110:9996;etx=26;timer=200"	Specify server address. Add "etx" parameter to set Transparent mode.
OK	

## 8.16.11 Opening and Closing Transparent UDP Service

AT^SISO=1	Open Transparent UDP service.
OK	
^SISW: 1, 1	URC indicates that Transparent UDP service is ready.
at^sist=1	Open transparent access mode.
CONNECT	Service is ready to send or receive data. Client is sending data, e.g Good Morning (not visible in example).
	Client types +++ to return to AT command mode (+++ not visible in example).
OK	
at^sisi=1	Check the service state of service profile 1.
^SISI: 1,4,0,16,0,0	Service is in state "Up" (4). 0 bytes received, 16 bytes sent.
OK	
AT^SISO?	Check the service state of service profile 1.
^SISO: 0, ""	
^SISO:	Service is in state "Up" (4). Socket is assigned. 0 bytes received, 16 bytes sent. Local and remote IP address are assigned.
1,"Socket",4,2,0,16,"10.10.0.200:1024","10. 10.0.108:65532"	
^SISO: 2, ""	
^SISO: 3, ""	
^SISO: 4, ""	
OK	
AT^SISC=1	Close the service.
OK	
at^sisi=1	Check the service state of service profile 1.
^SISI: 1,2,0,0,0,0	Service profile 1 is in state "Allocated" (2).
OK	
AT^SISO?	Check the service state of service profile 1.
^SISO: 0, ""	
^SISO:	Service is in state "Allocated" (2). Socket not assigned.
1,"Socket",2,1,0,0,"0.0.0.0:0","10.10.0.108 :65532"	
^SISO: 2, ""	

## 8.16 Examples of how to Configure and Use Internet Service Profiles

```

^SISO: 3, ""
^SISO: 4, ""
OK

```

## 8.16.12 MQTT Scenario

The following example shows communication between MQTT clients and an MQTT server.

Part 1 shows the MQTT client sends a PUBLISH request with static setup in URC Mode.

Part 2 shows the MQTT client sends a SUBSCRIBE request with static setup and a PUBLISH request received from a server in URC Mode.

Part 3 shows the MQTT client sends a SUBSCRIBE and 2 PUBLISH request with dynamic setup in URC Mode.

Part 4 shows the MQTT client sends a SUBSCRIBE with dynamic setup and 2 PUBLISH request received from server in Polling Mode.

Part 5 shows communication between 2 MQTT client with MQTT characteristics(Retain, KeepAlive, Will and CleanSession):

Part 1 - Sends a PUBLISH request with a static setup(URC Mode):

AT^SISS=0, "srvType", "Mqtt"	Select service type Mqtt.
OK	
AT^SISS=0, "alphabet", "1"	The character set of string parameters(ASCII).
OK	
AT^SISS=0, "conId", 1	Select connection profile 1.
OK	
AT^SISS=0, "address", "mqtt:// user:pass@mqtt.gemalto.com"	Specify the address of MQTT server.
OK	
AT^SISS=0, cmd, "publish"	Select PUBLISH request to be send by MQTT client to server.
OK	
AT^SISS=0, "hcContentLen", 10	Specify PUBLISH request payload will have 10 bytes and will be defined by SISW command.
OK	
AT^SISS=0, topic, "alarm/code"	Select topic of PUBLISH request.
OK	
AT^SISS=0, qos, "1"	Select Qos of PUBLISH request.
OK	
AT^SISO=0	Open connection to MQTT server using static profile.
OK	
^SIS: 0, 0, 2500, "MQTT server accepted connection on clean session"	URC indicates that connection with MQTT server is acknowledged.
^SISW: 0, 1	URC indicates that client is ready to send data.
AT^SISW=0, 10	client starts to write 10 bytes.
^SISW: 0, 10, 0	client is writing the message "1234567890" (10 bytes).
OK	
^SISW: 0, 1	URC confirms that client could send data.
^SIS: 0, 0, 3520, "alarm/code"	URC indicates that PUBLISHED request was acknowledged by server.
AT^SISC=0	Close the service.
OK	
^SIS: 0, 0, 2505, "Disconnected by client"	URC indicates connection was closed.

Part 2 - Sends a SUBSCRIBE request with static setup and a PUBLISH request received(URC Mode):

AT^SISS=0, "srvType", "Mqtt"	Select service type Mqtt.
OK	

## 8.16 Examples of how to Configure and Use Internet Service Profiles

AT^SISS=0,"alphabet","1"	The character set of string parameters(ASCII).
OK	
AT^SISS=0,"conId",1	Select connection profile 1.
OK	
AT^SISS=0,"address","mqtt:// user:pass@mqtt.gemalto.com"	Specify the address of MQTT server.
OK	
AT^SISS=0,cmd,"subscribe"	Select SUBSCRIBE request to be sent by MQTT client to server.
OK	
AT^SISS=0,topicFilter,"heating/temperature/ change;lights/kitchen/off"	Select topics to which client wants to be subscribed.
OK	
AT^SISS=0,topicQos,"0;1"	Select Qos for each topic.
OK	
AT^SISO=0	Open connection to MQTT server using static profile.
OK	
^SIS: 0,0,2500,"MQTT server accepted connection on clean session"	URC indicates that connection with MQTT server is acknowledged.
^SISW: 0, 1	URC indicates that client is ready to send data.
^SIS: 0,0,2520,"heating/temperature/change"	URC confirms successful subscription to topic heating/temperature/change.
^SIS: 0,0,2521,"lights/kitchen/off"	URC confirms successful subscription to topic lights/kitchen/off.
^SIS: 0,0,3488,"heating/temperature/change"	URC indicates received PUBLISH request from server with topic heating/temperature/change and payload.
AT^SISR=0,100	client requests to read 100 bytes.
^SISR: 0, 6	Response indicates that the client has received 6 characters from server.
19.0 C	
OK	
AT^SISC=0	Close the service.
OK	
^SIS: 0,0,2505,"Disconnected by client"	URC indicates connection was closed.

## Part 3 - Sends a SUBSCRIBE request and 2 PUBLISH request with dynamic setup (URC Mode):

AT^SISS=3,"srvType","Mqtt"	Select service type Mqtt.
OK	
AT^SISS=3,"alphabet","1"	The character set of string parameters(ASCII).
OK	
AT^SISS=3,"conId",1	Select connection profile 1.
OK	
AT^SISS=3,"address","mqtt:// user:pass@mqtt.gemalto.com"	Specify the address of MQTT server.
OK	
AT^SISO=3,2	Open connection to MQTT server using dynamic profile.
OK	
^SIS: 3,0,2500,"MQTT server accepted connection on clean session"	URC indicates that connection with MQTT server is acknowledged.
^SISW: 3, 1	URC indicates that client is ready to send data.
AT^SISD=3,setParam,cmd,"subscribe"	Select SUBSCRIBE request to be sent by MQTT client to server.
OK	
AT^SISD=3,setParam,topicFilter,"heating/ temperature/change;lights/kitchen/off"	Select topics to which client wants to be subscribed.
OK	

## 8.16 Examples of how to Configure and Use Internet Service Profiles

AT^SISD=3, setParam, topicQos, "0;1"	Select Qos for each topic.
OK	
AT^SISU=3	SUBSCRIBE request send. Dynamically setup parameters used for request creation..
OK	
^SIS: 3,0,2520, "heating/temperature/change"	URC confirms successful subscription to topic heating/temperature/change.
^SIS: 3,0,2521, "lights/kitchen/off"	URC confirms successful subscription to topic lights/kitchen/off.
^SIS: 3,0,3488, "heating/temperature/change"	URC indicates received PUBLISH request from server with topic heating/temperature/change and payload.
AT^SISR=3,100	client requests to read 100 bytes.
^SISR: 3, 6	Response indicates that client has received 6 characters from server.
19.0 C	
OK	
AT^SISD=3, setParam, cmd, "publish"	Select PUBLISH request to be send by MQTT client to server.
OK	
AT^SISD=3, setParam, hcContentLen, 0	Specify PUBLISH request payload is defined by parameter hcContent.
OK	
AT^SISD=3, setParam, hcContent, ""	Specify PUBLISH request has no payload.
OK	
AT^SISD=3, setParam, topic, "iotbutton/button-press/button1"	Select topic of PUBLISH request.
OK	
AT^SISD=3, setParam, qos, "1"	Select Qos of PUBLISH request
OK	
AT^SISU=3	PUBLISH request send. Dynamically setup parameters used for request creation.
OK	
^SIS: 3,0,3520, "lights/kitchen/off"	URC indicates that PUBLISHED request was acknowledged by server.
AT^SISU=3, "publish", "1:temperature/floor_6:0:25.5 C:0"	PUBLISH request defined and send.
OK	
^SIS: 3,0,3520, "temperature/floor_6"	URC indicates that PUBLISHED request was acknowledged by server.
AT^SISC=3	Close the service.
OK	
^SIS: 3,0,2505, "Disconnected by client"	URC indicates connection was closed.

## Part 4 - Sends a SUBSCRIBE request with dynamic setup and 2 PUBLISH request received(Polling Mode):

AT^SCFG=tcp/withurcs, off	Select polling mode.
OK	
AT^SISO=3,2	Open connection to MQTT server using dynamic profile.
OK	
AT^SISE=3,1	Polling for the oldest unread event (URC).
^SIS: 3,0,2500, "MQTT server accepted connection on clean session"	Connection with MQTT server is acknowledged.
OK	
AT^SISU=3, "subscribe", "heating/temperature/change;lights/kitchen/off:0;1"	SUBSCRIBE request defined and send.
OK	
AT^SISE=3,1	Polling for the oldest unread event (URC).

## 8.16 Examples of how to Configure and Use Internet Service Profiles

<code>^SIS: 3,0,2520,"heating/temperature/change"</code>	Confirmation of successful subscription to topic heating/temperature/change.
<code>OK</code>	
<code>AT^SISE=3,1</code>	Polling for the oldest unread event (URC).
<code>^SIS: 3,0,2521,"lights/kitchen/off"</code>	Confirmation of successful subscription to topic lights/kitchen/off.
<code>OK</code>	
<code>AT^SISE=3,1</code>	Polling for the oldest unread event (URC).
<code>OK</code>	
<code>AT^SISE=3,1</code>	Polling for oldest unread event (URC).
<code>^SIS: 3,0,3488,"heating/temperature/change"</code>	Received PUBLISH request from server with topic heating/temperature/change and payload.
<code>OK</code>	
<code>AT^SISR=3,100</code>	client requests to read 100 bytes.
<code>^SISR: 3, 6</code>	Response indicates that the client has received 6 characters from server.
<code>19.0 C</code>	
<code>OK</code>	
<code>AT^SISE=3,1</code>	Polling for othe ldest unread event (URC).
<code>OK</code>	
<code>AT^SISC=3</code>	Close the service.
<code>OK</code>	
<code>AT^SISE=3,1</code>	Polling for the oldest unread event (URC).
<code>^SIS: 3,0,2505,"Disconnected by client"</code>	URC indicates connection was closed.
<code>OK</code>	

## Part 5 - Communication between 2 MQTT client:

<code>Client-1:</code>	
<code>AT^SISS=3,"srvType","Mqtt"</code>	Select service type Mqtt.
<code>OK</code>	
<code>AT^SISS=3,"alphabet","1"</code>	The character set of string parameters(ASCII).
<code>OK</code>	
<code>AT^SISS=3,"conId",1</code>	Select connection profile 1.
<code>OK</code>	
<code>AT^SISS=3,"address","mqtt:// user:pass@mqtt.gemalto.com"</code>	Specify the address of MQTT server.
<code>OK</code>	
<code>AT^SISS=3,"clientId","TestClient1"</code>	Specify Client-1's identifier.
<code>OK</code>	
<code>at^siss=3,"keepAlive",20</code>	Specify keep alive time interval.
<code>OK</code>	
<code>at^siss=3,"cleanSession",0</code>	Server MUST store the Session after the Client and Server are disconnected.
<code>OK</code>	
<code>at^siss=3,"lastWillTopic","Indication"</code>	Specify last will topic.
<code>OK</code>	
<code>at^siss=3,"lastWillMessage","No network!"</code>	Specify last will message.
<code>OK</code>	
<code>at^siss=3,"lastWillQos",1</code>	Specify Qos level of publishing last will message.
<code>OK</code>	
<code>AT^SISO=3,2</code>	Open Client-1 connection to MQTT server using dynamic profile.
<code>OK</code>	
<code>^SIS: 3,0,2501,"MQTT server accepted connec- tion on retain session"</code>	URC indicates that connection with MQTT server is acknowledged.
<code>^SISW: 3, 1</code>	URC indicates that client is ready to send data.
<code>AT^SISU=3,"publish","1:heating/temperature/ change:1:25.5 C:0"</code>	PUBLISH request defined and send(retain=1).



