



Technical Document

G23-GSM PROTOCOL STACK

AIR

**MULTI COMPONENT TEST SPECIFICATION
(RR-DL-ALR)**

Document Number:	8443.400.01.002
Version:	0.3
Status:	Draft
Approval Authority:	
Creation Date:	2001-Aug-01
Last changed:	2015-Mar-08 by XGUTTEFE
File Name:	air.doc

Important Notice

Texas Instruments Incorporated and/or its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products, software and services at any time and to discontinue any product, software or service without notice. Customers should obtain the latest relevant information during product design and before placing orders and should verify that such information is current and complete.

All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment. TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI products, software and/or services. To minimize the risks associated with customer products and applications, customers should provide adequate design, testing and operating safeguards.

Any access to and/or use of TI software described in this document is subject to Customers entering into formal license agreements and payment of associated license fees. TI software may solely be used and/or copied subject to and strictly in accordance with all the terms of such license agreements.

Customer acknowledges and agrees that TI products and/or software may be based on or implement industry recognized standards and that certain third parties may claim intellectual property rights therein. The supply of products and/or the licensing of software does not convey a license from TI to any third party intellectual property rights and TI expressly disclaims liability for infringement of third party intellectual property rights.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products, software or services are used.

Information published by TI regarding third-party products, software or services does not constitute a license from TI to use such products, software or services or a warranty, endorsement thereof or statement regarding their availability. Use of such information, products, software or services may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

No part of this document may be reproduced or transmitted in any form or by any means, electronically or mechanically, including photocopying and recording, for any purpose without the express written permission of TI.

Change History

Date	Changed by	Approved by	Version	Status	Notes
2001-Aug-01	MSB		0.1	Planned	1
2001-Oct-17	MSB		0.2	Being Processed	
2003-May-12	XGUTTEFE		0.3	Draft	

Notes:

1. Initial version

Table of Contents

1.1	References	5
1.2	Terms	5
1.3	Abbreviations	5
2	Introduction	5
3	Parameters	7
	/* declarations A-L */	7
	/* declarations M-Z */	7
	/* constants A-Z */	8
	/* arrays A-L */	10
	/* arrays M-Z */	13
	/* structures A-L */	14
	/* structures M-Z */	16
4	Test Cases	19
4.1	Configuration, Suite Members	19
4.1.1	AIR000: Configuration (Filter, Reset, Redirect)	19
4.1.2	AIR001: Power Measurement campaign	20
4.1.3	AIR005: BCCH Valid Detection	22
4.1.4	AIR006: BCCH Invalid Detection	23
4.1.5	AIR010: Timeout	24
4.1.6	AIR011: Timeout Wait	24
4.1.7	AIR012: Mute	24
4.1.8	AIR013: DummyWait	25
4.1.9	AIR020: Serving Cell Selection	25
4.1.10	AIR030: Read Serving Cell BCCH	26
4.1.11	AIR035: Serving Cell System Informations	27
4.1.12	AIR036: Invalid Serving Cell System Informations	28
4.1.13	AIR039: Stop Serving Cell Reading BCCH	29
4.1.14	AIR040: Read Neighbour Cell BCCH	29
4.1.15	AIR045: Neighbour Cell System Information	30
4.1.16	AIR048: Stop Neighbour Cell Synchronization	31
4.1.17	AIR049: Stop Neighbour Cell Reading BCCH	31
4.1.18	AIR060: CCCH Configuration	32
4.1.19	AIR061: Stop CCCH Reading	33
4.2	Cell Selection and Reselection	33
4.2.1	AIR100: Start cell selection	33
4.2.2	AIR101: Start cell selection with or without BCCH information (std 6)	35
4.2.3	AIR105: Cell Selection Confirmation	35
4.2.4	AIR110: Synchronization of RR and MM	36
4.3	Idle Mode	36
4.3.1	AIR200: Periodically Measurement Report Request	36
4.3.2	AIR201: Stop Periodically Measurements	37
4.3.3	AIR205: Periodically Measurement Report Indication	37
4.3.4	AIR210: Read Neighbour Cell in Idle mode	38
4.3.5	AIR215: Answer to the Neighbour Cell Synchronization Request	39
5	Suites	39
	/* Internal Suites, Named Test Cases */	39
	/* Serving Cell BCCH Read & Selection */	40
	/* NC process */	41

/* SUI_CellSelectionLim */	42
/* SUI_CellSelectionFull */	43
/* SUI_Idle */	43
/* SUI_CellReselection */	44
Appendices	46
A. Acronyms	46
B. Glossary	46

List of Figures and Tables

List of References

- [ISO 9000:2000] International Organization for Standardization. Quality management systems - Fundamentals and vocabulary. December 2000

1.1 References

- [C_7010.801] 7010.801, References and Vocabulary, Condat AG
[8415.028.99.201] TCC - Test Case Control, User Guide, Condat AG

1.2 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).

