

---

## **GSM Fax & Data Services**

# **Test Specification**

# **ACI LC**

**Author:** **Texas Instruments Berlin AG**  
Alt-Moabit 91d  
D-10559 Berlin  
Germany

**Date:** 24. September 2002  
**Document No.:** 8462.400.01.00  
**File:** ACILC.DOC

## 0 Table of Contents

0	Table of Contents .....	2
1	Document Control .....	3
1.1	Document History .....	3
1.2	References .....	4
1.3	Abbreviations.....	7
1.4	Terms.....	9
2	Overview.....	10
2.1	RA - Rate Adaptation .....	10
2.2	RLP - Radio Link Protocol .....	10
2.3	L2R - Layer 2 Relay Functionality .....	10
2.4	FAD 03.45 - Fax Adaptation Protocol.....	11
2.5	T.30 - Fax Protocol Entity .....	11
2.6	ACI - AT Command Interpreter.....	11
2.7	USART - Universal Synchronous Asynchronous Receiver Transmitter Driver .....	11
3	Parameters .....	12
4	TEST CASES.....	19
4.1	Routing (internal) (ACILC001 -) .....	19
4.1.1	ACILC001: Setup the Routing and the PCO view for the ACI test, and set ACI to transparent mode .....	19
4.2	Initialisation (ACILC011 - ACILC020) .....	20
4.2.1	ACILC011: Power On.....	20
4.2.2	ACILC020: Set/Query Location Service Source and Destination Address .....	22
4.2.3	ACILC021: Initialize Location Service, receive SMS with MLC source address.(CLSA) .....	23
4.2.4	ACILC022: Set Operating Mode of Location Service (CLOM).....	26
4.2.5	ACILC023: Set Operating Mode of Location Service 2 (CLOM).....	30
4.2.6	ACILC024: Set Operating Mode of Location Service 3 (CLOM).....	32
4.2.7	ACILC025: Set Periodic Update of Location Service (CLPS, failed, is to rebuilt) .....	34
4.2.8	ACILC026: Set Privacy Parameter for LC client (CLSR).....	38
4.2.9	ACILC027: Request position data, send data in SMS to MLC destination address.(CLSA, failed currently) .....	41

## 1 Document Control

| Copyright Condat AG, 2002.

All rights reserved.

Every effort has been made to ensure that the information contained in this document is accurate at the time of printing. However, the software described in this document is subject to continuous development and improvement. Condat GmbH reserves the right to change the specification of the software. Information in this document is subject to change without notice and does not represent a commitment on the part of Condat GmbH. Condat GmbH accepts no liability for any loss or damage arising from the use of any information contained in this document.

The software described in this document is furnished under a licence agreement and may be used or copied only in accordance with the terms of the agreement. It is an offence to copy the software in any way except as specifically set out in the agreement. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Condat GmbH.

Texas Instruments Berlin AG  
Alt Moabit 91d  
10559 Berlin  
Germany

Telephone: +49.30.3983-0  
Fax: +49.30.3983-  
Internet:  
E-mail:

### 1.1 Document History

Document Id.	Date	Author	Remarks
8462.400.01.00.	24 September 2002	RM	Initial

## 1.2 References

- [1] Rec. T.4 Standardisation of group 3 facsimile apparatus for document transmission;  
(CCITT-T.4, 1984)
- [2] ITU-T Recommendation T.30; Series T: Terminal equipments and protocols for telematic services;  
Procedures for document facsimile transmission in the general switched  
telephone network;  
(ITU-T.30, 1996)
- [3] ITU-T Recommendation T.31; Terminals for telematic services;  
Asynchronous facsimile DCE control - service class 1  
(ITU-T.31, 1995)
- [4] ITU-T Recommendation T.32; Terminals for telematic services;  
Asynchronous facsimile DCE control - service class 2  
(ITU-T.32, 1995)
- [5] Rec. T.35; Terminal equipment and protocols for telematic services;  
Procedures for the allocation of CCITT definite codes for non-standard facilities;  
(CCITT-T.35, 1991)
- [6] ITU-T Recommendation V.25 ter; Series V: data communication over the telephone network;  
Interfaces and voiceband modems; Serial asynchronous automatic dialling and control  
(ITU-T V.25 ter, 1997)
- [7] Rec. V.42 bis Data compression procedures for data circuit terminating equipment (DCE) using error correction procedures;  
(CCITT-V.42 bis, 1990)
- [8] Rec. V.110 (Blue book, Vol. VIII, Fascicle VIII.1) Support of data terminal equipments (DTEs) with V-series type interfaces by an  
integrated services digital network (ISDN);  
(CCITT-V.110, 1988)
- [9] European digital cellular telecommunications system (Phase 2);  
GSM Public Land Mobile Network (PLMN) connection types;  
(GSM 3.10, September 1994, version 4.3.1)
- [10] European digital cellular telecommunications system (Phase 2);  
Technical realisation of facsimile group 3 transparent;  
(GSM 3.45, September 1995, version 4.5.0)
- [11] Digital cellular telecommunications system (Phase 2);  
Mobile radio interface layer 3 specification;  
(GSM 4.08, November 1996, version 4.17.0)
- [12] European digital cellular telecommunications system (Phase 2);  
Rate adaptation on the Mobile Station - Base Station System (MS - BSS) Interface;  
(GSM 4.21, May 1995, version 4.6.0)
- [13] European digital cellular telecommunications system (Phase 2);  
Radio Link Protocol (RLP) for data and telematic services on the Mobile Station - Base Station System (MS - BSS) interface and  
the Base Station System - Mobile-service Switching Centre (BSS - MSC) interface  
(GSM 4.22, September 1994, version 4.3.0)
- [14] European digital cellular telecommunications system (Phase 2);  
Radio Link Protocol (RLP) for data and telematic services on the Mobile Station - Base Station System (MS - BSS) interface and  
the Base Station System - Mobile-service Switching Centre (BSS - MSC) interface  
(Amendment prA1 for GSM 4.22, version 4.3.0)  
(GSM 4.22, March 1995, version 4.4.0)
- [15] European digital cellular telecommunications system (Phase 2);  
General on Terminal Adaptation Functions (TAF) for Mobile Stations (MS);  
(GSM 7.01, December 1995, version 4.10.0)
- [16] European digital cellular telecommunications system (Phase 2);  
Terminal Adaptation Functions (TAF) for services using asynchronous bearer capabilities;  
(GSM 7.02, September 1994, version 4.5.1)

- [17] European digital cellular telecommunications system (Phase 2);  
Terminal Adaptation Functions (TAF) for services using synchronous bearer capabilities;  
(GSM 7.03, September 1994, version 4.5.1)
- [18] Digital cellular telecommunications system (Phase 2);  
Use of Data Terminal Equipment - Data Circuit terminating Equipment (DTE - DCE) interface for Short Message Service (SMS)  
and Cell Broadcast Services (CBS);  
(GSM 7.05, November 1996, version 4.8.0)
- [19] Digital cellular telecommunications system (Phase 2);  
AT command set for GSM Mobile Equipment (ME)  
(GSM 7.07, May 1996, version 4.1.0)
- [20] Digital cellular telecommunication system (Phase 2);  
Mobile Station (MS) conformance specification;  
Part 1: Conformance specification  
(GSM 11.10-1, November 1996, version 4.17.0)
- [21] Digital cellular telecommunications system (Phase 2);  
Mobile Station (MS) conformance specification;  
Part 2: Protocol Implementation Conformance Statement (PICS)  
proforma specification  
(GSM 11.10-2, May 1996, version 4.15.0)
- [22] Digital cellular telecommunications system (Phase 2);  
Mobile Station (MS) conformance specification;  
Part 3: Layer 3 (L3) Abstract Test Suite (ATS)  
(GSM 11.10-3, November 1996, version 4.17.0)
- [23] Proposal for Rate Adaptation implemented on a DSP;  
(C. Bianconi, Texas Instruments, January 1998, version 1.0)
- [24] MCU-DSP Interfaces for Data Applications;  
Specification S844  
(C. Bianconi, Texas Instruments, March 1998, version 0.1)
- [25] Users Guide  
6147.300.96.100; Condat GmbH
- [26] Service Access Point RA  
8411.100.98.100; Condat GmbH
- [27] Service Access Point RLP  
8411.101.98.100; Condat GmbH
- [28] Service Access Point L2R  
8411.102.98.100; Condat GmbH
- [29] Service Access Point FAD  
8411.103.98.100; Condat GmbH
- [30] Service Access Point T30  
8411.104.98.100; Condat GmbH
- [31] Service Access Point ACI  
8411.105.98.100; Condat GmbH
- [32] Message Sequence Charts RLP  
8411.201.98.100; Condat GmbH
- [33] Message Sequence Charts L2R  
8411.202.98.100; Condat GmbH
- [34] Message Sequence Charts FAD  
8411.203.98.100; Condat GmbH
- [35] Message Sequence Charts T30  
8411.204.98.100; Condat GmbH
- [36] Message Sequence Charts ACI  
8411.205.98.100; Condat GmbH
- [37] Proposal for Fax & Data Integration; March 1998  
8411.300.98.100; Condat GmbH
- [38] Test Specification RLP  
8411.401.98.100; Condat GmbH

