Developers guidelines

DEVELOPER WORLD THE FAST TRACK FROM MIND TO MARKET

October 2009

AT commands

for Sony Ericsson phones

Sony Ericsson

Preface

Purpose of this document

The Developers guideline for AT commands is designed to give the reader a deeper insight into how to design applications with AT commands supported by mobile phones. The information here is not relevant for the day-to-day operation of the phone. This is described in the User guide supplied with the mobile phone.

This document is for advanced users who require detailed information in order to:

- Develop new communications software.
- Add the mobile phone to a list of compatible modems in an application.
- Adjust the settings of their mobile phones.

People who can benefit from this document include:

- Application providers
- Content providers
- Content aggregators
- Operators and service providers
- Software developers
- Business decision-makers

It is assumed that the reader has a basic understanding of AT commands.

These Developers guidelines are published by:

Sony Ericsson Mobile Communications AB,
SE-221 88 Lund, Sweden

Phone: +46 46 19 40 00
Fax: +46 46 19 41 00
www.sonyericsson.com/

© Sony Ericsson Mobile Communications AB, 2006. All rights reserved. You are hereby granted a license to download and/or print a copy of this document.
Any rights not expressly granted herein are reserved.

14th edition (October 2009)
Publication number: 1206-6103.14

This document is published by Sony Ericsson Mobile Communications AB, without any warranty*. Improvements and changes to this text necessitated by typographical errors, inaccuracies of current information or improvements to programs and/or equipment, may be made by Sony Ericsson Mobile Communications AB at any time and without notice. Such changes will, however, be incorporated into new editions of this document. Printed versions are to be regarded as temporary reference copies only.

*All implied warranties, including without limitation the implied warranties of merchantability or fitness for a particular purpose, are excluded. In no event shall Sony Ericsson or its licensors be liable for incidental or consequential damages of any nature, including but not limited to lost profits or commercial loss, arising out of the use of the information in this document.

Sony Ericsson Developer World

At www.sonyericsson.com/developer, developers find the latest technical documentation and development tools such as phone White papers, Developers guidelines for different technologies, Getting started tutorials, SDKs (Software Development Kits) and tool plugins. The Web site also features news articles, go-to-market advice, moderated discussion forums offering free technical support and a Wiki community sharing expertise and code examples.

For more information about these professional services, go to the Sony Ericsson Developer World Web site.

Document conventions

Products

Sony Ericsson mobile phones are referred to in this document using generic names as in the table below. In this document, the chapter “AT commands” contains command specifications valid for the majority of phones mentioned. Added, deleted or changed commands for groups of phones are specified in appendices as in the rightmost column of the table.

Generic names Series	**Sony Ericsson mobile phones**	**Appendix**
Aino™	Aino™ U10i, Aino™ U10a	4
C510	C510, C510c, C510a	3
C702	C702, C702c, C702a	2
C901	C901, C901a, C901 GreenHeart™	3
C902	C902, C902c	2
C903	C903, C903a	3
C905	C905, C905c, C905a	3
G502	G502, G502c	1
G705	G705, G705u	3
Jalou™	Jalou™ F100i, BeJoo™ F100i (French market)	3
K530	K530i	
K550	K550i, K550c	
K610	K610i, K610c, K618i	

Generic names Series	**Sony Ericsson mobile phones**	**Appendix**
K630	K630i	1
K660	K660i	1
K770	K770i	
K790	K790i, K790c, K790a	
K800	K800i, K800c	
K810	K810i, K818c	
K850	K850i, K858c	1
Naite™	Naite™ J105i, Naite™ J105a	3
Pureness™	Xperia™ Pureness™ X5, Xperia™ Pureness™ X5i	3
S500	S500i, S500c	
T650	T650i, T658c	
T700	T700	2
T707	T707, T707a	3
T715	T715, T715a	3
V640	V640i	1
W350	W350i, W350c	
W380	W380i, W380c	
W508	W508, W508c, W508a, W518a	3
W580	W580i, W580c	
W595	W595, W595s	2
W610	W610i, W610c	
W705	W705, W705u	3
W715	W715	3
W710	W710i, W710c	
W760	W760i, W760c	2
W830	W830i, W830c	
W850	W850i, W850c	
W880	W880i, W888c	
W890	W890i	1
W902	W902	2
W910	W910i, W908c	1

Generic names Series	**Sony Ericsson mobile phones**	**Appendix**
W980	W980i	2
W995	W995, W995a	4
Yari™	Yari™ U100i, Yari™ U100a	4
Z310	Z310i, Z310a	
Z555	Z555i, Z555a	
Z610	Z610i	
Z710	Z710i, Z710c	
Z750	Z750i	1
Z770	Z770i	2
Z780	Z780i, Z780a	2

Typographical conventions

The standard text in this manual is modified to distinguish between the text displayed on the screen, typed instructions and examples of command dialogue. The distinctions are as follows:

- Typed commands and option values are written in bold text; for example: **S2=**<esc>; <esc>=**0-127**.
- Any key strokes are written in bold text in brackets; for example **<CR>**.
- Examples of command dialogue, including keyboard entries and on-screen responses, are written in `Courier` text.
- The default parameter setting used by a command is indicated by the text "**Default**".

Trademarks and acknowledgements

GreenHeart, Aino, BeJoo, Jalou, Naite, Pureness, Xperia and Yari are trademarks or registered trademarks of Sony Ericsson Mobile Communications AB.

Windows is either a trademark or registered trademark of Microsoft Corporation in the United States and/or other countries.

Bluetooth is a trademark or registered trademark of Bluetooth SIG Inc.

The IrDA Feature Trademark is owned by the Infrared Data Association and is used under licence there from.

Other product and company names mentioned herein may be the trademarks of their respective owners.

Document history

Change history		
2007-12-04	Doc. No. 1206-6103.1	First version published on Developer World
2008-01-20	Doc. No. 1206-6103.2	Second version. Information about W350, W760 and Z555 series added
2008-02-20	Doc. No. 1206-6103.3	Third version. Information about C702, C902, W980 and Z770 series added
2008-04-20	Doc. No. 1206-6103.4	Fourth version. Minor revision
2008-05-20	Doc. No. 1206-6103.5	Fifth version. Information about G502 and Z780 series added
2008-09-09	Doc. No. 1206-6103.6	Sixth version. Information about C905, G705, T700, W595 and W902 series added
2008-10-17	Doc. No. 1206-6103.6 (rev. B)	Sixth revised version. New document template
2008-11-09	Doc. No. 1206-6103.7	Seventh version. Information about W705 series added
2009-01-08	Doc. No. 1206-6103.8	Eighth version. Information about C510 and W508 series added
2009-02-15	Doc. No. 1206-6103.9	Ninth version. Information about C901, C903, W715 and W995 series added
2009-03-26	Doc. No. 1206-6103.10	Tenth version. Information about T707 series added
2009-05-28	Doc. No. 1206-6103.11	Eleventh version. Information about Aino™ and Yari™ series added
2009-06-25	Doc. No. 1206-6103.12	12th version. Information about Naite™ and T715 series added
2009-08-12	Doc. No. 1206-6103.13	13th version. Information about Jalou™ series added
2009-10-26	Doc. No. 1206-6103.14	14th version. Information about Pureness™ phones added

Contents

Overview **9**
Introduction 9
Result and error codes 10
AT commands 14

AT command list **19**

Result codes **24**

AT commands **26**
Ensemble C2: Control and identification 26
Ensemble C3: Call control 33
Ensemble C4: Interface commands 41
Ensemble C6: Data compression 51
Ensemble C9: Mode management 54
Ensemble C18: Fax class 1 55
Ensemble C20: Audio control 55
Ensemble C25: GSM 07.10 56
Ensemble C26: Accessory UI 58
Ensemble C27: Accessory UI 59
Ensemble C38: Bluetooth commands 80
Ensemble S1: GSM DTE-DCE interface 88
Ensemble S2: Call control 89
Ensemble S3: GSM data/fax 93
Ensemble S4: Extended error reporting 96
Ensemble S5: GSM HSCSD 97
Ensemble S6: GSM network services 103
Ensemble S7: GSM USSD 131
Ensemble S8: GSM facility lock 134
Ensemble S9: Mobile equipment, control and status 139
Ensemble S10: GSM mobile equipment error control 175
Ensemble S11: SMS and PDU mode 176
Ensemble S15: GPRS/packet domain 191
Ensemble S16: Phonebook 218
Ensemble S18: GSM clock, date and alarm handling 224
Ensemble S19: GSM subscriber information 228
Ensemble S20: Ericsson specific AT commands for GSM 229
Ensemble S26: Voice control 233
Ensemble S27: OBEX 235
Ensemble S29: WAP browser 236
Ensemble S34: Internet account commands 238
Ensemble S35: Sony Ericsson commands 272
OBEX Formats 300
Appendix 1 ..... **308**
Added AT commands 309
Updated AT commands 310
Appendix 2 ..... **313**
Added AT commands 314
Updated AT commands 315
Appendix 3 ..... **318**

Added AT commands 319
Updated AT commands 322
Appendix 4 325
Added AT commands 326
Updated AT commands 330

Index 341

Overview

Introduction

This document describes the operation of AT commands supported by Sony Ericsson phones.

This reference document is helpful for advanced users who require detailed information in order to:

- Develop new communications software.
- Add the mobile phone to a list of compatible modems in an application.
- Adjust the settings of their mobile phones.

Communications programs

Please refer to the User guide and support information found on www.sonyericsson.com for instructions on the installation and use of the Sony Ericsson built-in modem software drivers.

Configuring third-party communications programs

If you want to use a communications program which does not include the Sony Ericsson built-in modem in the list of supported hardware, the following options are suggested:

Configure for V.25ter

The built-in modem supports the V.25ter command set. If your communications program can generate and support a V.25ter command, the built-in modem does not require the installation of a specific driver.

Locate a mobile phone modem driver

A Mobile Phone Modem driver for the communications program may be available on either the Sony Ericsson disk supplied with the phone or from one of the online services, for example, the support pages on www.sonyericsson.com

Configure the data communications program manually

To configure your data communications program manually:

1. Select a generic mobile phone modem driver from the list.

2. Set the Init string to `AT&F`

3. Set the optional setup string to Asynchronous RLP: `AT+CBST=0,0,1`

Result and error codes

Result codes

When you send a command from your PC or PDA to the built-in modem, the response is terminated by a result code, which is shown on the screen of the sending device. This code is used to confirm correct operation or to identify any problem with the command. There are two types of result codes:

- Final result codes related to the operation of AT commands
- Result codes associated with call connections

Final result codes from AT commands

The built-in modem always terminates each response to an AT command with a final result code:

`OK`	The command(s) and any specified parameters were valid and the command has completed execution.

Some AT commands are not relevant to the built-in modem operations or can only be set to one parameter value. For completeness and to allow the parameter to be read, some of these commands are supported but not implemented. Calling a command of this type produces the `OK` result code but does not cause any change to the built-in modem.

`ERROR`	An error has occurred during the command processing. This could arise because: • There is a fault in the command syntax • One or more parameters are outside the permitted range • The command you issued is not implemented in the built-in modem • The command is not appropriate to the service • Of the class the built-in modem is operating in

When an error is reported, the `ERROR` message is preceded by a copy of the text response from the last valid AT command. This is shown in the following example:

Valid command:	`AT+CBC=?`
Response:	`+CBC:(0,2),(0-100)`
	`OK`
Invalid command:	`AT+CBC=?;+FCLASS=3`
Response:	`+CBC:(0,2),(0-100)`
	`ERROR`

Result codes from call connections

During online operation of the telephone, result codes inform you about the progress of call connections:

CONNECT	<speed>	A connection has been established and the data rate <speed> is shown.
BUSY		The number you called is engaged.
NO DIALTONE		Unable to establish the initial connection.
NO CARRIER		A connection could not be established or an existing connection has been lost.
RING		There is an incoming call. This is not a consequence of local activity and is referred to as an unsolicited result code.

Format of the result codes

The result codes described above are in verbose format. You can command the built-in modem to display result codes in verbose or numeric format or you can switch them off completely.

To switch between verbose and numeric format, refer to the use of the ATV command on page 45.

To switch the display of result codes on or off, refer to the use of the ATQ command on page 45.

Error codes

The +CME ERROR result codes indicate an error relating to the functionality of the built-in modem or mobile phone and replace the final result code ERROR when enabled by the AT+CMEE command.

Report mobile phone failure (+CME)

+CME ERROR: 0	Phone failure
+CME ERROR: 1	No connection to phone
+CME ERROR: 2	Phone-adaptor link reserved
+CME ERROR: 3	Operation not permitted
+CME ERROR: 4	Operation not supported
+CME ERROR: 5	PH-SIM PIN required
+CME ERROR: 6	PH-FSIM PIN required
+CME ERROR: 7	PH-FSIM PUK required
+CME ERROR: 10	SIM card not inserted
+CME ERROR: 11	SIM card PIN required
+CME ERROR: 12	SIM card PUK required
+CME ERROR: 13	SIM card failure
+CME ERROR: 14	SIM card busy
+CME ERROR: 15	SIM card wrong
+CME ERROR: 16	Incorrect password
+CME ERROR: 17	SIM PIN2 required
+CME ERROR: 18	SIM PUK2 required

+CME ERROR: 20	Memory full
+CME ERROR: 21	Invalid index
+CME ERROR: 22	Not found
+CME ERROR: 23	Memory failure
+CME ERROR: 24	Text string too long
+CME ERROR: 25	Invalid character in text string
+CME ERROR: 26	Dial string too long
+CME ERROR: 27	Invalid characters in dial string
+CME ERROR: 30	No network service
+CME ERROR: 31	Network timeout
+CME ERROR: 32	Network not allowed – emergency calls only
+CME ERROR: 40	Network personalisation PIN required
+CME ERROR: 41	Network personalisation PUK required
+CME ERROR: 42	Network subset personalisation PIN required
+CME ERROR: 43	Network subset personalisation PUK required
+CME ERROR: 44	Service provider personalisation PIN required
+CME ERROR: 45	Service provider personalisation PUK required
+CME ERROR: 46	Corporate personalisation PIN required
+CME ERROR: 47	Corporate personalisation PUK required
+CME ERROR: 100	Unknown

Report operational/access failure (+CMS)

The +CMS ERROR result codes indicate an error relating to the built-in modem, mobile phone or network relating to the Short Message Service (SMS). This replaces the final result code ERROR.

+CMS ERROR: 0...127	GSM 04.11 Annex E-2 values
+CMS ERROR: 128...255	GSM 03.40 Section 9.2.3.22 values
+CMS ERROR: 300	ME failure
+CMS ERROR: 301	SMS service of ME reserved
+CMS ERROR: 302	Operation not allowed
+CMS ERROR: 303	Operation not supported
+CMS ERROR: 304	Invalid PDU mode parameter
+CMS ERROR: 305	Invalid text mode parameter
+CMS ERROR: 310	(U)SIM card not inserted
+CMS ERROR: 311	(U)SIM PIN required
+CMS ERROR: 312	PH-(U)SIM PIN required
+CMS ERROR: 313	(U)SIM card failure
+CMS ERROR: 314	(U)SIM card busy
+CMS ERROR: 315	(U)SIM card wrong
+CMS ERROR: 316	(U)SIM PUK required
+CMS ERROR: 317	(U)SIM PIN2 required

+CMS ERROR: 318	(U)SIM PUK2 required
+CMS ERROR: 320	Memory failure
+CMS ERROR: 321	Invalid memory index
+CMS ERROR: 322	Memory full
+CMS ERROR: 330	SMSC address unknown
+CMS ERROR: 331	No network service
+CMS ERROR: 332	Network timeout
+CMS ERROR: 340	No +CNMA acknowledgement expected
+CMS ERROR: 500	Unknown error
+CMS ERROR: 256...511	Values in range 256...511 are reserved
+CMS ERROR: 512	Manufacturer specific

Service report (+CR)

When a data connection is being established, the +CR messages are sent to the PC before the final result code CONNECT. Use AT+CR to enable these messages.

+CR: ASYNC	Asynchronous transparent
+CR: SYNC	Synchronous transparent
+CR: REL ASYNC	Asynchronous non-transparent
+CR: REL SYNC	Synchronous non-transparent

Cellular result codes (+CRING)

The +CRING messages replace the unsolicited result code RING and provide more information about the type of the incoming call. Use AT+CRC to enable these messages.

+CRING: ASYNC	Asynchronous transparent
+CRING: SYNC	Synchronous transparent
+CRING: REL ASYNC	Asynchronous non-transparent
+CRING: REL SYNC	Synchronous non-transparent
+CRING: FAX	Facsimile
+CRING: VOICE	Normal voice

AT commands

Introduction to AT commands

This chapter describes how AT commands are used to exchange information with the phone, the built-in modem and Bluetooth module. The AT commands are listed at the end of this chapter. For a description of each command, refer to “AT commands” on page 26.

You use AT commands to:

- Configure the phone to connect via USB cable, infrared port, Bluetooth or the system bus.
- Configure the modem to connect via USB cable, infrared port, Bluetooth or the system bus.
- Request information about the current configuration or operational status of the phone or the modem.
- Test availability in the phone or modem and, when applicable, request the range of valid parameters for an AT command.

Built-in modem operating modes

The built-in modem can be set in any one of the following three modes of operation:

Offline command mode:	When first switched on, the built-in modem is automatically placed in offline command mode and is then ready to receive AT commands.
Online data mode:	This allows normal operation of the built-in modem, to exchange data or facsimile with a remote modem.
Online command mode:	This allows sending AT commands to the built-in modem while still remaining connected to the remote modem.

Changing the built-in modem operating mode

The following illustration summarises the methods that are used to switch between the three built-in modem operating modes:

Operating in offline command mode

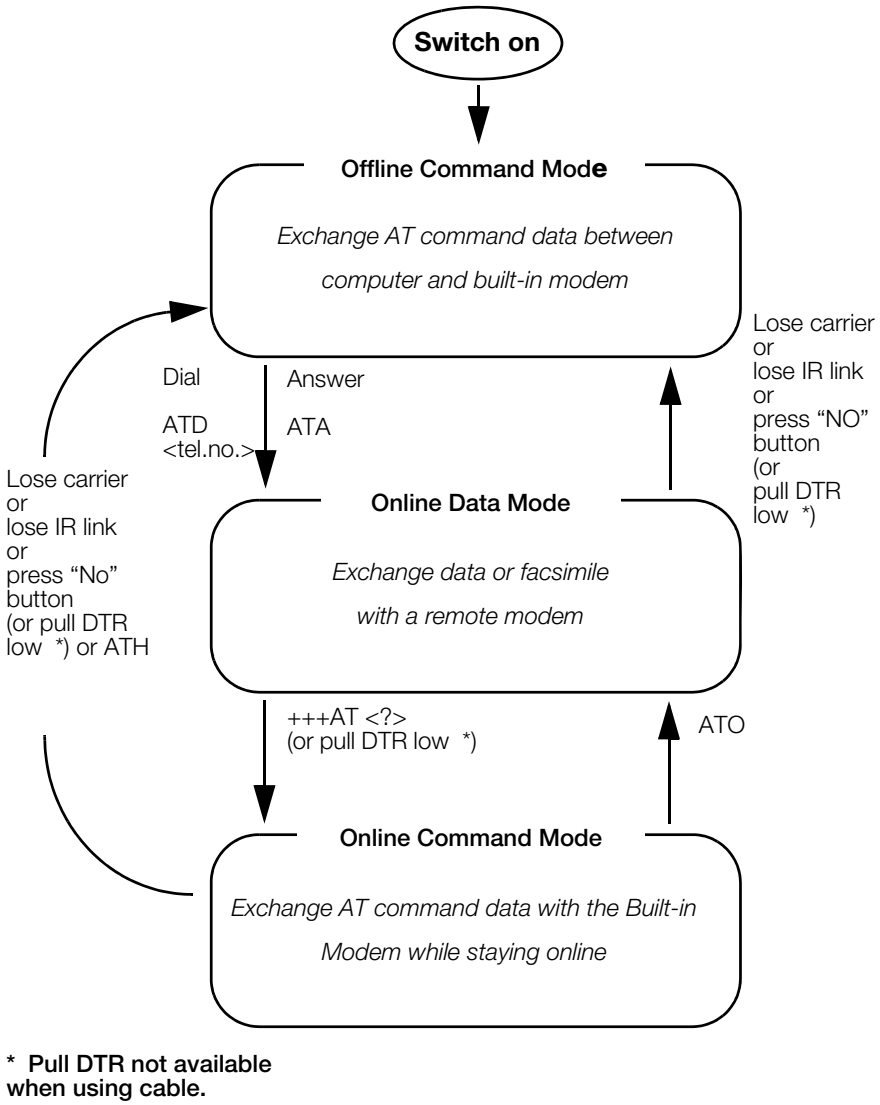

Figure 1. In offline command mode, the built-in modem accepts data as commands and not as normal communications traffic. You enter commands by typing at the PC/PDA keyboard.

Switching to online data mode

To enter online data mode, for data to be exchanged with the modem at the other end of the link, enter the **ATD** command followed by the telephone number to make the call. Alternatively, typing **ATA** to answer an incoming call also places the built-in modem in online mode.

Switching back to offline command mode

Any of the following will return the built-in modem to offline command mode from online data mode:

- Loss of connection (`NO CARRIER` error)
- Loss of the link between the built-in modem and your computer
- Pressing the "NO" button on your mobile phone
- Pulling DTR low (not available when using cable)

Using AT commands during a data connection

To use AT commands while connected to a remote modem in online data mode and maintain connection with the remote modem, first enter online command mode.

There are two ways to switch from online data mode to online command mode:

- Type the escape sequence “+++” followed by an appropriate AT command. This command must be selected from the options **AT**, **ATE**, **ATH**, **ATI**, **ATQ**, **ATV** or **ATX**. By using this method, an AT function, such as moving into online command mode, can be performed. For example, switching using

  `+++ATH<CR>`

  switches the built-in modem to online command mode. The AT command is executed, causing the connection to be terminated (hang-up executed). Typing the escape sequence “+++” without any following command causes the system to wait one second, switch to online command mode and respond `OK`;

- Pull DTR low after setting **AT&D=1**.

Switching from online command mode to online data mode

To return to online data mode while in online command mode, type:

`AT0<CR>`

Switching from online command mode to offline command mode

To return the built-in modem to offline command mode from online command mode:

- Use any of the methods described in “Switching back to offline command mode” above
- Type **+++ATH <CR>** to switch to online command mode and hang up at once.

Operating the AT commands

In command mode, the following types of commands can be issued:

- A set command to adjust the operating parameters of the built-in modem
- An execution command to direct action without any need for parameters
- A read command to view the current command settings
- A test command to view the available command parameters

Not all AT commands support all functions listed above. The descriptions in “AT commands” on page 26 list the functions available for each AT command.

1. Entering a set command

The standard format for entering a set command is:

AT<command>=<parameters>**<CR>**

where	AT	Notifies the built-in modem that a command is being entered.
	<command>	The name of the command being entered.
	<parameters>	The values to be used by the command.
	<CR>	All command lines are terminated by pressing the **<CR>** (Return or Enter) key.

Note: All command lines are completed by pressing the **<CR>** key on the computer keyboard. For the remainder of this manual, appropriate use of the **<CR>** key is assumed.

To set the built-in modem to operate with autobaud over an asynchronous connection, the command line would be:

```
AT+CBST=0,0,1
```

However, many commands also have default values. For example, the above command can be entered as:

```
AT+CBST=,,1
```

Default values used by the commands are indicated by bold text in the following descriptions.

When the parameter is a character string, for example, “<name>”, then the value should be entered between quotes, for example, “Peter”.

Optional parameters are shown in square brackets, for example, [<value>].

2. Entering an execution command

Execution commands are very similar to set commands. They usually do not require any parameters and are used to obtain information about the mobile phone or built-in modem or to execute an event.

For example, to find out information about the mobile phone battery, enter the +CBC command:

```
AT+CBC
```

The built-in modem responds:

```
CBC: 0,60
```

indicating that the mobile phone battery is connected (0) and that the remaining charge is 60%.

To answer an incoming call, you execute the A command:

```
ATA
```

3. Using read command to view command settings

To check the current settings of a command, use the ? option.

For example, to check the current settings of the +CBST command, enter:

```
AT+CBST?
```

If CBST has been set according to the previous example, the settings are displayed as

```
+CBST: 0,0,1
```

4. Using test command to request command help

To test the availability of a command and the range of parameters, use the =? option with the command. For example, to check the parameters available to the command line in the example above, enter:

```
AT+CBST=?
```

The line:

```
+CBST: (0,4,6,7,68,70,71),(0),(1)
```

is displayed indicating the range of valid entries that can be set for the parameters <data rate>, <bearer service> and <connection element>.

AT command list

AT	Attention command	26
AT*	List all supported AT commands	26
ATZ	Restore to user profile (ver. 2)	26
AT&F	Set to factory-defined configuration (ver. 2)	27
ATI	Identification information (ver. 3)	27
AT&W	Store user profile	28
AT+CLAC	List all available AT commands	28
AT+CGMI	Request manufacturer identification (ver. 1)	29
AT+CGMM	Request model identification	29
AT+CGMR	Request revision identification	30
AT+CGSN	Request product serial number identification	30
AT+GCAP	Request modem capabilities list	30
AT+GMI	Request manufacturer information	31
AT+GMM	Request model identification	31
AT+GMR	Request revision identification	32
ATA	Answer incoming call command (ver. 2)	33
ATH	Hook control (ver. 2)	33
ATD	Dial command (ver. 5)	34
ATO	Return to online data mode	36
AT+CVHU	Voice hangup control	37
AT+CLCC	List current calls	37
AT*CPI	Call progress information	39
ATE	Command echo (ver. 2)	41
ATS0	Automatic answer control	42
ATS2	Escape sequence character	42
ATS3	Command line termination character (ver. 3)	42
ATS4	Response formatting character (ver. 3)	43
ATS5	Command line editing character (ver. 3)	43
ATS7	Completion connection timeout	44
ATS10	Automatic disconnect delay control	44
ATQ	Result code suppression (ver. 2)	45
ATV	DCE response mode (ver. 2)	45
ATX	Call progress monitoring control	46
AT&C	Circuit 109 (DCD) control	47
AT&D	Circuit 108 (DTR) response	47
AT+IFC	Cable interface DTE-DCE local flow control	48
AT+ICF	Cable interface character format (ver. 2)	48
AT+IPR	Cable interface port rate	49
AT+ILRR	Cable interface local rate reporting	50
AT+DS	Data compression (ver. 3)	51
AT+DR	Data compression reporting	52
AT+WS46	Mode selection	54
AT+FCLASS	Select mode	55
AT*ECBP	CHF button pushed (ver. 2)	55
AT+CMUX	Switch to 07.10 multiplexer (ver. 2)	56
AT*EINA	Ericsson system interface active	58
AT*SEAM	Add menu item	59
AT*SESAF	SEMC show and focus	60
AT*SELERT	SEMC create alert (information text)	61

AT*SESTRI	SEMC create string Input	63
AT*SELIST	SEMC create list	65
AT*SETICK	SEMC create ticker	67
AT*SEDATE	SEMC create date field	68
AT*SEGAUGE	SEMC create gauge (bar graph/progress feedback)	70
AT*SEGUP	SEMC update gauge (bar graph/ progress feedback)	71
AT*SEONO	SEMC create on/off input	72
AT*SEYNQ	SEMC create yes/no question	73
AT*SEDEL	SEMC GUI delete	74
AT*SESLE	SEMC soft key label (ver. 1)	74
AT*SERSK	SEMC remove soft key	76
AT*SEUIS	SEMC UI session establish/terminate	76
AT*EIBA	Ericsson Internal Bluetooth address	80
AT+BINP	Bluetooth input	80
AT+BLDN	Bluetooth last dialled number	81
AT+BVRA	Bluetooth voice recognition activation	81
AT+NREC	Noise reduction and echo cancelling	82
AT+VGM	Gain of microphone	82
AT+VGS	Gain of speaker	83
AT+BRSF	Bluetooth retrieve supported	83
AT+GCLIP	Graphical caller ID presentation	84
AT+CSCS	Select TE character set (ver. 3)	88
AT+CHUP	Hang up call	89
AT+CRC	Cellular result codes (ver. 2)	89
AT+CR	Service reporting control	89
AT+CV120	V.120 rate adaption protocol	90
AT+VTS	DTMF and tone generation	91
AT+CBST	Select bearer service type (ver. 3)	93
AT+CRLP	Radio link protocol (ver. 2)	94
AT+CEER	Extended error report (ver. 2)	96
AT+CHSD	HSCSD device parameters (ver. 2)	97
AT+CHSN	HSCSD non-transparent call configuration (ver. 2)	98
AT+CHSC	HSCSD current call parameters (ver. 2)	100
AT+CHSR	HSCSD parameters report (ver. 2)	101
AT+CHSU	HSCSD automatic user-initiated upgrade	102
AT+CNUM	Subscriber number (ver. 2)	103
AT+CREG	Network registration (ver. 2)	104
AT+COPS	Operator selection (ver. 2)	106
AT+CLIP	Calling line identification (ver. 2)	108
AT+CLIR	Calling line identification restriction	109
AT+CCFC	Calling forwarding number and conditions (ver. 2)	110
AT+CCWA	Call waiting (ver. 2)	112
AT+CHLD	Call hold and multiparty (ver. 1)	113
AT+CSSN	Supplementary service notification (ver. 2)	115
AT+CAOC	Advice of charge	116
AT+CACM	Accumulated call meter (ver. 2)	117
AT+CAMM	Accumulated call meter maximum	117
AT+CDIP	Called line identification presentation	118
AT+COLP	Connected line identification presentation	119
AT+CPOL	Preferred operator list	120
AT+COPN	Read operator names	121
AT*EDIF	Divert function (ver. 2)	121
AT*EIPS	Identify presentation set	122
AT+CUSD	Unstructured supplementary service data (ver. 2)	131
AT+CLCK	Facility lock (ver. 5)	134

AT+CPWD	Change password (Ver. 3)	136
AT+CFUN	Set phone functionality (ver. 2)	139
AT+CPAS	Phone activity status (ver. 3)	140
AT+CPIN	PIN control (ver. 2)	140
AT+CBC	Battery charge (ver. 2)	142
AT+CSQ	Signal quality (ver.1)	142
AT+CKPD	Keypad control (ver. 7)	143
AT+CIND	Indicator control (ver. 5)	146
AT+CMAR	Master reset	147
AT+CMER	Mobile equipment event reporting	148
AT*ECAM	Ericsson call monitoring (ver. 2)	149
AT+CLAN	Language	151
AT*EJAVA	Ericsson Java application function	152
AT+CSIL	Silence Command	153
AT*ESKL	Key-lock mode	154
AT*ESKS	Key sound	154
AT*EAPP	Application function (ver. 5)	155
AT+CMEC	Mobile equipment control mode	159
AT+CRSM	Restricted SIM access	160
AT*EKSE	Ericsson keystroke send	162
AT+CRSL	Ringer sound level (ver. 2)	162
AT+CLVL	Loudspeaker volume level	163
AT+CMUT	Mute control	163
AT*EMEM	Ericsson memory management	164
AT+CRMP	Ring melody playback (ver. 2)	165
AT*EKEY	Keypad/joystick control (ver. 2)	166
AT*ECDF	Ericsson change dedicated file	168
AT*STKC	SIM application toolkit configuration	169
AT*STKE	SIM application toolkit envelope command send	169
AT*STKR	SIM application toolkit command response	170
AT+CMEE	Report mobile equipment error	175
AT+CSMS	Select message service (ver.2)	176
AT+CPMS	Preferred message storage (ver. 4)	177
AT+CMGF	Message format (ver. 1)	178
AT+CSCA	Service centre address (ver. 2)	179
AT+CSAS	Save settings	179
AT+CRES	Restore settings	180
AT+CNMI	New messages indication to TE (ver. 4)	180
AT+CMGL	List message (ver. 2)	182
AT+CMGR	Read message (ver. 2)	183
AT+CMGS	Send message (ver. 2)	184
AT+CMSS	Send from storage (ver. 2)	185
AT+CMGW	Write message to memory (ver. 2)	186
AT+CMGD	Delete message	187
AT+CMGC	Send command (ver. 1)	188
AT+CMMS	More messages to send	189
AT+CGDCONT	Define PDP context (ver. 1)	192
AT+CGSMS	Select service for MO SMS messages	193
AT+CGATT	Packet service attach or detach	194
AT+CGACT	PDP context activate or deactivate	194
AT+CGDATA	Enter data state	195
AT+CGEREP	Packet domain event reporting (ver. 1)	196
AT+CGREG	Packet domain network registration status	197
AT+CGPADDR	Show PDP address	197
AT+CGDSCONT	Define secondary PDP context	198

AT+CGTFT	Traffic flow template	200
AT+CGEQREQ	3G quality of service profile (requested)	203
AT+CGEQMIN	3G quality of service profile (minimum acceptable)	208
AT+CGEQNEG	3G quality of service profile (negotiated)	212
AT+CGCMOD	PDP context modify	215
Extension of ATD	– Request GPRS service	215
Extension of ATD	– Request packet domain IP service	216
AT+CPBS	Phonebook storage (ver. 3)	218
AT+CPBR	Phonebook read (ver. 2)	220
AT+CPBF	Phonebook find (ver. 2)	221
AT+CPBW	Phonebook write (ver. 4)	222
AT+CCLK	Clock (ver. 4)	224
AT+CALA	Alarm (ver. 3)	225
AT+CALD	Alarm delete	226
AT+CAPD	Postpone or dismiss an alarm (ver. 2)	226
AT*EDST	Ericsson daylight saving time	226
AT+CIMI	Request international mobile subscriber identity	228
AT*EPEE	PIN event	229
AT*EAPS	Active profile set	229
AT*EAPN	Active profile rename	230
AT*EBCA	Battery and charging algorithm (ver. 4)	230
AT*ELIB	Ericsson list Bluetooth devices	232
AT*EVAA	Voice answer active (ver. 1)	233
AT*EMWS	Magic word set	234
AT+CPROT	Enter protocol mode	235
AT*EWDT	WAP download timeout	236
AT*EWBA	WAP bookmark add (ver. 2)	237
AT*EWCT	WAP connection timeout	237
AT*EIAC	Internet account, create	241
AT*EIAD	Internet account configuration, delete	242
AT*EIAW	Internet account configuration, write general parameters	243
AT*EIAR	Internet account configuration, read general parameters	244
AT*EIAPSW	Internet account configuration, write PS bearer parameters	245
AT*EIAPSR	Internet account configuration, read PS bearer parameters	246
AT*EIAPSSW	Internet account configuration, write secondary PDP context parameters	248
AT*EIAPSSR	Internet account configuration, read secondary PDP context parameters	249
AT*EIACSW	Internet account configuration, write CSD bearer parameters	251
AT*EIACSR	Internet account configuration, read CSD bearer parameters	252
AT*EIABTW	Internet account configuration, write Bluetooth bearer parameters	253
AT*EIABTR	Internet account configuration, read Bluetooth bearer parameters	254
AT*EIAAUW	Internet account configuration, write authentication parameters	255
AT*EIAAUR	Internet account configuration, read authentication parameters	257
AT*EIALCPW	Internet account configuration, write PPP parameters – LCP	258
AT*EIALCPR	Internet account configuration, read PPP parameters – LCP	260
AT*EIAIPCPW	Internet account configuration, write PPP parameters – IPCP	263
AT*EIAIPCPR	Internet account configuration, read PPP parameters – IPCP	264
AT*EIADNSV6W	Internet account configuration, write DNS parameters – IPv6CP	266
AT*EIADNSV6R	Internet account configuration, read DNS parameters – IPv6CP	267
AT*EIARUTW	Internet account configuration, write routing table parameters	268
AT*EIARUTD	Internet account configuration, delete routing table parameters	269
AT*EIARUTR	Internet account configuration, read routing table parameters	271
AT*SEACC	Accessory class report	272
AT*SEACID	Accessory identification	273
AT*SEACID2	Accessory identification (Bluetooth)	274
AT*SEAUDIO	Accessory class report	275

AT*SECHA Charging control....277
AT*SELOG SE read log....277
AT*SEPING SE ping command....278
AT*SEAULS SE audio line status....278
AT*SEFUNC SE functionality status (ver. 2)....279
AT*SEFIN SE flash Information....280
AT*SEFEXP Flash auto exposure setting from ME....281
AT*SEMOD Camera mode indicator to the flash....281
AT*SEREDI Red eye reduction indicator to the flash....282
AT*SEFRY Ready indicator to the ME....282
AT*SEAUP Sony Ericsson audio parameters....283
AT*SEVOL Volume level....286
AT*SEVOLIR Volume indication request....286
AT*SEBIC Status bar icon....287
AT*SEANT Antenna identification....287
AT*SESP Speakermode on/off....288
AT*SETBC Text to bitmap converter....288
AT*SEAVRC Sony Ericsson audio video remote control....290
AT*SEMMIR Sony Ericsson multimedia information request....292
AT*SEMMIR ver. 1 292
AT*SEMMIR ver. 2 293
AT*SEAPP Sony Ericsson application....293
AT*SEAPPIR Sony Ericsson application indication request....294
AT*SEJCOMM Sony Ericsson Java comm....295
AT*SEDUC Sony Ericsson disable USB charge....296
AT*SEABS Sony Ericsson accessory battery status....296
AT*SEAVRCIR Sony Ericsson audio video remote control indication request....297
AT*SEGPSA Sony Ericsson global positioning system accessory....309
AT*SEAUDIO Accessory class report....310
AT*SEGPSA Sony Ericsson global positioning system accessory....314
AT*SEAUDIO Accessory Class Report....315
AT*SEGPSA Sony Ericsson global positioning system accessory....319
AT*SETIR Sony Ericsson time information request....319
AT*SEMCM Sony Ericsson memory card management....320
AT*SEAUDIO Accessory Class Report....322
AT*SEGPSA Sony Ericsson global positioning system accessory....326
AT*SETIR Sony Ericsson time information request....326
AT*SEMCM Sony Ericsson memory card management....327
AT*SEDISKL Sony Ericsson Disable Keylock....328
AT*SEAUDIO Accessory Class Report....330

Result codes

*CPII	Call progress information	40
+ILRR	+ILRR result code	51
+DR	Data compression indication	53
*SEGUII	SEMC GUI indication	77
*SESFI	SEMC session focus indication	79
*SEAAI	SEMC menu item indication	79
+BVRA	Bluetooth voice recognition activation indication	85
+VGM	Gain of microphone indication	85
+VGS	Gain of speaker indication	85
+BSIR	Bluetooth setting of in-band ring tone indication	86
+BINP	Bluetooth input indication	86
+GCLIP	Graphical caller ID presentation	87
+CME	Mobile equipment error result	91
+CR	Service reporting control	92
+CRING	Call mode indication	92
+CHSR	HSCSD parameters report result code	102
+CREG	Network registration	122
+CLIP	Calling line identification indication (ver. 2)	123
*ELIP	Calling line alpha tag	124
*EOLP	Connected line alpha tag	125
+CCWA	Call waiting notification	125
+CSSI	Supplementary service notification	125
+CSSU	Supplementary service notification	126
+CCCM	Advice of charge call meter notification	126
*EDIF	Divert function (ver. 2)	127
+COLP	Connected line identification indication	129
+CDIP	Called line identification presentation	129
+CUSD	CUSD indication	133
+CKEV	Keypad event	170
+CIEV	Indicator event	171
*ECAV	Call monitoring event	171
*STKI	SIM application toolkit command sent from SIM	172
*STKN	SIM application toolkit notify	173
+CBM	Received cell broadcast	189
+CMTI	New message indication	190
+CMT	Received message	190
+CDS	SMS status report	190
+CGEV	GPRS event reporting	216
+CGREG	Network registration reporting	217
+CALV	Alarm event	227
*EPEV	PIN code event	232
*EBCA	Indication algorithm status (ver. 1)	232
*SEFEXP	Flash auto exposure setting result code	297
*SEMOD	Camera mode indicator result code	297
*SEREDI	Red eye reduction result code	298
*SEAULSI	Audio line status result code	298
*SEFUNCI	Functionality status result code	298
*SEVOLI	Volume level result code	298
*SEMMII	Sony Ericsson Multimedia Information Indication	299

*SEAVRCI Sony Ericsson audio video remote control indication 299
*SETIRI Time Information Request Indicator 320
*SETIRI Time Information Request Indicator 329

AT commands

Ensemble C2: Control and identification

Commands

AT — Attention command

Description: Checks the communication between the phone and any accessory. Determines the presence of a phone.

Execution command: **AT**

AT* — List all supported AT commands

Description: Execution command causes the ME to return one or more lines of AT commands. The command is identical to AT+CLAC

Execution command: **AT***

Response: <AT Command1> [<CR> <LF> <AT Command2>[…]]

Test command: **AT*=?** Test if command is supported

Parameter:

<AT Command>:

<AT Command*n*>	**Description**
AT ...	AT command

ATZ — Restore to user profile (ver. 2)

Description: Instructs the DCE to set all parameters to their default values as specified by the user. It uploads a set of parameters set by AT&W. This may include taking into consideration the settings of hardware configuration switches or non-volatile parameter storage (if implemented). If AT&W is not used, ATZ gives the same effect as AT&F and ATZ can be interpreted as ATH&F.

Execution command: **ATZ**

Extended format command: **ATZ=<profile>**

Test command: **ATZ=?** Shows if the command is supported.

Test command response: Z: (list of supported <profile>s)

Parameter:

<profile>:

<profile>	Description
0	User profile to restore

AT&F Set to factory-defined configuration (ver. 2)

Description: Instructs the DCE to set all parameters to default values specified by the manufacturer, which may take in consideration hardware configuration and other manufacturer-defined criteria.

Execution command: **AT&F**[=<profile>]

Test command: **AT&F=?** Shows if the command is supported.

Test command response: &F: (list of supported <profile>s)

Parameter:

<profile>:

<profile>	Description
0	Resets all settings to factory defaults

ATI Identification information (ver. 3)

Description: Causes the DCE to transmit one or more lines of information text, determined by the manufacturer, followed by a final result code. The <value> parameter may optionally be used to select among multiple types of identifying information, specified by the manufacturer.
This command provides compatibility with Microsoft Windows 95.

Execution command: **ATI**[<value>]

Execution command response: <information>

Parameters:

<value>:

<value>	Description
0	Same information as **AT+GMM** command (model identification)
1	Software ID
3	Modem model description
5	Active settings

<value>	Description
7	Modem configuration profile (brief listing of the modem functionality: fax classes, Bluetooth, IrDA, modem type, and so on)
8	DCE hardware type version
9	PnP (Plug and Play) information
10	Same information as **AT+GMI** command (manufacturer identification)

<information>:

<information>	Description
string type	The total number of characters, including line terminators, in the information text returned in response to this command may not exceed 2048 characters. **Note**: The information text may not contain the sequence “0” or “OK”, so that DTE can avoid false detection of the end of this information text

AT&W Store user profile

Description: Stores the current user profile to non-volatile storage.

Execution command: **AT&W**[<pr>]

Test command: **AT&W=?** Shows if the command is supported.

Test command response: &W: (list of supported <pr>s)

Parameter:

<pr>:

<pr>	Description
0	Stores current settings in User Profile 0

AT+CLAC List all available AT commands

Description: Causes the ME to return one or more lines of AT commands.
Note: Only commands available to the user are returned.

Execution command: **AT+CLAC**

Response: <AT Command1> [<CR> <LF> <AT Command2>[…]]

Test command: **AT+CLAC=?** Test if command is supported

Parameter:

<AT Command>:

<AT Command*n*>	Description
AT ...	Defines the AT command including the prefix AT

AT+CGMI Request manufacturer identification (ver. 1)

Description: Causes the phone to return one or more lines of information text <manufacturer>, determined by the phone manufacturer, which is intended to permit the user of the ITAE/ETAE to identify the manufacturer of the phone to which it is connected to. Typically, the text will consist of a single line containing the name of the manufacturer, but manufacturers may choose to provide more information if desired.

Execution command: **AT+CGMI**

Execution command response: <manufacturer>

Test command: **AT+CGMI=?** Shows if the command is supported.

Parameter:

<manufacturer>:

<manufacturer>	Function
Sony Ericsson	Manufacturer name. The total number of characters, including line terminators, in the information text may not exceed 2048 characters. Text must not contain the sequence 0**<CR>** or OK**<CR>**

AT+CGMM Request model identification

Description: Causes the phone to return one or more lines of information text <model>, determined by the phone manufacturer, which is intended to permit the user of the ITAE/ETAE to identify the specific model of the phone to which it is connected to. Typically, the text will consist of a single line containing the name of the product, but manufacturers may choose to provide more information if desired.

Execution command: **AT+CGMM**

Execution command response: <model>

Test command: **AT+CGMM=?** Shows if the command is supported.

Parameters:

<model>:

<model>	Description
String Example: AAB-1022011-BV	A unique 10-character ASCII string, padded with space if needed. The response may include blank characters

AT+CGMR Request revision identification

Description: Causes the phone to return a string containing information regarding SW version.

Execution command: **AT+CGMR**

Execution command response: <revision>

Test command: **AT+CGMR=?** Shows if the command is supported.

Parameter:

<revision>:

<revision>	**Description**
String	An ASCII string containing software revision plus KRC number

AT+CGSN Request product serial number identification

Description: Returns the IMEI number of the phone.

Execution command: **AT+CGSN**

Execution command response: +CGSN:<sn>

Test command: **AT+CGSN=?** Shows if the command is supported.

Parameter:

<sn>:

<sn>	**Description**
string	Contains the phone IMEI

AT+GCAP Request modem capabilities list

Description: Returns a list of valid modem command prefixes.

Execution command: **AT+GCAP**

Execution command response: +GCAP: (list of supported <capability>s)

Test command: **AT+GCAP=?** Shows if the command is supported.

Parameter:

<capability>:

<capability>	**Description**
+CGSM	GSM commands
+FCLASS	Facsimile class 1 and 2 commands

<capability>	Description
+DS	V.42 bis compression

AT+GMI Request manufacturer information

Description: Causes the DCE to transmit one or more lines of information text, determined by the manufacturer, which is intended to permit the user of the DCE to identify the manufacturer. Typically, the text will consist of a single line containing the name of the manufacturer, but manufacturers may choose to provide more information if desired, for example, address, telephone number for customer service, and so on).

Execution command: **AT+GMI**

Execution command response: <manufacturer>

Test command: **AT+GMI=?** Shows if the command is supported.

Parameter:

<manufacturer>:

<manufacturer>	Description
string	The total number of characters, including line terminators, in the information text returned in response to this command may not exceed 2048 characters. **Note:** The information text must not contain the sequence 0 <CR> or OK<CR>, so that DTE can avoid false detection of the end of this information text

Example:

```
AT+GMI
Sony Ericsson
OK

AT+GMI=?
OK
```

AT+GMM Request model identification

Description: Causes the DCE to transmit one or more lines of information text, determined by the manufacturer, which is intended to permit the user of the DCE to identify the specific model of device. Typically, the text will consist of a single line containing the name of the product, but manufacturers may choose to provide any information desired.

Execution command: **AT+GMM**

Execution command response: <model>

Test command: **AT+GMM=?** Shows if the command is supported.

Parameter:

<model>:

<model>	Description
String Example: Sony Ericsson K750i	The total number of characters, including line terminators, in the information text returned in response to this command may not exceed 2048 characters. **Note**: The information text must not contain the sequence “0 <CR>” or “OK<CR>”, so that DTE can avoid false detection of the end of this information text. The command returns the phone model number

AT+GMR Request revision identification

Description: Causes the DCE to transmit one or more lines of information text, determined by the manufacturer, which is intended to permit the user of the DCE to identify the version, revision level or date, or other pertinent information of the device. Typically, the text will consist of a single line containing the version of the product, but manufacturers may choose to provide any information desired.
The response of this command is equal to that of the AT+CGMR command

Execution command: **AT+GMR**

Execution command response: <revision>

Test command: **AT+GMR=?** Shows if the command is supported

Parameter:

<revision>:

<revision>	Description
String (ASCII)	The total number of characters, including line terminators, in the information text returned in response to this command may not exceed 2048 characters. **Note**: The information text must not contain the sequence “0<CR>” or “OK<CR>”, so that the DTE can avoid false detection of the end of this information text

Ensemble C3: Call control

Commands

ATA Answer incoming call command (ver. 2)

Description: Answers and initiates a connection to an incoming call.

Execution command: **ATA**

Possible responses:

`CONNECT`

`CONNECT <text>`

<text>	Description
28800	Connected with data bit rate of 28800 bps (HSCSD)
19200	Connected with data bit rate of 19200 bps (HSCSD)
14400	Connected with data bit rate of 14400 bps (HSCSD)
9600	Connected with data bit rate of 9600 bps
4800	Connected with data bit rate of 4800 bps
2400	Connected with data bit rate of 2400 bps

`NO CARRIER` The mobile phone is not registered.

`ERROR` If ATA is unsuccessfully executed by the phone.

ATH Hook control (ver. 2)

Description: Signals the MS to terminate an active call.

Execution command: **ATH**

ATD Dial command (ver. 5)

Description:

Initiates a phone connection, which may be data or voice (phone number terminated by semicolon). The phone number used to establish the connection consists of digits and modifiers or a stored number specification. It is also possible to initiate a phone connection with the use of the alphanumeric field for a phonebook entry location or by the use of the entry location, <n>, itself.

AT+CPBS is the recommended command to select memory storage.
Note: Only phone and SM (SIM Memory) storage are supported by ATD.
If the dial string is followed by a semicolon, this informs the phone that the number is a voice rather than a data number.

If the dial string is omitted but the semicolon included the command instructs the phone to do a network detect. If the network is available, “OK” is returned.

Aborting an ATD command is accomplished by the transmission from the DTE to the DCE of any character. A single character is sufficient to abort the command in progress. However, characters transmitted during the first 125 milliseconds after transmission of the termination character are ignored to allow for the DTE to append additional control characters, such as line feed after the command line termination character.

Execution command:

ATD[<dial_string>][I][G][;]
Originates a call and dials the phone number specified in the command as <dial_string> or does a network detect.

ATD>ME<n>[I][G][;]
Dials the phone number stored in phone memory and is located by the index <n>.

ATD>SM<n>[I][G][;]
Dials the phone number stored on the SIM card and is located by the index <n>.

ATD>LD<n>[I][G][;]
Dials the phone number stored in the Last dialled number list on the SIM card and is located by the index <n>. The most recently dialled number is assumed to have <n>=”1”.

ATD><str>[I][G][;]
Originates a call to the phone number corresponding to the alphanumeric field <str>. If possible, all available memories are searched for the correct entry.

ATD><n>[I][G][;]
Originates call to phone number in entry location <n>. The **AT+CPBS** command setting is recommended to be used to select memory storage.
Note: Only phone and SM memory storages are supported by ATD.

ATDL[I][G][;]
Re-dials the last phone number dialled.

Execution command response:

- CONNECT
- CONNECT <text>
- NO CARRIER
- ERROR
- NO DIAL TONE
- BUSY
- OK

Parameters:

<dial_string>:

<dial_string>	**Description**
"0 1 2 3 4 5 6 7 8 9 * # + A B C"	Valid characters for origination
D	The D modifier is ignored but is included only for compatibility purposes
W	The W modifier is ignored but is included only for compatibility purposes
,	The comma modifier is ignored but is included only for compatibility purposes
T	The T modifier is ignored but is included only for compatibility purposes
P	The P modifier is ignored but is included only for compatibility purposes
!	The ! modifier is ignored but is included only for compatibility purposes
@	The @ modifier is ignored but is included only for compatibility purposes

<Final Result Code>:

<Final Result Code>	**Description**
CONNECT	Connection is successfully established. Only valid for data connections
CONNECT <text>	Connection is successfully established. Only valid for data connections
NO CARRIER	Unable to establish a connection or the connection attempt was aborted by the user
ERROR	An unexpected error occurred while trying to establish the connection
NO DIALTONE	The mobile phone is being used for a voice call or is not within coverage of the network

<Final Result Code>	Description
BUSY	The phone number called is engaged. Valid for data and voice connections
OK	Only valid for voice connections

<text>:

<text>	Description
28800	Connected with data bit rate of 28800 bps (HSCSD)
19200	Connected with data bit rate of 19200 bps (HSCSD)
14400	Connected with data bit rate of 14400 bps (HSCSD)
9600	Connected with data bit rate of 9600 bps
4800	Connected with data bit rate of 4800 bps
2400	Connected with data bit rate of 2400 bps

<str>:

<str>	Description
string type	String type value, which should be equal to an alphanumeric field in a phonebook entry in the searched memories. **Note**: The character specifying which number in the contact entry that should be used must be included in the string. “/H” stands for home number, “/M” for mobile number, and so on. The character set used should be the one selected with **AT+CSCS**

[I] [G]:

<Character>	Description
I or i	Overrides the CLIR supplementary service subscription default value for this call. I = invocation (restrict CLI presentation) and i = suppression (allow CLI presentation). See **AT+CLIR**
G or g	Controls the CUG supplementary service information for this call. G = enable CUG supplementary service and g = disable CUG supplementary service

ATO Return to online data mode

Description: Switch from online command mode to online data mode during an active call. Returns ERROR when not in online command mode.

Execution command: **ATO**[<value>]

Parameter:

<value>:

<value>	Description
0	Returns from online command state to online data state

AT+CVHU Voice hangup control

Description: Selects whether **ATH** or "drop DTR" will cause a voice connection to be disconnected or not. "Voice connection" may also refer to alternating mode calls that are currently in voice mode.
Note: When <mode> = 2, this command must be used in conjunction with the V.25ter, *Serial Asynchronous Automatic Dialing and Control*, command &D. Otherwise &D is ignored.

Set command: **AT+CVHU=**[<mode>]

Read command: **AT+CVHU?** Displays the current <mode> setting.

Test command: **AT+CVHU=?** Shows if the command is supported.

Test command response: +CVHU (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	"Drop DTR" is ignored but OK response given. ATH disconnects the call
1	"Drop DTR" and ATH are ignored but OK response is given
2	"Drop DTR" behaves according to &D setting. ATH disconnects the call

AT+CLCC List current calls

Description: Returns the list of current calls. If command succeeds but no calls are available, no information response is sent to TE.

Execution command: **AT+CLCC=**[<mode>]

Execution command response:
[+CLCC:
<id1>,<dir>,<stat>,<mode>,<mpty>[,<number>,<type>[,<alpha>[,<priority>]]] [**<CR><LF>**
+CLCC:
<id2>,<dir>,<stat>,<mode>,<mpty>[,<number>,<type>[,<alpha>[,<priority>]]]
[...]]]

Test command: **AT+CLCC=?** Shows if the command is supported.

Parameters:

<idx>:

<idx>	Description
Integer	Call identification number as described in 3GPP TS 22.030. This number can be used in **AT+CHLD** command operations

<dir>:

<dir>	Description
0	Mobile originated (MO) call
1	Mobile terminated (MT) call

<stat>:

<stat>	Description
0	Active
1	Held
2	Dialling (MO call)
3	Alerting (MO call)
4	Incoming (MT call)
5	Waiting (MT call)

<mode>:

<mode>	Description
0	Voice
1	Data
2	Fax
9	Unknown

<mpty>:

<mpty>	Description
0	Call is not one of multiparty (conference) call parties
1	Call is one of multiparty (conference) call parties

<number>:

<number>	Description
String type	String type phone number in the format specified by <type>

<type>:

<type>	Description
Integer format	Type of address octet (refer to GSM 04.08 section 10.5.4.7)
128	Unknown numbering plan, national/international number unknown
129	ISDN/telephony numbering plan, national/ international unknown

<type>	Description
145	ISDN/telephony numbering plan, international number
161	ISDN/telephony numbering plan, national number
128–255	Other values refer to GSM 04.08 section 10.5.4.7

<alpha>:

<alpha>	Description
String	Alphanumeric representation of <number> corresponding to the entry found in the phonebook. The used character set should be the one selected with command **AT+CSCS**

<priority>:

<priority>	Description
Integer	Optional digit type parameter indicating the eMLPP priority level of the call. Values are specified in 3GPP TS 22.067
0–4	Valid values

AT*CPI Call progress information

Description: Activates or deactivates unsolicited result code ***CPI**:<cId>,<msgType>,<ibt>,<tch>[,<dir>][,<mode>][,<number>][,<ton>] in the ME.
Read command reports current <n> setting and current radio access technology used.

Set command: **AT*CPI=**<n>

Read command: **AT*CPI?**

Read command response: *CPI:<n>

Test command: **AT*CPI=?** Test if the command is supported

Test command response: *CPI: (list of supported <n>s)

Parameters:

<mode>:

<mode>	Description
0	*CPI reporting disabled. **Default value**
1	*CPI reporting enabled with short list of parameters: <cId>,<msgType>,<ibt>,<tch>
2	*CPI reporting enabled with extended parameter set

Unsolicited result code

*CPII Call progress information

Description: Enabled by **AT*CPI**.

Unsolicited result code: ***CPI:** <cld>,<msgType>,<ibt>,<tch>[,<dir>[,<mode>[,<number>[,<ton>]]]]

Parameters:

<cld>:

<cld>	Description
Integer	Call identification number as described in 3GPP TS 22.030

<msgType>: Integer

<msgType>	Description
0	Setup message
1	Disconnect message
2	Alert message
3	Call proceeding message
6	Call connected message

<ibt>: Integer

<ibt>	Description
0	No in-band tones
1	In-band tones

<tch>: Integer

<tch>	Description
0	TCH not assigned
1	TCH assigned

<dir>: Integer

<dir>	Description
0	Mobile originated call
1	Mobile terminated call

<mode>: Integer

<mode>	Description
0	Voice
1	Data

<number>:

<cld>	Description
String type	Phone number of format specified by <ton>

<ton>: Integer. Type of address octet (refer GSM 04.08 section 10.5.4.7).

