

GSM Protocol Stack

Test Specification RRLP

Author:

Condat AG
Alt-Moabit 90a
D-10559 Berlin
Germany

Date:

27-sep - 2002

Document No.:

8443.403.02.001

File:

rrlp.doc

Confidential

Table of Contents

0	DOCUMENT CONTROL	3
0.1	Document History	3
0.2	References	3
0.3	Abbreviations	4
0.4	Terms	4
1	OVERVIEW	5
2	PARAMETERS	6
3	TEST CASES	21
3.1	Internal Routing	21
3.1.1	RRLP000: Configure internal routing and PCO view	21
3.2	Position Request	22
3.2.1	RRLP001: RRLP: Position calculation	22
3.2.2	RRLP002: RRLP: Position calculation	23
3.2.3	RRLP003: RRLP: Position calculation	24
3.2.4	RRLP004: RRLP: Position calculation	25
3.2.5	RRLP005: RRLP: Assistance data consecutive	26
3.2.6	RRLP006: RRLP: Position calculation	27
3.2.7	RRLP007: RRLP: Position calculation	28
3.2.8	RRLP008: RRLP: Assistance data Erroneous	29
3.2.9	RRLP009: RRLP: Assistance data Consecutive2	30
3.2.10	RRLP0010: RRLP: Assistance data Erroneous2	31

0 Document Control

© Copyright Condat AG, 2000-2002.

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 AG 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 AG. Condat AG 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 AG.

Condat AG
Alt Moabit 90a
10559 Berlin
Germany

Telephone: +49.30.3949-0
Fax: +49.30.3949-1300
Internet: <http://www.condat.de>
E-mail: gsm@condat.de

0.1 Document History

Document Id.	Date	Author	Remarks
	30-Sep-2002	MNC	Initial
	01-Nov-2002	TML	Made SDU def. for encoding of APDUs in RRLP entity
	06-Nov-2002	TML	Changed test cases to comply with RRLP design specification

0.2 References

- [3GPP_03.71] Digital cellular telecommunications system (Phase 2+), Location services (LCS), (Functional description) - Stage 2, (V8.5.0 Release 1999)
- [3GPP_04.18] 3rd Generation Partnership Project, Technical Specification Group GSM/EDGE Radio Access Network; Mobile radio interface layer 3 specification, Radio Resource Control Protocol, (V8.13.0 Release 1999)
- [3GPP_04.31] 3rd Generation Partnership Project, Technical Specification Group GSM EDGE Radio Access Network; Location Services (LCS); Mobile Station (MS) – Serving Mobile Location Centre (SMLC); Radio Resource LCS Protocol (RRLP), (V8.8.0 Release 1999)
- [3GPP_24.007] 3rd Generation Partnership Project, Technical Specification Group Core Network; Mobile radio interface signalling layer 3; (V3.8.0 Release 1999)
- [3GPP_24.030] 3rd Generation Partnership Project, Technical Specification Group Core Network; Location Services (LCS); Supplementary service operations - Stage 3), (V3.3.0 Release 1999)
- [3GPP_22.071] 3rd Generation Partnership Project, Technical Specification Group services and System aspects; Location services (LCS); Service description; Stage 1 (V3.4.0 Release 1999)
- <http://www.fcc.gov/911/enhanced/> For further information of E911 requirements.

- [X.209] ITU-T Recommendation X.209: "Specification of Abstract Syntax Notation One (ASN.1)
- [CU 0107] CPS - GSM Cursor™ Mobile Terminal Requirements, Full Specification – Issue 4.2
- [CU 0115] CPS - GSM Cursor™ Mobile Terminal Requirements, MMI Specification – Issue 1.1
- [CU 0126] CPS - GSM Cursor™ Mobile Terminal Requirements, AT Commands for Feature Control – Issue 3.3

[CU 0149]	CPS - GSM Cursor™ Enhanced Observed Time Difference, (Software implementation project) – Issue 3.0
[CU 0305]	CPS - GSM Cursor™ Mobile Terminal Requirements, Cursor™ Task Bridge Description – Issue 1.3
[TI S922]	L1/RR interface for neighbour cells monitoring (Idle mode) and EOTD
[8353.507]	UPLCS Requirements for Location Services in a UE – Requirement Analysis
[4711.005.02]	G23-GPRS Protocol Stack Release 1.4.0, Product Specification, May 2002
[4711.020.02]	G23-GPRS Protocol, E-OTD Lower Level Design LC, August 2002
[4711.021.02]	G23-GPRS Protocol, E-OTD Lower Level Design RR, August 2002
[TI 7010.801]	7010.801, References and Vocabulary, Texas Instruments
[C_7010.801]	7010.801, References and Vocabulary, Condat AG

0.3 Abbreviations

ACI	Application Control Interface (AT Commands)
AL	Application Layer
APDU	Application Protocol Data Unit
CBCH	Common Broadcast Channel
CPS	Cambridge Positioning System
DTAP	Direct Transfer Application Part
E-OTD	Enhanced Observed Time Difference positioning method
LCS	Location Services
LC	Location Service handling Entity
LR	Location Request
NA-ESRD	North American Emergency Services Routing Digits
NA-ESRK	North American Emergency Services Routing Key
PS	Protocol Stack
RRLP	Radio Resource Location Service Protocol
SMLC	Serving Mobile Location Center

0.4 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).