- [39] Test Specification L2R  
8411.402.98.100; Condat GmbH
- [40] Test Specification FAD  
8411.403.98.100; Condat GmbH
- [41] Test Specification T30  
8411.404.98.100; Condat GmbH
- [42] Test Specification ACI  
8411.405.98.100; Condat GmbH
- [43] SDL Specification RLP  
8411.501.98.100; Condat GmbH
- [44] SDL Specification L2R  
8411.502.98.100; Condat GmbH
- [45] SDL Specification FAD  
8411.503.98.100; Condat GmbH
- [46] SDL Specification T30  
8411.504.98.100; Condat GmbH
- [47] SDL Specification ACI  
8411.505.98.100; Condat GmbH
- [48] Technical Documentation RLP  
8411.701.98.100; Condat GmbH
- [49] Technical Documentation L2R  
8411.702.98.100; Condat GmbH
- [50] Technical Documentation FAD  
8411.703.98.100; Condat GmbH
- [51] Technical Documentation T30  
8411.704.98.100; Condat GmbH
- [52] Technical Documentation ACI  
8411.705.98.100; Condat GmbH

### 1.3 Abbreviations

ACI	AT Command Interpreter
AGCH	Access Grant Channel
AT	Attention sequence "AT" to indicate valid commands of the ACI
BCCH	Broadcast Control Channel
BCS	Binary Coded Signals
BS	Base Station
BSIC	Base Station Identification Code
C/R	Command/Response
C1	Path Loss Criterion
C2	Reselection Criterion
CBCH	Cell Broadcast Channel
CBQ	Cell Bar Qualify
CC	Call Control
CCCH	Common Control Channel
CCD	Condat Coder Decoder
CKSN	Ciphering Key Sequence Number
CRC	Cyclic Redundancy Check
DCCH	Dedicated Control Channel
DISC	Disconnect Frame
DL	Data Link Layer
DM	Disconnected Mode Frame
DTX	Discontinuous Transmission
EA	Extension Bit Address Field
EL	Extension Bit Length Field
EMMI	Electrical Man Machine Interface
EOL	End Of Line
F	Final Bit
F&D	Fax and Data Protocol Stack
FACCH	Fast Associated Control Channel
FHO	Forced Handover
GP	Guard Period
GSM	Global System for Mobile Communication
HDLC	High level Data Link Control
HISR	High level Interrupt Service Routine
HPLMN	Home Public Land Mobile Network
I	Information Frame
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
ITU	International Telecommunication Union
IWF	Interworking Function
Kc	Authentication Key
L	Length Indicator
LAI	Location Area Information
LISR	Low level Interrupt Service Routine
LPD	Link Protocol Discriminator
M	More Data Bit
MCC	Mobile Country Code

---

MM	Mobility Management
MMI	Man Machine Interface
MNC	Mobile Network Code
MS	Mobile Station
MSG	Message phase in the GSM 3.45 protocol
N®	Receive Number
N(S)	Send Number
NCC	National Colour Code
NECI	New Establishment Causes included
OTD	Observed Time Difference
P	Poll Bit
P/F	Poll/Final Bit
PCH	Paging Channel
PCO	Point of Control and Observation
PDU	Protocol Description Unit
PL	Physical Layer
PLMN	Public Land Mobile Network
RACH	Random Access Channel
REJ	Reject Frame
RNR	Receive Not Ready Frame
RR	Radio Resource Management
RR	Receive Ready Frame
RTD	Real Time Difference
RTOS	Real Time Operating System
SABM	Set Asynchronous Balanced Mode
SACCH	Slow Associated Control Channel
SAP	Service Access Point
SAPI	Service Access Point Identifier
SDCCH	Slow Dedicated Control Channel
SIM	Subscriber Identity Module
SMS	Short Message Service
SMSCB	Short Message Service Cell Broadcast
SS	Supplementary Services
T.4	CCITT Standardisation for Document coding of Group 3 Facsimile Apparatus
TAP	Test Application Program
TCH	Traffic Channel
TCH/F	Traffic Channel Full Rate
TCH/H	Traffic Channel Half Rate
TDMA	Time Division Multiple Access
TE	Terminal Equipment - e. g. a PC
TMSI	Temporary Mobile Subscriber Identity
UA	Unnumbered Acknowledgement Frame
UI	Unnumbered Information Frame
V(A)	Acknowledgement State Variable
V®	Receive State Variable
V(S)	Send State Variable
VPLMN	Visiting Public Land Mobile Network



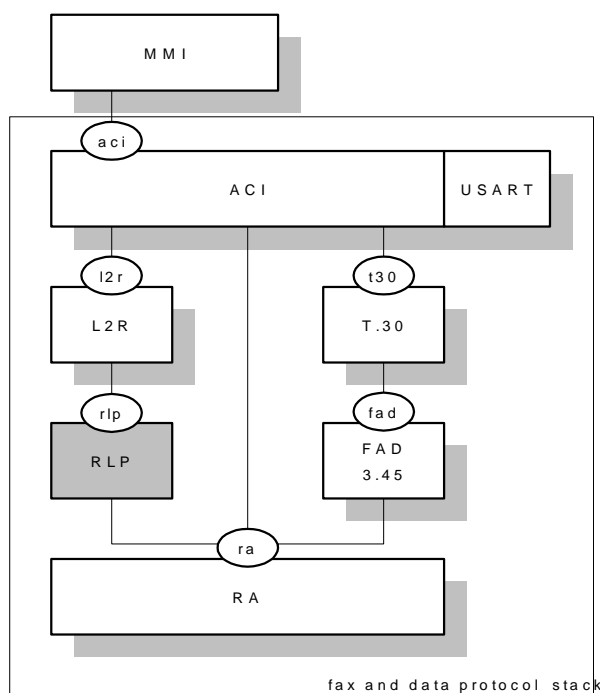
## 1.4 Terms

Entity:	Program which executes the functions of a layer
Message:	A message is a data unit which is transferred between the entities of the same layer (peer-to-peer) of the mobile and infrastructure side. Message is used as a synonym to protocol data unit (PDU). A message may contain several information elements.
Primitive:	A primitive is a data unit which is transferred between layers on one component (mobile station or infrastructure). The primitive has an operation code which identifies the primitive and its parameters.
Service Access Point	A Service Access Point is a data interface between two layers on one component (mobile station or infrastructure).

## 2 Overview

The Protocol Stacks are used to define the functionality of the GSM protocols for interfaces. The GSM specifications are normative when used to describe the functionality of interfaces, but the stacks and the subdivision of protocol layers does not imply or restrict any implementation.

The protocol stack for fax and data transmission consists of several entities. Each entity has one or more service access points, over which the entity provides a service for the upper entity. The entity, which is described in this document, is coloured grey in the following figure :



**Figure 2-1: Architecture of the fax and data protocol stack**

The information units passed via the SAPs are called primitives and consists of an operation code and several parameters. See the Users Guide for details.