## 8.16 Examples of how to Configure and Use Internet Service Profiles

```
OK
^SIS: 3,0,3520,"heating/temperature/change"
AT^SISU=3,"subscribe","lights/kitchen/off:1"
OK
^SIS: 3,0,2521,"lights/kitchen/off"
```

URC indicates that PUBLISHED request was acknowledged by server.  
SUBSCRIBE request defined and send.

Confirmation of successful subscription to topic lights/kitchen/off.

```
Client-2:
AT^SISS=3,"srvType","Mqtt"
OK
AT^SISS=3,"alphabet","1"
OK
AT^SISS=3,"conId",1
OK
AT^SISS=3,"address","mqtt://
user:pass@mqtt.gemalto.com"
OK
AT^SISS=3,"clientId","TestClient2"
OK
at^siss=3,"keepAlive",0
OK
at^siss=3,"cleanSession",0

OK
AT^SISO=3,2

OK
^SIS: 3,0,2501,"MQTT server accepted connec-
tion on retain session"
^SISW: 3, 1
AT^SISU=3,"subscribe","heating/temperature/
change;Indication:1;1"
OK
^SIS: 3,0,2521,"heating/temperature/change"
^SIS: 3,0,2521,"Indication"
^SIS: 3,0,3488,"heating/temperature/change"

AT^SISR=3,100
^SISR: 3, 6
25.5 C
OK
```

Select service type Mqtt.

The character set of string parameters(ASCII).

Select connection profile 1.

Specify the address of MQTT server.

Specify Client-1's identifier.

Specify keep alive time interval.

Server MUST store the Session after the Client and Server are disconnected.

Open Client-1 connection to MQTT server using dynamic profile.

URC indicates that connection with MQTT server is acknowledged.

URC indicates that client is ready to send data.  
SUBSCRIBE request defined and send.

URC confirms successful subscription to topic heating/temperature/change.

URC confirms successful subscription to topic Indication.

Received PUBLISH request from server with topic heating/temperature/change and payload.  
client requests to read 100 bytes.

Response indicates that client has received 6 characters from server.

```
Client-1:
Client-1 no network service!
Server does not receive a Control Packet from the Client-1 within one and a half times
the Keep Alive time period.
```

```
Client-2:
^SIS: 3,0,3489,"Indication"

AT^SISR=3,100
^SISR: 3, 11
No network!
OK
```

Received PUBLISH request from server with Indication(Client-1 last will).  
client requests to read 100 bytes.

Response indicates that client has received 11 characters from server.

## 8.16 Examples of how to Configure and Use Internet Service Profiles

```
AT^SISU=3,"publish","1:lights/kitchen/
off:0:off:0"
OK
^SIS: 3,0,3520,"lights/kitchen/off"
```

PUBLISH request defined and send.(Client-1 subscribed this topic, but Currently not on-line)

URC indicates that PUBLISHED request was acknowledged by server.

*Client-1:*

Client-1 reconnect

```
AT^SISO=3,2
```

OK

```
^SIS: 3,0,2501,"MQTT server accepted connection on retain session"
```

```
^SISW: 3, 1
```

```
^SIS: 3,0,3489,"lights/kitchen/off"
```

Server resume connection with Client-1.

URC indicates that connection with MQTT server is acknowledged.

URC indicates that client is ready to send data.

Received PUBLISH request from server with topic lights/kitchen/off(Server stored Client-2's QoS1 message).

client requests to read 100 bytes.

Response indicates that client has received 3 characters from server.

```
AT^SISR=3,100
```

```
^SISR: 3, 3
```

off

OK

## 8.16.13 Ping

The following example shows a Ping output:

```
AT^SISX="Ping",0,192.168.1.2,4,2000
```

```
^SISX: "Ping",1,0,"192.168.1.2",1043
```

```
^SISX: "Ping",1,0,"192.168.1.2",586
```

```
^SISX: "Ping",1,0,"192.168.1.2",531
```

```
^SISX: "Ping",1,0,"192.168.1.2",415
```

```
^SISX: "Ping",2,1,4,4,0,0
```

```
^SISX: "Ping",3,1,415,1043,643
```

OK

## 9. Packet Domain Related Commands

The AT commands described in this chapter allow the Customer Application to control packet switched services in networks.

### 9.1 AT+CGACT PDP context activate or deactivate

The [AT+CGACT](#) write command is used to activate or deactivate the specified PDP context(s). After the command has completed, the MT remains in V.250 command state. If any PDP context is already in the requested state, the state for that context remains unchanged. If the MT is not PS attached when the activation form of the command is executed, the MT first performs a PS attach and then attempts to activate the specified contexts.

For EPS, if an attempt is made to disconnect the last PDN connection, then the MT responds with ERROR or, if extended error responses are enabled, a +CME ERROR. For EPS, the activation request for an EPS bearer resource will be answered by the network by either an EPS dedicated bearer activation or EPS bearer modification request. The request must be accepted by the MT before the PDP context can be set into established state.

The [AT+CGACT](#) read command returns the current activation states for all the defined PDP contexts.

The [AT+CGACT](#) test command is used for requesting information on the supported PDP context activation states.

#### Syntax

##### Test Command

```
AT+CGACT=?
```

Response(s)

```
+CGACT: (list of supported <state>s)
```

```
OK
```

##### Read Command

```
AT+CGACT?
```

Response(s)

```
+CGACT: [<cid>, <state>]
```

```
[+CGACT: <cid>, <state>]
```

```
...
```

```
OK
```

##### Write Command

```
AT+CGACT=<state>, <cid>
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
+ -
```

Reference(s)

3GPP TS 27.007 [38]

---

### Parameter Description

`<state>`<sup>(num)</sup>

Indicates the state of PDP context activation.

0	Detached
[1]	Attached

`<cid>`<sup>(num)</sup>

Parameter specifies a particular PDP context definition (see [AT+CGDCONT](#) parameter `<cid>`). The parameter is used in other PDP context-related commands.

### Note

- A maximum of 11 contexts can be activated at the same time, no matter on which interface. Note that, depending on the provider, the number of activated contexts may be further restricted.

## 9.2 AT+CGATT PS attach or detach

The [AT+CGATT](#) write command is used to attach the MT to, or detach the MT from the Packet Domain service. After the command has completed, the MT remains in V.250 command state. If the MT is already in the requested state, the command is ignored and the OK response is returned. Any active PDP contexts will be automatically deactivated when the attachment state changes to detached.

The [AT+CGATT](#) read command returns the current Packet Domain service state.

The [AT+CGATT](#) test command is used for requesting information on the supported Packet Domain service states.

### Syntax

Test Command	
AT+CGATT=?	
Response(s)	
+CGATT: (list of supported <state>s)	
OK	
Read Command	
AT+CGATT?	
Response(s)	
+CGATT: <state>	
OK	
Write Command	
AT+CGATT=[<state>]	
Response(s)	
OK	
ERROR	
+CME ERROR: <err>	
PIN Last	Reference(s)
+ -	3GPP TS 27.007 [38]

### Parameter Description

<state> <sup>(num)</sup>	
Indicates the state of Packet Domain attachment. The parameter is global for all interfaces.	
0	Detached
[1]	Attached

## 9.3 AT+CGAUTH Define PDP Context Authentication Parameters

The [AT+CGAUTH](#) write command specifies the type of authentication protocol for the specified PDP context. The [AT+CGAUTH](#) read command returns the current settings for each defined PDP context. If no PDP context is defined the read command returns only OK result code.

### Syntax

<p>Test Command</p> <p>AT+CGAUTH=?</p> <p>Response(s)</p> <p>+CGAUTH: (range of supported&lt;cid&gt;s), (list of supported &lt;auth_prot&gt;s), (max. string length of &lt;userId&gt;s), (max. string length of &lt;password&gt;s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Read Command</p> <p>AT+CGAUTH?</p> <p>Response(s)</p> <p>[+CGAUTH: &lt;cid&gt;, &lt;auth_prot&gt;, &lt;userId&gt;, &lt;password&gt;]</p> <p>[+CGAUTH: &lt;cid&gt;, &lt;auth_prot&gt;, &lt;userId&gt;, &lt;password&gt;]</p> <p>[+CGAUTH: ...]</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>Write Command</p> <p>AT+CGAUTH=&lt;cid&gt;[, &lt;auth_prot&gt;[, &lt;userId&gt;, &lt;password&gt;]]</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Parameter Description

<b>&lt;cid&gt;<sup>(num)</sup></b>	
Specifies a particular PDP context definition (see <a href="#">AT+CGDCONT</a> parameter <cid>).	
0...10	
<b>&lt;auth_prot&gt;<sup>(num)</sup></b>	
Types of authentication to be used for specified PDP context.	
[0]	none Used to indicate that no authentication protocol is used for this PDP context. Username and password are removed if previously specified.
1	PAP
2	CHAP

---

`<userId>`<sup>(str)</sup>

Specifies the user name used for authentication.

`<password>`<sup>(str)</sup>

Specifies the password used for authentication.

**Note**

- The length of `<userId>` and `<password>` can be 0 to 60 characters.

## 9.4 AT+CGDCONT Define PDP Context

The **AT+CGDCONT** write command specifies the parameters for a PDP context identified by the context identifier **<cid>**. The number of contexts that may be in a defined state at the same time is given by the range indicated in the test command response. A special form of the write command (**AT+CGDCONT=<cid>**) causes the values for context **<cid>** to become undefined. The initial PDP context (+CGDCONT: 0,"IPV4V6",,0,0,0) is supported, and **AT+CGDCONT=0** resets context number 0 to its particular default settings.

The **AT+CGDCONT** read command returns the current settings for each defined PDP context.

The **AT+CGDCONT** test command returns supported values as a compound value.

### Syntax

<p>Test Command</p> <pre>AT+CGDCONT=?</pre> <p>Response(s)</p> <pre>+CGDCONT: (range of supported&lt;cid&gt;s), &lt;PDP_type&gt;, , (list of supported &lt;d_comp&gt;s), (list of supported &lt;h_comp&gt;s), (list of supported &lt;NSLPI&gt;s) OK ERROR +CME ERROR: &lt;err&gt;</pre>	
<p>Read Command</p> <pre>AT+CGDCONT?</pre> <p>Response(s)</p> <pre>+CGDCONT: &lt;cid&gt;, &lt;PDP_type&gt;, &lt;APN&gt;, &lt;d_comp&gt;, &lt;h_comp&gt;, &lt;NSLPI&gt; [+CGDCONT: ...] OK ERROR +CME ERROR: &lt;err&gt;</pre>	
<p>Write Command</p> <pre>AT+CGDCONT=&lt;cid&gt;[, &lt;PDP_type&gt;[, &lt;APN&gt;[, &lt;d_comp&gt;[, &lt;h_comp&gt;[, &lt;NSLPI&gt;]]]]</pre> <p>Response(s)</p> <pre>OK ERROR +CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>- -</pre>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Parameter Description

<p><b>&lt;cid&gt;</b><sup>(num)(NV)</sup></p> <p>PDP Context Identifier</p> <p>Parameter specifies a particular PDP context definition. The parameter is used in other PDP context-related commands.</p> <p>The <b>&lt;cid&gt;</b> value cannot be set to 7 when Bearer Independent Protocol (BIP) is enabled.</p> <p>Only the context definition with <b>&lt;cid&gt;=0</b> is stored in non-volatile (NV) memory.</p> <p>0<sup>(D)</sup>...10</p>
--



<PDP\_type><sup>(str)(NV)</sup>

#### Packet Data Protocol type

Specifies the type of the packet data protocol. It is stored in non-volatile (NV) memory only when <cid>=0.

"IP"	Internet Protocol (IETF STD 5)
"IPV6"	Internet Protocol, version 6 (IETF RFC 2460)
"IPV4V6" <sup>(D)</sup>	Virtual <PDP_type> introduced to handle dual IP stack UE capability (see 3GPP TS 24.301)
"NONIP"	None IP

<APN><sup>(str)(NV)</sup>

#### Access Point Name

The logical name that is used to select the GGSN or the external packet data network. Maximum length: 63 characters. If the value is null or omitted, then the subscription value will be requested. It is stored in non-volatile (NV) memory only when <cid>=0.

<d\_comp><sup>(num)(NV)</sup>

#### Data Compression

Controls the PDP data compression; applicable for Subnetwork Dependent Convergence Protocol (SNDCP) only. It is stored in non-volatile (NV) memory only when <cid>=0. See 3GPP TS 44.065 for details.