1 Overview

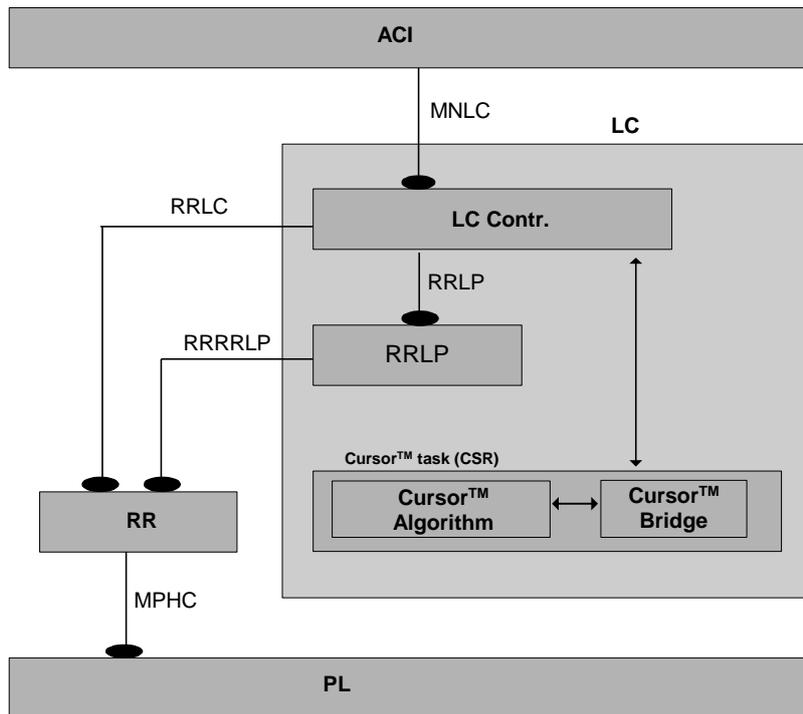


Figure 1: LC details

Figure 1 shows the details of the Cursor™ task as provided by CPS. It consists of two parts, the Cursor™ Algorithm and the Cursor™ Bridge code. To have a more generic approach for further developments, the LC controller co-ordinates the positioning requests to the appropriate implementation, e.g. E-OTD or GPS. Texas Instruments will develop the LC controller and the RRLP. The Cursor™ Bridge code is available in source code, representing the interface between the LC controller and the Cursor™ Algorithm. It is used to exemplify the data handling between the modules but can be adapted to the Texas Instruments needs. The Cursor™ Algorithm module is provided in object code, receiving the correlation data and BTS identifier and calculates precise OTD values. The Cursor™ Algorithm remains property of CPS and will not be part of the Protocol Stack software release. The data handling is taking place inside RRLP or ACI (only during test phase 1 & 2 or for further implementations of assistance data). All point-to-point data, e.g. assistance data from the network and the location response from the MS, are handled by RRLP. General positioning data, e.g. assistance data to all mobiles in a cell, might be distributed via CBCH and is then transported via ACI to the LC. The interface between LC controller and RRLP is primitive based with functional character in order to run RRLP in another task context, e.g. RR.

2 Parameters

```
/* DECLARATION */
DECLARATION (EOTD_SC_RES)           /* */
DECLARATION (EOTD_SC_RES1)         /* */
DECLARATION (EOTD_NC_RES_1)        /* */
DECLARATION (EOTD_NC_RES_2)        /* */
DECLARATION (EOTD_NC_RES_3)        /* */
DECLARATION (EOTD_NC_RES_4)        /* */
DECLARATION (EOTD_NC_RES_5)        /* */
DECLARATION (EOTD_NC_RES_6)        /* */
DECLARATION (EOTD_NC_RES_7)        /* */
DECLARATION (EOTD_NC_RES_8)        /* */
DECLARATION (EOTD_NC_RES_9)        /* */
DECLARATION (EOTD_NC_RES_10)       /* */
DECLARATION (EOTD_NC_RES_11)       /* */
DECLARATION (EOTD_NC_RES_12)       /* */
DECLARATION (ASSIST_DATA_1)         /* */
DECLARATION (ASSIST_DATA_2)         /* */
DECLARATION (ASSIST_DATA_3)         /* */
DECLARATION (ASSIST_DATA_4)         /* */
DECLARATION (ASSIST_DATA_5)         /* */
DECLARATION (ASSIST_DATA_6)         /* */
DECLARATION (ASSIST_DATA_7)         /* */
DECLARATION (ASSIST_DATA_8)         /* */
DECLARATION (ASSIST_DATA_9)         /* */
DECLARATION (ASSIST_DATA_10)        /* */
DECLARATION (ASSIST_DATA_11)        /* */
DECLARATION (ASSIST_DATA_12)        /* */
DECLARATION (ASSIST_DATA_13)        /* */
DECLARATION (ASSIST_DATA_14)        /* */
DECLARATION (ASSIST_DATA_15)        /* */
DECLARATION (MAX_REF_NEIGH_BTS)     /* */
DECLARATION (REF_BTS_DATA)          /* */
DECLARATION (REF_BTS_DATA_1)        /* */
DECLARATION (REF_BTS_DATA_2)        /* */
DECLARATION (REF_BTS_DATA_3)        /* */
DECLARATION (BTS_DATA)
DECLARATION (BTS_DATA_1)
DECLARATION (BTS_DATA_2)
DECLARATION (BTS_DATA_3)
DECLARATION (BTS_DATA_4)
DECLARATION (BTS_DATA_5)
DECLARATION (BTS_DATA_6)
DECLARATION (BTS_DATA_7)
DECLARATION (BTS_DATA_8)
DECLARATION (BTS_DATA_9)
DECLARATION (BTS_DATA_10)
DECLARATION (BTS_DATA_11)
DECLARATION (BTS_DATA_12)
DECLARATION (BTS_DATA_13)
DECLARATION (BTS_DATA_14)
DECLARATION (BTS_DATA_15)
DECLARATION (EXOR_NO)               /* */
DECLARATION (MNC)                   /* */
DECLARATION (MCC)                   /* */
DECLARATION (EOTD_NC_RES)           /* */
```

```

DECLARATION (ASSIST_DATA)          /* */
DECLARATION (SDU)                   /* */
DECLARATION (SDU_1)                 /* */
DECLARATION (TIMING_DATA)          /* */
DECLARATION (RRLP_POS_REQ)
DECLARATION (RRLP_ASSIST_DATA)
DECLARATION (RRLP_ERROR)
DECLARATION (RRLP_POS_REQ_NEW_REF)
DECLARATION (RRLP_POS_REQ_ERR_P)
DECLARATION (RRLP_POS_REQ_NEW_REF_ERR_P)
DECLARATION (RRLP_POS_REQ_OLD_REF)
DECLARATION (RRLP_ASSIST_DATA_M)
DECLARATION (RRLP_ASSIST_DATA_F)
DECLARATION (RRLP_ASSIST_DATA_ERR_P)
DECLARATION (RRLP_ASSIST_ACK)
DECLARATION (RRLP_ASSIST_DATA_S)

/* Constants */
/* CR */
/*
   Table 1/3GPP TS 04.06: C/R field bit usage
   Type          Direction          C/R value
   Command       BS side to MS side  1
                 MS side to BS side  0
   Response      BS side to MS side  0
                 MS side to BS side  1
*/
BYTE CR_BS_CMD          0x01
BYTE CR_MS_CMD          0x00
BYTE CR_BS_RES          0x00
BYTE CR_MS_RES          0x01

/* Location method error cause */
BYTE LC_METH_ERROR      0x07

/* Serving cell different than ref. BTS */
BYTE LC_REF_BTS_ERROR   0x0A

/* The ms is in a handover situation */
BYTE LC_HANDOVER_INFO   0x14

/* tav */
BYTE TAV                0xFF          /* NO TAV */

/* loc_method */
BYTE LOC_METHOD          0x00          /* */

/* pos_method */
BYTE POS_METHOD          0x00          /* */

/* bsic */
BYTE BSIC                0            /* Base Station Code */

/* arfcn */
SHORT ARFCN              0            /* Channal number */

/* fn */