The entities of the fax and data protocol stack are:

### 2.1 RA - Rate Adaptation

This entity performs an adaptation between an asynchronous or synchronous data stream with several bit rates on to the fixed bit rate used at the TCH. This is performed by the rate adaptation functions RA1' and RA0 described in GSM 04.21.

### 2.2 RLP - Radio Link Protocol

This entity provides a Layer 2 protocol for asynchronous reliable data transfer as specified in GSM 04.22. It includes error correction, sequence numbers and a mechanism for repeating corrupted and lost messages.

### 2.3 L2R - Layer 2 Relay Functionality

The L2R provides relay functions in order to adapt the character-oriented data received from the TE via USART to the bit-oriented RLP protocol.

## 2.4 FAD 03.45 - Fax Adaptation Protocol

The fax adaptation protocol, as specified in GSM 03.45, provides synchronisation with the BCS and MSG modems of the peer entity. It uses byte repetition in conjunction with a voting algorithm to handle corruption on the TCH data stream. The non-transparent fax protocol in accordance with GSM 03.46 is not part of this implementation.

The fax adapter enables T.30 to send BCS at 300 BPS and T.4 MSG in 2400, 4800, 7200 and 9600 BPS.

## 2.5 T.30 - Fax Protocol Entity

The protocol uses binary coded signals packed in HDLC frames to set up and release a connection in the message phase of the FAX transmission. This entity is specified in the ITU-T.30. The main tasks of this unit are:

- ☐ Building the HDLC frames with CRC.
- ☐ Performing bit stuffing/de-stuffing.
- ☐ Executing a sequence of 5 phases: 1.) set up, 2.) pre-message procedures, 3.) transmission/reception, 4.) post message procedures, 5.) waiting for call release.

## 2.6 ACI - AT Command Interpreter

The ACI is specified in GSM 07.07. It is responsible for call establishment via the GSM voice protocol stack and terminal adaptation for asynchronous transparent character-oriented data transmission. The ACI is able to receive AT commands and send the replies over the USART driver to a remote PC. This makes it possible to control the voice and data protocol stack from a remote application running on a PC. The ACI also provides a unique interface for an internal MMI in the MS.

## 2.7 USART - Universal Synchronous Asynchronous Receiver Transmitter Driver

The USART is a hardware component that facilitates a connection between the mobile station and terminal equipment (e.g. a PC). This interface uses some of the circuits described in V.24.

The data exchange provided by this unit is serial and asynchronous (synchronous communication is not in the scope of this document). A driver that uses interrupts to manage a circular buffer for the sending and receiving direction is necessary in order to use this component in the F&D. The driver has to be able to perform flow control.

### 3 Parameters

```
/* array declarations */
DECLARATION (F_SIM_SRV_NO_CC)
DECLARATION (A_ECC_FIELD )
DECLARATION (A_AD_FIELD_CI_DISABLED )
DECLARATION (CBM_MID_DEF)
DECLARATION (CBM_DCS_DEF)
DECLARATION (F_SIM_SRV_4)
DECLARATION (DELIVER_01)
DECLARATION (D_DELIVER_01)
DECLARATION (DELIVER_02)
DECLARATION (D_DELIVER_02)
DECLARATION (SM7_ABCDEFGHI_08)
DECLARATION (D_SM7_ABCDEFGHI_08)
DECLARATION (POS_DATA1)
```

```
/* structure declarations */
```

```
/* Number definitions */
```

```
BYTE NUM_0 0
BYTE NUM_1 1
BYTE NUM_2 2
BYTE NUM_3 3
BYTE NUM_4 4
BYTE NUM_5 5
BYTE NUM_6 6
BYTE NUM_7 7
BYTE NUM_8 8
BYTE NUM_9 9
BYTE NUM_10 10
BYTE NUM_12 12
BYTE NUM_FF 0xFF
BYTE REC_NUM_MAX 0xFF
BYTE L_DELIVER_01 31
BYTE L_DELIVER_02 31
BYTE L_SM7_ABCDEFGHI_08 30
```

```
/*Message Indication OK*/
```

```
STRING(M_OK, "OK" )
```

```
BYTE LM_OK 2
```

```
/* message reference */
```

```
SHORT MSG_REF_1 1
BYTE VALID 0
```

```
/*Command: +CFUN set phone functionality*/
```

```
STRING(C_PLUS_CFUN_FULL, "AT+CFUN=1 " )
```

```
BYTE LC_PLUS_CFUN_FULL 9
```

**/\*Message Indication +CME error result code\*/**

STRING(M\_ERR\_PIN\_REQ, "+CME ERROR: SIM PIN required" )

BYTE LM\_ERR\_PIN\_REQ 28

**/\*Message Indication +CMEE extended error report mode\*/**

STRING(C\_PLUS\_CMEE\_VERB, "AT+CMEE=2 " )

BYTE LC\_PLUS\_CMEE\_VERB 9

**/\*Command +CLSA = Set source and destination address of mobile location center (E-OTD)\*/**

STRING(C\_PLUS\_CLSA\_1234, "AT+CLSA=+4412345678,+449876543210")

BYTE LC\_PLUS\_CLSA\_1234 33

**/\*Command +CLSA = Set source and destination address of mobile location center (E-OTD)\*/**

STRING(C\_PLUS\_CLSA\_1234\_1, "AT+CLSA=4412345678,449876543210")

BYTE LC\_PLUS\_CLSA\_1234\_1 31

**/\*Command +CLSA? query source and destination address of mobile location center (E-OTD)\*/**

STRING(C\_PLUS\_CLSA\_QUERY, "AT+CLSA?")

BYTE LC\_PLUS\_CLSA\_QUERY 8

**/\*Message Indication+CLSA source/destination address MLC\*/**

STRING(M\_PLUS\_CLSA, "+CLSA:+4412345678,+449876543210")

BYTE LM\_PLUS\_CLSA 31

**/\*Command +CLOM = Set LCS Operating Mode (E-OTD) ON/ALL clients\*/**

STRING(C\_PLUS\_CLOM\_ON\_ALL, "AT+CLOM=1,1" )

BYTE LC\_PLUS\_CLOM\_ON\_ALL 11

STRING(C\_PLUS\_CLOM\_ON\_0, "AT+CLOM=1,0" )

BYTE LC\_PLUS\_CLOM\_ON\_0 11

**/\*Command +CLOM = Set LCS Operating Mode (E-OTD) ON/ client\*/**

STRING(C\_PLUS\_CLOM\_ON\_1, "AT+CLOM=1,4412345678")

BYTE LC\_PLUS\_CLOM\_ON\_1 20

STRING(C\_PLUS\_CLOM\_ON\_2, "AT+CLOM=1,+4412345678")

BYTE LC\_PLUS\_CLOM\_ON\_2 21

STRING(C\_PLUS\_CLOM\_ON\_3, "AT+CLOM=1,9876543210")

BYTE LC\_PLUS\_CLOM\_ON\_3 20

**/\*Command +CLOM = Set LCS Operating Mode (E-OTD)OFF/ALL clients\*/**

STRING(C\_PLUS\_CLOM\_OFF\_ALL, "AT+CLOM=0,0")

BYTE LC\_PLUS\_CLOM\_OFF\_ALL 11

**/\*Command +CLOM = Set LCS Operating Mode (E-OTD)OFF/ client\*/**

STRING(C\_PLUS\_CLOM\_OFF\_1, "AT+CLOM=0,4412345678")

BYTE LC\_PLUS\_CLOM\_OFF\_1 20

STRING(C\_PLUS\_CLOM\_OFF\_2, "AT+CLOM=0,+4412345678")

BYTE LC\_PLUS\_CLOM\_OFF\_2 21

STRING(C\_PLUS\_CLOM\_OFF\_3, "AT+CLOM=0,9876543210")