1.3 Abbreviations

A list of GSM related abbreviations can be obtained at
http://www.3gpp.org/ftp/Specs/2001-03/R1999/01_series/0104-800.zip

2 Introduction

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.



Figure 1: Mobile station protocol architecture

The base of the Protocol Stack rests on the physical layer.

The Data Link Layer (DL) is used to handle an acknowledged connection between mobile and base station. The LAPDm protocol is used.

Radio Resource (RR) manages the resources of the air-interface. That means configuration of physical layer, cell selection and cell reselection, data transfer, RR-Connection handling.

Mobility Management (MM) handles registration aspects for the mobile station. It detects changes of location areas and updates a mobile station in the new location area.

Call Control (CC) provides the call functionality. This includes call establishment, call maintenance procedures like Hold, Retrieve or Modify, and call disconnection.

Supplementary Services (SS) handles all call independent supplementary services like call forwarding or call barring.

Short Message Services (SMS) is used for sending and receiving point-to-point short messages. Additionally the reception of cell broadcast short messages is included.

The man machine interface (MMI) is the interface to the user. Normally it is connected with a keypad as input device and a display as output device.

Between the several entities data interfaces are defined. These data interfaces are called Service Access Points (SAPs), indicating that an upper layer uses the services of a lower layer.

The GSM specification do not set out any implementation of the Protocol Stack. The following diagrams show the implementation described in all these documents for the mobile station. All entities except the Man Machine Interface and Physical Layer are implemented as part of the Protocol Stack.

This document describes the test cases for the three low layer entities RR, DL and PL as a block.

3 Parameters

/* declarations A-L */

DECLARATION (A_BCCH_CONTENT_24)
DECLARATION (A_BCCH_CONTENT_EMPTY)
DECLARATION (A_BCCH_CONTENT_NO_24)
DECLARATION (CHLIST_ANY)
DECLARATION (CHLIST_23)
DECLARATION (CHLIST_23_CONTENT)
DECLARATION (CHLIST_23_14_124_1)
DECLARATION (CHLIST_23_14_124_1_CONTENT)
DECLARATION (EMPTY_FRAME)
DECLARATION (EMPTY_SCELL_NBCCH)
DECLARATION (FULL_READ)
DECLARATION (FULL_READ_ARRAY)
DECLARATION (IMSI_0010147114912)
DECLARATION (IMSI_1233247114912)
DECLARATION (KC_12345678)
DECLARATION (KCV_12345678)
DECLARATION (KCV_EMPTY)
DECLARATION (L2_NO_CONTENT)
DECLARATION (L2_SYS_INFO_1)
DECLARATION (L2_SYS_INFO_1_ARRAY)
DECLARATION (L2_SYS_INFO_1_NEW)
DECLARATION (L2_SYS_INFO_1_NEW_ARRAY)
DECLARATION (L2_SYS_INFO_2)
DECLARATION (L2_SYS_INFO_2_ARRAY)
DECLARATION (L2_SYS_INFO_3_LAI1)
DECLARATION (L2_SYS_INFO_3_LAI1_ARRAY)
DECLARATION (L2_SYS_INFO_3_LAI2)
DECLARATION (L2_SYS_INFO_3_LAI2_ARRAY)
DECLARATION (L2_SYS_INFO_4_ACS)
DECLARATION (L2_SYS_INFO_4_ACS_ARRAY)
DECLARATION (L2_SYS_INFO_4_LAI1)
DECLARATION (L2_SYS_INFO_4_LAI1_ARRAY)
DECLARATION (L2_SYS_INFO_4_LAI2)
DECLARATION (L2_SYS_INFO_4_LAI2_ARRAY)
DECLARATION (L2_SYS_INFO_5)
DECLARATION (L2_SYS_INFO_5_ARRAY)
DECLARATION (L2_SYS_INFO_5BIS)
DECLARATION (L2_SYS_INFO_5BIS_ARRAY)
DECLARATION (L2_SYS_INFO_6)
DECLARATION (L2_SYS_INFO_6_ARRAY)
DECLARATION (L2_SYS_INFO_7)
DECLARATION (L2_SYS_INFO_7_ARRAY)
DECLARATION (L2_SYS_INFO_8)
DECLARATION (L2_SYS_INFO_8_ARRAY)

/* declarations M-Z */

DECLARATION (MCC_123)
DECLARATION (MM_INFO_1)
DECLARATION (MM_INFO_2)
DECLARATION (MNC_32)
DECLARATION (MNC_33)
DECLARATION (MOBILE_ID_IMSI_HPLMN)
DECLARATION (MOBILE_ID_NOT_SET)
DECLARATION (MOBILE_ID_TMSI)