```

```

LONG   FN           0           /* TIME_TAG */

/* rssi */
LONG   RSSI        0           /* RSSI */

/* time tag */
LONG   TIME_TAG    0           /* TIME_TAG */

/* lac */
SHORT  LAC         0           /*Location Area Code */

/* l_buf */
SHORT  L_BUF       0           /* Length in bits*/

/* o_buf */
SHORT  O_BUF       0           /* Offset in bits*/

/* cell_id */
SHORT  CELL_ID     0           /*Cell identification */

/* req_id */
SHORT  REQ_ID      0           /* */

/* mfrm_offset*/
BYTE   MFRM_OFFSET 0           /* Offset in multiframes between the Scell and the
                                speceficed cell */

/* buf */
BYTE   BUF         0           /* */

/* reference_relation */
BYTE   REFERENCE_RELATION 0     /* */

/* sb_flag */
BYTE   SB_FLAG     0           /* TRUE */

/* time_slot */
BYTE   TIME_SLOT   0           /* */

/* std_resolution */
BYTE   STD_RESOLUTION 0         /* */

/*
 */
BYTE   NUM_MEASUREMENT_SETS    0     /* */

/*
 */
BYTE   NUM_MEASUREMENTS       0     /* */

/* num_reference_cells*/
BYTE   NUM_REFERENCE_CELLS    0     /* */

/* cause*/
BYTE   RRLC_ERROR_IND_CAUSE   0     /* */

/* old type*/

```

```

BYTE   OTD_TYPE      0          /* ROUGH_RTD */

/* uncertainty */
BYTE   UNCERTAINTY  0          /* Max 2 Bit */

/* ta_correction_present */
BYTE   TA_CORRECTION_PRESENT      0          /* */

/* quality */
BYTE   QUALITY      0          /* */

/* timing_offset */
BYTE   TIMING_OFFSET      0          /* */

/* total_neigh_bts */
BYTE   TOTAL_NEIGH_BTS      0          /* */

/* frame_number */
BYTE   FRAME_NUMBER      0          /* */

/* ta_correction */
LONG   TA_CORRECTION      0          /* */

/* timing_advance */
LONG   TIMING_ADVANCE      0          /* TIME_TAG */

/* exp_otp */
SHORT  EXP_OTD      0          /* Expected OTD from network */

/* rough_rtd */
SHORT  ROUGH_RTD      0          /* Rough RTD from network */

/* max_ref_neigh_bts */
BEGINARRAY(MAX_REF_NEIGH_BTS,15)          /* */
0x01,0x02,0x03,0x04,0x05,
0x06,0x07,0x08,0x09,0x0A,
0x0B,0x0C,0x0D,0x0E,0x0F
ENDARRAY

/* mnc */
BEGINARRAY(MNC,3)          /* Mobile network code */
0x01,0x02,0x03
ENDARRAY

/* mcc */
BEGINARRAY(MCC,3)          /* Mobile country code */
0x01,0x02,0x03
ENDARRAY

BEGIN_PSTRUCT ("ref_bts_data",REF_BTS_DATA_1)
    SET_COMP ("cell_id",CELL_ID)          /* */
    SET_COMP ("lac",LAC)          /* */
    SET_COMP ("quality",QUALITY)          /* */
    SET_COMP ("num_measurements",NUM_MEASUREMENTS)          /* */
    SET_COMP ("timing_offset",TIMING_OFFSET)          /* */
    SET_COMP ("total_neigh_bts",TOTAL_NEIGH_BTS)          /* */
    
```

```
SET_COMP ("bts_data",BTS_DATA) /* */
SET_COMP ("frame_number",FRAME_NUMBER) /* */
SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("ref_bts_data",REF_BTS_DATA_2)
SET_COMP ("cell_id",CELL_ID) /* */
SET_COMP ("lac",LAC) /* */
SET_COMP ("quality",QUALITY) /* */
SET_COMP ("num_measurements",NUM_MEASUREMENTS) /* */
SET_COMP ("timing_offset",TIMING_OFFSET) /* */
SET_COMP ("total_neigh_bts",TOTAL_NEIGH_BTS) /* */
SET_COMP ("bts_data",BTS_DATA) /* */
SET_COMP ("frame_number",FRAME_NUMBER) /* */
SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("ref_bts_data",REF_BTS_DATA_3)
SET_COMP ("cell_id",CELL_ID) /* */
SET_COMP ("lac",LAC) /* */
SET_COMP ("quality",QUALITY) /* */
SET_COMP ("num_measurements",NUM_MEASUREMENTS) /* */
SET_COMP ("timing_offset",TIMING_OFFSET) /* */
SET_COMP ("total_neigh_bts",TOTAL_NEIGH_BTS) /* */
SET_COMP ("bts_data",BTS_DATA) /* */
SET_COMP ("frame_number",FRAME_NUMBER) /* */
SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_STRUCT_ARRAY (REF_BTS_DATA, 3) /* REF_BTS_DATA*/
REF_BTS_DATA_1,
REF_BTS_DATA_2,
REF_BTS_DATA_3
```

ENDARRAY

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_1)
SET_COMP ("bsic",CELL_ID) /* */
SET_COMP ("arfcn",LAC) /* */
SET_COMP ("quality",QUALITY) /* */
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
SET_COMP ("timing_offset", TIMING_OFFSET) /* */
SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_2)
SET_COMP ("bsic",CELL_ID) /* */
SET_COMP ("arfcn",LAC) /* */
SET_COMP ("quality",QUALITY) /* */
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
```

```
SET_COMP ("timing_offset", TIMING_OFFSET) /* */  
SET_COMP ("timing_advance", TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data", BTS_DATA_3)  
SET_COMP ("bsic", CELL_ID) /* */  
SET_COMP ("arfcn", LAC) /* */  
SET_COMP ("quality", QUALITY) /* */  
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */  
SET_COMP ("timing_offset", TIMING_OFFSET) /* */  
SET_COMP ("timing_advance", TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data", BTS_DATA_4)  
SET_COMP ("bsic", CELL_ID) /* */  
SET_COMP ("arfcn", LAC) /* */  
SET_COMP ("quality", QUALITY) /* */  
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */  
SET_COMP ("timing_offset", TIMING_OFFSET) /* */  
SET_COMP ("timing_advance", TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data", BTS_DATA_5)  
SET_COMP ("bsic", CELL_ID) /* */  
SET_COMP ("arfcn", LAC) /* */  
SET_COMP ("quality", QUALITY) /* */  
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */  
SET_COMP ("timing_offset", TIMING_OFFSET) /* */  
SET_COMP ("timing_advance", TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data", BTS_DATA_6)  
SET_COMP ("bsic", CELL_ID) /* */  
SET_COMP ("arfcn", LAC) /* */  
SET_COMP ("quality", QUALITY) /* */  
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */  
SET_COMP ("timing_offset", TIMING_OFFSET) /* */  
SET_COMP ("timing_advance", TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data", BTS_DATA_7)  
SET_COMP ("bsic", CELL_ID) /* */  
SET_COMP ("arfcn", LAC) /* */  
SET_COMP ("quality", QUALITY) /* */  
SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */  
SET_COMP ("timing_offset", TIMING_OFFSET) /* */  
SET_COMP ("timing_advance", TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_8)
  SET_COMP ("bsic",CELL_ID)           /* */
  SET_COMP ("arfcn",LAC)              /* */
  SET_COMP ("quality",QUALITY)        /* */
  SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
  SET_COMP ("timing_offset", TIMING_OFFSET) /* */
  SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_9)
  SET_COMP ("bsic",CELL_ID)           /* */
  SET_COMP ("arfcn",LAC)              /* */
  SET_COMP ("quality",QUALITY)        /* */
  SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
  SET_COMP ("timing_offset", TIMING_OFFSET) /* */
  SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_10)
  SET_COMP ("bsic",CELL_ID)           /* */
  SET_COMP ("arfcn",LAC)              /* */
  SET_COMP ("quality",QUALITY)        /* */
  SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
  SET_COMP ("timing_offset", TIMING_OFFSET) /* */
  SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_11)
  SET_COMP ("bsic",CELL_ID)           /* */
  SET_COMP ("arfcn",LAC)              /* */
  SET_COMP ("quality",QUALITY)        /* */
  SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
  SET_COMP ("timing_offset", TIMING_OFFSET) /* */
  SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_12)
  SET_COMP ("bsic",CELL_ID)           /* */
  SET_COMP ("arfcn",LAC)              /* */
  SET_COMP ("quality",QUALITY)        /* */
  SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
  SET_COMP ("timing_offset", TIMING_OFFSET) /* */
  SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_13)
    SET_COMP ("bsic",CELL_ID)           /* */
    SET_COMP ("arfcn",LAC)              /* */
    SET_COMP ("quality",QUALITY)        /* */
    SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
    SET_COMP ("timing_offset", TIMING_OFFSET) /* */
    SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_14)
    SET_COMP ("bsic",CELL_ID)           /* */
    SET_COMP ("arfcn",LAC)              /* */
    SET_COMP ("quality",QUALITY)        /* */
    SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
    SET_COMP ("timing_offset", TIMING_OFFSET) /* */
    SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("bts_data",BTS_DATA_15)
    SET_COMP ("bsic",CELL_ID)           /* */
    SET_COMP ("arfcn",LAC)              /* */
    SET_COMP ("quality",QUALITY)        /* */
    SET_COMP ("num_measurements", NUM_MEASUREMENTS) /* */
    SET_COMP ("timing_offset", TIMING_OFFSET) /* */
    SET_COMP ("timing_advance",TIMING_ADVANCE) /* */
```

ENDSTRUCT