BYTE LC\_PLUS\_CLOM\_OFF\_3 20

```
/*Command +CLOM = Set LCS Operating Mode (E-OTD)Create/ client*/  
STRING(C_PLUS_CLOM_CREA_1, "AT+CLOM=3,4412345678")  
BYTE LC_PLUS_CLOM_CREA_1 20
```

```
STRING(C_PLUS_CLOM_CREA_2, "AT+CLOM=3,+4412345678")  
BYTE LC_PLUS_CLOM_CREA_2 21
```

```
STRING(C_PLUS_CLOM_CREA_3, "AT+CLOM=3,9876543210")  
BYTE LC_PLUS_CLOM_CREA_3 20
```

```
/*Command +CLOM = Set LCS Operating Mode (E-OTD)Delete/ client*/  
STRING(C_PLUS_CLOM_DEL_1, "AT+CLOM=2,+4412345678")  
BYTE LC_PLUS_CLOM_DEL_1 21
```

```
STRING(C_PLUS_CLOM_DEL_2, "AT+CLOM=2,4412345678")  
BYTE LC_PLUS_CLOM_DEL_2 20
```

```
STRING(C_PLUS_CLOM_DEL_3, "AT+CLOM=2,9876543210")  
BYTE LC_PLUS_CLOM_DEL_3 20
```

```
/*Command +CLOM? query LC operating mode (E-OTD)*/  
STRING(C_PLUS_CLOM_QUERY, "AT+CLOM?")  
BYTE LC_PLUS_CLOM_QUERY 8
```

```
STRING(C_PLUS_CLOM_DEL_5, "AT+CLOM=2,6666663210")  
BYTE LC_PLUS_CLOM_DEL_5 20
```

```
/*Message Indication+CLOM operating mode */  
STRING(M_PLUS_CLOM_QU1, "+CLOM:3" )  
STRING(M_PLUS_CLOM_QU5, "+CLOM:2" )  
STRING(M_PLUS_CLOM_QU11, "0,4412345678")  
STRING(M_PLUS_CLOM_QU51, "1,4412345678")  
STRING(M_PLUS_CLOM_QU12, "0,+4412345678")  
STRING(M_PLUS_CLOM_QU52, "1,+4412345678")  
STRING(M_PLUS_CLOM_QU13, "0,9876543210")  
BYTE LM_PLUS_CLOM_QU1 7  
BYTE LM_PLUS_CLOM_QU5 7  
BYTE LM_PLUS_CLOM_QU11 12  
BYTE LM_PLUS_CLOM_QU51 12  
BYTE LM_PLUS_CLOM_QU12 13  
BYTE LM_PLUS_CLOM_QU52 13  
BYTE LM_PLUS_CLOM_QU13 12
```

```
/*Message Indication+CLOM operating mode */  
STRING(M_PLUS_CLOM_QU2, "+CLOM:2" )  
STRING(M_PLUS_CLOM_QU21, "0,+4412345678")  
STRING(M_PLUS_CLOM_QU22, "0,9876543210")  
BYTE LM_PLUS_CLOM_QU2 7  
BYTE LM_PLUS_CLOM_QU21 13  
BYTE LM_PLUS_CLOM_QU22 12
```

```
/*Message Indication+CLOM operating mode */
```

```
STRING(M_PLUS_CLOM_QU4,"+CLOM:2" )
STRING(M_PLUS_CLOM_QU41,"0,4412345678")
STRING(M_PLUS_CLOM_QU42,"0,+4412345678")
BYTE LM_PLUS_CLOM_QU4 7
BYTE LM_PLUS_CLOM_QU41 12
BYTE LM_PLUS_CLOM_QU42 13
```

```

/*Message Indication+CLOM operating mode*/
STRING(M_PLUS_CLOM_QU3,"+CLOM:2")
STRING(M_PLUS_CLOM_QU31,"0,4412345678")
STRING(M_PLUS_CLOM_QU32,"0,9876543210")
BYTE LM_PLUS_CLOM_QU3 7
BYTE LM_PLUS_CLOM_QU31 12
BYTE LM_PLUS_CLOM_QU32 12

```

```

/*Command +CLPS = Set Periodic Update (E-OTD) ON/ ALL clients*/
STRING(C_PLUS_CLPS_ON_ALL, "AT+CLPS=1,1")
BYTE LC PLUS CLPS ON ALL 11

```

```
STRING(C_PLUS_CLPS_ON_1,"AT+CLPS=1,+4412345678,30")
BYTE LC PLUS CLPS ON 1 24
```

```
STRING(C_PLUS_CLPS_ON_2,"AT+CLPS=1,4412345678,120")
BYTE LC PLUS CLPS ON 2 24
```

```
STRING(C_PLUS_CLPS_ON_3,"AT+CLPS=1,9876543210,15")
BYTE LC PLUS CLPS ON 3 23
```

```

/*Command +CLPS = Set Periodic Update (E-OTD)OFF/ALL clients*/
STRING(C_PLUS_CLPS_OFF_ALL, "AT+CLPS=0,0")
BYTE LC PLUS CLPS OFF ALL 11

```

```

/*Command +CLPS = Remove Periodic Update (E-OTD)OFF/ client*/
STRING(C_PLUS_CLPS_OFF_2, "AT+CLPS=0,4412345678")
BYTE LC PLUS CLPS OFF 2 20

```

```

/*Command +CLPS? query LC operating mode (E-OTD)*/
STRING(C_PLUS_CLPS_QUERY,"AT+CLPS?")
BYTE LC PLUS CLPS QUERY 8

```

```

/*Message Indication+CLPS operating mode */
STRING(M_PLUS_CLPS_QU1,"+CLPS:2" )
STRING(M_PLUS_CLPS_QU11,"+4412345678,30")
STRING(M_PLUS_CLPS_QU12,"9876543210,15")
BYTE LM_PLUS_CLPS_QU1 7
BYTE LM_PLUS_CLPS_QU11 14
BYTE LM_PLUS_CLPS_QU12 13

```

```
/*Message Indication+CLPS delivered position data*/
STRING(M_CLPS_POS1,"% CLPS:12345678901234567890
1234567890123456789012345678901234567890123456789012345678901234567890
1234567890123456789012345678901234567890123456789012345678901234567890
1234567890123456789012345678901234567890123456789012345678901234567890
1234567890123456789012345678901234567890123456789012345678901234567890
12345678901288997766")
```

BYTE LM\_CLPS\_POS1 146

**/\*Command +CLSR = Set Privacy Parameter\*/**

STRING(C\_PLUS\_CLSR\_2\_NNNC, "AT+CLSR=0,0,+4412345678")

BYTE LC\_PLUS\_CLSR\_2\_NNNC 23

STRING(C\_PLUS\_CLSR\_1\_NNC, "AT+CLSR=1,0,4412345678")

BYTE LC\_PLUS\_CLSR\_1\_NNC 22

STRING(C\_PLUS\_CLSR\_3\_NC, "AT+CLSR=1,1,9876543210")

BYTE LC\_PLUS\_CLSR\_3\_NC 22

**/\* command: CNMI \*/**

STRING(C\_CNMI\_ON, "AT+CNMI=1,2,2,1,0")

BYTE LC\_CNMI\_ON 17

**/\* message: CMT \*/**

STRING(M\_CMT\_ABCDEFGHI\_HEADER\_1,

"+CMT: \"4412345678\", \"98/01/07,12:34:56+04\",145,0,64,0, \"12345\",129,9")

BYTE LM\_CMT\_ABCDEFGHI\_HEADER\_1 67

STRING(M\_CMT\_ABCDEFGHI\_HEADER\_2,

"+CMT: \"4412345678\", \"98/01/07,12:34:56+04\",129,0,64,0, \"12345\",129,9")

BYTE LM\_CMT\_ABCDEFGHI\_HEADER\_2 67

STRING(M\_CMT\_ABCDEFGHI, "ABCDEFGHI")

BYTE LM\_CMT\_ABCDEFGHI 9

**/\* command: CMGF \*/**

STRING(C\_CMGF\_SET\_TXT, "AT+CMGF=1")

BYTE LC\_CMGF\_SET\_TXT 9

STRING(C\_CMGF\_SET\_PDU, "AT+CMGF=0")