<ton>	Description
129	ISDN/telephony numbering plan, national/ international unknown. **Default value if "+" is not in <sca>**
145	ISDN/telephony numbering plan, international number. **Default value if "+" is in <sca>**
161	ISDN/telephony numbering plan, national number
128–255	Other values refer GSM 04.08 section 10.5.4.7

Ensemble C4: Interface commands

Commands

ATE Command echo (ver. 2)

Description: Determines if the DCE echoes characters received from the DTE during command state and online command state.

Set command: **ATE**[<value>]

Read command: **ATE?** Displays the current <value> setting.

Test command: **ATE=?** Shows if the command is supported.

Test command response: E: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
0	DCE does not echo characters during command state and online command state
1	DCE echoes characters during command state and online command state. **Default value**

ATS0 Automatic answer control

Description: Defines the automatic answering feature of the modem. A non-zero value specifies the number of rings before the call is answered.
Note: The call always answers in the current fax class, regardless of whether the incoming call is voice, data or fax.

Set command: **ATS0=**[<rcnt>]

Read command: **ATS0?** Displays the current <rcnt> setting.

Test command: **ATS0=?** Shows if the command is supported.

Test command response: S0: (list of supported <rcnt>s)

Parameter:

<rcnt>:

<rcnt>	**Description**
0	Disable automatic answer. **Default value**
1–7	Answer after the specified number of rings

ATS2 Escape sequence character

Description: Defines the character to be used as the escape sequence character when switching from online data mode to online command mode.

Set command: **ATS2=**[<esc>]

Parameter:

<esc>:

<esc>	**Description**
0–255	Supported values. **Note**: If the **<esc>** parameter is set to a value in the range 128–255, the escape sequence detection is disabled
43	Escape sequence character = “+”. **Default value**

ATS3 Command line termination character (ver. 3)

Description: This S-parameter represents the decimal IA5 value of the character recognised by the DCE from the DTE to terminate an incoming command line. It is also generated by the DCE as part of the header, trailer and terminator for result codes and information text, along with the **S4** parameter.
The previous value of **S3** is used to determine the command line termination character for entry of the command line containing the **S3** setting command. However, the result code issued will use the value of **S3** as set during the processing of the command line. For example, if S3 was previously set to 13 and the command line “**ATS3=30**” is issued, the command line will be terminated with a <CR> character (IA5 0/13), but the result code issued will use the character with the ordinal value 30 (IA5 2/14) in place of the <CR>.

Set command: **ATS3=**<value>

Read command: **ATS3?** Displays the current <value> setting.

Test command: **ATS3=?** Shows if the command is supported.

Test command response: S3: (list of supported <value>s)

Parameter:

<value>:

<value>	**Description**
0–127	Supported values
13	Command line termination character = **<CR>**. **Default value**

ATS4 Response formatting character (ver. 3)

Description: This S-parameter represents the decimal IA5 value of the character generated by the DCE as part of the header, trailer and terminator for result codes and information text, along with the **S3** parameter.
If the value of **S4** is changed in a command line, the result code issued in response to that command line will use the new value of **S4**.

Set command: **ATS4=**<value>

Read command: **ATS4?** Displays the current <value> setting.

Test command: **ATS4=?** Shows if the command is supported.

Test command response: S4: (list of supported <value>s)

Parameter:

<value>:

<value>	**Description**
0–127	Supported values
10	Formatting character = **<LF>**. **Default value**

ATS5 Command line editing character (ver. 3)

Description: This S-parameter represents the decimal IA5 value of the character recognised by the DCE as a request to delete from the command line the immediately preceding character.

Set command: **ATS5=**<value>

Read command: **ATS5?** Displays the current <value> setting.

Test command: **ATS5=?** Shows if the command is supported.

Test command response: S5: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
0–127	Supported values
8	Editing character = **<BS>** (Backspace). **Default value**

ATS7 Completion connection timeout

Description: Defines the maximum time allowed between completion of dialling and the connection being established. If this time is exceeded, the connection is aborted.

Set command: **ATS7=**[<tmo>]

Read command: **ATS7?** Displays the current <tmo> setting.

Test command: **ATS7=?** Shows if the command is supported.

Test command response: S7: (list of supported <tmo>s)

Parameter:

<tmo>:

<tmo>	Description
1–255	Possible timeout values in seconds
50	Timeout value in seconds. **Default value**

ATS10 Automatic disconnect delay control

Description: Specifies the amount of time the DCE will remain connected to the line after the absence of received line signal.
Note: For mobile phones this is not applicable and the command is ignored by the TE. This command is included for compatibility reasons only.

Set command: **ATS10=**[<value>]

Read command: **ATS10?** Displays the current <value> setting.

Test command: **ATS10=?** Shows if the command is supported.

Test command response: S10: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
1–254	Delay, specified in tenths of a second
2	Remains connected for two tenths of a second. **Default value**

ATQ Result code suppression (ver. 2)

Description: The setting of this parameter determines whether or not the DCE transmits result codes to the DTE. When result codes are being suppressed, no portion of any intermediate, final or unsolicited result code (header, result text, line terminator or trailer) is transmitted.

Set command: **ATQ**[=]<value>

Read command: **ATQ?** Displays the current <value> setting.

Read command response: Q: <value>

Test command: **ATQ=?** Shows if the command is supported.

Test command response: Q: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
0	DCE transmits result codes. **Default value**
1	Result codes are suppressed and not transmitted

ATV DCE response mode (ver. 2)

Description: Selects either verbose or numeric response codes.

Set command: **ATV**[=]<value>

Read command: **ATV?** Displays the current <value> setting.

Read command response: V: <value>

Test command: **ATV=?** Shows if the command is supported.

Test command response: V: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
0	Display numeric result code
1	Display verbose result code. **Default value**

Result code (ATV1)	Result code (ATV0)	Description
OK	**0**	Acknowledges execution of a command
CONNECT	**1**	A connection has been established. The DCE is moving from command state to online data state

Result code (ATV1)	Result code (ATV0)	Description
RING	**2**	The DCE has detected an incoming call from the network
NO CARRIER	**3**	The connection has been terminated or the attempt to establish a connection failed
ERROR	**4**	Command not recognised, command line maximum length exceeded, parameter value invalid or other problem with processing the command line
NO DIALTONE	**6**	No dial tone detected.
BUSY	**7**	Engaged (busy) signal detected
NO ANSWER	**8**	"@" (Wait for Quiet Answer) dial modifier was used, but remote ringing followed by five seconds of silence was not detected before expiration of the connection timer, **S7**

ATX Call progress monitoring control

Description: Defines the format of the CONNECT message and if the BUSY and NO DIALTONE result codes will be used during a data call setup.
Not applicable for voice calls.

Set command: **ATX**=[<n>] or **ATX**[<n>]

Read command: **ATX?** Displays the current <n> setting.
X:<n>

Test command: **ATX=?** Shows if the command is supported.

Test command response: X: (list of supported <n>s)

Parameter:

<n>:

<n>	Description
0	Neither BUSY nor NO DIALTONE result code is given. No line speed reported together with CONNECT result code
1	As for <n>=0, but reports line speed together with CONNECT result code
2	BUSY result code is not given. NO DIALTONE result code returned if no network. Reports line speed together with CONNECT result code

<n>	Description
3	BUSY result code given if called line is busy. NO DIALTONE result code is not given. Reports line speed together with the CONNECT result code
4	BUSY result code given if called line is busy. NO DIALTONE result code returned if no network. Reports line speed together with CONNECT result code. **Default value**

AT&C Circuit 109 (DCD) control

Description: Determines the behaviour of the carrier detect signal (CT109).

Set command: **AT&C**[=][<value>]

Read command: **AT&C?** Displays the current <value> setting.

Read command response: &C: <value>

Test command: **AT&C=?** Shows if the command is supported.

Test command response: &C: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
0	DCD always on
1	DCD follows the connection. **Default value**

AT&D Circuit 108 (DTR) response

Description: Controls how the DCE responds when the Data Terminal Ready (DTR) signal (ct 108.2) is changed from on to off condition.

Set command: **AT&D**[=][<value>]

Read command: **AT&D?** Displays the current <value> setting.

Read command response: &D: <value>

Test command: **AT&D=?** Shows if the command is supported.

Test command response: &D: (list of supported <value>s)

Parameter:

<value>:

<value>	Description
0	Ignore. **Default value**
1	When in online data mode: Switch to online command mode. All other states: Disconnect and switch to offline command mode

<value>	Description
2	Disconnect and switch to offline command mode

AT+IFC Cable interface DTE-DCE local flow control

Description: Defines the flow control between the modem and the computer when in online data mode. No flow control is enabled in any of the command modes.

Set command: **AT+IFC**=[<by_te>,[<by_ta>]]

Read command: **AT+IFC?** Displays the current <by_te> and <by_ta> settings.

Read command response: +IFC: <by_te>,<by_ta>

Test command: **AT+IFC=?** Shows if the command is supported.

Test command response: +IFC: (list of supported <by_te>s),(list of supported <by_ta>s)

Parameters:

<by_te>:

<by_te>	Description
0	No flow control on DTE
1	Xon/Xoff flow control on DCE. Control characters are removed by the DCE interface
2	RTS flow control on DCE. **Default value**
3	Xon/Xoff flow control on DCE. Control characters are passed to the remote DCE/DTE

<DTE_by_DCE>:

<DTE_by_DCE>	Description
0	No flow control on DCE
1	Xon/Xoff flow control on DTE
2	CTS flow control on DCE. **Default value**

AT+ICF Cable interface character format (ver. 2)

Description: This extended-format compound parameter is used to determine the local serial port start/stop (asynchronous) character framing that the DCE will use while accepting DTE commands and while transmitting information text and result code, if this is not automatically determined. AT+IPR=0 forces +ICF=0 (see **AT+IPR**).
Note: Only applicable for RS-232, dummy command on IrDA and USB.

Set command: **AT+ICF**=[<format>[,<parity>]

Read command: **AT+ICF?** Displays the current <format> and <parity> settings.

Read command response: +ICF: <format>,<parity>

Test command: **AT+ICF=?** Shows if the command is supported.

Test command response: +ICF: (list of supported <format>s),(list of supported <parity>s)

Parameters:

<format>: Determines the number of data bits, parity bits and stop bits in the start/stop frame.

<format>	Description
0	Auto-detect
1	8 Data bits, 2 Stop bits
2	8 Data bits, 1 Parity bit, 1 Stop bit
3	8 Data bits, 1 Stop bit. **Default value**
4	7 Data bits, 2 Stop bits
5	7 Data bits, 1 Parity bit, 1 Stop bit
6	7 Data bits, 1 Stop bit

<parity>: Determines how the parity bit is generated and checked, if present.

<parity>	Description
0	Odd
1	Even
2	Mark
3	Space, **Default value**

AT+IPR Cable interface port rate

Description: This numeric extended-format parameter specifies the data rate at which the DCE will accept commands, in addition to 1200 bit/s or 9600 bit/s (as required in v25ter subclause 4.3). It may be used to select operation at rates at which the DCE is not capable of automatically detecting the data rate being used by the DTE.
Specifying a value of 0 disables the function and allows operation only at rates automatically detectable by the DCE. The specified rate takes effect following the issuance of any result code(s) associated with the current command line.
Note: Only applicable for RS-232, dummy command on IrDA and USB.

Set command: **AT+IPR**=[<rate>]

Read command: **AT+IPR?** Displays the current <rate> setting.

Test command: **AT+IPR=?** Shows if the command is supported.

Test command response: +IPR: (list of supported auto detectable <rate>s)[,(list of fixed-only <rate>s)].

Parameter:

<rate>: The <rate> value specified will be the rate in bits per second at which the DTE-DCE interface should operate, for example, “19200” or “115200”. If unspecified or set to 0, automatic detection is selected for the range determined by v25ter, subclause 4.3 and the character format is also forced to autodetect, (**AT+ICF**=0).

<rate> (bps)	Autodetect possible	Manual setting possible
0 Default value	Yes	Yes
300	Yes	Yes
600	Yes	Yes
1200	Yes	Yes
2400	Yes	Yes
3600	Yes	Yes
4800	Yes	Yes
7200	Yes	Yes
9600	Yes	Yes
14400	Yes	Yes
19200	Yes	Yes
28800	Yes	Yes
38400	Yes	Yes
57600	Yes	Yes
115200	Yes	Yes
230400	Yes	Yes
460800	Yes	Yes

AT+ILRR Cable interface local rate reporting

Description: Specifies whether or not the **+ILRR** intermediate result code is transmitted from the DCE to the DTE. The <rate> reported represents the current (negotiated or renegotiated) DTE-DCE rate. If enabled, the intermediate result code is transmitted after any modulation, error control or data compression reports are transmitted and before any final result code, for example, CONNECT is transmitted. The <rate> is applied after the final result code is transmitted.
The DTE-DCE port rate will change only if neither buffered mode nor error controlled means are enabled (+ES=x,0) and if the negotiated carrier rate (+MRR) does not match the current DTE-DCE port rate (set by +IPR command or autodetected from the previous command line).

Set command: **AT+ILRR**=<value>

Read command: **AT+ILRR?** Displays the current <value> setting.

Test command: **AT+ILRR=?** Shows if the command is supported.

Test command response: +ILRR: (list of supported auto detectable <value>s)

Parameter:

<value>:

<value>	Description
0	Disables reporting of local port rate. (+ILRR: is not transmitted) **Default value**
1	Enables reporting of local port rate. (+ILRR: is transmitted)

Intermediate result codes

+ILRR +ILRR result code

Description: Reports cable interface speed and represents the current DTE-DCE rate. This response is enabled by **AT+ILRR**.

Intermediate result code: +ILRR: <rate>[,rx_rate>]

Parameter:

<rate>

<rate>	Description
Numeric	See command AT+IPR for possible values of <rate>

<rx_rate>

<rx_rate>	Description
Numeric	Same coding as <rate>, used in case RX rate is different from TX rate

Ensemble C6: Data compression

Commands

AT+DS Data compression (ver. 3)

Description: Controls the V.42 bis data compression function, if provided in the phone. **Note:** This command is only applicable to CS (Circuit Switched) data calls.

Set command: **AT+DS=**[<direction>[,<compression_negotiation>[,<max_dict>[,<max-string>]]]]

Read command: **AT+DS?** Displays the current <direction>, <compression_negotiation>, <max_dict> and <max_string> settings.

Test command: **AT+DS=?** Shows if the command is supported.

Test command response: +DS: (list of supported <direction>s),(list of supported <compression_negotiation>s),(list of supported <max_dict>s),(list of supported <max_string>s)

Parameters:

<direction>: Specifies the desired direction(s) of operation of the data compression function from the TE point of view.

<direction>	Description
0	Disable V.42 bis
1	Enable V.42 bis in transmit direction only
2	Enable V.42 bis in receive direction only
3	Enable V.42 bis compression in both directions. **Default value**

<compression_negotiation>: Specifies if the phone should continue to operate if the desired result is not obtained.

<compression_negotiation>	Description
0	Do not disconnect if compression is not negotiated according to direction. **Default value**
1	Disconnect if compression is not negotiated according to direction

<max_dict>: Maximum number of dictionary entries to be negotiated.

<max_dict>	Description
512 to 2048	Maximum dictionary size **Note**: Must be given in multiples of 512
1024	**Default value**

<max_string>: Maximum string length to be negotiated.

<max_string>	Description
6 to 250	Maximum string length
32	**Default value**

AT+DR Data compression reporting

Description: Controls whether or not the extended-format **+DR** intermediate result code is transmitted from the phone to the terminal equipment.The +DR: <type> reported represents the current (negotiated or renegotiated) TAE-TE data compression type.
If enabled, the intermediate result code is transmitted after error control negotiation (handshaking), when the TAE has determined which data compression technique will be used (if any) and the direction of operation.
Note: This command is only applicable to CS (Circuit Switched) data calls.

Set command: **AT+DR=**<value>

Read command: **AT+DR?** Displays the current <value> setting.

Test command: **AT+DR=?** Shows if the command is supported.

Test command response: +DR: (list of supported <values>s)

Parameter:

<value>:

<value>	Description
0	Intermediate compression mode reporting disabled. **Default value**
1	Intermediate compression mode reporting enabled

Intermediate result codes

+DR Data compression indication

Description: Data compression report. Enabled by using **AT+DR**.

Intermediate result code: +DR: <type>

Parameter:

<type>:

<type>	Description
NONE	No data compression negotiated
V42B	V.42 bis data compression negotiated
V42B RD	V.42 bis half duplex compression negotiated on received data
V42B TD	V.42 bis half duplex compression negotiated on transmitted data

Ensemble C9: Mode management

Commands

AT+WS46 Mode selection

Description: Allows an accessory to query and control the cellular protocol mode of a multimode phone.
AT+WS46=<n> allows an accessory to set the cellular protocol mode of a multimode phone. The setting remains in effect until another AT+WS46=<n> command is issued, the phone is reset, a call is terminated or the phone itself makes a mode change.
Note: Not all cellular protocol modes can be set via AT+WS46=<n>. ERROR is returned if an attempt is made to set an MS into a mode that cannot be set via the system bus.
Supported values of <n> are unique for every Sony Ericsson phone product. Any procedures needed to change from the old to the new mode must complete prior to generation of the OK result code.

Set command: **AT+WS46=**<n>

Read command: **AT+WS46?** Displays the current <n> setting.

Test command: **AT+WS46=?** Shows if the command is supported.

Test command response: WS46: (list of supported <n>s)

Parameter:

<n>: Supported values differ between phone models.

<n>	Description
12	GSM Digital Cellular. This value is used for GSM at 900 Mhz, DCS-1800 and PCS-1900 phones
22	Wideband CDMA. This parameter is used by 3G phones

Ensemble C18: Fax class 1

Commands

AT+FCLASS Select mode

Description: Puts the TA into a particular mode of operation (data, fax, voice, and so on). This causes the TAE to process information in a manner suitable for that type of information.
Reset + hangup on failed ATA sets mode to class 0, data mode.

Execution command: **AT+FCLASS=**<n>

Read command: **AT+FCLASS?**

Read command response: <n> Show list of supported services

Test command: **AT+FCLASS=?** Shows if the command is supported.

Test command response: (list of supported <n>s)

Parameter:

<n>:

<n>	Description
0	Data modem

Ensemble C20: Audio control

Commands

AT*ECBP CHF button pushed (ver. 2)

Description: This command is used by the cordless handsfree (CHF) to indicate to the phone that a button on the CHF has been pushed.

Action command: **AT*ECBP**[=<button>[,<time>]]

Test command: **AT*ECBP=?** Shows if the command is supported.

Test command response: *ECBP: (list of supported <button>s),(list of supported <time>s)

Parameters:

<button>:

<button>	Description
1	Button pressed on HBH-10 handsfree. **Default value**
2	First (“YES”) button pressed on HBH-20 handsfree
3	Second (“NO”) button pressed on HBH-20 handsfree

<time>:

<time>	Description
1	Short press. **Default value**
2	Long press

Ensemble C25: GSM 07.10

Commands

AT+CMUX Switch to 07.10 multiplexer (ver. 2)

Description: Turns on the 07.10 multiplexer.

Set command: **AT+CMUX=**<transparency>[,<subset>[,<port_speed>[,<N1>[,<T1>[,<N2>[,<T2>[,<T3>[,<k>]]]]]]]]

Read command: **AT+CMUX?** Displays the current <transparency>, <subset>, <port_speed>, <N1>, <T1>, <N2>, <T2>,<T3> and [<k>] settings.

Read command response

+CMUX:<transparency>,<subset>,<port_speed>,<N1>,<T1>,<N2>,<T2>,<T3>[,<k>]

Test command: **AT+CMUX=?** Shows if the command is supported.

Test command response:

+CMUX: (list of supported <transparency>s),(list of supported <subset>s),(list of supported <port_speed>s),(list of supported <N1>s),(list of supported <T1>s),(list of supported <N2>s),(list of supported <T2>s),(list of supported <T3>s), (list of supported <k>)

Parameters:

<transparency>:

<transparency>	Description
0	No transparency. **Default value**

<subset>:

<subset>	Description
0	Only UIH frames used. **Default value**

<port_speed>:

<port_speed>	Description
1	9600 bps
2	19200 bps
3	38400 bps
4	57600 bps
5	115200 bps
6	230400 bit/s
7	460800 bit/s

<N1>:

<N1>	Description
31	Maximum frame size. **Default value**

<T1>:

<T1>	Description
10	100 ms acknowledgement timer. **Default value**

<N2>:

<N2>	Description
3	Maximum number of retransmissions. **Default value**

<T2>:

<T2>	Description
30	300 ms control channel response timer. **Default value**

<T3>:

<T3>	Description
10	10 s wake up response timer. **Default value**

<k>: This parameter is not used

Ensemble C26: Accessory UI

Commands

AT*EINA — Ericsson system interface active

Description: Returns the active interface, that is, the interface currently used for communication

Get command: **AT*EINA** Get active interface

Get command response: *EINA: <interface>

Test command: **AT*EINA=?** Shows if the command is supported.

Test command response: *EINA: (list of supported <interface>s)

Parameters:

<interface>:

<interface>	**Description**
1	System connector
2	IR **Not supported**
3	Bluetooth
4	USB

Ensemble C27: Accessory UI

Commands

AT*SEAM Add menu item

Description: Adds the persistent menu item to the phone menu structure and assigns a category to this menu item. The menu is placed in one of the categories specified by <category>. It is possible for an accessory to add a persistent menu item to more than one category. This command should then be called once for every new menu item. When the phone receives this AT command, it must:

- Create the additional menu if it is not already present.
- Add an item with the text specified in <persistent menu item text>.
- Upon successful request, the phone answers with a <menu_ID>. This ID is being used when sending unsolicited *SEAAI to the accessory.
- When the user selects the menu an unsolicited result code *SEAAI sent.
- If the phone receives a new persistent menu item when there already exist a persistent menu the old menu item is deleted and a new created.
- If the accessory is disconnected, the corresponding accessory menu item is deleted. If there are no items in the additional menu the additional menu is deleted.

Execution command: **AT*SEAM=**<persistent menu item text>[,<category>]

Execution command response: *SEAM:<menu_id>

Test command: **AT*SEAM=?** Shows if the command is supported.

Parameters:

<persistent_menu_item_text>:

<persistent_menu_item_text>	**Description**
String type	The menu item text in the additional menu

<category>:

<category>	**Description**
Integer type	
0	Connectivity (placed directly under Connectivity)
1	Bluetooth
2	Entertainment
3	Messaging
4	Organiser
5	Settings – General

<category>	Description
6	Settings – Sounds and alarms
7	Settings – Display
8	Settings – Calls
9	Multimedia
10	Imaging
11	Phonebook
12	Applications (downloaded applications)
13	Accessories. **Default value**
14–256	Reserved for further use

<menu_id>:

<menu_id>	Description
Integer type	The menu ID sent to the accessory

AT*SESAF SEMC show and focus

Description: Demands focus for a specific object with object index <object_index>.

Execution command: **AT*SESAF=**<object_index>[,<form>]

Test command: **AT*SESAF=?** Shows if the command is supported.

Test command response: *SESAF: (list of supported <form>s)

Parameters:

<object_index>:

<object_index>	Description
Integer type	Index of the object for which focus is wanted

<form>:

<form>	Description
Integer type	The object ID for the form in which the object is placed
0	The object is not placed in a form. **Default value**
1–255	The object ID for the form

AT*SELERT SEMC create alert (information text)

Description: Displays an alert dialog via the phone UI. An alert is a dialog that shows data to the user and waits for the user to respond to the information. An alert can contain text (string) and an icon. Alerts inform the user about errors and other exceptional conditions.
When calling AT*SELERT with a timeout (<alert_type> = 6) and the given amount of time has elapsed, an unsolicited result code ***SEGUII** is sent to the accessory.
The dialog is **not** be removed when the timeout has been reached. It is the callers responsibility to listen to the unsolicited result code and act upon it.
If a user presses the left softkey (OK), ***SEGUII** is also sent.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SELERT. An alert dialog is deleted if **AT*SEDEL=<object_index>** is called or the UI session connected to the object is destroyed (AT*SEUIS=0).

Set command:
With <alert_type> <= 5:
AT*SELERT=<alert_text>,<alert_type>,<show>[,<title>][,<time_out>][,<image>]
With <alert_type> = 6:
AT*SELERT=<alert_text>[,[<alert_type>],[<show>],[<timeout>]]

Set command response: *SELERT:<object_index>

Test command: **AT*SELERT=?** Shows if the command is supported.

Test command response:
Range of general parameters:
*SELERT: ((list of supported <alert_type>s),(list of supported <show>s),(list of supported <timeout>s),(list of supported <icon_id>s)

Parameters:

<title>:

<title>	Description
String	Title of the alert. **Note**: Not valid for <alert_type> = 6 (Text feedback)

<alert_text>:

<alert_text>	Description
String	Text to be included in the alert

<alert_type>:

<alert_type>	Description
0	NONE
1	ALARM An ALARM AlertType alerts the user to an event for which the user has previously requested to be notified. For example, the message might say, “Staff meeting in five minutes”

<alert_type>	Description
2	CONFIRMATION A CONFIRMATION AlertType confirms user actions. For example, “Saved!” might be shown to indicate that a Save operation has completed
3	ERROR An ERROR AlertType alerts the user to an erroneous operation. For example, an error alert might show the message, “There is not enough room to install the application”
4	INFO An INFO AlertType provides information to the user. For example, a simple splash screen might be an INFO AlertType
5	WARNING A WARNING AlertType warns the user of a potentially dangerous operation. For example, the warning message may contain the text, “Warning: this operation will erase your data”
6	TEXT FEEDBACK A text feedback is a popup box with a message to the user. **Note**: No <title> should be given It is possible to set a timeout in the <timeout> parameter

<image>:

<icon_id>	Description
Integer	The specific ID number of the icon that should be displayed

<object_index>:

<object_index>	Description
Integer type	Index of the alert

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will **not** be displayed when created
1	Object will be displayed when created

<timeout>:

<timeout>	Description
Integer type	Timeout telling how long the text feedback should be displayed. Value given in ms. 0 (zero) value gives infinite time. **Note**: Only valid for <alert_type> = 6 (Text feedback)

AT*SESTRI SEMC create string Input

Description: Displays a string input dialogue via the phone UI. The contents in the dialogue can be changed by the user. The accessory will be notified when a user action has occurred and the unsolicited result code ***SEGUII** is then sent.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SESTRI.
A string input dialog is deleted if **AT*SEDEL=<object_index>** is called.

Execution command: **AT*SESTRI=**<title>,<prompt_text>,<default_text>,<predictive>,<input_mode>,<show>[,<form>]

Execution command response: *SESTRI:<object_index>

Test command: **AT*SESTRI=?** Shows if the command is supported.

Test command response: Range of general parameters:
*SESTRI: <maxsize>,(list of supported <input_mode>s),
(list of supported <show>s), (list of supported <form>s)

Parameters:

<title>:

<title>	Description
String	Title of the dialogue

<prompt_text>:

<prompt_text>	Description
String	The text string to be put as prompt text in front of the text editing area of the dialogue

<default_text>:

<default_text>	Description
String	The text string to be put in the text editing area of the dialogue. The type of text input is determined by the value of the <predictive> parameter

<predictive>:

<predictive>	Description
Bitflags	Flags used to control the behaviour of a string object
Bit 1 (LSB)	Password – Entered text is confidential data. Content must never be divulged to the user
Bit 2	Edit disallowed – User must not edit the text
Bit 3	Sensitive data – Entered text is sensitive data. Must never be stored, for example, credit card number
Bit 4	Non-predictive – Predictive input facilities are disallowed
Bit 5	Initial CAPS word – Initial letter of each word should be capitalised
Bit 6	Initial CAPS sentence – Initial letter of each sentence should be capitalised

<input_mode>:

<input_mode>	Description
Integer	Constraints that are put on the <text> parameter – what type of text should be entered in the dialogue
0	Any
1	Real input
2	Integer input
3	Phone number input
4	URL input
5	Email input

<maxsize>:

<maxsize>	Description
Integer	The maximum number of characters in <text>

<object_index>:

<object_index>	Description
Integer type	Index of the dialogue

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will not be displayed when created
1	Object will be displayed when created

<form>:

<form>	Description
Integer type	Tells whether the GUI object should be placed in a form or not
0	The object should not be placed in a form. **Default value**
1	The object should be placed in a form

AT*SELIST SEMC create list

Description:

Implements a List object and can also be used to create a submenu.
A List can consist of many items. Each item is composed of a text string and an optional image. If the item does not have an image connected to it the user must specify "" for the <item_image>. If an image is provided, the implementation may choose to ignore the image if it exceeds the capacity of the device to display it. If the implementation displays the image, it will be displayed adjacent to the text string and the pair will be treated as a unit.
Images within any particular List object should all be of the same size, because the implementation is allowed to allocate the same amount of vertical space for every element.
When creating a new List object, the user has to select which type of List, <list_type>, that should be implemented, one-of-many, data list type or Nbr-of-many list type.
When the user has made a selection in the list, the indexes of the selected menu items are returned with the unsolicited result code ***SEGUII**. The index is one-based (the first item has index 1). Other user actions such as rejecting the list are also sent in the unsolicited ***SEGUII**.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SELIST.
A list object dialog is deleted if **AT*SEDEL=<object_index>** is called.

Execution command:

AT*SELIST=<title>,<list_type>,<item_to_focus>,<number_of_items>,<overlay_style>,<show>,<item_string1>,<item_image1>,<dimmed1>,<selected1>,<delete1>[,<item_string2>,<item_image2>,<dimmed2>,<selected2>,<delete2>...]

Execution command response:

*SELIST: <object_index>

Test command:

AT*SELIST=? Shows if the command is supported.

Test command response:

*SELIST: (list of supported <list_type>s),(list of supported <overlay_style>s),(list of supported <show>s)

Parameters:

<title>:

<title>	Description
String	Title of the list

<list_type>:

<list_type>	Description
1	One-of-many. The user must select one and only one item
2	Nbr-of-many The user can select many items in the list
3	Data list This list can be used as a menu

<selected>:

<selected>	Description
Integer type	Tells if the item is selected or not. **Note:** For a one-of-many list only one item can be selected
0	Item is not selected
1	Item is selected

<item_to_focus>:

<item_to_focus>	Description
Integer type	The item in the list that will be in focus when the list is shown

<number_of_items>:

<number_of_items>	Description
Integer type	Number of list items

<item_string>:

<item_string>	Description
String	Name of an item in the list

<item_image>:

<item_image>	Description
Integer type	ID of the image to be displayed with the list item

<object_index>:

<object_index>	Description
Integer type	Index of the list

<dimmed>:

<dimmed>	Description
Integer type	Tells whether the list item will be dimmed (grey-colored, not accessible) or not
0	List item will not be dimmed
1	List item will be dimmed

<delete>:

<delete>	Description
Integer type	Tells whether a specific list item in the list can be deleted by the user
0	List item cannot be deleted by the user
1	List item can be deleted by the user

<overlay_style>:

<overlay_style>	Description
Integer type	Tells how the GUI object should be presented
0	Overlay style not defined
1	Overlay style default. Use original frame settings
2	No frame
3	Frame
4	Fullscreen with frame
5	Fullscreen without frame

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will not be displayed when created
1	Object will be displayed when created

AT*SETICK SEMC create ticker

Description: Implements a “ticker tape”, a piece of text that runs continuously on the display of the ME.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before running this command. A Ticker object is deleted if **AT*SEDEL=<object_index>** is called.
Note: Creating a ticker locks the UI until **AT*SEUIS=0** has been called.

Execution command: **AT*SETICK=**<text>,<show>

Execution command response: *SETICK:<object_index>

Test command: **AT*SETICK=?** Shows if the command is supported.

Test command response: *SETICK:(list of supported <show>s)

Parameters:

<text>:

<text>	Description
String	Text to be included in the ticker

<object_index>:

<object_index>	Description
Integer type	Index of the ticker

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will not be displayed when created
1	Object will be displayed when created

AT*SEDATE SEMC create date field

Description:

A Date field is an editable component for presenting date and time (calendar) information. The value for this field can be set initially. If the value is not set, then the UI for the field shows this clearly.
An instance of a Date field can be configured to accept date or time information. This mode is set by the <mode> parameter. The DATE input mode allows only date information (year, month, day) to be set and the TIME mode allows only time information (hours, minutes, seconds) to be set.
When the user has modified the contents of the date field and accepts it, the unsolicited result code ***SEGUII** is sent.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SEDATE.
A Date field object is deleted if **AT*SEDEL=<object_index>** is called.
Note: It is not possible to call AT*SEDATE with <mode> == 2 and then include <date> first and <time> after that in the command parameter string. Doing so results in an error. The correct usage of the SET command is shown in the two examples below:
AT*SEDATE="Set the date",1,1,0,"2005/05/29"
AT*SEDATE="Set the time",2,1,0,"12:30:00"

Execution command:

AT*SEDATE=<title>,<mode>,<show>[,<form> [,<date>][,<time>]]

Execution command response:

*SEDATE:<object_index>

Test command:

AT*SEDATE=? Shows if the command is supported.

Test command response:

Range of general parameters:
*SEDATE:(list of supported <mode>s), (list of supported <show>s), (list of supported <form>s)

Parameters:

<title>:

<title>	Description
String	Item title

<mode>:

<mode>	Description
1	DATE mode. Possible to set and display the date (year, month, day)
2	TIME mode. Possible to set and display the time (hours, minutes, seconds)

<date>:

<date>	Description
Integer type	Format is “yy/MM/dd” or “yyyy/MM/dd”, where characters indicate year (two last digits or four digits, depending on the **AT+CSDF** setting), month and day

<time>:

<time>	Description
String type	Format is “hh:mm:ss”, where characters indicate hour, minutes and seconds

<object_index>:

<object_index>	Description
Integer type	Index of the date field

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will not be displayed when created.
1	Object will be displayed when created

<form>:

<form>	Description
Integer type	Tells whether the GUI object will be placed in a form or not
0	The object will not be placed in a form
1	The object will be placed in a form

AT*SEGAUGE SEMC create gauge (bar graph/progress feedback)

Description:
Creates a gauge (progress feedback).
If the gauge is interactive with <interactive>=1 and the user has changed the value of the gauge, the unsolicited result code ***SEGUII** is sent. This also happens if the user cancels the gauge.
If <interactive> = 2 then the user (accessory) has the ability to update the gauge via the **AT*SEGUP** command.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SEGAUGE.
A gauge object is deleted if **AT*SEDEL=<object_index>** is called.

Execution command:
AT*SEGAUGE=<label>,<interactive>,<show>[,<form>[,<initial_value>[,<maxvalue>]]]

Execution command response:
*SEGAUGE:<object_index>

Test command:
AT*SEGAUGE=? Shows if the command is supported.

Test command response:
Range of general parameters:
*SEGAUGE: 0, (list of supported <show>s), (list of supported <form>s)
*SEGAUGE: 1, (list of supported <show>s),(list of supported <form>s),(list of supported <initial_value>s)
*SEGAUGE: 2, (list of supported <show>s),(list of supported <form>s),(list of supported <initial_value>s),(list of supported maxvalues)

Parameters:

<label>:

<label>	**Description**
String	Item label

<interactive>:

<interactive>	**Description**
0	Non-interactive mode. The user cannot change the value of the bar graph. The gauge is used as a “progress feedback”. Initial_value and maxvalue ignored
1	Interactive mode – to update the value of the gauge use the AT command **AT*SEGUP**. Used by, for example, accessories that want to control a progress feedback by themselves. Maxvalue ignored
2	Interactive mode – the user is allowed to modify the value of the gauge by using the keyboard. The accessory will be notified when the gauge has been updated via ***SEGUII**

<maxvalue>:

<maxvalue>	**Description**
Integer	The maximum value of the gauge. In range [1–28]

<initial_value>:

<initial_value>	Description
Integer	The initial value of the gauge. • In range [0; 100] for interactive=1 • In range [0; 28] for interactive=2

<object_index>:

<object_index>	Description
Integer type	Index of the object

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will not be displayed when created
1	Object will be displayed when created

<form>:

<form>	Description
Integer type	Tells whether the GUI object will be placed in a form or not
0	The object will not be placed in a form
1	The object will be placed in a form

AT*SEGUP SEMC update gauge (bar graph/ progress feedback)

Description: Updates an existing bar graph (progress feedback) with a new value. The object ID of the bar graph must be given together with the new bar graph value. That is, an object created by AT*SEGAUGE has to be created with parameter <interactive> = 1.

Execution command: **AT*SEGUP=**<object_index>,<new_value>[,<form>]

Test command: **AT*SEGUP=?** Shows if the command is supported.

Parameters:

<new_value>:

<new_value>	Description
Integer	The new value of the bar graph

<object_index>:

<object_index>	Description
Integer type	Index of the bar graph to be updated

<form>:

<form>	Description
Integer type	The object ID for the form in which the object is placed. 0 means that the object is stand-alone
0	The object is not placed in a form. **Default value**
1–255	The object ID for the form

AT*SEONO SEMC create on/off input

Description: Displays an On/Off input screen with two radio buttons showing “on” and “off”. A value could be accepted by the user or cancelled. The unsolicited result code ***SEGUII** is sent to the accessory when the user has accepted or cancelled the On/Off input.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SEONO.
An On/Off object is deleted if **AT*SEDEL=<object_index>** is called.

Execution command: **AT*SEONO=**<title>,<default_value>,<show>

Execution command response: *SEONO:<object_index>

Test command: **AT*SEONO=?** Shows if the command is supported.

Test command response: Range of general parameters:
*SEONO: (list of supported <default_value>s), (list of supported <show>s)

Parameters:

<title>:

<title>	Description
String	Title of the On/Off input

<default_value>:

<default_value>	Description
0	Off
1	On

<object_index>:

<object_index>	Description
Integer type	Index of the On/Off input

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later

<show>	Description
0	Object will not be displayed when created
1	Object will be displayed when created

AT*SEYNQ SEMC create yes/no question

Description: Creates a Yes/No question GUI object with an image and a question to be answered Yes or No.
When the user presses a relevant key, the unsolicited ***SEGUII** is sent to the accessory.
To be able to use this command, a UI session has to be established. That is, the AT command **AT*SEUIS=1** has to be called before calling AT*SEYNQ.
A Yes/No question object is deleted if **AT*SEDEL=<object_index>** is called.

Execution command: **AT*SEYNQ=**<title>,<question>,<show>[,<image_id>]

Execution command response: *SEYNQ:<object_index>

Test command: **AT*SEYNQ=?** Shows if the command is supported.

Test command response: Range of general parameters:
*SEYNQ: (list of supported <show>s)

Parameters:

<title>:

<title>	Description
String	Title of the Yes/No question

<question>:

<question>	Description
String	The question to be answered by the user

<image_id>:

<image_id>	Description
Integer type	Id of the image (icon) to be used in the question box. Valid range [0, 65535]

<object_index>:

<object_index>	Description
Integer type	Object index of the Yes/No question

<show>:

<show>	Description
Integer type	Tells whether the GUI object will be visible on the screen directly when created. If <show> = 0 then it is possible to use the **AT*SESAF** command to display the object later
0	Object will not be displayed when created
1	Object will be displayed when created

AT*SEDEL SEMC GUI delete

Description: Deletes a GUI object specified with <object_index>.

Execution command: **AT*SEDEL=**<object_index>

Test command: **AT*SEDEL=?** Shows if the command is supported.

Parameter:

<object_index>:

<object_index>	Description
Integer type	Index of the object that will be deleted

AT*SESLE SEMC soft key label (ver. 1)

Description: Defines the labels to be used for the soft keys. Parameter <object_id> verifies to which GUI object the soft key(s) will be added.
If <nbr_of_actions> = 0: The user can add a new soft key label for the right soft key specified in the <short_text>.
If <nbr_of_actions> > 0: Right soft key is named “More”. <short_text> is added as first element in the “More menu followed by the <long_textX> parameters.
The new soft key IDs are sent to the user in the <softkey_idX> parameters.

Set command: **AT*SESLE=**<object_id>,<nbr_of_actions>,<icons_or_texts>, <show>,<short_text>[,<long_text1>[,<long_text2>…]]]

Set command response: *SESLE: <softkey_id1>[,<softkey_id2>[<softkey_id3>[,…]]]

Test command: **AT*SESLE=?** Shows if the command is supported.

Parameters:

<object_id>:

<object_id>	Description
Integer	The ID of the GUI object with the soft keys

<nbr_of_actions>:

<nbr_of_actions>	Description
Integer	Defines the number of <long_textX> soft key(s) to be added to the GUI object in the “More” menu

<short_text>:

<short_text>	Description
String if <icons_or_texts> = 0	Text label for the right soft key or the first element in “More” menu (see description)
Integer if <icons_or_texts> = 1	Icon ID for the icon to be used instead of the text

<long_text>:

<long_text>	Description
String – if <icons_or_texts> = 0	Text label(s) for the text to be used in the “More” menu
Integer – if <icons_or_texts> = 1	Icon ID for the icon to be used instead of the text

<action>:

<action>	Description
Integer value	Reference value for soft key action

<icons_or_texts>:

<icons_or_texts>	Description
Integer	
0	The new soft key labels to be placed in the right or the “More” menu, are text strings. This implies that the parameters <short_text> and <long_textX> contains ordinary text strings
1	The new soft key labels to be placed in the right or the “More” menu are icons. This implies that the parameters <short_text> and <long_text> contains icon IDs to the icons to be displayed

<show>:

<show>	Description
0	Do not display the soft key
1	Display the soft key directly. **Default value**

<softkey_id>:

<softkey_id>	Description
Integer	Reference value for AT soft key ID

AT*SERSK SEMC remove soft key

Description: Removes a soft key action defined with command **AT*SESLE**.
Note: This command has to be entered after AT*SESLE. When the GUI object is removed the soft keys are also removed.

Execution command: **AT*SERSK=**<object_id>,<softkey_id>

Test command: **AT*SERSK=?** Shows if the command is supported.

Parameter:

<object_id>:

<object_id>	**Description**
Integer value	ID number for the object

<softkey_id>:

<softkey_id>	**Description**
Integer value	Reference value for soft key

AT*SEUIS SEMC UI session establish/terminate

Description: This command is used by an accessory to establish or to terminate a UI session.
If the UI session was established/terminated, OK will be returned.
When an accessory wants to show something on the display, it must request a UI session. Also, if an accessory wants to remove all of its objects then it just ends a UI session. Within a UI session an accessory has freedom to create new, modify and/or remove objects.
When the UI session has gained or lost focus the unsolicited result code ***SESFI** is returned.

Execution command: **AT*SEUIS=**<action>

Read command: **AT*SEUIS?**

Read command response: *SEUIS:<action>

Test command: **AT*SEUIS=?** Shows if the command is supported.

Parameter:

<action>:

<action>	**Description**
0	Terminate session
1	Establish session

Unsolicited result codes

*SEGUII SEMC GUI indication

Description:

This unsolicited result code is sent when an action on a GUI object has occurred. The action is related to the type of the GUI object. The object specific information that can be included in <object_specific_info> is dependent on the action as well as the type of GUI object.
Note: The GUI object can create its own actions by using the command **AT*SESLE** (creating softkey actions). These new actions are given an action ID according to the specification **AT*SESLE**.
This result code is activated by **AT*SELERT**, **AT*SESTRI**, **AT*SELIST**, **AT*SETICK**, **AT*SEDATE**, **AT*SEGAUGE**, **AT*SEONO**, **AT*SEYNQ** or **AT*SESLE**

Unsolicited result code:

*SEGUII: <object_index>,<action>[,<object_specific_info1>[,<object_specific_info2>..]] When a GUI action has occurred.

Parameters:

<object_index>:

<object_index>	**Description**
Integer type	Index of the object for which the action has occurred. **Note:** A stand-alone object and a form can have the same object_index

<action>:

<action>	**Description**	**GUI object**	**<object_specific_info>**
0	CANCEL action	All	-
1	PREVIOUS action	All	-
2	NO action	Yes/no question	-
3	YES action	Yes/no question	-
4	ACCEPT action, the user has accepted a form	Form	-
5	ACCEPT INDEX action, the user has selected an item in a list	List (exclusive or implicit)	Integer. Index of the selected list item
6	DELETE INDEX action, the user has selected an item to be deleted in a list	List	Integer. Index of the selected list item

<action>	Description	GUI object	<object_specific_info>
7	ACCEPT N_OF_MANY action. The user has selected one or many item(s) in a list	List (multiple)	Integer. Index(es) of the selected list item(s): <object_specific_info1> [,<object_specific_info2>…]
8	ACCEPT DATE action. The user has accepted a date value in a GUI object	Date input	String. Date format is depending on the **AT+CSDF** setting
9	ACCEPT TIME action. The user has accepted a time value in a GUI object	Time input	String. Time format is depending on the **AT+CSDF** setting
10	ACCEPT BOOLEAN action, the user has accepted a Boolean value in a GUI object	On/off question	Boolean. True – On, False – Off
11	ACCEPT STRING action, the user has accepted a string in a GUI object	String input dialogue	String. The text string in the GUI object
12	ACCEPT INTEGER action, the user has entered a new value in the progress feedback	Gauge (progress feedback)	Integer. The new value
13	SOFT KEY ACTION	Softkey	Integer. The action ID of the softkey that has been pressed
108	FORM ACCEPT DATE	Date input placed in a form	String. Date format is depending on the **AT+CSDF** setting
109	FORM ACCEPT TIME	Time input placed in a form	String. Time format is depending on the **AT+CSDF** setting
111	FORM ACCEPT STRING	String input dialogue placed in a form	String. The text string in the GUI object
112	FORM ACCEPT INTEGER	Gauge (progress feedback) placed in a form	Integer. The new value

*SESFI SEMC session focus indication

Description:

This unsolicited result code is sent when the session has got focus or when focus for the session has been lost.
This result code is activated by **AT*SEUIS**.

Unsolicited result code:

*SESFI:<focus>
When focus for the session is obtained or lost.

Parameter:

<focus>:

<focus>	Description
0	Focus is lost
1	Focus is obtained

*SEAAI SEMC menu item indication

Description:

This indication is sent to the accessory when the menu item with ID <menu_id> is activated by the user.
This result code is activated by **AT*SEAM**.

Unsolicited result code:

*SEAAI:<menu_id>
When menu item is activated.

Parameter:

<menu_id>:

<menu_id>	Description
Integer type	The menu ID for the item activated

Ensemble C38: Bluetooth commands

Commands

AT*EIBA Ericsson Internal Bluetooth address

Description: Command that is generated internally in the platform. It forwards the Bluetooth address of a connected Bluetooth device.

Execution command: **AT*EIBA=**<bt_address>

Test command: **AT*EIBA=?** Shows if the command is supported.

Parameter:

<bt_address>:

<bt_address>	**Description**
String	The Bluetooth address given in hexadecimal format

AT+BINP Bluetooth input

Description: Requests some specific data input from the phone. On reception of this command the phone performs the proper actions such that the requested information is sent back to the HF using the **+BINP** response.
The type of data the HF will expect in the <dataresp> parameter returned by the phone depends on the information requested in each case.

Execution command: **AT+BINP=**<datarequest>

Execution command response: **AT+BINP:**<dataresp>1...<dataresp>n

Test command: **AT+BINP=?** Shows if the command is supported.

Test command response: +BINP: (list of supported <datarequest>s)

Parameters:

<datarequest>:

<datarequest>	**Description**
1	Request phone number corresponding to the last voice tag recorded in the HF

<dataresp>:

<dataresp>	**Description**
<dataresp>1..<dataresp>n	Data parameters returned by the phone. Their contents depends on the value of the <datarequest> parameter

Supported values on <dataresp> depending on <datarequest>:

<datarequest>	Description
1	<Phone number>. Phone number string (max. 32 digits). The format (type of address) of the phone number string should conform with the rules stated in *Digital cellular telecommunications system (Phase 2+); Abbreviations and acronyms*, subclause 10.5.4.7, for a value (in integer format) of the type of address octet of 145, if dialling string includes international access code character "+" and for a value of 129 otherwise

AT+BLDN Bluetooth last dialled number

Description: Calls the last phone number dialled. On reception of this command, the phone sets up a voice call to the last phone number dialled.

Execution command: **AT+BLDN**

Test command: **AT+BLDN=?** Shows if the command is supported.

AT+BVRA Bluetooth voice recognition activation

Description: Enables/disables the voice recognition function in the phone. This command activates the result code **+BVRA**
Note: This command gives an error if the request is not made from a bluetooth handsfree device. That is, it is not possible to connect to a bluetooth handsfree device and then in parallell connect to the ME with a terminal program (hyperterminal and substitutes) and try sending the AT command, this also results in an error.
Note: A voice message has to be recorded to make it possible to use this command. If no such message exist, then the command gives an error in the response.

Execution command: **AT+BVRA=**<vrec>

Test command: **AT+BVRA=?** Shows if the command is supported.

Test command response: +BVRA: (list of supported <vrec>s)

Parameter:

<vrec>:

<vrec>	Description
0	Disable Voice recognition in the phone
1	Enable Voice recognition in the phone

AT+NREC Noise reduction and echo cancelling

Description: Enables/disables any Echo Cancelling and Noise Reduction functions embedded in the phone.
Note: This command only works over Bluetooth wireless technology. This is because all AT commands for audio preferences are deleted and replaced with an audio class and id. However, the command is standard for the BTHF profile and is therefore used in this context.

Execution command: **AT+NREC=**<nrec>

Test command: **AT+NREC=?** Shows if the command is supported.

Test command response: +NREC: (list of supported <nrec>s)

Parameter:

<nrec>:

<nrec>	**Description**
0	Disables EC/NR in the phone
1	Enables EC/NR in the phone

AT+VGM Gain of microphone

Description: Command issued by the HF to report its current microphone gain level setting to the phone. <gain> is a decimal numeric constant, relating to a particular (implementation dependent) volume level controlled by the HF. This command does not change the microphone gain of the phone, it simply indicates the current value of the microphone gain in the HF.
This command activates the result code **+VGM**

Execution command: **AT+VGM=**<gain>

Test command: **AT+VGM=?** Shows if the command is supported.

Test command response: +VGM: (list of supported <gain>s)

Parameter:

<gain>:

<gain>	**Description**
0–15	0 – Minimum gain 15 – Maximum gain

AT+VGS Gain of speaker

Description: Command issued by the HF to report its current speaker gain level setting to the phone. <gain> is a decimal numeric constant, relating to a particular (implementation dependent) volume level controlled by the HF. This command does not change the speaker gain of the phone, it simply indicates the current value of the speaker gain in the HF.
Note: This command returns an error if the request is not made from a bluetooth handsfree device.
This command activates the result code **+VGS**

Execution command: **AT+VGS=**<gain>

Test command: **AT+VGS=?** Shows if the command is supported.

Test command response: +VGS: (list of supported <gain>s)

Parameter:

<gain>:

<gain>	Description
0–15	0 – Minimum gain 15 – Maximum gain

AT+BRSF Bluetooth retrieve supported

Description: Notifies the ME of the supported features available in the HF and requests information about the supported features in the ME. The supported features are represented by a decimal value.
<HF supported bitmap> is a 32 bit unsigned integer representing a bitmap of the supported features in the HF according to table 1. The unused bits are initialised to zero.
<ME supported bitmap> is a 32 bit unsigned integer representing a bitmap of the supported features in the ME according to table 2. The unused bits are initialised to zero.

Execution command: **AT+BRSF=**<HF supported features bitmap>

Execution command response: +BRSF: <ME supported features bitmap>

Test command: **AT+BRSF=?** Shows if the command is supported

Parameter:

<HF supported features bitmap>

bit	Description
0	EC and/or NR function
1	Call waiting and 3-way calling
2	CLI presentation capability

bit	Description
3	Voice recognition activation (enables unsolicited +BVRA)
4	Remote volume control. (enables unsolicited +VGS)
5–31	Unused

<ME supported features bitmap>

bit	Description
0	Three-way calling (AT+CHLD supported)
1	AC and/or NR function (AT+NREC supported)
2	Voice recognition function (AT+BVRA supported)
3	In-band ring tone capability (not supported)
4	Attach a number to a voice tag (AT+BINP supported)
5	Ability to reject call (AT+CHUP supported)
6–31	Unused

AT+GCLIP Graphical caller ID presentation

Description: Activates an unsolicited result code +GCLIP. +GCLIP is used to transmit a graphical representation of the CLIP data when there is an incoming call. For detail information, please see Unsolicited result code in **+GCLIP**.
Note: The functionality of this command has been replaced by AT*SETBC.
Note: The unsolicited result code(s) will only return one image. The bitmap that is returned by the +GCLIP command will only contain the information/ characters that fit into the image (98x16).

Execution command: **AT+GCLIP**=<display_type>

Test command: **AT+GCLIP=?** Shows if the command is supported

Test command response: +GCLIP:[<list of supported display types>]

Parameter:

<display_type>

<display_type>	Description
integer	1: 96x16 BW0 2…255 Reserved

Unsolicited result codes

+BVRA Bluetooth voice recognition activation indication

Description: Unsolicited result code used to notify the HF when the voice recognition function in the phone has been terminated autonomously. This result code is activated by **AT+BVRA**.

Unsolicited result code: **+BVRA:** <vrect>

Parameter:

<vrect>:

<vrect>	Description
0	Voice recognition is disabled in the phone
1	Voice recognition is enabled in the phone

+VGM Gain of microphone indication

Description: Unsolicited result code issued by the phone to set the microphone gain of the HF. <gain> is a decimal numeric constant, relating to a particular (implementation dependent) volume level controlled by the HF.
This result code is activated by **AT+VGM**.

Unsolicited result code: **+VGM:** <gain>

Note: Due to the small inconsistency between the GSM 07.07 standard and the current headset specification (*Specification of the Bluetooth System; Profiles, v1.1, Part K:6, Headset Profile.*), the HF will also accept the "=" symbol in place of ":" as a valid separator for this unsolicited result code.

Parameter:

<gain>:

<gain>	Description
0–15	0 – Minimum gain 15 – Maximum gain

+VGS Gain of speaker indication

Description: Unsolicited result code issued by the phone to set the speaker gain of the HF. Parameter <gain> is a decimal numeric constant, relating to a particular (implementation dependent) volume level controlled by the HF.
This result code is activated by **AT+VGS**.

Unsolicited result code: **+VGS:** <gain>

Note: Due to the small inconsistency between the GSM 07.07 standard and the current Headset specification (*Specification of the Bluetooth System; Profiles, v1.1, Part K:6, Headset Profile.*), the HF will also accept the "=" symbol in place of ":" as a valid separator for this unsolicited result code.

Parameter:

<gain>:

<gain>	**Description**
0–15	0 – Minimum gain 15 – Maximum gain

+BSIR Bluetooth setting of in-band ring tone indication

Description: Unsolicited result code issued by the phone to indicate to the HF that the in-band ring tone setting has been locally changed. The HF may react accordingly by changing its own alert method.

Unsolicited result code: **+BSIR:** <bsir>

Parameter:

<bsir>	**Description**
0	The phone provides no in-band ring tone
1	The phone provides an in-band ring tone

+BINP Bluetooth input indication

Description: Unsolicited result code issued by the phone in response to a request from the terminal equipment to provide information of a specified type.

Unsolicited result code: **+BINP:** <datarespn1>[,...,<datarespn>]

Parameter:

<datarespn> type is dependent on the <datarequest> parameter. See **AT+BINP**

+GCLIP Graphical caller ID presentation

Description:

Unsolicited result code activated by **AT+GCLIP**.
Encoding of bitmaps into GCLIP Data
The GCLIP data is considered a stream of data segmented into a series of maximum 255 chunks.
The chunks are encoded using hexadecimal format. Hence every byte is encoded using two ASCII digits/characters.
The length of the chunks is not defined, it is only required that they are sent in the right order. For every chunk the index is incremented. The chunks are reassembled at the receiving side, in order to retrieve the total bitmap.
The bitmap is divided into 8-bit wide horisontal bands. The data starts with the uppermost band. Every band is followed by the band below. It is not required to start a new chunk in order to send the next band.
Within a band the leftmost element will be sent first until finally the right-most element of the band has been reached.
Each element of the band consists of one single byte. The byte represents the 8 pixels, vertically stacked upon each other. The least significant bit (b0) indicates the uppermost of the 8 encoded bits, followed by b1, the one below the uppermost pixel.

Unsolicited result code:

+GCLIP: <gclip_index>,<gclip_data>

Parameter:

<gclip_index>: Integer

<gclip_index>	**Description**
0…255 (0 = first)	Sequence number of gclip_data element

<gclip_data>:

<gclip_data>	**Description**
String of hexadecimal encoded data.	Max 96 hex encoded bytes per result

Ensemble S1: GSM DTE-DCE interface

Commands

AT+CSCS Select TE character set (ver. 3)

Description: Informs TA which character set <chset> is used by the TE. TA is then able to convert character strings correctly between TE and ME character sets. When TA – TE interface is set to 8-bit operation and the used TE alphabet is 7-bit, the highest bit is set to zero.
Note: The manufacturer specifies how the internal alphabet of ME is converted to and from the TE alphabet.

Set command: **AT+CSCS=**<chset>

Read command: **AT+CSCS?** Displays the current <chset> setting.

Read command response: +CSCS: <chset>

Test command: **AT+CSCS=?** Shows if the command is supported.

Test command response: +CSCS: (list of supported <chset>s)

Parameter:

<chset>:

<chset>	**Description**
"GSM"	GSM default alphabet (GSM 03.38 subclause 6.2.1). This setting often causes software flow control (XON/XOFF) problems. **Default value**
"IRA"	International reference alphabet (ITU-T T.50) **Note**: Recommended Default value by GSM 07.07
"8859-n"	ISO 8859 Latin *n* (1-6) character set. Only number 1
"UTF-8"	Universal Text Format, 8 bits

Ensemble S2: Call control

Commands

AT+CHUP Hang up call

Description: Requests hang up.

Execution command: **AT+CHUP**

Test command: **AT+CHUP=?** Shows if the command is supported.

AT+CRC Cellular result codes (ver. 2)

Description: Controls whether or not the extended format of incoming call indication or GPRS network request for PDP context activation or notification for VBS/VGCS calls is used. When enabled, an incoming call is indicated to the TE with unsolicited result code **+CRING**: <type> instead of the normal **RING**.