[0]	off
-----	-----

<h\_comp><sup>(num)(NV)</sup>

#### Header Compression

Controls the PDP header compression. It is stored in non-volatile (NV) memory only when <cid>=0. See 3GPP TS 44.065, 3GPP TS 25.323 for details.

[0]	off
-----	-----

<NSLPI><sup>(num)</sup>

Numeric parameter used to indicate the NAS signalling priority requested for this PDP context. The MT utilises the provide NSLPI information as specified in 3GPP TS 24.301; and 3GPP TS 24.008.

[0]	Indicates that this PDP context is to be activated with the value for the low priority indicator configured in the MT.
1	Indicates that this PDP context is to be activated with the value for the low priority indicator set to "MS is not configured for NAS signalling low priority".

### Notes

- Only the context definitions with <cid>=0 can be stored in non-volatile (NV) memory.
- Only the context definitions with <cid>=0 can be stored when module is in minimum functionality (AT+CFUN=0). If the module is set to minimum functionality, all the other context definitions will be lost.

## 9.5 AT+CEREG EPS Network Registration Status

**AT+CEREG** write command enables presentation of URC "+CEREG: <stat>" when <n>=1 and UE's EPS network registration status in LTE changes, or URC "+CEREG: <stat>[, <tac>, <ci>[, <ActT>]]" when <n>=2 and the current network cell in LTE changes. The parameters <ActT>, <tac> and <ci> are provided only if available. The value <n>=3 further extends the URC with [<CauseType>,<RejectCause>], when available, when the value of <stat> changes.

If the UE wants to apply PSM for reducing its power consumption the set command controls the presentation of an URC "+CEREG: <stat>[, [<tac>], [<ci>], [<ActT>], [<CauseType>], [<RejectCause>], [<ActiveTime>], [<PeriodicTAU>]]]". When <n>=4 the URC will provide the UE with additional information for the Active Time value and the extended periodic TAU value if there is a change of the network cell in E-UTRAN. The value <n>=5 further enhances the URC with <CauseType> and <RejectCause> when the value of <stat> changes. The parameters <ActT>, <tac>, <ci>, <CauseType>, <RejectCause>, <ActiveTime> and <PeriodicTAU> are provided only if available.

**AT+CEREG** read command queries the current URC presentation status and <stat> which shows whether the network has currently indicated the registration of the ME.

### Syntax

<p>Test Command</p> <p>AT+CEREG=?</p> <p>Response(s)</p> <p>+CEREG: (list of supported &lt;n&gt;s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+CEREG?</p> <p>Response(s)</p> <p>+CEREG: &lt;n&gt;, &lt;stat&gt;[, [&lt;tac&gt;], [&lt;ci&gt;], [&lt;ActT&gt;], [&lt;CauseType&gt;], [&lt;RejectCause&gt;], [&lt;ActiveTime&gt;], [&lt;PeriodicTAU&gt;]]]</p> <p>OK</p>	
<p>Write Command</p> <p>AT+CEREG=&lt;n&gt;</p> <p>Response(s)</p> <p>OK</p> <p>ERROR</p> <p>+CME ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>- -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Unsolicited Result Codes

#### URC 1

+CEREG: <stat>

Indicates a change in the UE's EPS network registration status.

#### URC 2

+CEREG: <stat>[, <tac>, <ci>[, <ActT>]]

Indicates a change in the UE's EPS network registration status or a change of the network cell including location information.

URC 3

+CEREG: <stat>[, [<tac>], [<ci>], [<AcT>][, <CauseType>, <RejectCause>]]

Indicates a change in the UE's EPS network registration status, or a change of the network cell including location information, and the EMM cause value information.

URC 4

+CEREG: <stat>[, [<tac>], [<ci>], [<AcT>][, , [, [<ActiveTime>], [<PeriodicTAU>]]]]

For a UE that wants to apply PSM, indicates a change in the UE's EPS network registration status or a change of the network cell including location information

URC 5

+CEREG: <stat>[, [<tac>], [<ci>], [<AcT>][, [<CauseType>], [<RejectCause>][, [<ActiveTime>], [<PeriodicTAU>]]]]

For a UE that wants to apply PSM, indicates a change in the UE's EPS network registration status or a change of the network cell including location information and EMM cause value information

**Parameter Description**

<n> <sup>(num)(&amp;V)(&amp;W)</sup>	
0(&F)(D)	Disable network registration URC
1	Enable network registration URC "+CEREG: <stat>".
2	Enable network registration URC "+CEREG: <stat>[, <tac>, <ci>[, <AcT>]]".
3	Enable network registration, location information and EMM cause value information URC "+CEREG: <stat>[, [<tac>], [<ci>], [<AcT>][, <CauseType>, <RejectCause>]]".
4	For a UE that wants to apply PSM, enable network registration and location information URC "+CEREG: <stat>[, [<tac>], [<ci>], [<AcT>][, , [, [<ActiveTime>], [<PeriodicTAU>]]]]".
5	For a UE that wants to apply PSM, enable network registration, location information and EMM cause value information URC "+CEREG: <stat>[, [<tac>], [<ci>], [<AcT>][, [<CauseType>], [<RejectCause>][, [<ActiveTime>], [<PeriodicTAU>]]]]".

<stat> <sup>(num)</sup>	
0	Not registered, ME is not currently searching an operator to register to.
1	Registered, home network.
2	Not registered, but ME is currently trying to attach or searching an operator to register to.
3	Registration denied.
4	Unknown, e.g. out of E-UTRAN coverage.
5	Registered, roaming.

<tac><sup>(str)</sup>  
Two byte tracking area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)

<ci><sup>(str)</sup>  
Four byte LTE cell ID in hexadecimal format.

<AcT><sup>(num)</sup>

Radio access technology

9 E-UTRAN (NB-S1 mode)

Note: 3GPP TS 36.331 specifies the System Information blocks which give the information about whether the serving cell supports NB-IoT, and which corresponds to E-UTRAN (NB-S1 mode).

<CauseType><sup>(num)</sup>

Indicates the type of <RejectCause>

0 Indicates that <RejectCause> contains an EMM cause value.

1 Indicates that <RejectCause> contains a manufacturer-specific cause.

<RejectCause><sup>(num)</sup>

Contains the cause of the failed registration. The value is of type as defined by <CauseType>.

<ActiveTime><sup>(str)</sup>

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 [33], table 10.5.163a, 3GPP TS 23.682, and 3GPP TS 23.401.

<PeriodicTAU><sup>(str)</sup>

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 [33] Table 10.5.163a/3GPP TS 24.008. See also 3GPP TS 23.682 and 3GPP TS 23.401.

## 9.6 AT+CGPADDR Show PDP Address

The [AT+CGPADDR](#) exec command returns a list of PDP addresses for all defined contexts.

The [AT+CGPADDR](#) write command returns a list of PDP addresses for the specified context identifiers. If a context is not defined, then no output line is generated for it. If no `<cid>` is specified, the addresses for all defined contexts are returned.

The [AT+CGPADDR](#) test command returns a list of defined `<cid>`s.

### Syntax

<p>Test Command</p> <pre>AT+CGPADDR=?</pre> <p>Response(s)</p> <pre>[+CGPADDR: (list of defined &lt;cid&gt;s)] OK</pre>	
<p>Exec Command</p> <pre>AT+CGPADDR</pre> <p>Response(s)</p> <pre>[+CGPADDR: &lt;cid&gt;[, &lt;PDP_address_1&gt;[, &lt;PDP_address_2&gt;]]] [+CGPADDR: ...] OK ERROR +CME ERROR: &lt;err&gt;</pre>	
<p>Write Command</p> <pre>AT+CGPADDR=&lt;cid&gt;[,&lt;cid&gt;[, ...]]</pre> <p>Response(s)</p> <pre>[+CGPADDR: &lt;cid&gt;[, &lt;PDP_address_1&gt;[, &lt;PDP_address_2&gt;]]] [+CGPADDR: ...] OK ERROR +CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>+ -</pre>	<p>Reference(s)</p> <p>3GPP 27.007</p>

### Parameter Description

`<cid>`<sup>(num)</sup>

Parameter specifies a particular PDP context definition (see [AT+CGDCONT](#) parameter `<cid>`).

`<PDP_address_1>`<sup>(str)</sup>

A string that identifies the MT in the address space applicable to the PDP. The address may be static or dynamic. If address is not available parameter is omitted.

Parameter specifies the assigned address as a dot-separated numeric (0-255) parameter of the form "a1.a2.a3.a4" for IPv4, or colon-separated hex numeric (0000-FFFF) parameter of the form "a1:a2:a3:a4:a5:a6:a7:a8" for IPV6.

---

<PDP\_address\_2><sup>(str)</sup>

A string that identifies the MT in the address space applicable to the PDP. The address may be static or dynamic. Parameter is displayed only when both IPv4 and IPv6 addresses are assigned, with <PDP\_address\_1> containing the IPv4 address and this parameter the IPv6 address. Parameter specifies the assigned IPv6 address as a colon-separated hex numeric (0000-FFFF) parameter of the form "a1:a2:a3:a4:a5:a6:a7:a8".

## 9.7 AT+CSODCP Sending of originating data via the control plane

The [AT+CSODCP](#) command is used by the TE to transmit NONIP data over control plane to network via MT.

### Syntax

<p>Test Command</p> <pre>AT+CSODCP=?</pre> <p>Response(s)</p> <pre>+CSODCP: (range of supported &lt;cid&gt;s), (maximum number of bytes of the &lt;cpdata_length&gt;), (list of supported &lt;RAI&gt;s), (list of supported &lt;type_of_user_data&gt;s) OK</pre>	
<p>Write Command</p> <pre>AT+CSODCP=&lt;cid&gt;, &lt;cpdata_length&gt;, &lt;cpdata&gt;[, &lt;RAI&gt;[, &lt;type_of_user_data&gt;]]</pre> <p>Response(s)</p> <pre>OK ERROR +CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <p>+ -</p>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38]</p>

### Parameter Description

<cid><sup>(num)</sup>

PDP Context Identifier

Parameter specifies a particular PDP context definition (see [AT+CGDCONT](#) parameter <cid>). The parameter is used in other PDP context-related commands.

0...10

<cpdata\_length><sup>(num)</sup>

cpdata length

Indicates the number of octets of the <cpdata> information element. When there is no data to transmit, the value shall be set to zero. The maximum data length is 691 bytes when NONIP is used. Data exceeding the maximum length will be discarded.

<cpdata><sup>(str)</sup>

string of octets. Contains the user data container contents (refer to 3GPP TS 24.301 subclause 9.9.4.24). When there is no data to transmit, the <cpdata> shall be an empty string (""). This parameter shall not be subject to conventional character conversion as per +CSCS. The coding format of the user data container and the maximum length of <cpdata> are implementation specific.

<RAI><sup>(num)</sup>

release assistance indication

Indicates the value of the release assistance indication. Refer to 3GPP TS 24.301 subclause 9.9.4.25.

0	No information is available.
1	The MT expects that exchange of data will be completed with the transmission of the ESM DATA TRANSPORT message.
2	The MT expects that exchange of data will be completed with the receipt of an ESM DATA TRANSPORT message.

---

<type\_of\_user\_data><sup>(num)</sup>

Indicates whether the user data that is transmitted is regular or exceptional.

- |   |                |
|---|----------------|
| 0 | Regular data   |
| 1 | Exception data |



## 9.8 AT+CRTDCP Reporting of terminating data via the control plane

The [AT+CRTDCP](#) write command is used to enable and disable reporting of data from the network to the MT that is transmitted via the control plane in downlink direction.

The [AT+CRTDCP](#) read command returns the current settings.

The [AT+CRTDCP](#) test command returns supported values as compound values.

### Syntax

<p>Test Command</p> <pre>AT+CRTDCP=?</pre> <p>Response(s)</p> <pre>+CRTDCP: (list of supported &lt;reporting&gt;s), (range of supported &lt;cid&gt;s), (maximum number of bytes of the &lt;cpdata_length&gt;)</pre> <pre>OK</pre>	
<p>Read Command</p> <pre>AT+CRTDCP?</pre> <p>Response(s)</p> <pre>+CRTDCP: &lt;reporting&gt;</pre> <pre>OK</pre>	
<p>Write Command</p> <pre>AT+CRTDCP=&lt;reporting&gt;</pre> <p>Response(s)</p> <pre>OK</pre> <pre>ERROR</pre> <pre>+CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>+ -</pre>	<p>Reference(s)</p> <pre>3GPP TS 27.007 [38]</pre>

### Unsolicited Result Code

If reporting is enabled, [<reporting>](#)=1, the MT returns the unsolicited result code:

```
+CRTDCP: <cid>, <cpdata_length>, <cpdata>
```

when data is received from the network.