BYTE LC\_CMGF\_SET\_PDU 9

**/\* SIM EF(SMSP) Requests \*/****/\* empty list of CB message identifier ranges \*/**

BEGIN\_SHORT\_ARRAY (CBM\_MID\_DEF, 20)

0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF,

0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF,

0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF,

0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF,

0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF

ENDARRAY

**/\* empty list of CB data coding scheme ranges \*/**

BEGINARRAY (CBM\_DCS\_DEF, 20)

0xFF, 0xFF,

0xFF, 0xFF,

0xFF, 0xFF,

0xFF, 0xFF,

0xFF, 0xFF,

0xFF, 0xFF,

0xFF, 0xFF,



```
0xFF, 0xFF,  
0xFF, 0xFF,  
0xFF, 0xFF  
ENDARRAY
```

```
/* SMS-DELIVER */
```

```
/* SA_12345, OA_4412345678, PID_SM_TYPE_0, DCS_DEF_ALPH, VP_A9801071234564 (TP-SCTS),  
SM7_ABCDEFGHI */
```

```
BEGIN_PSTRUCT ("sms_sdu", DELIVER_01)  
    SET_COMP ("l_buf", L_DELIVER_01*8)  
    SET_COMP ("o_buf", 0x00)  
    SET_COMP ("buf", D_DELIVER_01)
```

```
ENDSTRUCT
```

```
BEGINARRAY_PART(D_DELIVER_01, L_DELIVER_01)  
    0x04, 0x81, 0x21, 0x43, 0xF5,  
    0x00, 0x0A, 0x91, 0x44, 0x21, 0x43, 0x65, 0x87,  
    0x40, 0x00, 0x89, 0x10, 0x70, 0x21, 0x43, 0x65, 0x40,  
    0x09, 0x41, 0xE1, 0x90, 0x58, 0x34, 0x1E, 0x91, 0x49
```

```
ENDARRAY
```

```
/* SA_12345, OA_+4412345678, PID_SM_TYPE_0, DCS_DEF_ALPH, VP_A9801071234564 (TP-SCTS),  
SM7_ABCDEFGHI */
```

```
BEGIN_PSTRUCT ("sms_sdu", DELIVER_02)  
    SET_COMP ("l_buf", L_DELIVER_02*8)  
    SET_COMP ("o_buf", 0x00)  
    SET_COMP ("buf", D_DELIVER_02)
```

```
ENDSTRUCT
```

```
BEGINARRAY_PART(D_DELIVER_02, L_DELIVER_02)  
    0x04, 0x81, 0x21, 0x43, 0xF5,  
    0x00, 0x0A, 0x81, 0x44, 0x21, 0x43, 0x65, 0x87,  
    0x40, 0x00, 0x89, 0x10, 0x70, 0x21, 0x43, 0x65, 0x40,  
    0x09, 0x41, 0xE1, 0x90, 0x58, 0x34, 0x1E, 0x91, 0x49
```

```
ENDARRAY
```

```
/* message: CMGS
```

```
STRING(M_CMGS_MSG_REF_1, "+CMGS: 1")
```

```
BYTE LM_CMGS_MSG_REF_1 8*/
```

```
/* SIM service table */
```

```
BEGINARRAY (F_SIM_SRV_NO_CC, 10)  
0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3C, 0x03, 0x00, 0x30  
ENDARRAY
```

```
/* SIM Service Table with Nr. 4 */
```

```
BEGINARRAY (F_SIM_SRV_4, 10) 0xC0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00  
ENDARRAY
```

```
/* EF ECC field array */
```

```
BEGINARRAY (A_ECC_FIELD, 12) 0x11, 0xF2, 0xFF, 0x99, 0xF9, 0xFF, 0x21, 0x43, 0x65, 0xFF, 0xFF, 0xFF ENDARRAY
```

```
/* EF AD field array , disable CI */
```

```
BEGINARRAY_PART (A_AD_FIELD_CI_DISABLED, 4) 0x00, 0x00, 0x00, 0x02 ENDARRAY
```

```
/* DA_654321, SA_12345, PID_SM_TYPE_0, DCS_DEF_ALPH, VP_A9801071234564, SM7_ABCDEFGHI */
```

```
BEGIN_PSTRUCT ("sms_sdu", SM7_ABCDEFGHI_08)
    SET_COMP ("l_buf", L_SM7_ABCDEFGHI_08*8)
    SET_COMP ("o_buf", 0x00)
    SET_COMP ("buf", D_SM7_ABCDEFGHI_08)
ENDSTRUCT
BEGINARRAY_PART(D_SM7_ABCDEFGHI_08, L_SM7_ABCDEFGHI_08)
    0x04, 0x81, 0x21, 0x43, 0xF5,
    0x19, 0x00, 0x06, 0x81, 0x56, 0x34, 0x12,
    0x40, 0x00, 0x89, 0x10, 0x70, 0x21, 0x43, 0x65, 0x40,
    0x09, 0x41, 0xE1, 0x90, 0x58, 0x34, 0x1E, 0x91, 0x49
ENDARRAY
```

**/\* SMS field array for position data \*/**

```
BEGINARRAY (POS_DATA1, 140) 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12,
0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34,
0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56,
0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78,
0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90,
0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12,
0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x88, 0x99, 0x77, 0x66 ENDARRAY
```

## 4 TEST CASES

### 4.1 Routing (internal) (ACILC001 -)

#### 4.1.1 ACILC001: Setup the Routing and the PCO view for the ACI test, and set ACI to transparent mode

Description:

Routings for the ACI tests are set

Preamble:

None

APL	ACI	PS
COMMAND (TAP RESET)		
COMMAND (CC RESET)		
COMMAND (MM RESET)		
COMMAND (SIM RESET)		
COMMAND (SS RESET)		
COMMAND (MMI RESET)		
COMMAND (SMS RESET)		
COMMAND (RR RESET)		
COMMAND (hCommGRR RESET)		
COMMAND (hCommGMM RESET)		
COMMAND (PL RESET)		
COMMAND (hCommLC RESET)		
COMMAND (TAP REDIRECT CLEAR)		
COMMAND (CC REDIRECT CLEAR)		
COMMAND (MM REDIRECT CLEAR)		
COMMAND (SIM REDIRECT CLEAR)		
COMMAND (SS REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CLEAR)		
COMMAND (SMS REDIRECT CLEAR)		
COMMAND (RR REDIRECT CLEAR)		
COMMAND (hCommGRR REDIRECT CLEAR)		
COMMAND (hCommGMM REDIRECT CLEAR)		
COMMAND (PL REDIRECT CLEAR)		
COMMAND (hCommLC REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CC TAP)		
COMMAND (MMI REDIRECT MM TAP)		
COMMAND (MMI REDIRECT SIM TAP)		
COMMAND (MMI REDIRECT SS TAP)		
COMMAND (MMI REDIRECT MMI TAP)		
COMMAND (MMI REDIRECT SMS TAP)		
COMMAND (MMI REDIRECT T30 TAP)		
COMMAND (MMI REDIRECT L2R TAP)		
COMMAND (MMI REDIRECT RA TAP)		
COMMAND (MMI REDIRECT RR TAP)		
COMMAND (MMI REDIRECT hCommGRR TAP)		
COMMAND (MMI REDIRECT hCommGMM TAP)		
COMMAND (MMI REDIRECT hCommLC TAP)		
COMMAND (PL REDIRECT MMI NULL)		

COMMAND (TAP REDIRECT TAP MMI)		
COMMAND (MMI REDIRECT MMI TAP)		

## Parametrization:

Primitive	Parameter	Value
History:	14.12.98	AK
		Initial

## 4.2 Initialisation (ACILC011 - ACILC020)

### 4.2.1 ACILC011: Power On

#### Description:

activate SIM card at power on, SIM indicates SAT features

#### Preamble:

ACILC001

APL	ACI	PS
(1)	ACI_CMD_REQ (cmd: +CMEE=2)	
	*=====>*	
(2)	ACI_CMD_IND (msg: OK)	
	*<=====*	
(3)	ACI_CMD_REQ (cmd: +CFUN=1)	
	*=====>*	
(4)		SIM_ACTIVATE_REQ
		*=====>*
(5)		SIM_ACTIVATE_CNF
		*<=====*
(6)		SIM_MMI_INSERT_IND
		*<=====*
(7)		SIM_READ_REQ
		*=====>*
(8)		SIM_READ_CNF
		*<=====*
(9)		SIM_READ_REQ
		*=====>*
(10)		SIM_READ_CNF
		*<=====*
(11)	ACI_CMD_IND (msg: OK)	
	*<=====*	