Set command: **AT+CRC=**[<mode>]

Read command: **AT+CRC?** Displays the current setting.

Read command response: +CRC: <mode>

Test command: **AT+CRC=?** Shows if the command is supported.

Test command response: +CRC: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	Disables extended format. **Default value**
1	Enables extended format

AT+CR Service reporting control

Description: Enables or disables display of intermediate **+CR:<serv>** result code to be returned during the call setup phase. The code is returned before the intermediate result code CONNECT is returned.

Set command: **AT+CR=**<mode>

Read command: **AT+CR?** Displays the current <mode> setting.

Test command: **AT+CR=?** Shows if the command is supported.

Test command response: +CR: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	Disable reporting. **Default value**
1	Enable reporting

AT+CV120 V.120 rate adaption protocol

Description: Sets the values of the V.120 protocol parameters (defined in CCITT V.120) that are carried in the GSM BC and/or LLC information elements.
Test command returns values supported by the TA as a compound value.

Set command: **AT+CV120=**[<rah>[,<mfm>[,<mode>[,<llineg>[,<assign>[,<negtype>]]]]]]

Read command: **AT+CV120?** Displays the current <n> and <m> settings.

Read command response: +CV120: <rah>,<mfm>,<mode>,<llineg>,<assign>,<negtype>

Test command: **AT+CV120=?** Shows if the command is supported.

Test command response: +CV120: (list of supported <rah>s),(list of supported <mfm>s),(list of supported <mode>s),(list of supported <llineg>s),(list of supported <assign>s),(list of supported <negtype>s)

Parameters:

<rah>:

<rah>	Description
0	Rate adaption header not included
1	Rate adaption header included (mandatory for protocol sensitive modes)

<mfm>:

<mfm>	Description
0	Multiple frame establishment not supported, only UI frames allowed
1	Multiple frame establishment supported, both I and UI frames allowed

<mode>:

<mode>	Description
0	Bit transparent mode of operation
1	Protocol sensitive mode of operation

<llineg>:

<llineg>	Description
0	No negotiation, LLI = 256 only
1	Negotiation allowed. **Note**: <negtype> indicates the connection over which the negotiation is performed

<assign>:

<assign>	Description
0	Message originator is “default assignee”
1	Message originator is “assignor only”

<negtype>:

<negtype>	Description
0	Negotiation is done using logical link zero
1	Negotiation is done with USER INFORMATION messages on a temporary signalling connection

AT+VTS DTMF and tone generation

Description: Allows the transmission of DTMF tones. These tones may be used, for example, when announcing the start of a recording period. The command is write-only. The command is used only during voice calls.
Note: The ATD command is used only for dialling. It is not possible to generate arbitrary DTMF tones using the ATD command.

Set command: **AT+VTS=**<DTMF>

Test command: **AT+VTS=?** Shows if the command is supported.

Parameter:

<DTMF>: An ASCII character string with entries in the set “0-9, #, *, A-D”, separated by commas. Each entry is interpreted as a single DTMF tone.
Example: The string “8,9” sends two DTMF tones, “8” followed by “9”.

Unsolicited result codes

+CME Mobile equipment error result

Description: Produced to indicate completion of a command. Produced when the command is not recognised, the command line maximum length is exceeded, the parameter value is invalid or there are other problems with processing the command line.

Unsolicited result code: **+CME:** <err>

Parameter:

<err>: Numeric or verbose format. Decided by **AT+CMEE**.

+CR Service reporting control

Description: Transmitted during connect negotiation at the point where the phone has determined what speed and quality-of-service will be used, before any error control or data compression reports are transmitted and before any final result code is transmitted.

Unsolicited result code: **+CR:** <serv>

Parameter:

<serv>:

<type>	Description
ASYNC	Asynchronous transparent
SYNC	Synchronous transparent
REL ASYNC	Asynchronous non-transparent
REL SYNC	Synchronous non-transparent

+CRING Call mode indication

Description: When enabled by using **AT+CRC**, an incoming call is indicated with +CRING instead of **+RING**.

Unsolicited result code: **+CRING:** <type>

Parameter:

<type>:

<type>	Description
ASYNC	Asynchronous transparent
SYNC	Synchronous transparent
REL ASYNC	Asynchronous non-transparent
FAX	Facsimile
VOICE	Normal voice
VOICE/XXX	Voice followed by data (“XXX” is SYNC, ASYNC, REL ASYNC or REL SYNC)
ALT VOICE/XXX	Alternating voice/data, voice first
ALT XXX/VOICE	Alternating voice/data, data first
ALT VOICE/FAX	Alternating voice/fax, voice first
ALT FAX/VOICE	Alternating voice/fax, fax first

Ensemble S3: GSM data/fax

Commands

AT+CBST Select bearer service type (ver. 3)

Description: Selects the bearer service <name> with data rate <speed> and the connection element <ce> to be used when data calls are originated. Values may also be used during mobile terminated data call setup, especially in the case of single numbering scheme calls.

Set command: **AT+CBST=**[<speed>,[<name>,[<ce>]]]

Read command: **AT+CBST?** Displays the current setting.

Read command response: +CBST: <speed>,<name>,<ce>

Test command: **AT+CBST=?** Shows if the command is supported.

Test command response: +CBST: (list of supported <speed>s, list of supported <name>s, list of supported <ce>s)

Parameter:

<speed>:

<speed>	Description
0	Auto selection of baud setting. **Default value**
7	9600 bps V.32
12	9600 bps V.34
14	14400 bps V.34
15	19200 bps V.34
16	28800 bps V.34
39	9600 bps V.120
43	14400 bps V.120
47	19200 bps V.120
48	28800 bps V.120
71	9600 bps V.110 (ISDN)
75	14400 bps V.110 (ISDN)
79	19200 bps V.110 (ISDN)
80	28800 bps V.110 (ISDN)

<name>:

<name>	Description
0	Asynchronous connection (UDI or 3.1kHz modem). **Default value**

<name>	Description
4	Data circuit asynchronous (RDI)

<ce>:

<ce>	Description
1	Non transparent. **Default value**

AT+CRLP Radio link protocol (ver. 2)

Description: Radio Link Protocol (RLP) parameters used when non-transparent data calls are originated, may be altered with the set command. Available command subparameters depend on the RLP versions implemented by the device. For example, <ver> may not be available if the device supports only versions 0 and 1.
Note: If radio link protocol is not used, but some other error correcting protocol (for transparent data calls) is used, V.25ter Error Control Selection test command +ES=? may be used to indicate the presence of the protocol.
The test command returns values supported by the TA as a compound value. If the ME/TA supports several RLP versions <verx>, the RLP parameter value ranges for each <verx> are returned in a separate line.

Set command: **AT+CRLP=**[<iws>[,<mws>[,<T1>[,<N2>[,<ver>[,<T4>]]]]]]

Read command: **AT+CRLP?** Displays the current parameter settings for each supported RLP version. Only RLP parameters applicable to the corresponding <verx> are returned.

Read command response:
+CRLP: <iws>,<mws>,<T1>,<N2>[,<ver1>[,<T4>]]**<CR><LF>**
[+CRLP: <iws>,<mws>,<T1>,<N2>[,<ver2>[,<T4>]]**<CR><LF>**
[...]]

Test command: **AT+CRLP=?** Shows if the command is supported.

Test command response:
+CRLP: (list of supported <iws>s),(list of supported <mws>s),(list of supported <T1>s),(list of supported <N2>s)[,<ver1>[,(list of supported <T4>s)]]**<CR><LF>**

[+CRLP: (list of supported <iws>s),(list of supported <mws>s),(list of supported <T1>s),(list of supported <N2>s)[,<ver2>[,(list of supported <T4>s)]]**<CR><LF>**

[...]]

Parameters: Default values and value ranges depend on RLP version. See GSM 04.22 subclause 5.4

<iws>:

<iws>	Description
0–61	IWF to phone window size
61	**Default value**

<mws>:

<mws>	Description
0–61	MS to IWF window size
61	**Default value**

<T1>:

<T1>	Description
38–100	Acknowledgement timer T1 setting, in 10 ms steps
48	T1=480 ms. **Default value**

<N2>:

<N2>	Description
0–255	Number of retransmission attempts, N2.
6	**Default value**

<ver>:

<ver>	Description
Integer	RLP version – When version indication is not present, <ver>=0 is assumed

<T4>:

<T4>	Description
3–255	Resequencing period T4, in 10ms steps
5	**Default value**

Ensemble S4: Extended error reporting

Commands

AT+CEER Extended error report (ver. 2)

Description: Causes the TA to return one or more lines of information text <report>, determined by the ME manufacturer, which offer the user of the TA an extended report of the reason for:

- the failure in the last unsuccessful call setup (originating or answering) or in-call modification
- the reason for last call release
- the last unsuccessful attempt to attach GPRS or unsuccessful activation of PDP context
- the last detachment of GPRS or deactivation of PDP context

Typically, the text consists of a single line containing the failure information given by GSM/UMTS network in textual format.

Execution command: **AT+CEER**

Execution command response: +CEER: <report>

Test command: **AT+CEER=?** Shows if the command is supported.

Parameter:

<report>:

<report>	**Description**
Characters	The total number of characters, including line terminators, in the information text must not exceed 2041 characters. Text must not contain the sequence 0**<CR>** or OK**<CR>**

Ensemble S5: GSM HSCSD

Commands

AT+CHSD HSCSD device parameters (ver. 2)

Description: The execution command returns information about HSCSD features (refer to GSM 02.34) supported by the ME.
The test command does not return any values, only OK to show that the command is supported.

Execution command: **AT+CHSD**

Execution command response: +CHSD: <mclass>,<maxRx>,<maxTx>,<sum>,<codings>

Test command: **AT+CHSD=?** Shows if the command is supported.

Parameters:

<mclass>:

<mclass>	Description
8	Multislot class 8

<maxRx>:

<maxRx>	Description
4	Maximum number of receive timeslots that is supported by the MS

<maxTx>:

<maxTx>	Description
1	Maximum number of transmit timeslots that is supported by the MS

<sum>:

<sum>	Description
5	Total number of receive and transmit timeslots that ME can support at the same time is 5 (that is, 4+1). The following applies in a HSCSD call: 2 <= (receive slots) + (transmit slots) <= <sum>

<codings>: This is a sum of integers each representing a supported channel coding. For example, value 12 indicates that 9.6 Kbps and 14.4 Kbps are supported

<codings>	Description
4	Indicates that the accepted channel coding for the next established non-transparent HSCSD call is 9.6 Kbps
8	Indicates that the accepted channel coding for the next established non-transparent HSCSD call is 14.4 Kbps
12	Indicates that the accepted channel codings for the next established non-transparent HSCSD call are both 9.6 Kbps and 14.4 Kbps. **Default value**

AT+CHSN HSCSD non-transparent call configuration (ver. 2)

Description: Controls parameters for non-transparent HSCSD calls. Changing <topRx> or <codings> during a call does not affect the current call. Changing <wAiur> or <wRx> affects the current call only if <topRx> was non-zero when the call was established. (When using the command in this way it comes in the "action" command category). This is what is referred to as User initiated modification in GSM 22.034 and User initiated upgrading and downgrading in GSM 23.034.
Note: Recommended value for parameter <speed> in **AT+CBST** is 0.

Set command: **AT+CHSN=**[<wAiur>[,<wRx>[,<topRx>[,<codings>]]]]

Read command: **AT+CHSN?** Displays the current setting.

Read command response:

+CHSN: <wAiur>, <wRx>, <topRx>, <codings>

Test command: **AT+CHSN=?** Shows if the command is supported.

Test command response:

+CHSN: (list of supported <wAiur>s), (list of supported <wRx>s), (list of supported <topRx>s), (list of supported <codings>s)

Parameters:

<wAiur>:

<wAiur>	Description
0	TA/ME calculates a proper number of receive time slots from currently selected fixed network user rate (<speed> parameter from **AT+CBST** command, ref and <codings> and <wRx> (or <maxRx> from **AT+CHSD** command if <wRx>=0). See note below. **Default value**
1	Wanted air interface user rate is 9.6 Kbps
2	Wanted air interface user rate is 14.4 Kbps
3	Wanted air interface user rate is 19.2 Kbps
4	Wanted air interface user rate is 28.8 Kbps
6	Wanted air interface user rate is 43.2 Kbps, UMTS only
7	Wanted air interface user rate is 57.6 Kbps, UMTS only

<wRx>:

<wRx>	Description
0	TA/ME calculates a proper number of receive time slots from currently selected <wAiur> and <codings>. See note below
1	Wanted number of receive time slots is 1. **Default value**
2	Wanted number of receive time slots is 2

Note: The Description text above is copied from GSM 27.007 and should be interpreted as follows:
If the <wAiur> and <wRx> are both set to “0”, the number of receive time slots is calculated from <speed> and <codings>. Furthermore, if <speed> is “0” (autobauding), then the number of receive time slots is mapped from <maxRx> from **AT+CHSD** command.

<topRx>:

<topRx>	Description
0	Indicates that the user is not going to change <wAiur> and/or <wRx> during the next call. **Default value**
1	Top value for <wRx> that user is going to request during the next established non-transparent HSCSD call is 1
2	Top value for <wRx> that user is going to request during the next established non-transparent HSCSD call is 2

<codings>: This is a sum of integers each representing a supported channel coding. For example, value 12 (4+8) indicates that 9.6 and 14.4 kbits/s are supported.

<codings>	Description
0	Indicates that all codings are accepted.
4	Indicates that the accepted channel coding for the next established non-transparent HSCSD call is 9.6 Kbps only
8	Indicates that the accepted channel coding for the next established non-transparent HSCSD call is 14.4 Kbps only
12	Indicates that the accepted channel codings for the next established non-transparent HSCSD call are both 9.6 Kbps and 14.4 Kbps. **Default value**

AT+CHSC HSCSD current call parameters (ver. 2)

Description: This execution command returns information about the current HSCSD call parameters:

- The current number of receive and transmit time slots
- Air interface user rate
- Channel coding

Execution command: **AT+CHSC**

Execution command response: +CHSC:<rx>,<tx>,<aiur>,<coding>

Test command: **AT+CHSC=?** Shows if the command is supported.

Parameters:

<rx>:

<rx>	Description
0	No HSCSD call is active. See also note below
1	The number of receive time slots currently in use is 1
2	The number of receive time slots currently in use is 2
3	The number of receive time slots currently in use is 3
4	The number of receive time slots currently in use is 4

<tx>:

<tx>	Description
0	No HSCSD call is active. See also note below
1	The number of transmit time slots currently in use is 1

<aiur>:

<aiur>	Description
0	No HSCSD call is active. See also note below
1	Current air interface user rate is 9.6 Kbps
2	Current air interface user rate is 14.4 Kbps
3	Current air interface user rate is 19.2 Kbps
4	Current air interface user rate is 28.8 Kbps
5	Current air interface user rate is 38.4 Kbps
6	Current air interface user rate is 43.2 Kbps
7	Current air interface user rate is 57.6 Kbps

<coding>:

<coding>	Description
0	No HSCSD call is active. See also note below

<coding>	Description
4	Current channel coding is 9.6 Kbps. (TCH/F9.6)
8	Current channel coding is 14.4 Kbps. (TCH/F14.4)

Note: The value “0” only applies when no HSCSD call is active (general BS 20 or 30) and in such a case all four parameters will be “0”

AT+CHSR HSCSD parameters report (ver. 2)

Description: When this command is enabled the intermediate result code **+CHSR**: <rx>,<tx>,<aiur>, <coding> is returned from the TA to the TE when an HSCSD call is being set up. The result code represents the current (negotiated or renegotiated) HSCSD parameters. If enabled, the intermediate result code is transmitted at the point of the call setup negotiation where the ME/TA has determined what type of an HSCSD connection will be used. Result code transmission is done after possible service (+CR), error control (+ER) and/or compression (+DR) reporting but before possible TE – TA rate (+ILRR) reporting and before the intermediate result code CONNECT is transmitted. The format of the intermediate result code is:
+CHSR: <rx>,<tx>,<aiur>,<coding>
For the value definitions, see the **AT+CHSC** command. For instance, for a non-transparent HSCSD call, result code “+CHSR: 2, 1, 4, 8” means that the call has two time slots downlink, one time slot uplink, the air interface user rate is 28.8 Kbps and the used channel coding is TCH/F14.4.

Execution command: **AT+CHSR=**[<mode>]

Read command: **AT+CHSR?**

Read command response: +CHSR: <mode>

Test command: **AT+CHSR=?** Shows if the command is supported.

Test command response: +CHSR: (list of supported <modes>s)

Parameter:

<mode>:

<mode>	Description
0	Disable reporting. **Default value**
1	Enable reporting

AT+CHSU HSCSD automatic user-initiated upgrade

Description: Controls whether or not automatic user initiated service level upgrading will be used for non-transparent HSCSD calls. “Automatic” means that, if enabled, the ME/TA will use the UP bit in the received RLP frames to determine when to initiate user initiated service level upgrading (that is, when to modify the +CHSN parameters <wAiur> and/or <wRx> for the current call). Refer to GSM 07.01 for details on the interpretation of the UP bit(s).
Note: The validity of the UP bit in the RLP frames depends on the result of the RLP negotiations. The UP bit will only be used if the result of the RLP negotiations were successful with respect to the UP bit.

Set command: **AT+CHSU=**[<mode>]

Read command: **AT+CHSU?** Displays the current <mode> setting.

Test command: **AT+CHSU=?** Shows if the command is supported.

Test command response: +CHSU: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	Disable use of UP bit for upgrading
1	Enable use of UP bit for upgrading. **Default value**

Intermediate result codes

+CHSR HSCSD parameters report result code

Description: When enabled by using the **AT+CHSR** command, this intermediate result code is transmitted at the point of call setup negotiation where the phone has determined what type of HSCSD connection will be used.

Intermediate result code: **AT+CHSR:** <rx>, <tx>,<auir>,<coding>

Parameters: See **AT+CHSC**.

Ensemble S6: GSM network services

Commands

AT+CNUM Subscriber number (ver. 2)

Description: Returns the MSISDN related to the subscriber (this information can be stored in the SIM or in the ME). If a subscriber has different MSISDN for different services, each MSISDN is returned in a separate line.
Note: The implementation of this command is according to Bluetooth HFP 1.5, which deviates somewhat from the 3GPP. The parameters <alpha> and <itc> are not supported but included to show the full command as specified in 3GPP TS 27.005.

Action command: **AT+CNUM**

Action command response:
+CNUM: [<alpha1>],<number1>,<type1>[,<speed>,<service>[,<itc>]][**<CR><LF>**
+CNUM: [<alpha2>],<number2>,<type2>[,<speed>,<service> [,<itc>]] […]]

Test command: **AT+CNUM=?** Shows if the command is supported.

Parameters:

<alphax>: **Not supported**

<alphax>	**Description**
String type	Associated with <numberx>. **Not supported**

<numberx>:

<numberx>	**Description**
String type	Phone number of format specified by <typex>

<typex>:

<typex>	**Description**
Integer format	Type of address, (refer to refer 3GPP 24.008)

<speed>: **Not supported**

<speed>	**Description**
Integer	Data rate, as defined in subclause 6.7 3GPP 27.007 (+CBST command). **Not supported**

<service>: service related to the phone number.

<service>	**Description**
0	Asynchronous modem
1	Synchronous modem. **Not supported**

<service>	Description
2	PAD Access (asynchronous). **Not supported**
3	Packet Access (synchronous). **Not supported**
4	Voice
5	Fax
6–127	All other values below 128 are reserved by GSM 07.07. **Not supported**

<itc>: **Not supported**

<itc>	Description
0	3.1 kHz. **Not supported**
1	UDI. **Not supported**

AT+CREG Network registration (ver. 2)

Description: Controls the presentation of an unsolicited result code **+CREG**: <stat> when <n>=1 and there is a change in the ME network registration status or code +CREG: <stat>[,<lac>,<ci>] when <n>=2 and there is a change of the network cell.
Read command returns the status of result code presentation and an integer <stat>, which shows whether the network has currently indicated the registration of the ME. Location information elements <lac> and <ci> are returned only when <n>=2 and ME is registered in the network.

Set command: **AT+CREG=**[<n>]

Read command: **AT+CREG?**

Read command response: +CREG: <n>,<stat>[,<lac>,<ci>]

Test command: **AT+CREG=?** Shows if the command is supported.

Test command response: +CREG: (list of supported <n>s)

Parameters:

<n>:

<n>	Description
0	Disable network registration unsolicited result code. **Default value**
1	Enable network registration unsolicited result code, +CREG: <stat>
2	Enable network registration and location information unsolicited result code, +CREG: <stat>[,<lac>,<ci>]

<stat>:

<stat>	Description
0	Not registered, the phone is not currently searching a new operator to register to

<stat>	Description
1	Registered, home network
2	Not registered, but ME is currently searching a new operator to register to
3	Registration denied
4	Unknown
5	Registered, roaming

<lac>:

<lac>	Description
String type	Two byte location area code in hexadecimal format. For example, “00C3” equals 195 in decimal

<ci>:

<ci>	Description
String type	Four byte cell ID in hexadecimal format. Four bytes are required for UMTS, whereas only two bytes are applicable for GSM and the two first bytes are then zeros, for example, 00001A02

AT+COPS Operator selection (ver. 2)

Description: Forces an attempt to select and register the GSM/UMTS network operator. <mode> is used to select whether the selection is done automatically by the ME or is forced by this command to operator <oper>, given in format <format>. If the selected operator is not available, no other operator is selected, except when <mode>=4. The selected operator name format applies to further read commands, +COPS? also. <mode>=2 forces an attempt to deregister from the network. The selected mode affects to all further network registration, for example, after <mode>=2, ME will be unregistered until <mode>=0 or 1 is selected. This command is abortable when registration/deregistration attempt is made.
Read command returns the current mode and the currently selected operator. If no operator is selected, <format> and <oper> are omitted.
Test command returns a list of quadruplets, each representing an operator present in the network. A quadruplet consists of an integer indicating the availability of the operator <stat>, long and short alphanumeric format of the name of the operator and numeric format representation of the operator. Any of the formats may be unavailable and should then be an empty field. The list of operators is in order: home network, networks referenced in SIM and other networks.
It is recommended (although optional) that after the operator list, TA returns lists of supported <mode>s and <format>s. These lists are delimited from the operator list by two commas.

Set command: **AT+COPS=**[<mode>[,<format>[,<oper>[,AcT]]]]

Read command: **AT+COPS?**

Read command response: +COPS: <mode>[,<format>,<oper>]

Test command: **AT+COPS=?** Shows if the command is supported.

Test command response: +COPS: [list of supported (<stat>,long alphanumeric <oper>,short alphanumeric <oper>,numeric <oper>)s][,,(list of supported <mode>s),(list of supported <format>s)]

Parameters:

<mode>:

<mode>	Description
0	Automatic (<oper> field is ignored.) **Default value**
1	Manual (<oper> field will be present)
2	Deregister from network. **Not supported**
3	Set only <format> (for read command +COPS?), do not attempt registration/deregistration (<oper> field is ignored). This value is not applicable in read command response
4	Manual/automatic (<oper> field will be present). If manual selection fails, automatic mode (<mode>=0) is entered

<format>:

<format>	Description
0	Long format alphanumeric <oper>. **Default value.**
1	Short format alphanumeric <oper>
2	Numeric <oper>

<oper>:

<oper>	Description
string type	<format> indicates if the format is alphanumeric or numeric. Long alphanumeric format can be up to 16 characters long and short format up to 8 characters (see GSM MoU SE.13). Numeric format is the GSM Location Area Identification number (see GSM 04.08) which consists of a three BCD digit country code coded as in ITU-T E.212 Annex A, plus a two BCD digit network code, which is administration specific: Returned <oper> will not be in BCD format, but in IRA characters converted from BCD. Hence the number has the structure: (country code digit 3)(country code digit 2)(country code digit 1)(network code digit 2)(network code digit 1)

<stat>:

<stat>	Description
0	Unknown
1	Available
2	Current
3	Forbidden

<AcT>:

<stat>	Description
0	GSM. **Default value**
1	GSM Compact. **Not supported**
2	UTRAN

AT+CLIP Calling line identification (ver. 2)

Description: This command refers to the GSM/UMTS supplementary service CLIP (Calling Line Identification Presentation) that enables a called subscriber to get the Calling Line Identity (CLI) of the calling party when receiving a mobile terminated call. Set command enables or disables the presentation of the CLI at the terminal equipment. It has no effect on the execution of the supplementary service CLIP in the network.
Read command gives the status of <n> and also triggers an interrogation of the provision status of the CLIP service according to 3GPP TS 22.081 (given in <m>).
Test command returns values supported by the phone as a compound value.
This command activates the result code **+CLIP**.

Set command: **AT+CLIP=**<n>

Read command: **AT+CLIP?**

Read command response: +CLIP:<n>,<m>

Test command: **AT+CLIP=?** Shows if the command is supported.

Test command response: +CLIP: (list of supported <n>s)

Parameters:

<n>: Sets/shows the result code representation status in the phone.

<n>	Description
0	Disable. **Default value**
1	Enable

<m>: Shows the subscriber CLIP service status in the network.

<m>	Description
0	CLIP not provisioned
1	CLIP provisioned
2	Unknown, for example, no network

Note: When CLI is not available (<CLI validity>=2), <number> will be an empty string (““) and <type> value will not be significant. Nevertheless, the phone may return the recommended value 128 for <type> (TON/NPI unknown in accordance with GSM 04.08 subclause 10.5.4.7).
When CLI has been withheld by the originator, (<CLI validity>=1) and the CLIP is provisioned with the “override category” option (refer to 3GPP TS 22.081 and 3GPP TS 23.081), <number> and <type> is provided. Otherwise, the phone will return the same setting for <number> and <type> as if the CLI was not available.

AT+CLIR Calling line identification restriction

Description: This command refers to CLIR-service according to 3GPP TS 22.081 that allows a calling subscriber to enable or disable the presentation of the CLI to the called party when originating a call.
Set command overrides the CLIR subscription when temporary mode is provisioned as a default adjustment for all following outgoing calls. Using the opposite command can revoke this adjustment. If this command is used by a subscriber without provision of CLIR in permanent mode the network will act according to 3GPP TS 22.081.
Set command writes directly to non-volatile memory so that the setting is preserved also after turning off/on the MS. The &F command does not affect the setting.
Read command gives the default adjustment for all outgoing calls (given in <n>) and also triggers an interrogation of the provision status of the CLIR service (given in <m>).
Test command returns values supported by the TA as a compound value.

Set command: **AT+CLIR=**[<n>]

Read command: **AT+CLIR?** Displays the current <n> and <m> settings.

Test command: **AT+CLIR=?** Shows if the command is supported.

Test command response: +CLIR: (list of supported <n>s)

Parameters:

<n>:

<n>	Description
0	Presentation indicator is used according to the subscription of the CLIR service. **Default value**
1	CLIR invocation, that is, number is hidden
2	CLIR suppression, that is, number is shown

<m>:

<m>	Description
0	CLIR not provisioned
1	CLIR provisioned in permanent mode
2	Unknown, for example, no network
3	CLIR temporary mode presentation restricted
4	CLIR temporary mode presentation allowed

AT+CCFC Calling forwarding number and conditions (ver. 2)

Description: Allows control of the call forwarding supplementary service according to 3GPP TS 22.082. Registration, erasure, activation, deactivation and status query are supported. When querying the status of a network service (<mode>=2) the response line for “not active” case (<status>=0) should be returned only if the service is not active for any <class>.

Execution command: **AT+CCFC=**<reason>,<mode>[,<number>[,<type>[,<class>[,<subaddr>[,<satype>[,<time>]]]]]]

Execution command response: When <mode>=2 and command successful:
+CCFC:<status>,<class1>[,<number>,<type>,<subaddr>,<satype>[,<time>]]][**<CR><LF>**
+CCFC:<status>,<class2>[,<number>,<type>[,<subaddr>,<satype>[,<time>]]]
[...]]

Test command: **AT+CCFC=?** Shows if the command is supported.

Test command response: +CCFC: (list of supported <reason>s)

Parameters:

<reason>:

<reason>	**Description**
0	Unconditional
1	Mobile busy
2	No reply
3	Not reachable
4	All call forwarding (refer 3GPP TS 22.030)
5	All conditional call forwarding (refer 3GPP TS 22.030)

<mode>:

<mode>	**Description**
0	Disable
1	Enable
2	Query status
3	Registration
4	Erasure

<number>:

<number>	**Description**
String type	Phone number of forwarding address in format specified by <type>

<type>:

<type>	Description
Integer format	Type of address octet in integer format (GSM 04.08). Default value is 145 when dialling string includes international access code character “+”, otherwise 129
129	ISDN/telephony numbering plan, national/international unknown. **Default value if “+” is not in <sca>**
145	ISDN/telephony numbering plan, international number. **Default value if '+' is in <sca>**
161	ISDN/telephony numbering plan, national number
128–255	Other values refer GSM 04.08 section 10.5.4.7

<subaddr>:

<subaddr>	Description
string type	String type subaddress of format specified by <satype> **Not supported**

<satype>:

<satype>	Description
integer type	Type of subaddress octet **Not supported**

<classx>:

<classx>	Description
Integer	Sum of integers each representing a class of information. **Default value = 7**
1	Voice L1
2	Data
4	Fax
8	Short message service
16	Data circuit sync
32	Data circuit async
64	Dedicated packet access
128	Dedicated PAD access

<time>:

<time>	Description
1–30	When no reply is enabled or queried, this gives the time in seconds to wait before a call is forwarded, default value is 20. **Not supported**

<status>:

<status>	Description
0	Not active
1	Active

AT+CCWA Call waiting (ver. 2)

Description: Allows control of the Call Waiting supplementary service according to 3GPP TS 22.083. Activation, deactivation and status query are supported. When querying the status of a network service (<mode>=2) the response line for 'not active' case (<status>=0) should be returned only if service is not active for any <class>. Parameter <n> is used to disable/enable the presentation of an unsolicited result code **+CCWA**: <number>,<type>,<class> to the TE when call waiting service is enabled. Command is abortable when network is interrogated.
The interaction of this command with other commands based on other GSM/UMTS supplementary services is described in the GSM/UMTS standards.

Execution command: **AT+CCWA=**[<n>[,<mode>[,<class>]]]

Execution command response:
When<mode>=2 and command successful:
+CCWA:<status>,<class1>[**<CR><LF>**
+CCWA: <status>,<class2>
[…]]

Read command: **AT+CCWA?**

Read command response:
+CCWA: <n>

Test command: **AT+CCWA=?** Shows if the command is supported.

Test command response:
+CCWA: (list of supported <n>s)

Parameters:

<n>:

<n>	Description
0	Disable. **Default value**
1	Enable

<mode>:

<mode>	Description
0	Disable
1	Enable
2	Query status

<classx>:

<classx>	Description
Integer	Sum of integers each representing a class of information. **Default value=3**
1	Voice L1
2	Data
4	Fax
8	Short message service
16	Data circuit sync
32	Data circuit async
64	Dedicated packet access
128	Dedicated PAD access

<status>:

<status>	Description
0	Not active
1	Active

AT+CHLD Call hold and multiparty (ver. 1)

Description: This command refers to a service that allows a call to be temporarily disconnected from the ME, but the connection is to be retained by the network and to a service that allows multiparty conversation. Calls can be put on hold, recovered, released and added to conversation similarly as defined in GSM 02.30.
This is based on the GSM supplementary services HOLD (Call Hold) (refer GSM 02.83 clause 2) and MPTY (MultiParty, see GSM 02.84). The interaction of this command with other commands based on other GSM supplementary services is described in the GSM standard.
Note: Call Hold and MultiParty are only applicable to teleservice 11.
It is recommended (although optional) that test command returns a list of operations which are supported. The call number required by some operations will be denoted by “x”. For example, +CHLD: (0,1,1x,2,2x,3).

Set command: **AT+CHLD=**<n>

Test command: **AT+CHLD=?** Shows if the command is supported.

Test command response: +CHLD: (list of supported <n>s)

Parameter:

<n>: Integer type. Equals numbers entered before SEND button in GSM 02.30 subclause 4.5.5.1.

<n>	Description
0	Releases all held calls or sets User Determined User Busy (UDUB) for a waiting call
1	Releases all active calls and accepts the other (waiting or held) call

<n>	Description
1X	Releases the specific active call X
2	Places all active calls on hold and accepts the other (held or waiting) call
2X	Places all active calls, except call X, on hold
3	Adds a held call to the conversation
4	Connects two calls and disconnects the subscriber from both calls

Note: "X" is the numbering (starting with 1) of the call given by the sequence of setting up or receiving the calls (active, held or waiting) as seen by the served subscriber. Calls hold their number until they are released. New calls take the lowest available number. Where both a held and a waiting call exists, the above procedures will apply to the waiting call (that is, not to the held call) in conflicting situation.
The "directory number" case will be handled with dial command D and the END case with hangup command H (or +CHUP).

AT+CSSN Supplementary service notification (ver. 2)

Description: This command refers to supplementary service related network initiated notifications. The set command enables/disables the presentation of notification result codes from TA to TE.
When <n>=1 and a supplementary service notification is received after a mobile originated call setup, the unsolicited result code **+CSSI**: <code1>[,<index>] is sent to TE before any other MO call setup result codes presented in this ETS or in V.25ter. When several different <code1>s are received from the network, each of them has its own +CSSI result code.
When <m>=1 and a supplementary service notification is received during a mobile terminated call setup, during a call or when a forward check supplementary service notification is received, the unsolicited result code **+CSSU**: <code2>[,<index>] is sent to TE. In case of ME call setup, result code is sent after every **+CLIP** result code and when several different <code2>s are received from the network, each of them has its own +CSSU result code.
Test command returns values supported by the TA as a compound value.

Set command: **AT+CSSN=**[<n>[,<m>]]

Read command: **AT+CSSN?**

Read command response: +CSSN: <n>,<m>

Test command: **AT+CSSN=?** Shows if the command is supported.

Test command response: +CSSN: (list of supported <n>s),(list of supported <m>s)

Parameters:

<n>:

<n>	Description
0	Disable the **+CSSI** result code presentation status in the TA. **Default value**
1	Enable the **+CSSI** result code presentation status in the TA

<m>:

<m>	Description
0	Disable the **+CSSU** result code presentation status in the TA. **Default value**
1	Enable the **+CSSU** result code presentation status in the TA

<code1>:

<code1>	Description
0	Unconditional call forwarding is active
1	Some of the conditional call forwarding are active
2	Call has been forwarded
3	Call is waiting

<code1>	Description
5	Outgoing calls are barred
6	Incoming calls are barred
7	CLIR suppression rejected

<index>:

<index>	Description
0–9	CUG index
10	No index (preferred CUG taken from subscriber data)

<code2>:

<code2>	Description
0	This is a forwarded call (MT call setup)
1	This is a CUG call (also <index> present) (MT call setup)
2	Call has been put on hold (during a voice call)
3	Call has been retrieved (during a voice call)
4	Multiparty call entered (during a voice call)
5	Call on hold has been released (during a voice call). This is not a SS notification
6	Forward check SS message received (can be received whenever). **Not supported**

AT+CAOC Advice of charge

Description: This refers to the Advice of Charge supplementary service (GSM 02.24 and GSM 02.86) that enables a subscriber to get information about the cost of calls. With <mode>=0, the execution command returns the Current Call Meter (CCM) value from the ME. The command also includes the possibility to enable an unsolicited event reporting of the CCM information. The unsolicited result code **+CCCM**: <ccm> is sent when the CCM value changes, but no more than every 10 seconds. Deactivation of the unsolicited event reporting is made with the same command.
Read command indicates whether the unsolicited reporting is activated or not. Read command is available when the unsolicited result code is supported.

Execution command: **AT+CAOC**[=<mode>]

Read command: **AT+CAOC** Displays the current <mode> setting.

Test command: **AT+CAOC=?** Shows if the command is supported.

Test command response: +CAOC: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	Query CCM value. **Default value**
1	Deactivate the unsolicited reporting of CCM value
2	Activate the unsolicited reporting of CCM value

<ccm>:

<ccm>	Description
String type	Three bytes of the current call meter value in hexadecimal format. For example, “00001E” indicates decimal value 30. The value is in home units and bytes are similarly coded as the ACMmax value in the SIM

AT+CACM Accumulated call meter (ver. 2)

Description: Resets the Advice of Charge related Accumulated Call Meter (ACM) value in SIM file EF_{ACM}. ACM contains the total number of home units for both the current and preceding calls. SIM PIN2 is usually required to reset the value.
Note: This command must take into account what line is chosen via the MMI.
Read command returns the current value of ACM.

Set command: **AT+CACM=**[<passwd>]

Read command: **AT+CACM?**

Read command response: +CACM: <acm>

Test command: **AT+CACM=?** Shows if the command is supported.

Parameters:

<passwd>:

<passwd>	Description
String type	SIM – PIN2

<acm>:

<acm>	Description
String type	Accumulated call meter value. Similarly coded as <ccm> under **AT+CAOC**

AT+CAMM Accumulated call meter maximum

Description: Sets the maximum Advice-of-Charge related Accumulated Call Meter (ACM) value in the SIM file EF_{ACMmax}. ACMmax contains the maximum number of home units allowed to be consumed by the subscriber. When ACM (see **AT+CACM**) reaches ACMmax, calls are prohibited (see also 3GPP 22.024). SIM PIN2 is usually required to set the value.

Set command: **AT+CACM=**[<acmmax>[,<passwd>]]

Read command: **AT+CAMM?** Displays the current <acmmax> value

Test command: **AT+CAMM=?** Shows if the command is supported

Parameters:

<passwd>:

<passwd>	Description
String	SIM – PIN2

<acmmax>:

<acmmax>	Description
String	Accumulated call meter maximum value. Similarly coded as <ccm> under **AT+CAOC**. The value “0” disables the ACMmax feature

AT+CDIP Called line identification presentation

Description: This command relates to a network service that provides “multiple called numbers (called line identifications) service” to an MT. This command enables a called subscriber to get the called line identification of the called party when receiving a mobile terminated call. Set command enables or disables the presentation of the called line identifications at the TE.
When the presentation of the called line identification at the TE is enabled, +CDIP:<number>,<type>[,<subaddr>,<satype>] response is returned after every **RING** (or **+CRING**: <type>) result code sent from TA to TE. The manufacturer specifies if this response is used when normal voice call is answered.
Read command gives the status of <n> and also triggers an interrogation of the provision status of the “multiple called numbers” service.

Set command: **AT+CDIP=**[<n>]
Enables/disables a called subscriber to get the called line identification of the called party when receiving a mobile terminated call.

Read command: **AT+CDIP?** Displays the current <n> and <m> settings.

Read command response: +CDIP: <n>,<m>

Test command: **AT+CDIP=?** Shows if the command is supported.

Test command response: +CDIP: (list of supported <n>s)

Parameters:

<n>:

<n>	Description
0	Disable presentation of +CDIP result code. **Default value**
1	Enable presentation of +CDIP result code

<m>:

<m>	Description
0	”Multiple called numbers service” is not provisioned
1	”Multiple called numbers service” is provisioned
2	Unknown (no network, and so on)

AT+COLP Connected line identification presentation

Description: This command refers to the GSM/UMTS supplementary service COLP (Connected Line Identification Presentation) that enables a calling subscriber to get the connected line identity (COL) of the called party after setting up a mobile originated call. The command enables or disables the presentation of the COL at the terminal equipment. It has no effect on the execution of the supplementary service COLR in the network.
When enabled (and called subscriber allows), +COLP: <number>,<type>[,<subaddr>,<satype> [,<alpha>]] intermediate result code is returned from the phone to terminal equipment before any **+CR** or V.25ter responses.

Execution command: **AT+COLP=**[<n>]

Read command: **AT+COLP?** Displays the current <n> and <m> settings.

Test command: **AT+COLP=?** Shows if the command is supported.

Test command response: +COLP: (list of supported <n>s)

Parameters:

<n>: Sets/shows the result code presentation status in the phone.

<n>	Description
0	Enable result code presentation status. **Default value**
1	Disable result code presentation status

<m>: Shows the subscriber COLP service status in the network.

<m>	Description
0	COLP not provisioned
1	COLP provisioned
2	Unknown, for example, no network

AT+CPOL Preferred operator list

Description: Edits the user preferred list of networks in the active application on the UICC (GSM or USIM) or preferred list of networks in the SIM card. Execution command writes an entry in the SIM list of preferred operators ($EFPLMN_{sel}$), when the SIM card is present or when the UICC is present with an active GSM application. When UICC is present with an active USIM application, execution commands writes an entry in the User controlled PLMN selector with Access Technology list (EFPLMNwAcT), only the PLMN field could be entered, the Access Technologies for each PLMN in this list is not accessible with this command (New command for accessing the Access Technologies for each PLMN in this list is FFS). If <index> is given but <oper> is left out, entry is deleted. If <oper> is given but <index> is left out, <oper> is put in the next free location. If only <format> is given, the format of the <oper> in the read command is changed.
Note: ME may also update this list automatically when new networks are selected.
Read command returns all used entries from the active application in the UICC (GSM or USIM) user preferred list of networks or SIM card list of preferred operators.
Test command returns the whole index range supported by the active application in the UICC (GSM or USIM) user preferred list of networks or SIM card.

Execution command: **AT+CPOL=**[<index>][, <format>[,<oper>]]

Read command: **AT+CPOL?**

Read command response:
+CPOL: <index1>,<format>,<oper1>[**<CR><LF>**
+CPOL: <index2>,<format>,<oper2>
[...]]

Test command: **AT+CPOL=?** Shows if the command is supported.

Test command response: +CPOL: (list of supported <index>s),(list of supported <format>s)

Parameters:

<indexn>:

<indexn>	**Description**
Integer	The order number of operator in the active application in the UICC (GSM or USIM) user preferred list of networks or SIM card preferred operator list

<format>:

<format>	**Description**
0	Long format alphanumeric <oper>
1	Short format alphanumeric <oper>
2	Numeric <oper>. **Default value**

<opern>:

<opern>	Description
string type	<format> indicates if the format is alphanumeric or numeric (see **AT+COPS**)

AT+COPN Read operator names

Description: Returns the list of operator names from the ME. Each operator code <numeric*n*> that has an alphanumeric equivalent <alpha*n*> in the ME memory is returned.

Execution command: **AT+COPN**

Execution command response: +COPN: <numeric1>,<alpha1> [**<CR><LF>**+COPN: <numeric2>,<alpha2>[...]]

Test command: **AT+COPN=?** Shows if the command is supported.

Parameters:

<numeric*n*>:

<numeric*n*>	Description
String type	Operator in numeric format (see **AT+COPS**)

<alpha*n*>:

<alpha*n*>	Description
String type	Operator in long alphanumeric format (see **AT+COPS**)

AT*EDIF Divert function (ver. 2)

Description: Enables and disables notification of divert status changes with the unsolicited result code ***EDIF**.

Set command: **AT*EDIF=**<onoff>

Read command: **AT*EDIF?** Displays the current <onoff> setting.

Test command: **AT*EDIF=?** Shows if the command is supported.

Test command response: *EDIF: (List of supported <onoff>s)

Parameter:

<onoff>:

<onoff>	Description
0	Disable notification with the unsolicited result code *EDIF
1	Enable notification with the unsolicited result code *EDIF

AT*EIPS Identify presentation set

Description: Enables or disables the presentation of the alpha tag (first name and last name) of the caller ID and called ID to the terminal equipment if the ID is recognised. The presentation is performed by unsolicited result codes, ***ELIP** for caller ID and ***EOLP** for called ID.

Set command: **AT*EIPS=**<ID>,<alphatag_mode>

Read command: **AT*EIPS?** Displays the current parameter settings.

Read command response:
*EIPS: <ID1>,<alphatag_mode1>**<CR><LF>**
*EIPS: <ID2>,<alphatag_mode2>

Test command: **AT*EIPS=?** Shows if the command is supported.

Test command response: *EIPS: (List of supported <ID>s),(list of supported <alphatag_mode>s)

Parameters:

<ID>:

<ID>	**Description**
1	Caller ID (***ELIP**)
2	Called ID (***EOLP**)

<alphatag_mode>:

<alphatag_mode>	**Description**
0	Off
1	First name and last name displayed

Unsolicited result codes

+CREG Network registration

Description: Indicates that there is a change in the phone network registration status. This result code is enabled by using **AT+CREG**.

Unsolicited result code: **+CREG:** <stat>

Parameter:

<stat>:

<stat>	**Description**
0	Not registered – The phone is currently not searching for a new operator to register to
1	Registered – Home network
2	Not registered – The phone is currently searching for a new operator to register to
3	Registration denied

<stat>	Description
4	Unknown
5	Registered – Roaming

+CLIP Calling line identification indication (ver. 2)

Description: Enables a called subscriber to get the calling line identity (CLI) of the calling party when receiving a mobile terminated call.
This result code is activated by **AT+CLIP**.

Unsolicited result code: **+CLIP:**<number>,<type>[,<subaddr>,<satype>[,[<alpha>][,<CLI validity>]]]

Parameters:

<number>:

<number>	Description
String type	Phone number of format specified by <type>

<type>:

<type>	Description
Integer format	Type of address octet Refer to (*Digital cellular telecommunications system (Phase 2) (GSM); Mobile radio interface; Layer 3*, section 10.5.4.7)
129	ISDN/telephony numbering plan, national/international unknown. **Default value if "+" is not in <sca>**
145	ISDN/telephony numbering plan, international number. **Default value if "+" is in <sca>**
161	ISDN/telephony numbering plan, national number
128–255	Other values, refer to (*Digital cellular telecommunications system (Phase 2) (GSM); Mobile radio interface; Layer 3*, section 10.5.4.7)

<subaddr>:

<subaddr>	Description
String type	String type subaddress of format specified by <satype>. As described in ITU_T I.330: “The subaddress is a sequence of digits, the maximum length of which will be 20 octets (40 digits). All ISDNs will be capable of conveying the ISDN subaddress transparently and will not be required to examine or operate on any of the subaddress information. Special attention is drawn to the fact that subaddressing is not to be considered as part of the numbering plan, but constitutes an intrinsic part of ISDN addressing capabilities. The subaddress will be conveyed in a transparent way as a separate entity from both ISDN number and user-to-user information. See also Recommendation I.334”

<satype>:

<satype>	Description
Integer format	Type of subaddress octet
128	NSAP (X.213/ISO 8348 AD2), even number of address signals
136	NSAP (X.213/ISO 8348 AD2), odd number of address signals
160	User defined, even number of address signals
168	User defined, odd number of address signals
128–255	Other values reserved

<alpha>:

<alpha>	Description
String type	Optional string type alphanumeric representation of <number> corresponding to the entry found in phonebook. Used character set should be the one selected with command **AT+CSCS**

<CLI_validity>:

<CLI_validity>	Description
0	CLI valid
1	CLI has been withheld by the originator
2	CLI is not available due to interworking problems or limitations of originating network

*ELIP Calling line alpha tag

Description: This result code is returned after every **RING** (or **+CRING**) result code sent from phone to terminal equipment. This response is also sent when a normal voice call is answered. This result code is enabled by using **AT*EIPS**.

Unsolicited result code: ***ELIP:** <alpha_tag>

Parameter:

<alpha_tag>: String. A text with the first name and last name of the caller ID.

*EOLP Connected line alpha tag

Description: This result code is returned after every **RING** (or **+CRING**) result code sent from phone to terminal equipment. This response is also sent when a normal voice call is answered. This result code is enabled by using **AT*EIPS**.

Unsolicited result code: ***EOLP:** <alpha_tag>

Parameter:

<alpha_tag>: String. A text with the first name and last name of the called ID.

+CCWA Call waiting notification

Description: This unsolicited result code displays the specifics concerning the call waiting supplementary service. This result code is enabled by using **AT+CCWA**.

Unsolicited result code: **+CCWA:** <number>,<type>,<class>

Parameters:

<number>: String. Phone number. Format specified by <type>.

<type> Integer. Type of address octet.

<class>: Integer. Sum of integers, each representing a class of information.

<class>	Description
1	Voice L1
128	Voice L2

+CSSI Supplementary service notification

Description: Refers to supplementary service related network-initiated notifications. This unsolicited result code is sent when AT+CSSN <n>='1' and a supplementary service notification is received after a mobile-originated call setup. This result code is enabled by using **AT+CSSN**.

Unsolicited result code: **+CSSI:** <code1>[,<cindex>]

Parameters:

<code1>:

<code1>	Description
0	Unconditional call forwarding is active
1	Some of the conditional call forwardings are active
2	A call has been forwarded
3	A call is waiting
5	Outgoing calls are barred
6	Incoming calls are barred
7	CLIR suppression rejected
8	This is a CUG call (<cindex> present)

<cindex>: Integer. CUG index. Range: 0–32767.

+CSSU Supplementary service notification

Description: Refers to supplementary service related network-initiated notifications. This unsolicited result code is sent when AT+CSSN <m>='1' and a supplementary service notification is received during a mobile-originated call setup, during a call or when a forward check supplementary service notification is received. This result code is enabled by using **AT+CSSN**.

Unsolicited result code: **+CSSU:** <code2>[,<cindex>]

Parameters:

<code2>:

<code2>	Description
0	This is a forwarded call
2	A call has been put on hold (during voice call)
3	A call has been retrieved (during voice call)
4	A multiparty call entered (during voice call)
5	The call on hold has been released (during voice call). This is not an SS notification
6	Forward check SS messages received (can be received whenever)
10	This is a CUG call (<cindex> present)

<cindex>: Integer. CUG index. Range: 0–32767.

+CCCM Advice of charge call meter notification

Description: This unsolicited result code is sent when the CCM value changes, but not more often than every 10 seconds. The result code is enabled by using **AT+CAOC**.

Unsolicited result code: **+CCCM:** <ccm>

Parameter:

<ccm>: String. Hexadecimal form of three bytes of the current call meter value. The value is in home units and the bytes are coded similarly as the ACMmax value in the SIM.

*EDIF Divert function (ver. 2)

Description: This unsolicited result code will be generated when a divert notification is sent from the network provider. These notifications could be sent when a service is actived, deactived or interrogated.
If a notification affects more than one class, then the <classx> will be reported as a sum of the affected classes. If, for example, no bearerservice is active the unsolicited report code will be:
*EDIF: 0, 0, 240,"",0
The phonenumber, <number>, will be empty and the <type> will be set to zero if a service is erased. If a service is registered but not activated the response will contain the registered <number> and the <type>.

Unsolicited result code: *EDIF: <reason>,<status>,<classx>[,<number>,<type>]

Parameters:

<reason>:

<reason>	Description
0	**Unconditional** (CFU – Call Forwarding Unconditional). This service lets a called mobile subscriber have the network send incoming calls to the called mobile subscriber directory number or to another directory number. The subscriber can send all calls, or just those associated with a specific basic service group. CFU forwards all calls without regard to the condition. It does not matter if the phone is on or off
1	**Mobile busy subscriber** (CFB – Call forwarding on mobile busy subscriber). This service lets a called mobile subscriber have the network send incoming calls which meet mobile subscriber busy to another directory number. The subscriber can send all calls, or just those associated with a specific basic service group
2	**No reply** (CFNny – Call forwarding on no reply). This service lets a called mobile subscriber have the network send incoming calls which meet no reply to another directory number. The subscriber can send all calls, or just those associated with a specific basic service group

<reason>	Description
3	**Not reachable** (CFNrc – Call forwarding on not reachable). This service lets a called mobile subscriber have the network send incoming calls that do not reach the subscriber number to another directory number. The subscriber can send all calls, or just those associated with a specific basic service group
4	**All call forwarding**. This code describes all call forwarding services. It is used in the deactivation command dialogue if a user disables all services in one action
5	**All conditional call forwarding.** This code describes all conditional call forwarding services

<status>:

<status>	Description
0	Disabled
1	Enabled. The phone is diverted for the <reason> above

<class*x*>: Integer. Bit field representing the affected service.

<class*x*>	Description
1	Voice L1 (teleservice)
2	Data (teleservice)
4	Fax (teleservice)
8	SMS (teleservice)
16	Data circuit sync (bearerservice)
32	Data circuit async (bearerservice)
64	Dedicated packet access (bearerservice)
128	Dedicated PAD access (bearerservice)

<number>: String. Phone number of forwarding address. Format specified by <type>.

<type>: Integer. Type of address octet.

<type>	Description
Integer	Type of address octet in integer format. **Default value** is 145 when dialing string includes international access code character “+”, otherwise 129. **Note**: Some network providers force the type of number to international, that is, “+” will always be appended and the <type> will be set to 145, regardless of the format of the input. * Output MSB TypeOfNumber NumberPlanID * 128 0x80 0 (unknown) 0 (unknown) * 129 0x80 0 (unknown) 1 (ISDN/Telephony) * 144 0x80 1 (international) 0 (unknown) * 145 0x80 1 (international) 1 (ISDN/Telephony) * 161 0x80 2 (national) 1 (ISDN/Telephony)

+COLP Connected line identification indication

Description: Enables a calling subscriber to get the connected line identity (COL) of the called party when setting up a mobile originated call. This result code is activated by **AT+BVRA**.

Unsolicited result code: **+COLP:** <number>,<type>[,<subaddr>,<satype> [,<alpha>]]

Parameters: See the **+CLIP** result code.

+CDIP Called line identification presentation

Description: Returned after every **RING** (or **+CRING**: <type>) result code sent from TA to TE. This result code is activated by the **AT+CDIP** command.

Unsolicited result code: **+CDIP:**<number>,<type>[,<subaddr>,<satype>]

Parameters: See the **+CLIP** (version 2) result code.

Use scenarios

Calling line identification

This use scenario performs the following steps:

1. Enable calling line identification
2. Receive calling line identity indication when receiving a mobile-terminated call
3. Disable calling line identification

AT command	Response	Comment
`AT+CLIP=1`		Enable calling line identification

AT command	Response	Comment
	OK	
	+CRING: VOICE +CLIP: ”0706123456”, 129	After every CRING, the calling line identity is presented
		Reject call
AT+CLIP?		
	+CLIP: 1,1 OK	CLIP enabled and provisioned
AT+CLIP=0		Disable calling line identification
	OK	

Call hold and multiparty

This use scenario uses the call hold functionality to switch between two calls.

AT command	Response	Comment
AT+CCWA=1,1		Activate call waiting
ATD046193000;	OK	Originate a voice call
	+CCWA: ”+46706123456”, 145	Another call is waiting
AT+CHLD=2		Put first call on hold and answer the second call
	OK	
AT+CHLD		Release the second call and recover the first call
	OK	

Ensemble S7: GSM USSD

Commands

AT+CUSD Unstructured supplementary service data (ver. 2)

Description: Allows control of the Unstuctured Supplementary Service Data (USSD) according to 3GPP TS 22.090. Both network and mobile initiated operations are supported. Parameter <n> is used to disable/enable the presentation of an unsolicited result code (USSD response from the network or network initiated operation) **+CUSD**: <m>[,<str>] to the TE. In addition, value <n>=2 cancels an ongoing USSD session.
When <str> is given, a mobile initiated USSD-string or a response USSD-string to a network initiated operation is sent to the network. The response USSD-string from the network is returned in a subsequent unsolicited +CUSD: result code.
The interaction of this command with other commands based on other GSM/UMTS supplementary services is described in the GSM/UMTS standards.
Test command returns values supported by the TA as a compound value. In one session only the ME or the accessory can be active and send USSD-strings.

Execution command: **AT+CUSD=**[<n>[,<str>[,<dcs>]]]

Read command: **AT+CUSD?**

Read command response: +CUSD: <n>

Test command: **AT+CUSD=?** Shows if the command is supported.

Test command response: +CUSD: (list of supported <n>s)

Parameters:

<n>:

<n>	Description
0	Disable result code presentation in the TA. **Default value**
1	Enable result code presentation in the TA
2	Terminate (abort) USSD dialogue. This value is not applicable to the read command response. **Not supported**

<str>:

<str>	Description
string	USSD-string (when <str> parameter is not given, network is not interrogated)

<str>	Description
If <dcs> indicates that 3GPP TS 23.038 7-bit default alphabet is used:	• If TE character set other than “HEX” (refer command Select TE Character Set +CSCS): ME/TA converts GSM alphabet into current TE character set according to rules of 3GPP TS 27.005 Annex A • If TE character set is “HEX”: ME/TA converts each 7-bit character of GSM alphabet into two IRA character long hexadecimal number. For exampe, character P (GSM 23) is presented as 17 (IRA 49 and 55)
If <dcs> indicates that 8-bit data coding scheme is used:	ME/TA converts each 8-bit octet into two IRA character long hexadecimal number. For example, octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65)

<dcs>:

<dcs>	Description
Integer	3GPP TS 23.038 Cell Broadcast Data Coding Scheme in integer format. **Default value = 0**

Scenarios:

1a An incoming network initiated USSD-Notify should be presented on the ME display.
1b An incoming network initiated USSD-Notify should also be presented to the accessory as a unsolicited result code +CUSD: if the accessory has Enabled result code presentation.
2 An incoming USSD-request asking for a reply should be presented both on the display of the ME and to the accessory as a unsolicited result code +CUSD: if the accessory has Result code enabled.
2a If the ME answer to the request then the accessory should get a +CUSD: telling the accessory that Other I/O client has responded.
2b If the accessory answer to the request with the command AT+CUSD then the ME is notified of the answer but there should be no presentation of the reply on the ME display. The display should be cleared.
3a A USSD request initiated and sent from the ME should not be presented to the accessory.
3b A USSD request sent with the command AT+CUSD from the accessory should not be presented on the display of the ME.

	Network	Mobile Equipment	Mobile Accessory
1a	Signal ->	Show in display	Result code presentation disabled, Not presented to accessory
1b			Result code presentation enabled, Presented to accessory
2	Signal asking for reply ->	Show in display	Result code presentation disabled, Not presented to accessory
			Result code presentation enabled, Presented to accessory
2a		Answer	
		Other I/O client has responded	

	Network	Mobile Equipment	Mobile Accessory
2b		Answer from accessory not shown in display. Display cleared.	Answer
3a		Edit in display and send to network	
3b		Signal from accessory not shown in display	Signal

Unsolicited result codes

+CUSD CUSD indication

Description: Indicates a network-initiated operation. This command is enabled by using **AT+CUSD**.