### Parameter Description

<a href="#">&lt;reporting&gt;</a> <sup>(num)</sup>	controlling reporting of mobile terminated control plane data events.
0	Disable reporting of MT control plane data.
1	Enable reporting of MT control plane data by the unsolicited result code +CRTDCP.
<a href="#">&lt;cid&gt;</a> <sup>(num)</sup>	PDP Context Identifier
	Parameter specifies a particular PDP context definition (see <a href="#">AT+CGDCONT</a> parameter <a href="#">&lt;cid&gt;</a> ). The parameter is used in other PDP context-related commands.
0...10	

---

`<cpdata_length>`<sup>(num)</sup>

cpdata length

Indicates the number of octets of the `<cpdata>` information element. When there is no data to receive, the value is set to 0. The maximum data length is 1358 bytes when NONIP is used. Data exceeding the maximum length will be discarded.

`<cpdata>`<sup>(str)</sup>

string of octets. Contains the user data container contents (refer to 3GPP TS 24.301 subclause 9.9.4.24). When there is no data to receive, the `<cpdata>` is an empty string (""). This parameter shall not be subject to conventional character conversion as per +CSCS. The coding format of the user data container and the maximum length of `<cpdata>` are implementation specific.

### Notes

- Only one message is buffered at one time.
- Terminating data via the control plane before enabling reporting will be discarded.

## 9.9 AT+CNMPD No more PS data

The [AT+CNMPD](#) command indicates that no application on the MT is expected to exchange data. Upon receiving this command, the final result code OK is returned.

When in E-UTRAN, if further conditions defined in 3GPP TS 36.331 [\[41\]](#) are met, this can cause transmission of a UEAssistanceInformation message with powerPrefIndication set to "lowPowerConsumption" to the network.

### Syntax

Test Command

AT+CNMPD=?

Response(s)

OK

Exec Command

AT+CNMPD

Response(s)

OK

ERROR

+CME ERROR: <err>

PIN Last

+ +

Reference(s)

3GPP TS 27.007 [\[38\]](#)

## 9.10 AT+CGCONTRDP PDP context read dynamic parameters

The [AT+CGCONTRDP](#) write command returns dynamic parameters for the active non-secondary PDP context specified with `<cid>`.

The [AT+CGCONTRDP](#) execute command returns dynamic parameters for all active non-secondary PDP contexts.

### Syntax

#### Test Command

```
AT+CGCONTRDP=?
```

#### Response(s)

```
+CGCONTRDP: (list of supported <cid>s) associated with active contexts
OK
```

#### Exec Command

```
AT+CGCONTRDP
```

#### Response(s)

```
+CGCONTRDP: <cid>, <Bearer_ID>, <APN>[, <LocalAddr and SubNetMask>[, <GwAddr>[,
<DNS_prim_addr>[, <DNS_sec_addr>]]]]
+CGCONTRDP: <cid>, <Bearer_ID>, <APN>[, <LocalAddr and SubNetMask>[, <GwAddr>[,
<DNS_prim_addr>[, <DNS_sec_addr>]]]]
[+CGCONTRDP: ...]
OK
ERROR
+CME ERROR: <err>
```

#### Write Command

```
AT+CGCONTRDP=<cid>
```

#### Response(s)

```
+CGCONTRDP: <cid>, <Bearer_ID>, <APN>[, <LocalAddr and SubNetMask>[, <GwAddr>[,
<DNS_prim_addr>[, <DNS_sec_addr>]]]]
OK
ERROR
+CME ERROR: <err>
```

#### PIN Last

```
+ -
```

#### Reference(s)

3GPP TS 27.007 [38]

### Parameter Description

`<cid>`<sup>(num)</sup>

Specifies a particular non-secondary PDP context definition. The parameter is local to the TE-MT interface and is used in other PDP context-related commands. See [AT+CGDCONT](#).

`<Bearer_ID>`<sup>(num)</sup>

Identifies the bearer, EPS Bearer in EPS and NSAPI.

`<APN>`<sup>(str)</sup>

Access Point Name

Logical name used to select the GGSN or the external packet data network.

`<LocalAddr and SubNetMask>`<sup>(str)</sup>

Shows the IP address and subnet mask of the UE. The string is given as dot-separated numeric (0-255) parameters on the form:

"a1.a2.a3.a4.m1.m2.m3.m4" for IPv4 or

"a1.a2.a3.a4.a5.a6.a7.a8.a9.a10.a11.a12.a13.a14.a15.a16.m1.m2.m3.m4.m5.m6.m7.m8.m9.m10.m11.m12.m13.m14.m15.m16" for IPv6.

`<GwAddr>`<sup>(str)</sup>

Shows the Gateway Address of the UE. The string is given as dot-separated numeric (0-255) parameters.

`<DNS_prim_addr>`<sup>(str)</sup>

Shows the IP address of the primary DNS server.

`<DNS_sec_addr>`<sup>(str)</sup>

Shows the IP address of the secondary DNS server.

### Notes

- `<DNS_prim_addr>`,`<DNS_sec_addr>` could display only when `<PDP_type>` is "IP" or "IPV6".
- `<Bearer_ID>`,`<LocalAddr and SubNetMask>`,`<GwAddr>` could not display or display an empty string.

## 10. Short Message Service (SMS) Commands

The AT Commands described in this chapter allow an external application to use the Short Message Service with the ENS22-E.

### 10.1 SMS Parameters

#### Parameter Description

`<ackpdu>`<sup>(num)</sup>

Format is same for `<pdu>` in case of SMS, but without 3GPP TS 24.011 [32] SC address field and parameter shall be bounded by double quote characters like a normal string type parameter.

`<alpha>`<sup>(str)</sup>

String type alphanumeric representation of Destination Address or Originating Address corresponding to the entry found in phonebook. This feature is not supported.  
For ENS22-E, the `<alpha>` value is always 0.

`<length>`<sup>(num)</sup>

Message Length

Integer type value indicating in PDU mode the length of the actual TP data unit in octets.

`<mr>`<sup>(num)</sup>

Message Reference

3GPP TS 23.040 [29] TP-Message-Reference in integer format.

`<pdu>`<sup>(num)</sup>

In the case of SMS: 3GPP TS 24.011 [32] SC address followed by 3GPP TS 23.040 [29] TPDU in hexadecimal format: UE converts each octet of TP data unit into hexadecimal numbers containing two IRA characters, e.g. octet with integer value 42 is represented as two characters "2A", IRA 50 and 65 (decimal).

`<sca>`<sup>(str)(+CSCS)</sup>

Service Center Address

3GPP TS 24.011 [32] RP SC address Address-Value field in string format; BCD numbers (or GSM default alphabet characters) are converted to characters of the currently selected TE character set (`AT+CSCS`); type of address given by `<tosca>`. The `<sca>` cannot be an empty string.

`<tosca>`<sup>(num)</sup>

Type of Service Center Address

3GPP TS 24.011 [32] RP SC address Type-of-Address octet in integer format (when first character of `<sca>` is + (IRA 43) default is 145, otherwise default is 129).

## 10.2 AT+CMGC Send SMS Command

### Syntax

Test Command	
AT+CMGC=?	
Response(s)	
OK	
Write Command	
PDU mode	
AT+CMGC=<length><CR> PDU can be entered <CTRL-Z>/<ESC>	
Response(s)	
+CMGC: <mr>[, <ackpdu>]	
OK	
If sending fails	
ERROR	
+CMS ERROR: <err>	
PIN Last	Reference(s)
+ -	3GPP TS 27.005 [37]

### Notes

- After invoking the commands [AT+CMGS](#) or [AT+CMGC](#), it is necessary to wait for the prompt ">" before entering PDU.
- In general but especially at baudrates below 19200, it is recommended to only use the command line termination character ([ATS3](#)) before starting PDU input. Use of line termination character followed by an additional response formatting character ([ATS4](#)) may cause the problem that the latter will become part of the input data.
- The [<length>](#) value ranges from 8 to 174.

## 10.3 AT+CMGS Send SMS

[AT+CMGS](#) write command transmits a short message to network (SMS-SUBMIT).

After invoking the write command wait for the prompt ">" and then start to write the message. To send the message simply enter <CTRL-Z>.

To abort sending use <ESC>. Abortion is acknowledged with "OK", though the message will not be sent.

The message reference <mr> is returned by the UE on successful message delivery. The value can be used to identify the message in a delivery status report provided as an unsolicited result code.

### Syntax

Test Command	
AT+CMGS=?	
Response(s)	
OK	
Write Command	
PDU mode	
AT+CMGS=<length><CR> PDU can be entered. <CTRL-Z>/<ESC>	
Response(s)	
+CMGS: <mr>[, <ackpdu>]	
OK	
If sending fails see notes below.	
PIN Last	Reference(s)
+ -	3GPP TS 27.005 [37]

### Notes

- The <length> value ranges from 7 to 164.
- In general but especially at baudrates below 19200, it is recommended to only use the command line termination character ([ATS3](#)) before starting PDU input. Use of line termination character followed by an additional response formatting character ([ATS4](#)) may cause the problem that the latter will become part of the input data.



## 10.4 AT+CNMA New Message Acknowledgement to UE/TE

In PDU mode write command is used to send either positive (RP-ACK) or negative (RP-ERROR) acknowledgement to the network. Parameter `<n>` defines which one will be sent.

When `AT+CSMS <service>=1`, ENS22-E will not send another "+CMT" URC to the TE until previous one is acknowledged. When `AT+CSMS <service>=0`, ENS22-E will send "+CMT" URC to the TE for each incoming SMS, independent of acknowledgement of the previous SMS.

If the UE does not receive acknowledgement within required time (network timeout), it sends an "RP-ERROR" message to the network.

### Syntax

<p>Test Command</p> <pre>AT+CNMA=?</pre> <p>Response(s)</p> <pre>+CNMA: (list of supported &lt;n&gt;s) OK</pre>	
<p>Write Command</p> <pre>AT+CNMA=&lt;n&gt;[, &lt;length&gt;&lt;CR&gt; PDU can be entered &lt;CTRL-Z&gt;/&lt;ESC&gt;]</pre> <p>Response(s)</p> <pre>OK ERROR +CMS ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>+ -</pre>	<p>Reference(s)</p> <p>3GPP TS 27.005 [37]</p>

### Unsolicited Result Code

```
+CMT: [<alpha>], <length><CR><LF><pdu>
```

Indicates that new message has been received.

### Parameter Description

<code>&lt;n&gt;</code> <sup>(num)</sup>	Description
1	Send positive (RP-ACK) acknowledgement to the network. Accepted only in PDU mode.
2	Send negative (RP-ERROR) acknowledgement to the network. Accepted only in PDU mode.

### Notes

- Write command shall only be used when `AT+CSMS <service>=1` (Phase 2+), and an URC is issued by the module: `+CMT: [<alpha>], <length> <CR> <LF> <pdu>`.
- The `<length>` for CNMA in PDU mode ranges from 0 to 232.

## 10.5 AT+CSCA SMS Service Center Address

The [AT+CSCA](#) write command updates the SMSC address, through which mobile originated SMS are transmitted. In PDU mode, setting is used by the send and write commands, when the length of the SMSC address coded into the [<pdu>](#) parameter equals zero.

### Syntax

Test Command	
AT+CSCA=?	
Response(s)	
OK	
Read Command	
AT+CSCA?	
Response(s)	
+CSCA: <a href="#">&lt;sca&gt;</a> , <a href="#">&lt;tosca&gt;</a>	
OK	
Write Command	
AT+CSCA= <a href="#">&lt;sca&gt;</a> [, <a href="#">&lt;tosca&gt;</a> ]	
Response(s)	
OK	
PIN Last	Reference(s)
+ -	3GPP TS 27.005 <a href="#">[37]</a>

### Notes

- This command writes the service center address to non-volatile memory.
- The SMS service center address should be entered as specified by the service provider.
- The service center address cannot be an empty string.