```
BEGIN_STRUCT_ARRAY (BTS_DATA, 15)
    BTS_DATA_1,
    BTS_DATA_2,
    BTS_DATA_3,
    BTS_DATA_4,
    BTS_DATA_5,
    BTS_DATA_6,
    BTS_DATA_7,
    BTS_DATA_8,
    BTS_DATA_9,
    BTS_DATA_10,
    BTS_DATA_11,
    BTS_DATA_12,
    BTS_DATA_13,
    BTS_DATA_14,
    BTS_DATA_15
```

ENDARRAY

```
BEGIN_PSTRUCT ("assist_data",ASSIST_DATA_1)
    SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
    SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
    SET_COMP ("otd_type",OTD_TYPE)     /* */
    SET_COMP ("exp_otd",EXP_OTD)       /* */
    SET_COMP ("uncertainty",UNCERTAINTY) /* */
```

```
        SET_COMP ("rough_rtd",ROUGH_RTD)           /* */

ENDSTRUCT

BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_2)
    SET_COMP ("arfcn",ARFCN)                       /* Channal Number*/
    SET_COMP ("bsic",BSIC)                         /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET)           /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)                   /* */
    SET_COMP ("uncertainty",UNCERTAINTY)           /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)               /* */

ENDSTRUCT

BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_3)
    SET_COMP ("arfcn",ARFCN)                       /* Channal Number*/
    SET_COMP ("bsic",BSIC)                         /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET)           /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)                   /* */
    SET_COMP ("uncertainty",UNCERTAINTY)           /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)               /* */

ENDSTRUCT

BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_4)
    SET_COMP ("arfcn",ARFCN)                       /* Channal Number*/
    SET_COMP ("bsic",BSIC)                         /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET)           /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)                   /* */
    SET_COMP ("uncertainty",UNCERTAINTY)           /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)               /* */

ENDSTRUCT

BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_5)
    SET_COMP ("arfcn",ARFCN)                       /* Channal Number*/
    SET_COMP ("bsic",BSIC)                         /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET)           /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)                   /* */
    SET_COMP ("uncertainty",UNCERTAINTY)           /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)               /* */

ENDSTRUCT

BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_6)
    SET_COMP ("arfcn",ARFCN)                       /* Channal Number*/
    SET_COMP ("bsic",BSIC)                         /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET)           /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)                   /* */
    SET_COMP ("uncertainty",UNCERTAINTY)           /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)               /* */

ENDSTRUCT
```

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_7)
    SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
    SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)       /* */
    SET_COMP ("uncertainty",UNCERTAINTY) /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_8)
    SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
    SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)       /* */
    SET_COMP ("uncertainty",UNCERTAINTY) /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_9)
    SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
    SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)       /* */
    SET_COMP ("uncertainty",UNCERTAINTY) /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_10)
    SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
    SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)       /* */
    SET_COMP ("uncertainty",UNCERTAINTY) /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_11)
    SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
    SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
    SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
    SET_COMP ("otd_type", OTD_TYPE) /* */
    SET_COMP ("exp_otd",EXP_OTD)       /* */
    SET_COMP ("uncertainty",UNCERTAINTY) /* */
    SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_12)
  SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
  SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
  SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
  SET_COMP ("otd_type", OTD_TYPE) /* */
  SET_COMP ("exp_otd",EXP_OTD)       /* */
  SET_COMP ("uncertainty",UNCERTAINTY) /* */
  SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_13)
  SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
  SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
  SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
  SET_COMP ("otd_type", OTD_TYPE) /* */
  SET_COMP ("exp_otd",EXP_OTD)       /* */
  SET_COMP ("uncertainty",UNCERTAINTY) /* */
  SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_14)
  SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
  SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
  SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
  SET_COMP ("otd_type", OTD_TYPE) /* */
  SET_COMP ("exp_otd",EXP_OTD)       /* */
  SET_COMP ("uncertainty",UNCERTAINTY) /* */
  SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_PSTRUCT ("assist_data", ASSIST_DATA_15)
  SET_COMP ("arfcn",ARFCN)           /* Channal Number*/
  SET_COMP ("bsic",BSIC)             /* Base Station Identity Code*/
  SET_COMP ("mfrn_offset",MFRM_OFFSET) /* */
  SET_COMP ("otd_type", OTD_TYPE) /* */
  SET_COMP ("exp_otd",EXP_OTD)       /* */
  SET_COMP ("uncertainty",UNCERTAINTY) /* */
  SET_COMP ("rough_rtd",ROUGH_RTD)   /* */
```