Unsolicited result code: **+CUSD:** <m>[,<str>,<dcs>]

Parameters:

<m>:

<m>	Description
0	No further user action needed. (Network-initiated USSD notify or no further information needed after mobile-initiated operation)
1	Further user action needed. (Network-initiated USSD request or further information needed after mobile-initiated operation)
2	USSD dialogue terminated
3	Other I/O client has responded. This result code is received if the network initiates a USSD dialogue and some other I/O client responds
4	Operation not supported
5	Network time out

<str>: String. USSD string.

<dcs>: Integer. Cell broadcasting Data Coding Scheme.

Ensemble S8: GSM facility lock

Commands

AT+CLCK Facility lock (ver. 5)

Description: Execution command is used to lock, unlock or interrogate an ME or a network facility <fac>. Password is normally needed to do such actions. When querying the status of a network service (<mode>=2) the response line for “not active” case (<status>=0) should be returned only if service is not active for any <class>. This command should be abortable when network facilities are set or interrogated
Call barring facilities are based on GSM/UMTS supplementary services (refer to 3GPP TS 22.088). The interaction of these with other commands based on other GSM/UMTS supplementary services is described in the GSM/UMTS standard.
Test command returns facility values supported by the phone as a compound value.
Note:

- “PS” and <mode>=1 correspond to Auto Lock
- Which <passwd> (PIN-code) that will be used for authentication is manufacturer specific.

Execution command: **AT+CLCK=**<fac>,<mode>[,<passwd>[,<class>]]

Execution command response:
When <mode>=2 and command successful:
+CLCK: <status>[,<class1>[**<CR><LF>**
+CLCK: <status>,<class2>
[...]]

Test command: **AT+CLCK=?** Shows if the command is supported.

Test command response: +CLCK: (list of supported <fac>s)

Parameters:

<fac>:

<fac>	**Description**
“CS”	CNTRL (lock control surface), for example, phone keyboard
“PS”	PH-SIM (lock phone to SIM card). Phone asks for password when other than current SIM card is inserted. **Not supported**
“PF”	Lock Phone to the very First inserted SIM/UICC card (also referred to as PH-FSIM). Phone asks for password when other than the first SIM/UICC card is inserted. **Not supported**

<fac>	Description
“SC”	SIM (lock SIM card). SIM asks for a password when the phone is powered-up and when this lock command is issued
“P2”	SIM PIN 2.
”AO”	BAOC (Barr All Outgoing Calls). Refer to 3GPP TS 22.088 clause 1
”OI”	BOIC (Barr Outgoing International Calls). Refer to 3GPP TS 22.088 clause 1
“AI”	BAIC (Barr All Incoming Calls). Refer to 3GPP TS 22.088 clause 2
“IR”	BIC-Roam (Barr Incoming Calls when Roaming outside the home country). Refer to 3GPP TS 22.088 clause 2
“OX”	BOIC-exHC (Barr Outgoing International Calls except to Home Country). Refer to 3GPP TS 22.088 clause 1
“NT”	Barr incoming calls from numbers that are not stored in TA memory. **Not supported**
“NM”	Barr incoming calls from numbers that are not stored in phone memory. **Not supported**
“NS”	Barr incoming calls from numbers that are not stored in SIM memory. **Not supported**
“NA”	Barr incoming calls from numbers that are not stored in any memory. **Not supported**
“AB”	All barring services. Refer to 3GPP TS 22.030 (applicable only for <mode>=0)
“AG”	All outgoing barring services, refer to 3GPP TS 22.030 (applicable only for <mode>=0)
“AC”	All incoming barring services. Refer to 3GPP TS 22.030 (applicable only for <mode>=0)
“FD”	SIM card or active application in the UICC (GSM or USIM) fixed dialling memory feature. If PIN2 authentication has not been done during the current session, PIN2 is required as <passwd>
“PN”	Network personalisation. Refer to 3GPP TS 22.022. **Not supported**
“PU”	Network subset personalisation Refer to 3GPP TS 22.022. **Not supported**
“PP”	Service provider personalisation. Refer to 3GPP TS 22.022. **Not supported**
“PC”	Corporate personalisation. Refer to 3GPP TS 22.022. **Not supported**

<mode>:

<mode>	Description
0	Unlock

<mode>	Description
1	Lock
2	Query status

<status>:

<status>	Description
0	Not active
1	Active
2	Not available

<passwd>:

<passwd>	Description
string type	Shall be the same as password specified for the facility from the phone user interface or with command **AT+CPWD**

<classx>: A sum of integers each representing a class of information.
Default value =7

<classx>	Description
1	Voice
2	Data. Refers to all bearer services. With <mode>=2 this may only refer to some bearer service if the phone does not support values 16, 32, 64 and 128
4	Fax
8	Short message service
16	Data circuit sync
32	Data circuit async
64	Dedicated packet access
128	Dedicated PAD access

AT+CPWD Change password (Ver. 3)

Description: Sets a new password for the facility lock function defined by command Facility Lock **AT+CLCK**.
Test command returns a list of pairs which presents the available facilities and the maximum length of their password.

Action command: **AT+CPWD=**<fac>,<oldpwd>,<newpwd>

Test command: **AT+CPWD=?** Shows if the command is supported.

Test command response: +CPWD: list of supported (<fac>,<pwdlength>)s

Parameters:

<fac>:

<fac>	Description
“CS”	CNTRL (lock control surface). For example, phone keyboard. **Not supported**
“PS”	PH-SIM (lock phone to SIM card). Phone asks for a password when other than current SIM card is inserted
“SC”	SIM (Lock SIM card). SIM asks for a password when the phone is powered-up and when this lock command is issued
“P2”	SIM PIN2
“AO”	BAOC (Barr All Outgoing Calls). Refer to GSM 02.88 clause 1
“OI”	BOIC (Barr Outgoing International Calls). Refer to GSM 02.88 clause 1
“AI”	BAIC (Barr All Incoming Calls). Refer to GSM 02.88 clause 2
“IR”	BIC-Roam (Barr Incoming Calls when Roaming outside the home country). Refer to GSM 02.88 clause 2
“OX”	BOIC-exHC (Barr Outgoing International Calls except to Home Country). Refer to GSM 02.88 clause 1
“NT”	Barr incoming calls from numbers that are not stored in TA memory. **Not supported**
“NM”	Barr incoming calls from numbers that are not stored in phone memory. **Not supported**
“NS”	Barr incoming calls from numbers that are not stored in SIM memory. **Not supported**
“NA”	Barr incoming calls from numbers that are not stored in any memory. **Not supported**
“AB”	All barring services. Refer to GSM 02.30
“AG”	All outgoing barring services. Refer to GSM 02.30
“AC”	All incoming barring services. Refer to GSM 02.30
“FD”	SIM fixed dialling memory feature (if PIN2 authentication has not been done during the current session, PIN2 is required as <passwd>). **Not supported**

<oldpwd>:

<oldpwd>	Description
String type	<oldpwd> will be the same as password specified for the facility from the ME user interface or with command Change Password **AT+CPWD**

<newpwd>:

<newpwd>	Description
String type	<newpwd> is the new password. Maximum length of password can be determined with <pwdlength>

<pwdlength>:

<pwdlength>	Description
Integer type	Maximum length of the password for the facility

Use scenarios

Phone lock function

This scenario describes:

- Phone lock status query
- Set lock
- Set auto lock
- Set full lock

AT command	Response	Comment
`AT+CLCK=”PS”,2`		Query status
	`OK`	
`AT+CLCK=”SC”,1,”1234”`		Set lock
	`OK`	
`AT+CLCK=”PS”,1,”1234”`		Set automatic lock
	`OK`	
`AT+CLCK=”PS”,10,”1234”`		Set full lock
	`OK`	

Ensemble S9: Mobile equipment, control and status

Commands

AT+CFUN Set phone functionality (ver. 2)

Description: Selects the level of functionality <fun> in the MS. Level “full functionality” is where the highest level of power is drawn. “Minimum functionality” is where minimum power is drawn, that is, the ME is switched off and only the RTC clock is running.
Note: ME resetting with <rst> parameter is not supported.
Test command returns values supported by the ME as a compound value.
Note: “AT+CFUN=” is interpreted as “AT+CFUN=0”

Set command: **AT+CFUN=**[<fun>]

Read command: **AT+CFUN?** Shows the current setting.

Read command response: +CFUN: <fun>

Test command: **AT+CFUN=?** Shows if the command is supported.

Test command response: +CFUN: (list of supported <fun>s)

Parameters:

<fun>:

<fun>	Description
0	Minimum functionality, that is, the phone is turned off. **Default value**
1	Full functionality, that is, the phone is turned on
2	Disable phone transmit RF circuits only. **Not supported**
3	Disable phone receive RF circuits only. **Not supported**
4	Disable phone transmit and receive RF circuits**.** **Note**: This is often referred to as “flight mode”
5	GSM only (WCDMA radio off)
6	WCDMA only (GSM radio off)

AT+CPAS Phone activity status (ver. 3)

Description: Returns the activity status <pas> of the ME. It can be used to interrogate the ME before requesting action from the phone.
When the command is executed without the <mode> argument, the command returns <pas> values from 0 to 128. When, on the other hand, the command is executed with the <mode> argument set to 1, the command may return <pas> values from 129 to 255.
Test command returns values supported by the ME as a compound value.

Execution command: **AT+CPAS**

Execution command response: +CPAS: <pas>

Test command: **AT+CPAS=?** Shows if the command is supported.

Test command response: +CPAS: (list of supported <pas>s)

Parameters:

<pas>:

<pas>	Description
0	Ready – ME allows commands from TA/TE
1	Unavailable – ME does not allow commands from TA/TE. **Not supported**
2	Unknown – ME is not guaranteed to respond to instructions. **Not supported**
3	Ringing – ME is ready for commands from TA/TE, but the ringer is active
4	Call in progress – ME is ready for commands from TA/TE, but a call is in progress
5	Asleep – ME is unable to process commands from TA/TE because it is in a low functionality state. **Not supported**

AT+CPIN PIN control (ver. 2)

Description: The set command sends the password to the ME, which is necessary to make the ME operational (SIM PIN, SIM PUK or PH-SIM). If the PIN is to be entered twice, the TA will autonomously repeat the PIN. If no PIN request is pending, no action is taken towards the ME and an error message is returned to the TE.
If the PIN required is PUK, the second pin is required. This second PIN, <newpin>, replaces the old PIN in the SIM.

Set command: **AT+CPIN=**<pin>[,<newpin>]

Read command: **AT+CPIN?**

Read command response +CPIN: <code>

Test command: **AT+CPIN=?** Shows if the command is supported.

Test command response: +CPIN: (supported <code>s)

Parameters:

<pin><newpin>:

<pin><newpin>	**Description**
string	The range for the SIM PIN and the PH-SIM PIN is 4–8 digits. The SIM PUK consists of 8 digits. PH-NET PIN, PH-NETSUB PIN, PH-SP PIN, PH-CORP PIN, PH-ESL PIN and PH-SIMLOCK PIN are 8–16 digits

<code>:

<code>	**Description**
READY	ME is not pending for any password
SIM PIN	ME is waiting for SIM PIN to be given
SIM PUK	ME is waiting for SIM PUK to be given
PH-SIM PIN	ME is waiting for Phone lock password to be given
SIM PIN2	ME is waiting for SIM PIN2 to be given. This <code> is returned only when the last executed command resulted in PIN2 authentication failure. If PIN2 is not entered right after the failure, ME does not block its operation
SIM PUK2	ME is waiting for SIM PUK2 to be given. This <code> is returned only when the last executed command resulted in PUK2 authentication failure. If PUK2 and new PIN2 are not entered right after the failure, ME does not block its operation
PH-NET PIN	ME is waiting for network personalisation password to be given
PH-NETSUB PIN	ME is waiting for network subset personalisation password to be given
PH-SP PIN	ME is waiting for service provider personalisation password to be given
PH-CORP PIN	ME is waiting for corporate personalisation password to be given
PH-ESL PIN	Extended SIM lock
BLOCKED	The SIM card is blocked for the user

AT+CBC Battery charge (ver. 2)

Description: Execution and read command returns battery connection status <bcs> and battery level <bcl> of the phone.
Note: Even when a charger is connected, the parameter <bcl> still will give the status of the battery capacity in percent.

Execution command: **AT+CBC**

Execution command response: +CBC: <bsc>,<bcl>

Read command: **AT+CBC?** Displays the current <bcs> and <bcl> values.

Test command: **AT+CBC=?** Shows if the command is supported.

Test command response: +CBC: (list of supported <bcs>s),(list of supported <bcl>s)

Parameters:

<bcs>:

<bcs>	Description
0	Phone powered by the battery. No charger connected
1	Phone has a battery connected, but it is powered by the charger
2	Phone does not have a battery connected

<bcl>:

<bcl>	Description
0	Battery exhausted
1–99	Battery charging level. The battery has 1–99 percent of capacity remaining
100	Battery fully charged

AT+CSQ Signal quality (ver.1)

Description: Returns received signal strength indication <rssi> and channel bit error rate <ber> from the phone.

Execution command: **AT+CSQ**

Execution command response: +CSQ: <rssi>,<ber>

Test command: **AT+CSQ=?** Shows if the command is supported.

Test command response: +CSQ: (list of supported <rssi>s),(list of supported<ber>s)

Parameters:

<rssi>:

<rssi>	Description
0	-113 dBm or less
1	-111 dBm
2–30	-109 dBm to -53 dBm
31	-51 dBm or greater
99	Not known or not detectable

<ber>:

<ber>	Description
0–7	As RXQUAL values in the table in GSM 05.08 subclause 8.2.4
99	Not known or not detectable

AT+CKPD Keypad control (ver. 7)

Description: Emulates ME keypad by giving each keystroke as a character in a string <keys>. <time>*0.1 seconds is the time to strike each key and <pause>*0.1 seconds is the length of pause between two strokes. This command should be accepted (OK returned) before actually starting to press the keys. Thus unsolicited result codes of keys that have been pressed and display events can be returned (see **AT+CMER**).
The physical keypad will always have higher priority than emulation of key-strokes via AT+CKPD. That is, if the physical keypad is operated during execution of a series of keystrokes generated by AT+CKPD the emulated keypad operation is to be terminated immediately.
Note: The default GSM character set does not contain the "[" and "]" characters used to emulate the left and right selection keys. Before sending any of these keys with AT+CKPD, the character set needs to be changed, for example, to 8859-1 by sending the command AT+CSCS="8859-1".

Execution command: **AT+CKPD=**<keys>[,<time>[,<pause>]]

Test command: **AT+CKPD=?** Shows if the command is supported.

Parameters:

<keys>: String of characters representing keys as listed in the following table (based on PCCA STD-101 Annex table I – 3). Colon character (IRA 58) followed by one character can be used to indicate a manufacturer specific key not listed here. All characters from a semicolon character (IRA 59) to the next single semicolon characters are treated as alpha entries and are not converted to key equivalents. All semicolon characters inside alpha entries should be duplicated in the TE and stripped to one before entering to the ME. All IRA values not listed here are reserved.
Note: The SEND and END keypad values should be mapped to appropriate keys.

Char	IRA (dec)	Comment (+ some known key symbols)
#	35	Hash (number sign)
*****	42	Star (*)

Char	IRA (dec)	Comment (+ some known key symbols)
0–9	48– 57	Number keys
:	58	Escape character for manufacturer specific keys
<	60	Left arrow
>	62	Right arrow
C/c	67/99	Clear display (C/CLR)
D/d	68/100	Volume down
L/l	76/108	Phone lock (LOCK) **If supported by ME**
P/p	80/112	Power (PWR)
U/u	85/117	Volume up **If supported by ME**
V/v	86/118	Down arrow
[	91	Soft key 1
]	93	Soft key 2
^	94	Up arrow
:G	58+71	Go music button. **If supported by ME**
:J	58+74	Joystick button pressed
:C	58+99	Camera button (full press on camera button). **Note**: CKPD with ":F" must be called before this key command will work. **If supported by ME**
:O	58+79	Operator button. **If supported by ME**
:R	58+82	Return button
H/h	200	Button pushed on the MC link (Bluetooth) headset
:M	58+77	Video call **If supported by ME**
:F	58+70	Camera focus (camera key half press) **If supported by ME**
:(	58+40	Flip closed **If supported by ME**
:)	58+41	Flip opened **If supported by ME**
:{	58+123	Camera lens cover closed **If supported by ME**
:}	58+125	Camera lens cover opened **If supported by ME**
:[	58+91	Jack knife closed **If supported by ME**

Char	IRA (dec)	Comment (+ some known key symbols)
:]	58+93	Jack knife opened **If supported by ME**
:D	58+68	Multi task button (shortcut to desktop) **If supported by ME**
:L	58+76	Flash lamp button **If supported by ME**
:P	58+80	“Push to talk” button **If supported by ME**
:S	58+83	Media player button **If supported by ME**
:=	58+61	Fire (gamepad)
:<	58+60	Up left (gamepad)
:|	58+124	Up right (gamepad)
:V	58+86	Down left (gamepad)
:>	58+62	Down right (gamepad
:1	58+49	Game A (gamepad)
:2	58+50	Game B (gamepad)
:3	58+51	Game C (gamepad)
:4	58+51	Game D (gamepad)
:A	58+65	Game Internal A. **If supported by ME**
:B	58+66	Game Internal B. **If supported by ME**
:	58+92	Slide closed. **If supported by ME**
:/	58+47	Slide opened. **If supported by ME**
:X	58+88	Jog Dial up. **If supported by ME**
:Y	58+89	Jog Dial down. **If supported by ME**
:Z	58+90	Jog Dial press. **If supported by ME**

<time>:

<time>	Description
0–255	0 to 25.5 seconds (default values are manufacturer specific, but should be so long that a normal ME can handle keystrokes correctly)

<pause>:

<pause>	Description
0–255	0 to 25.5 seconds (default values are manufacturer specific, but should be so long that a normal ME can handle keystrokes correctly)

AT+CIND Indicator control (ver. 5)

Description: Check the current status of indicators and states in the phone, for example, check if a charger is connected, check the current state for a call setup, and so on. The order in which the indicators are shown is based on the information received when running the test command.
The read command returns status of ME indicators. If the ME is not currently reachable, **+CME ERROR**: <err> is returned.
The test command returns pairs, where string value <descr> is a maximum 16 character description of the indicator and compound value is the allowed values for the indicator. If ME is not currently reachable, **+CME ERROR**: <err> is returned.
The RX level is sent every 60 seconds. But it is also sent if the abs (difference between last sent level) is greater than two. What could happen is that the level changes two steps in the 59th second and an indication of this is sent. The next second the 60 second interval ends and another indication is sent.

Read command: Read current setting:
AT+CIND?

Read command response: +CIND: <ind>,<ind>,…

Test command: **AT+CIND=?** Test if the command is supported and list supported parameters.

Test command response: +CIND: (<descr>,(list of supported <ind>s)),(<descr>,(list of supported <ind>s)),…

Parameters:

<ind>:

<ind>	Description
Integer type	Value will be in range of the corresponding <descr>

<descr>:

<descr>	Description
“battchg”	Battery charge level (0–5)
“signal”	Signal quality (0–5)
“batterywarning”	Battery warning (0–1)
“chargerconnected”	Charger connected (0–1)
“service”	Service availability (0–1) – Net contact status, 1 = Net contact
“message”	Message received (0–1)
“call”	Call in progress (0–1)

<descr>	Description
“roam”	Roaming indicator (0–1) – Home net status, 0 = Home Net
“callsetup”	Bluetooth proprietary call set up status indicator. Possible values are as follows (range is 0–3): 0: Not currently in call set up 1: Incoming call process ongoing 2: Outgoing call set up is ongoing 3: Remote party being alerted in an outgoing call
“callheld”	Indicates the status of any held calls on the AG: 0 = No held calls. 1 = Call is placed on hold or active/held calls swapped 2 = Call on hold **Supported if phone uses Bluetooth handsfree profile 1.5**

AT+CMAR Master reset

Description: Requests the phone to reset user data (factory reset). The user data in the phone will be reset to default values.
If the phone is locked and this command is used, then the phone is unlocked after the master reset.
The parameter <option> is not in the 3GPP standard. This is an extension of the command for Sony Ericsson.

Execution command: **AT+CMAR=**<phone_lock_code>[,<option>]

Test command: **AT+CMAR=?** Shows if the command is supported.

Parameter:

<phone_lock_code>:

<phone_lock_code>	Description
String	Security code (Phone Lock code) must be verified before performing the master reset, see also **AT+CLCK**

<option>:

<option>	Description
0	Initiates a “Master Reset”. All settings in the phone are restored to the factory settings. All user data such as contacts in phonebook, downloaded files, WAP settings, and so on, are also erased
1	Initiates a “Reset Settings”. Settings in the phone will be restored to the factory settings. User data such as contacts in phonebook, downloaded files, WAP settings, and so on, are kept

AT+CMER Mobile equipment event reporting

Description: Enables or disables sending of unsolicited result codes from ME to TE in the case of key pressings, display changes and indicator state changes. <mode> controls the processing of unsolicited result codes specified within this command. <bfr> controls the effect on buffered codes when <mode> 1, 2 or 3 is entered. If the ME does not support a setting, +CME ERROR: <err> is returned.

Set command: **AT+CMER=**[<mode>[,<keyp>[,<disp>[,<ind>[,<bfr>]]]]]

Read command: **AT+CMER?** Displays the current <mode>, <keyp>, <disp>, <ind> and <bfr> settings.

Test command: **AT+CMER=?** Shows if the command is supported.

Test command response: +CMER: (list of supported <mode>s),(list of supported <keyp>s),(list of supported <disp>s),(list of supported <ind>s),(list of supported <bfr>s)

Parameters:

<mode>:

<mode>	Description
0	Buffer unsolicited result codes in the phone. If the phone result code buffer is full, codes can be buffered elsewhere or the oldest result codes can be removed to make room for the new result codes. **Default value**
3	Forward the unsolicited result codes directly to the terminal equipment. Phone terminal equipment link-specific in-band technique is used to embed result codes and data when phone is in online data mode

<keyp>:

<keyp>	Description
0	No keypad event reporting. **Default value**
2	Keypad event reporting using **+CKEV**:<key>,<press>. Enables keypad event reporting of all key pressing. **Note**: When this mode is enabled, corresponding result codes of all keys currently pressed should be flushed to the TA regardless of <bfr> setting

<disp>:

<disp>	Description
0	No display event reporting. **Default value**

<ind>:

<ind>	Description
0	No indicator event reporting. **Default value**

<ind>	Description
1	Indicator event reporting using **+CIEV**: <ind>,<value>. <ind> indicates the indicator order number (as specified for +CIND) and <value> is the new value of indicator. Only those indicator events, which are not caused by +CIND will be indicated by the TA to the TE

<bfr>:

<bfr>	Description
0	TA buffer of unsolicited result codes defined within this command is cleared when <mode> 1...3 is entered. **Default value**

AT*ECAM Ericsson call monitoring (ver. 2)

Description: This command activates or deactivates the call monitoring function in the ME. When this log function is activated in the ME, the ME informs about call events, such as incoming call, connected, hang up, and so on.
It is preferrable that the current status is always sent with result code ***ECAV**:<ccid>,<ccstatus>,<calltype>, <processid> , <exitcause> ,<number>,<type> when activating the log function. The purpose of this is:

- to gather relevant information for the call log in a TE.
- to make it possible for the TE to display call state information for an ongoing call.

Set command: **AT*ECAM=**<onoff>

Read command: **AT*ECAM?** Read the current status for Call Monitoring.

Read command response: *ECAM: <onoff>

Test command: **AT*ECAM=?** Shows if the command is supported.

Test command response: *ECAM: list of supported <onoff>s

Parameters:

<onoff>:

<onoff>	Description
0	The call log function is disabled (off). **Default value**
1	The call log function is enabled (on)

<ccid>:

<ccid>	Description
Integer (1–7)	A number which uniquely defines a call in the phone (= number of call control process). There cannot be two call IDs with the same number simultaneously. The maximum number of call control processes is seven, five multiparty members, one call on hold and one waiting call

<ccstatus>:

<ccstatus>	Description
0	IDLE
1	CALLING (MO)
2	CONNECTING (MO)
3	ACTIVE (connection between A and B)
4	HOLD
5	WAITING (MT)
6	ALERTING (MT)
7	BUSY

<calltype>:

<calltype>	Description
1	VOICE
2	DATA
4	FAX **Not supported**
128	VOICE2

<processid>:

<processid>	Description
Integer	Reported when returning to the IDLE state (<ccstatus> = 0). 8 = H'08 = CC (Call control) 68 =H'44 = MM (Mobile Management) 69 = H'45 = MS (Mobile Station) 122 = H'7A = RR (Radio Resources)

<exit cause>:

<exit cause>	Description
Integer	Exit cause according to GSM 04.08. Reported when returning to IDLE state (<ccstatus> = 0)

<number>:

<number>	Description
String	String type phone number of format specified by <type>. Only valid for <ccstatus> = 1 (CALLING)

<type>:

<type>	Description
Integer	Type of address octet in integer format (refer to GSM 04.08 subclause 10.5.4.7). **Default value** is 145 when dialling string includes international access code character “+”, otherwise 129. Only valid for: • <ccstatus> = 1 (CALLING) • <ccstatus> = 5 (WAITING) • <ccstatus> = 6 (ALERTING)

AT+CLAN Language

Description: Sets the language in the phone. If the language has been set to “AUTO”, the read command returns the current language set from the SIM card. Hence, the “AUTO” code is never returned by the read command.

Set command: **AT+CLAN=**<code>

Read command: **AT+CLAN?** Displays the current language setting.

Test command: **AT+CLAN=?** Shows if the command is supported.

Test command response: +CLAN: (list of supported <code>s)

Parameter:

<code>: Language codes defined in ISO 639. Consists of two characters, for example, “sv”, “en”, and so on.

<code>	Description
“AUTO”	Read the language code from the SIM card. “AUTO” is never returned by the read command
	Miscellaneous language codes

AT*EJAVA Ericsson Java application function

Description: Requests the MT to perform a Java application function specified by <application> and <action>.

Note:

- There is no guarantee that the application will execute. The command will return OK if the command, including parameters, is supported. This also means that there is no correlation between the OK response and the time the application function is performed by the MT.
- If the AT*EJAVA command is issued and the <application> parameter references an application that is already running, a second instance of this application will **not** be started. The application already running should however perform the action indicated with the <action> parameter.

Set command: **AT*EJAVA=**<action>[,<application>]

Set command response:

If <action>=1 (list applications):
*EJAVA:[<application_name1>,<object_id1>]
*EJAVA:[,<application_name2>,<object_id2>...]

Test command: **AT*EJAVA=?** Shows if the command is supported.

Test command response: *EJAVA: List of supported <action>s[, (list of <application_id>s)]

Parameters:

<action>:

<action>	**Description**
0	Run a java application. The search path to the application must be provided in <application>. **Not supported (obsolete)**
1	List installed java applications. No value on <application> needed
2	Delete a java application. The <suite_id> of the application must be provided in <application>
3	Install a java application. The search path to the application must be provided in <application>. **Note**: This parameter can use two application variables for the support of JAD and JAR installation. For example, at*ejava=3,"/tpa/user/other/XYZ.jad","/tpa/user/other/XYZ.jar"
4	Run an installed java application. The <application_id> of the application must be provided in <application>
5	List all running JAVA applications in ME
6	Terminate a MIDlet. For example, "at*ejava=6,XYZ-APP-SUITE-ID". The application instance with ID XYZ-APP-SUITE-ID should be stopped

<application>: **Note**: Should not be given for <action> = 1 (list applications).

<application>	Description
String	For <action> = 0, 3: The search path to the application to be run/ installed
Integer	For <action> = 2: The <suite_id> of the application
Integer	For <action> = 4: The <application_id> of the application

<application_name>:

<application_name>	Description
String	The name of a java application located in the specified directory

<vendor>:

<vendor>	Description
String	The name of the vendor

<version>:

<version>	Description
String	The actual version of the application

<application_id>:

<application_id>	Description
Integer	The application ID for the application

<suite_id>:

<suite_id>	Description
Integer	The unique ID for the suite

AT+CSIL Silence Command

Description: Orders the phone to be in silent mode or orders the phone to leave the silent mode. When the phone is in silent mode, all sounds from the phone must be prevented. An icon will show the user that silent mode is active. If no parameter is given to the SET command it will use <mode> = 0 as parameter.

Execution command: **AT+CSIL=**[<mode>]

Read command: **AT+CSIL?** Displays the current <mode> setting.

Read command response: +CSIL: <mode>

Test command: **AT+CSIL=?** Shows if the command is supported.

Test command response: +CSIL: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	Silent mode off. **Default value**
1	Silent mode on

AT*ESKL Key-lock mode

Description: Sets the key lock mode in the phone.

Set command: **AT*ESKL=**<mode>

Read command: **AT*ESKL?** Displays the current <mode> setting.

Test command: **AT*ESKL=?** Shows if the command is supported.

Test command response: *ESKL: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	MANUAL. The user has to manually lock the keyboard. **Default value**
1	AUTOMATIC. The phone will, after a time delay, automatically lock the keyboard

AT*ESKS Key sound

Description: Sets the key sound in the phone.

Set command: **AT*ESKS=**<mode>

Read command: **AT*ESKS?** Displays the current <mode> setting.

Test command: **AT*ESKS=?** Shows if the command is supported.

Test command response: *ESKS: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	SILENT – no sound when a key is pressed. **Default value**
1	CLICK – short click when a key is pressed.
2	TONE – a continuous tone when a key is pressed.

AT*EAPP Application function (ver. 5)

Description:

Requests the MT to perform an application function specified by <app> and <subfunc>. The <subfunc> parameter specifies which function within the specified application to call. The <text> parameters can be used to pass data to the application. The use of the <text> parameters are specified with each subfunction.
Note: There is no guarantee that the application will execute. The command will return OK if the command, including parameters, is supported. There is no correlation between the OK response and the time the application function is performed by the MT.
Note: If the *EAPP command is issued and the <app> parameter references an application that is already running, a second instance of this application will not be started. The application already running should however perform the subfunction indicated with the <subfunc> parameter.
Note: The syntax for **MMS** (<app>=6) is as follows:
AT*EAPP=6,<subfunc1>[,<text1>[,<subfunc2>,<text2>[,<subfunc3>,<text3>…]]]
It is thus possible to add different attachments (image, video clips, and so on) to a message.

Example: Sending a message with text, “Look at my new car!” and an image located in “//filesystem/pictures/mycar.jpg”:
AT*EAPP=6,0,“Look at my new car!”,4,“//filesystem/pictures/mycar.jpg”

The syntax for **email** (<app>=2) is the same as the syntax for MMS:
AT*EAPP=2,0,“Look at my new car!”,4,“//filesystem/pictures/mycar.jpg”

Test command shows which applications and subfunctions are supported by the MT.
AT*EAPP=?
*EAPP: 0,(0-5)
*EAPP: 1,(1,3,4-5)
*EAPP: 3,(0,4)
*EAPP: 4,(0-2)

Set command: **AT*EAPP=**<app>[,<subfunc>[,<text1>[,<text2>]]]

Test command: **AT*EAPP=?** Shows if the command is supported.

Test command response:

*EAPP: <app>,(list of supported <subfunc>s)[**<CR><LF>**
<app>,(list of supported <subfunc>s)[...]]

Parameters:

<app>:

<app>	**Description**
0	Message application
1	Phonebook application
2	Email application
3	WAP application
4	Calendar application
5	**Not supported**
6	Multimedia messaging application

<app>	Description
7	Notes application
8	Image browser
9	Sound browser
10	Camera application
11	Media player application

<subfunc>: Application specific information, see tables below.

<subfunc>, <app=0>	Description
0	Send new SMS message. Pre-entered message text can be provided in <text1>. **Default value**
1	Inbox
2	Unsent
3	Add new template. Pre-entered message text can be provided in <text1>
4	Sent items.
5	Send new message to specific phonebook entry. Pre-entered message text can be provided in <text1>. The name of the phonebook entry to send message to will be provided in <text2>
6	Send new message and include formatting characters and PB entry for Email. **Note**: It is up to the MT to insert the formatting characters and the PB entry
7	Send new message and include formatting characters for www. **Note**: It is up to the MT to insert the formatting characters and the PB entry
8	Add picture
9	Add melody

<subfunc>, <app=1>	Description
0	Add new number. Pre-entered number can be provided in <text1>. **Default value**
1	Find and call. Pre-entered name can be provided in <text1>. **Note:** If a name is provided, the search is started without user interaction
2	Find and edit. Pre-entered name can be provided in <text1>. **Note:** If a name is provided, the search is started without user interaction
3	Add new voice label

<subfunc>, <app=1>	Description
4	Add new group. Pre-entered name can be provided in <text1>
5	Add new email address. Pre-entered address can be provided in <text1>

<subfunc>, <app=2>	Description
0	Send new message. Pre-entered message (body) text can be provided in <text1>. **Default value**
1	Inbox (read new mail): • <text1>='Y' => check for new mail • <text1>='N' => do not check for new mail
2	Outbox
3	Draft
4	Add attachment – image. Search path in the file system to the image will be included in <text>
5	Add attachment – voice or sound clip. Search path in the file system to the audio file will be included in <text>
6	Add attachment – motion. Search path in the file system to the video clip will be included in <text>
7	Add attachment – vCard. Search path in the file system to the vCard object will be included in <text>
8	Add attachment – vCalender. Search path in the file system to the vCalender object will be included in <text>
9	Add attachment – URL

<subfunc>, <app=3>	Description
0	Enter address (URL). Pre-entered URL can be provided in <text1>. **Default value**
1	Go to address. Pre-entered URL **must be** provided in <text1>. The connection is initiated without user interaction
2	Add new bookmark
3	Edit homepage
4	Go to homepage
5	Go to last visited page

<subfunc>, <app=4>	Description
0	Add new appointment. **Default value**
1	Add new ToDo
2	ToDo view
3	Today view
4	Week view
5	Month view

<subfunc>, <app=6>	Description
0	Send new text message. Text will be included in <text1>
1	Inbox (read new mail). <text1>=”Y” => Check for new mail. <text1>=”N” => Do not check for new mail
2	Outbox
3	Draft
4	Add attachment – image. Search path in the file system to the image will be included in <text>
5	Add attachment – voice or sound clip. Search path in the file system to the audio file will be included in <text>
6	Add attachment – motion. Search path in the file system to the video clip will be included in <text>
7	Add attachment – vCard. Search path in the file system to the vCard object will be included in <text>
8	Add attachment – vCalender. Search path in the file system to the vCalender object will be included in <text>
9	Add attachment – vNote. Search path in the file system to the vNote object will be included in <text>
10	Add attachment – Theme

<subfunc>, <app=7>	Description
0	Create new note. Pre-entered message text can be provided in <text1>. **Default value**
1	Display list of notes. If only notes of a certain class should be shown, its name can be provided in <text1>

<subfunc>, <app=8>	Description
0	Display an image in fullscreen mode. This is done by choosing a directory that contains only one picture. The directory is specified in <text1>
1	Display thumbnail images. The command shows thumbnail images of all pictures in the directory specified by <text1>
2	Delete one or several image(s). The image name is specified in <text1>. **Note**: Request from image handler, not image browser
255	Close image browser

<subfunc>, <app=9>	Description
0	Play a certain sound The search path to the sound file will be provided in <text1>

<subfunc>, <app=9>	Description
255	Close sound browser

<subfunc>, <app=10>	Description
0	Start the camera application

<subfunc>, <app=11>	Description
0	Start the media player application and play the file located in the search path provided in <text1>

Example:

```
AT*EAPP=?
*EAPP: 0,(0-7)
*EAPP: 1,(0-5)
*EAPP: 2,(0-4)
*EAPP: 3,(0-4)
*EAPP: 4,(0-5)
*EAPP: 7,(0-3)
*EAPP: 8,(0-2,255)
OK
```

AT+CMEC Mobile equipment control mode

Description: Selects which equipment operates the ME keypad, writes to the the ME display and sets ME indicators. If operation mode is not allowed by the phone, **+CME ERROR**: <err> is returned.

Execution command: **AT+CMEC=**[<keyp>[,<disp>[,<ind>]]]

Read command: **AT*CMEC?** Reads the current settings. +CMEC: <keyp>,<disp>,<ind>

Test command: **AT+CMEC=?** Tests if the commands is supported.

Test command response: +CMEC: (list of supported <keyp>s),(list of supported <disp>s),(list of supported <ind>s)

Parameters:

<keyp>:

<keyp>	Description
0	ME can be operated only through its keypad (execution command of AT+CKPD cannot be used)
1	ME keypad can be operated only from TE (with command AT+CKPD)
2	ME keypad can be operated from both ME keypad and TE

<disp>:

<disp>	Description
0	only ME can write to its display (command AT+CDIS can only be used to read the display)

<disp>	Description
1	only TE can write to ME display (with command AT+CDIS)
2	ME display can be written by both ME and TE

<ind>:

<ind>	Description
0	only ME can set the status of its indicators (command AT+CIND can only be used to read the indicators)
1	only TE can set the status of ME indicators (with command AT+CIND)
2	ME indicators can be set by both ME and TE

AT+CRSM Restricted SIM access

Description: By using this command instead of Generic SIM Access, +CSIM, TE application has easier but more limited access to the SIM database. Set command transmits the SIM <command> and its required parameters to the ME. ME handles all SIM – ME interface locking and file selection routines internally. In response to the command, ME sends the actual SIM information parameters and response data. ME error result code **+CME ERROR** may be returned when the command cannot be passed to the SIM, but failure in the execution of the command in the SIM is reported in <sw1> and <sw2> parameters.

Execution command: **AT+CRSM**=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>]]]

Response: +CRSM: <sw1>,<sw2>[,<response>]

Test command: **AT+CRSM=?** Test if the command is supported

Parameters:

<command>:

<command>	Description
176	READ BINARY
178	READ RECORD
192	GET RESPONSE
214	UPDATE BINARY
220	UPDATE RECORD
242	STATUS. **Not supported**

<fileid>: **Note**: The range of valid file identifiers depends on the actual SIM and is defined in 3GPP 51.011. Optional files may not be present at all.

<fileid>	Description
Integer	Parameters passed on by the ME to the SIM. These parameters are mandatory for every command, except GET RESPONSE and STATUS. The values are described in 3GPP 51.011

<P1>, <P2>, <P3>:

<P1>, <P2>, <P3>	Description
Integer	Parameters passed on by the ME to the SIM. These parameters are mandatory for every command, except GET RESPONSE and STATUS. The values are described in 3GPP 51.011

<data>:

<data>	Description
String	Information which is written to the SIM

<sw1>,<sw2>:

<sw1>,<sw2>	Description
Integer	Information from the SIM about the execution of the actual command. These parameters are delivered to the TE in both cases, on successful or failed execution of the command.

<response>:

<response>	Description
String	Response of a successful completion of the command previously issued. GET RESPONSE return data, which gives information about the current elementary datafield. This information includes the type of file and its size (refer 3GPP 51.011. After READ BINARY or READ RECORD command the requested data will be returned. <response> is not returned after a successful UPDATE BINARY or UPDATE RECORD command.

AT*EKSE Ericsson keystroke send

Description: Sends a keystroke identifier to the ME. The ME makes a context sensitive interpretation of the keystroke based on the state of the MMI (for instance, SMS input mode, Standby, Charge-only Mode).
Note: If the value of the keystroke identifier does not correspond to a supported Unicode value, the correct character is not given. See also www.unicode.org.

Execution command: **AT*EKSE=**<key>

Test command: **AT*EKSE=?** Shows if the command is supported.

Test command response: *EKSE: (list of supported <key> range)

Parameters:

<key>:

<key>	**Description**
0–65535	Keystroke identifier given in Unicode

AT+CRSL Ringer sound level (ver. 2)

Description: Selects the incoming call ringer sound level of the ME. If <level> is set to 255, the ringer level will be increasing.

Line 1 is default for <calltype> if the parameter is not given.

All <calltype>s are set to <level>, even if only one unique <calltype> is set in the command.

Execution command: **AT+CRSL=**<level>[,<calltype>]

Read command: **AT+CRSL?** Displays current settings.

Read command response: +CRSL: <level1>[,<calltype1> [**<CR><LF>**…
+CRSL: <level n>[, <calltype n>]]]

Test command: **AT+CRSL=?** Shows if the command is supported.

Test command response: +CRSL: (list of supported <level>s)[,(list of supported <calltype>s)]

Parameters:

<level>:

<level>	**Description**
0	Ringer off. **Default value**
1	Ringer level 1
2	Ringer level 2
3	Ringer level 3
4	Ringer level 4

<level>	Description
5	Ringer level 5
6	Ringer level 6
7	Ringer level 7
8	Ringer level 8
255	Increasing ringer level

<calltype>: There is no specific setting for the calltype parameters in the phone. The three supported calltypes are pointing at the the same audio setting in the phone. For example, if Line 1 is changed, also Line 2 and Data will change.

<calltype>	Description
1	Line 1. **Default value**
2	Line 2
4	Data

AT+CLVL Loudspeaker volume level

Description: Selects the volume of the internal loudspeaker (call volume) of the ME.

Execution command: **AT+CLVL=**<level>

Read command: **AT+CLVL?** Displays current settings.

Read command response: +CLVL: <level>

Test command: **AT+CLVL=?** Shows if the command is supported.

Test command response: +CLVL: (list of supported <level>s)

Parameter:

<level>:

<level>	Description
0	Loudspeaker off. **Default value**
1	Loudspeaker level 1
2	Loudspeaker level 2
3	Loudspeaker level 3
4	Loudspeaker level 4
5	Loudspeaker level 5
6	Loudspeaker level 6
7	Loudspeaker level 7
8	Loudspeaker level 8

AT+CMUT Mute control

Description: Enables and disables the uplink voice muting during a voice call.

Execution command: **AT+CMUT=**<n>

Read command: **AT+CMUT?** Displays current settings.

Read command response: +CMUT: <n>

Test command: **AT+CMUT=?** Shows if the command is supported.

Test command response: +CMUT: (list of supported <n>s)

Parameter:

<n>:

<n>	**Description**
0	Mute off. **Default value**
1	Mute on

AT*EMEM Ericsson memory management

Description: Reports the file system memory usage.

Action command: **AT*EMEM**

Action command response: *EMEM: <free_mem>,<tot_mem>,<image_mem>,<sound_mem>,<theme_mem>

Test command: **AT*EMEM=?** Shows if the command is supported.

Parameters:

<free_mem>:

<free_mem>	**Description**
Integer	Remaining free memory (in bytes) in the file system

<tot_mem>:

<tot_mem>	**Description**
Integer	Total memory size (in bytes) of the file system

<image_mem>:

<image_mem>	**Description**
Integer	Number of bytes (in the file system) used by images

<sound_mem>:

<sound_mem>	**Description**
Integer	Number of bytes (in the file system) used by sounds

<theme_mem>:

<theme_mem>	**Description**
Integer	Number of bytes (in the file system) used by themes

AT+CRMP Ring melody playback (ver. 2)

Description: Causes the phone to playback a specific ring type.
The playback is **not** stopped when keys are pressed on the phone.
The playback is stopped when the issuing accessory is disconnected from the phone.

Execution command: **AT+CRMP=**<call_type>[,<volume> [,<type>]]

Test command: **AT+CRMP=?** Shows if the command is supported.

Test command response: +CRMP: (list of supported <call_type>s),(list of supported <volume>s),(list of supported <type>s)

Parameters:

<call_type>:

<call_type>	**Description**
0	Stop playing sound
1	Line 1. **Default value**
2	Line 2
3	Fax
4	Data
5	Alarm

<volume>:

<volume>	**Description**
0–8	Volume level (0 = Off, 8 = maximal volume). **Default value = 3**
255	Step. **Not supported**

<type>:

<type>	**Description**
0	Manufacturer defined
1	User defined. **Default value**

AT*EKEY Keypad/joystick control (ver. 2)

Description:

Emulates ME keypad by giving each keystroke as a character <key>. If emulation fails in an ME, an error, **+CME ERROR**: <err> is returned. This command should be accepted (OK returned) before actually starting to press the keys. Thus unsolicited result codes of key pressings and display events can be returned (see **AT+CMER**). It will also be possible to receive unsolicited event for CKEV when an EKEY is sent to the ME. For example, this makes it possible for a connected Bluetooth device to be controlled by the EKEY command.
The physical keypad always has higher priority than emulation of key-strokes via AT*EKEY. That is, if the physical keypad is operated during execution of a series of keystrokes generated by AT*EKEY the emulated keypad operation is terminated immediately.
Note: To use some of the keys, a specific character set has to be set via command AT+CSCS=<chset>. For example, the characters "[" and "]" used to emulate the left and right selection keys on the phone are not included in the default GSM character table. To handle this issue, the 8859-1 character table can be set by first sending AT+CSCS="8859-1" to the phone.
Note: This command is an upgrade of AT+CKPD ver. 3 supporting the same keys. The parameters <time> and <pause> have been removed and two new parameters have been added. The <keyfunc> parameter indicating whether the key was pressed or released and the <nr_of_keys> parameter telling how many keys that will be sent in the command. Also the returned result for EKEY=? is changed.

Execution command:

AT*EKEY=<nr_of_keys>,<key>,<keyfunc>[,<key>,<keyfunc>]…

Test command:

AT*EKEY=?

Test command response

EKEY: supported <nr_of_keys>, (list of supported <key>), (list of supported <keyfunc>)

Parameter:

<nr_of_keys>: Integer type. The number of <key> characters and corresponding <keyfunc> parameters that will be sent. Maximum value=20.

<key> Character representing keys as listed in the following table. Colon character (IRA 58) followed by one character can be used to indicate a manufacturer specific key.

Char	IRA (dec)	Comment plus some known key symbols
#	35	Hash (number sign)
*	42	Star (*)
0–9	48–57	Number keys
:	58	Escape character for manufacturer specific keys
<	60	Left joystick direction
>	62	Right joystick direction
C/c	67/99	Clear display (C/CLR)
D/d	68/100	Volume down

Char	IRA (dec)	Comment plus some known key symbols
F/f	70/102	Function (FCN) – option key
G/g	71/103	Voice note
P/p	80/112	Power (PWR)
S/s	83/115	Connection start (SEND)
U/u	85/117	Volume up
V/v	86/118	Down joystick direction
[	91	Soft key 1
]	93	Soft key 2
^	94	Up joystick direction
H/h	200	Button pushed on the MC link (BT) headset
:R	58 + 82	Back
:C	58 + 67	Camera
:G	58 + 71	Go music button. **If supported by ME**
:O	58 + 79	Operator
:J	58 + 74	Joystick button pressed
:<	58 + 60	Left Up joystick direction
:I	58 +73	Right Up joystick direction
:V	58 + 86	Left Down joystick direction
:>	58 + 62	Right Down joystick direction
:=	58 + 61	Abstract game event FIRE
:1	58 + 1	Abstract game event GAME_A
:2	58 + 2	Abstract game event GAME_B
:3	58 + 3	Abstract game event GAME_C
:4	58 + 4	Abstract game event GAME_D
:M	58 + 77	Video call
:F	58 + 70	Flash button
:(	58 + 40	Flip closed
:)	58 + 41	Flip opened
:{	58 + 123	Camera lens cover closed
:}	58 + 125	Camera lens cover opened
:[	58 + 93	Jack knife closed
:]	58 + 95	Jack knife opened
:D	58 + 63	Multi task button (shortcut to desktop)
:L	58 + 76	Flash light button
:P	58 + 80	“Push to talk” button
:S	58 + 83	Media player button

Char	IRA (dec)	Comment plus some known key symbols
Definition of an “abstract game event”: An event that is not absolutely mapped to one specific game event and not to a specific input device on the phone. For example, the fire button on the game controller will be mapped to the fire action in both Mophun games and Java games, even if fire action for Java is mapped to the left soft key and for Mophun to the 5 key		

<keyfunc>: Parameter used to define whether the key was pressed or released.

<keyfunc>	Action
0	Key pressed
1	Key released
2	Key pressed and released

Example: Joystick is moved in the left direction and at the same time the fire key is released:

```
AT*EKEY=2,”<”,0,”:=”,1
OK
```

The command will send a dispatch each time a key is either pressed or released.

AT*ECDF Ericsson change dedicated file

Description: Selects a Dedicated File (DF) on the (U)SIM, so Elementary Files (EFs) that are to be accessed by command **+CRSM** can be uniquely identified. The **+CRSM** command only takes EF as parameter and, since EFs in different DFs can have the same file ID, certain files are not accessible with **+CRSM** alone.

A DF can be seen as a “directory” and *ECDF could be regarded as a “change directory” command.

Note: *ECDF has impact on the standard behaviour of **+CRSM**:
AT+CRSM does not specify how files in different DFs will be accessed, but the introduction of ***ECDF** imposes the restriction that only EFs in the currently selected DF may be accessed with **+CRSM**.

Even if an EF can be uniquely identified throughout all DFs, it is required that ***ECDF** has selected the correct DF for the EF to be accessed.

Execution command: *Change DF:*
AT*ECDF=[<DF file ID>]

Read command: **AT*ECDF?**
Read the current setting. *ECDF: <DF file ID>

Test command: **AT*ECDF=?** Test if command is supported and show supported parameters

Test command response *ECDF: (list of supported <DF file ID>s)

Parameter:

<DF File ID>:

<DF File ID>	Description
32528	0x7F10 : DF Telecom
32544	0x7F20 : DF GSM. **Default DF**
32579	0x7F43 DF VI: VIAG HomeZone specific DF

AT*STKC SIM application toolkit configuration

Description: Indicates to the SIM which SAT features are supported by AT commands (ref. 3GPP TS 11.14) and enables unsolicited resultcodes ***STKI** and ***STKN** sent from ME to TE.

Execution command: *Configure SIM application toolkit:*
AT*STKC=<n>,<stkPrfl>

Read command: **AT*STKC?**
Show current SIM application toolkit configuration. *STKC: <n>,<stkPrfl>

Test command: **AT*STKC=?** Test if command is supported and show supported parameters

Test command response *STKC: (list of supported <n>s, <prfLen>

Parameter:

<n>:

<n>	Description
0	Disable SAT UR codes
1	Enable SAT UR codes

<stkPrfl>:

<stkPrfl>	Description
String type	SIM application toolkit profile in hexadecimal format starting with first byte of the profile. See 3GPP TS 11.14 for details

AT*STKE SIM application toolkit envelope command send

Description: Sends SIM application toolkit envelope commands to the SIM (refer to 3GPP TS 11.14 for details about the envelope commands).

Execution command: *Send STK envelope command:*
AT*STKE=<stkCmd>

Execution command response: *STKE: <stkRsp>

Test command: **AT*STKE=?** Test if command is supported

Parameter:

<stkCmd>:

<stkCmd>	Description
String type	SIM application toolkit command (hexadecimal format) starting with command tag. See 3GPP TS 11.14 for details

<stkRsp>:

<stkRsp>	Description
String type	SIM application toolkit response (hexadecimal format) starting with first byte of response data

AT*STKR SIM application toolkit command response

Description: This command is used to send SIM application toolkit command responses to the SIM (refer to 3GPP TS 11.14 for details about the response).

Execution command: *Send STK command respnse:*
AT*STKR=<stkRsp>

Test command: **AT*STKR=?** Test if command is supported

Parameter:

<stkRsp>:

<stkRsp>	Description
String type	SIM application toolkit response (hexadecimal format) starting with first byte of response data. See 3GPP TS 11.14 for details

Unsolicited result codes

+CKEV Keypad event

Description: Keypad event reporting is enabled by the **AT+CMER** command and indicates key press/release.

Unsolicited result code: **+CKEV:** <keys>,<press>

Parameters:

<keys>: See **AT+CKPD**.

<press>:

<press>	Description
0	Key released
1	Key pressed

+CIEV Indicator event

Description: Indicates changes in indicator levels. Enabled with **AT+CMER**.

Unsolicited result code: **+CKEV:** <ind>,<value>

Parameters:

<ind>: Indicates the indicator order number (as specified for **AT+CIND**)

<ind>	Description
1	Battery charge level indicator
2	Signal quality indicator
3	Battery warning indicator
4	Charger connected indicator
5	Service availability indicator
6	Sounder activity indicator
7	Message received indicator
8	Call-in-progress indicator
9	Transmit activated by voice activity indicator
10	Roaming indicator
11	Short message memory storage indicator in the SMS

<value>: Integer. New value of the specific indicator.

*ECAV Call monitoring event

Description: Reports changes in call state for a certain call, indicated by <coid>. Enabled by **AT*ECAM**.

Unsolicited result code: ***ECAV:** <ccid>,<ccstatus><calltype>[,<processid>][,<exit_cause>][,<number>,<type>]

Parameters:

<ccid>:

<ccid>	Description
1–7	A number that uniquely identifies a call in the phone. The maximum number of call control processes is seven, five multiparty members, one call on hold and one waiting call)

<ccstatus>:

<ccstatus>	Description
0	IDLE
1	CALLING
2	CONNECTING
3	ACTIVE
4	HOLD
5	WAITING
6	ALERTING
7	BUSY

<calltype>:

<calltype>	Description
1	VOICE
2	DATA
4	FAX
128	VOICE2

<processid>: Integer. Reported when returning to IDLE state (<ccstatus>=0)

<processid>	Description
8=H´08	CC (Call Control)
68=H´44	MM (Mobile Management)
69=H´45	MS (Mobile Station)
122=H´7A	RR (Radio Resources)

<exit_cause>: Integer. Reported when returning to IDLE state (<ccstatus>=0).

<number>: Integer string. Phone number. Format specified by <type>. Only valid for <ccstatus>=1 (CALLING).

<type>: Type of address octet. Only valid for <ccstatus>=1 (CALLING).

<type>	Description
145	**Default value** when a dialling string includes the international access code character “+”
129	**Default value** when a dialling string does not include the international access code character “+”.

*STKI SIM application toolkit command sent from SIM

Description: Displays SIM AT command received from SIM. Enabled by **AT*STKC**.

Unsolicited result code: ***STKI:** <stkCmd>

Parameters:

<stkCmd>:

<stkCmd>	Description
String type	SIM application toolkit command (hexadecimal format) starting with command tag. See 3GPP TS 11.14 for details

*STKN SIM application toolkit notify

Description: Indicates commands and responses handled by ME. Enabled by **AT*STKC**.

Unsolicited result code: ***STKN:** <stkNtfy>

Parameters:

<stkNtfy>:

<stkNtfy>	Description
String type	SIM application toolkit command or resonse (hexadecimal format) starting with command tag or first byte of response. See 3GPP TS 11.14 for details

Use scenarios

Mobile equipment control mode and event reporting

This scenario operates the keypad and reads the keypad and indicator status.

AT command	Response	Comment
AT+CKPD="046193000S",5,1		Dial number 046193000 by emulating a sequence of key presses. Each key is pressed for half a second and the pause between the keystrokes is 0.1 seconds
	OK	
AT+CKPD="E",5		End connection by emulating a stroke of the "on hook" button for half a second
	OK	
AT+CIND?		Query the current indicator values
	+CIND: 3,4,0,0,1,0,0,0,0,0,0 OK	
AT+CMER=,2,,1,		Request unsolicited result codes for keypad and indicator events
	OK	
	+CKEV: 49,1	Number key 1 is pressed
	+CKEV: 49,0	Number key 1 is released

AT command	Response	Comment
	+CIEV: 2,5	Signal strength indicator changes its state to “5”
AT+CMER=,0,,0,		Disable unsolicited result codes for keypad and indicator events
	OK	

Call monitoring

This scenario shows how call monitoring is activated and how call events are received.

AT command	Response	Comment
AT*ECAM=1		Enable the call log function
	*ECAM: 1,0,1 OK	IDLE
ATD046193000;		Dial number
	OK	
	*ECAV: 1,1,1,,,046193000,129	CALLING, VOICE1
	*ECAV: 1,2,1,,	CONNECTING, VOICE1
	*ECAV: 1,3,1,,	ACTIVE CALL, VOICE1
AT+CHLD		Put call on hold
	OK	
	*ECAV: 1,4,1,,	HOLD, VOICE1
AT+CHLD=2		Retrieve held call
	OK	
	*ECAV: 1,3,1	ACTIVE CALL, VOICE1
ATH		Hang up
	OK	
	*ECAV: 1,0,1,8,16	IDLE. Call Control exit cause 16 (normal clearing)
	RING	Incoming call
	*ECAV: 1,6,128,,	ALTERING, VOICE2
	RING	
	RING	

MMI configuration

This scenario shows various settings of the MMI.

AT command	Response	Comment
AT*ESKS=1		Set key pressed sound to CLICK
	OK	
AT*ESKL=1		Set key lock mode to AUTOMATIC

AT command	Response	Comment
	OK	The phone keyboard will, after a time delay, be locked

Ensemble S10: GSM mobile equipment error control

Commands

AT+CMEE Report mobile equipment error

Description: Disables or enables the use of result code **+CME ERROR** as an indication of an error relating to the functionality of the ME. When enabled, ME related errors cause +CME ERROR final result code instead of the regular ERROR final result code. ERROR is returned only when the error is related to syntax, invalid parameters or TA functionality.

Set command: **AT+CMEE=**[<n>]

Read command: **AT+CMEE?** Displays the current <n> setting.

Test command: **AT+CMEE=?** Shows if the command is supported.

Test command response: +CMEE: (list of supported <n>s)

Parameter:

<n>:

<n>	Description
0	Disable +CME ERROR result code. Use ERROR instead. **Default value**
1	Enable +CME ERROR result code and use numeric <err> values
2	Enable +CME ERROR result code and use verbose <err> values

Ensemble S11: SMS and PDU mode

Commands

AT+CSMS Select message service (ver.2)

Description: Selects messaging service <service>. It returns the types of messages supported by the ME – <mt> for mobile terminated messages, <mo> for mobile originated messages and <bm> for broadcast type messages. If the chosen service is not supported by the ME, but is supported by the TA, final result code, **+CME ERROR** is returned.
The command is aborted when a break command is received by the MS. A break command is sent by setting the DTMS to low, which is obtained when the accessory is detached.
It is possible to use ATZ and AT&F to set all parameters to their factory defaults as specified by the manufacturer.