## 10.6 AT+CSMS Select Message Service

### Syntax

<p>Test Command</p> <p>AT+CSMS=?</p> <p>Response(s)</p> <p>+CSMS: (list of supported&lt;service&gt;s)</p> <p>OK</p>	
<p>Read Command</p> <p>AT+CSMS?</p> <p>Response(s)</p> <p>+CSMS: &lt;service&gt;, &lt;mt&gt;, &lt;mo&gt;, &lt;bm&gt;</p> <p>OK</p>	
<p>Write Command</p> <p>AT+CSMS=&lt;service&gt;</p> <p>Response(s)</p> <p>+CSMS: &lt;mt&gt;, &lt;mo&gt;, &lt;bm&gt;</p> <p>OK</p> <p>ERROR</p> <p>+CMS ERROR: &lt;err&gt;</p>	
<p>PIN Last</p> <p>+ -</p>	<p>Reference(s)</p> <p>3GPP TS 27.005 [37]</p>

### Parameter Description

<b>&lt;service&gt;</b> <sup>(num)(&amp;V)(&amp;W)</sup>	
0(&F)(D)	3GPP TS 23.040 [29] and 3GPP TS 23.041 [30] (the syntax of SMS AT commands is compatible with 3GPP TS 27.005 [37] Phase 2 version 4.7.0; Phase 2+ features which do not require new command syntax may be supported, e.g. correct routing of messages with new Phase 2+ data coding schemes)
1	3GPP TS 23.040 [29] and 3GPP TS 23.041 [30] (the syntax of SMS AT commands is compatible with 3GPP TS 27.005 [37] Phase 2+ version; the requirement of <service> setting 1 is mentioned under corresponding command descriptions).
<b>&lt;mt&gt;</b> <sup>(num)(&amp;V)</sup>	
Mobile Terminated Messages:	
0	Type not supported
1(P)	Type supported
<b>&lt;mo&gt;</b> <sup>(num)(&amp;V)</sup>	
Mobile Originated Messages:	
0	Type not supported
1(P)	Type supported



## 11. (U)SIM related Commands

## 11. (U)SIM related Commands

AT commands described in this chapter are related to the Subscriber Identity Module ((U)SIM) connected to the ENS22-E.

### 11.1 AT+CCID (U)SIM Card Identification Number

[AT+CCID](#) serves to query the (U)SIM card identification number. This information is retrieved from (U)SIM Elementary File EF<sub>ICCID</sub>. For details refer to 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12].

#### Syntax

Test Command

```
AT+CCID=?
```

Response(s)

```
OK
```

Read Command

```
AT+CCID?
```

Response(s)

```
+CCID: <ICCID>
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

Exec Command

```
AT+CCID
```

Response(s)

```
+CCID: <ICCID>
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

#### Parameter Description

<ICCID><sup>(str)</sup>

ICCID (Integrated Circuit Card ID) of the (U)SIM card

## 11.2 AT+CRSM Restricted SIM Access

AT+CRSM offers easy access of the Elementary Files on the (U)SIM. Access to the (U)SIM database is restricted to the commands which are listed at [<command>](#).

All parameters of AT+CRSM are used as specified by 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12]. ENS22-E handles internally all required (U)SIM interface locking and file selection routines.

As response to the command, the ENS22-E sends the actual (U)SIM information parameters and response data. Error result code "+CME ERROR" may be returned if the command cannot be passed to the (U)SIM, e.g. if the (U)SIM is not inserted. However, failure in the execution of the command in the (U)SIM is reported in [<sw1>](#) and [<sw2>](#) parameters.

### Syntax

<p>Test Command</p> <pre>AT+CRSM=?</pre> <p>Response(s)</p> <pre>OK</pre>	<p>Write Command</p> <pre>AT+CRSM=&lt;command&gt;[, &lt;fileID&gt;[, &lt;P1&gt;, &lt;P2&gt;, &lt;P3&gt;[, &lt;data&gt;][, &lt;pathId&gt;]]]</pre> <p>Response(s)</p> <pre>+CRSM: &lt;sw1&gt;,&lt;sw2&gt;[,&lt;response&gt;] OK ERROR +CME ERROR: &lt;err&gt;</pre>
<p>PIN Last</p> <pre>± -</pre>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38], 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12]</p>

### Parameter Description

<b>&lt;command&gt;<sup>(num)</sup></b>	
(U)SIM command number.	
176	READ BINARY
178	READ RECORD
192	GET RESPONSE
214	UPDATE BINARY
220	UPDATE RECORD
242	STATUS
<b>&lt;fileID&gt;<sup>(num)</sup></b>	
Identifier for an elementary data file on (U)SIM, if used by <a href="#">&lt;command&gt;</a> . Mandatory for every command except STATUS and GET RESPONSE.	
<b>&lt;P1&gt;<sup>(num)</sup></b>	
Parameter to be passed on by the ENS22-E to the (U)SIM.	
0...255	

<P2><sup>(num)</sup>

Parameter to be passed on by the ENS22-E to the (U)SIM.

0...255

<P3><sup>(num)</sup>

Parameter to be passed on by the ENS22-E to the (U)SIM.

0...255

<data><sup>(str)</sup>

Information which shall be written to the (U)SIM (hexadecimal character format).

<pathId><sup>(str)</sup>

Contains the path of an elementary file on the SIM/UICC in hexadecimal format (e.g. "7F106F40").

<sw1><sup>(num)</sup>

Status information from the (U)SIM about the execution of the actual command. It is returned in both cases, on successful or failed execution of the command.

0...255

<sw2><sup>(num)</sup>

Status information from the (U)SIM about the execution of the actual command. It is returned in both cases, on successful or failed execution of the command.

0...255

<response><sup>(str)</sup>

Response data in case of a successful completion of the previously issued command. "STATUS" and "GET RESPONSE" commands return data, which gives information about the currently selected elementary data field. This information includes the type of file and its size.

After "READ BINARY" or "READ RECORD" commands the requested data will be returned.

<response> is empty after "UPDATE BINARY" or "UPDATE RECORD" commands.

## 11.3 AT+CSIM Generic (U)SIM Access

AT+CSIM allows direct control of the (U)SIM.

Compared to Restricted (U)SIM Access command AT+CRSM, the definition of AT+CSIM allows the UE to take more control over the (U)SIM interface.

For parameter and (U)SIM result coding please refer to 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12].

However, only the following (U)SIM commands are fully supported by AT+CSIM: SELECT, STATUS, READ BINARY, UPDATE BINARY, READ RECORD, UPDATE RECORD, SEEK, INCREASE and GET RESPONSE. Other (U)SIM commands will work in conjunction with the (U)SIM but the correct interaction of the (U)SIM with the ENS22-E is not guaranteed.

The TE processes SIM information within the frame specified only by basic channel.

### Syntax

<p>Test Command</p> <pre>AT+CSIM=?</pre> <p>Response(s)</p> <pre>OK</pre>	
<p>Write Command</p> <pre>AT+CSIM=&lt;length&gt;, &lt;command&gt;</pre> <p>Response(s)</p> <pre>+CSIM: &lt;length&gt;,&lt;response&gt;</pre> <pre>OK</pre> <pre>ERROR</pre> <pre>+CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>± -</pre>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38], 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12]</p>

### Parameter Description

<p>&lt;length&gt;<sup>(num)</sup></p> <p>Number of characters of &lt;command&gt; or &lt;response&gt; string.</p>
<p>&lt;command&gt;<sup>(str)</sup></p> <p>Command passed on by the UE to the (U)SIM. To query eUICCID, it is recommended to use AT^SIND="euccid",2.</p>
<p>&lt;response&gt;<sup>(str)</sup></p> <p>Response data of the command returned by the (U)SIM.</p>

### Example

The following examples show SELECT and GET RESPONSE commands for a (U)SIM card.

<pre>AT+CSIM=14, "00A40004023F00"</pre> <pre>+CSIM: 4, "6138"</pre> <pre>OK</pre>	<p>SELECT Master File (MF). Command successful. Indicates that x38 bytes response data are available and can be requested by using GET RESPONSE.</p>
---	--



---

```
AT+CSIM=10,"00C0000038"
```

```
+CSIM: 116,"6236.....A9000"  
OK
```

```
GET RESPONSE command requests x38 bytes MF  
data.  
(Response truncated here.)
```

## 11.4 AT+CCHO Open logical channel

**AT+CCHO** asks the UE to open a communication session with (U)SIM. The UICC will open a new logical channel for the selected application identified by `<dfname>` and the UE shall return `<sessionid>` as response. The UE shall restrict the communication between the TE and the UICC to this logical channel.

The `<sessionid>` shall be used when sending commands with the Generic Logical Channel access **AT+CGLA** command. It will also be used with **AT+CCHC** to close the channel.

### Syntax

Test Command

```
AT+CCHO=?
```

Response(s)

```
OK
```

Write Command

```
AT+CCHO=<dfname>
```

Response(s)

```
+CCHO: <sessionid>
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
± -
```

Reference(s)

3GPP TS 27.007 [38],  
3GPP TS 11.11 [10], 3GPP TS  
31.101 [11], 3GPP TS 31.102 [12]

### Parameter Description

`<dfname>`<sup>(str)</sup>

All selectable applications in (U)SIM are referenced by DF name coded on 1 to 16 bytes.

`<sessionid>`<sup>(num)</sup>

A session Id to be used in order to target a specific application on (U)SIM using logical channels mechanism.

## 11.5 AT+CGLA Generic logical channel access

**AT+CGLA** allows direct control of the (U)SIM by a remote application on the TE. The UE manages the channel relevant bits of the class byte depending on the `<sessionid>` and then sends the `<command>` as it is to the (U)SIM. The UE takes care of processing (U)SIM information within the frame specified by GSM/UMTS. For parameter and (U)SIM result coding please refer to 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12].

### Syntax

<p>Test Command</p> <pre>AT+CGLA=?</pre> <p>Response(s)</p> <pre>OK</pre>	
<p>Write Command</p> <pre>AT+CGLA=&lt;sessionid&gt;, &lt;length&gt;, &lt;command&gt;</pre> <p>Response(s)</p> <pre>+CGLA: &lt;length&gt;,&lt;response&gt;</pre> <pre>OK</pre> <pre>ERROR</pre> <pre>+CME ERROR: &lt;err&gt;</pre>	
<p>PIN Last</p> <pre>± -</pre>	<p>Reference(s)</p> <p>3GPP TS 27.007 [38], 3GPP TS 11.11 [10], 3GPP TS 31.101 [11], 3GPP TS 31.102 [12]</p>

### Parameter Description

<p><code>&lt;sessionid&gt;</code><sup>(num)</sup></p> <p>Identifier of the session to be used in order to send the APDU commands to the (U)SIM. The number is the <code>&lt;sessionid&gt;</code> returned by the channel open command <code>AT+CCHO</code>.</p>
<p><code>&lt;length&gt;</code><sup>(num)</sup></p> <p>Number of characters of <code>&lt;command&gt;</code> or <code>&lt;response&gt;</code> string.</p>
<p><code>&lt;command&gt;</code><sup>(str)</sup></p> <p>Command passed on by the UE to the (U)SIM.</p>
<p><code>&lt;response&gt;</code><sup>(str)</sup></p> <p>Response data of the command returned by the (U)SIM.</p>

## 11.6 AT+CCHC Close logical channel

**AT+CCHC** asks the UE to close a communication session with the (U)SIM. The UE shall close the previously opened logical channel, and no more commands can be sent on this logical channel. The UICC will close the logical channel when receiving this command.

### Syntax

Test Command

```
AT+CCHC=?
```

Response(s)

```
OK
```

Write Command

```
AT+CCHC=<sessionid>
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
± -
```

Reference(s)

3GPP TS 27.007 [38],  
3GPP TS 11.11 [10], 3GPP TS  
31.101 [11], 3GPP TS 31.102 [12]

### Parameter Description

<sessionid><sup>(num)</sup>

Session Id to be used in order to target a specific application on the (U)SIM using the logical channels mechanism. The number is the <sessionid> returned by the channel open command **AT+CCHO**.

## 12. Miscellaneous Commands

The AT Commands described in this chapter are related to various areas.

### 12.1 AT3 Command Line Termination

[AT3](#) is implemented for compatibility reasons only, and has no effect.

#### Syntax

Read Command	
AT3?	
Response(s)	
<n>	
OK	
Write Command	
AT3=<n>	
Response(s)	
OK	
ERROR	
PIN Last	Reference(s)
- -	ITU-T V.250 [9]

#### Parameter Description

<n> <sup>(num)(&amp;V)(&amp;W)</sup>
000...013 <sup>(&amp;F)</sup> ...127

#### Notes

- Command [AT3](#) is dummy, and has no effect on module behavior.
- Read command will return value written by user or default value n=13.
- Write command will return OK or ERROR. OK - when parameter <n> is in range, ERROR otherwise.
- Despite OK returned in case of any parameter from range, no changes will be made.
- Parameter is volatile.
- If changing this setting the new value has no effect for "[^SHUTDOWN](#)" URC.