ENDSTRUCT

```
BEGIN_STRUCT_ARRAY (ASSIST_DATA, 15)           /* ASSIST_DATA */
  ASSIST_DATA_1,
  ASSIST_DATA_2,
  ASSIST_DATA_3,
  ASSIST_DATA_4,
  ASSIST_DATA_5,
  ASSIST_DATA_6,
  ASSIST_DATA_7,
```

```
        ASSIST_DATA_8,  
        ASSIST_DATA_9,  
        ASSIST_DATA_10,  
        ASSIST_DATA_11,  
        ASSIST_DATA_12,  
        ASSIST_DATA_13,  
        ASSIST_DATA_14,  
        ASSIST_DATA_15  
ENDARRAY  
  
BEGIN_PSTRUCT ("sdu", SDU_1) /* */  
    SET_COMP ("l_buf", L_BUF) /* */  
    SET_COMP ("o_buf", O_BUF) /* */  
    SET_COMP ("buf", BUF) /* */  
  
ENDSTRUCT  
  
BEGIN_STRUCT_ARRAY (SDU, 1) /* NC Measurements*/  
    SDU_1  
ENDARRAY  
  
BEGIN_PSTRUCT ("timing_data", TIMING_DATA)  
    SET_COMP ("mcc", MCC) /* */  
    SET_COMP ("mnc", MNC) /* */  
    SET_COMP ("reference_relation", REFERENCE_RELATION) /* */  
    SET_COMP ("time_slot", TIME_SLOT) /* */  
    SET_COMP ("std_resolution", STD_RESOLUTION) /* */  
    SET_COMP ("num_measurement_sets", NUM_MEASUREMENT_SETS) /* */  
    SET_COMP ("num_reference_cells", NUM_REFERENCE_CELLS) /* */  
    SET_COMP ("ta_correction_present", TA_CORRECTION_PRESENT) /* */  
    SET_COMP ("ta_correction", TA_CORRECTION) /* */  
    SET_COMP ("ref_bts_data", REF_BTS_DATA) /* */  
  
ENDSTRUCT
```

```
/* ASN1 RRLP message identifiers used for build correspondent message in RRLP entity. */
/* Note: the compiler switch RRLP_TDS has to be defined if an ASN.1 message has to be encoded in the
/* RRLP entity */
/* Where nothing else is noted the ref. number 2 is used */
/* -----"Default" meessages -----*/

/* RRLP_ASN1_RRLP_COMPONENT__MSR_POS_REQ__TAG = 0x0,*/
/* Enable this def. to encode Asn.1 message in RRLP entity */
/*
SET_BITBUF( "sdu", RRLP_POS_REQ, 0x20)
    0x00
ENDBITBUF
*/
/*0x21, 0x88, 0x19, 0x2d,0x70,0x62,0xb0,0x41,0x8f,0x38,0x57,0x58,0x30,0x54,0x80*/
SET_BITBUF( "sdu", RRLP_POS_REQ, 0x78)
    0x20, 0xC4, 0x0C, 0x96, 0xB8, 0x31, 0x58, 0x20, 0xC7, 0x9C, 0x2B, 0xAC, 0x18, 0x2A, 0x40
ENDBITBUF

/* RRLP_ASN1_RRLP_COMPONENT__ASSISTANCE_DATA__TAG = 0x2 */
/* More to come flag not set - single element */
/* Enable this def. to encode Asn.1 message in RRLP entity */
/*
SET_BITBUF( "sdu", RRLP_ASSIST_DATA, 0x20)
    0x02
ENDBITBUF
*/
SET_BITBUF( "sdu", RRLP_ASSIST_DATA, 0x10)
    0x44, 0x00
ENDBITBUF

/* RRLP_ASN1_RRLP_COMPONENT__ASSISTANCE_DATA_ACK__TAG = 0x3 */
/* Enable this def. to encode Asn.1 message in RRLP entity */
/*
SET_BITBUF( "sdu", RRLP_ASSIST_ACK, 0x20)
    0x0C
ENDBITBUF
*/
SET_BITBUF( "sdu", RRLP_ASSIST_ACK, 0x08)
    0x46
ENDBITBUF

/* RRLP_ASN1_RRLP_COMPONENT__PROTOCOL_ERROR__TAG = 0x4 */
/* Enable this def. to encode Asn.1 message in RRLP entity */
SET_BITBUF( "sdu", RRLP_ERROR, 0x20)
    0x04
ENDBITBUF
```

```
/*  
SET_BITBUF( "sdu", RRLP_ERROR, 0x18)  
    0x01,0x40,0x00  
ENDBITBUF  
*/
```

```
/* -----RRLP message identifiers used for exception cases-----*/
```

```
/* RRLP reference number newer than default = 0x5 */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
/*
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_NEW_REF, 0x20)  
    0x05  
ENDBITBUF  
*/
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_NEW_REF, 0x18)  
    0x60, 0x00, 0x00  
ENDBITBUF
```

```
/* RRLP position request with an error in a component = 0x6 */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
/*
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_ERR_P, 0x20)  
    0x06  
ENDBITBUF  
*/
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_ERR_P, 0x20)  
    0x99  
ENDBITBUF
```

```
/* RRLP position request with reference number newer than default and an error in a component = 0x7 */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
/*
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_NEW_REF_ERR_P, 0x20)  
    0x07  
ENDBITBUF  
*/
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_NEW_REF_ERR_P, 0x20)  
    0x99  
ENDBITBUF
```

```
/* RRLP position request with reference number older than default = 0x8 */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
/*
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_OLD_REF, 0x20)  
    0x08  
ENDBITBUF  
*/
```

```
SET_BITBUF( "sdu", RRLP_POS_REQ_OLD_REF, 0x24)  
    0x20,0x00,0x00  
ENDBITBUF
```

```
/* RRLP assistant data – more to come flag set= 0x9 */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
/*  
SET_BITBUF( "sdu", RRLP_ASSIST_DATA_M, 0x20)  
    0x09  
ENDBITBUF  
*/  
SET_BITBUF( "sdu", RRLP_ASSIST_DATA_M, 0x10)  
    0x44,0x0A  
ENDBITBUF
```

```
/* RRLP assistant data – final assist data - no more to come flag set= 0xA */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
/*  
SET_BITBUF( "sdu", RRLP_ASSIST_DATA_F, 0x20)  
    0x0A  
ENDBITBUF  
*/  
SET_BITBUF( "sdu", RRLP_ASSIST_DATA_F, 0x10)  
    0x44,0x08  
ENDBITBUF
```

```
/* RRLP assistant data with error on one of the components = 0xB */  
/* Enable this def. to encode Asn.1 message in RRLP entity */  
SET_BITBUF( "sdu", RRLP_ASSIST_DATA_ERR_P, 0x20)  
    0x0B  
ENDBITBUF  
/*  
SET_BITBUF( "sdu", RRLP_ASSIST_DATA_ERR_P, 0x20)  
    0x0B  
ENDBITBUF  
*/
```

3 TEST CASES

3.1 Internal Routing

3.1.1 RRLP000: Configure internal routing and PCO view

Description: Internal routing is configured and the duplication of primitives for performing the component tests with TAP and PCO view is carried out

Preamble: None

LC	RRLP	RR
COMMAND (LC RESET)		
COMMAND (RRLP RESET)		
COMMAND (RR RESET)		
COMMAND (TAP REDIRECT CLEAR)		
COMMAND (RRLP REDIRECT CLEAR)		
COMMAND (RRLP REDIRECT RR TAP)		
COMMAND (RRLP REDIRECT LC TAP)		
COMMAND (TAP REDIRECT TAP RRLP)		