Set command: **AT+CSMS=**<service>

Response: +CSMS: <mt>,<mo>,<bm>

Read command: **AT+CSMS?** Displays the current <service>, <mt>, <mo> and <bm> settings.

Test command: **AT+CSMS=?** Shows if the command is supported.

Test command response: +CSMS: (list of supported <service>s)

Parameters:

<service>:

<service>	Description
0	GSM 03.40 and 03.41. The syntax of SMS AT commands is compatible with GSM 07.05 Phase 2 version 4.7.0. Phase 2+ features which do not require new command syntax may be supported, for example, correct routing of messages with new Phase 2+ data coding schemes. **Default value**
1	GSM 03.40 and 03.41 (The syntax of SMS AT commands is compatible with GSM 07.05 Phase 2+ version). **Not supported**
2–12	Reserved
128	Manufacturer specific

<mt>:

<mt>	Description
0	Mobile terminated messages not supported
1	Mobile terminated messages supported

<mo>:

<mo>	Description
0	Mobile originated messages not supported
1	Mobile originated messages supported

<bm>:

<bm>	Description
0	Broadcast messages not supported
1	Broadcast messages supported

AT+CPMS Preferred message storage (ver. 4)

Description: Set command selects memory storage <mem1>, <mem2> and <mem3> to be used for reading, writing, and so on. If chosen storage is not appropriate for the ME (but is supported by the TA), final result code +CMS ERROR: <err> is returned.

Set command: **AT+CPMS=**<mem1>[,<mem2>[,<mem3>]]

Set command response: +CPMS: <used1>,<total1>,<used2>,<total2>,<used3>,<total3>

Read command: **AT+CPMS?** Displays the current <mem1>,<used1>,<total1>, <mem2>,<used2>,<total2>,<mem3>,<used3>,<total3> values.

Test command: **AT+CPMS=?** Shows if the command is supported.

Test command response: +CPMS: (list of supported <mem1>s),(list of supported <mem2>s),(list of supported <mem3>s)

Parameters:

<mem1>:

<mem1>	Description
string type	Memory from which messages are read and deleted (see commands List Messages **AT+CMGL**, Read Messages **AT+CMGR** and Delete Messages **AT+CMGD**)
"ME"	Phone message storage
"SM"	SIM message storage

<mem2>:

<mem2>	Description
string type	Memory to which writing and sending operations are made (see commands Send Message from Storage **AT+CMSS** and Write Message to Memory **AT+CMGW**)
"ME"	Phone message storage
"SM"	SIM message storage

<mem3>:

<mem3>	Description
string type	Memory to which received SMs are preferred to be stored (unless forwarded directly to terminal equipment). Received CBMs (Cell Broadcast Messages) are always stored in “BM” (Broadcast Message storage) or some manufacturer specific storage, unless directly forwarded to terminal equipment
“ME”	Phone message storage.
“SM”	SIM message storage

<used1>,<used2>,<used3>:

<used1>,<used2>, <used3>	Description
Integer type	Total number of messages currently in <mem1>, <mem2> and <mem3> respectively

<total1>,<total2>,<total3>:

<total1>,<total2>, <total3>	Description
Integer type	Total number of messages that can be stored in <mem1>, <mem2> and <mem3> respectively

AT+CMGF Message format (ver. 1)

Description: Tells the TA which input and output format of messages to use. <mode> indicates the format of messages used with send, list, read and write commands and unsolicited result codes resulting from received messages. Mode can be either PDU mode (entire TP data units used) or text mode (headers and body of the messages given as separate parameters).

Set command: **AT+CMGF=**<mode>

Read command: **AT+CMGF?** Displays the current <mode> setting.

Read command response: +CMGF: <mode>

Test command: **AT+CMGF=?** Shows if the command is supported.

Test command response: +CMGF: (list of supported <mode>s)

Parameter:

<mode>:

<mode>	Description
0	PDU mode

AT+CSCA Service centre address (ver. 2)

Description: Updates the SMCS address, through which mobile originated SMS are transmitted. In text mode, the setting is used by send (**AT+CMGS**) and write (**AT+CMGW**) commands. In PDU mode, the setting is used by the same commands, but only when the length of the SMCS address (coded into <pdu> parameter) equals zero.
Note: A + sign in front of the number in <sca> has precedence over the <tosca> parameter, so that a number starting with "+" will always be treated as an international number.

Set command: **AT+CSCA=**<sca>[,<tosca>]

Read command: **AT+CSCA?** Displays the current <sca> and <tosca> settings.

Test command: **AT+CSCA=?** Shows if the command is supported.

Parameters:

<sca>: String. GSM 04.11 RP SC address-value field in string format. BCD numbers (or GSM default alphabet characters) are converted to characters in the currently selected TE character set.

<tosca>: Integer. GSM 04.11 RP SC type-of-address octet in integer format.

<tosca>	Description
129	ISDN/telephony numbering plan, national/ international unknown. **Default value if "+" is not in <sca>**
145	ISDN/telephony numbering plan, international number. **Default value if "+" is in <sca>**
161	ISDN/telephony numbering plan, national number
128–255	Other values, see GSM 04.08 section 10.5.4.7

AT+CSAS Save settings

Description: Saves the active message service settings to a non-volatile memory. A phone can contain several profiles of settings. The settings specified in **AT+CSCA** are saved. Certain settings, for example, SIM SMS parameters, may not be supported by the storage and can therefore not be saved.

Execution command: **AT+CSAS**[=<profile>]

Test command: **AT+CSAS=?** Shows if the command is supported.

Test command response: +CSAS: (list of supported <profile>s)

Parameter:

<profile>:

<profile>	Description
0..255	Manufacturer specific profile number where settings are to be stored. **Default value: 0**

AT+CRES Restore settings

Description: Restores the message service settings from non-volatile memory to active memory. A TA can contain several profiles of settings. The settings specified in **AT+CSCA** are restored. Certain settings, for example, SIM SMS parameters, may not be supported by the storage and can therefore not be restored.

Execution command: **AT+CRES**[=<profile>]

Test command: **AT+CRES=?** Shows if the command is supported.

Test command response: +CRES: (list of supported <profile>s)

Parameter:

<profile>:

<profile>	Description
0..255	Manufacturer specific profile number where settings are to be stored. **Default value: 0**

AT+CNMI New messages indication to TE (ver. 4)

Description: Selects procedure how receiving of new messages from the network is indicated to the TE when TE is active, for example, DTR signal is ON. If TE is inactive, for example, DTR signal is OFF, message receiving should be done as specified in GSM 03.38 (3G TS 23.038).

Set command: **AT+CNMI=**[<mode>[,<mt>[,<bm>[,<ds>[,<bfr>]]]]]

Read command: **AT+CNMI?** Displays the current settings.

Read command response: +CNMI: <mode>,<mt>,<bm>,<ds>,<bfr>

Test command: **AT+CNMI=?** Shows if the command is supported.

Test command response: +CNMI: (list of supported <mode>s),(list of supported <mt>s),(list of supported <bm>s),(list of supported <ds>s),(list of supported <bfr>s)

Parameters:

<mode>:

<mode>	Description
2	Buffer unsolicited result code in TA when TA – TE link is reserved, for example, in online data mode, and flush them to the TE after reservation. Otherwise forward them directly to the TE

<mt>:

<mt>	Description
0	No SMS-DELIVER indications are routed to the TE. **Default value**

<mt>	Description
1	If SMS-DELIVER is stored into ME/TA, indication of the memory location is routed to the TE using unsolicited result code: **+CMTI**: <mem>, <index>
3	Class 3 SMS-DELIVERs are routed directly to TE using unsolicited result codes **+CMT**: <length>**<CR><LF>**<pdu>. Messages of other data coding schemes result in indication as defined in <mt>=1

<bm>:

<bm>	Description
0	Store message to “BM” (or some manufacturer specific memory). No CBM indications are routed to the TE. **Default value**
2	New CBMs are routed directly to the TE using unsolicited result code: **+CBM**: <length>**<CR><LF>**<pdu> (PDU mode enabled) or **+CBM**: <sn>,<mid>,<dcs>,<page>,<pages>**<CR><LF>**<data> (text mode enabled)

<ds>:

<ds>	Description
0	No SMS-STATUS-REPORTs are routed to the TE. **Default value**
1	SMS-STATUS-REPORTs are routed to the TE using unsolicited result code: **+CDS**: <length>**<CR><LF>**<pdu> (PDU mode enabled) or +CDS: <fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st> (text mode enabled)

<bfr>:

<bfr>	Description
0	TA buffer of unsolicited result codes defined within this command is flushed to the TE when <mode> 1–3 is entered (OK response is given before flushing the codes). **Default value**

AT+CMGL List message (ver. 2)

Description: Returns messages with status value <stat> from preferred message storage <mem1> to the TE. Entire data units <pdu> are returned. If status of the message is “received unread”, status in the storage changes to “received read”.

Execution command: **AT+CMGL**[=<stat>]

Execution command response: +CMGL:<index>,<stat>,[<alpha>],<length>**<CR><LF>**<pdu>[**<CR><LF>**
+CMGL: <index>,<stat>,[<alpha>],<length>**<CR><LF>**<pdu>[...]]

Test command: **AT+CMGL=?** Shows if the command is supported.

Test command response: +CMGL: (list of supported <stat>s)

Parameters:

<stat>:

<stat>	**Description**
0	Received unread message, that is, new message. **Default value**
1	Received read message
2	Stored unsent message. (Only applicable to SMs)
3	Stored sent message. (Only applicable to SMs)
4	All messages. (Only applicable to +CMGL command)

<index>:

<index>	**Description**
Integer type	Value in the range of location numbers supported by the associated memory

<alpha>:

<alpha>	**Description**
String type	Manufacturing specific. Should be left empty but not omitted, that is, commas mark the place were it should be. Used character set should be the one selected with command **AT+CSCS**

<length>:

<length>	**Description**
Integer type	Value indicating in PDU mode (**AT+CMGF**=”0”), the length of the actual TP data unit in octets. The RP layer SMSC address octets are not counted in the length

<pdu>:

<pdu>	Description
Hexadecimal value	In the case of SMS: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format. The phone converts each octet of TP data unit into two IRA character long hexadecimal number, for example, octet with integer value 42 is presented to terminal equipment as two characters 2A (IRA 50 and 65). In the case of CBS: GSM 03.41 TPDU in hexadecimal format

<mem1>: See **AT+CPMS**.

AT+CMGR Read message (ver. 2)

Description: Returns message with location value <index> from preferred message storage <mem1> to the TE. Status of the message and entire message data unit <pdu> is returned. If status of the message is “received unread”, status in the storage changes to “received read”.

Execution command: **AT+CMGR=**<index>

Execution command response: +CMGR: <stat>,[<alpha>],<length>**<CR><LF>**<pdu>

Test command: **AT+CMGR=?** Shows if the command is supported.

Parameters:

<stat>:

<stat>	Description
0	Received unread message (new message)
1	Received read message
2	Stored unsent message. (Only applicable to SMs)
3	Stored sent message. (Only applicable to SMs)
16	Template message

Integer type in PDU mode (default 0), indicates the status of message in memory.

<index>:

<index>	Description
Integer type	Value in the range of location numbers supported by the associated memory

<alpha>:

<alpha>	Description
String type	Manufacturing specific. Should be left empty but not omitted, that is, commas should mark the place where it should be. Used character set should be the one selected with command **AT+CSCS**

<length>:

<length>	Description
Integer type	Value indicating in PDU mode (**AT+CMGF**=”0”), the length of the actual TP data unit in octets. The RP layer SMSC address octets are not counted in the length)

<pdu>:

<pdu>	Description
Hexadecimal value	In the case of SMS: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format. ME/TA converts each octet of TP data unit into two IRA character long hexadecimal number, for example, octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65). In the case of CBS: GSM 03.41 TPDU in hexadecimal format

<mem1>: See **AT+CPMS**

AT+CMGS Send message (ver. 2)

Description: Sends message from a TE to the network (SMS-SUBMIT). Message reference value <mr> is returned to the TE on successful message delivery. Optionally (when **AT+CSMS** <service> value is 1 and network supports it) <ackpdu> is returned. Values can be used to identify message upon unsolicited delivery status report result code as follows:

- <length> must indicate the number of octets coded in the TP layer data unit to be given (SMSC address octets are excluded).
- The TA sends a four character sequence **<CR><LF>**<greater_than><space> (IRA 13, 10, 62, 32) after command line is terminated with **<CR>**. After that PDU can be given from TE to ME/TA.
- The DCD signal will be in ON state while PDU is given.
- The echoing of given characters back from the TA is controlled by V.25ter echo command E.
- The PDU is in hexadecimal format (similarly as specified for <pdu>) and is given in one line. ME/TA converts this coding into the actual octets of PDU.
- When the length octet of the SMSC address (given in the <pdu>) equals zero, the SMSC address set with command Service Centre Address +CSCA is used.
- Sending can be cancelled by giving **<ESC>** character (IRA 27).
- **<ctrl-Z>** (IRA 26) must be used to indicate the ending of PDU.

Execution command: **AT+CMGS=**<length>**<CR>**<pdu>**<ctrl-Z/ESC>**

Execution command response: +CMGS: <mr>[,<ackpdu>]

Test command: **AT+CMGS=?** Shows if the command is supported.

Parameters:

<mr>:

<mr>	**Description**
Integer type	GSM 03.40 TP-Message-Reference in integer format

<length>:

<length>	**Description**
Integer type	Value indicating in PDU mode (**AT+CMGF**=”0”), the length of the actual TP data unit in octets. The RP layer SMSC address octets are not counted in the length

<ackpdu>:

<ackpdu>	**Description**
Hexadecimal value	GSM 03.40 RP-User Data element of RP-ACK PDU. Format is the same as for <pdu> in case of SMS, but without GSM 04.11 SC address field and parameter are bounded by double quote characters like a normal string type parameter

AT+CMSS Send from storage (ver. 2)

Description: Sends message with location value <index> from message storage <mem2> (see **AT+CPMS**) to the network (SMS-SUBMIT or SMS-COMMAND). <mr> is returned after successful delivery.
Note: Parameters <da> and <toda> will be accepted but ignored by ME (that is, ME will not use the parameters).

Execution command: **AT+CMSS=**<index>[,<da>[,<toda>]]

Execution command response: +CMSS: <mr>

Test command: **AT+CMSS=?** Shows if the command is supported.

Parameters:

<index>: Integer. Value in the range of location numbers supported by the associated memory.

<da>: GSM 03.40 TP-Destination-Address. Address value field in string format. BCD numbers are converted into characters of the currently selected terminal equipment character set. The type of address is given by <toda>.

<toda>: GSM 04.11 TP-Address Type-Of-Address octet in integer format.

<toda>	Description
129	ISDN/telephony numbering plan, national/ international unknown. **Default value if "+" is not in <da>**
145	ISDN/telephony numbering plan, international number. **Default value if "+" is in <da>**
161	ISDN/telephony numbering plan, national number
128–255	Valid values, see GSM 04.08 section 10.5.4.7

<mr>: Integer. GSM 03.40 TP-Message-Reference.

AT+CMGW Write message to memory (ver. 2)

Description: Stores a message to memory storage <mem2>. Memory location <index> of the stored message is returned. By default message status will be set to "stored unsent", but parameter <stat> allows also other status values to be given. (ME/TA manufacturer may choose to use different default <stat> values for different message types.) The entering of PDU is done in a similar way as with command **AT+CMGS**.

Execution command: **AT+CMGW=**<length>[,<stat>]**<CR>**<pdu>**<ctrl-Z/ESC>**

Execution command response: +CMGW: <index>

Test command: **AT+CMGW=?** Shows if the command is supported.

Parameters:

<stat>:

<stat>	Description
0	Received unread message (new message)
1	Received read message
2	Stored unsent message. (Only applicable to SMs)
3	Stored sent message. (Only applicable to SMs)
16	Template message

<index>:

<index>	Description
Integer type	Value in the range of location numbers supported by the associated memory

<length>:

<length>	Description
Integer type	Value indicating in PDU mode (**AT+CMGF**="0"), the length of the actual TP data unit in octets. The RP layer SMSC address octets are not counted in the length

<pdu>:

<pdu>	Description
Hexadecimal value	In the case of SMS: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format. ME/TA converts each octet of TP data unit into two IRA character long hexadecimal number. For example, octet with integer value 42 is presented to TE as two characters, 2A (IRA 50 and 65). In the case of CBS: GSM 03.41 TPDU in hexadecimal format

AT+CMGD Delete message

Description: Deletes message from preferred message <mem1> (see **AT+CPMS**) storage location <index>.
If <delflag> is present and not set to 0 then the ME will ignore <index> and follow the rules for <delflag> shown below.

Execution command: **AT+CMGD=**<index>,<delflag>

Test command: **AT+CMGD=?** Shows if the command is supported.

Test command response: +CMGD: <index>, <list of delflags>

Parameter:

<index>:

<index>	Description
Integer	Value in the range of location numbers supported by the associated memory

<delflag>:

<delflag>	Description
Integer	Integer indicating request for multiple message deletion
0	Delete the message specified in <index>
1	Delete all read messages from preferred message storage leaving unread messages and stored mobile originated messages (whether sent or not) untouched
2	Delete all read messages from preferred message storage and sent mobile originated messages, leaving unread messages and unsent mobile originated messages untouched
3	Delete all read messages from preferred message storage and sent mobile originated messages, leaving unread messages untouched
4	Delete all messages from preferred message storage including unread messages

AT+CMGC Send command (ver. 1)

Description: Sends a command message from a TE to the network (SMS-COMMAND). The entering of PDU is done in a similar way as with the command Send Message, **AT+CMGS**. Message reference value <mr> is returned to the TE on successful message delivery. Values can be used to identify the message upon unsolicited delivery status report result code.

Execution command: **AT+CMGC=**<length>**<CR>**<pdu>**<ctrl-Z/ESC>**

Execution command response: If PDU mode (+CMGF=0) and sending successful:
+CMGC: <mr>[,<ackpdu>]

Test command: **AT+CMGC=?** Shows if the command is supported.

Parameters:

<length>:

<length>	**Description**
Integer type	Value indicating in PDU mode (+CMGF=0), the length of the actual TP data unit in octets. The RP layer SMSC address octets are not counted in the length

<pdu>:

<pdu>	**Description**
Hexadecimal value	In the case of SMS: 3G TS 24.011 SC address followed by 3G TS 23.040 TPDU in hexadecimal format. ME converts each octet of TP data unit into two IRA character long hexadecimal number. For example, octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65). In the case of CBS: 3G TS 23.041 TPDU in hexadecimal format

<mr>:

<mr>	**Description**
Integer type	3GPP TS 23.040 TP-Message-Reference in integer format

<ackpdu>:

<ackpdu>	**Description**
Hexadecimal value	GSM 03.40 RP-User-Data element of RP-ACK PDU. Format is the same as for <pdu> in case of SMS, but without GSM 04.11 SC address field. The parameter is bounded by double quote characters like a normal string type parameter. **Not supported**

AT+CMMS More messages to send

Description: Controls the continuity of SMS relay protocol link. When the feature is enabled (and supported by the network), multiple messages can be sent much faster as the link is kept open.

Set command: *Set mode of SMS relay protocol link:*
AT+CMMS=<n>

Read command: **AT+CMMS?**
Display current setting. +CMMS:<n>

Test command: **AT+CMMS=?** Test if command is supported and return supported parameter values as a compound value.

Test command response +CMMS: (list of supported <n>s)

Parameter:

<n>:

<n>	Description
0	Disable. **Default value**
1	Keep enabled until the time between the response of the latest message send command (+CMGS, +CMSS, and so on) and the next send command exceeds 1–5 seconds (the exact value is up to ME implementation), then ME closes the link and TA switches <n> automatically back to 0
2	Enable. if the time between the response of the latest message send command and the next send command exceeds 1–5 seconds (the exact value is up to ME implementation), ME closes the link but TA does not switch automatically back to <n>=0

Unsolicited result codes

+CBM Received cell broadcast

Description: Received CBMs are routed directly to the terminal equipment. Enabled by **AT+CNMI**.

Unsolicited result code: **+CBM:** <length><pdu>

Parameters:

<length>: Integer. With **AT+CMGF**=”0”, this value indicates the length of the actual TP data unit (in octet units).

<pdu>: **In case of SMS**: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format. phone converts each octet of TP data unit into two IRA-character long hexadecimal numbers.
In case of CBS: GSM TPDU in hexadecimal format.

+CMTI New message indication

Description: Indication of the message memory location is routed to the terminal equipment. Enabled by **AT+CNMI**.

Unsolicited result code: **+CMTI:** <mem>,<index>

Parameters:

<mem>:

<mem>	**Description**
"ME"	Phone message storage
"SM"	SIM message storage

<index>: Integer. Value is in the range of location numbers supported by the associated memory.

+CMT Received message

Description: Received SMs are routed directly to the terminal equipment. Enabled by **AT+CNMI**.

Unsolicited result code: **+CMT:** <length>**<CR><LF>**<pdu>

Parameters:

<length>: Integer. With **AT+CMGF**="0", this value indicates the length of the actual TP data unit (in octet units).

<pdu>: **In case of SMS**: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format. The phone converts each octet of TP data unit into two IRA-character long hexadecimal numbers.
In case of CBS: GSM TPDU in hexadecimal format.

+CDS SMS status report

Description: SMS status is indicated to the terminal equipment. Enabled by **AT+CNMI**.

Unsolicited result code: **+CDS:** <length>**<CR><LF>**<pdu>

Parameters:

<length>: Integer. With **AT+CMGF**="0", this value indicates the length of the actual TP data unit (in octet units).

<pdu>: **In case of SMS**: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format. The phone converts each octet of TP data unit into two IRA-character long hexadecimal numbers.
In case of CBS: GSM TPDU in hexadecimal format.

Use scenarios

New message indication

This scenario shows how the new message indication result codes are handled.

AT command	Response	Comment
AT+CNMI=?		Query new message unsolicited result code modes
	+CNMI: (3),(0-1),(0,2),(0),(0) OK	
AT+CNMI=0,1,2,0,0		Send SM indications to terminal equipment. Forward unsolicited CBM result codes directly to the terminal equipment
	OK	
AT+CNMI?		Query current settings
	+CNMI: 3,1,2,0,0	
		The phone receives and stores incoming SM
	+CMTI: “ME”,3	New message stores in index 3 of <mem1> storage
		The phone receives a CBM and routes it directly to the terminal equipment
	+CBM: 128 <128 byte PDU>	New CBM PDU of 128 byte received at terminal equipment

Ensemble S15: GPRS/packet domain

Locked PDP contexts

In Sony Ericsson phones every PDP context has a one-to-one relationship with an Internet Account. If a certain Internet account is locked, the corresponding PDP context will also be locked for editing. As a consequence, an attempt to select PDP context parameters with **AT+CGDCONT** may fail even though the CID of the context is within the range reported with the test command. The read and test commands in this ensemble are not affected by these restrictions.

Commands

AT+CGDCONT Define PDP context (ver. 1)

Description: Specifies the PDP context parameter values for a PDP context identified by the <cid> parameter. The number of PDP contexts that may be in a defined state at the same time is given by the range returned by the test command.
A special form of the set command, +CGDCONT= <cid> causes the values for context number <cid> to become undefined.
The test command returns values supported as a compound value. Each PDP type will have the parameter value range returned on a separate line.
Note:

- The PDP context, identified by the <cid>, does not have to be earlier defined (using the +CGDCONT command).
- Letting values for context number <cid> become undefined, means that the values of the <cid> are given the default values.
- The read command returns the current settings for each context defined by the +CGDCONT set command.

Set command: **AT+CDGCONT=**[<cid>[,<pdp_type>[,<APN>[,<pdp_addr>[,<d_comp>[,<h_comp>[,<pd1>[,...[,<pd*N*>]]]]]]]]

Read command: **AT+CGDCONT?** Displays the current parameter settings.

Read command response: +CGDCONT: <cid>, <PDP_type>,<APN>, <PDP_addr>,<d_comp>, <h_comp>[,<pd1>[,…[,pdN]]] [**<CR><LF>**+CGDCONT: <cid>, <PDP_type>,<APN>, <PDP_addr>,<d_comp>, <h_comp>[,<pd1>[,…[,pdN]]] [...]]

Test command: **AT+CGDCONT=?** Shows if the command is supported.

Test command response: +CGDCONT: (range of supported <cid>s), <PDP_type>,,,(list of supported <d_comp>s), (list of supported <h_comp>s)[,(list of supported <pd1>s)[,…[,(list of supported <pdN>s)]]] [<CR><LF>+CGDCONT: (range of supported <cid>s), <PDP_type>,,,(list of supported <d_comp>s), (list of supported <h_comp>s)[,(list of supported <pd1>s)[,…[,(list of supported <pdN>s)]]] [...]]

Parameters:

<cid>: Integer. Specifies the particular PDP context definition. The parameter is local to the TE – MT interface and is used in other PDP-context related commands. The range of permitted values (minimum value=1) is returned by the test command.

<pdp_type>:

<pdp_type>	**Description**
"IP"	Internet Protocol (IETF STD 5). **Default value**
"IPV6"	Internet Protocol, version 6 (IETF RFC 2460)

<APN>: Access Point Name – a string parameter which is a logical name, used to select the GGSN or the external packet data network.
If the value is null or omitted, then the subscription value will be requested.

<pdp_address>: A string parameter that identifies the ME in the address space applicable to the PDP.
If the value is null or omitted, then a value may be provided by the TE during the PDP startup procedure or, failing that, a dynamic address will be requested. See 3GPP TS 27.007

<d_comp>:

<d_comp>	Description
0	PDP data compression OFF. **Default value**
1	PDP data compression ON
2–255	Reserved

<h_comp>:

<h_comp>	Description
0	PDP header compression OFF. **Default value**
1	PDP header compression ON
2–255	Reserved

<pd*N*>: Zero to *N* string parameters whose meanings are specific to the <pdp_type>. **Not supported**

AT+CGSMS Select service for MO SMS messages

Description: The command specifies the service or service preference that the MT will use to send MO SMS messages.

Set command: **AT+CGSMS=**[<service>]

Read command: **AT+CGSMS?** Displays the current <service> setting.

Test command: **AT+CGSMS=?** Shows if the command is supported.

Test command response: +CGSMS: (list of supported <service>s)

Parameter:

<service>:

<service>	Description
0	GPRS/Packet Domain
1	Circuit switched
2	GPRS/Packet Domain preferred (use circuit-switched if GPRS/Packet Domain not available)
3	Circuit-switched preferred (use GPRS/Packet Domain if circuit-switched not available). **Default value**

AT+CGATT Packet service attach or detach

Description: Attaches the ME to, or detaches the ME from, the GPRS/Packet Domain service. After the command has completed, the ME remains in V.25ter command state. If the ME is already in the requested state, the command is ignored and the OK response is returned. If the requested state cannot be achieved, an ERROR or +CME ERROR response is returned.
Any active PDP contexts will be automatically deactivated when the attachment state changes to detached.
Note: This command has the characteristics of both the V.25ter action and parameter commands. Hence it has the read form in addition to the execution/set and test forms.

Set command: **AT+CGATT=**[<state>]

Read command: **AT+CGATT?** Displays the current <state> settings

Test command: **AT+CGATT=?** Shows if the command is supported.

Test command response: +CGATT: (list of supported <state>s)

Parameter:

<state>:

<state>	Description
0	Detached
1	Attached

AT+CGACT PDP context activate or deactivate

Description: Activates or deactivates the specific PDP context(s).
After the command has completed, the ME remains in V.25ter command state. If any PDP context is already in the requested state, the state for that context remains unchanged. If the requested state for any specified context cannot be achieved, an ERROR or +CME ERROR response is returned. If the ME is not GPRS/Packet Domain attached when the activation form of the command is executed, the ME first performs a GPRS/Packet Domain attach and then attempts to activate the specified contexts. If the attach fails then the ME responds with ERROR or, if extended error responses are enabled, with the appropriate failure-to-attach error message.
If no <cid>s are specified the activation/deactivation form of the command activates/deactivates all defined contexts.
Note: This command has the characteristics of both the V.25ter action and parameter commands. Hence it has the read form in addition to the execution/set and test forms.

Set command: **AT+CGACT=**[<state>[,<cid>[,<cid>[,...]]]]

Read command: **AT+CGACT?** Displays the current <cid> and <state> settings.

Read command response: +CGACT: <cid>,<state>**<CR><LF>**
[+CGACT: <cid>,<state>**<CR><LF>**
[...]]

Test command: **AT+CGACT=?** Shows if the command is supported.

Test command response: +CGACT: (list of supported <state>s)

Parameters:

<state>:

<state>	Description
0	PDP context activation deactivated
1	PDP context activation activated

<cid>: Integer. Specifies the particular PDP context definition. See the **AT+CGDCONT** command.

AT+CGDATA Enter data state

Description: Causes the ME to perform whatever actions are necessary to establish communication between the TE and the network using one or more GPRS/ Packet Domain PDP types. This may include performing a GPRS/Packet Domain attach and one or more PDP context activations.
Commands following +CGDATA command in the AT command line are not processed by the ME.
During each PDP startup procedure the ME may have access to some or all of the following information:

- The command may have provided an <L2P> parameter value.
- The TE may provide a PDP type and/or PDP address to the ME during in the PDP startup procedure.

If any of this information is in conflict, the command will fail.
Any PDP type and/or PDP address present in the above information are compared with the PDP type and/or PDP address in any context definitions specified in the command in the order in which their <cid>s appear.
For a context definition to match:

- The PDP type must match exactly.
- The PDP addresses are considered to match if they are identical or if either or both addresses are unspecified. For example, a PPP NCP request specifying PDP type = IP and no PDP address would cause the ME to search through the specified context definitions for one with PDP type = IP and any PDP address.

The context is activated using the matched value for PDP type and a static PDP address, if available, together with the other information found in the PDP context definition. If a static PDP address is not available then a dynamic address is requested.
If no <cid> is given or if there is no matching context definition, the ME will attempt to activate the context with whatever information is available to the ME. The other context parameters will be set to their default values.
If the activation is successful, data transfer may proceed.
After data transfer is complete and the layer 2 protocol termination procedure has completed successfully, the V.25ter command state is re-entered and the ME returns the final result code OK.
In the event of an erroneous termination or a failure to start up, the V.25ter command state is re-entered and the ME returns the final result code NO CARRIER or, if enabled, +CME ERROR. Attach, activate and other errors may be reported.

Set command: **AT+CGDATA=**[<L2p>[,<cid>[,<cid>[,...]]]]

Test command: **AT+CGDATA=?** Shows if the command is supported.

Test command response: +CGDATA: (list of supported <L2p>s)

Parameters:

<L2p>: Layer 2 protocol used between ME and TE.

<L2p>	Description
"PPP"	Point-to-Point Protocol for a PDP, such as IP

<cid>: Integer. Specifies the particular PDP context definition. See the **AT+CGDCONT** command.

AT+CGEREP Packet domain event reporting (ver. 1)

Description: Enables or disables sending of the unsolicited result code **+CGEV** from ME to TE in the case of certain events occurring in the GPRS/Packet Domain ME or the network. <mode> controls the processing of unsolicited result codes specified within this command. <bfr> controls the effect on buffered codes when <mode> 1 or 2 is entered.

Set command: **AT+CGEREP=**[<mode>[,<bfr>]]

Read command: **AT+CGEREP?** Displays the current <mode> and <bfr> settings.

Test command: **AT+CGEREP=?** Shows if the command is supported.

Test command response: +CGEREP: (list of supported <mode>s),(list of supported <bfr>s)

Parameters:

<mode>:

<mode>	Description
0	Buffer unsolicited result codes in the ME. No codes are forwarded to the TE. **Default value**
1	Discard unsolicited result codes when MT – TE link is reserved, for example, in online data mode, otherwise forward them directly to the TE

<bfr>:

<bfr>	Description
0	MT buffer of unsolicited result codes defined within this command is cleared when <mode> 1 is entered. **Default value**

AT+CGREG Packet domain network registration status

Description: Controls the presentation of the unsolicited result code **+CGREG: <stat>** when <n>='1' and there is a change in the ME GPRS network registration status or **+CGREG: <stat>[,<lac>,<ci>]** when <n>='2' and there is a change of the network cell.
Note: If the GPRS MT also supports circuit mode services, **AT+CREG** and the +CREG result code apply to the registration status and location information for those services.
The read command returns the status of result code presentation and an integer <stat> which shows whether the network has currently indicated the registration of the ME. Location information elements <lac> and <ci> are returned only when <n>=2 and ME is registered in the network.

Set command: **AT+CGREG=**[<n>]

Read command: **AT+CGREG?** Displays the current <n>, <stat>[, <lac>,<ci>] settings.

Test command: **AT+CGREG=?** Shows if the command is supported.

Test command response: +CGREG: (list of supported <n>s)

Parameters:

<n>:

<n>	Description
0	Disable network registration unsolicited result code. **Default value**
1	Enable network registration unsolicited result code
2	Enable network registration and location information unsolicited result code

<stat>:

<stat>	Description
0	Not registered, ME is not currently searching a new operator to register to
1	Registered, home network
2	Not registered, but ME is currently searching a new operator to register to
3	Registration denied
4	Unknown
5	Registered, roaming

<lac>: Two byte location area code in hexadecimal format.

<ci>: Four byte cell ID in hexadecimal format. Four bytes are required for UMTS, whereas only two bytes are applicable for GSM. For GSM the two first bytes are zeros, for example, 00001A02

AT+CGPADDR Show PDP address

Description: Returns a list of PDP addresses for the specified context identifiers.

Execution command:	**AT+CGPADDR=**[<cid>[,<cid>[,...]]]
Response:	+CGPADDR: <cid>,<pdp_addr>**<CR><LF>** [+CGPADDR: <cid>,<pdp_addr>**<CR><LF>** [...]]
Test command:	**AT+CGPADDR=?** Shows if the command is supported.
Test command response:	+CGPADDR: (list of defined <cid>s)
Parameters:	
<cid>:	Integer. Specifies a particular PDP context definition (see **AT+CGDCONT**). If no <cid> is specified, the addresses for all defined contexts are returned.
<pdp_address>:	A string that identifies the ME in the address space applicable to the PDP. The address may be static or dynamic. For a static address, it will be the one set by the +CGDCONT and +CGDSCONT commands when the context was defined. For a dynamic address it will be the one assigned during the last PDP context activation that used the context definition referred to by <cid>. <PDP_address> is omitted if none is available

AT+CGDSCONT Define secondary PDP context

Description:	Specifies PDP context parameter values for a Secondary PDP context identified by the (local) context identification parameter, <cid>. The number of PDP contexts that may be in a defined state at the same time is given by the range returned by the test command. A special form of the set command, +CGDSCONT= <cid> causes the values for context number <cid> to become undefined. The read command returns the current settings for each defined context. The test command returns values supported as a compound value. If the MT supports several PDP types, <PDP_type>, the parameter value ranges for each <PDP_type> are returned on a separate line. **Note**: If <cid> states an already existing, primary context, this will be converted to a secondary one, provided of course, that stated <p_cid> is a different and existing primary account defined with +CGDSCONT. Any secondary contexts attached to the converted context disappears.
Set command:	**AT+CGDSCONT=**[<cid> ,<p_cid> [,<d_comp> [,<h_comp>]]]
Read command:	**AT+CGDSCONT?** (Read the current general parameter settings)

Read command response:

+CGDSCONT: <cid>, <p_cid>, <data_comp>, <head_comp>[**<CR><LF>**
+CGDSCONT: <cid>, <p_cid>, <data_comp>, <head_comp>
[...]]

Test command: **AT+CGDSCONT=?** Shows if the command is supported.

Test command response:

+CGDSCONT: (range of supported <cid>s), (list of <cid>s for active primary contexts), <PDP_type>,,,(list of supported <d_comp>s),(list of supported <h_comp>s)[**<CR><LF>**
+CGDSCONT: (range of supported <cid>s), (list of <cid>s for active primary contexts),<PDP_type>,,,(list of supported <d_comp>s), (list of supported <h_comp>s)
[...]]

Parameters:

<p_cid>:

<p_cid>	**Description**
Integer	Primary PDP Context Identifier – a numeric parameter which specifies a particular PDP context definition which has been specified by use of the **AT+CGDCONT** command. The parameter is local to the TE – MT interface. The list of permitted values is returned by the test form of the command

Other parameters: See **AT+CGDCONT**

AT+CGTFT Traffic flow template

Description:

Allows the TE to specify a Packet Filter for a Traffic Flow Template (TFT) that is used in the GGSN for routing of downlink packets onto different QoS flows towards the TE. The concept is further described in the 3GPP TS 23.060. A TFT consists of from one to up to eight Packet Filters, each identified by a unique <packet filter identifier>. A Packet Filter also has an <evaluation precedence index> that is unique within all TFTs associated with all PDP contexts that are associated with the same PDP address.
The set command specifies a Packet Filter that is to be added to the TFT stored in the ME and used for the context identified by the (local) context identification parameter, <cid>. The specified TFT will be stored in the GGSN only at activation or MS-initiated modification of the related context. Since this is the same parameter that is used in the **AT+CGDCONT** and **AT+CGDSCONT** commands, the +CGTFT command is effectively an extension to these commands. The Packet Filters consist of a number of parameters, each of which may be set to a separate value.

A special form of the set command, +CGTFT= <cid> causes all of the Packet Filters in the TFT for context number <cid> to become undefined. At any time there may exist only one PDP context with no associated TFT amongst all PDP contexts associated to one PDP address. At an attempt to delete a TFT, which would violate this rule, an ERROR or **+CME ERROR** response is returned.
The read command returns the current settings for all Packet Filters for each defined context. In case no filter is defined the read command will return “OK” only.
The test command returns values supported as a compound value. If the MT supports several PDP types, the parameter value ranges for each PDP type are returned on a separate line. TFTs will be used for PDP-type IP and PPP only. For PDP-type PPP a TFT is applicable only when IP traffic is carried over PPP. If PPP carries header-compressed IP packets, then a TFT cannot be used.

Set command:

AT+CGTFT=[<cid>, [<packet filter identifier>, <evaluation precedence index> [,<source address and subnet mask> [,<protocol number (ipv4)/ next header (ipv6)> [,<destination port range> [,<source port range> [,<ipsec security parameter index (spi)> [,<type of service (tos) (ipv4) and mask/traffic class (ipv6) and mask> [,<flow label (ipv6)>]]]]]]]]>

Read command:

AT+CGTFT? (Read the current general parameter settings)

Read command response:

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

Test command:

AT+CGTFT=? Shows if the command is supported.

Test command response:

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

Parameters:

<cid>:

See the **AT+CGDCONT** command. The <cid> may refer to a primary or a secondary account.

<packet filter identifier>:

<packet filter identifier>	**Description**
1–8	Supported values

<source address and subnet mask>:

<source address and subnet mask>	**Description**
String	Consists of dot-separated numeric (0–255) parameters on the form “a1.a2.a3.a4.m1.m2.m3.m4” for IPv4 and “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

<protocol number (ipv4)/next header (ipv6)>:

<protocol number (ipv4)/next header (ipv6)>	Description
0–255	Supported values

<destination port range>:

<destination port range>	Description
String	Consists of dot-separated numeric (0–65535) parameters on the form “f.t”

<source port range>:

<source port range>	Description
String	Consists of dot-separated numeric (0–65535) parameters on the form “f.t”

<ipsec security parameter index (spi)>:

<ipsec security parameter index (spi)>	Description
00000000–FFFFFFFF	Supported values (hexadecimal)

<type of service (tos) (ipv4) and mask/traffic class (ipv6) and mask>:

<type of service (tos) (ipv4) and mask/traffic class (ipv6) and mask>	Description
String	Dot-separated numeric (0–255) parameters on the form “t.m”

<flow label (ipv6)>:

<flow label (ipv6)>	Description
00000–FFFFF	Supported values. **Valid for IPv6 only**

<evaluation precedence index>:

<evaluation precedence index>	Description
0–255	Supported values

AT+CGEQREQ 3G quality of service profile (requested)

Description: Allows the TE to specify a UMTS Quality of Service profile that is used when the MT sends an Activate PDP context request message to the network.
The set command specifies a profile for the context identified by the (local) context identification parameter, <cid>. The specified profile will be stored in the ME and sent to the network only at activation or ME-initiated modification of the related context. Since this is the same parameter as the one used in the **AT+CGDCONT** and **AT+CGDSCONT** commands, the +CGEQREQ command is effectively an extension to these commands. The QoS profile consists of a number of parameters, each of which may be set to a separate value.
The read command returns the current settings for each defined context.
The test command returns values supported as a compound value. If the MT supports several PDP types, the parameter value ranges for each PDP type are returned on a separate line.

Set command: **AT+CGEQREQ=**[<cid> [,<Traffic class> [,<Maximum bitrate UL> [,<Maximum bitrate DL> [,<Guaranteed bitrate UL> [,<Guaranteed bitrate DL> [,<Delivery order> [,<Maximum SDU size> [,<SDU error ratio> [,<Residual bit error ratio> [,<Delivery of erroneous SDUs> [,<Transfer delay> [,<Traffic handling priority>]]]]]]]]]]]]

Read command: **AT+CGEQREQ?** (Read the current general parameter settings)

Read command response:

+CGEQREQ: <cid>, <Traffic class>,<Maximum bitrate UL>,<Maximum bitrate DL>,<Guaranteed bitrate UL>,<Guaranteed bitrate DL>,<Delivery order>,<Maximum SDU size>,<SDU error ratio>,<Residual bit error ratio>,<Delivery of erroneous SDUs>,<Transfer delay>,<Traffic handling priority>[**<CR><LF>**
+CGEQREQ: <cid>, <Traffic class>,<Maximum bitrate UL>,<Maximum bitrate DL>,<Guaranteed bitrate UL>,<Guaranteed bitrate DL>,<Delivery order>,<Maximum SDU size>,<SDU error ratio>,<Residual bit error ratio>,<Delivery of erroneous SDUs>,<Transfer delay>,<Traffic handling priority>
[…]]

Test command: **AT+CGEQREQ=?** Shows if the command is supported.

Test command response:

+CGEQREQ: <PDP_type>, (list of supported <Traffic class>s),(list of supported <Maximum bitrate UL>s), (list of supported <Maximum bitrate DL>s), (list of supported <Guaranteed bitrate UL>s), (list of supported <Guaranteed bitrate DL>s),(list of supported <Delivery order>s),(list of supported <Maximum SDU size>s),(list of supported <SDU error ratio>s),(list of supported <Residual bit error ratio>s),(list of supported <Delivery of erroneous SDUs>s),(list of supported <Transfer delay>s),(list of supported <Traffic handling priority>s)[**<CR><LF>**
+CGEQREQ: <PDP_type>, (list of supported <Traffic class>s),(list of supported <Maximum bitrate UL>s), (list of supported <Maximum bitrate DL>s), (list of supported <Guaranteed bitrate UL>s), (list of supported <Guaranteed bitrate DL>s),(list of supported <Delivery order>s),(list of supported <Maximum SDU size>s),(list of supported <SDU error ratio>s),(list of supported <Residual bit error ratio>s),(list of supported <Delivery of erroneous SDUs>s),(list of supported <Transfer delay>s),(list of supported <Traffic handling priority>s)
[…]]

Parameters:

<cid>: See the **AT+CGDCONT** command

<Traffic class>:

<Traffic class>	**Description**
0	Conversational
1	Streaming
2	Interactive
3	Background
4	Subscribed value

<Maximum bitrate UL>:

<Maximum bitrate UL>	Description
Integer	The maximum number of Kbps delivered to UMTS (uplink traffic) at a SAP. **Note**: Not all integer values are valid. The general rule is that the integer will be rounded down to the nearest valid value. It is therefore possible that reading back this parameter may produce a different value than that used upon setting
UMTS: **0–384** GPRS/EDGE: **0–128** GPRS: **0–44**	**Default value = 0** – subscribed value will be requested

<Maximum bitrate DL>:

<Maximum bitrate DL>	Description
Integer	Maximum number of Kbps delivered by UMTS (downlink traffic) at a SAP. **Note**: Not all the integer values are valid. The general rule is that the integer will be rounded down to the nearest valid value. It is therefore possible that reading back this parameter may produce a different value than that used upon setting
UMTS: **0–384** GPRS/EDGE: **0–256** GPRS: **0–88**	**Default value = 0** – subscribed value will be requested

<Guaranteed bitrate UL>:

<Guaranteed bitrate UL>	Description
Integer	Guaranteed number of Kbps delivered to UMTS (uplink traffic) at a SAP (provided that there is data to deliver). **Note**: Not all the integer values are valid. The general rule is then that the integer will be rounded down to the nearest valid value. It is therefore possible that reading back this parameter may produce a different value than that used upon setting
UMTS: **0–384** GPRS/EDGE: **0–128** GPRS: **0–44**	**Default value = 0** – subscribed value will be requested

<Guaranteed bitrate DL>:

<Guaranteed bitrate DL>	Description
Integer	Guaranteed number of Kbps delivered by UMTS (downlink traffic) at a SAP (provided that there is data to deliver). **Note**: Not all integer values are valid. The general rule is that the integer will be rounded down to the nearest valid value. It is therefore possible that reading back this parameter may produce a different value than that used upon setting
UMTS: **0–384** GPRS/EDGE: **0–256** GPRS: **0–88**	**Default value = 0** – subscribed value will be requested

<Delivery order>:

<Delivery order>	Description
0	UMTS will not provide in-sequence SDU delivery
1	UMTS will provide in-sequence SDU delivery
2	Subscribed value

<Maximum SDU size>:

<Maximum SDU size>	Description
Integer	Indicates the maximum allowed SDU size in octets
0–153	**Default value = 0** – subscribed value will be requested

<SDU error ratio>:

<SDU error ratio>	Description
String	Indicates the target value for the fraction of SDUs lost or detected as erroneous. SDU error ratio is defined only for conforming traffic. The value is specified as “mEe”. A target SDU error ratio of 510-3 would be specified as “5E3”, for example, AT+CGEQREQ=…,’5E3’,…). “0E0” means subscribed value

<Residual bit error ratio>:

<Residual bit error ratio>	Description
String	Indicates the target value for the undetected bit error ratio in the delivered SDUs. If no error detection is requested, Residual bit error ratio indicates the bit error ratio in the delivered SDUs. The value is specified as “mEe”. As an example a target residual bit error ratio of 510-3 would be specified as “5E3”

<Delivery of erroneous SDUs>:

<Delivery of erroneous SDUs>	Description
0	No delivery of erroneous SDUs
1	Erroneous SDUs delivered
2	No detection of erroneous SDUs
3	Subscribed value. **Default value**

<Transfer delay>:

<Transfer delay>	Description
Integer	The targeted time between request to transfer an SDU at one SAP to its delivery at the other SAP, in milliseconds
0–254	0: **Default value** – subscribed value will be requested

<Traffic handling priority>:

<Traffic handling priority>	Description
Integer	Specifies the relative importance for handling of all SDUs belonging to the UMTS bearer compared to the SDUs of other bearers.
0–3	**Default value = 0** – subscribed value will be requested

<PDP_type>: See the **AT+CGDCONT** command.

AT+CGEQMIN 3G quality of service profile (minimum acceptable)

Description: This command allows the TE to specify a minimum acceptable profile, which is checked by the MT against the negotiated profile returned in the Activate/Modify PDP Context Accept message.
The set command specifies a profile for the context identified by the (local) context identification parameter, <cid>. The specified profile will be stored in the MT and checked against the negotiated profile only at activation or MS-initiated modification of the related context. Since this is the same parameter that is used in the **AT+CGDCONT** and **AT+CGDSCONT** commands, the +CGEQMIN command is effectively an extension to these commands. The QoS profile consists of a number of parameters, each of which may be set to a separate value.
The read command returns the current settings for each defined context.
The test command returns values supported as a compound value. If the MT supports several PDP types, the parameter value ranges for each PDP type are returned on a separate line.

Set command: **AT+CGEQMIN=**[<cid> [,<Traffic class> [,<Maximum bitrate UL> [,<Maximum bitrate DL> [,<Guaranteed bitrate UL> [,<Guaranteed bitrate DL> [,<Delivery order> [,<Maximum SDU size> [,<SDU error ratio> [,<Residual bit error ratio> [,<Delivery of erroneous SDUs> [,<Transfer delay> [,<Traffic handling priority>]]]]]]]]]]]]

Read command: **AT+CGEQMIN?** (Read the current general parameter settings)

Read command response:

+CGEQMIN: <cid>, <Traffic class>,<Maximum bitrate UL>, <Maximum bitrate DL>,<Guaranteed bitrate UL>,<Guaranteed bitrate DL>, <Delivery order>,<Maximum SDU size>,<SDU error ratio>,<Residual bit error ratio>, <Delivery of erroneous SDUs>,<Transfer delay>,<Traffic handling priority>
[**<CR><LF>**
+CGEQMIN: <cid>, <Traffic class>,<Maximum bitrate UL>,<Maximum bitrate DL>,<Guaranteed bitrate UL>,<Guaranteed bitrate DL>, <Delivery order>,<Maximum SDU size>,<SDU error ratio>,<Residual bit error ratio>,<Delivery of erroneous SDUs>,<Transfer delay>,<Traffic handling priority>
[…]]

Test command: **AT+CGEQMIN=?** Shows if the command is supported.

Test command response:

+CGEQMIN: <PDP_type>, (list of supported <Traffic class>s),(list of supported <Maximum bitrate UL>s),(list of supported <Maximum bitrate DL>s), (list of supported <Guaranteed bitrate UL>s), (list of supported <Guaranteed bitrate DL>s),(list of supported <Delivery order>s),(list of supported <Maximum SDU size>s),(list of supported <SDU error ratio>s),(list of supported <Residual bit error ratio>s),(list of supported <Delivery of erroneous SDUs>s),(list of supported <Transfer delay>s),(list of supported <Traffic handling priority>s)
[**<CR><LF>**
+CGEQMIN: <PDP_type>, (list of supported <Traffic class>s),(list of supported <Maximum bitrate UL>s), (list of supported <Maximum bitrate DL>s),(list of supported <Guaranteed bitrate UL >s), (list of supported <Guaranteed bitrate DL >s),(list of supported <Delivery order>s),(list of supported <Maximum SDU size>s),(list of supported <SDU error ratio>s),(list of supported <Residual bit error ratio>s),(list of supported <Delivery of erroneous SDUs>s),(list of supported <Transfer delay>s),(list of supported <Traffic handling priority>s)
[…]]

Parameters:

<cid>: See the **AT+CGDCONT** command

<Traffic class>:

<Traffic class>	**Description**
0	Conversational
1	Streaming
2	Interactive
3	Background. **Default value**

<Maximum bitrate UL>:

<Maximum bitrate UL>	Description
Integer	The maximum number of Kbps delivered to UMTS (uplink traffic) at a SAP. **Note**: Not all integer values are valid. The general rule is that the integer will be rounded down to the nearest valid value. It is therefore possible to read out a value other than that given
UMTS: **0–384** GPRS/EDGE: **0–128** GPRS: **0–44**	0: **Default value** – subscribed value will be requested

<Maximum bitrate DL>:

<Maximum bitrate DL>	Description
Integer	Maximum number of Kbps delivered by UMTS (downlink traffic) at a SAP. **Note**: Not all integer values are valid. The general rule is that the integer will be rounded down to the nearest valid value. It is therefore possible to read out a value other than that given
UMTS: **0–384** GPRS/EDGE: **0–256** GPRS: **0–44**	0: **Default value** – subscribed value will be requested

<Guaranteed bitrate UL>:

<Guaranteed bitrate UL>	Description
Integer	Guaranteed number of Kbps delivered to UMTS (up-link traffic) at a SAP (provided there is data to deliver)
UMTS: **0–384** GPRS/EDGE: **0–128** GPRS: **0–44**	0: **Default value** – subscribed value will be requested

<Guaranteed bitrate DL>:

<Guaranteed bitrate DL>	Description
Integer	Guaranteed number of Kbps delivered by UMTS (downlink traffic) at a SAP (provided there is data to deliver)
UMTS: **0–384** GPRS/EDGE: **0–256** GPRS: **0–44**	0: **Default value** – subscribed value will be requested

<Delivery order>:

<Delivery order>	Description
0	UMTS will not provide in-sequence SDU delivery
1	UMTS will provide in-sequence SDU delivery

<Maximum SDU size>:

<Maximum SDU size>	Description
Integer	Indicates the maximum allowed SDU size in octets **Note**: Not all integer values are valid. The general rule is that the integer will be rounded down to the nearest valid value. It is therefore possible to read out a value other than that given
0–153	0: **Default value** – subscribed value will be requested

<SDU error ratio>:

<SDU error ratio>	Description
String	Indicates the target value for the fraction of SDUs lost or detected as erroneous. SDU error ratio is defined only for conforming traffic. The value is specified as “mEe”. For example, a target SDU error ratio of 510-3 would be specified as “5E3”, AT+CGEQREQ=…,’5E3’,…

<Residual bit error ratio>:

<Residual bit error ratio>	Description
String	String parameter that indicates the target value for the undetected bit error ratio in the delivered SDUs. If no error detection is requested, <Residual bit error ratio> indicates the bit error ratio in the delivered SDUs. The value is specified as “mEe”. For example, a target residual bit error ratio of 510-3 would be specified as “5E3”, AT+CGEQREQ=…,’5E3’,…

<Delivery of erroneous SDUs>:

<Delivery of erroneous SDUs>	Description
0	No delivery of erroneous SDUs
1	Erroneous SDUs delivered
2	No detection of erroneous SDUs

<Transfer delay>:

<Transfer delay>	Description
Integer	The targeted time between request to transfer an SDU at one SAP to its delivery at the other SAP, in milliseconds
0–254	0: **Default value** – subscribed value will be requested

<Traffic handling priority>:

<Traffic handling priority>	Description
Integer	Specifies the relative importance for handling of all SDUs belonging to the UMTS bearer compared to the SDUs of other bearers
0–3	0: **Default value** – subscribed value will be requested

<PDP_type>: See the **AT+CGDCONT** command.

AT+CGEQNEG 3G quality of service profile (negotiated)

Description: Allows the TE to retrieve the negotiated QoS profiles returned in the Activate PDP Context Accept message.
The execution command returns the negotiated QoS profile for the specified context identifiers, <cid>s. The QoS profile consists of a number of parameters, each of which may have a separate value.
The test command returns a list of <cid>s associated with active contexts.

Execution command: **AT+CGEQNEG =**[<cid>[,<cid>[,…]]]

Execution command response:
+CGEQNEG: <cid>, <Traffic class>,<Maximum bitrate UL>, <Maximum bitrate DL>,<Guaranteed bitrate UL>, <Guaranteed bitrate DL>,<Delivery order>,<Maximum SDU size>,<SDU error ratio>,<Residual bit error ratio>,<Delivery of erroneous SDUs>,<Transfer delay>,<Traffic handling priority>[**<CR><LF>**
+CGEQNEG: <cid>, <Traffic class>,<Maximum bitrate UL>, <Maximum bitrate DL>,<Guaranteed bitrate UL>, <Guaranteed bitrate DL>,<Delivery order>,<Maximum SDU size>,<SDU error ratio>,<Residual bit error ratio>,<Delivery of erroneous SDUs>,<Transfer delay>,<Traffic handling priority>
[…]]

Test command: **AT+CGEQREQ=?** Shows if the command is supported.

Test command response: +CGEQNEG: (list of <cid>s associated with active contexts)

Parameters:

<cid>: See the **AT+CGDCONT** command

<Traffic class>:

<Traffic class>	Description
0	Conversational
1	Streaming
2	Interactive
3	Background

<Maximum bitrate UL>:

<Maximum bitrate UL>	Description
Integer	The maximum number of Kbps delivered to UMTS

<Maximum bitrate DL>:

<Maximum bitrate DL>	Description
Integer	Maximum number of Kbps delivered by UMTS (downlink traffic) at a SAP

<Guaranteed bitrate UL>:

<Guaranteed bitrate UL>	Description
Integer	Guaranteed number of Kbps delivered to UMTS (uplink traffic) at a SAP (provided there is data to deliver)

<Guaranteed bitrate DL>:

<Guaranteed bitrate DL>	Description
Integer	Guaranteed number of Kbps delivered by UMTS (downlink traffic) at a SAP (provided there is data to deliver)

<Delivery order>:

<Delivery order>	Description
0	UMTS will not provide in-sequence SDU delivery
1	UMTS will provide in-sequence SDU delivery

<Maximum SDU size>:

<Maximum SDU size>	Description
Integer 0–153	Indicates the maximum allowed SDU size in octets. **Default = 0** – subscribed value will be requested

<SDU error ratio>:

<SDU error ratio>	Description
String	Indicates the target value for the fraction of SDUs lost or detected as erroneous. SDU error ratio is defined only for conforming traffic. The value is specified as “mEe”. For example, a target SDU error ratio of 510-3 would be specified as “5E3”, AT+CGEQREQ=…,’5E3’,…

<Residual bit error ratio>:

<Residual bit error ratio>	Description
String	String parameter that indicates the target value for the undetected bit error ratio in the delivered SDUs. If no error detection is requested, Residual bit error ratio indicates the bit error ratio in the delivered SDUs. The value is specified as “mEe”. For example, a target residual bit error ratio of 510-3 would be specified as “5E3”, AT+CGEQREQ=…,’5E3’,…

<Delivery of erroneous SDUs>:

<Delivery of erroneous SDUs>	Description
0	No delivery of erroneous SDUs
1	Erroneous SDUs delivered
2	No detection of erroneous SDUs

<Transfer delay>:

<Transfer delay>	Description
Integer	The targeted time between request to transfer an SDU at one SAP to its delivery at the other SAP, in milliseconds

<Traffic handling priority>:

<Traffic handling priority>	Description
Integer	Specifies the relative importance for handling of all SDUs belonging to the UMTS bearer compared to the SDUs of other bearers

AT+CGCMOD PDP context modify

Description: Modifies the specified PDP context(s) with respect to QoS profiles and TFTs. After the command has completed, the MT returns to V.25ter online data state.
If no <cid>s are specified the activation form of the command modifies all active contexts.