## 12.2 ATS4 Response Formatting

[ATS4](#) is implemented for compatibility reasons only, and has no effect.

### Syntax

Read Command	
ATS4?	
Response(s)	
<a href="#">&lt;n&gt;</a> OK	
Write Command	
ATS4= <a href="#">&lt;n&gt;</a>	
Response(s)	
OK ERROR	
PIN Last	Reference(s)
- -	ITU-T V.250 <a href="#">[9]</a>

### Parameter Description

<a href="#">&lt;n&gt;</a> <sup>(num)(&amp;V)(&amp;W)</sup>
000...010 <sup>(&amp;F)</sup> ...127

### Notes

- Command [ATS4](#) is dummy, and has no effect on module behavior.
- Read command will return value written by user or default value n=10.
- Write command will return OK or ERROR. OK - when parameter [<n>](#) is in range, ERROR otherwise.
- Despite OK returned in case of any parameter from range, no changes will be made.
- Parameter is volatile.

## 12.3 ATS5 Command Line Editing

[ATS5](#) is implemented for compatibility reasons only, and has no effect.

### Syntax

Read Command	
ATS5?	
Response(s)	
<a href="#">&lt;n&gt;</a> OK	
Write Command	
ATS5= <a href="#">&lt;n&gt;</a>	
Response(s)	
OK ERROR	
PIN Last	Reference(s)
- -	ITU-T V.250 <a href="#">[9]</a>

### Parameter Description

<a href="#">&lt;n&gt;</a> <sup>(num)(&amp;V)(&amp;W)</sup>
000...008 <sup>(&amp;F)</sup> ...127

### Notes

- Command [ATS5](#) is dummy, and has no effect on module behavior.
- Read command will return value written by user or default value n=8.
- Write command will return OK or ERROR. OK - when parameter [<n>](#) is in range, ERROR otherwise.
- Despite OK returned in case of any parameter from range, no changes will be made.
- Parameter is volatile.

## 12.4 AT^SBNR Binary Read

### Syntax

Write Command

If **<type>**="ciphersuites": Read default values of TLS Cipher Suites for embedded TCP/IP stack:

AT^SBNR="ciphersuites", "default"

Response(s)

(default ciphers string)

OK

ERROR

+CME ERROR: **<err>**

Write Command

If **<type>**="ciphersuites": Read current TLS Cipher Suites for embedded TCP/IP stack.

AT^SBNR="ciphersuites", "current"

Response(s)

(current ciphers string)

OK

ERROR

+CME ERROR: **<err>**

Write Command

If **<type>**="is\_cert": Read certificates for secure connection of client IP services

AT^SBNR="is\_cert"

Response(s)

^SBNR:**<index>**, **<size>**, **<issuer>**, **<serial-number>**, **<subject>**, **<signature-algorithm>**,  
**<thumbprint-algorithm>**, **<thumbprint>**

[... ]

OK

ERROR

+CME ERROR: **<err>**

PIN Last

- -

### Parameter Description

**<type>**<sup>(str)</sup>

"ciphersuites"	Read TLS Cipher Suites for embedded TCP/IP stack
"is_cert"	Read certificate details See also <a href="#">AT^SISS</a> , <a href="#">AT^SIND</a> and <a href="#">AT^SBNW</a> .

**<index>**<sup>(num)</sup>

Certificate index	
0...2	Index 0 is handled as private client certificate (only 1 allowed). It is intended for client authentication and for protecting access to the (server) certificate store in the UE's NVRAM. Index 1 is handled as server certificate. Index 2 is handled as pre-loaded certificate.

**<size>**<sup>(str)</sup>

Size of certificate file in bytes
-----------------------------------



---

<issuer><sup>(str)</sup>

Certificate issuer

<serial-number><sup>(str)</sup>

Serial number of certificate

<subject><sup>(str)</sup>

Subject of certificate

<signature-algorithm><sup>(str)</sup>

Signature algorithm of certificate

<thumbprint-algorithm><sup>(str)</sup>

Thumbprint algorithm of certificate

<thumbprint><sup>(str)</sup>

Thumbprint of certificate

## 12.5 AT^SBNW Binary Write

### Syntax

Write Command

Certificate management for secure connection of client IP services

AT^SBNW=<type>, <subType>

Response(s)

CONNECT

SECURE CMD READY: SEND COMMAND ...

(Indicates that UE has entered binary data mode. Secure command data can be transferred.)

When secure command data is processed the UE will send one of following answers:

SECURE CMD LENGTH ERROR

SECURE CMD ERROR

SECURE CMD END OK

PIN Last

- -

### Parameter Description

<type><sup>(str)</sup>

“is\_cert”

Secure Command Mode (Manage Certificates)  
See also [AT^SISS](#), [AT^SIND](#) and [AT^SBNR](#).

<subType><sup>(num)</sup>

1

Start Secure Command Block Transfer mode

## 12.6 +++ Escape from Data Mode to AT Command Mode

The +++ escape sequence is only available during a packet switched connection. The +++ character sequence causes the ENS22-E to pause data mode and return to AT command mode. This allows to enter AT commands while maintaining the data connection to the remote device. The same task can be performed by toggling the DTR line if [AT&D](#) is set to 1.

To prevent the +++ character sequence from being misinterpreted as data, it must be preceded and followed by a pause of 1000 ms. The +++ characters must be entered in quick succession, all within 1000 ms.

### Syntax

Exec Command	
+++	
Response(s)	
OK	
PIN Last	Reference(s)
- -	ITU-T V.250 <a href="#">[9]</a>

## 12.7 AT^SNFWPUPDS Incremental Firmware Update

The `AT^SNFWPUPDS` command allows to update the firmware of the module via serial port using an incremental firmware update file.

### Syntax

Test Command

```
AT^SNFWPUPDS=?
```

Response(s)

```
^SNFWPUPDS: (0-5)
```

```
OK
```

Write Command

```
AT^SNFWPUPDS=<Command>[, <SN>, <Length>, <Data>, <CRC>]
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

`<Command>`<sup>(num)</sup>

This parameter determines the processes of the incremental firmware update. To download incremental firmware patch, the value must be 1.

0	Erase existing firmware package in the module's flash file system (FFS).
1	Download a firmware package segment to the module's FFS. For Download the parameters <code>&lt;SN&gt;</code> , <code>&lt;Length&gt;</code> , <code>&lt;Data&gt;</code> and <code>&lt;CRC&gt;</code> are mandatory.
2	Show validation result of the last firmware package update.
3	Show firmware package's name.
4	Show firmware package's version.
5	Start the firmware update process.

`<SN>`<sup>(num)</sup>

An integer value indicating the sequence number of firmware patch segment. It starts with 0 and increments by 1 for each package segment. This parameter is required only for `<Command>= 1`.

`<Length>`<sup>(num)</sup>

This parameter is the segment length in bytes. The length can be 32, 64, 128, 256 or 512 bytes. This parameter is required only for `<Command>= 1`.

`<Data>`<sup>(str)(NV)</sup>

The package segment data to be transmitted, in hex string format. The data length must be equal to the `<Length>` value. This parameter is required only for `<Command>= 1`.

<CRC><sup>(str)</sup>

This parameter is xor8 of each byte in every package segment. It is sent as a hex format. This parameter is required only for <Command>= 1.

## 13. Hardware related Commands

All AT commands described in this chapter are related to the hardware interface of the ENS22-E. Further information regarding this interface is available in the "ENS22-E Hardware Interface Description" [2].

### 13.1 AT+CCLK Real Time Clock

The [AT+CCLK](#) write command sets the Real Time Clock in the UE.  
The [AT+CCLK](#) read command shows the time.

#### Syntax

Test Command	
AT+CCLK=?	
Response(s)	
OK	
Read Command	
AT+CCLK?	
Response(s)	
+CCLK: <time>	
OK	
Write Command	
AT+CCLK=<time>	
Response(s)	
OK	
ERROR	
+CME ERROR: <err>	
PIN Last	Reference(s)
- -	3GPP TS 27.007 [38]

#### Parameter Description

<time><sup>(str)</sup>

Format is "yy/mm/dd,hh:mm:ss+zz", where the characters indicate the two last digits of the year, followed by month, day, hour, minutes, seconds and time zone. For example 6th of July 2016, 22:10:00 GMT+2 hours equals to "16/07/06,22:10:00+08".

Time zone tz is given as a positive (east) or negative (west) offset from UTC in units of 15 minutes: Format is "yy/mm/dd,hh:mm:ss+zz" or "yy/mm/dd,hh:mm:ss-zz".

#### Notes

- The <time> will be reset to its factory default upon power down (via AT^SMSO), power reset, or module restart. In this case, the clock starts with <time>= "70/01/01,00:00:00+00".
- Once the module is registered to the network successfully, <time> will be automatically updated with the time from the network.

## 13.2 AT^SBV Battery/Supply Voltage

[AT^SBV](#) allows to monitor the supply (or battery) voltage of the module. The voltage is periodically measured. The displayed value is averaged.

The measurement is related to the reference points of BATT+ and GND. For details on the reference points please refer to the Hardware Interface Description [2]. If the measured average voltage drops below or rises above the given voltage thresholds the UE will report alert messages by sending the "^SBC" URCs listed in Section 1.7.1, [Common URCs](#).

### Syntax

Test Command

```
AT^SBV=?
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

Exec Command

```
AT^SBV
```

Response(s)

```
^SBV: <Voltage>
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

<Voltage><sup>(num)</sup>

Supply (or battery) voltage in mV

## 13.3 AT^SCTM Critical Operating Temperature Monitoring

AT^SCTM allows to monitor the operating temperature range of the ENS22-E device. Refer to "ENS22-E Hardware Interface Description" [2] for specifications on critical temperature ranges. To avoid damage the module will shut down once the critical temperature is exceeded. The procedure is equivalent to the power-down initiated with AT^SMSO.

The AT^SCTM write command controls the presentation of URCs to report critical operating temperature limits. Use parameter <UrcMode> to enable (1) and disable (0) URC presentation.

URCs indicating alert level "-1" is intended to enable the user to take precautions, such as protect the ENS22-E from exposure to extreme conditions, or save or back up data etc. The presentation of level "-1" or "0" URCs depends on <UrcMode>.

Level "2" or "-2" URCs are followed by immediate shutdown. The presentation of these URCs is always enabled, i.e. they will be output even though <UrcMode> equals "0".

AT^SCTM read command returns:

- The URC presentation mode.
- Information about the current temperature range of the ENS22-E device.
- The board temperature (in degree Celsius) if parameter <tempCtrl>=1.

### Syntax

#### Test Command

```
AT^SCTM=?
```

```
Response(s)
```

```
^SCTM:(list of supported <UrcMode>s)[, (list of supported <tempCtrl>s)]
```

```
OK
```

#### Read Command

```
AT^SCTM?
```

```
Response(s)
```

```
^SCTM: <UrcMode>, <UrcCause>[, <temp>]
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

#### Write Command

```
AT^SCTM=<UrcMode>[, <tempCtrl>]
```

```
Response(s)
```

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

- -



### Unsolicited Result Codes

#### URC 1

URCs will be sent to the TE when the temperature reaches or exceeds the critical level, or when it is back to normal.

^SCTM\_B: <UrcCause>

#### URC 2

URC indicated after "^SCTM\_B" URC with <UrcCause> value 2 or -2 before the UE switches off.

^SHUTDOWN

### Parameter Description

<UrcMode><sup>(num)</sup>

URC presentation mode

0 <sup>(P)</sup>	Disable URC presentation (except for <UrcCause> equal to -2 or +2).
1	Enable URC presentation.

<UrcCause><sup>(num)</sup>

-2	Below lowest temperature limit (causes immediate switch-off)
-1	Below low temperature alert limit
0	Normal operating temperature
2	Above uppermost temperature limit (causes immediate switch-off)

<tempCtrl><sup>(num)</sup>

0 <sup>(P)</sup>	Suppress output of <temp> in read command.
1	Output <temp> in read command.

<temp><sup>(num)</sup>

Board temperature in Celsius. Is comprised between the lowest temperature warning level and the uppermost temperature warning level.

### Examples

#### EXAMPLE 1

URCs issued when the operating temperature is out of range:

^SCTM_B: 2	Alert: Module is above overtemperature limit and switches off.
^SCTM_B: -1	Caution: Module close to undertemperature limit.
^SCTM_B: -2	Alert: Module is below undertemperature limit and switches off.

#### EXAMPLE 2

URCs issued when the temperature is back to normal (URC is output once):

^SCTM_B: 0	Module back to normal temperature.
------------	------------------------------------

## 14. General Purpose I/O (GPIO) Pin related Commands

This chapter describes the AT commands used to access and configure the GPIO pins of ENS22-E.