Parametrization

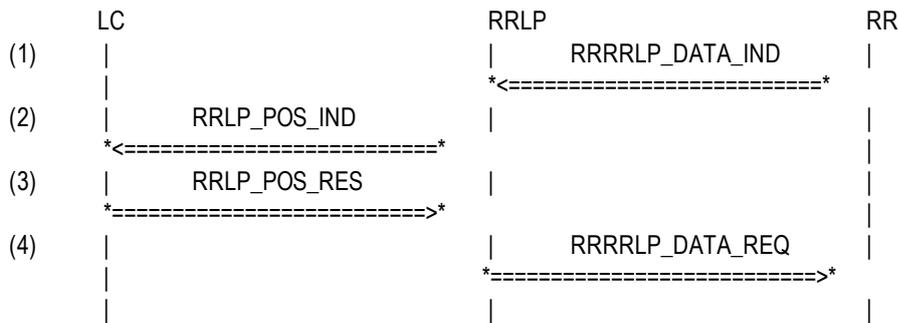
<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>	
History:	23.09.02	MNC	Initial

3.2 Position Request

3.2.1 RRLP001: RRLP: Measure position request

Description: The network starts a measure position request. The MS process the request and send the measure position response to the network.

Preamble: RRLP000



Parametrization

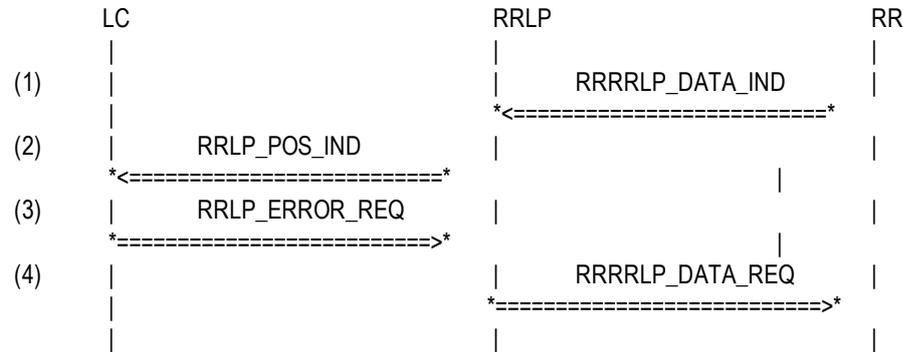
Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr	CR_BS_CMD
	sdu	RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data	NOT_USED
	loc_method	NOT_USED
	pos_method	NOT_USED
	arfcn	NOT_USED
	bsic	NOT_USED
(3) RRLP_POS_RES	timing_data	TIMING_DATA
(4) RRRRLP_DATA_REQ	cr	NOT_USED
	sdu	NOT_USED
	MNC	Initial

History: 23.09.02

3.2.2 RRLP002: RRLP: Measure position request location method Erroneous

Description: In case of non-supported location method is requested from the network then a RRLP message response is generated with a location error.

Preamble: RRLP000



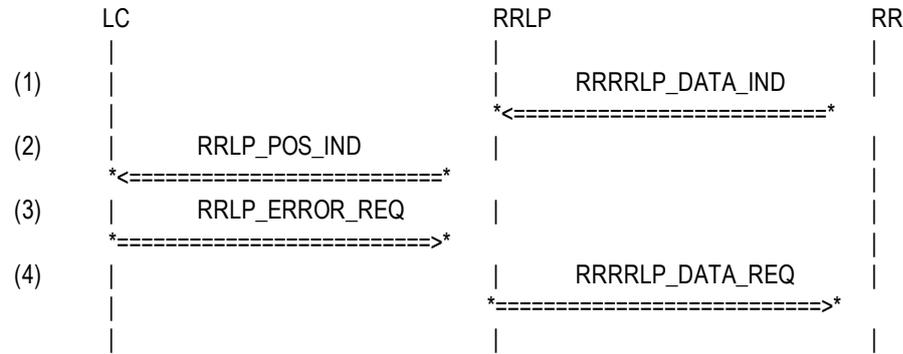
Parametrization

Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr	CR_BS_CMD
	sdu	RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data	NOT_USED
	loc_method	NOT_USED
	pos_method	NOT_USED
	arfcn	NOT_USED
	bsic	NOT_USED
(3) RRLP_ERROR_REQ	cause	LC_METH_ERROR
(4) RRRRLP_DATA_REQ	cr	CR_MS_RES
	sdu	NOT_USED
History: 23.09.02	MNC	Initial

3.2.3 RRLP003: RRLP: Measure position request serving cell erroneous

Description: In case of that the reference BTS service is different then the serving BTS then a RRLP message response is generated with a location error. The BTS comparison is done in the RR entity.

Preamble: RRLP000



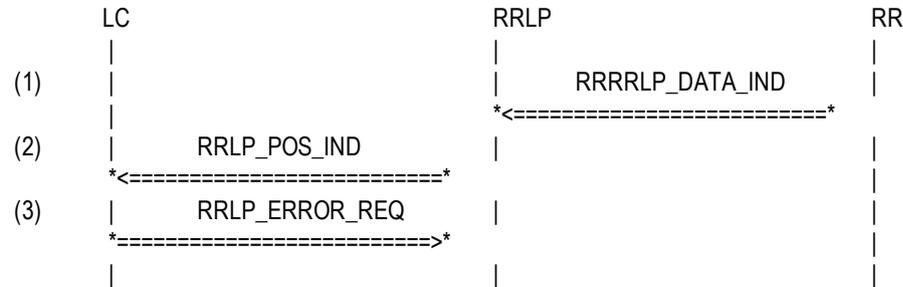
Parametrization

Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr	CR_BS_CMD
	sdu	RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data	NOT_USED
	loc_method	NOT_USED
	pos_method	NOT_USED
	arfcn	NOT_USED
	bsic	NOT_USED
(3) RRLP_ERROR_REQ	cause	LC_REF_BTS_ERROR
(4) RRRRLP_DATA_REQ	cr	CR_MS_RES
	sdu	NOT_USED
History: 23.09.02	MNC	Initial

3.2.4 RRLP004: RRLP: Measure position request Handover Erroneous

Description: If a handover BTS interrupts the measurements procedure then the LC controller receives an indication about this. The LC controller then informs the RRLP entity about this. The RRLP does not create a RRLP error message for this scenario as it is assumed that the RRLP request is received again after the handover procedure is finished.