Execution command: **AT+CGCMOD=**[<cid>[,<cid>[,…]]]

Test command: **AT+CGCMOD=?** Shows if the command is supported.

Test command response: +CGCMOD: (list of <cid>s associated with active contexts)

Parameter:

<cid>: See the **AT+CGDCONT** command.

Extension of ATD – Request GPRS service

Description: The V.25ter ATD command causes the MT to enter the V.25ter online data state and, with the TE, to start the specified layer 2 protocol. The MT returns CONNECT to confirm acceptance of the command prior to entering the V.25ter online data state. No further commands may follow on the AT command line.
When the layer 2 protocol has terminated, either as a result of an orderly shut down of the PDP or an error, the MT enters V.25ter command state and returns the NO CARRIER final result code.
<L2P> and <cid> usage are the same as in the +CGDATA command. The +CGDCONT, +CGQREQ, and so on, commands may be used in the modem initialisation AT command string to set values for PDP type, APN, QoS, and so on.
This command may be used in both normal and modem compatibility modes.
Note: The dial string conforms to the syntax specified in GSM 02.30.

Execution command: **ATD*<GPRS_SC>[*[<called_address>][*[<L2p>][*[<cid>]]]]#**

Parameters:

<GPRS_SC>: Digit string. A digit string (value='99') which identifies a request to use the GPRS/Packet Domain.

<called_address>: String. Identifies the called party in the address space applicable to the PDP.

<L2p>:

<L2p>	Description
1	PPP
9yyyy	M-xxxx

<cid>: Digit string. Specifies a particular PDP context definition.

Extension of ATD – Request packet domain IP service

Description: The V.25ter ATD dial command causes the MT to enter the V.25ter online data state and, with the TE, to start the specified layer 2 protocol. The MT returns CONNECT to confirm acceptance of the command before entering the V.25ter online data state. No further commands may follow on the AT command line.
The detailed behaviour after the online data state has been entered is dependent on the PDP type. It is described briefly in clause 9 (for IP) of 3GPP TS 27.060. PS attachment and PDP context activation procedures may take place prior to or during the PDP startup, if they have not already been performed using the +CGATT and +CGACT commands.
When the layer 2 protocol has terminated, either as a result of an orderly shut down of the PDP or an error, the MT enters V.25ter command state and returns the NO CARRIER final result code. <cid> usage will be the same as in the +CGDATA command. The +CGDCONT, +CGQREQ, and so on, commands may then be used in the modem initialisation AT command string to set values for PDP type, APN, QoS, and so on.
If <cid> is omitted, the MT attempts to activate the context using one of the following:

- Any information provided by the TE during the PDP startup procedure, for example, the TE may provide a PDP type and/or PDP address to the MT
- A priori knowledge, for example, that the MT may implement only one PDP type
- Using the Empty PDP type (GSM 04.08). No PDP address or APN will be sent in this case and only one PDP context subscription record will be present in the HLR for this subscriber.

This command may be used in both normal and modem compatibility modes.
Note: The dial string conforms to the syntax specified in GSM 02.30.

Execution command: **ATD***<GPRS_SC_IP>[*<cid>]#

Parameters:

<GPRS_SC>: Digit string. A digit string (value='98') which identifies a request to use GPRS/Packet Domain with IP (PDP types IP and PPP).

<cid>: Digit string. Specifies a particular PDP context definition.

Unsolicited result codes

+CGEV GPRS event reporting

Description: This result code is enabled by using the **AT+CGEREP** command.

Possible unsolicited result codes: +CGEV: X, where X is shown below.

<X>	Description
REJECT <pdp_type>,<pdp_addr>	A network request for PDP context activation occurred when the MT was unable to report it to the terminal equipment with a **+CRING** unsolicited result code and was automatically rejected
NW REACT <pdp_type>,<pdp_addr>[,<cid>]	The network has forced a network reactivation. The <cid> that was used to reactivate the context is provided, if known to the MT
NW DEACT <pdp_type>,<pdp_addr>[,<cid>]	The network has forced a network deactivation. The <cid> that was used to deactivate the context is provided, if known to the MT
ME DEACT <pdp_type>,<pdp_addr>[,<cid>]	The mobile equipment has forced a network deactivation. The <cid> that was used to deactivate the context is provided, if known to the MT
NW DETACH	The network has forced a GPRS detach. This implies that all networks have been deactivated. These are not reported separately
ME DETACH	The mobile equipment has forced a GPRS detach. This implies that all MEs have been deactivated. These are not reported separately
NW CLASS <class>	The network has forced a change of phone class. The highest available class is reported
ME CLASS <class>	The mobile equipment has forced a change of phone class. The highest available class is reported

Parameters: See **AT+CGDCONT**.

+CGREG Network registration reporting

Description: This result code is enabled by using the **AT+CGREG** command.

Possible unsolicited result codes:
If AT+CGREG <n>='1'
+CGREG: <stat>
If AT+CGREG <n>='2'
+CGREG: <stat>[,<lac>,<ci>]

Parameters:

<stat>:

<stat>	Description
0	Not registered. ME is currently searching for an operator to register to
1	Registered, home network
2	Registered, but ME is searching for a new operator to register to
3	Registration denied

<stat>	Description
4	Unknown
5	Registered, roaming

<lac>:

<lac>	Description
String	Two byte location area code in hexadecimal format

<ci>:

<ci>	Description
String	Two byte cell ID in hexadecimal format

Ensemble S16: Phonebook

Commands

AT+CPBS Phonebook storage (ver. 3)

Description: Selects phonebook memory storage <storage>, which is used by other phonebook commands. If setting fails in a ME error, **+CME ERROR**: <err> is returned.
Read command returns currently selected memory and, when supported by manufacturer, number of used locations and total number of locations in the memory.
Test command returns supported storage as compound value.
Note: Each one of the defined profiles corresponds to one (and only one) list of allowed callers.

Set command: **AT+CPBS=**<storage>[,<password>]

Read command: **AT+CPBS?** Displays the current <name> setting.

Read command response: +CPBS: <storage>

Test command: **AT+CPBS=?** Shows if the command is supported.

Test command response: +CPBS: (list of supported <storage>s)

Parameters:

<storage>:

<storage>	Function
String type value	Storage
FD	SIM fix-dialling-phonebook

<storage>	Function
LD	SIM last-dialling-phonebook
ME	ME phonebook
MT	Combined ME and SIM phonebook. **Not supported**
SM	SIM phonebook
TA	TA phonebook
DC	ME dialled calls list
RC	ME received calls list
MC	ME missed calls list
MV	ME voice activated dialling list
GR	Group list. **Ericsson specific, not supported**
HP	Hierarchical phonebook. **Ericsson specific**
BC	Own business card. Protected by phone lock code. **Ericsson specific**
SM	SIM/UICC phonebook. If a SIM card is present or if a UICC with an active GSM application is present, the EF_{ADN} under $DF_{Telecom}$ is selected. If a UICC with an active USIM application is present, the global phonebook, $DF_{PHONEBOOK}$ under $DF_{Telecom}$ is selected. **Not supported**
EN	Emergency number. **Not supported**
CN	SIM (or ME) own numbers (MSISDNs) list (reading of this storage may be available through +CNUM also). When storing information in the SIM/UICC, if a SIM card is present or if a UICC with an active GSM application is present, the information in EF_{MSISDN} under $DF_{Telecom}$ is selected. If a UICC with an active USIM application is present, the information in EF_{MSISDN} under ADF_{USIM} is selected. **Not supported**
AP	Selected application phonebook. If a UICC with an active USIM application is present, the application phonebook, $DF_{PHONEBOOK}$ under ADF_{USIM} is selected. **Not supported**

<password>:

<password>	Function
string type	Value represents the password required when selecting password protected <storage>s, for instance, PIN-2 for storage FD

AT+CPBR Phonebook read (ver. 2)

Description: Returns phonebook entries in location number range <index1>...<index2> from the current phonebook memory storage selected by **AT+CPBS**. If <index2> is left out, only location <index1> is returned. Entry fields returned are location number <index*n*>, phone number <number> (of format <type>) and text <text> associated with the number.
When the Received Calls list (RC), the Missed Calls list (MC) or the Dialled Calls list (DC) is selected, the two additional fields <text_date> and <text_time> containing date and time is returned. In this case the <text> field containing text associated with the number has to be extracted from one of the phonebooks (SIM, ME or TA).
Note: Flags are used to indicate the contact field where the number is stored. See <contact_flag> below.

Set command: **AT+CPBR=**<index1>[,<index2>]

Set command response:
+CPBR:
<index1>,<number>,<type>,<text>[,<text_date>,<text_time>]**<CR><LF>**
+CPBR: <index2>,<number>,<type>,<text>[,<text_date>,<text_time>]

Test command: **AT+CPBR=?** Shows if the command is supported.

Test command response: +CPBR: (list of supported <index>s),<nlength>,<tlength>.

Parameters:

<index*n*>: Integer. Values in the range of location numbers of phonebook memory.

<number>: String. Phone number of format <type>.

<type>:

<type>	Description
128	Unknown numbering plan, national/international number unknown
129	ISDN/telephony numbering plan, national/ international number unknown
145	ISDN/telephony numbering plan, international number
161	ISDN/telephony numbering plan, national number
128–255	Other valid values, see GSM 04.08 section 10.5.4.7

<text>: String. Maximum length <tlength>. Character set as specified by **AT+CSCS**.

<nlength>: Integer. Maximum length of <number> field.

<tlength>: Integer. Maximum length of <text> field.

<contact_flag>:

<contact_flag>	Description
"/H"	Home. **Default value**
"/W"	Work
"/M"	Mobile
"/F"	Fax

<contact_flag>	Description
"/O"	Other

AT+CPBF Phonebook find (ver. 2)

Description: Returns phonebook entries from the current phonebook memory storage selected with **AT+CPBS**, whose alphanumeric field starts with the string <findtext>.
Entry fields returned are location number <index*n*>, phone number stored there <number> (of format <type>) and text <text> associated with the number.
Note:

- Searching in DC, RC or MC storage is not supported.
- When searching in the phone, the execution command returns phonebook entries (from the current phonebook memory storage selected with **AT+CPBS**) whose first/last name field start with string <findtext>. If <findtext> is given as "xyz", entries whose first name and/or last name field begins with "xyz" are displayed. If <findtext> is given as " xyz" (space followed by characters), only entries whose last name field begins with "xyz" are displayed.

Execution command: **AT+CPBF=**<findtext>

Execution command response:
+CPBF: <index1>,<number>,<type>,<text>[[...]<CR><LF>
+CBPF: <index2>,<number>,<type>,<text>]

Test command: **AT+CPBF=?** Shows if the command is supported.

Test command response: +CPBF: <nlength>,<tlength>.

Parameters:

<findtext>: String. Maximum length <tlength>. Character set as specified by **AT+CSCS**.

<index1>: Integer. Values in the range of location numbers of phonebook memory.

<number>: String. Phone number of format <type>.

<type>:

<type>	Description
128	Unknown numbering plan, national/international number unknown
129	ISDN/telephony numbering plan, national/ international number unknown
145	ISDN/telephony numbering plan, international number
161	ISDN/telephony numbering plan, national number
128–255	Valid values, see GSM 04.08 section 10.5.4.7

<text>: String type. Field of maximum length <tlength>. Character set as specified by command **AT+CSCS**.

<nlength>: Integer. Maximum length of field <number>

<tlength>: Integer. Maximum length of <findtext> field.

AT+CPBW Phonebook write (ver. 4)

Description: Writes phonebook entry in location number <index> in the current phonebook memory storage selected with **AT+CPBS**. Entry fields written are phone number <number> (in the format <type>) and text <text> associated with the number. If those fields are omitted the phonebook entry is deleted.

Note:
- If MV, BC or HP is the currently selected phonebook storage, **+CME ERROR**: <err> will be returned.
- DC, RC and MC storages are not supported.
- Flags may be used to indicate the contact field where the number should be stored. If no flag is used, the phone number will be stored as type “home”.
- A new phonebook entry will always use the first free position, independent of what <index> has been set to.
- The <type> parameter is ignored, the + sign in <number> determines if the number is international or national.
- If phone is the currently selected phonebook storage and AT+CPBW is used with an <index> that is already used by another number, the old number will be overwritten and removed from whatever contact it was previously a part of.

The name of the contact will be changed if the phone is the currently selected phonebook storage and the following criteria are met:
- AT+CPBW is used with an <index> that is part of a certain contact
- all other parameters except <text> are omitted
- the <text> parameter differs from the name of the contact in question

The <number> of the contact will be changed if:
- <number> parameter differs from the number of the contact in question
- <text> parameter is the same as the contacts <text> in question
- all the other parameters are omitted

Note: <type> is set to its default value, 129, if it is omitted.

Execution command: **AT+CPBW=**[<index>][,<number>[,<type>[,<text>]]]

Test command: **AT+CPBW=?** Shows if the command is supported.

Test command response: +CPBW: (list of supported <index>s),<nlength>,(list of supported <type>s), <tlength>.

Parameters:

<index>: Integer. Values in the range of location numbers of phonebook memory.

<number>: String. Phone number of format <type>.

<type>:

<type>	Description
128	Unknown numbering plan, national/international number unknown
129	ISDN/telephony numbering plan, national/ international number unknown. **Default value if dialling string does not include international access code character "+"**
145	ISDN/telephony numbering plan, international number. **Default value if dialling string includes international access code character "+"**
161	ISDN/telephony numbering plan, national number
128–255	Other valid values, see GSM 04.08 section 10.5.4.7

<text>: String. Maximum length <tlength>. Character set as specified by **AT+CSCS**.

Flag	Description
"H"	Home. **Default value**
"W"	Work
"O"	Other
"M"	Mobile
"F"	Fax

<nlength>: Integer. Maximum length of <number> field.

<tlength>: Integer. Maximum length of <text> field.

Use scenarios

Phonebook read

This scenario shows how reading from the phonebook is performed.

AT command	Response	Comment
`AT+CPBR=?`		Read index range and element lengths.
	`+CPBR: (1-99), 30,30` `OK`	Max 99 entries. Max number length equals 30
`AT+CPBR=2`		Read one entry at index 2
	`+CPBR: 2,"90510", 129,"Dieter"` `OK`	
`AT+CPBR=1,4`		Read entries from index 1 to 4. Only entries set are returned

AT command	Response	Comment
	+CPBR: 1,”12356”,129,”Klaus” +CPBR: 2,”90510”,129,”Dieter” +CPBR: 4,”54321”,129,”Helmut” OK	Index 1 Index 2 Index 4

Ensemble S18: GSM clock, date and alarm handling

Commands

AT+CCLK Clock (ver. 4)

Description: Sets the real-time clock in the phone.

Set command: **AT+CCLK=**<time>

Read command: **AT+CCLK?** Displays the current <time> setting.

Test command: **AT+CCLK=?** Shows if the command is supported.

Parameter:

<time>: String. “yy/MM/dd,hh:mm:ss±zz” or “yyyy/MM/dd,hh:mm:ss±zz”, where characters indicate year, month, day, hour, minutes, seconds and time zone (indicates the difference, expressed in quarters of an hour, between the local time and GMT. Range -47 to +48). For instance, 6th of May 1994, 22:10:00 GMT+2 hours equals to “94/05/06,22:10:00+08”

AT+CALA Alarm (ver. 3)

Description: Sets an alarm time in the ME. There can be an array of different alarms. Each alarm may be recurrent. If setting fails in an ME error, +CME ERROR: <err> is returned. When the alarm is timed out and executed, the unsolicited result code **+CALV**: <n> is always returned, even if the alarm is setup to be silent.
The alarm time is set in hours and minutes. Date, seconds and time zone are not used.

The read command returns the list of current active alarm settings in the ME.

The test command returns supported array index values, alarm types and maximum length of the text strings.

Note: Parameters that are not supported should be left empty between the comma signs.

Example: To set a recurrent alarm for alarm position 3 (Monday to Friday):

AT+CALA=”11:15”,3,1,,”1,2,3,4,5”

The <text> parameter is omitted because it is not supported.

Set command: **AT+CALA**=<time>[,<n>[,[<type>],[<text>],<recurr>[,<silent>]]]

Read command: **AT+CALA?**

Read command response

[+CALA: [<time1>,<n1>,<type1>,[<text1>],[<recurr1>],<silent1>]
[+CALA: <time2>,<n2>,<type2>,[<text2>],[<recurr2>],<silent2>]

Test command: **AT+CALA=?** Shows if the command is supported.

Test command response:

+CALA: (list of supported <n>s),(list of supported type>s),<tlength>,<rlength>,(list of supported <silent>s)

Parameters:

<time>: String. Only hours and minutes of format “HH:MM” are used. Date, time zone and seconds are not used.

<n>: Integer. Index identifying an alarm position. It is phone specific how many alarms there could be in the phone.

<type>: Integer. Value indicating the type of alarm.

<type>	Description
0	RECURRENT ALARM
1	TIME

<text>: Text to be displayed when the alarm time is reached.

<tlength>: Integer. Maximum length of the <text> parameter. Maximum value=20.

<recurr>:

<recurr>	Description
Format: "<1..7>[,<1..7>[…]]"	For setting an alarm for one or more days in the week. The digits 1 to 7 corresponds to the days of the week, Monday (1), Sunday (7). *Example*: The string "1,2,3,4,5" may be used for setting an alarm for all weekdays
Format: "0"	Sets the alarm for all days in the week

<rlength>: Integer. Maximum length of the <recurr> parameter. Maximum value=13.

<silent>: **Not supported**.

AT+CALD Alarm delete

Description: Removes an active alarm.

Execution command: **AT+CALD=**<n>

Test command: **AT+CALD=?** Shows if the command is supported.

Parameter:

<n>: Integer. Identifies an active alarm.

AT+CAPD Postpone or dismiss an alarm (ver. 2)

Description: Controls an active alarm by either postponing or dismissing it. If more than one active alarm occurs, this command influences the last activated alarm.

Execution command: **AT+CAPD=**[<sec>]

Test command: **AT+CAPD=?** Shows if the command is supported.

Test command response: +CAPD: (list of supported <sec>s)

Parameter:

<sec>:

<sec>	Description
0	Dismisses the alarm. **Default value**
540	Postpones the alarm (snooze) for 540 seconds (9 minutes). This is the **only** supported value

AT*EDST Ericsson daylight saving time

Description: Sets the daylight saving time hours.
Note: This command affects the MS clock set with the AT+CCLK command. To avoid confusion, it is recommended that the daylight saving time (DST) is set with this command before setting the actual local time with AT+CCLK.

Execution command: **AT*EDST=**<dst>

Read command: **AT*EDST?** Read current daylight saving time.

Read command response: *EDST: <dst>

Test command: **AT*EDST=?** Shows if the command is supported.

Test command response: *EDST: (list of supported <dst>s)

Parameter:

<dst>:

<dst>	Description
0	Standard time. **Default value**
1	Daylight saving time, +1 hour
2	Daylight saving time, +2 hours

Unsolicited result codes

+CALV Alarm event

Description: This unsolicited result code is returned when an alarm is activated. The alarm is set using **AT+CALA**.

Unsolicited result code: **+CALV:** <n>

Parameter:

<n>: Integer. Identifies an alarm event.

Use scenarios

Alarm functionality

AT command	Response	Comment
AT+CALA=?		Test if the command is supported
	+CALA: 1,,0,13,(0-13) OK	Only one alarm is supported, <type> is not supported
AT+CALA="14:25"		Set alarm time to 14:25
	OK	
AT+CALA?		Shows all active alarms

AT command	Response	Comment
	+CALA: “14:25”,1,,, OK	One alarm is set. The alarm index is “1”. The alarm has no text set – default is set. The alarm is not recurrent
AT+CALA=”06:10”,2,,,”1,2,3,4,5”		Set a new alarm for 06:10 on all weekdays
	OK	
AT+CALA?		
	+CALA: “14:25”,1,,, +CALA: “06:10”,2,,,”1,2,3,4,5” OK	
	+CALV: 1	Alarm event reported. Alarm is executed (at 06:10 every weekday)
AT+CAPD=540		Postpone the alarm for 9 minutes
	OK	
	+CALV: 1	9 minutes later, Alarm event report
AT+CAPD=0		Dismiss the alarm
	OK	

Ensemble S19: GSM subscriber information

Commands

AT+CIMI Request international mobile subscriber identity

Description: Causes the TA to return <IMSI>, identifying the individual SIM attached to the ME.

Execution command: **AT+CIMI**

Execution command response: +CIMI: <IMSI>

Test command: **AT+CIMI=?** Shows if the command is supported.

Parameter:

<IMSI>: String without double quotes. International Mobile Subscriber Identity.

Ensemble S20: Ericsson specific AT commands for GSM

Commands

AT*EPEE PIN event

Description: Requests the phone to inform when the PIN code has been entered and accepted. This command activates the unsolicited result code ***EPEV**.

Set command: **AT*EPEE=**<onoff>

Read command: **AT*EPEE?** Displays the current <onoff> setting.

Test command: **AT*EPEE=?** Shows if the command is supported.

Test command response: *EPEE: (list of supported <onoff>s)

Parameter:

<onoff>:

<onoff>	Description
0	Request for report on entered PIN is not activated (off). **Default value**
1	Request for report on entered PIN is activated (on)

AT*EAPS Active profile set

Description: Selects the active phone profile. The profiles may be renamed using **AT*EAPN**. The profile consists of the parameters and settings for the following commands:

AT command	Name	Ensemble
AT+CCFC	Call Forwarding Number and Conditions	S6
AT*EDIF	Divert Function and Reporting	S6

Set command: **AT*EAPS=**<index>

Read command: **AT*EAPS?** Displays the current <index> and <name_tagx> settings.

Test command: **AT*EAPS=?** Shows if the command is supported.

Test command response: *EAPS: (list of supported <index>s),<nlength>

Parameters:

<index>:

<index>	Description
Integer	Number of profiles

<name_tag*x*>: String. Profile name tag.

<nlength>: Integer. Maximum length of <name_tag*x*>.

AT*EAPN Active profile rename

Description: Sets a new name for the active profile. The number of profiles and the default names of the profiles are depending on the phone MMI.
Note: The name of the Normal profile (profile index 1) is read-only.

Set command: **AT*EAPN=**<name_tag>

Read command: **AT*EAPN?** Read the name of all the routing profiles in the phone.

Read command response:
*EAPN: <index1>,<name_tag1>**<CR><LF>**
[*EAPN: <index2>,<name_tag2>**<CR><LF>**
[...]]

Test command: **AT*EAPN=?** Shows if the command is supported.

Test command response:
*EAPN: <nlength>

Parameters:

<index>:

<index>	Description
Integer	Index of profile as defined in AT*EAPS

<name_tag>: String. Name tag for the profile, for example, Home, Office, Meeting

<nlength>: Integer. Maximum length of field <name_tag>.

AT*EBCA Battery and charging algorithm (ver. 4)

Description: Tests the charging algorithm in the phone and turn on/off unsolicited signal result codes (***EBCA**). When turned on the unsolicited result code is given once per second.
Voltage, current and capacity are physically limited, that is, they are platform dependent.
Note: For batteries without internal intelligence, some of the parameters listed below might not be available. In these cases the value “0” (zero) will be returned.
Note: Compared to earlier versions of this command, the name of the parameter <remcapacity> has been changed to <remcapacitypercent> because there is a new parameter showing the remaining capacity in mAh. This new parameter gets the same name as the old remaining capacity parameter, <remcapacity>.

Execution command: **AT*EBCA=**<onoff>

Execution command response: *EBCA: <vbat>,<dcio>,<icharge>,<iphone>,<tempbattery>,<tempphone>, <chargingmethod>,<chargestate>, <remainingcapacity>,<remcapacity>, <powerdissipation>,<noccycles>, <nosostimer>,<suspensioncause>

Read command: **AT*EBCA?** Displays the current <onoff> setting.

Test command: **AT*EBCA=?** Shows if the command is supported.

Test command response: *EBCA: (range of <onoff>)

Parameters:

<onoff>:

<onoff>	Description
0	Disable unsolicited result code ***EBCA**. **Default value**
1	Enable unsolicited result code ***EBCA**

<vbat> Battery voltage in number of mV. Range 0–65500.

<dcio>: Battery voltage from the charger in mV. Range 0–65500.

<icharge>: Current charge in mA. That is, a value of 1A is reported as 1000. Range 0–65500.

<iphone>: Phone current consumption in mA. That is, a value of 1A is reported as 1000. Range 0–65500.

<tempbattery> Signed integer. Battery temperature in °C, -20 to +70.

<tempphone>: Signed integer. Phone temperature in °C, -20 to +70.

<chargingmethod>: Integer.

<chargestate>:

<chargestate>	Description
0	Start
2	Charge – Charging the battery until it is fully charged
3	Await – The battery temperature is outside the limits for normal charging
4	Await, extended temperature – The battery temperature is outside the limits for maintenance charging
5	Await, safety timer – The safety timer has expired. Further charging is disabled for as long as the charger is connected
7	Handheld – No charger is attached
8	Charge completed – Maintaining the charge of a fully charged battery
15	Paused – Charging is paused by the user

<remcapacity>: Integer. Remaining capacity in mAh.

<remcapacitypercent>: Integer. Remaining capacity in percent. Range 0–100%.

<powerdissipation>: Integer. The FET power dissipation in mW.
<noccycles>: Integer. The number of completed charging cycles.
<nosostimer>: Integer. The number of seconds on the safety timer.
<suspensioncause>: Integer. The cause for suspension of charging.

AT*ELIB Ericsson list Bluetooth devices

Description: Lists the Bluetooth devices registered in the phone.

Execution command: **AT*ELIB**

Execution command response:
[*ELIB: <device1>**<CR><LF>**
[*ELIB: <device2>**<CR><LF>**
[...]]]

Test command: **AT*ELIB=?** Shows if the command is supported.

Parameters:

<devicex>:

<devicex>	**Description**
Character string	BT device name

Unsolicited result codes

*EPEV PIN code event

Description: This unsolicited result code is returned when a PIN code has been entered and accepted. The result code is activated using **AT*EPEE**.

Unsolicited result code: ***EPEV**

*EBCA Indication algorithm status (ver. 1)

Description: This unsolicited result code indicates the changes in status for the parameters of the charging algorithm. The result code is activated using **AT*EBCA**.

Unsolicited result code: *EBCA: <vbat>,<dcio>,<icharge>,<iphone>,<tempbattery>,<tempphone>, <chargingmethod>,<chargestate>, <remainingcapacity>,<remcapacity>, <powerdissipation>,<noccycles>, <nosostimer>,<suspensioncause>

Parameters: See **AT*EBCA**.

Use scenarios

Environment and profiles

AT command	Response	Comment
AT*EAPS?		Read the current profile
	*EAPS: 1,"Normal" OK	"Normal" is the current profile
AT*EAPS=3		Change profile to "Car"
	OK	
AT*EACS=4,1		An IR-device is now connected to the phone. The new accessory is added to the list of known environments
	OK	
AT*EAPS=1		Change profile to "Normal"
	OK	

Ensemble S26: Voice control

Commands

AT*EVAA Voice answer active (ver. 1)

Description: Activates and deactivates the voice answering function for the chosen type.
Note: If the Voice Answer function is activated and the associated voice tag has not yet been trained, the phone returns ERROR.

Set command: **AT*EVAA=**<type>,<onoff>

Read command: **AT*EVAA?**

Read command response:
EVAA: <type1>,<onoff1>[**<CR><LF>**
EVAA: <type2>,<onoff2>[**<CR><LF>**
...]]

Test command: **AT*EVAA=?** Shows if the command is supported.

Test command response: *EVAA: (list of supported <type>s),(list of supported <onoff>s)

Parameters:

<type>:

<type>	Description
0	Car handsfree
1	Portable handsfree
2	Speakerphone

<onoff>:

<onoff>	Description
0	Deactivate voice answering function. **Default value**
1	Activate voice answering function

AT*EMWS Magic word set

Description: Activates the Magic Word function. When activated, the voice recogniser continuously listens for the trained magic word. When the magic word is detected, the complete voice control functionality is activated.

Set command: **AT*EMWS=**<type>,<onoff>

Read command: **AT*EMWS?**

Read command response:
EMWS: <type1>,<onoff1>[**<CR><LF>**
EMWS: <type2>,<onoff2>[**<CR><LF>**
...]]

Test command: **AT*EMWS=?** Shows if the command is supported.

Test command response: *EMWS: (list of supported <type>s),(list of supported <onoff>s)

Parameters:

<type>:

<type>	Description
0	Car handsfree
1	Portable handsfree
2	Speakerphone

<onoff>:

<onoff>	Description
0	The magic word function is not activated. **Default value**
1	The magic word function is activated

Ensemble S27: OBEX

Commands

AT+CPROT Enter protocol mode

Description: Informs TA that TE wants to establish a peer-to-peer protocol <proto> or upper layer connection (indicated by the <lsap> settings) with the ME on the link from which the command was received. This command can be used in case the link between TE and ME does not provide such a mechanism itself.
If ME has succeeded in establishing a logical link between application protocols and external interface, it sends a CONNECT message to the TE. Otherwise, the NO CARRIER response is returned.
If the CONNECT response is received, TE can start sending <proto> or upper layer frames.
The connection always returns to <proto> mode when the protocol session is ended. When the ME receives a disconnect request from its peer entity, it processes it and sends a OK response to the TE indicating its capability for receiving new AT commands. Since <proto> or upper layers can be accessed in other ways, TA must have prior knowledge of the fact that connection is initiated with AT+CPROT command. This means that switch to <proto> mode must include some sort of notification to the protocol entity.
This command can be aborted by sending a <proto> or upper layer disconnection frame. In that case, ME returns to command mode by sending the OK response.

Set command: **AT+CPROT=**<proto>[,<version>[,<lsap1>[,...[<lsapN>]]]]

Test command: **AT+CPROT=?** Shows if the command is supported.

Test command response:

+CPROT: <proto1>[,(list of supported <version>s)[,(list of supported <lsap1>s)[,...[,(list of supported <lsapN>s)]]]][**<CR><LF>**
+CPROT: <proto2>[,(list of supported <version>s)[,(list of supported <lsap1>s)[,...[,(list of supported <lsapN>s)]]]]
[...]]]]]

Parameters:

<proto>:

<proto>	**Description**
0	OBEX

<version>:

<version>	**Description**
String	Version number of <proto>. **Note**: The total number of characters, including line terminators, in the information text may not exceed 16 characters

<version>	Description
...	Only value supported in 3G1

<lsap1>:

<lsap1>	Description
Integer type	Defines a level of service or application protocol on the top of <proto> layer. It may refer to services or protocols defined in other standards development organisations (SDOs)
8	IrMC level 1, 2 and 4 (Minimum, Access and Sync Levels) only. Implies unique index support

<lsap2>...<lsapN>:

<lsap2>...<lsapN>	Description
Integer type	In case <lsapN>, <lsapN+1> received in the +CPROT command identifies protocol layers, the protocol identified by N+1 will be on top of the protocol identified by N on a framework point of view

Ensemble S29: WAP browser

Commands

AT*EWDT WAP download timeout

Description: Sets the server response time used when downloading a WAP page.

Set command: **AT*EWDT=**<sec>

Read command: **AT*EWDT?** Displays the current <sec> setting.

Test command: **AT*EWDT=?** Shows if the command is supported.

Test command response: *EWDT: (list of supported <sec>s)

Parameter:

<sec>:

<sec>	Function
Integer	Number of seconds. Range: 15–300

AT*EWBA WAP bookmark add (ver. 2)

Description: Adds or deletes a bookmark in the list of bookmarks.
To add a bookmark the <option> parameter is set to 1. To delete a bookmark, <option> is set to 0.
If the <title> parameter is omitted the bookmark title is set to the first <ntitle> number of characters of the <URL>.
Note: The bookmarks added with this command are added to all WAP profiles in the phone.

Set command: **AT*EWBA**=<option>,<URL>[,<title>]

Read command: **AT*EWBA?** List number of bookmarks.

Read command response: *EWBA: <nBookmarks>

Test command: **AT*EWBA=?** Shows if the command is supported.

Test command response: *EWBA: (list of supported <options>),<nURL>,<ntitle>,<MaxBookmarks>

Parameters:

<option>:

<option>	Description
0	Deletes a bookmark
1	Adds a bookmark

<url>: String. The URL representing the bookmark.

<nurl>: Integer. Maximum length of the <url> parameter.

<title>: String. The title representing the bookmark. If omitted the bookmark title is set equal to the first <ntitle> number of characters of the <URL>.

<ntitle>: Integer. Maximum length of the <title> parameter.

AT*EWCT WAP connection timeout

Description: Sets timeout time used when connecting to a WAP supplier, that is, the time the WAP browser will wait for a CSD call to be established.

Read command: **AT*EWCT=**<sec>

Read command response: *EWCT: <sec>

Test command: **AT*EWCT=?** Shows if the command is supported.

Test command response: *EWCT: (list of supported <sec>s)

Parameter:

<sec>:

<sec>	Description
Integer	Number of seconds
60–300	Valid values

Use scenarios

WAP browser settings

AT command	Response	Comment
AT*EWDT=10		Set download timeout to 10 seconds
	OK	
AT*EWCT=10		Set connection timeout to 10 seconds
	OK	

Ensemble S34: Internet account commands

Common internet account command parameters

Some of the AT command parameters are shared between several S34 internet account AT commands. These parameters and their value definitions are specified in this section.

Parameter:

<index>: Index of an account within the specific bearer type

<index>	Description
0	This value is used in some commands to indicate that **all** accounts that match filter conditions set on another parameter should be affected by the command, for example, **all** accounts with a given bearer_type
1–255	Several accounts with the same index value may exist, but only one account within a given bearer type. The combination of bearer type and index forms the unique reference to one specific Internet account

<bearer_type>:

<bearer_type>	Description
0	Used to specify that accounts of all bearer types will be affected by the command
1	PS bearer. PS connection over UMTS/GPRS network

<bearer_type>	Description
2	CS bearer. NTCSD connection over UMTS/GSM network
3	Bluetooth bearer To connect with a remote Bluetooth LAN device
4	External Interface. Test value, for dial-in use

<name>:

<name>	Description
String type	“Friendly” name of the Internet account. Maximum 20 characters. Mainly used in the MMI of the mobile phone

<pref_serv>:

<pref_serv>	Description
0	Preferred service – Packet Switched only. An incoming call will be denied when running PS connection(s), if system resources are insufficient to serve both
1	Preferred service – Automatic. An incoming call will put GPRS or UMTS packet switched connection(s) on hold if system resources are insufficient to serve both. **Default Value**

<traffic_class>:

<traffic_class>	Description
0	Conversational
1	Streaming. For example, for Voice over IP and other QoS (delay) critical applications
2	Interactive. For example, for Video/Audio over IP and other QoS (delay+data volume) critical applications
3	Background. For example, for Chat and applications with some time requirements and low – medium data volume
4	Subscribed value. For non-time critical applications. **Default value**

<data_rate>:

<data_rate>	Description
1	9600 Kbps. GSM: 1 time slot * 9600
2	14400 Kbps. GSM: 1 time slot * 14400

<data_rate>	Description
3	19200 Kbps GSM: 2 time slots * 9600
4	28800 Kbps. GSM: 2 time slots * 14400 (or 3 TS*9600). **Default Value**
5	38400 Kbps. GSM: 4 time slots * 9600
6	43200 Kbps. GSM: 3 time slots * 14400
7	57600 Kbps. GSM: 4 time slots * 14400

<bt_addr>:

<bt_addr>	Description
Hex string	Bluetooth Address of remote Bluetooth enabled LAN Access device

<auth_prot>:

<auth_prot>	Description
00000–11111 or 0–7	**Default value**: 00111 (7) The authentication method is represented as a 5 bit long field in which each bit indicates a specific authentication method. The bitmask set, represents the authentication methods supported by the Internet account in question. (MSB)Bit4=1: MS-CHAPv2 Bit3=1: MS-CHAP Bit2=1: CHAP Bit1=1: PAP (LSB)Bit0=1: None For example, 00111 (=7), indicates support for CHAP, PAP and None. None indicates that it does not matter what authentication method is supported by the peer. The value 0 (all the bits set to 0) is not allowed. At least one bit has to be set to 1. **Note**: Leading zeroes do not need to be stated. For example, 111 is the same as 00111

<show>:

<show>	Description
0	Do not show
1	Show

S34 commands

AT*EIAC Internet account, create

Description:
Defines the general parameters of an Internet Account (IA).
When a new account is defined, it is assigned an index which is subsequently returned as an informational text response together with bearer type and name of account. When using the IA configuration command, the value of the index cannot be forced.
The other Internet Account commands cannot be used to create an account.
The other AT commands have to indicate the index value of an existing account in combination with what kind of bearer the parameters are set for. The exception is the Internet Account configuration commands where it is implicit what the bearer type is, for example, AT*EIAPSW – write PS bearer parameters. In this case only the index value is necessary.
Note: When a PDP Context is defined via an AT command, an Internet Account is automatically created with Packet Domain Service as the bearer and it gets an index value with a one-to-one mapping to the specified <cid> parameter value of the GPRS command. If an IA with that mapping to CID value already exists, the specific parameters of that IA is overwritten (also when IA parameters are originally specified for another bearer than PS). In the same way a PDP Context with default values is defined when an IA is created with Packet Domain Service as the bearer, using the AT*EIAC command. The <cid> of the PDP context will have a one-to-one mapping to the PS bearer IA index.
Note: If the user does not specify any bearer type, the command results in an error response.
Note: If the user does not specify a name of the account, an autogenerated name will be added to the account.
Note: The temporary or locked type of accounts are reserved for internal application use and is not listed in the read command and is not possible to create using this command.

Execution command:
Create account/define general parameters:
AT*EIAC=<bearer_type>[,[<name>]]

Response:
*EIAC: <index>,<name>

Read command:
Read the current general parameter settings:
AT*EIAC?

Read command response:
List of created Internet accounts:
*EIAC: <index>,<bearer_type>,<name>

Test command:
AT*EIAC=? Test if command is supported and show ranges of supported parameter values.

Test command response:
*EIAC: (list of supported <bearer_type>s),(“”)

Parameter:

<bearer_type>: Integer, 1–4

<bearer_type>	Description
1	PS bearer, that is, PS connection over UMTS/GPRS network
2	CS bearer, that is, NTCSD connection over UMTS/ GSM network
3	Bluetooth bearer, that is, connection with remote Bluetooth LAN device
4	External Interface. Test value for dial-in use

<name>: String. Each Internet Account has a “friendly” name.
Max 50 bytes, the actual number is dependent upon the character format chosen with +CSCS command.

<index>: 1–255.
Index of an account within the specific bearer type.
There might be several accounts that has the same index value, but only one account within a given bearer type. So it is the combination of bearer type and index that forms the unique reference to one specific Internet Account.

AT*EIAD Internet account configuration, delete

Description: Deletes one specific (or all) existing Internet account(s). Other Internet account commands or GPRS AT commands, cannot be used to delete an account.
Note: When <index>=0, <bearer_type> must also be =0 and vice versa.
Note: Both the <index> and <bearer_type> parameters must be specified, otherwise an error is returned.

Execution command: Delete account:
AT*EIAD=<index>,<bearer_type>

Test command: **AT*EIAD=?** Test if command is supported and show supported parameters

Test command response: Range of parameter values:
*EIAD: (0-255),(0-4)

Parameter:

<index>:

<index>	Description
0	Delete all existing <bearer_type> Internet Accounts
1–255	Delete Internet Account with index as specified and bearer type as specified. For ranges and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
0	All bearers

<bearer_type>	Description
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

AT*EIAW Internet account configuration, write general parameters

Description: Specifies the general parameters of the Internet Account.
Note: Both the <index> and <bearer_type> parameters must be specified, otherwise an error is returned.

Execution command: **AT*EIAW**=<index>,<bearer_type>[,[<name>]]

Test command: **AT*EIAW=?** Test if command is supported and show supported parameters.

Test command response: Range of parameter values:
*EIAW: (1-255),(1-4),(“”)

Parameter:

<index>:

<index>	Description
1–255	Write general parameters of <bearer_type> Internet. Account with index as specified

<bearer_type>:

<bearer_type>	Description
0	All bearers. Used together with Index=0, reading all Internet Accounts of all bearer types. **Default value**
1	PS bearer. PS connection over UMTS/GPRS network
2	CS bearer. NTCSD connection over UMTS/GSM network
3	Bluetooth bearer. To connect with a remote Bluetooth LAN device
4	External Interface. Test value, for dial-in use

<name>:

<name>	Description
String type	Each Internet Account has a “friendly” name. Used mostly by MMI application of phone. Max 20 characters

AT*EIAR Internet account configuration, read general parameters

Description: Reads the general parameters of one or several Internet accounts.
Note: When <index>=0, <bearer_type> must also be =0 and vice versa.

Execution command: **AT*EIAR**=[<index>][,[<bearer_type>]]

Response: *EIAR: <index>,<bearer_type>,<name>[:]

Test command: **AT*EIAR=?** Test if command is supported and show supported parameters.

Test command response: Range of parameter values:
*EIAR: (0-255),(0-4),(“”)

Parameter:

<index>:

<index>	**Description**
0	Read general parameters of all existing <bearer_type> Internet Accounts. **Default value**
1–255	Read general parameters of <bearer_type> Internet Account with given index

<bearer_type>:

<bearer_type>	**Description**
0	All bearers. Used together with Index=0, reading all Internet Accounts of all bearer types. **Default value**
1	PS bearer. PS connection over UMTS/GPRS network
2	CS bearer. NTCSD connection over UMTS/GSM network
3	Bluetooth bearer. To connect with a remote Bluetooth LAN device
4	External Interface. Test value, for dial-in use

<name>:

<name>	**Description**
String type	Each Internet Account has a “friendly” name. Used mostly by MMI application of phone. Max 20 characters

AT*EIAPSW Internet account configuration, write PS bearer parameters

Description: Specifies PS specific parameters of one (or all) PS bearer Internet Account. This command is used to define the most relevant Packet Switched (PS) data connection parameters.
Note: All PS parameters of this command except <pref_serv> can also be configured using the normal R'99 GPRS commands (see Ensemble S15 commands).
Note: For PS bearers, the <index> maps to the <ContextId> used in the GPRS commands.
Note: If the user does not specify the parameter <Index>, an error is returned.

Execution command: **AT*EIAPSW**=<index>[,[<pref_serv>][,[<apn>][,[<traffic_class>][,[<header_compr>][,[<data_compr>]]]]]]

Test command: **AT*EIAPSW=?** Test if command is supported and show supported parameters

Test command response: Range of parameter values:
*EIAPSW: (1-255),(0-1),(“”),(0-4),(0-1),(0-1)

Parameter:

<index>:

<index>	Description
1–255	Write packet switched bearer parameters of Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<pref_serv>:

<pref_serv>	Description
0	Preferred service – Packet Switched only. Means that an incoming call will be denied when running PS connection(s), if there are not system resources to serve both
1	Preferred service – Automatic. Means that an incoming call will put GPRS or UMTS packet switched connection(s) on hold if there are not system resources to serve both. **Default value**

<apn>:

<apn>	Description
String type	APN

<traffic_class>:

<traffic_class>	Description
0	Conversational. Gives best effort

<traffic_class>	Description
1	Streaming. For example, for Voice over IP and other QoS (delay) critical applications
2	Interactive. For example, for Video/Audio over IP and other QoS (delay+data volume) critical applications
3	Background. For example, for Chat and applications with some time requirements and low – medium data volume
4	Subscribed value. For non-time-critical applications. **Default value**

<header_compr>:

<header_compr>	Description
0	No. **Default value**
1	Yes. RFC 1144 (Van Jacobson) or RFC 2507 depending on UMTS or GSM network

<data_compr>:

<data_compr>	Description
0	No. **Default value**
1	Yes. Using default V42 bis parameters for dictionary size. Negotiates compression in both directions, Rx and Tx

AT*EIAPSR Internet account configuration, read PS bearer parameters

Description: Reads the wanted Packet Switched (PS) parameters from one (or all) primary PS bearer Internet account(s).

Execution command: **AT*EIAPSR**=[<index>]

Response: List of Internet accounts with PS parameters:
*EIAPSR: <index>,<pref_serv>,<apn>,<traffic_class>,<header_compr>, <data_compr>[:]

Test command: **AT*EIAPSR=?** Test if command is supported and show supported parameters.

Test command response: Range of parameter values:
*EIAPSR: (0-255)

Parameter:

<index>:

<index>	Description
0	Read packet switched bearer parameters of all existing PS bearer Internet Accounts. If no PS bearer IA exists, only OK is submitted. **Default value**
1–255	Read packet switched bearer parameters of Internet Account with Index as specified. For range and more details on <index>, see command **AT*EIAC**

<pref_serv>:

<pref_serv>	Description
0	Preferred service – Packet Switched only. Means that an incoming call will be denied when running PS connection(s), if there are not system resources to serve both
1	Preferred service – Automatic. Means that an incoming call will put GPRS or UMTS packet switched connection(s) on hold if there are not system resources to serve both. **Default value**

<apn>:

<apn>	Description
String type	APN

<traffic_class>:

<traffic_class>	Description
0	Conversational. Gives best effort
1	Streaming. For example, for Voice over IP and other QoS (delay) critical applications
2	Interactive. For example, for Video/Audio over IP and other QoS (delay+data volume) critical applications
3	Background. For example, for Chat and applications with some time requirements and low – medium data volume
4	Subscribed value. For non-time-critical applications. **Default value**

<header_compr>:

<header_compr>	Description
0	No. **Default value**
1	Yes. RFC 1144 (Van Jacobson) or RFC 2507 depending on UMTS or GSM network

<data_compr>:

<data_compr>	Description
0	No. **Default value**
1	Yes. Using default V42 bis parameters for dictionary size. Negotiates compression in both directions, Rx and Tx

AT*EIAPSSW Internet account configuration, write secondary PDP context parameters

Description:

Specifies secondary PDP context specific parameters of one (or all) secondary PS bearer Internet accounts.
If <index> in the command refers to a primary account, it will be converted to a secondary one. Incidentally, this is the way to create a secondary account when using EIA commands, otherwise AT+CGDSCONT need to be used. The command does not create an account if it does not already exist as a primary or secondary account.
This command is used to define the most relevant secondary PDP context connection parameters.
Note: All secondary PS parameters of this command can also be configured using the normal (Ensemble 15) GPRS commands.
Note: For PS bearers, the <index> maps to the <cid> and <p-index> maps to <p-cid> used in the GPRS commands.
Note: If the user does not specify an index, the command results in an error.

Execution command:

AT*EIAPSSW=
<index>,<p-index>[,<traffic_class>][,[header_compr>][,[data_compr>]]]]]

Test command:

AT*EIAPSSW=? Test if command is supported and show supported parameters

Test command response:

Range of Secondary PDP context parameters:
*EIAPSSW: (1-255),(1-255) (0-4),(0-1),(0-1)

Parameter:

<index>:

<index>	Description
1–255	Index of the Secondary PDP context for which Internet account parameters are written. <index> is a local context identification parameter

<p-index>:

<p-index>	Description
1–255	Index of the primary account that the secondary account is attached to. The primary account must exist, that is, it must have been previously defined by, for example, AT*EIAC. <p-index> can also be attached to accounts defined with AT+CGDCONT, but then the p-index is not immediately available, as the account has been referenced by <cid>

<traffic_class>:

<traffic_class>	Description
0	Conversational. Gives best effort
1	Streaming. For example, for Voice over IP and other QoS (delay) critical applications
2	Interactive. For example, for Video/Audio over IP and other QoS (delay+data volume) critical applications
3	Background. For example, for Chat and applications with some time requirements and low – medium data volume
4	Subscribed value. For non-time-critical applications. **Default value**

<header_compr>:

<header_compr>	Description
0	No. **Default value**
1	Yes. RFC 1144 (Van Jacobson) or RFC 2507 depending on UMTS or GSM network

<data_compr>:

<data_compr>	Description
0	No. **Default value**
1	Yes. Using default V42 bis parameters for dictionary size. Negotiates compression in both directions, Rx and Tx

AT*EIAPSSR Internet account configuration, read secondary PDP context parameters

Description: Reads the PDP context specific parameters of one (or all) secondary PS bearer Internet accounts.

Execution command: **AT*EIAPSSR**=[<index>]

Response: List of Internet accounts with Secondary PDP context parameters:
*EIAPSSR:
<index>,<p_index>,<traffic_class>,<header_compr>,<data_compr>[:]

Test command: **AT*EIAPSSR=?** Test if command is supported and show supported parameters.

Test command response: Range of parameter values:
*EIAPSSR: (0-255)

Parameter:

<index>:

<index>	Description
0	Read Secondary PDP context parameters of all existing PS bearer Internet Accounts. If no PS bearer IA exists, only OK is submitted. **Default value**
1–255	Read Secondary PDP context parameters of the Internet Account with index as specified

<traffic_class>:

<traffic_class>	Description
0	Conversational. Gives best effort
1	Streaming. For example, for Voice over IP and other QoS (delay) critical applications
2	Interactive. For example, for Video/Audio over IP and other QoS (delay+data volume) critical applications
3	Background. For example, for Chat and applications with some time requirements and low – medium data volume
4	Subscribed value. For non-time-critical applications. **Default value**

<header_compr>:

<header_compr>	Description
0	No. **Default value**
1	Yes. RFC 1144 (Van Jacobson) or RFC 2507 depending on UMTS or GSM network

<data_compr>:

<data_compr>	Description
0	No. **Default value**

<data_compr>	Description
1	Yes. Using default V42 bis parameters for dictionary size. Negotiates compression in both directions, Rx and Tx

AT*EIACSW Internet account configuration, write CSD bearer parameters

Description:

Defines the CS bearer parameters of one (or all) CS bearer Internet Account(s).
Note: CS bearer Internet accounts can only be used for internal applications to dial out to an ISP providing IP network access, for example, for WAP over CS. Normal CS “modem style” dial-up networking and plain CS modem connections are done by TE issuing the “legacy” AT commands. The parameters for such calls (RLP parameters, V42bis parameters, HSCSD parameters, and so on) are only stored in volatile memory, if not stored by using &W command.
Note: If the user does not specify the <index> parameter, the command results in an error response.

Execution command:

AT*EIACSW=<index>[,[<dialout_nbr>][,[<dial_type>][,[<data_rate>][,[<data_compr>]]]]]

Test command:

AT*EIACSW=? Test if command is supported and show supported parameters.

Test command response:

Range of CS parameters:
*EIACSW: (1-255),(“”),(0,1),(1-7),(0-1)

Parameter:

<index>:

<index>	Description
1–255	CSD bearer parameters of the specified Internet Account. For range and more details on <index>, see command **AT*EIAC**

<dialout_nbr>:

<dialout_nbr>	Description
String type	ISP phone number for internal dial out application to call

<dial_type>:

<dial_type>	Description
0	Analogue modem. **Default value**
1	ISDN modem

<data_rate>

<data_rate>	Description
1	9600 Kbps, GSM: 1 time slot * 9600
2	14400 Kbps, GSM: 1 time slot * 14400
3	19200 Kbps, GSM: 2 time slots * 9600
4	28800 Kbps, GSM: 2 time slots * 14400 (or 3 time slots * 9600). **Default value**
5	38400 Kbps, GSM: 4 time slots * 9600 **Not supported**
6	43200 Kbps, GSM: 3 time slots * 14400
7	57600 Kbps, GSM: 4 time slots * 14400

<data_compr>:

<data_compr>	Description
0	V42bis data compression off
1	V42bis data compression on. Using default V42 bis parameters for dictionary size. Negotiates compression in both directions, Rx and Tx. **Default value**

AT*EIACSR Internet account configuration, read CSD bearer parameters

Description: Reads the CS bearer parameters of one (or all) CS bearer Internet account(s).

Read command: **AT*EIACSR**=[<index>]

Response: List of Internet accounts with CSD parameters:
*EIACSR:
<index>,<dialout_nbr>,<dial_type>,<data_rate>,<data_compr>[:]

Test command: **AT*EIACSR=?** Test if command is supported and show supported parameters

Test command response: *EIACSR: (list of supported <index>s)

Parameter:

<index>: Integer, 0, 1-255

<index>	Description
0	Read CSD bearer parameters of all CSD bearer Internet Accounts. If no CS bearer IAs exists, only OK is submitted. **Default value**
1–255	Read CSD bearer parameters of the specified Internet Account. For range and more details on <index>, see command **AT*EIAC**

<dialout_nbr>:

<dialout_nbr>	Description
String type	ISP phone number for internal dial out application to call

<dial_type>:

<dial_type>	Description
0	Analogue modem. **Default value**
1	ISDN modem

<data_rate> 1–7

<data_rate>	Description
1	9600 Kbps, GSM: 1 time slot * 9600
2	14400 Kbps, GSM: 1 time slot * 14400
3	19200 Kbps, GSM: 2 time slots * 9600
4	28800 Kbps, GSM: 2 time slots * 14400 (or 3 time slots * 9600). **Default value**
5	38400 Kbps, GSM: 4 time slots * 9600 **Not supported**
6	43200 Kbps, GSM: 3 time slots * 14400
7	57600 Kbps, GSM: 4 time slots * 14400

<data_compr>:

<data_compr>	Description
0	No. **Default value**
1	Yes. Using default V42 bis parameters for dictionary size. Negotiates compression in both directions, Rx and Tx

AT*EIABTW Internet account configuration, write Bluetooth bearer parameters

Description: Defines the Bluetooth bearer parameters of one (or all) existing Bluetooth bearer Internet accounts.
Note: Bluetooth bearer Internet accounts can only be used for internal applications, to connect to a Bluetooth LAN access device. The PPP negotiations will bring up an IP connection for the internal applications to use.
Note: If <index> is not specified, the command results in an error response.

Execution command: **AT*EIABTW**=<index>[,[<bt_addr>],[<service>]]]

Test command: **AT*EIABTW=?** Test if command is supported and show supported parameters.

Test command response: *EIABTW: (list of supported <index>s),(“”),(list of supported <service>s)

Parameter:

<index>:

<index>	Description
1–255	Write CSD bearer parameters of the specified Internet Account. For range and more details on <index>, see command **AT*EIAC**

<bt_addr>:

<bt_addr>	Description
Hex string	Bluetooth device address, range: 48 bits. <bt_addr> is represented as 12 hexadecimal characters, for example, “0x000000AABBCC”. Any number will have the MSB -> LSB (from left to right)

<service>

<service>	Description
0	LAN Access profile. **Default value**
1	PAN profile: role PANU (PAN User)
2	PAN profile: role NAP (Network Access Point)
3	PAN profile: role GN (Group ad hoc Network)

AT*EIABTR Internet account configuration, read Bluetooth bearer parameters

Description: Reads the Bluetooth bearer parameters of one (or all) Bluetooth bearer Internet accounts.

Execution command: **AT*EIABTR**=[<index>]

Response: List of Internet accounts with Bluetooth parameters:
*EIABTR: <index>,<bt_addr>, <service>[:]

Test command: **AT*EIABTR=?** Test if command is supported and show supported parameters.

Test command response: Range of parameters:
*EIABTR: (0-255)

Parameter:

<index>:

<index>	Description
0	Read Bluetooth bearer parameters of all existing Bluetooth bearer Internet Accounts. If no Bluetooth bearer IAs exists, only OK is submitted. **Default value**

<index>	Description
1–255	Read Bluetooth bearer parameters of Internet Account with Index as specified For range and more details on <index>, see command **AT*EIAC**

<bt_addr>:

<bt_addr>	Description
Hex string	Bluetooth device address, range: 48 bits. <bt_addr> is represented as 12 hexadecimal characters, for example, “0x000000AABBCC”. Any number will have the MSB -> LSB (from left to right)

<service>

<service>	Description
0	LAN Access profile. **Default value**
1	PAN profile: role PANU (PAN User).
2	PAN profile: role NAP (Network Access Point)
3	PAN profile: role GN (Group ad hoc Network)

AT*EIAAUW Internet account configuration, write authentication parameters

Description: Specifies the authentication parameters of one (or all) existing Internet account(s).
Authentication parameters are used under any PPP negotiation as well as under PS network connection establishment (context activation).
Note: If the user does not specify both parameters <Index> and <bearer_type>, the command results in an error response.

Execution command: **AT*EIAAUW**=<index>,<bearer_type>[,[<userid>][,[<password>][,[<auth_prot>][,[<ask4pwd>]]]]]

Test command: **AT*EIAAUW=?** Test if command is supported and show supported parameters

Test command response: *EIAAUW: (list of supported <index>s),(list of supported <bearer_type>s),(“”),(“”),(list of supported <auth_prot>s),(list of supported <ask4pwd>s)

Parameter:

<index>: 1-255.
Write of <bearer_type> Internet Account with index as specified.
For range and more details on <index>, see under command **AT*EIAC**

<bearer_type>

<bearer_type>	Description
1	PS bearer, that is, PS connection over UMTS/GPRS network

<bearer_type>	Description
2	CS bearer, NTCSD connection over UMTS/GSM network
3	Bluetooth bearer, to connect with remote Bluetooth LAN device
4	External interface, test value, for dial-in use

<userid>:

<userid>	Description
String type	User Identification for access to the IP network. Max 50 8-bit characters

<password>:

<password>	Description
String type	Password for access to the IP network. Max 50 bytes, the actual number is dependent upon the character format chosen with +CSCS command

<auth_prot> The authentication method is organised as 5-bit-long field in which each bit indicates a specific authentication method. The bitmask set represents the Authentication methods supported by the Internet Account in question (<index>, <bearer>). **Default value** = 00111.
As an example, 00111, that is, bit2, bit1 and bit0 are set to 1, indicating support for CHAP, PAP and None.
None means that it does not matter what authentication method is supported by the peer.
The value 0 (all bits set to 0) is not allowed. At least one bit has to be set to 1.
Note: Leading zeroes do not need to be stated. For example, 111 is the same as 00111.