## Parametrization:

Primitive	Parameter	Value
-----------	-----------	-------

(1) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CMEE_VERB C_PLUS_CMEE_VERB
(2) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(3) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CFUN_FULL C_PLUS_CFUN_FULL
(4) SIM_ACTIVATE_REQ	proc mmi_pro_file stk_pro_file	SIM_INITIALISATION NOT_USED NOT_USED
(5) SIM_ACTIVATE_CNF	cause pin_cnt puk_cnt pin2_cnt puk2_cnt ec_code pref_lang	SIM_NO_ERROR NUM_3 NUM_10 NUM_3 NUM_10 NOT_USED NOT_USED
(6) SIM_MMI_INSERT_IND	func sim_serv imsi_field pref_plmn phase access_acm access_acmmax access_puct	SIM_ADN_ENABLED F_SIM_SRV_4 NOT_USED NOT_USED PHASE_2PLUS_SIM NOT_USED NOT_USED NOT_USED
(7) SIM_READ_REQ	source offset datafield length max_length	SRC_MMI NUM_0 SIM_ECC NOT_PRESENT_8BIT NUM_0
(8) SIM_READ_CNF	datafield cause length trans_data	SIM_ECC SIM_NO_ERROR NUM_12 A_ECC_FIELD
(9) SIM_READ_REQ	source offset datafield length max_length	SRC_MMI NUM_0 SIM_AD NOT_PRESENT_8BIT NUM_0
(10) SIM_READ_CNF	datafield	SIM_AD

	cause	SIM_NO_ERROR
	length	NUM_4
	trans_data	A_AD_FIELD_CI_DISABLED
(11) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK

History: 25.09.2002 RM Initial

## 4.2.2 ACILC020: Set/Query Location Service Source and Destination Address

Description:

Preamble:

ACILC011			
APL		ACI	PS
(1)	ACI_CMD_REQ (cmd: AT+CLOM ON)		
	*=====>*		
(2)	ACI_CMD_IND (msg: OK)		
	*<=====*		
(3)	ACI_CMD_REQ (cmd: AT+CLSA =)		
	*=====>*		
(4)	ACI_CMD_IND (msg: OK)		
	*<=====*		
(5)	ACI_CMD_REQ (cmd: AT+CLSA?)		
	*=====>*		
(6)	ACI_CMD_IND (msg: +CLSA: "+44...)		
	*<=====*		
(7)	ACI_CMD_IND (msg: OK)		
	*<=====*		

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_ON_ALL
	cmd_seq	C_PLUS_CLOM_ON_ALL
(2) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK
(3) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT

	cmd_len	LC_PLUS_CLSA_1234
	cmd_seq	C_PLUS_CLSA_1234
(4) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(5) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLSA_QUERY
	cmd_seq	C_PLUS_CLSA_QUERY
(6) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLSA
	cmd_seq	M_PLUS_CLSA
(7) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK

History:            10.10.02        RM        Initial

#### 4.2.3 ACILC021: Initialize Location Service, receive SMS with MLC source address.(CLSA)

Description:

Variants: <A>...<B>

Preamble:

ACILC011

APL	ACI	PS
(1)		
	ACI_CMD_REQ	
	(cmd: AT+CLOM ON)	
	*=====>*	
(2)		
	ACI_CMD_IND	
	(msg: OK)	
	*<=====*	
(3)		
	ACI_CMD_REQ	
	(cmd: AT+CLSA =)	
	*=====>*	
(4)		
	ACI_CMD_IND	
	(msg: OK)	
	*<=====*	
(5)		
	ACI_CMD_REQ	
	(cmd: CMGF=0)	
	*=====>*	
(6)		
	ACI_CMD_IND	
	(msg: OK)	
	*<=====*	
TIMEOUT (1000)		
(7)		
	MNSMS_REPORT_IND	
	*<=====*	
(8)		
	MNSMS_MESSAGE_IND	
	*<=====*	
(9)		
	MNSMS_MESSAGE_IND	
	*<=====*	
(10)		
	MNSMS_REPORT_IND	
	*<=====*	
(11)		
	ACI_CMD_REQ	
	(cmd: CNMI)	
	*=====>*	
(12)		
	MNSMS_CONFIGURE_REQ	
	*=====>*	
(13)		
	ACI_CMD_IND	
	(msg: OK)	
	*<=====*	
(14)		
	MNSMS_MESSAGE_IND	
	*<=====*	
(15)		
	ACI_CMD_IND	
	(msg: CMT)	
	*<=====*	
(16)		
	ACI_CMD_IND	
	(msg: CMT edit)	
	*<=====*	

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT



	cmd_len	LC_PLUS_CLOM_ON_ALL
	cmd_seq	C_PLUS_CLOM_ON_ALL
(2) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(3) ACI_CMD_REQ		
<A>	cmd_src	CMD_SRC_EXT
<B>	cmd_len	LC_PLUS_CLSA_1234
<A>	cmd_len	LC_PLUS_CLSA_1234_1
<B>	cmd_seq	C_PLUS_CLSA_1234
	cmd_seq	C_PLUS_CLSA_1234_1
(4) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(5) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_CMGF_SET_TXT
	cmd_seq	C_CMGF_SET_TXT
(6) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(7) MNSMS_REPORT_IND		
	state	SMS_STATE_INITIALISING
(8) MNSMS_MESSAGE_IND		
	mem_type	MEM_ME
	rec_num	SMS_RECORD_NOT_EXIST
	rec_max	REC_NUM_MAX
	status	SMS_RECORD_FREE
	sms_sdu	NOT_USED
(9) MNSMS_MESSAGE_IND		
	mem_type	MEM_SM
	rec_num	SMS_RECORD_NOT_EXIST
	rec_max	REC_NUM_MAX
	status	SMS_RECORD_FREE
	sms_sdu	NOT_USED
(10) MNSMS_REPORT_IND		
	state	SMS_STATE_READY
(11) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_CNMI_ON
	cmd_seq	C_CNMI_ON
(12) MNSMS_CONFIGURE_REQ		
	pref_mem_3	NOT_USED
	mt	NOT_USED
	ds	NOT_USED
	mhc	NOT_USED
(13) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(14) MNSMS_MESSAGE_IND		
	mem_type	MEM_ME

	rec_num	SMS_RECORD_NOT_EXIST
	rec_max	REC_NUM_MAX
	status	SMS_RECORD_REC_UNREAD
<A>	sms_sdu	DELIVER_01
<B>	sms_sdu	DELIVER_02
(15) ACI_CMD_IND		
<A>	cmd_len	LM_CMT_ABCDEFGHI_HEADER_1
<B>	cmd_len	LM_CMT_ABCDEFGHI_HEADER_2
<A>	cmd_seq	M_CMT_ABCDEFGHI_HEADER_1
<B>	cmd_seq	M_CMT_ABCDEFGHI_HEADER_2
(16) ACI_CMD_IND		
	cmd_len	LM_CMT_ABCDEFGHI
	cmd_seq	M_CMT_ABCDEFGHI

History: 10.10.02 RM Initial

## 4.2.4 ACILC022: Set Operating Mode of Location Service (CLOM)

Description: switch on/off EOTD Flag, create, activate, deactivate, remove LC clients

Variants: <A>...<C>

Preamble:

ACILC011

	APL	ACI	PS
(1)	             ACI_CMD_REQ         (cmd: AT+CLOM ON)   * =====> *	       	       
(2)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(3)	             ACI_CMD_REQ         (cmd: AT+CLOM CR)   * =====> *	       	       
(4)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(5)	             ACI_CMD_REQ         (cmd: AT+CLOM CR)   * =====> *	       	       
(6)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(7)	             ACI_CMD_REQ         (cmd: AT+CLOM CR)   * =====> *	       	       
(8)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(9)	             ACI_CMD_REQ         (cmd: AT+CLOM OFF)   * =====> *	       	       
(10)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(11)	             ACI_CMD_REQ         (cmd: AT+CLOM ON)   * =====> *	       	       
(12)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(13)	             ACI_CMD_REQ         (cmd: AT+CLOM DIS)   * =====> *	       	       
(14)	             ACI_CMD_IND         (msg: OK)   * <===== *	       	       
(15)	             ACI_CMD_REQ         (cmd: AT+CLOM?)   * =====> *	       	       
(16)	             ACI_CMD_IND         (msg: +CLOM:...)   * <===== *	       	       
(17)	             ACI_CMD_IND         (msg: +CLOM:...)   * <===== *	       	       
(18)	             ACI_CMD_IND 	       	       

	(msg: +CLOM:...)	
	* <=====	
(19)	ACI_CMD_IND	
	(msg: +CLOM:...)	
	* <=====	
(20)	ACI_CMD_IND	
	(msg: OK)	
	* <=====	
(21)	ACI_CMD_REQ	
	(cmd: AT+CLOM DEL)	
	* =====> *	
(22)	ACI_CMD_IND	
	(msg: OK)	
	* <=====	
(23)	ACI_CMD_REQ	
	(cmd: AT+CLOM?)	
	* =====> *	
(24)	ACI_CMD_IND	
	(msg: +CLOM:...)	
	* <=====	
(25)	ACI_CMD_IND	
	(msg: +CLOM:...)	
	* <=====	
(26)	ACI_CMD_IND	
	(msg: +CLOM:...)	
	* <=====	
(27)	ACI_CMD_IND	
	(msg: OK)	
	* <=====	

## Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_ON_ALL C_PLUS_CLOM_ON_ALL
(2) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(3) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_CREA_1 C_PLUS_CLOM_CREA_1
(4) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(5) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_CREA_2 C_PLUS_CLOM_CREA_2

(6) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(7) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_CREA_3 C_PLUS_CLOM_CREA_3
(8) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(9) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_OFF_ALL C_PLUS_CLOM_OFF_ALL
(10) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(11) ACI_CMD_REQ	cmd_src cmd_len <A> <B> <C> <A> <B> <C> cmd_seq cmd_seq cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_ON_1 LC_PLUS_CLOM_ON_2 LC_PLUS_CLOM_ON_3 C_PLUS_CLOM_ON_1 C_PLUS_CLOM_ON_2 C_PLUS_CLOM_ON_3
(12) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(13) ACI_CMD_REQ	cmd_src cmd_len <A> <B> <C> <A> <B> <C> cmd_seq cmd_seq cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_OFF_1 LC_PLUS_CLOM_OFF_2 LC_PLUS_CLOM_OFF_3 C_PLUS_CLOM_OFF_1 C_PLUS_CLOM_OFF_2 C_PLUS_CLOM_OFF_3
(14) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(15) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_QUERY C_PLUS_CLOM_QUERY
(16) ACI_CMD_IND	cmd_len cmd_seq	LM_PLUS_CLOM_QU1 M_PLUS_CLOM_QU1
(17) ACI_CMD_IND	cmd_len cmd_seq	LM_PLUS_CLOM_QU11 M_PLUS_CLOM_QU11

(18) ACI_CMD_IND	cmd_len cmd_seq	LM_PLUS_CLOM_QU12 M_PLUS_CLOM_QU12
(19) ACI_CMD_IND	cmd_len cmd_seq	LM_PLUS_CLOM_QU13 M_PLUS_CLOM_QU13
(20) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(21) ACI_CMD_REQ	cmd_src cmd_len cmd_len cmd_len cmd_seq cmd_seq cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_DEL_2 LC_PLUS_CLOM_DEL_1 LC_PLUS_CLOM_DEL_3 C_PLUS_CLOM_DEL_2 C_PLUS_CLOM_DEL_1 C_PLUS_CLOM_DEL_3
(22) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(23) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_QUERY C_PLUS_CLOM_QUERY
(24) ACI_CMD_IND	cmd_len cmd_seq	LM_PLUS_CLOM_QU2 M_PLUS_CLOM_QU2
(25) ACI_CMD_IND	<A> <B> <C> <A> <B> <C> cmd_len cmd_len cmd_len cmd_seq cmd_seq cmd_seq	LM_PLUS_CLOM_QU21 LM_PLUS_CLOM_QU31 LM_PLUS_CLOM_QU41 M_PLUS_CLOM_QU21 M_PLUS_CLOM_QU31 M_PLUS_CLOM_QU41
(26) ACI_CMD_IND	<A> <B> <C> <A> <B> <C> cmd_len cmd_len cmd_len cmd_seq cmd_seq cmd_seq	LM_PLUS_CLOM_QU22 LM_PLUS_CLOM_QU32 LM_PLUS_CLOM_QU42 M_PLUS_CLOM_QU22 M_PLUS_CLOM_QU32 M_PLUS_CLOM_QU42
(27) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK

History: 09.10.02 RM Initial

#### 4.2.5 ACILC023: Set Operating Mode of Location Service 2 (CLOM)

Description: switch on/off EOTD Flag, create, activate, deactivate, remove LC clients

Preamble:

ACILC011

	APL	ACI	PS
(1)	   ACI_CMD_REQ   (cmd: AT+CLOM ON)   * =====> *	       	       
(2)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(3)	   ACI_CMD_REQ   (cmd: AT+CLOM CR)   * =====> *	       	       
(4)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(5)	   ACI_CMD_REQ   (cmd: AT+CLOM CR)   * =====> *	       	       
(6)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(7)	   ACI_CMD_REQ   (cmd: AT+CLOM?)   * =====> *	       	       
(8)	   ACI_CMD_IND   (msg: +CLOM:...)   * <===== *	       	       
(9)	   ACI_CMD_IND   (msg: +CLOM:...)   * <===== *	       	       
(10)	   ACI_CMD_IND   (msg: +CLOM:...)   * <===== *	       	       
(11)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(12)	   ACI_CMD_REQ   (cmd: AT+CLOM DEL)   * =====> *	       	       

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_ON_ALL C_PLUS_CLOM_ON_ALL
(2) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(3) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT

	cmd_len	LC_PLUS_CLOM_CREA_1
	cmd_seq	C_PLUS_CLOM_CREA_1
(4) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(5) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_CREA_2
	cmd_seq	C_PLUS_CLOM_CREA_2
(6) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(7) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_QUERY
	cmd_seq	C_PLUS_CLOM_QUERY
(8) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLOM_QU5
	cmd_seq	M_PLUS_CLOM_QU5
(9) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLOM_QU51
	cmd_seq	M_PLUS_CLOM_QU51
(10) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLOM_QU52
	cmd_seq	M_PLUS_CLOM_QU52
(11) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(12) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_DEL_5
	cmd_seq	C_PLUS_CLOM_DEL_5

History:            01.11.02        RM        Initial

#### 4.2.6 ACILC024: Set Operating Mode of Location Service 3 (CLOM)

Description: switch on/off EOTD Flag, create, activate, deactivate, remove LC clients



Preamble:

ACILC011

	APL	ACI	PS
(1)	   ACI_CMD_REQ   (cmd: AT+CLOM ON)   * =====> *	       	       
(2)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(3)	   ACI_CMD_REQ   (cmd: AT+CLOM CR)   * =====> *	       	       
(4)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(5)	   ACI_CMD_REQ   (cmd: AT+CLOM CR)   * =====> *	       	       
(6)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(7)	   ACI_CMD_REQ   (cmd: AT+CLOM?)   * =====> *	       	       
(8)	   ACI_CMD_IND   (msg: +CLOM:...)   * <===== *	       	       
(9)	   ACI_CMD_IND   (msg: +CLOM:...)   * <===== *	       	       
(10)	   ACI_CMD_IND   (msg: +CLOM:...)   * <===== *	       	       
(11)	   ACI_CMD_IND   (msg: OK)   * <===== *	       	       
(12)	   ACI_CMD_REQ   (cmd: AT+CLOM ON, 0)   * =====> *	       	       