Preamble: RRLP000



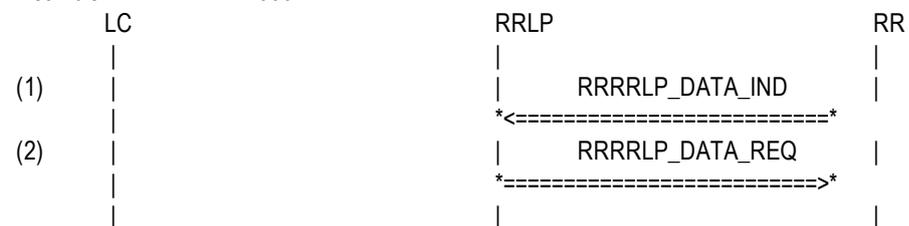
Parametrization

Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr	CR_BS_CMD
	sdu	RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data	NOT_USED
	loc_method	NOT_USED
	pos_method	NOT_USED
	arfcn	NOT_USED
	bsic	NOT_USED
(3) RRLP_ERROR_REQ	cause	LC_HANDOVER_INFO
History: 23.09.02	MNC	Initial

3.2.5 RRLP005: RRLP: Measure position request Erroneous

Description: The network starts a measure position request. The MS detect one erroneous component inside the request with a protocol error message.

Preamble: RRLP000

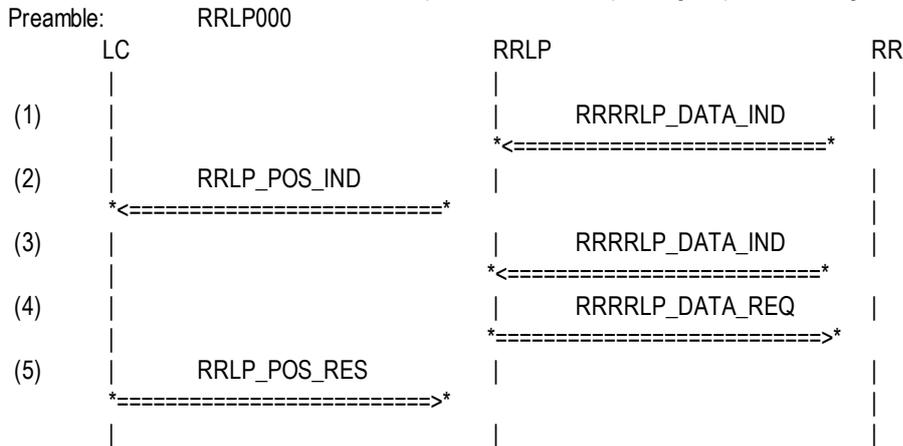


Parametrization

	Primitive	Parameter	Value
(1)	RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_POS_REQ_ERR_P
(2)	RRRRLP_DATA_REQ	cr sdu	CR_MS_RES NOT_USED
History: 23.09.02		MNC	Initial

3.2.6 RRLP006: RRLP: Measure position request repeated component Erroneous – New reference number

Description: The network starts a Measure Position Request. Before the MS finishes the request the network sends a new request. The second request received has a newer reference number than the first but at least one component is erroneous and the MS terminates the request with a Protocol Error message. The first request fails and the response is not send to the network. This is only one example; the second request can arise any time between the first request and the corresponding response message.

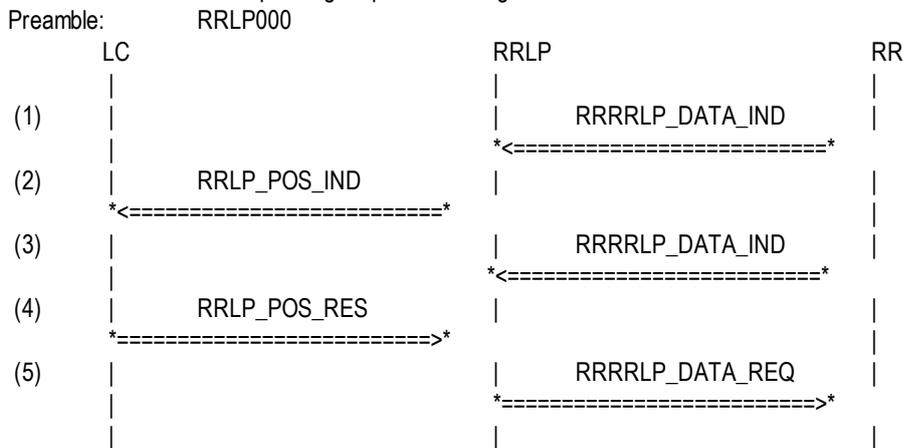


Parametrization

Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr	CR_BS_CMD
	sdu	RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data	NOT_USED
	loc_method	NOT_USED
	pos_method	NOT_USED
	arfcn	NOT_USED
	bsic	NOT_USED
(3) RRRRLP_DATA_IND	cr	CR_BS_CMD
	sdu	RRLP_POS_REQ_NEW_REF_ERR_P
(4) RRRRLP_DATA_REQ	cr	CR_MS_RES
	sdu	NOT_USED
(5) RRLP_POS_RES	timing_data	TIMING_DATA
History: 23.09.02	MNC	Initial

3.2.7 RRLP007: RRLP: Measure Position Request – Repeated Component – Old Reference number

Description: The network starts a Measure Position Request. Before the MS finishes the request the network sends a new request. The MS discards the second request because the reference number is older than the first request. This is only one example; the second request can arise any time between the first request and the corresponding response message.



Parametrization

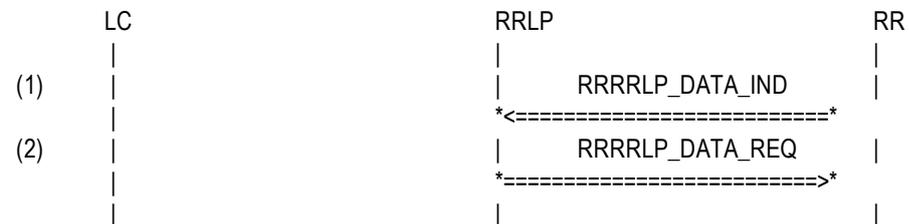
Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data loc_method pos_method arfcn bsic	NOT_USED NOT_USED NOT_USED NOT_USED NOT_USED
(3) RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_POS_REQ_OLD_REF
(4) RRLP_POS_RES	timing_data	TIMING_DATA
(5) RRRRLP_DATA_REQ	cr sdu	CR_MS_RES NOT_USED

History: 23.09.02 MNC Initial

3.2.8 RRLP008: RRLP: Assistance Data - Non-consecutive

Description: The network sends Assistance Data to the MS. The MS processes them and sends an Assistance Data Acknowledge to the network.

Preamble: RRLP000



Parametrization

	Primitive	Parameter	Value
(1)	RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_ASSIST_DATA
(2)	RRRRLP_DATA_REQ	cr sdu	CR_MS_RES RRLP_ASSIST_ACK

History: 23.09.02 MNC Initial

3.2.9 RRLP009: RRLP: Assistance Data - Consecutive

Description: If the entire Assistance Data set is distributed in consecutive sets the network sends consecutive Assistance Data messages until variable MoreAssDataToBeSent is equal zero.