<auth_prot>	Description
bit0	None
bit1	PAP
bit2	CHAP
bit3	MS-CHAP
bit4	MS-CHAPv2

<ask4pwd>:

<ask4pwd>	Description
0	No. **Default value**
1	Yes. Triggers MMI application to ask user for password and user ID, instead of using the (eventually) stored user ID and password

AT*EIAAUR Internet account configuration, read authentication parameters

Description: Reads the authentication parameters of one (or all) existing Internet account(s).
Note: When <bearer_type>=0 it is necessary that <index>=0 and vice versa.

Execution command: **AT*EIAAUR**=[<index>,<bearer_type>]

Response: List of existing IA(s) authentication parameters:
*EIAAUR:
<index>,<bearer_type>,<userid>,<password>,<auth_prot>,<ask4pwd>[:]

Test command: **AT*EIAAUR=?** Test if command is supported and show supported parameters.

Test command response: *EIAAUR: (list of supported <index>s),(list of supported <bearer_type>s)

Parameter:

<index>:

<index>	Description
0	Read Authentication parameters of all existing Internet Accounts. **Default value**
1–255	Write parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see under command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
0	All bearers. **Default value**
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<userid>:

<userid>	Description
String type	User Identification for access to the IP network. Max 50 8-bit characters

<password>:

<password>	Description
String type	Password for access to the IP network. Max 50 bytes, the actual number depends on the character format chosen with AT+CSCS command

<auth_prot>

The authentication method is organised as 5 bit long field in which each bit indicates a specific authentication method. The bitmask set, represents the Authentication methods supported by the Internet Account in question (<index>, <bearer>). **Default value** = 00111.
As an example, 00111, that is, bit2, bit1 and bit0 are set to 1, indicating support for CHAP, PAP and None.
Note: None means that it does not matter what authentication method is supported by the peer.

<auth_prot>	Description
bit0	None
bit1	PAP
bit2	CHAP
bit3	MS-CHAP
bit4	MS-CHAPv2

<ask4pwd>:

<ask4pwd>	Description
0	No. **Default value**
1	Yes. Triggers MMI application to ask user for password and user ID, instead of using the (eventually) stored user ID and password

AT*EIALCPW Internet account configuration, write PPP parameters – LCP

Description:

Defines the PPP LCP parameters of an Internet account.
This command cannot be used to create an Internet account.
Note: If the user does not specify both parameters <index> and <bearer_type>, the command results in an error response.

Execution command:

AT*EIALCPW=<index>,<bearer_type>[,[<accm>][,[<mru>][,[<pfc>][,[<acfc>][,[<keep_alive>][,[<allowed_ncp>]]]]]]]

Test command:

AT*EIALCPW=? Test if command is supported and show supported parameters.

Test command response:

Range of LCP parameters:
*EIALCPW: (1-255),(1-4),(0-ffffffff),(0-1500),(0-16),(0-16),(0-1),(0-3)

Parameter:

<index>:

<index>	Description
1–255	Write LCP parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
1	PS bearer – PS connection over UMTS/GPRS network
2	CS bearer – NTCSD connection over UMTS/GSM network
3	Bluetooth bearer – connection with remote Bluetooth LAN device
4	External Interface – Test value, for dial-in use

<accm>:

<accm>	Description
0–FFFFFFFF	Asynchronous Control Character Map value, as a hexadecimal value. **Default value** = 0

<mru>:

<mru>	Description
1–1500	Max Receive Unit. The value specified is the recommended, but any MRU between this specified value and 1500 will be accepted. **Default value** = 1500

<pfc>:

<pfc>	Description
0	00 Protocol Field Compression negotiation might not be suggested. Protocol Field Compression negotiation might not be accepted
1	01 Protocol Field Compression negotiation might be suggested. Protocol Field Compression negotiation might not be accepted
2	10 Protocol Field Compression negotiation might not be suggested. Protocol Field Compression negotiation might be accepted
3	11 Protocol Field Compression negotiation might be suggested. Protocol Field Compression negotiation might be accepted. **Default value**

<acfc>:

<acfc>	Description
0	00 Address Control Field Compression negotiation might not be suggested. Address Control Field Compression negotiation might not be accepted
1	01 Address Control Field Compression negotiation might be suggested. Address Control Compression negotiation might not be accepted
2	10 Address Control Field Compression negotiation might not be suggested. Address Control Field Compression negotiation might be accepted
3	11 Address Control Field Compression negotiation might be suggested. Address Control Field Compression negotiation might be accepted. **Default value**

<keep_alive>:

<keep_alive>	Description
0	LCP keepalive messages should not be sent. **Default value**
1	LCP keepalive messages should be sent

<allowed_ncp>:

<allowed_ncp>	Description
0	**Not supported**
1	01 Network Control Protocol IPCP allowed. **Default value**
2	10 Network Control Protocol IPv6CP allowed
3	11 Both network control protocols allowed

AT*EIALCPR Internet account configuration, read PPP parameters – LCP

Description: Reads the PPP LCP parameters of one (or all) Internet account(s).
Note: When <index>=0 it is necessary that <bearer_type>=0 and vice versa.

Execution command: **AT*EIALCPR**=[<index>,<bearer_type>]

Response: List of Internet accounts with their LCP parameters:
*EIALCPR: <index>,<bearer_type>,<accm>,<mru>,<pfc>,<acfc>,<keep_alive>,<allowed_ncp>[:]

Test command: **AT*EIALCPR=?** Test if command is supported and show supported parameters

Test command response: Range of LCP parameters:
*EIALCPR: (0-255),(0-4)

Parameter:

<index>:

<index>	Description
0	Read LCP parameters of all <bearer_type> Internet Accounts. **Default value**
1–255	Read LCP parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
0	All bearers. Used together with Index=0, reading all Internet Accounts of all bearer types
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<accm>:

<accm>	Description
0–FFFFFFFF	Asynchronous Control Character Map value, as a hexadecimal value. **Default value**: 0

<mru>:

<mru>	Description
1–1500	Max Receive Unit. **Default: 1500** The specified value is the recommended, but any MRU between this value and 1500 will be accepted

<pfc>:

<pfc>	Description
0	00 Protocol Field Compression negotiation might not be suggested. Protocol Field Compression negotiation might not be accepted

<pfc>	Description
1	01 Protocol Field Compression negotiation might be suggested. Protocol Field Compression negotiation might not be accepted
2	10 Protocol Field Compression negotiation might not be suggested. Protocol Field Compression negotiation might be accepted
3	11 Protocol Field Compression negotiation might be suggested. Protocol Field Compression negotiation might be accepted. **Default value**

<acfc>:

<acfc>	Description
0	00 Address Control Field Compression negotiation might not be suggested. Address Control Field Compression negotiation might not be accepted
1	01 Address Control Field Compression negotiation might be suggested. Address Control Compression negotiation might not be accepted
2	10 Address Control Field Compression negotiation might not be suggested. Address Control Field Compression negotiation might be accepted
3	11 Address Control Field Compression negotiation might be suggested. Address Control Field Compression negotiation might be accepted. **Default value**

<keep_alive>:

<keep_alive>	Description
0	LCP keepalive messages should not be sent. **Default value**
1	LCP keepalive messages should be sent

<allowed_ncp>:

<allowed_ncp>	Description
0	**Not supported**
1	01 Network Control Protocol IPCP allowed. **Default value**
2	10 Network Control Protocol IPv6CP allowed
3	11 Both network control protocols allowed

AT*EIAIPCPW Internet account configuration, write PPP parameters – IPCP

Description:

Specifies the PPP IPCP parameters of one (or all) Internet accounts.
The command is used to specify the IP addresses to be used, both under PPP negotiations as well as under PS network connection establishment (context activation). If “0” values are given, necessary parameters are requested to be generated dynamically by the network to be attached to. If specific values are set, static IP addresses are to be requested.
Note: Values stored here are not the ones used when doing PS dial-up connection from external application (TE), since TE in this case has its own IP addresses to negotiate with the network (what TE sends is forwarded by the ME PPP proxy to network).
Note: If the user does not specify both parameters <index> and <bearer_type>, the command results in an error response.
Note: PPP parameters are relevant for all bearer types, but mostly for internal calls.

Execution command:

AT*EIAIPCPW=<index>,<bearer_type>[,[<ip_addr>][,[<prim_dns_addr>][,[<sec_dns_addr>][,[<header_compr>]]]]]

Test command:

AT*EIAIPCPW=? Test if command is supported and show supported parameters

Test command response:

Range of PPP IPCP parameters:
*EIAIPCPW: (1-255)(1-4),(““),(““),(““),(0-1)

Parameter:

<index>:

<index>	Description
1–255	Write IPCP parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
1	PS bearer, that is, PS connection over UMTS/GPRS network

<bearer_type>	Description
2	CS bearer, NTCSD connection over UMTS/GSM network
3	Bluetooth bearer, to connect with remote Bluetooth LAN device
4	External interface, test value, for dial-in use

<ip_addr>:

<ip_addr>	Description
String of format “a.b.c.d”	IPv4 host address. ME's own IP address. **Default** set to 0.0.0.0, which means request for dynamic IP address to be allocated by network upon connection

<prim_dns_addr>:

<prim_dns_addr>	Description
String of format “a.b.c.d”	IPv4 primary DNS server address. **Default** set to 0.0.0.0, which means request for dynamic IP address to be allocated by network upon connection

<sec_dns_addr>:

<sec_dns_addr>	Description
String of format “a.b.c.d”	IPv4 secondary DNS server address. **Default** set to 0.0.0.0, which means request for dynamic IP address to be allocated by network upon connection

<header_compr>:

<header_compr>	Description
0	Header compression off. **Default value**
1	Header compression on

AT*EIAIPCPR Internet account configuration, read PPP parameters – IPCP

Description: Reads the PPP IPCP parameters of one (or all) Internet account(s).
Note: When <index>=0 it is necessary that <bearer_type>=0 and vice versa.

Execution command: **AT*EIAIPCPR**=[<index>,<bearer_type>]

Response: List of Internet accounts with PPP IPCP parameters:
*EIAIPCPR: <index>,<bearer_type>,<ip_addr>,<prim_DNS_addr>, <sec_dns_addr>,<header_compr>[:]

Test command: **AT*EIAIPCPR=?** Test if command is supported and show supported parameters

Test command response:

Range of parameters:
*EIAIPCPR: (0-255),(0-4)

Parameter:

<index>:

<index>	Description
0	Read IPCP parameters of all <bearer_type> Internet Accounts. **Default value**
1–255	Read IPCP parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
0	All bearers. Used together with Index=0, reading all Internet Accounts of all bearer types. **Default value**
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<ip_addr>:

<ip_addr>	Description
String of format “a.b.c.d”	IPv4 host address. IP address of the ME. **Default** set to 0.0.0.0, which means request for dynamic IP address to be allocated by network upon connection

<prim_dns_addr>:

<prim_dns_addr>	Description
String of format “a.b.c.d”	IPv4 primary DNS server address. **Default** set to 0.0.0.0, which means request for dynamic IP address to be allocated by network upon connection

<sec_dns_addr>:

<sec_dns_addr>	Description
String of format “a.b.c.d”	IPv4 secondary DNS server address. **Default** set to 0.0.0.0, which means request for dynamic IP address to be allocated by network upon connection

<header_compr>:

<header_compr>	Description
0	Header compression off. **Default value**
1	Header compression on

AT*EIADNSV6W Internet account configuration, write DNS parameters – IPv6CP

Description: Specifies the DNS IPv6CP parameters of one (or all) Internet accounts. The command is used to specify the IP addresses to be used, both under PPP negotiations as well as under PS network connection establishment (context activation). If “0” values are given, necessary parameters are requested to be generated dynamically by the network to be attached to. If specific values are set, requests for static IP addresses is to be used.
Note: Values stored here are not the ones used when doing PS dial-up connection from external application (TE), since TE in this case has its own IP addresses to negotiate with the network (what TE sends is forwarded by the ME PPP proxy to network).
Note: If the user does not specify both parameters <index> and <bearer_type>, the command results in an error response.
Note: PPP parameters are relevant for all bearer types, but mostly for internal calls.
Note: For IPv6 addresses the notation :: can be used, but only once, for example, destination address FFFF:FFFF:0:0:0:0:0:0:0:0:0:0:0:1 can be written like FFFF:FFFF::1

Execution command: **AT*EIADNSV6W**=<index>,<bearer_type>[,<DNS_addr>]

Test command: **AT*EIADNSV6W=?** Test if command is supported and show supported parameters

Test command response: Range of PPP IPv6CP parameters:
*EIADNSV6W: (1-255),(1-4),(“”)

Parameter:

<index>:

<index>	Description
1–255	Write parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<DNS_addr>:

<DNS_addr>	Description
String of format “x:x:x:x:x:x:x:x”	IPv6 primary DNS server address. Each “x” is the hexadecimal value for one of the eight 16bit pieces of the address. **Default** set to ::, which means request for dynamic IP address to be allocated by network upon connection

AT*EIADNSV6R Internet account configuration, read DNS parameters – IPv6CP

Description: Reads the IPv6CP parameters of one (or all) Internet account(s).
Note: For IPv6 addresses the notation :: can be used, but only once, for example, destination address FFFF:FFFF:0:0:0:0:0:0:0:0:0:0:0:1 can be written like FFFF:FFFF::1

Execution command: **AT*EIADNSV6R**=[<index>,<bearer_type>]

Response: List of Internet accounts with PPP IPv6CP parameters:
*EIADNSV6R:<index>,<bearer_type>,<DNS_addr>[:]

Test command: **AT*EIADNSV6R=?** Test if command is supported and show supported parameters

Test command response: Range of parameters:
*EIADNSV6R: (0-255),(0-4)

Parameter:

<index>:

<index>	Description
0	Read IPv6CP parameters of all <bearer_type> Internet Accounts. **Default value**
1–255	Read IPv6CP parameters of <bearer_type> Internet Account with index as specified. For range and more details on <index>, see command **AT*EIAC**

<bearer_type>:

<bearer_type>	Description
0	All bearers. Used together with Index=0, reading all Internet Accounts of all bearer types
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<DNS_addr>:

<DNS_addr>	Description
String of format “x:x:x:x:x:x:x:x”	IPv6 primary DNS server address. Each “x” is the hexadecimal value for one of the eight 16bit pieces of the address. **Default** set to ::, which means request for dynamic IP address to be allocated by network upon connection

AT*EIARUTW Internet account configuration, write routing table parameters

Description: Specifies routing table parameters of one (or all) Internet accounts. All parameters, <IP-version>, <prefix>, <destination_address> and <nexthop_address>, must be given regardless if only one parameter is to be set.
Note: If the user does not specify any Index and bearer type, the command results in an error response.
Note: For IPv6 addresses the notation :: can be used, but only once, for example, destination address FFFF:FFFF:0:0:0:0:0:0:0:0:0:0:0:1 can be written like FFFF:FFFF::1

Execution command: **AT*EIARUTW**=<index>,<bearer_type>,<IP-version>,<prefix>,<destination_address>,<nexthop_address>

Test command: **AT*EIARUTW=?** Test if command is supported and show supported parameters

Test command response: Range of Routing table parameters:
*EIARUTW: (1-255),(1-4),(“”),(0-1),(0-32 / 0-128),(“”),(“”)

Parameter:

<index>:

<index>	Description
1–255	Write Routing table parameters of <bearer_type> Internet Account with index as specified

<bearer_type>:

<bearer_type>	Description
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<IP-version>:

<IP-version>	Description
“IP”	Internet Protocol (IETF STD 5)
“IPV6”	Internet Protocol, version 6 (IETF RFC 2460)

<prefix>:

<prefix>	Description
0–32	IPv4: Value used to create a IPv4 subnet mask. Indicates how many bits should be set to 1 in the mask. 0 indicates default route. A value of 24 will result in the following subnet mask: 255.255.255.0
0–128	IPv6: Value used to create a IPv6 subnet mask. Indicates how many bits should be set to 1 in the mask. 0 indicates default route. A value of 24 will result in the following subnet mask: FFFF:FF00::0

<destination_address>:

<destination_ address>	Description
String of format “a.b.c.d”	IPv4 address of the destination host. An entry with a value of 0.0.0.0 is considered as the default route
“x:x:x:x:x:x:x:x”	IPv6 address of the destination host. Each “x” is the hexadecimal value for one of the eight 16bit pieces of the address. An entry with a value of :: is considered as the default route

<nexthop_address>:

<nexthop_address>	Description
String of format “a1.a2.a3.a4”	IPv4 address of the adjacent host or router to which the packet should be sent next. Not utilised for point-to-point connections
String of format “x:x:x:x:x:x:x:x”	IPv6 address of the adjacent host or router to which the packet should be sent next. Each “x” is the hexadecimal value for one of the eight 16bit pieces of the address. Not utilised for point-to-point connections

AT*EIARUTD Internet account configuration, delete routing table parameters

Description: Deletes the routing table parameters of one Internet Account. All parameters, <IP-version>, <prefix>, <destination_address> and <nexthop_address>, must be given to delete one row in the specified routing table.
Note: For IPv6 addresses, the notation :: can be used, but only once. For example, destination address FFFF:FFFF:0:0:0:0:0:0:0:0:0:0:0:1 can be written like FFFF:FFFF::1
Note: If the user specify <index> and <bearer_type> parameters only, all defined IPv4 and IPv6 routes are removed.
Note: If the user does not specify both parameters <index> and <bearer_type>, the command results in an error response.

Execution command: **AT*EIARUTD=**<index>,<bearer_type>[,<IP-version>,<prefix>,<destination_address>,<nexthop_address>]

Test command: **AT*EIARUTD=?** Show if the command is supported.

Test command response: Range of Routing table parameters:
*EIARUTD: (1-255),(1-4),(“”),(0-1),(0-32 / 0-128),(“”),(“”)

Parameter:

<index>:

<index>	Description
1–255	Delete parameters of <bearer_type> Internet Account with index as specified

<bearer_type>:

<bearer_type>	Description
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<prefix>:

<prefix>	Description
0–32	IPv4: Value used to create a IPv4 subnet mask. Indicates how many bits should be set to 1 in the mask. 0 indicates default route. A value of 24 will result in the following subnet mask: 255.255.255.0
0–128	IPv6: Value used to create a IPv6 subnet mask. Indicates how many bits should be set to 1 in the mask. 0 indicates default route. A value of 24 will result in the following subnet mask: FFFF:FF00::0

<ip_version>:

<ip_version>	Description
String format “IP”	Internet Protocol (IETF STD 5)
String format “IPV6”	Internet Protocol, version 6 (IETF RFC 2460)

<destination_address>:

<destination_address>	Description
String of format “a.b.c.d”	IPv4 address of the destination host. An entry with a value of 0.0.0.0 is considered as the default route
“x:x:x:x:x:x:x:x”	IPv6 address of the destination host. Each “x” is the hexadecimal values for one of the eight 16bit pieces of the address. An entry with a value of :: is considered as the default route

<nexthop_address>:

<nexthop_address>	Description
String of format “a1.a2.a3.a4”	IPv4 address of the adjacent host or router to which the packet should be sent next. Not utilised for point-to-point connections
String of format “x:x:x:x:x:x:x:x”	IPv6 address of the adjacent host or router to which the packet should be sent next. Each “x” is the hexadecimal values for one of the eight 16bit pieces of the address. Not utilised for point-to-point connections

AT*EIARUTR Internet account configuration, read routing table parameters

Description: Reads the routing table parameters of one Internet account.

Execution command: **AT*EIARUTR**=[<index>,<bearer_type>]

Response: List of IAs with their Routing table parameters:
*EIARUTR: <index>,<bearer_type>,<IP-version>,<prefix>,
<destination_address>,<nexthop_address>[:]

Test command: **AT*EIARUTR=?** Test if command is supported and show supported parameters

Test command response: Range of Routing table parameters:
*EIARUTR:(1-255),(0-4),(“”),(0-32/0-128),(“”),(“”)

Parameter:

<index>:

<index>	**Description**
0	Read Routing table parameters of all <bearer_type> Internet accounts. **Default value**
1–255	Read Routing table parameters of <bearer_type> Internet account with index as specified

<bearer_type>:

<bearer_type>	**Description**
0	All bearers. Used together with Index=0, reading all Internet Accounts of all bearer types
1–4	Specific bearer. For more information on bearer types see <bearer_type> parameter definitions under command **AT*EIAC**

<IP-version>:

<IP-version>	**Description**
“IP”	Internet Protocol (IETF STD 5)
“IPV6”	Internet Protocol, version 6 (IETF RFC 2460)

<prefix>:

<prefix>	**Description**
0–32	IPv4: Value used to create a IPv4 subnet mask. Indicates how many bits should be set to 1 in the mask. 0 indicates default route. A value of 24 will result in the following subnet mask: 255.255.255.0
0–128	IPv6: Value used to create a IPv6 subnet mask. Indicates how many bits should be set to 1 in the mask. 0 indicates default route. A value of 24 will result in the following subnet mask: FFFF:FF00::0

<destination_address>:

<destination_ address>	Description
String of format “a.b.c.d”	IPv4 address of the destination host. An entry with a value of 0.0.0.0 is considered as the default route
“x:x:x:x:x:x:x:x”	IPv6 address of the destination host. Each “x” is the hexadecimal values for one of the eight 16bit pieces of the address. An entry with a value of :: is considered as the default route

<nexthop_address>:

<nexthop_address>	Description
String of format “a1.a2.a3.a4”	IPv4 address of the adjacent host or router to which the packet should be sent next. Not utilised for point-to-point connections
String of format “x:x:x:x:x:x:x:x”	IPv6 address of the adjacent host or router to which the packet should be sent next. Each “x” is the hexadecimal values for one of the eight 16bit pieces of the address. Not utilised for point-to-point connections

Ensemble S35: Sony Ericsson commands

Commands

AT*SEACC Accessory class report

Description: Informs the MS about the attachment of an accessory at its downstream port. The command reports the measured value according to the resistive identification mechanism. The value is the 8 bit number produced by the A/D converter. It is the responsibility of the MS to interpret this value as an accessory category.

Execution command: **AT*SEACC=**<rid_value>
Report resistive ID value.

Test command: **AT*SEACC=?**
Test if command is supported and show supported parameters.

Test command response: *SEACC:(range of supported <rid_value>s)

Parameter:

<rid_value>:

<rid_value>	Description
0–255	The resistive ID value measured by the A/D converter in the accessory

AT*SEACID Accessory identification

Description: Replaces the AT*EACS command and is used by an accessory to inform the phone about its exact identity. This ID is to be used by the phone to activate specific functionality that is required by the accessory.
Note: The audio settings in the phone are made according to the audio identity transmitted from the accessory with AT*SEAUDIO
Note: This command may **not** be used by Bluetooth accessories. The <acc_id> list contain bluetooth accessories only because AT*SEACID2 is using the same list.

Execution command: **AT*SEACID=**<acc_id>

Test command: **AT*SEACID=?** Test if command is supported and show supported parameters

Test command response: *SEACID: (Range of accessory ID)

Parameter:

<acc_id>:

<acc_id>	Description
0–2^32	The unique identity of the accessory
1000–1999	Camera flashes
2000–2999	Input device
3000–3999	Gaming accessories
4000–4999	Imaging
5000–5999	VHF – Vehicle Handsfree
6000–6999	PHF – Portable Handsfree
7000–7999	Gadget
8000–8999	BVHF
9000–9999	BVHF – Budget Vehicle Handsfree (without accessory mic)
10000–10999	BT – Bluetooth Headset or Handsfree
11000–11999	BTC – Bluetooth Handsfree for installation in car
12000–12999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
13000–13999	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile

<acc_id>	Description
14000–14999	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in car
15000–15999	DSS – Desk Speaker Stand
16000–16999	BDSS – Budget Desk Speaker Stand
17000–17999	LO – Line out accessory
18000–18999	LI – Line in accessory

AT*SEACID2 Accessory identification (Bluetooth)

Description: Replaces the AT*EACS command and is used by an accessory to inform the phone about its exact identity. This ID is to be used by the phone to activate specific functionality that is required by the accessory.

Note: The audio settings in the phone are made according to the audio identity transmitted from the accessory with AT*SEAUDIO.

Note: This command has to be used by Bluetooth accessories, using SEACID would lead to the accessory identifying itself twice

Execution command: **AT*SEACID2=**<acc_id>

Test command: **AT*SEACID2=?** Test if command is supported and show supported parameters

Test command response: *SEACID2: (Range of accessory ID)

Parameter:

<acc_id>:

<acc_id>	Description
0–2^32	The unique identity of the accessory
1000–1999	Camera flashes
2000–2999	Input device
3000–3999	Gaming accessories
4000–4999	Imaging
5000–5999	VHF – Vehicle Handsfree
6000–6999	PHF – Portable Handsfree
7000–7999	Gadget
8000–8999	BVHF
9000–9999	BVHF – Budget Vehicle Hands Free (without accessory mic)
10000–10999	BT – Bluetooth headset or handsfree
11000–11999	BTC – Bluetooth handsfree for installation in Car
12000–12999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)

<acc_id>	Description
13000–13999	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
14000–14999	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car
15000–15999	DSS – Desk Speaker Stand
16000–16999	BDSS – Budget Desk Speaker Stand
17000–17999	LO – Line out accessory
18000–18999	LI – Line in accessory

AT*SEAUDIO Accessory class report

Description: Informs the phone about the general audio class and the unique audio ID of an accessory. If there is a specific audio settings container available for the unique id, the phone will use that configuration. If not, the configuration for the general audio class will be used.
When the accessory identifies itself acoustically with AT*SEAUDIO the phone will respond with a result code indicating what audio class and what unique audio ID have been used when configuring audio. If no specific audio configuration was available for the unique audio ID used by the accessory this will be indicated by setting <unique_audio_id>=0 in the result code. If the accessory for some reason does not have audio capabilities it will send AT*SEAUDIO=0,0.

Execution command: **AT*SEAUDIO=**<audio_class>,<unique_audio_id>

Execution command response *SEAUDIO:<audio_class>,<unique_audio_id>

Read command: **AT*SEAUDIO?** Read current setting

Test command: **AT*SEAUDIO=?** Test if command is supported and show supported parameters.

Test command response: *SEAUDIO:(range of supported <audio_class>s),(range of supported <unique_audio_id>s)

Parameters:

<audio_class>:

<audio_class>	Description
0–255	The default audio class of the accessory
0	The accessory has no audio capabilities
1	PHF – Portable Handsfree
2	VHF – Vehicle Handsfree
3	BVHF – Budget Vehicle Handsfree (without accessory mic)
4	BT – Bluetooth headset or handsfree
5	BTC – Bluetooth handsfree for installation in Car

<audio_class>	Description
6	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
7	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car
9	DSS – Desk Speaker Stand
10	BDSS – Budget Desk Speaker Stand
11	LO – Line out accessory
12	LI – Line inaccessory

<audio_id>:

<audio_id>	Description
0–2^32	The unique audio identity of the accessory
0	Used in response codes to indicate that the terminal has not applied a specific audio configuration for the unique audio ID of the accessory
1–999	**Note**: Reserved for internal use in the telephone
1	PHF1 (**Note**: Not to be used by any accessory!)
2	PHF2 (**Note**: Not to be used by any accessory!)
3	PHF3 (**Note**: Not to be used by any accessory!)
4	PHF4 (**Note**: Not to be used by any accessory!)
5	Line in (**Note**: Not to be used by any accessory!)
6	Line out (**Note**: Not to be used by any accessory!)
7	BT Headset (**Note**: Not to be used by any accessory!)
8	BT Handsfree (**Note**: Not to be used by any accessory!)
1000–1999	PHF – Portable handsfrees
2000–2999	VHF – Vehicle handsfrees
3000–3999	BVHF – Budget Vehicle Handsfree (without accessory mic)
4000–4999	BT – Bluetooth headset or handsfree
5000–5999	BTC – Bluetooth handsfree for installation in car
6000–6999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
7000–7999	BTBTL – A Bluetooth audio device supporting the Headset or the Handsfree profile and the Advanced Audio Distribution Profile

<audio_id>	Description
8000–8999	BTBTLC – A Bluetooth audio device supporting the Headset or the Handsfree profile and the Advanced Audio Distribution Profile for installation in car
9000–9999	DSS – Desk Speaker Stand
10000–10999	BDSS – Budget Desk Speaker Stand
11000–11999	LO – Line out accessory
12000–12999	LI – Line in accessory

AT*SECHA Charging control

Description: This command is used by the accessory to tell the phone to pause the charging of the battery. During the pause the accessory will perform the measurement according to the AID mechanism to identify the new accessory attached.
If charging is not switched on again the charging pause will end and the charging resume anyway.

Execution command: **AT*SECHA=**<time>

Test command: **AT*SECHA=?** Test if command is supported and show supported parameters

Test command response: *SECHA: (range of supported <time>s)

Parameter:

<time>:

<time>	Charging mode
0	Charging resumed. **Default value**
1–5000	Charging will be paused for the number of miliseconds stated in this parameter

AT*SELOG SE read log

Description: Reads the customisation log file placed in tpa/preset/log.txt
The result is encoded into a hexadecimal representation.

Execution command: **AT*SELOG**

Command responses: *SELOG:[<data>]
SELOG:<ecode>,<nr_of_bytes>

Test command: **AT*SELOG=?** Shows if the command is supported

Parameters:

<data>:

<data>	Description
Character string	Each byte of data is encoded into a hexadecimal number represented by two characters

<ecode>:

<ecode>	Description
0–63	Valid values
0	No error
1	Operation not permitted
2	No such file or directory
20	Not a directory
28	Not enough space

<nr_of_bytes>:

<nr_of_bytes>	Description
Integer number	The number of bytes that was read, that is, the size of the file

AT*SEPING SE ping command

Description: Informs accessories if the AT command server in the “application” part of the phone is up and running.
Note: There are two AT command servers in the phone: one in the platform and one in the application part of the phone, this command regards the AT command server in the application part)
Note: OK response does not necessarily mean that all AT commands are supported. For instance, if the phone is in charging only mode or waiting for PIN verification, only a limited set of AT commands are accepted.

Execution command: **AT*SEPING**

Test command: **AT*SEPING=?** Test if command is supported.

AT*SEAULS SE audio line status

Description: Provides information about the audio line status and audio type. With the set command, it is possible to enable the unsolicted ***SEAULSI**.
*SEAULSI is sent each time either the audio channel switches accessories or when some audio starts or stops playing. Audio line status = 1 is sent to the accessory that has the audio authority and 0 to the rest. The audio type value is sent to all accessories that subscribes on *SEAULSI.

Set command: **AT*SEAULS =** <activation>

Set command response: *SEAULS:<activation>,<audio_line>,<audio_type>

Read command: **AT*SEAULS?**

Read command response: *SEAULS:<activation>,<audio_line>,<audio_type>

Test command: **AT*SEAULS=?** Test if command is supported and show supported parameters

Test command response: *SEAULS: (list of supported <activation>s)

Parameter:

<activation>:

<activation>	Description
0	Disable audio status indication (*SEAULSI)
1	Enable audio status indication (*SEAULSI)

<audio_line>:

<audio_line>	Description
0	Audio line inactive
1	Audio line active

<audio_type>:

<audio_type>	Description
0	No audio
1	Speech
2	Media

AT*SEFUNC SE functionality status (ver. 2)

Description: Gets the <mode> and different states of the ME. Only one <mode> may be active, but <states> are bit flags, so combinations of them are possible. The unsolicited ***SEFUNCI** is triggered when a <mode> is changed or when a <states> value is changed and will be sent to subscribing accessories.
The bits in <states> represents activation status, not if the functionality is available in phone or not.

Set command: **AT*SEFUNC =** <activation>

Set command response: *SEFUNC: <activation>, <mode>, <states>

Read command: **AT*SEFUNC?**

Read command response: *SEFUNC: <activation>, <mode>, <states>

Test command: **AT*SEFUNC=?** Test if command is supported and show supported parameters

Test command response: *SEFUNC: (list of supported <activation>), (list of supported <mode>),(list of supported <states>)

Parameter:

<activation>:

<activation>	Description
0	Disable functional status indication (*SEFUNCI)
1	Enable functional status indication (*SEFUNCI)

<mode>:

<mode>	Description
0	Shutdown mode
1	Charging only mode
2	Normal mode

<states>:

<states>	Description	Comments
1	Waiting for PIN	Phone is waiting for pin1 confirmation. **Note**: If pin1 is disabled this flag will never be set
2	Waiting for phone lock	Phone is waiting for phone lock confirmation. **Note**: If phone lock is disabled this flag will never be set
4	Bluetooth	Bluetooth is active
8	Mobile Radio	The GSM radio and/or the WCDMA radio is active
16	WLAN	WLAN is active
32	FM Radio	The FM radio is active. **Not supported**
...	...	...

AT*SEFIN SE flash Information

Description: Informs the ME about:
- Capacity in terms of the standardised photometric term Guide Number.
- Min/max exposure is the minumum and maximum value that the flash can handle. These are the values that are used by AT*SEFEXP.
- Color Temperature of the flash light given as hundreds of Kelvin.

The ME uses this input to make appropriate settings in the camera module.

Execution command: **AT*SEFIN=**<guide_number>,<min_exposure>,<max_exposure>,<color_temp>

Test command: **AT*SEFIN=?** Test if command is supported and show supported parameters

Test command response: *SEFIN=(list of supported <guide_number>s),(list of supported <min_exposure>s),(list of supported <max_exposure>s),(list of supported <color_temp>s)

Parameter:

<guide_number>:

<guide_number>	Description
0–255	The Guide Number of the flash @ ISO100

<min_exposure>:

<min_exposure>	Description
0–255	Minimum value of exposure that the flash can handle

<max_exposure>:

<max_exposure>	Description
0–255	Maximum value of exposure that the flash can handle

<color_temp>:

<color_temp>	Description
0–255	The Color Temperature of the flash in hundreds of Kelvin, for example, 6500K gives color_temp=65

AT*SEFEXP Flash auto exposure setting from ME

Description: Requests information from the ME about what level of flash would make the picture brighter or darker by adjusting its auto exposure control circuit. The flash uses this input to make appropriate settings in its auto exposure control circuit.
Note: The unsolicited result code ***SEFEXP** will be sent once when the AT*SEFEXP is issued (and only if AT*SEFEXP is called when the flash is plugged in).

Set command: **AT*SEFEXP**

AT*SEMOD Camera mode indicator to the flash

Description: Requests information from the ME if it is in a mode were the flash should be charged up and ready. The flash uses this input to start or abort charging its internal capacitor.
Note: With this command it is only possible to turn on request for unsolicited result codes.

Execution command: **AT*SEMOD**

Response: Unsolicited result code. *SEMOD=<action>

Parameter:

<action>:

<action>	Description
0	ME is not in a mode were the flash has to be ready to fire. The flash disables its charging
1	ME is in a mode were the flash has to be ready to fire. The flash starts to charge if not already charged

AT*SEREDI Red eye reduction indicator to the flash

Description: Performs the following:
- Informs the ME that it supports red eye reduction with a certain time out period.
- Requests information from the ME when red eye reduction should start (as un unsolicited event).

Set command: **AT*SEREDI=**<time_out>

Test command: **AT*SEREDI=?**
Test if command is supported and show supported parameters

Test command response: *SEREDI: (list of supported <time_out>s)

Unsolicited result code: ***SEREDI**

Parameter:

<time_out>:

<time_out>	Description
0–65535	The time, in milliseconds, between the red eye reduction flashes and the picture taking.

AT*SEFRY Ready indicator to the ME

Description: Informs the ME if the camera flash is ready to fire or not.

Command: **AT*SEFRY=**<action>
The flash informs the ME whether it is ready to fire or not.

Test command: **AT*SEFRY=?**
Test if command is supported and show supported parameters.

Test command response: *SEFRY=(list of supported <action>s)

Parameter:

<action>:

<action>	Description
0	The flash is not ready to fire by means of a strobe signal through the system connector
1	The flash is ready to fire on a strobe signal through the system connector

AT*SEAUP Sony Ericsson audio parameters

Description: Sends specific audio settings to configure the MS for a specific audio device type. The accessory sends this command with all audio parameters when ME connects to a BT audio device. It also sends it when it is connected (when necessary), in this case the accessory only sends the parameters it wishes to update.

Execution command:

AT*SEAUP=<unique_audio_id>,
<audio_class>,
[<ASC_Interface_TX_Impendence>],
[<ASC_Interface_RX_Impendence>],
[<ASC_Interface_Line_in_capable>],
[<ASC_Call_TXNOM>],
[<ASC_Call_TXGAIN>],
[<ASC_Call_SLRTarget>],
[<ASC_Call_TXNC>],
[<ASC_Call_TXFilter8>],
[<ASC_Call_RXNOM>],
[<ASC_Call_RXGAIN>],
[<ASC_Call_RLRTarget>],
[<ASC_Call_RXMAS>],
[<ASC_Call_RXFilter8>],
[<ASC_Call_Sidetone>],
[<ASC_Call_EC>],
[<ASC_Leisure_TXClip>],
[<ASC_Leisure_TXNC>],
[<ASC_Leisure_TXFilter48>],
[<ASC_Leisure_RXMAS>],
[<ASC_Leisure_RXNG>],
[<ASC_Leisure_RXFilter441>],
[<ASC_Leisure_RXFilter48>]

Response: Unsolicited result code: *SEAUPI: <status>

Test command: **AT*SEAUP=?** Test if command is supported

Parameter:

<unique_audio_id>:

<unique_audio_id>	Description
0–2^32	The unique audio identity of the accessory, defined in AT*SEAUDIO

<audio_class>:

<audio_class>	Description
0–2^32	The default audio class of the accessory. Defined in the AT*SEAUDIO command

Other parameters:

Binary	Valid values	Format
Interface		
ASC_Interface_TX_Impendence	0 – Low 1 – 150 2 – 1k 3 – High	Integer
ASC_Interface_RX_Impendence	0 – Low 1 – 150 2 – 1k 3 – High	Integer
ASC_Interface_Line_in_capable	0 – Off 1 – On	Integer
Call		
ASC_Call_TXNOM	-43+[-32,31] (2c) gives values between -75 and -12	Integer
ASC_Call_TXGAIN	[-8,7] (2 comp dB) gives values between -8 and 7	Integer
ASC_Call_SLRTarget	13+[-8,7] (2 comp) gives values between 5 and 20	Integer
ASC_Call_TXNC	0 – Off 1 – Low 2 – Normal 3 – Aggressive	Integer
ASC_Call_TXFilter8	SOS1(b0,b1,b2,a1,a2) SOS2(b1,b2,a1,a2) SOS3(b1,b2,a1,a2) SOS4(b1,b2,a1,a2) SOS5(b1,b2,a1,a2)	String type, Hex (672 bits) *Example*: 0X01234567 0X89BCDEF… “0123456789ABCDEF…”
ASC_Call_RXNOM	-34+[-32,31] (2c) gives values between -66 and -3	Integer
ASC_Call_RXGAIN	[-8,7] (2 comp dB)	Integer
ASC_Call_RLRTarget	[-8,7] (2 comp)	Integer
ASC_Call_RXMAS	-15+[-16,15] (2c) gives values between -31 and 0	Integer
ASC_Call_RXFilter8	SOS1(b0,b1,b2,a1,a2) SOS2(b1,b2,a1,a2) SOS3(b1,b2,a1,a2) SOS4(b1,b2,a1,a2) SOS5(b1,b2,a1,a2)	String type, Hex (672 bits) *Example*: 0X01234567 0X89BCDEF… “0123456789ABCDEF…”

Binary	Valid values	Format
ASC_Call_Sidetone	5*(8+[-8,7]) (2c) gives values between 5-75 in steps of 5	Integer
ASC_Call_EC (Echo Canceller)	0 – Off 1 – On Comfort noise generation, connection to NC	Integer
Leisure		
ASC_Leisure_TXClip	-30+[-32,31] (2c) gives values between -62 and 1	Integer
ASC_Leisure_TXNC	SOS1(b0,b1,b2,a1,a2) SOS2(b1,b2,a1,a2) SOS3(b1,b2,a1,a2) SOS4(b1,b2,a1,a2) SOS5(b1,b2,a1,a2)	String type, Hex (672 bits) *Example*: 0X01234567 0X89BCDEF… “0123456789ABCDEF…”
ASC_Leisure_RXMAS	-15+[-16,15] (2c) gives values between -31 and 0	Integer
ASC_Leisure_RXNG	[-32,31] (2c) give values between -32 and 31	Integer
ASC_Leisure_RXFilter441	SOS1(b0,b1,b2,a1,a2) SOS2(b1,b2,a1,a2) SOS3(b1,b2,a1,a2) SOS4(b1,b2,a1,a2) SOS5(b1,b2,a1,a2)	String type, Hex (672 bits) *Example*: 0X01234567 0X89BCDEF… “0123456789ABCDEF…”
ASC_Leisure_RXFilter48	SOS1(b0,b1,b2,a1,a2) SOS2(b1,b2,a1,a2) SOS3(b1,b2,a1,a2) SOS4(b1,b2,a1,a2) SOS5(b1,b2,a1,a2)	String type, Hex (672 bits) *Example*: 0X01234567 0X89BCDEF… “0123456789ABCDEF…”

<status>: Unsolicited result code sent when the Audio parameters have been saved and calculated

<status>	Description
0	OK
1	ERROR

AT*SEVOL Volume level

Description: This command is used to set the volume for all different sound types in the ME. Each setting is responded with unsolicited result code ***SEVOL**, returning information to accessories that the volume has changed.

Set command: **AT*SEVOL=**<sound type>,<level>

Display current settings command

AT*SEVOL?
*SEVOL:1,<level>
*SEVOL:2,<level>
*SEVOL:3,<level>

Test command: **AT*SEVOL=?**

Test command response

Shows if the command is supported.
*SEVOL:1,(0-8)
*SEVOL:2,(0-8)
*SEVOL:3,(0-15)

Parameter:

<sound_type>:

<sound_type>	Description	<level> range
1	Ring volume	0–8
2	Call volume	0–8
3	Media volume	0–15

AT*SEVOLIR Volume indication request

Description: Activates or deactivates subscription to volume levels for all different sound types in the ME. Each setting is responded with unsolicited result code ***SEVOL**, returning information to accessories that the volume has changed.

Activation command:

AT*SEVOLIR=<activation>
Request to subscribe or stop subscribing for volume levels for all sound types in the ME.

Activation command response:

*SEVOLIR:1,<level>
*SEVOLIR:2,<level>
*SEVOLIR:3,<level>

Read command: **AT*SEVOLIR?** Display current settings.
*SEVOLIR:<activation>

Test command: **AT*SEVOLIR=?**

Test command response

Shows if the command is supported.
*SEVOLIR:(0,1)

Parameter:

<activation>:

<activation>	Description
0	Deactivate subscription
1	Activate subscription

AT*SEBIC Status bar icon

Description: Controls which status bar images to be shown. The <image> parameter points out the image and the <show> parameter states if the image is to be shown or not.

Set command: **AT*SEBIC**=<image>,<show>

Test command: **AT*SEBIC=?**

Test command response

Shows if the command is supported.
*SEBIC: (list of supported <image>s)

Parameter:

<image>:

<image>	Description
1	Radio image in status bar

<show>:

<show>	Description
0	Do not show
1	Show

AT*SEANT Antenna identification

Description: Informs the ME that it has an antenna. It is possible to turn on or zoff one or many antenna types with one request. The parameter <status> indicates if the accessory has an antenna and <pin> points out which pin it is connected to. <type> indicates the type of the connected antenna and which frequency span it can handle.

Execution command:

Antenna status request:
AT*SEANT=<status>,<pin>,<type>[[,<pin>],<type] ...

Read command: Read current setting:
AT*SEANT?

Read command response:

*SEANT:<status>,<pin>,<type>[[,<pin>],<type] ...

Test command: **AT*SEANT=?** Test if command is supported and show supported parameters

Test command response

*SEANT:(list of supported <status>s),<pin>,(list of supported <type>s)

Parameter:

<status>:

<status>	Description
0	Disable antenna. **Default value**
1	Enable antenna

<type>:

<type>	Description
1	FM. Low – 87.5 MHz, High – 108.0 MHz
2	TV. Low – 470.0 MHz, High – 862.0 MHz

AT*SESP Speakermode on/off

Description: Disables and enables speaker mode in ME.

Execution command: Speakermode status:
AT*SESP=<status>

Read command: Read current status:
AT*SESP?

Read command response: *SESP:<status>

Test command: **AT*SESP=?** Test if command is supported and show supported parameters

Test command response *SESP:(list of supported <status>s)

Parameter:

<status>:

<status>	Description
0	Disable speakermode. **Default value**
1	Enable speakermode

AT*SETBC Text to bitmap converter

Description: Converts a string received from the accessory to a bitmap and sends it to the accessory. It will return a bitmap formatted according to the given height, width and quality. The smallest font default size in ME is used when converting the text. The command will only convert the amount of text that fits in the given BMP size. When the conversion is done the ME will send the bitmap in one or many AT commands to the accessory.

Execution command: **AT*SETBC=** <bmp_width>, <bmp_height>, <quality>, <string>

Response: *SETBC:<setbc_index>,<setbc_data>

Test command: **AT*SETBC=?** Test if command is supported and show supported parameters

Test command response *SETBC: list of supported <quality>s

Parameter:

<bmp_width> Integer. Bitmap width in pixels

<bmp_height> Integer. Bitmap height in pixels. Must be a multiple of 8.

<string> String. The text format is dependent on the AT+CSCS setting.

<setbc_index>: Integer

<setbc_index>	Description
0…255 (0 = first)	Sequence number of setbc_data element

<quality>: Integer

<quality>	Description
1	Monochrome (B&W)

Encoding of bitmaps into setbc_data

The SETBC data is considered to be a stream of data segmented into a series of maximum 255 chunks. The chunks are encoded using hexadecimal format. Every byte is encoded using two ASCII digits/ character.

Each command sends one chunk. The length of the chunks is not defined, it is only required that they are sent in the right order. For every sent chunk the index is incremented. The chunks are re-assembled at the receiving side, in order to retrieve the total bitmap. A chunk can contain many complete rows.

The data starts with the most upper row from the left. Every row is followed by the row below.

The ME uses the smallest font available when transforming the text into a bitmap. The text will also be left adjusted.

Example:
A monochrome bitmap with size 80x56 pixels.

Step 1: The accessory sends a string of characters (“SE”) to the phone.

AT*SETBC: 80,56,1,”SE”

Step 2: The phone uses its smallest font to transform the text into a bitmap.

Step 2: The pixels of the bitmap are converted into values.

```
000000000000000000000000011111111111000000000000000 …
000000000000000000000001111111111111111111100000000 …
000000000000000000001111111111111111111111111100000 …
000000000000000000011111111111111111111111111111100 …
000000000000000000111111111111111111111111111111100 …
000000000000000001111111111111111111111111111111100 …
000000000000000111111111100000000000111111111111100 …
000000000000001111111110000000000000000011111111100 …
000000000000011111111100000000000000000000011111100 …
000000000000011111111000000000000000000000000111100 …
000000000000111111110000000000000000000000000001100 …
000000000001111111100000000000000000000000000000100 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000001111111100000000000000000000000000000000 …
000000000000111111111100000000000000000000000000000 …
000000000000011111111111000000000000000000000000000 …
000000000000001111111111111000000000000000000000000 …
000000000000000111111111111111000000000000000000000 …
000000000000000011111111111111111000000000000000000 …
000000000000000001111111111111111110000000000000000 …
000000000000000000111111111111111111000000000000000 …
000000000000000000011111111111111111110000000000000 …
000000000000000000001111111111111111111100000000000 …
000000000000000000000111111111111111111111000000000 …
000000000000000000000011111111111111111111110000000 …
000000000000000000000001111111111111111111111000000 …
000000000000000000000000111111111111111111111100000 …
000000000000000000000000011111111111111111111110000 …
```

…

Step 4: Values are converted into hex string (8 bit) starting with the upper row from left to right.

```
00 00 00 7F F0 00 00 … 00 00 07 FF FF 80 00 … 00 00 1F FF FF FC 00 … 00 00 3F
FF FF FF 80 …
```

Step 4: Chunks are created and sent to the accessory.

***SETBC:** 0,1,1,,,”000007FF00000…000007FFFF8000...00001FFFFFFC00…00003FFFFFFF80 … “
***SETBC:** 1,1,1,,,”...”

AT*SEAVRC Sony Ericsson audio video remote control

Description: Allows control of multimedia applications in the ME. Only a started multimedia application can be controlled with AT*SEAVRC. The different actions to control multimedia are the same as for the Bluetooth protocol AVRCP, which allows control of a multimedia application via Bluetooth. According to AVRCP, an action must be followed by a key press. A combined key pressed/released has been introduced to simplify the AT communication. The test command lists all supported actions.

Execution command: **AT*SEAVRC**=<action>,<key_press>

Test command: **AT*SEAVRC=?** Test if command is supported and show supported parameters

Test command response *SEAVRC: (<list of actions>)<CR><LF>

Parameter:

<action>:

<action>	Description
48	Channel up
49	Channel down
64	Power
65	Volume up
66	Volume down
67	Mute
68	Play
69	Stop
70	Pause
72	Rewind
73	Fast forward
75	Forward
76	Backward

<key_press>:

<key_press>	Description
1	Key released
2	Key pressed
3	Key pressed and released

AT*SEMMIR Sony Ericsson multimedia information request

Note: This command is implemented in two different versions.
AT*SEMMIR ver. 1 is valid for the following phones: K550, K610, K790, K800, K810, W350, W380, W610, W710, W830, W850, W880, Z310, Z555, Z610 and Z710.
AT*SEMMIR ver. 2 is valid for all other phones in this document.

AT*SEMMIR ver. 1

Description: Enables subscriptions for different multimedia content strings, for example, artist, song and frequency strings. When a content update occurs, the content string is sent with the unsolicted command ***SEMMII**. *SEMMII is only sent if a subscription has been made for the specific media type. Channel is defined as RDS data of PS (Programme Service) type

Execution command: Subscribe to different media information
AT*SEMMIR=<media_type>,<subscription>

Test command: **AT*SEMMIR=?** Test if command is supported and show supported parameters

Test command response <list supported content subscriptions>, <list supported subscribe values>

Parameter:

<media_type>:

<media_type>	**Description**
1	Artist
2	Song
3	Frequency
4	RDS
5	Channel
6..254	**Reserved for future use**
255	All media types

<subscription>:

<subscription>	**Description**
0	Subscription inactive
1	Subscription active

AT*SEMMIR ver. 2

Description: Controls subscriptions for different multimedia content strings, for example, artist, song and frequency strings. When a content update occurs, the content string is sent with the unsolicted command ***SEMMII**. *SEMMII is only sent if a subscription has been made for a specific media type.
The read command reports the sum of supported <media_type>s

Execute command:
Subscribe to different media information
AT*SEMMIR=<media_type_bmp>

Read command: **AT*SEMMIR?** Read current settings

Read command response
*SEMMIR: <media_type_bmp>

Test command: **AT*SEMMIR=?** Test if command is supported and list supported parameters

Test command response
*SEMMIR: (list of supported <media_type_bmp>s)
Supported media types are presented as bitmaps, for example:
“0” = No subscriptions
“1” = Subscription to artist
“3” = Subscription to artist and song

Parameter:

<media_type>:

<media_type>	**Description**
1	Artist
2	Song
4 – 2^{32}-2	**Reserved for future use**

<media_type_bmp>:

<media_type_bmp>	**Description**
0	Disable subscription
1 – 2^{32}-2	Bitmap of <media_type>s
4294967295 (2^{32}-1)	Enable all subscriptions

AT*SEAPP Sony Ericsson application

Description: Starts and terminates ME applications. The applications are assoicated with MIME strings. The MIME strings are used in AT*SEAPP to control a specific application. It is possible to start, terminate and, if allowed, start the application in minimal view. Symbian phones support all kind of MIME strings including the Sony Ericsson specific MIME strings, while OSE phones only supports the Sony Ericsson specified MIME strings. If a MIME string is not supported by the ME, AT_ERROR will be replied.

Execution command:
Start and terminate multimedia applications:
AT*SEAPP=<MIME>,<action>

Test command: **AT*SEAPP=?** Test if command is supported and show supported parameters

Test command response (list supported <action> values)

Parameter:

<MIME>:

<MIME>	**Description**
“application/SEMC.audioplayer”	Audio player
“application/SEMC.audiorecorder”	Audio recorder
“application/SEMC.FMtuner”	FM tuner
“application/SEMC.TV”	TV
“application/SEMC.videoplayer”	Video player
“application/SEMC.videorecorder”	Video recorder
“application/SEMC.imageviewer”	Image viewer
“application/SEMC.imagerecorder”	Image recorder
Free to use **other MIME strings** (see MIME specifications for existing MIME strings). ME will return ERROR if not supported	...

<action>:

<action>	**Description**
0	Stop application
1	Start application
2	Start application in minimal view (only if the application supports this feature through the MMI)

AT*SEAPPIR Sony Ericsson application indication request

Description: Activates the unsolicted result code, *SEAPPI, which provides information when an application is started or terminated. Application status is reported when the unsolicited result code is activated. If no started application exists, an empty string is returned.

Execution command: Start application indication subscription:
AT*SEAPPIR=<subscription>

Response: *SEAPPIR: <MIME>,<status>

Test command: **AT*SEAPPIR=?** Test if command is supported and show supported parameters

Test command response: <list supported subscription values>

Parameter:

<MIME>:

<MIME>	Description
“” (empty string)	No application, to be used when no application is started
“application/SEMC.audioplayer”	Audio player
“application/SEMC.audiorecorder”	Audio recorder
“application/SEMC.tuner”	Tuner
“application/SEMC.TV”	TV
“application/SEMC.videoplayer”	Video player
“application/SEMC.videorecorder”	Video recorder
“application/SEMC.imageviewer”	Image viewer
“application/SEMC.imagerecorder”	Image recorder
Free to use **other MIME strings** (see MIME specifications for existing MIME strings). ME will return ERROR if not supported	

<status>:

<status>	Description
0	Application stopped
1	Application started
2	Application started in minimal view

AT*SEJCOMM Sony Ericsson Java comm

Description: Registers a new Java virtual serial port, accessible to MIDlets as AT<port>. If the command is successful, CONNECT is returned and the AT channel enters transparent mode. Subsequent characters sent to the MS will appear as input on the virtual serial port. Characters sent to the virtual serial port will be transmitted to the AT channel. When the MIDlet is terminated or closes the virtual serial port, the AT channel leaves transparent mode and the command returns OK, unless <persistent> is 1, in which case the AT channel remains in transparent mode forever.

Execution command: Register Java serial port:
AT*SEJCOMM=<port>[,<persistent>]

Test command: **AT*SEJCOMM=?** Test if command is supported and show supported parameters

Test command response *SEJCOMM: (1-2^32),(0,1)

Parameter:

<port>:

<port>	Description
1–4294967295	Port identifier, used as an arbiter to allow this AT command to be executed from several AT channels in parallel. If this AT command is used by an accessory, it is recommended that the accessory-specific identification number, as specified in AT*SEACID, is used

<persistent>:

<persistent>	Description
0	The AT channel will leave transparent mode and the AT command will return OK when a connected MIDlet is terminated or closes the virtual serial port
1	The AT channel will remain in transparent mode for the remaining lifetime of the AT channel

AT*SEDUC Sony Ericsson disable USB charge

Description: Disables USB charging of the Mobile Equipment.
This command lets USB accessories tell the ME that it should not try to use the accessory as a power source.
This AT command should be sent before the accessory presents itself electrically as an USB device.
If USB charging is disabled by an accessory it will remain disabled until the AT-channel is destroyed.

Execution command: Disable USB charging:
AT*SEDUC

Test command: **AT*SEDUC=?** Test if command is supported

AT*SEABS Sony Ericsson accessory battery status

Description: The command is used by an accessory to report its battery level to the ME.

Execution command: Report battery status to ME
AT*SEABS=<status>[,<level>]

Test command: **AT*SEABS=?** Test if command is supported and show supported parameters

Test command response *SEABS: (list of supported <status>,list of supported <level>s)

Parameter:

<status>:

<status>	Description
0	Battery normal
1	Battery low

<level>:

<subscription>	Description
0	Battery exhausted
1–99	Battery charging level
100	Battery fully charged

AT*SEAVRCIR Sony Ericsson audio video remote control indication request

Description: Enables and disables the unsolicited result code ***SEAVRCI** that reports different multimedia states.

Set command: **AT*SEAVRCIR**=<status>
Enable/disable unsolicited *SEAVRCI.

Test command: **AT*SEAVRCIR=?**

Parameter:

<status>:

<status>	Description
0	Disable *SEAVRCI
1	Enable *SEAVRCI

Unsolicited result codes

*SEFEXP Flash auto exposure setting result code

Description: This unsolicited result code is returned when the flash auto exposure circuitry has been set in order to make picture lighter or darker. The result code is activated using **AT*SEFEXP**.

Unsolicited result code: ***SEFEXP=**<exposure>

Parameter:

<exposure>:

<exposure>	Description
0–255	New auto exposure amount according to control circuit algorithm

*SEMOD Camera mode indicator result code

Description: This unsolicited result code is returned when a user has set the flash in correct mode of operation using **AT*SEMOD**.

Unsolicited result code: ***SEMOD:** <action>

Parameter:

<action>:

<mode>	Description
0	ME is not in a mode were the flash has to be ready to fire. Flash disables its charging
1	ME is in a mode were the flash has to be ready to fire. Flash starts its charging if not already charged

*SEREDI Red eye reduction result code

Description: After *SEREDI is sent to the flash, the HW strobe signal has to come within the time out described for the AT command. This is for the red eye reduction to have effect (fire the flash while the pupils are constricted). If the strobe comes after the time out period, the flash will fire anyway but the red eye reduction effect will be less (pupils have started to dilate again). The result code is activated using **AT*SEREDI**.