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_ON_ALL C_PLUS_CLOM_ON_ALL
(2) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(3) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT

	cmd_len	LC_PLUS_CLOM_CREA_1
	cmd_seq	C_PLUS_CLOM_CREA_1
(4) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(5) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_CREA_2
	cmd_seq	C_PLUS_CLOM_CREA_2
(6) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(7) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_QUERY
	cmd_seq	C_PLUS_CLOM_QUERY
(8) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLOM_QU5
	cmd_seq	M_PLUS_CLOM_QU5
(9) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLOM_QU51
	cmd_seq	M_PLUS_CLOM_QU51
(10) ACI_CMD_IND		
	cmd_len	LM_PLUS_CLOM_QU52
	cmd_seq	M_PLUS_CLOM_QU52
(11) ACI_CMD_IND		
	cmd_len	LM_OK
	cmd_seq	M_OK
(12) ACI_CMD_REQ		
	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_ON_0
	cmd_seq	C_PLUS_CLOM_ON_0

History: 01.11.02 RM Initial

#### 4.2.7 ACILC025: Set Periodic Update of Location Service (CLPS, failed, is to rebuilt)

Description: switch on/off Periodic Update Flag, set/reset Periodic Update for clients, set periodic update value

Preamble:

ACILC011

	APL	ACI	PS
(1)	ACI_CMD_REQ		
	(cmd: AT+CLOM ON)		
	* =====> *		
(2)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(3)	ACI_CMD_REQ		
	(cmd: AT+CLOM CR)		
	* =====> *		
(4)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(5)	ACI_CMD_REQ		
	(cmd: AT+CLOM CR)		
	* =====> *		
(6)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(7)	ACI_CMD_REQ		
	(cmd: AT+CLOM CR)		
	* =====> *		
(8)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(9)	ACI_CMD_REQ		
	(cmd: AT+CLPS ON)		
	* =====> *		
(10)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(11)	ACI_CMD_REQ		
	(cmd: AT+CLPS OFF)		
	* =====> *		
(12)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(13)	ACI_CMD_REQ		
	(cmd: AT+CLPS ON1)		
	* =====> *		
(14)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(15)	ACI_CMD_REQ		
	(cmd: AT+CLPS ON2)		
	* =====> *		
(16)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(17)	ACI_CMD_REQ		
	(cmd: AT+CLPS ON3)		
	* =====> *		
(18)	ACI_CMD_IND		
	(msg: OK)		

```

* <===== *
(19) |      ACI_CMD_REQ      |
    |      (cmd: AT+CLPS OF2)      |
    * =====> *
(20) |      ACI_CMD_IND      |
    |      (msg: OK)      |
    * <===== *
(21) |      ACI_CMD_REQ      |
    |      (cmd: AT+CLPS?)      |
    * =====> *
(22) |      ACI_CMD_IND      |
    |      (msg: +CLPS:...)      |
    * <===== *
(23) |      ACI_CMD_IND      |
    |      (msg: +CLPS:...)      |
    * <===== *
(24) |      ACI_CMD_IND      |
    |      (msg: +CLPS:...)      |
    * <===== *
(25) |      ACI_CMD_IND      |
    |      (msg: +CLPS:...)      |
    * <===== *
    |

```

## Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_ON_ALL
	cmd_seq	C_PLUS_CLOM_ON_ALL
(2) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK
(3) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_CREA_1
	cmd_seq	C_PLUS_CLOM_CREA_1
(4) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK
(5) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_CREA_2
	cmd_seq	C_PLUS_CLOM_CREA_2
(6) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK
(7) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_CREA_3
	cmd_seq	C_PLUS_CLOM_CREA_3

(8) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(9) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_ON_ALL C_PLUS_CLPS_ON_ALL
(10) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(11) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_OFF_ALL C_PLUS_CLPS_OFF_ALL
(12) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(13) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_ON_1 C_PLUS_CLPS_ON_1
(14) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(15) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_ON_2 C_PLUS_CLPS_ON_2
(16) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(17) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_ON_3 C_PLUS_CLPS_ON_3
(18) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(19) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_OFF_2 C_PLUS_CLPS_OFF_2
(20) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(21) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLPS_QUERY C_PLUS_CLPS_QUERY

(22) ACI_CMD_IND	cmd_len	LM_PLUS_CLPS_QU1
	cmd_seq	M_PLUS_CLPS_QU1
(23) ACI_CMD_IND	cmd_len	LM_PLUS_CLPS_QU11
	cmd_seq	M_PLUS_CLPS_QU11
(24) ACI_CMD_IND	cmd_len	LM_PLUS_CLPS_QU12
	cmd_seq	M_PLUS_CLPS_QU12
(25) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK

History:                    15.10.02            RM            Initial

#### **4.2.8 ACILC026: Set Privacy Parameter for LC client (CLSR)**

Description: set/reset notify and confirmation state for LC clients

## Preamble:

### ACILC011

	APL	ACI	PS
(1)	ACI_CMD_REQ		
	(cmd: AT+CLOM ON)		
	* =====> *		
(2)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(3)	ACI_CMD_REQ		
	(cmd: AT+CLOM CR)		
	* =====> *		
(4)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(5)	ACI_CMD_REQ		
	(cmd: AT+CLOM CR)		
	* =====> *		
(6)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(7)	ACI_CMD_REQ		
	(cmd: AT+CLOM CR)		
	* =====> *		
(8)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(9)	ACI_CMD_REQ		
	(cmd: AT+CLRS NNNC)		
	* =====> *		
(10)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(11)	ACI_CMD_REQ		
	(cmd: AT+CLRS NNC)		
	* =====> *		
(12)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		
(13)	ACI_CMD_REQ		
	(cmd: AT+CLRS NC)		
	* =====> *		
(14)	ACI_CMD_IND		
	(msg: OK)		
	* <===== *		

## Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CLOM_ON_ALL
	cmd_seq	C_PLUS_CLOM_ON_ALL

(2) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(3) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_CREA_1 C_PLUS_CLOM_CREA_1
(4) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(5) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_CREA_2 C_PLUS_CLOM_CREA_2
(6) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(7) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLOM_CREA_3 C_PLUS_CLOM_CREA_3
(8) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(9) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLSR_2_NNNC C_PLUS_CLSR_2_NNNC
(10) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(11) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLSR_1_NNC C_PLUS_CLSR_1_NNC
(12) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK
(13) ACI_CMD_REQ	cmd_src cmd_len cmd_seq	CMD_SRC_EXT LC_PLUS_CLSR_3_NC C_PLUS_CLSR_3_NC
(14) ACI_CMD_IND	cmd_len cmd_seq	LM_OK M_OK



History:	25.10.02	RM	Initial
----------	----------	----	---------

#### 4.2.9 ACILC027: Request position data, send data in SMS to MLC destination address.(CLSA, failed currently)

Description:

Preamble:

ACILC021B			PS
APL		ACI	
TIMEOUT(1000)			
(1)		MNLC_SMS_MEAS_CNF	
		* <=====	*
(2)	ACI_CMD_IND		
	(msg: CLPS)		
		* <=====	*
(3)	ACI_CMD_IND		
	(msg: OK)		
		* <=====	*
(4)		MNSMS_SUBMIT_REQ	
		* =====>	*
(5)		MNSMS_REPORT_IND	
		* <=====	*
(6)	ACI_CMD_IND		
	(msg: CMGS)		
		* <=====	*
(7)	ACI_CMD_IND		
	(msg: OK)		
		* <=====	*

### Parametrization:

Primitive	Parameter	Value
(1) MNLC_SMS_MEAS_CNF	reference	NUM_1
	data_valid	VALID
	sms_text	POS_DATA1
(2) ACI_CMD_IND	cmd_src	CMD_SRC_EXT
	cmd_len	LM_CLPS_POS1
	cmd_seq	M_CLPS_POS1
(3) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK

History: 25.10.02 RM Initial