Preamble: RRLP000

	LC	RRLP	RR
(1)		RRRRLP_DATA_IND	
		*<=====	
(2)		RRRRLP_DATA_REQ	
		*=====>	
(3)		RRRRLP_DATA_IND	
		*<=====	
(4)		RRRRLP_DATA_REQ	
		*=====>	
(5)		RRRRLP_DATA_IND	
		*<=====	
(6)		RRRRLP_DATA_REQ	
		*=====>	

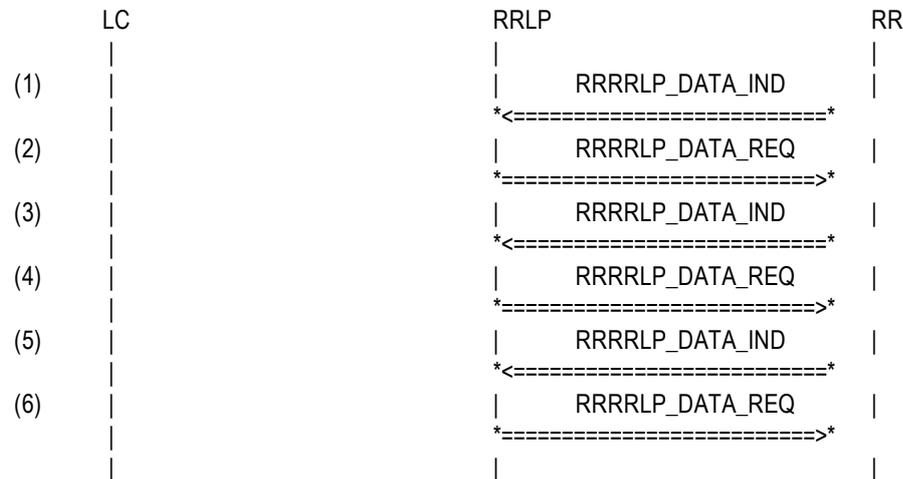
Parametrization

	Primitive	Parameter	Value
(1)	RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_ASSIST_DATA_M
(2)	RRRRLP_DATA_REQ	cr sdu	CR_MS_RES RRLP_ASSIST_ACK
(3)	RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_ASSIST_DATA_M
(4)	RRRRLP_DATA_REQ	cr sdu	CR_MS_RES RRLP_ASSIST_ACK
(5)	RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_ASSIST_DATA_F
(6)	RRRRLP_DATA_REQ	cr sdu	CR_MS_RES RRLP_ASSIST_ACK
History:	23.09.02	MNC	Initial

3.2.10 RRLP010: RRLP: Assistance Data Consecutive – Erroneous

Description: If the entire Assistance Data set is distributed in consecutive sets the network sends consecutive Assistance Data messages until variable MoreAssDataToBeSent is equal zero.

Preamble: RRLP000

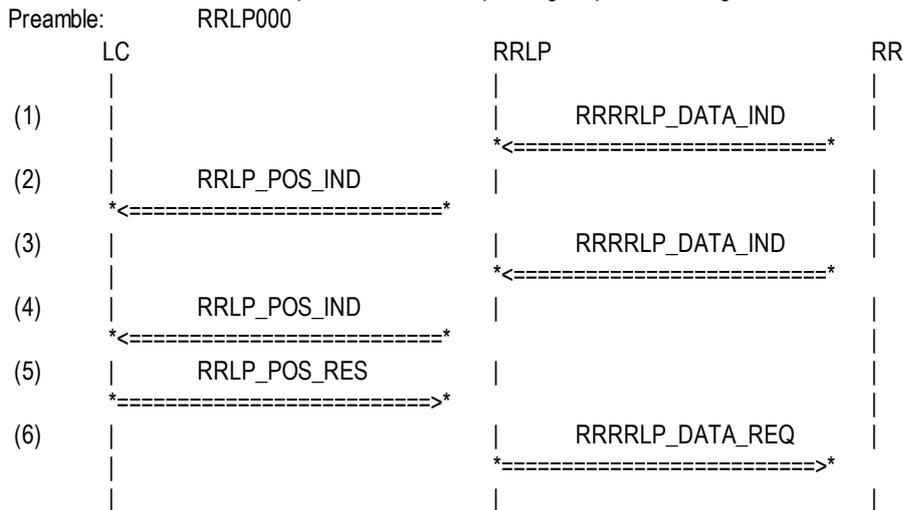


Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1)	RRRRLP_DATA_IND	cr	CR_BS_CMD
		sdu	RRLP_ASSIST_DATA_M
(2)	RRRRLP_DATA_REQ	cr	CR_MS_RES
		sdu	RRLP_ASSIST_ACK
(3)	RRRRLP_DATA_IND	cr	CR_BS_CMD
		sdu	RRLP_ASSIST_DATA_M
(4)	RRRRLP_DATA_REQ	cr	CR_MS_RES
		sdu	RRLP_ASSIST_ACK
(5)	RRRRLP_DATA_IND	cr	CR_BS_CMD
		sdu	RRLP_ASSIST_DATA_ERR_P
(6)	RRRRLP_DATA_REQ	cr	CR_MS_RES
		sdu	NOT_USED
History:	23.09.02	MNC	Initial

3.2.11 RRLP011: RRLP: Measure Position Request – Repeated Component – New Reference number number

Description: The network starts a Measure Position Request. Before the MS finishes the request the network sends a new request. The MS initiates a measurement procedure if the second request has a newer reference number than the first and responds to that. This is only one example; the second request can arise any time between the first request and the corresponding response message.

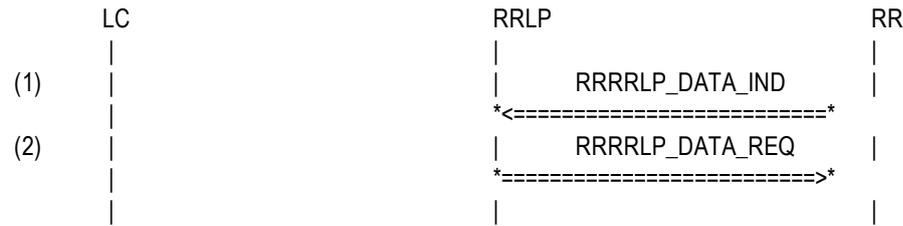


Parametrization

Primitive	Parameter	Value
(1) RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_POS_REQ
(2) RRLP_POS_IND	assist_data loc_method pos_method arfcn bsic	NOT_USED NOT_USED NOT_USED NOT_USED NOT_USED
(3) RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_POS_REQ_NEW_REF
(4) RRLP_POS_IND	assist_data loc_method pos_method arfcn bsic	NOT_USED NOT_USED NOT_USED NOT_USED NOT_USED
(5) RRLP_POS_RES	timing_data	TIMING_DATA
(6) RRRRLP_DATA_REQ	cr sdu	CR_MS_RES NOT_USED
History:	23.09.02	MNC Initial

3.2.12 RRLP012: RRLP: Assistance Data – Erroneous

Description: In case of erroneous assistance data the MS sends a Protocol Error message to the network.
 Preamble: RRLP000



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) RRRRLP_DATA_IND	cr sdu	CR_BS_CMD RRLP_ASSIST_DATA_ERR_P
(2) RRRRLP_DATA_REQ	cr sdu	CR_MS_RES NOT_USED
History:	23.09.02	MNC Initial