Please also refer to [2] for electrical specifications of the GPIO pins.

### 14.1 AT^SPIO GPIO Driver Open/Close

`AT^SPIO` write command opens and closes the General Purpose I/O (GPIO) driver. The command must be executed before any GPIO related command can be used. The command does not reserve any GPIO lines, only the driver required for their management will be started.

#### Syntax

Test Command

```
AT^SPIO=?
```

Response(s)

```
^SPIO:(list of supported <mode>s)
```

```
OK
```

Write Command

```
AT^SPIO=<mode>
```

Response(s)

```
OK
```

```
ERROR
```

```
+CME ERROR: <err>
```

PIN Last

```
- -
```

#### Parameter Description

<mode><sup>(num)</sup>

0	Close General purpose I/O driver
1	Open General purpose I/O driver

## 14.2 AT^SCPIN Pin Configuration

The [AT^SCPIN](#) write command serves to add (or remove) and configure GPIO lines. Keep in mind that some GPIO lines share functions with other interfaces. For details see [AT^SCFG](#) "GPIO/mode/xxx" settings, Section [14.2.1, GPIO Configuration Table](#) and following AT commands:

### Syntax

Test Command

```
AT^SCPIN=?
```

Response(s)

```
^SCPIN: (list of supported <mode>s), (list of supported <pin_id>s), (list of supported <direction>s), (list of supported <startValue>s)
OK
```

Write Command

```
AT^SCPIN=<mode>, <pin_id>, <direction>[, <startValue>]
```

Response(s)

```
OK
ERROR
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

**<mode>**<sup>(num)</sup>

0	Close GPIO line
1	Open GPIO line

**<pin\_id>**<sup>(num)</sup>

Pin identifier	
0	GPIO1
1	GPIO2
2	GPIO3
3	GPIO4
4	GPIO5
7	GPIO8
8	GPIO9
9	GPIO10
15	GPIO16
16	GPIO17
17	GPIO18
18	GPIO19
23	GPIO24

<direction><sup>(num)</sup>

Parameter <direction> is mandatory when opening a GPIO line, but can be omitted when closing a GPIO line.

0	Input
1	Output

<startValue><sup>(num)</sup>

Can be set only for outputs.

[0]	Low
1	High

### Notes

- For closing a GPIO line with the write command (<mode>=0), the parameter <direction> is not needed.
- Before changing the configuration of a GPIO line be sure to close the GPIO line.

## 14.2.1 GPIO Configuration Table

Some GPIO lines are shared with other interfaces, such as ASC1. The following table shows the configuration variants. All variants are mutually exclusive, i.e. a line configured for GPIO usage is locked for alternative use and vice versa.

GPIO Pin	GPIO Handler	ASC0	ASC1
GPIO1	yes	yes	no
GPIO2	yes	yes	no
GPIO3	yes	yes	no
GPIO4	yes	no	no
GPIO5	yes	no	no
GPIO8	yes	no	no
GPIO9	yes	no	no
GPIO10	yes	no	no
GPIO16	yes	no	yes
GPIO17	yes	no	yes
GPIO18	yes	no	yes
GPIO19	yes	no	yes
GPIO24	yes	yes	no

## 14.3 AT^SGIO Get IO State of a Specified Pin

### Syntax

Test Command

```
AT^SGIO=?
```

Response(s)

```
^SGIO:(list of supported <io_id>s)  
OK
```

Write Command

```
AT^SGIO=<io_id>
```

Response(s)

```
^SGIO: <value>  
OK  
ERROR  
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

<io\_id><sup>(num)</sup>

This is an already configured <pin\_id> (see AT^SCPIN).

<value><sup>(num)</sup>

State read on this <io\_id>.

0	Low for <pin_id>
1	High for <pin_id>

## 14.4 AT^SSIO Set IO State of a Specified Pin

### Syntax

Test Command

```
AT^SSIO=?
```

Response(s)

```
^SSIO:(list of supported <io_id>s), (list of supported <value>s)  
OK
```

Write Command

```
AT^SSIO=<io_id>, <value>
```

Response(s)

```
OK  
ERROR  
+CME ERROR: <err>
```

PIN Last

```
- -
```

### Parameter Description

<io\_id><sup>(num)</sup>

This is an already configured <pin\_id> (see [AT^SCPIN](#)).

<value><sup>(num)</sup>

State to be set for this <io\_id>.

0	Low for <pin_id>
1	High for <pin_id>

## 15. Appendix

### 15.1 Restricted access to SIM data after SIM PIN authentication

The following commands can be used only after data from the SIM have been read successfully for the first time. Reading starts after successful SIM authentication has been performed, and may take up to 30 seconds depending on the SIM used. While the read process is in progress, an attempt to use any of the following commands will result in "+CME Error: 14" (SIM busy).

- ∅ ... AT Command not available
- ... AT command accessible immediately after PIN entry has returned OK
- + ... AT command fully accessible after SIM PIN authentication has been completed

AT Command	Exec	Test	Read	Write
<a href="#">AT+CSCA</a>	∅	-	+	-

## 15.2 Available AT Commands and Dependency on SIM PIN

- ∅ ... Command not available
- ... Command does not require PIN1
- + ... Command requires PIN1
- ± ... Command sometimes requires PIN1

**Table 15.1:** Available AT Commands and Dependency on SIM PIN

AT Command	Exec	Test	Read	Write
Configuration Commands				
AT&F	-	∅	∅	∅
AT&V	-	∅	∅	∅
AT&W	-	∅	∅	∅
ATQ	-	∅	∅	∅
ATV	-	∅	∅	∅
ATZ	-	∅	∅	∅
AT+CFUN	∅	-	-	-
AT^SMSO	-	-	∅	∅
AT+CMEE	∅	-	-	-
AT+CSCS	∅	-	-	-
AT^SCFG	∅	-	-	-
AT^SPOW	∅	-	-	-
Status Control Commands				
AT^SIND	∅	-	-	-
Serial Interface Control Commands				
AT\Q	-	∅	∅	∅
AT&C	-	∅	∅	∅
AT&D	-	∅	∅	∅
AT&S	-	∅	∅	∅
ATE	-	∅	∅	∅
AT+ICF	∅	-	-	-
AT+IPR	∅	-	-	-
Identification Commands				
ATI	-	∅	∅	∅
AT+CGMI	-	-	∅	∅
AT+CGMM	-	-	∅	∅
AT+CGMR	-	-	∅	∅
AT+CGSN	-	-	∅	-
AT+CIMI	+	+	∅	∅



AT Command	Exec	Test	Read	Write
Security Commands				
AT+CPIN	∅	-	-	-
AT+CLCK	∅	-	∅	-
AT+CPWD	∅	+	∅	+
AT^SSECUC	∅	∅	∅	-
Network Service Commands				
AT+COPS	∅	+	-	±
AT+CESQ	+	+	∅	∅
AT+CTZR	∅	-	-	-
AT+CPSMS	∅	-	-	-
AT^SMONI	-	-	∅	∅
AT^SMONP	-	-	∅	∅
AT+CEDRXS	∅	-	-	-
AT+CEDRXRDP	-	-	∅	∅
AT^SNLWM2M	∅	-	-	-
AT+CIPCA	∅	-	-	-
AT+CCIOTOPT	∅	-	-	-
Internet Service Commands				
AT^SICS	∅	-	-	-
AT^SICI	∅	-	-	-
AT^SIPS	∅	-	∅	-
AT^SISS	∅	-	-	-
AT^SISI	∅	-	-	-
AT^SISO	∅	-	-	+
AT^SISC	∅	-	∅	+
AT^SISR	∅	-	∅	+
AT^SISW	∅	-	∅	+
AT^SIST	∅	-	∅	-
AT^SISD	∅	-	∅	-
AT^SISU	∅	-	∅	-
AT^SISX	∅	+	∅	+
AT^SISE	∅	-	∅	-
Packet Domain Related Commands				
AT+CGACT	∅	+	+	+
AT+CGATT	∅	+	+	+
AT+CGAUTH	∅	-	-	-
AT+CGDCONT	∅	-	-	-
AT+CEREG	∅	-	-	-
AT+CGPADDR	+	+	∅	+

AT Command	Exec	Test	Read	Write
AT+CSODCP	∅	+	∅	+
AT+CRTDCP	∅	+	+	+
AT+CNMPSP			∅	∅
AT+CGCONTRDP	+	+	∅	+
Short Message Service (SMS) Commands				
AT+CMGC	∅	+	∅	+
AT+CMGS	∅	+	∅	+
AT+CNMA	∅	+	∅	+
AT+CSCA	∅	+	+	+
AT+CSMS	∅	+	+	+
(U)SIM related Commands				
AT+CCID	-	-	-	∅
AT+CRSM	∅	-	∅	+
AT+CSIM	∅	-	∅	+
AT+CCHO	∅	-	∅	+
AT+CGLA	∅	-	∅	+
AT+CCHC	∅	-	∅	+
Miscellaneous Commands				
ATS3	∅	∅	-	-
ATS4	∅	∅	-	-
ATS5	∅	∅	-	-
AT^SBNR	∅	∅	∅	-
AT^SBNW	∅	∅	∅	-
+++	-	∅	∅	∅
AT^SNFWPUPDS	∅	-	∅	-
Hardware related Commands				
AT+CCLK	∅	-	-	-
AT^SBV	-	-	∅	∅
AT^SCTM	∅	-	-	-
General Purpose I/O (GPIO) Pin related Commands				
AT^SPIO	∅	-	∅	-
AT^SCPIN	∅	-	∅	-
AT^SGIO	∅	-	∅	-
AT^SSIO	∅	-	∅	-

## 15.3 AT Command Settings storable with AT&W

**Table 15.2:** Settings Stored to User Profile

AT Command	Stored Parameters
Configuration Commands	
ATQ	<n>
ATV	<value>
AT+CMEE	<errMode>
AT+CSCS	<chset>
Serial Interface Control Commands	
AT\Q	<n>
AT&C	<value>
AT&D	<value>
AT&S	<value>
ATE	<value>
AT+ICF	<format>, <parity>
Network Service Commands	
AT+COPS	<format>
Packet Domain Related Commands	
AT+CEREG	<n>
Short Message Service (SMS) Commands	
AT+CSMS	<service>
Miscellaneous Commands	
ATS3	<n>
ATS4	<n>
ATS5	<n>

## 15.4 Factory Default Settings Restorable with AT&F

**Table 15.3:** Factory Default Settings Restorable with AT&F

AT Command	Factory Defaults
Configuration Commands	
ATQ	<n>=0
ATV	<value>=1
AT+CMEE	<errMode>=0
AT+CSCS	<chset>="GSM"
Serial Interface Control Commands	
AT\Q	<n>=3
AT&C	<value>=1
AT&D	<value>=2
AT&S	<value>=0
ATE	<value>=1
AT+ICF	<format>=3
Network Service Commands	
AT+COPS	<format>=2
Internet Service Commands	
AT^SISE	<infoMode>=0
Packet Domain Related Commands	
AT+CEREG	<n>=0
Short Message Service (SMS) Commands	
AT+CSMS	<service>=0
Miscellaneous Commands	
ATS3	<n>=013
ATS4	<n>=010
ATS5	<n>=008