Unsolicited result code: *SEREDI

*SEAULSI Audio line status result code

Description: Unsolicited result code that is sent when mute has been changed or audio line has been lost or received. The result code is activated using **AT*SEAULS**.

Unsolicited result code: *SEAULSI:<audio_line>,<audio_type>

*SEFUNCI Functionality status result code

Description: Unsolicited result code that is sent when functionality status has been changed. The result code is activated using **AT*SEFUNC**.

Unsolicited result code: *SEFUNCI: <mode>, <states>

*SEVOLI Volume level result code

Description: This result code is sent every time a change in volume level occurs for any sound type. The result code carries information of the sound type and the volume level. The sound types and their corresponding volume level range are listed in the table below.

Unsolicited result code:

***SEVOLI:**<sound type>,<level>
When the volume level of any sound type has changed.

Parameter:

<sound_type>:

<sound_type>	Description	<level> range
1	Ring volume	0–8
2	Call volume	0–8
3	Media volume	0–15

*SEMMII Sony Ericsson Multimedia Information Indication

Description:

This result code reports content string for a particular media type. The unsolicted command for a particular media type is activated with **AT*SEMMIR**. The unsolicited *SEMMII is sent when the content string for a particular media type has been updated.

Unsolicited result code:

***SEMMII:**<media_type>,<media_string>
When the subscribed media information has been updated.

Parameter:

<media_string>:

<media_string>	Description
String type	Character set as specified by command “Select TE Character Set”, AT+CSCS

*SEAVRCI Sony Ericsson audio video remote control indication

Description:

Unsolicited result code that is sent when the media status has been changed. The unsolicited is enabled by **AT*SEAVRCIR**.

Unsolicited result code:

***SEAVRCI:** <state>

Parameter:

<state>:

<state>	Description
0	Stop
1	Play
2	Pause

OBEX Formats

OBEX file system overview

One of the most basic and desirable uses of the IrDA infrared communication protocols is simply to send an arbitrary data object from one device to another and to make it easy for both application developers and users to do so. This is referred to as object exchange.

With the exception of Level 1 Information Exchange, whereby the objects are pushed into a device inbox, the object names passed to OBEX PUT and GET operations always include the path information.

The paths are specified in the IrMC specification from IrDA.

File name	Description	Supported operations
Device Info		
telecom/devinfo.txt	Information hardware version, software version, serial number, and so on. Character sets	GET
telecom/rtc.txt	The Real Time Clock Object contains the current date and time of the device	GET/PUT
Phonebook		
telecom/pb.vcf	Level 2 access (access entire phone book database)	GET/PUT
telecom/pb/luid/.vcf	Add new entry	PUT
telecom/pb/0.vcf	Own business card	GET/PUT
telecom/pb/###.vcf	Level 3 static index access	GET/PUT
telecom/pb/luid/*.vcf	Level 4 unique index access	GET/PUT
telecom/pb/info.log	Supported properties and memory info	GET
telecom/pb/luid/###.log	Change log	GET
telecom/pb/luid/cc.log	Change counter	GET
Calendar		
telecom/cal.vcs	Level 2 access	GET/PUT
telecom/cal/luid/.vcs	Add new entry	PUT
telecom/cal/###.vcs	Level 3 static index access	GET/PUT
Telecom/cal/luid/*.vcs	Level 4 unique index access	GET/PUT
Telecom/cal/info.log	Supported properties and memory info	GET
Telecom/cal/luid/###.log	Change log	GET
Telecom/cal/luid/cc.log	Change counter	GET

eMelody Format

eMelody Object

Description: This is a definition of the eMelody object. This object is used when a user-defined melody is exchanged

Syntax:
<emelody-object>
“BEGIN:EMELODY”<CR><LF>
“NAME:”<name><CR><LF>
“COMPOSER:” <composer><CR><LF>
“VERSION:” <version><CR><LF>
“MELODY:”<melody><CR><LF>
“END:EMELODY”

File extension: emy

Example file name mymelody.emy

Parameters:

<version>: “1.0”

<name>: Alphanumeric string

<composer>: Alphanumeric string

<melody>: {<pause>|<tone>}

<pause>: “p”

<tone>: {[<octave_prefix>]<basic_tone>}

<basic_short_tone>: “c”|”d”|”e”|”f”|”g”|”a”|”b”

<ess_short_tone>: “(b)d”|”(b)e”|”(b)g”|”(b)a”|”(b)b”

<iss_short_tone>: “#d”|”#e”|”#g”|”#a”|”#b”

<basic_long_tone>: “C”|”D”|”E”|”F”|”G”|”A”|”B”

<ess_long_tone>: “(b)D”|”(b)E”|”(b)G”|”(b)A”|”(bB”

<iss_long_tone>: “#D”|”#E”|”#G”|”#A”|”#B”

<basic_tone>: <basic_short_tone>|<ess_short_tone>|<iss_short_tone>|<basic_long_tone>|<ess_long_tone>|<iss_long_tone>

<octave_high_prefix>:“+”

Maximum number of tones: 40

Maximum numbers of characters in melody: 120

Example:

```
BEGIN:EMELODY
VERSION:1.0
NAME:Test melody 1
COMPOSER:John Smith
MELODY:
+f+a+fa(b)bdcC+GA+d+#c+dfg+daea+d+#c+e+f+e+fa(b)bdC+EA+d+#c+
dfgba+d+#C
END:EMELODY
```

iMelody Format

iMelody Object

Description:	This is a definition of the iMelody object. This object is used when a user-defined melody is exchanged
Syntax:	**<imelody-object>** **“BEGIN:IMELODY”<CR><LF>** **“VERSION:” <version><CR><LF>** **“FORMAT:”<format>** **[“NAME:”<name><CR><LF>]** **[“COMPOSER:” <composer><CR><LF>]** **[“BEAT:”<beat>]** **[“STYLE:”<style>]** **[“VOLUME:”<volume>]** **“MELODY:”<melody><CR><LF>** **“END:IMELODY”**
File extension:	imy
Example file name	mymelody.imy
Parameters:	
<version>:	“1.0”
<format>:	“CLASS1.0” | “CLASS2.0”
<name>:	Alphanumeric string
<composer>:	Alphanumeric string
<beat>:	“25” | “26” | “27” | ... | “899” | “900“
<style>:	“S0“ | “S1“ | “S2“
<volume>:	V0“ | “V1“ | ... | “V15“ | “+“ | “-“ (+/- indicates volume change relative to current. Default is current)
<melody>:	{<silence>|<note>|<led>|<vib>|<backlight>|<repeat>}+
<silence>:	<rest ><duration>[<duration-specifier]
<rest>:	“r”
<duration>:	“0“ | “1“ | “2“ | “3“ | “4“ | “5“
<duration-specifier>:	“.“ | “:“ | “;“
<note>:	[<octave-prefix>]<basic-ess-iss-note><duration>[<duration-specifier]
<octave-prefix>:	“*0“ | “*1“ | … | “*8“ ((A=55Hz) | (A=110Hz) | … | (A=14080 Hz))
<basic-ess-iss-note>:	<basic-note> | <ess-note> | <iss-note>
<basic-note>:	“c“ | “d“ | “e“ | “f“ | “g“ | “a“ | “b“
<ess-note>:	“&d“ | “&e“ | “&g“ | “&a“ | “&b“ (flat notes)
<iss-note>:	“#c“ | “#d“ | “#f“ | “#g“ | “#a“ (sharp notes)

<led>:	“ledoff“	“ledon“
<vibe>:	“vibeon”	“vibeoff”
<backlight>:	“backon”	“backoff”
<repeat>:	“(“	“)“	“@“<repeat-count> (start of repeat block, end of repeat block and repetition count)
<repeat-count>:	“0“	“1“	“2“	... (0 is repeat forever)
Maximum number of notes:	40
Maximum numbers of characters in melody:	120

Example:

```
BEGIN:IMELODY
VERSION:1.0
NAME:Melody1
COMPOSER:Mozart
BEAT:120
STYLE:1
VOLUME:7
MELODY:&b2#c3-c2*4g3d3+#d1r3d2e2:d1+f2f3
END:IMELODY
```

vCard Format

The vCard object uses a subset of the properties defined in the vCard specification from the Internet Mail Consortium. The vCard standard is available from the Infrared Data Association at http://www.irda.org.

vCard Object

.

Description:	This is a definition of the vCard object. This object is used when a user-defined contact card is exchanged
Syntax:	**<vcard-object> “BEGIN:VCARD<CR><LF> “VERSION:”<version><CR><LF> “N:”<encoding>”;<character_set>”:”<name><CR><LF> [“FN:”<encoding>”;”<character_set>”:”<formatted_name><CR><LF>] [“TEL:”<telephone_number><CR><LF>] [“X-IRMC-LUID:”<x_irmc_local_unique_identifier><CR><LF>] “END:VCARD”**
File extension:	vcf
Example file name:	person.vcf
Parameters:	
<version>:	“2.1”
<encoding>:	(“QUOTED-PRINTABLE”|”BASE-64”|”8BIT”)
<character_set>:	(“ISO-8859-1”|”UTF-8”)

<name>:	String. Maximum length 18 bytes. Encapsulates the individual components of an object's name. The property value is a concatenation of the Family Name (first field), Given Name (second field), Additional Names (third field), Name Prefix (fourth field) and Name Suffix (fifth field) strings.
<formatted_name>:	String. Maximum length 20 bytes. Specifies the formatted name string associated with the vCard object. This is the way that the name is to be displayed.
<telephone_string>:	String: Maximum length 20 bytes. Specifies the canonical number string for telephony communication with the vCard object. The value of this property is specified in a canonical form in order to specify an unambiguous representation of the globally unique telephony endpoint. This property is based on the X.520 Telephony Number attribute.
<x_irmc_local_unique_identifier>:	String. Maximum length 12 bytes. IrMC Local Unique Identifier field label. Local Unique identifier 48 bits coded in its hexadecimal representation as 12 ASCII characters.

Example:	`BEGIN:VCARD`
	`VERSION:2.1`
	`N:QUOTED-PRINTABLE;CHARSET=ISO-8859-1:Book;Sven;Ola;Mr.`
	`FN:QUOTED-PRINTABLE;CHARSET=ISO-8859-1:Mr. Sven O. Book`
	`TEL:+4646123123`
	`END:VCARD`

vNote Format

Syntax:	**<vnote-object>** **"BEGIN:VNOTE<CR><LF>** **"VERSION:"<version><CR><LF>** **["X-IRMC-LUID:"<x_irmc_local_unique_identifier><CR><LF>]** **"N:"<encoding>";<character_set>":"<name><CR><LF>** **["FN:"<encoding>";"<character_set>":"<formatted_name><CR><LF>]** **["TEL:"<telephone_number><CR><LF>]** **"END:VCARD"**
File extension:	vnt
Example file name:	scribble.vnt
Parameters:	
<version>:	"2.1"
<encoding>:	("QUOTED-PRINTABLE"|"BASE-64"|"8BIT")
<character_set>:	("ISO-8859-1"|"UTF-8")
<name>:	String. Maximum length 18 bytes. Encapsulates the individual components of an object's name. The property value is a concatenation of the Family Name (first field), Given Name (second field), Additional Names (third field), Name Prefix (fourth field) and Name Suffix (fifth field) strings.
<formatted_name>:	String. Maximum length 20 bytes. Specifies the formatted name string associated with the vCard object. This is the way that the name is to be displayed.

<telephone_string>:	String. Maximum length 20 bytes. Specifies the canonical number string for telephony communication with the vCard object. The value of this property is specified in a canonical form in order to specify an unambiguous representation of the globally unique telephony endpoint. This property is based on the X.520 Telephony Number attribute.
<x_irmc_local_unique _identifier>:	String. Maximum length 12 bytes. IrMC Local Unique Identifier field label. Local Unique identifier 48 bits coded in its hexadecimal representation as 12 ASCII characters.

Example:

```
BEGIN:VCARD
VERSION:2.1
N:QUOTED-PRINTABLE;CHARSET=ISO-8859-1:Book;Sven;Ola;Mr.
FN:QUOTED-PRINTABLE;CHARSET=ISO-8859-1:Mr. Sven O. Book
TEL:+4646123123
END:VCARD
```

vCalendar Format

The vCalendar standard is available from the Infrared Data Association at http://www.irda.org.

vCalendar Object

Description:	This is a definition of the vCalendar object, which is related to the vEvent object. These objects are used when a user-defined calendar entry is exchanged
Syntax:	**<vcalendar-object>**
	“BEGIN:VCALENDAR”<CR><LF> **“VERSION:”<version><CR><LF>** **“PRODID:”<prodid><CR><LF>** **“BEGIN:VEVENT”<CR><LF>** **“END:VEVENT”<CR><LF>** **“BEGIN:VEVENT”<CR><LF>** **“END:VEVENT”<CR><LF>** **...** **“END:VCALENDAR”<CR><LF>**
File extension:	vcs
Example file name:	filename.vcs
VEVENT	See **vEvent** Object.
Parameters:	
<version>:	“1.0“
<prodid>:	“Sony Ericsson Calendar 1.0“

Example
vCalendar vEvent object (MEETING):

```
BEGIN:VCALENDAR
VERSION:1.0
PRODID:Sony Ericsson Calendar 1.0
BEGIN:VEVENT
DTSTART:19990125T123000
DTEND:19990125T170000
AALARM:19990125T121500
CATEGORIES:MEETING
SUMMARY;QUOTED-PRINTABLE;CHARSET=ISO-8859-1:Meeting
with Lars
LOCATION;QUOTED-PRINTABLE;CHARSET=ISO-8859-1:In my
room
X-IRMC-LUID:1E12FF7C01AB
END:VEVENT
END:VCALENDAR
```

vEvent Object

Description: This is a definition of the vEvent object, which is related to the vCalendar object. These objects are used when a user-defined calendar entry is exchanged. The phone supports all-day event meetings. The sync engine sends the vCalendar object with DTSTART, set the date YYYYMMDD and leave out the time THHMMSS out. The DTSTART is mandatory, as well as the DTEND. The same principles applies for DTEND, that is, THHMMSS is skipped.

Syntax: **<vevent-object>**

“BEGIN:VEVENT“<CR>
“DTSTART:“<date_and_time>
“DTEND:“<date_and_time>
“AALARM:“<date_and_time>
“CATEGORIES:“<category>
“SUMMARY;“<encoding>“;“<character_set>“:“<summary>
“LOCATION;“<encoding>“;“<character_set>“:“<location>
“X-IRMC-LUID:“<x_irmc_luid>
“END:VEVENT“

Parameters:

<date_and_time>: String. <year><month><day>T<hour><minute<second>.

The date and time values for all vCalendar properties are formatted as a string consistent with the ISO 8601 representation for combinations of dates and times.

Note: All time values are given in local time.

Example <date_and_time>: 19960415T083000. 8:30 AM on April 15, 1996 local time.

<category>: “MEETING“ | “PHONE CALL“ | “MISCELLANEOUS“

<encoding>: “QUOTED-PRINTABLE“ | “BASE-64“ | “8BIT“

<character_set>: “ISO-8859-1“ | “UTF-8“

<summary>: String. Maximum length 36 bytes.

<location>:	String. Maximum length 20 bytes
<x_irmc_luid>:	String. Maximum length 12 bytes. IrMC Local Unique Identifier field label. Local Unique identifier 48 bits coded in its hexadecimal representation as 12 ASCII characters. Holds the phone book index in decimal format.
Example DTSTART-DTEND:	DTSTART:1999-02-10, DTEND:1999-02-12. If the DTSTART and DTEND have different dates, the phone interprets it as a whole day event occurring over several days. In this example: the whole day on 1999-02-10, 1999-02-11 and 1999-02-12.

Appendix 1

This appendix contains information about specific AT commands for the G502, K630, K660, K850, V640, W890, W910 and Z750 series. For these phones, the AT commands in this appendix complement the commands found in chapter ”AT commands”.

Added AT commands

Ensemble S35: Sony Ericsson commands

Command

AT*SEGPSA Sony Ericsson global positioning system accessory

Description: Informs the ME about the presence of a Global Positioning System (GPS) capable accessory device. The accessory device may be used to provide the location based service engine of a ME without internal GPS with positioning data or to override the data generated from an internal GPS.
The accessory sends the command AT*SEGSPA when connected. The ME responds with CONNECT to indicate that the AT channel have changed into a link for GPCCCP/NMEA formatted data, the AT channel goes into transparent mode.
Upon reception of the GPSCCP Disconnect message the ME responds with OK and resets the link to normal AT mode.

Command: **AT*SEGPSA**
Tells the ME that there is a GPS capable device connected.

Response: CONNECT

Test command: **AT*SEGPSA=?** Test if command is supported.

Updated AT commands

Ensemble S35: Sony Ericsson commands

Command

AT*SEAUDIO Accessory class report

Description: Informs the phone about the general audio class and the unique audio ID of an accessory. If there is a specific audio settings container available for the unique id, the phone will use that configuration. If not, the configuration for the general audio class will be used.
When the accessory identifies itself acoustically with AT*SEAUDIO the phone responds with a result code indicating what audio class and what unique audio ID have been used when configuring audio. If no specific audio configuration was available for the unique audio ID used by the accessory this is indicated by setting <unique_audio_id>=0 in the result code. If the accessory for some reason do not have audiocapabilities it will send AT*SEAUDIO=0,0.

Execution command: **AT*SEAUDIO=**<audio_class>,<unique_audio_id>

Execution command response *SEAUDIO:<audio_class>,<unique_audio_id>

Read command: **AT*SEAUDIO?** Read current setting

Test command: **AT*SEAUDIO=?** Test if command is supported and show supported parameters

Test command response: *SEAUDIO:(range of supported <audio_class>s),(range of supported <unique_audio_id>s)

Parameters:

<audio_class>:

<audio_class>	Description
0–255	The default audio class of the accessory
0	The accessory has no audio capabilities
1	PHF – Portable Handsfree
2	VHF – Vehicle Handsfree
3	BVHF – Budget Vehicle Handsfree (without accessory mic)
4	BT – Bluetooth headset or handsfree
5	BTC – Bluetooth handsfree for installation in Car
6	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)

<audio_class>	Description
7	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in car
9	DSS – Desk Speaker Stand
10	BDSS – Budget Desk Speaker Stand
11	LOA – Line Out accessory Adjustable
12	LI – Line In accessory
13	LO – Line Out accessory

<audio_id>:

<audio_id>	Description
0–2^32	The unique audio identity of the accessory
0	Used in response codes to indicate that the terminal has not applied a specific audio configuration for the unique audio ID of the accessory
1–999	**Note**: Reserved for internal use in the telephone
1	PHF1 (**Note**: Not to be used by any accessory!)
2	PHF2 (**Note**: Not to be used by any accessory!)
3	PHF3 (**Note**: Not to be used by any accessory!)
4	PHF4 (**Note**: Not to be used by any accessory!)
5	Line in (**Note**: Not to be used by any accessory!)
6	Line out (**Note**: Not to be used by any accessory!)
7	BT headset (**Note**: Not to be used by any accessory!)
8	BT handsfree (**Note**: Not to be used by any accessory!)
1000–1999	PHF – Portable Handsfree
2000–2999	VHF – Vehicle Handsfree
3000–3999	BVHF – Budget Vehicle Handsfree (without accessory mic)
4000–4999	BT – Bluetooth headset or handsfree
5000–5999	BTC – Bluetooth handsfree for installation in Car
6000–6999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
7000–7999	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8000–8999	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in car

<audio_id>	Description
9000–9999	DSS – Desk Speaker Stand
10000–10999	BDSS – Budget Desk Speaker Stand
11000–11999	LOA – Line Out accessory Adjustable
12000–12999	LI – Line In accessory
13000–13999	LO – Line Out accessory

Appendix 2

This appendix contains information about specific AT commands for the C702, C902, T700, W595, W760, W902, W980, Z770 and Z780 series. The AT commands in this appendix complement the commands found in chapter ”AT commands”.

Added AT commands

Ensemble S35: Sony Ericsson commands

Commands

AT*SEGPSA Sony Ericsson global positioning system accessory

Description: Informs the ME about the presence of a Global Positioning System (GPS) capable accessory device. The accessory device may be used to provide the location based service engine of a ME without internal GPS with positioning data or to override the data generated from an internal GPS.
The accessory sends the command AT*SEGSPA when connected. The ME responds with CONNECT to indicate that the AT channel have changed into a link for GPCCCP/NMEA formatted data, the AT channel goes into transparent mode.
Upon reception of the GPSCCP Disconnect message the ME responds with OK and resets the link to normal AT mode.

Command: **AT*SEGPSA**
Tells the ME that there is a GPS capable device connected.

Response: CONNECT

Test command: **AT*SEGPSA=?** Test if command is supported.

Updated AT commands

Ensemble S35: Sony Ericsson commands

Command

AT*SEAUDIO Accessory Class Report

Description: Informs the phone about the general audio class and the unique audio ID of an accessory. If there is a specific audio settings container available for the unique id, the phone will use that configuration. If not, the configuration for the general audio class will be used.
When the accessory identifies itself acoustically with AT*SEAUDIO the phone responds with a result code indicating what audio class and what unique audio ID have been used when configuring audio. If no specific audio configuration was available for the unique audio ID used by the accessory this is indicated by setting <unique_audio_id>=0 in the result code. If the accessory for some reason do not have audiocapabilities it will send AT*SEAUDIO=0,0.

Execution command: **AT*SEAUDIO=**<audio_class>,<unique_audio_id>

Execution command response *SEAUDIO:<audio_class>,<unique_audio_id>

Read command: **AT*SEAUDIO?** Read current setting.

Test command: **AT*SEAUDIO=?** Test if command is supported and show supported parameters.

Test command response: *SEAUDIO:(range of supported <audio_class>s),(range of supported <unique_audio_id>s)

Parameters:

<audio_class>:

<audio_class>	Description
0–255	The default audio class of the accessory
0	The accessory has no audio capabilities
1	PHF – Portable Handsfree
2	VHF – Vehicle Handsfree
3	BVHF – Budget Vehicle Handsfree (without accessory mic)
4	BT – Bluetooth headset or handsfree
5	BTC – Bluetooth handsfree for installation in Car
6	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)

<audio_class>	Description
7	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car
9	DSS – Desk Speaker Stand
10	BDSS – Budget Desk Speaker Stand
11	LOA – Line Out accessory Adjustable
12	LI – Line In accessory
13	LO – Line Out accessory

<audio_id>:

<audio_id>	Description
0–2^32	The unique audio identity of the accessory
0	Used in response codes to indicate that the terminal has not applied a specific audio configuration for the unique audio ID of the accessory
1–999	**Note**: Reserved for internal use in the telephone
1	PHF1 (**Note**: Not to be used by any accessory!)
2	PHF2 (**Note**: Not to be used by any accessory!)
3	PHF3 (**Note**: Not to be used by any accessory!)
4	PHF4 (**Note**: Not to be used by any accessory!)
5	Line in (**Note**: Not to be used by any accessory!)
6	Line out (**Note**: Not to be used by any accessory!)
7	BT Headset (**Note**: Not to be used by any accessory!)
8	BT Handsfree (**Note**: Not to be used by any accessory!)
1000–1999	PHF – Portable Handsfree
2000–2999	VHF – Vehicle Handsfree
3000–3999	BVHF – Budget Vehicle Handsfree (without accessory mic)
4000–4999	BT – Bluetooth headset or handsfree
5000–5999	BTC – Bluetooth handsfree for installation in Car
6000–6999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
7000–7999	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8000–8999	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car

<audio_id>	Description
9000–9999	DSS – Desk Speaker Stand
10000–10999	BDSS – Budget Desk Speaker Stand
11000–11999	LOA – Line Out accessory Adjustable
12000–12999	LI – Line In accessory
13000–13999	LO – Line Out accessory

Appendix 3

This appendix contains information about specific AT commands for the C510, C901, C903, C905, G705, Jalou™, Naite™, Pureness™, T707, T715, W508, W705 and W715 series. The AT commands in this appendix complement the commands found in chapter ”AT commands”.

Added AT commands

Ensemble S35: Sony Ericsson commands

Commands

AT*SEGPSA Sony Ericsson global positioning system accessory

Description: Informs the ME about the presence of a Global Positioning System (GPS) capable accessory device. The accessory device may be used to provide the location based service engine of a ME without internal GPS with positioning data or to override the data generated from an internal GPS.
The accessory sends the command AT*SEGSPA when connected. The ME responds with CONNECT to indicate that the AT channel have changed into a link for GPCCCP/NMEA formatted data, the AT channel goes into transparent mode.
Upon reception of the GPSCCP Disconnect message the ME responds with OK and resets the link to normal AT mode.

Command: **AT*SEGPSA**
Tells the ME that there is a GPS capable device connected.

Response: CONNECT

Test command: **AT*SEGPSA=?** Test if command is supported.

AT*SETIR Sony Ericsson time information request

Description: Enables or disables sending of unsolicited result code ***SETIRI** from ME to TE in the case of a time change. A time change is defined as a change of the time that differs from normal time change, for example, manual time change, time zone change, PC sync, daylight saving change, and so on.
The read command returns the current time and daylight setting.

Set command: **AT*SETIR**=<ind>
Enable/disable time change reporting.

Set command response: *SETIR: <time>,<dst>

Read command: **AT*SETIR**?
Read the current setting.

Read command response: *SETIR: <time>,<dst>

Test command: **AT*SETIR**=? Test if command is supported.

Test command response: *SETIR: (list supported <ind> s)

Parameters:

<ind>:

<ind>	Description
0	Disable time change reporting
1	Enable time change reporting

<time>:

<time>	Description
String type	Time format is “yyyy/MM/dd,hh:mm:ss+zz”, where characters indicates year, month, day, hour, minutes, seconds and time zone. For example, “2007/06/01,11:08:54+01”

<dst>:

<dst>	Description
0	Standard time
1	Daylight saving

AT*SEMCM Sony Ericsson memory card management

Description: Reports the total and free memory on the memory card. If there is no memory card in the phone, the command returns 0 for both total and free memory.

Action command: **AT*SEMCM**

Action command response: *SEMCM: <free_mem>,<tot_mem>

Test command: **AT*SEMCM**=? Test if command is supported.

Parameters:

<free_mem>:

<free_mem>	Description
Integer	Remaning free user memory (in bytes) on memory card
0	No memory card available in the phone

<tot_mem>:

<tot_mem>	Description
Integer	Total memory size (in bytes) of the memory card
0	No memory card available in the phone

Unsolicited result code

*SETIRI Time Information Request Indicator

Description: This unsolicited result code is enabled with the **AT*SETIR** command and indicates changes in indicator levels. The <ind> parameter indicates the indicator order number.

Unsolicited result code:

***SETIRI:** <time>,<dst>
When a change of indicator level occurs.

Parameters:

<time>:

<time>	**Description**
String type	Time format is “yyyy/MM/dd,hh:mm:ss+zz”, where characters indicate year, month, day, hour, minutes, seconds and time zone. For example, “2007/06/01,11:08:54+01”

<dst>:

<dst>	**Description**
0	Standard time
1	Daylight saving

Updated AT commands

Ensemble S35: Sony Ericsson commands

Command

AT*SEAUDIO Accessory Class Report

Description: Informs the phone about the general audio class and the unique audio ID of an accessory. If there is a specific audio settings container available for the unique id, the phone will use that configuration. If not, the configuration for the general audio class will be used.
When the accessory identifies itself acoustically with AT*SEAUDIO the phone responds with a result code indicating what audio class and what unique audio ID have been used when configuring audio. If no specific audio configuration was available for the unique audio ID used by the accessory this is indicated by setting <unique_audio_id>=0 in the result code. If the accessory for some reason do not have audiocapabilities it will send AT*SEAUDIO=0,0.

Execution command: **AT*SEAUDIO=**<audio_class>,<unique_audio_id>

Execution command response *SEAUDIO:<audio_class>,<unique_audio_id>

Read command: **AT*SEAUDIO?** Read current setting.

Test command: **AT*SEAUDIO=?** Test if command is supported and show supported parameters.

Test command response: *SEAUDIO:(range of supported <audio_class>s),(range of supported <unique_audio_id>s)

Parameters:

<audio_class>:

<audio_class>	Description
0–255	The default audio class of the accessory
0	The accessory has no audio capabilities
1	PHF – Portable Handsfree
2	VHF – Vehicle Handsfree
3	BVHF – Budget Vehicle Handsfree (without accessory mic)
4	BT – Bluetooth headset or handsfree
5	BTC – Bluetooth handsfree for installation in Car
6	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)

<audio_class>	Description
7	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car
9	DSS – Desk Speaker Stand
10	BDSS – Budget Desk Speaker Stand
11	LOA – Line Out accessory Adjustable
12	LI – Line In accessory
13	LO – Line Out accessory

<audio_id>:

<audio_id>	Description
0–2^32	The unique audio identity of the accessory
0	Used in response codes to indicate that the terminal has not applied a specific audio configuration for the unique audio ID of the accessory
1–999	**Note**: Reserved for internal use in the telephone
1	PHF1 (**Note**: Not to be used by any accessory!)
2	PHF2 (**Note**: Not to be used by any accessory!)
3	PHF3 (**Note**: Not to be used by any accessory!)
4	PHF4 (**Note**: Not to be used by any accessory!)
5	Line in (**Note**: Not to be used by any accessory!)
6	Line out (**Note**: Not to be used by any accessory!)
7	BT Headset (**Note**: Not to be used by any accessory!)
8	BT Handsfree (**Note**: Not to be used by any accessory!)
1000–1999	PHF – Portable Handsfree
2000–2999	VHF – Vehicle Handsfree
3000–3999	BVHF – Budget Vehicle Handsfree (without accessory mic)
4000–4999	BT – Bluetooth headset or handsfree
5000–5999	BTC – Bluetooth handsfree for installation in Car
6000–6999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
7000–7999	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8000–8999	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car

<audio_id>	Description
9000–9999	DSS – Desk Speaker Stand
10000–10999	BDSS – Budget Desk Speaker Stand
11000–11999	LOA – Line Out accessory Adjustable
12000–12999	LI – Line In accessory
13000–13999	LO – Line Out accessory

Appendix 4

This appendix contains information about specific AT commands for the Aino™, W995 and Yari™ series. The AT commands in this appendix complement the commands found in chapter ”AT commands”.

Added AT commands

Ensemble S35: Sony Ericsson commands

Commands

AT*SEGPSA Sony Ericsson global positioning system accessory

Description: Informs the ME about the presence of a Global Positioning System (GPS) capable accessory device. The accessory device may be used to provide the location based service engine of a ME without internal GPS with positioning data or to override the data generated from an internal GPS.
The accessory sends the command AT*SEGSPA when connected. The ME responds with CONNECT to indicate that the AT channel have changed into a link for GPCCCP/NMEA formatted data, the AT channel goes into transparent mode.
Upon reception of the GPSCCP Disconnect message the ME responds with OK and resets the link to normal AT mode.

Command: **AT*SEGPSA**
Tells the ME that there is a GPS capable device connected.

Response: CONNECT

Test command: **AT*SEGPSA=?** Test if command is supported.

AT*SETIR Sony Ericsson time information request

Description: Enables or disables sending of unsolicited result code ***SETIRI** from ME to TE in the case of a time change. A time change is defined as a change of the time that differs from normal time change, for example, manual time change, time zone change, PC sync, daylight saving change, and so on.
The read command returns the current time and daylight setting.

Set command: **AT*SETIR**=<ind>
Enable/disable time change reporting.

Set command response: *SETIR: <time>,<dst>

Read command: **AT*SETIR**?
Read the current setting.

Read command response: *SETIR: <time>,<dst>

Test command: **AT*SETIR**=? Test if command is supported.

Test command response: *SETIR: (list supported <ind> s)

Parameters:

<ind>:

<ind>	Description
0	Disable time change reporting
1	Enable time change reporting

<time>:

<time>	Description
String type	Time format is “yyyy/MM/dd,hh:mm:ss+zz”, where characters indicates year, month, day, hour, minutes, seconds and time zone. For example, “2007/06/01,11:08:54+01”

<dst>:

<dst>	Description
0	Standard time
1	Daylight saving

AT*SEMCM Sony Ericsson memory card management

Description: Reports the total and free memory on the memory card. If there is no memory card in the phone, the command returns 0 for both total and free memory.

Action command: **AT*SEMCM**

Action command response: *SEMCM: <free_mem>,<tot_mem>

Test command: **AT*SEMCM**=? Test if command is supported.

Parameters:

<free_mem>:

<free_mem>	Description
Integer	Remaning free user memory (in bytes) on memory card
0	No memory card available in the phone

<tot_mem>:

<tot_mem>	Description
Integer	Total memory size (in bytes) of the memory card
0	No memory card available in the phone

AT*SEDISKL Sony Ericsson Disable Keylock

Description: The set command is used by an accessory to ensure that phone keys are not locked when the accessory accesses the phone. When an accessory disables keylock with the command, the original keylock status is a saved. When the accessory disconnects or requests the keylock to be enabled again via *SEDISKL the phone attempts to reset the original keylock state. The result of this is subject to the state of the phone.
The command is not supported for phones with a two-state keylock or phones without auto keylock.

Action command: **AT*SEDISKL**=<mode>
(Change keylock)

Action command response: *SEDISKL: <mode>

Read command: **AT*SEDISKL**?
Read the current setting.

Read command response: *SEDISKL: <mode>

Test command: **AT*SEDISKL**=? Test if command is supported.

Test command response: *SEDISKL: list of supported <mode>s

Parameters:

<mode>:

<mode>	**Description**
0	Keylock is disabled. **Default**
1	Keylock is enabled

Unsolicited result code

*SETIRI Time Information Request Indicator

Description: This unsolicited result code is enabled with the **AT*SETIR** command and indicates changes in indicator levels. The <ind> parameter indicates the indicator order number.

Unsolicited result code: ***SETIRI:** <time>,<dst>
When a change of indicator level occurs.

Parameters:

<time>:

<time>	**Description**
String type	Time format is “yyyy/MM/dd,hh:mm:ss+zz”, where characters indicate year, month, day, hour, minutes, seconds and time zone. For example, “2007/06/01,11:08:54+01”

<dst>:

<dst>	**Description**
0	Standard time
1	Daylight saving

Updated AT commands

Ensemble S35: Sony Ericsson commands

Command

AT*SEAUDIO Accessory Class Report

Description: Informs the phone about the general audio class and the unique audio ID of an accessory. If there is a specific audio settings container available for the unique id, the phone will use that configuration. If not, the configuration for the general audio class will be used.
When the accessory identifies itself acoustically with AT*SEAUDIO the phone responds with a result code indicating what audio class and what unique audio ID have been used when configuring audio. If no specific audio configuration was available for the unique audio ID used by the accessory this is indicated by setting <unique_audio_id>=0 in the result code. If the accessory for some reason do not have audiocapabilities it will send AT*SEAUDIO=0,0.

Execution command: **AT*SEAUDIO=**<audio_class>,<unique_audio_id>

Execution command response *SEAUDIO:<audio_class>,<unique_audio_id>

Read command: **AT*SEAUDIO?** Read current setting.

Test command: **AT*SEAUDIO=?** Test if command is supported and show supported parameters.

Test command response: *SEAUDIO:(range of supported <audio_class>s),(range of supported <unique_audio_id>s)

Parameters:

<audio_class>:

<audio_class>	Description
0–255	The default audio class of the accessory
0	The accessory has no audio capabilities
1	PHF – Portable Handsfree
2	VHF – Vehicle Handsfree
3	BVHF – Budget Vehicle Handsfree (without accessory mic)
4	BT – Bluetooth headset or handsfree
5	BTC – Bluetooth handsfree for installation in Car
6	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)

<audio_class>	Description
7	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car
9	DSS – Desk Speaker Stand
10	BDSS – Budget Desk Speaker Stand
11	LOA – Line Out accessory Adjustable
12	LI – Line In accessory
13	LO – Line Out accessory

<audio_id>:

<audio_id>	Description
0–2^32	The unique audio identity of the accessory
0	Used in response codes to indicate that the terminal has not applied a specific audio configuration for the unique audio ID of the accessory
1–999	**Note**: Reserved for internal use in the telephone
1	PHF1 (**Note**: Not to be used by any accessory!)
2	PHF2 (**Note**: Not to be used by any accessory!)
3	PHF3 (**Note**: Not to be used by any accessory!)
4	PHF4 (**Note**: Not to be used by any accessory!)
5	Line in (**Note**: Not to be used by any accessory!)
6	Line out (**Note**: Not to be used by any accessory!)
7	BT Headset (**Note**: Not to be used by any accessory!)
8	BT Handsfree (**Note**: Not to be used by any accessory!)
1000–1999	PHF – Portable Handsfree
2000–2999	VHF – Vehicle Handsfree
3000–3999	BVHF – Budget Vehicle Handsfree (without accessory mic)
4000–4999	BT – Bluetooth headset or handsfree
5000–5999	BTC – Bluetooth handsfree for installation in Car
6000–6999	BTL – Bluetooth Leisure (BT headset supporting the Advanced Audio Distribution Profile)
7000–7999	BTBTL – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile
8000–8999	BTBTLC – A Bluetooth audio device supporting the headset or the handsfree profile and the Advanced Audio Distribution Profile for installation in Car

<audio_id>	Description
9000–9999	DSS – Desk Speaker Stand
10000–10999	BDSS – Budget Desk Speaker Stand
11000–11999	LOA – Line Out accessory Adjustable
12000–12999	LI – Line In accessory
13000–13999	LO – Line Out accessory

Glossary

3GPP
3rd Generation Partnership Project. http://www.3gpp.org

Analog
An analog signal can have any value between two limits. For example, traditional telephone lines transfer the human voice, itself an analogue signal, by means of a continuously varying electrical voltage. This voltage is an electrical representation of the pressure produced by the sound on the telephone microphone.

ASCII
Acronym for American Standard Code for Information Interchange. A standard code used for transferring data between computers and associated equipment.

Asynchronous communication
Data communication in which data elements are NOT separated according to time. Instead, a special code such as a start bit and a stop bit is used. By using a code, in lieu of time, asynchronous communication is more tolerant of time variations and complex timing circuits are not needed. The serial port and the COM port of a computer are associated with asynchronous communication, as is the RS-232-C interface. Also some end to end modem protocols are asynchronous.

AT
Abbreviation for Attention and tells the phone modem that a command follows. AT must be used at the beginning of a command line or dial string.

AT command set
The set of commands used to control the modem.

Auto-answer mode
The state in which the modem automatically answers the telephone when it rings.

Beam
Sending an item to another phone or a compatible application using the infrared link. This can include ring signals, calendar entries and business cards.

Bearer
The method for accessing WAP from the phone, for example GSM Data (CSD) and SMS.

Bluetooth
Secure, fast, point-to-multipoint radio connection technology. http://www.bluetooth.com

bps
Acronym for ‘bits per second’ (bits/s). A measure of speed at which bits are transmitted over the telephone lines.

BTHF
Bluetooth Handsfree

Card
A single WML unit of navigation and user interface. May contain information to present to the user, instructions for gathering user input, and so on.

Carrier
The frequency used by two connecting modems to transmit and receive data.

CCITT
Consultative Committee for International Telephony and Telegraphy. A European-based advisory committee established by the United Nations to recommend international communication protocol standards.

CD
Carrier Detect. An EIA232 signal sent from the phone modem to your computer, usually indicating that the modem has detected a carrier signal over the communications line.

Command line
A line of alphanumeric characters sent to the modem to instruct the modem to perform the commands specified in the line of characters.

COM (communications) port
The name allocated to the serial port through which digital signals are exchanged between the computer and a serial peripheral. For example COM1 and COM2.

CSD
Circuit Switched Data

CTS
Clear To Send. An EIA232 signal sent from a modem to the computer, usually indicating that the modem is ready to receive data.

DCD
Data Carrier Connect. See AT&C.

DCE
Data Communications Equipment. This term applies to modems and to other equipment that provide communication between data terminal equipment and the telephone line.

Deck
A collection of WML cards.

Default value
A setting that the modem will use unless specified otherwise.

Digital transmission
A digital signal can have only two values. These can, for example, be ON and OFF, HIGH and LOW or 0 and 1. A digital signal is usually transferred by means of a voltage which is either HIGH or LOW. Conventional modems communicate by means of audio tones which can use the analog telephone network. The modem links through your mobile telephone to a digital network and therefore has no need to use audio encoding. However, when you use your mobile telephone for a voice call, the analog signal from the microphone must be converted into a digital signal.

This is done by a converter which samples the signal voltage several thousand times per second. Each sample is converted into a binary number which represents the voltage at that instant, for example, 10011010, and the binary numbers are sent as a serial stream down the digital network.

DSR
Data Set Ready. An EIA232 signal sent from the modem to the computer, usually indicating that the modem is ready to establish a connection.

DTE
Data Terminal Equipment. The equipment that provides data, such as a computer or terminal.

DTMF
Dial Tone Multi-Frequency

DTR
Data Terminal Ready. An EIA232 signal sent from the computer to the modem, usually indicating that the computer is ready to begin communication.

EIA
Electronics Industries Association. A U.S. based group that forms technical standards and coordinates ITU-TCCITT activities in the United States.

EMAE
End Mobile Accessory Equipment

EOL
End of line

EOP
End of page

EOM
End of message

Escape code
A series of three consecutive characters (default is +++) sent to the modem, causing it to exit online data mode and enter online command mode.

Factory default settings
The profile configuration that is in effect when the modem is shipped from the factory.

Fax Class
Standards for fax transmission are set as classes. Class I and II allow data transfer speeds ranging from 2400 bps to 9600 bps.

Final result code
A message sent from the modem to inform the PC that execution of an entered AT command has been completed. Examples are OK and ERROR.

Flow control
The use of characters or EIA232 signals to start and stop the flow of data to avoid data loss during buffering.

Full duplex
Communication involving data transmitted in two directions simultaneously.

Gateway
A WAP Gateway typically includes the following functionality:

- A Protocol Gateway. The protocol gateway translates requests from the WAP protocol stack to the WWW protocol stack (HTTP and TCP/IP)
- Content Encoders and Decoders. The content encoders translate Web content into compact encoded formats to reduce the size and number of packets travelling over the wireless data network

GIF
Graphics Interchange Format

Half duplex
Communication involving data being transmitted in two directions, but not at the same time.

HF
Handsfree

HSCSD
High Speed Circuit-Swithced Data

IMAE
Intermediate Mobile Accessory Equipment

Intermediate result code
Information sent from the modem to the PC as a response to an executed AT command. Intermediate result codes are always followed by a final result code. For example +CBC: 0,100.

IrMC
Infrared Mobile Communications standard.

IrDA
Infrared Data Association. http://www.irda.org

ISDN
The term used to refer to the digital public switched telephone network.

ISP
Internet Service Provider

ITU-T
The ITU Telecommunication Standardisation Sector (ITU-T), is a permanent organ of the International Telecommunication Union. The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardising telecommunication on a worldwide basis.

As a consequence of a reform process within the International Telecommunication Union (ITU), the CCITT ceased to exist as of 28 February 1993. In its place the ITU Telecommunication Standardisation Sector (ITU-T) was created as of 1 March 1993.

MMI
Man-Machine Interface

ME
Mobile Equipment. The Sony Ericsson wireless terminal, excluding the SIM card, which in most cases is a mobile phone.

Micro browser
Accesses and displays the Internet contents in your mobile phone, just as an ordinary browser does in your computer. The micro browser uses small file sizes and the bandwidth of the wireless handheld-network.

Modem
Modulator-Demodulator. A device that converts digital signals to analog for transmission over telephone lines, then converts them back to digital at the other end of the line.

MS
Mobile Station. This is the Sony Ericsson wireless terminal being controlled through the set of commands described in this document.

MSISDN
Mobile Station International Subscriber Directory Number

MT
Mobile Telephone

OBEX
The OBEX specification consists of two major parts: a protocol and an application framework. The OBEX protocol is a session level protocol that specifies the structure for the conversation between devices. It also contains a model for representing objects. The OBEX application framework is built on top of the OBEX protocol. Its main purpose is to facilitate interoperability between devices using the OBEX protocol. Please refer to http://www.irda.org.

Off hook
The modem state similar to picking up a telephone receiver. The modem goes off hook to dial or answer and remains off hook while connected.

Offline command mode
The operational state in which the modem can accept typed commands.

On hook
The modem state similar to hanging up a telephone receiver.

Online data mode
The state the modem is in when transmitting or receiving data over the telephone line.

OTA
Over-the-Air configuration. To provide settings for the phone by sending an SMS message over the network to the phone. This reduces the need for the user to configure the phone manually.

PIN
Personal Identification Number

PDA
Personal Digital Assistant

PDP
Packet Data Protocol

PDU
Protocol Description Unit

Phone Book
A memory in your mobile phone or SIM card where phone numbers can be stored and accessed by name or position.

Protocols
The rules or procedures all modems must follow to communicate.

QoS
Quality of Service

Result code
A message the modem sends to the computer containing information about the state of the modem.

RLP
Radio Link Protocol, an error correction protocol used during radio link connections.

RLSD
Received Line Signal Detect. See AT&C.

RTS
Request To Send. An EIA232 signal sent from the computer to the modem, usually indicating that the computer is ready to send data to the modem.

RS-232-C interface
A communication standard established by the Electronics Industry Association (Recommended Standard number 232, revision C). Originally established to standardise communication between computer and modem. It was later adapted to become a popular standard for communication between computer and any other peripheral equipment, including other computers.

SAP
Service Access Point

SC
Service Centre (for SMS)

SDU
Service Data Unit

Serial port
The port through which digital signals are exchanged between the modem and the computer.

Short message service (SMS)
A text messaging service permitting the transmission of up to 160 characters to a facsimile, X400, telex and voice services or mobile phone.

SIM card
Subscriber Identity Module card. It is a card that must be inserted in any GSM-based mobile phone. It contains subscriber details, security information and memory for a personal directory of numbers. The card can be a small plug-in type or credit card-sized but both types have the same functions. Your phone uses the small plug-in card.

SIR
Serial Infrared

SM
1. Short Message

2. SIM message storage.

Synchronous Communication:

V.22bis
ITU-T standard for 2400 bps.
V.27ter
ITU-T standard for 4800 bps full-duplex modems connected to switched telephone networks.
V.29
ITU-T standard for 9600 bps half-duplex modems included in FAX machines.
V.42bis
ITU-T standard for the compression of asynchronous data. V.42bis is based on a dictionary that looks up common strings and replaces the strings with code words. This reduces the amount of characters actually being transmitted. V.42bis has been found to be most effective for file transfers that contain long strings of repetitive information and least effective for short strings of unique data. It requires LAPM, MNP2, MNP3 or MNP4 as error correcting.

TA
Terminal Adaptor, which in most cases is a PCMCIA (Personal Computer Memory Card International Association) card.

TAE
Terminal Adaptor Equipment

TCP/IP
Transmission Control Protocol/Internet Protocol

TE
Terminal Equipment, which in most cases is a computer.

Unsolicited result code
A message sent from the modem to the PC that is not a response to an executed AT command. For example RING.

vCalendar
vCalendar and vEvent define a transport- and platform-independent format for exchanging calendar and scheduling information for use in PIMs/ PDAs and group schedulers. vCalendar and vEvent are specified by IMC and can be further studied at http://www.imc.org.

vCard
vCard automates the exchange of personal information typically found on a traditional business card, for use in applications such as Internet mail, voice mail, Web browsers, telephony applications, call centres, video conferencing, PIMs/PDAs, pagers, fax, office equipment and smart cards. vCard is specified by IMC at http://www.imc.org.

vEvent
See vCalendar.

WAP
Wireless Application Protocol. Handheld devices, low bandwidth, binary coded, a deck/card metaphor to specify a service. A card is typically a unit of interaction with the user, that is, either presentation of information or request for information from the user. A collection of cards is called a deck, which usually constitutes a service.

WAP Application
A collection of WML cards, with the new context attribute set in the entry card.

WAP service
A WML application residing on a Web site.

WBMP
WAP Bitmap

WML
Wireless Markup Language. A markup language used for authoring services, fulfilling the same purpose as HyperText Markup Language (HTML) on the World Wide Web (WWW). In contrast to HTML, WML is designed to fit small handheld devices.

Index

Numerics

3GPP 333

A

analog 333
ASCII 333
Asynchronous communication 333
AT command set 333
AT commands 333
auto-answer mode 333

B

beam 333
bearer 333
bits per second 333
Bluetooth 333
bps 333

C

card 334
carrier 334
carrier detect 334
CCITT 334, 335, 337
CD 334
COM port 334
command line 334
command state 337
Commands
- AT 26
- AT&C 47
- AT&D 47
- AT&F 27
- AT&W 28
- AT* 26
- AT*EAPN 230
- AT*EAPP 155
- AT*EAPS 229
- AT*EBCA 230
- AT*ECAM 149
- AT*ECBP 55
- AT*ECDF 168
- AT*EDIF 121
- AT*EDST 226
- AT*EIAAUR 257
- AT*EIAAUW 255
- AT*EIABTR 254
- AT*EIABTW 253
- AT*EIAC 241
- AT*EIACSR 252
- AT*EIACSW 251
- AT*EIAD 242
- AT*EIADNSV6R 267
- AT*EIADNSV6W 266
- AT*EIAIPCPR 264
- AT*EIAIPCPW 263
- AT*EIALCPR 260
- AT*EIALCPW 258
- AT*EIAPSR 246
- AT*EIAPSSR 249
- AT*EIAPSSW 248
- AT*EIAPSW 245
- AT*EIAR 244
- AT*EIARUTD 269
- AT*EIARUTR 271
- AT*EIARUTW 268
- AT*EIAW 243
- AT*EIBA 80
- AT*EINA 58
- AT*EIPS 122
- AT*EJAVA 152
- AT*EKEY 166
- AT*EKSE 162
- AT*ELIB 232
- AT*EMEM 164
- AT*EMWS 234
- AT*EPEE 229
- AT*ESKL 154
- AT*ESKS 154
- AT*EVAA 233
- AT*EWBA 237
- AT*EWCT 237
- AT*EWDT 236
- AT*EWSA 237
- AT*EWSG 237
- AT*SEABS 296
- AT*SEACC 272
- AT*SEACID 273
- AT*SEACID2 274
- AT*SEAM 59
- AT*SEANT 287
- AT*SEAPP 293
- AT*SEAPPIR 294
- AT*SEAUDIO 275, 310, 315, 322, 330
- AT*SEAULS 278
- AT*SEAUP 283
- AT*SEAVRC 288, 290
- AT*SEAVRCIR 297
- AT*SEBIC 287
- AT*SECHA 277
- AT*SEDATE 68

AT*SEDEL 74
AT*SEDISKL 328
AT*SEDUC 296
AT*SEFEXP 281
AT*SEFIN 280
AT*SEFRY 282
AT*SEFUNC 279
AT*SEGAUGE 70
AT*SEGPSA 309, 314, 319, 326
AT*SEGUP 71
AT*SEJCOMM 295
AT*SELERT 61
AT*SELIST 65
AT*SELOG 277
AT*SEMCM 320, 327
AT*SEMMIR 292
AT*SEMOD 281
AT*SEONO 72
AT*SEPING 278
AT*SEREDI 282
AT*SERSK 76
AT*SESAF 60
AT*SESLE 74
AT*SESP 288
AT*SESTRI 63
AT*SETICK 67
AT*SETIR 319, 326
AT*SEUIS 76
AT*SEVOL 286
AT*SEVOLIR 286
AT*STKC 169
AT*STKE 169
AT*STKR 170
AT+BINP 80
AT+BLDN 81
AT+BRSF 83
AT+BVRA 81
AT+CACM 117
AT+CALA 225
AT+CALD 226
AT+CAMM 117
AT+CAOC 116
AT+CAPD 226
AT+CBC 142
AT+CBST 93
AT+CCFC 110
AT+CCLK 224
AT+CCWA 112
AT+CDIP 118
AT+CEER 96
AT+CFUN 139
AT+CGACT 194
AT+CGATT 194
AT+CGCMOD 215
AT+CGDATA 195
AT+CGDCONT 192
AT+CGDSCONT 198
AT+CGEQMIN 208
AT+CGEQNEG 212
AT+CGEQREQ 203
AT+CGEREP 196
AT+CGMI 29
AT+CGMM 29
AT+CGMR 30
AT+CGPADDR 197
AT+CGREG 197
AT+CGSMS 193
AT+CGSN 30
AT+CGTFT 200
AT+CHLD 113
AT+CHSC 100
AT+CHSD 97
AT+CHSN 98
AT+CHSR 101
AT+CHSU 102
AT+CHUP 89
AT+CIMI 228
AT+CIND 146
AT+CKPD 143
AT+CLAC 28
AT+CLAN 151
AT+CLCC 37
AT+CLCK 134
AT+CLIP 108
AT+CLIR 109
AT+CLVL 163
AT+CMAR 147
AT+CMEC 159
AT+CMEE 175
AT+CMER 148
AT+CMGC 188
AT+CMGD 187
AT+CMGF 178
AT+CMGL 182
AT+CMGR 183
AT+CMGS 184
AT+CMGW 186
AT+CMMS 189
AT+CMSS 185
AT+CMUT 163
AT+CMUX 56
AT+CNMI 180
AT+CNUM 103
AT+COLP 119
AT+COPN 121
AT+COPS 106
AT+CPAS 140
AT+CPBF 221
AT+CPBR 220
AT+CPBS 218

AT+CPBW 222
AT+CPI 39
AT+CPIN 140
AT+CPMS 177
AT+CPOL 120
AT+CPROT 235
AT+CPWD 136
AT+CR 89
AT+CRC 89
AT+CREG 104
AT+CRES 180
AT+CRLP 94
AT+CRMP 165
AT+CRSL 162
AT+CRSM 160
AT+CSAS 179
AT+CSCA 179
AT+CSCS 88
AT+CSIL 153
AT+CSMS 176
AT+CSQ 142
AT+CSSN 115
AT+CUSD 131
AT+CV120 90
AT+CVHU 37
AT+DR 52
AT+DS 51
AT+FCLASS 55
AT+GCAP 30
AT+GCLIP 84
AT+GMI 31
AT+GMM 31
AT+GMR 32
AT+ICF 48
AT+IFC 48
AT+ILRR 50
AT+IPR 49
AT+NREC 82
AT+VGM 82
AT+VGS 83
AT+VTS 91
AT+WS46 54
ATA 33
ATD 34
ATD extension - Request GPRS Service ... 215
ATD extension - Request Packet Domain IP Service 216
ATE 41
ATH 33
ATI 27
ATO 36
ATQ 45
ATS0 42
ATS10 44
ATS2 42
ATS3 42
ATS4 43
ATS5 43
ATS7 44
ATV 45
ATX 46
ATZ 26
CSD 334
CTS 334

D

DCD 334
DCE 334
deck 334
default setting 334
factory default settings 335
digital transmission 334
DSR 335
DTE 335
DTR 335

E

EIA 335
Ensembles
C18 Fax class 1 55
C2 Control and Identification 26
C20 Audio control 55
C25 GSM 07.10 56
C26 Accessory UI 58
C27 Accessory UI 59
C3 Call control 33
C38 Bluetooth commands 80
C4 Interface commands 41
C6 Data compression 51
C9 Mode Management 54
S1 GSM DTE-DCE interface 88
S10 GSM mobile equipment error control . 175
S11 SMS and PDU mode 176
S15 GPRS packet domain 191
S16 Phonebook 218
S18 GSM clock, date, and alarm handling 224
S19 GSM subscriber information 228
S2 Call control 89
S20 Ericsson specific AT commands for GSM 229
S26 Voice control 233
S27 OBEX 235
S29 WAP browser 236
S3 GSM data/fax 93
S34 Internet account commands 238
S35 Sony Ericsson commands .272, 309, 310, 314, ...315, 319, 322, 326, 330
S4 Extended error reporting 96
S5 GSM HSCSD 97
S6 GSM network services 103

S7 GSM USSD 131
S8 GSM facility lock 134
S9 Mobile equipment, control and status .. 139
EOL 335
EOM 335
EOP 335
escape code 335

F
fax class 335
final result code 335
flow control 336
full duplex 336

G
gateway 336
GIF 336

H
half duplex 336

I
intermediate result code 336
intermediate result codes
+CHSR 102
+DR 53
+ILRR 51
IrDA 336
IrMC 336
ISDN 336
ISP 336
ITU-T 337

M
ME 337
micro browser 337
MMI 337
modem 337
MS 337

O
OBEX 337
OBEX Formats
eMelody 301
iMelody 302
vCalendar 305
vCARD 303
off hook 337
on hook 338
On-line data mode 338
OTA 338

P
PDA 338
phone book 338
PIN 338
protocols 338

R
result code 338
final 335
intermediate 336
unsolicited 340
RLP 338
RLSD 338
RS-232-C interface 338
RTS 338

S
SC 339
serial port 339
short message service 339
SIM card 339
SIR 339
SM 339

T
TA 339
TAE 339
TCP/IP 340
TE 340

U
unsolicited result code 340
unsolicited result codes
*CPII 40
*EBCA 232
*ECAV 171
*EDIF 127
*ELIP 124
*EOLP 125
*EPEV 232
*SEAAI 79
*SEAULSI 298
*SEAVRCI 299
*SEFEXP 297
*SEFUNCI 298
*SEGUII 77
*SEMMII 299
*SEMOD 297
*SEREDI 298
*SESFI 79
*SETIRI 320, 329
*SEVOLI 298
*STKI 172
*STKN 173
+BINP 86
+BSIR 86
+BVRA 85
+CALV 227

+CBM 189
+CCCM 126
+CCWA 125
+CDIP 129
+CDS 190
+CGEV 216
+CGREG 217
+CIEV 171
+CKEV 170
+CLIP 123
+CME 91
+CMT 190
+CMTI 190
+COLP 129
+CR 92
+CREG 122
+CRING 92
+CSSI 125
+CSSU 126
+CUSD 133
+GCLIP 87
+VGM 85
+VGS 85

V

V.22bis 339
V.27ter 339
V.42bis 339
vCalendar 340
vCard 340
vEvent 340

W

WAP 340
WAP Application 340
WAP service 340
WBMP 340
WML 340