## 15.5 Summary of Unsolicited Result Codes (URC)

**Table 15.4:** Summary of Unsolicited Result Codes (URC)

AT Command	URC
Unsolicited Result Code Presentation	
	<code>^SYSSTART</code>
	<code>^SBC: Undervoltage Warning</code>
	<code>^SBC: Undervoltage Shutdown</code>
	<code>^SBC: Overvoltage Warning</code>
	<code>^SBC: Overvoltage Shutdown</code>
Configuration Commands	
<code>AT^SMSO</code>	<code>^SHUTDOWN</code>
<code>AT^SCFG</code>	<code>^SYSRESUME, &lt;reason&gt;</code>
Status Control Commands	
<code>AT^SIND</code>	<code>+CIEV: &lt;indDescr&gt;, &lt;indValue&gt;</code>
<code>AT^SIND</code>	<code>+CIEV: &lt;indDescr&gt;, &lt;srvProfileId&gt;, &lt;issuer&gt;, &lt;serialNumber&gt;, &lt;subject&gt;, &lt;signatureAlgorithm&gt;, &lt;thumbprintAlgorithm&gt;, &lt;thumbprint&gt;</code>
<code>AT^SIND</code>	<code>+CIEV: &lt;indDescr&gt;, &lt;srvProfileId&gt;, &lt;pskUrl&gt;, &lt;pskId&gt;</code>
Network Service Commands	
<code>AT+CTZR</code>	<code>+CTZV: &lt;tz&gt;</code>
<code>AT+CTZR</code>	<code>+CTZE: &lt;tz&gt;, &lt;dst&gt;[, &lt;time&gt;]</code>
<code>AT+CTZR</code>	<code>+CTZEU: &lt;tz&gt;, &lt;dst&gt;[, &lt;utime&gt;]</code>
<code>AT+CEDRXS</code>	<code>+CEDRXP:&lt;AcT-type&gt;[, &lt;Requested_eDRX_value&gt;[, &lt;NW_provided_eDRX_value&gt;[, &lt;Paging_time_window&gt;]]]</code>
<code>AT^SNLWM2M</code>	<code>^SNLWM2M: "srv","dft",&lt;srv id&gt;,&lt;srv status&gt;[, &lt;status parameter&gt;]</code>
<code>AT^SNLWM2M</code>	<code>^SNLWM2M: "procedure","dft"[, &lt;srv id&gt;],&lt;procedure&gt;,&lt;procedure status&gt;[, &lt;indication text&gt;]</code>
<code>AT+CCIOTOPT</code>	<code>+CCIOTOPTI: &lt;supported_Network_opt&gt;</code>
Internet Service Commands	
<code>AT^SISR</code>	<code>^SISR: &lt;srvProfileId&gt;, &lt;urcCauseId&gt;</code>
<code>AT^SISW</code>	<code>^SISW: &lt;srvProfileId&gt;, &lt;urcCauseId&gt;</code>
Internet Service URC " <code>^SIS</code> "	<code>^SIS: &lt;srvProfileId&gt;, &lt;urcCause&gt;[, [&lt;urcInfoId&gt;][, &lt;urcInfoText&gt;]]</code>
Packet Domain Related Commands	
<code>AT+CEREG</code>	<code>+CEREG: &lt;stat&gt;</code>
<code>AT+CEREG</code>	<code>+CEREG: &lt;stat&gt;[, &lt;tac&gt;, &lt;ci&gt;[, &lt;AcT&gt;]]</code>
<code>AT+CEREG</code>	<code>+CEREG: &lt;stat&gt;[, [&lt;tac&gt;], [&lt;ci&gt;], [&lt;AcT&gt;][, &lt;CauseType&gt;, &lt;RejectCause&gt;]]</code>
<code>AT+CEREG</code>	<code>+CEREG: &lt;stat&gt;[, [&lt;tac&gt;], [&lt;ci&gt;], [&lt;AcT&gt;][, , [, [&lt;ActiveTime&gt;], [&lt;PeriodicTAU&gt;]]]]</code>

AT Command	URC
AT+CEREG	+CEREG: <stat>[, [<tac>], [<ci>], [<Act>][, [<CauseType>], [<RejectCause>][, [<ActiveTime>], [<PeriodicTAU>]]]]
AT+CRTDCP	+CRTDCP:<cid>, <cpdata_length>, <cpdata>
Short Message Service (SMS) Commands	
AT+CNMA	+CMT: [<alpha>], <length><CR><LF><pdu>
Hardware related Commands	
AT^SCTM	^SCTM_B: <UrcCause>
AT^SCTM	^SHUTDOWN

## 15.6 Alphabetical List of AT Commands

**Table 15.5:** Alphabetical List of AT Commands

AT Command	Description	Section and Page
+++	Escape from Data Mode to AT Command Mode	Section 12.6, page 195
AT&C	Set Data Carrier Detect (DCD) Line Mode	Section 4.2, page 52
AT&D	Set Data Terminal Ready (DTR) Line Mode	Section 4.3, page 53
AT&F	Reset AT Command Settings to Factory Default Values	Section 2.1, page 25
AT&S	Set Data Set Ready (DSR) Line Mode	Section 4.4, page 54
AT&V	Display current configuration	Section 2.2, page 26
AT&W	Store AT Command Settings to User Defined Profile	Section 2.3, page 28
AT+CCHC	Close logical channel	Section 11.6, page 188
AT+CCHO	Open logical channel	Section 11.4, page 186
AT+CCID	(U)SIM Card Identification Number	Section 11.1, page 181
AT+CCIOPT	CloT Optimization Configuration	Section 7.11, page 94
AT+CCLK	Real Time Clock	Section 13.1, page 198
AT+CEDRXRDP	Read dynamic eDRX parameters	Section 7.8, page 89
AT+CEDRXS	eDRX Setting	Section 7.7, page 86
AT+CEREG	EPS Network Registration Status	Section 9.5, page 162
AT+CESQ	Extended Signal Quality	Section 7.2, page 76
AT+CFUN	Functionality Level	Section 2.7, page 32
AT+CGACT	PDP context activate or deactivate	Section 9.1, page 155
AT+CGATT	PS attach or detach	Section 9.2, page 157
AT+CGAUTH	Define PDP Context Authentication Parameters	Section 9.3, page 158
AT+CGCONTRDP	PDP context read dynamic parameters	Section 9.10, page 172
AT+CGDCONT	Define PDP Context	Section 9.4, page 160
AT+CGLA	Generic logical channel access	Section 11.5, page 187
AT+CGMI	Request manufacturer identification	Section 5.2, page 61
AT+CGMM	Request model identification	Section 5.3, page 62
AT+CGMR	Request revision identification and software version	Section 5.4, page 63
AT+CGPADDR	Show PDP Address	Section 9.6, page 165
AT+CGSN	Request International Mobile Equipment Identity (IMEI)	Section 5.5, page 64
AT+CIMI	Request International Mobile Subscriber Identity (IMSI)	Section 5.6, page 66
AT+CIPCA	Initial PDP context activation	Section 7.10, page 93
AT+CLCK	Facility lock	Section 6.2, page 69
AT+CMEE	Error Message Format	Section 2.9, page 35
AT+CMGC	Send SMS Command	Section 10.2, page 175
AT+CMGS	Send SMS	Section 10.3, page 176
AT+CNMA	New Message Acknowledgement to UE/TE	Section 10.4, page 177
AT+CNMPD	No more PS data	Section 9.9, page 171
AT+COPS	Operator Selection	Section 7.1, page 73
AT+CPIN	PIN Authentication	Section 6.1, page 67
AT+CPSMS	Enable or Disable Power Saving Mode	Section 7.4, page 80

## 15.6 Alphabetical List of AT Commands

AT Command	Description	Section and Page
<a href="#">AT+CPWD</a>	Change Password	Section 6.3, page 71
<a href="#">AT+CRSM</a>	Restricted SIM Access	Section 11.2, page 182
<a href="#">AT+CRTDCP</a>	Reporting of terminating data via the control plane	Section 9.8, page 169
<a href="#">AT+CSCA</a>	SMS Service Center Address	Section 10.5, page 178
<a href="#">AT+CSCS</a>	Character Set	Section 2.10, page 39
<a href="#">AT+CSIM</a>	Generic (U)SIM Access	Section 11.3, page 184
<a href="#">AT+CSMS</a>	Select Message Service	Section 10.6, page 179
<a href="#">AT+CSODCP</a>	Sending of originating data via the control plane	Section 9.7, page 167
<a href="#">AT+CTZR</a>	Time Zone Reporting	Section 7.3, page 78
<a href="#">AT+ICF</a>	Character Framing	Section 4.6, page 56
<a href="#">AT+IPR</a>	Bit Rate	Section 4.7, page 58
<a href="#">AT\Q</a>	Flow Control	Section 4.1, page 51
<a href="#">AT^SBNR</a>	Binary Read	Section 12.4, page 192
<a href="#">AT^SBNW</a>	Binary Write	Section 12.5, page 194
<a href="#">AT^SBV</a>	Battery/Supply Voltage	Section 13.2, page 199
<a href="#">AT^SCFG</a>	Extended Configuration Settings	Section 2.11, page 40
<a href="#">AT^SCPIN</a>	Pin Configuration	Section 14.2, page 203
<a href="#">AT^SCTM</a>	Critical Operating Temperature Monitoring	Section 13.3, page 200
<a href="#">AT^SGIO</a>	Get IO State of a Specified Pin	Section 14.3, page 205
<a href="#">AT^SICI</a>	Internet Connection Information	Section 8.2, page 103
<a href="#">AT^SICS</a>	Internet Connection Setup Profile	Section 8.1, page 100
<a href="#">AT^SIND</a>	Extended Indicator Control	Section 3.1, page 47
<a href="#">AT^SIPS</a>	Internet Profile Storage	Section 8.3, page 105
<a href="#">AT^SISC</a>	Internet Service Close	Section 8.7, page 120
<a href="#">AT^SISD</a>	Internet Service MQTT Dynamic Setup Profile	Section 8.11, page 129
<a href="#">AT^SISE</a>	Internet Service Error Report	Section 8.14, page 135
<a href="#">AT^SISI</a>	Internet Service Information	Section 8.5, page 114
<a href="#">AT^SISO</a>	Internet Service Open	Section 8.6, page 116
<a href="#">AT^SISR</a>	Internet Service Read Data	Section 8.8, page 121
<a href="#">AT^SISS</a>	Internet Service Setup Profile	Section 8.4, page 107
<a href="#">AT^SIST</a>	Enter Transparent Mode	Section 8.10, page 127
<a href="#">AT^SISU</a>	MQTT Dynamic Send Request	Section 8.12, page 132
<a href="#">AT^SISW</a>	Internet Service Write Data	Section 8.9, page 124
<a href="#">AT^SISX</a>	Internet Service Execution	Section 8.13, page 133
<a href="#">AT^SMONI</a>	Monitoring Serving Cell	Section 7.5, page 82
<a href="#">AT^SMONP</a>	Monitoring Neighbour Cells	Section 7.6, page 84
<a href="#">AT^SMSO</a>	Switch Off ENS22-E	Section 2.8, page 34
<a href="#">AT^SNFWPUPDS</a>	Incremental Firmware Update	Section 12.7, page 196
<a href="#">AT^SNLWM2M</a>	Lwm2m Configuration Settings	Section 7.9, page 90
<a href="#">AT^SPIO</a>	GPIO Driver Open/Close	Section 14.1, page 202
<a href="#">AT^SPOW</a>	Set UART Mode and SLEEP Mode on UART	Section 2.12, page 46
<a href="#">AT^SSECUC</a>	Internet Service Certificate Store Update Configuration	Section 6.4, page 72



---

AT Command	Description	Section and Page
<a href="#">AT^SSIO</a>	Set IO State of a Specified Pin	Section <a href="#">14.4</a> , page <a href="#">206</a>
<a href="#">ATE</a>	AT Command Echo	Section <a href="#">4.5</a> , page <a href="#">55</a>
<a href="#">ATI</a>	Display product identification information	Section <a href="#">5.1</a> , page <a href="#">60</a>
<a href="#">ATQ</a>	Result Code Presentation Mode	Section <a href="#">2.4</a> , page <a href="#">29</a>
<a href="#">ATS3</a>	Command Line Termination	Section <a href="#">12.1</a> , page <a href="#">189</a>
<a href="#">ATS4</a>	Response Formatting	Section <a href="#">12.2</a> , page <a href="#">190</a>
<a href="#">ATS5</a>	Command Line Editing	Section <a href="#">12.3</a> , page <a href="#">191</a>
<a href="#">ATV</a>	Result code format mode	Section <a href="#">2.5</a> , page <a href="#">30</a>
<a href="#">ATZ</a>	Restore AT Command Settings from User Defined Profile	Section <a href="#">2.6</a> , page <a href="#">31</a>

## About Gemalto

Since 1996, Gemalto has been pioneering groundbreaking M2M and IoT products that keep our customers on the leading edge of innovation.

We work closely with global mobile network operators to ensure that Cinterion<sup>®</sup> modules evolve in sync with wireless networks, providing a seamless migration path to protect your IoT technology investment.

Cinterion products integrate seamlessly with Gemalto identity modules, security solutions and licensing and monetization solutions, to streamline development timelines and provide cost efficiencies that improve the bottom line.

As an experienced software provider, we help customers manage connectivity, security and quality of service for the long lifecycle of IoT solutions.

### For more information please visit

[www.gemalto.com/m2m](http://www.gemalto.com/m2m), [www.facebook.com/gemalto](https://www.facebook.com/gemalto), or [Follow@gemaltoIoT](https://twitter.com/gemaltoIoT) on Twitter.

**Gemalto M2M GmbH**  
Werinherstrasse 81  
81541 Munich  
Germany

➔ [GEMALTO.COM/M2M](http://GEMALTO.COM/M2M)

**gemalto**  
a Thales company