DECLARATION (MOBILE_ID_IMSI_TEST)
DECLARATION (NCELL_RESULTA_1)
DECLARATION (NCELL_RESULTA_14)
DECLARATION (NCELL_RESULTA_124)
DECLARATION (NCELL_RESULTA_23)
DECLARATION (NCELL_RESULTB_23)
DECLARATION (NCELL_RESULT_ARRAY1A)
DECLARATION (NCELL_RESULT_ARRAY1B)
DECLARATION (NCELL_RESULT_ARRAY2A)
DECLARATION (NCELL_RESULT_ARRAY2B)
DECLARATION (NCELL_RESULT_EMPTY)
DECLARATION (OP_MODE_EMPTY)
DECLARATION (OP_MODE_EMPTY_NO_SERV)
DECLARATION (OP_MODE_NORMAL)
DECLARATION (OP_MODE_NET_SRCH_MMI)
DECLARATION (OP_MODE_NET_SRCH_MMI_LIM_SRV)
DECLARATION (OP_MODE_NET_SRCH_MMI_NO_SRV)
DECLARATION (OP_MODE_NO_SIM_LIM_SERV)
DECLARATION (OP_MODE_TEST_SIM)
DECLARATION (OP_MODE_TEST_SIM_NO_SERV)
DECLARATION (OP_MODE_TEST_SIM_LIM_SERV)
DECLARATION (PLMN_ID_EMPTY)
DECLARATION (PLMN_ID_123)
DECLARATION (S_BCCH_INFO_24)
DECLARATION (S_BCCH_INFO_EMPTY)
DECLARATION (S_BCCH_INFO_NO_24)
DECLARATION (STOP_BCCH_ARRAY1)
DECLARATION (STOP_BCCH_ARRAY124)
DECLARATION (STOP_BCCH_ARRAY14)
DECLARATION (STOP_BCCH_ARRAY23)

/* constants A-Z */

SHORTACC_CTRL_CLASS_0000		0x0000
SHORTACC_CTRL_CLASS_0008		0x0008
SHORTACC_CTRL_CLASS_4000		0x4000
SHORT_ARFCN_1		1
SHORT_ARFCN_14	14	
SHORT_ARFCN_23	23	
SHORT_ARFCN_124	124	
BYTE_BA_ID_0	0	
BYTE_BA_ID_1	1	
BYTE_BA_ID_2	2	
BYTE_BA_ID_3	3	
BYTE_BA_ID_4	4	
BYTE_BCC_0	0	
BYTE_BCC_1	1	
BYTE_BS_AG_BLK_RES_2	2	
BYTE_BS_AG_BLK_RES_3	3	
BYTE_BS_AG_BLK_RES_5	5	
BYTE_BS_AG_BLK_RES_7	7	
BYTE_BS_PA_MFRMS_0	0	
BYTE_BS_PA_MFRMS_1	1	
BYTE_BS_PA_MFRMS_2	2	
BYTE_BS_PA_MFRMS_3	3	
BYTE_BS_PA_MFRMS_4	4	
BYTE_BS_PA_MFRMS_5	5	
BYTE_BS_PA_MFRMS_6	6	
BYTE_BS_PA_MFRMS_7	7	

BYTE	BS_PA_MFRMS_8	8	
BYTE	BS_PA_MFRMS_9	9	
BYTE	BSIC_1	1	
BYTE	CCCH_GROUP_0		0
BYTE	CCCH_GROUP_2		2
SHORTCELL_IDENT_0001		0x0001	
BYTE	CHAN_LIST_IDX_0	0	
BYTE	CHANNELS_1	1	
BYTE	CHANNELS_2	2	
BYTE	CHANNELS_4	4	
BYTE	CHANNELS_8	8	
BYTE	CKSN_6		6
LONG	FN_OFFSET_0	0	
LONG	FN_OFFSET_23		23
LONG	FN_OFFSET_101		101
LONG	FN_OFFSET_110		110
LONG	FN_OFFSET_114		114
LONG	FN_OFFSET_224		224
BYTE	L2_CHANNEL_SACCH	1	
BYTE	L2_CHANNEL_SDCCH	2	
BYTE	L2_CHANNEL_FACCH_H		3
BYTE	L2_CHANNEL_FACCH_F		4
BYTE	L2_CHANNEL_CCCH	5	
BYTE	L2_CHANNEL_NBCCH	6	
BYTE	L2_CHANNEL_PCH	7	
BYTE	L2_CHANNEL_EPCH	8	
BYTE	L2_CHANNEL_CBCH	9	
BYTE	L2_CHANNEL_EBCCH	10	
SHORTLAC_0001		0x0001	
BYTE	NCC_0	0	
SHORTNCELL_BCCH_SI_3_4		0x00CC	
BYTE	PAGE_GROUP_0		0
BYTE	PAGE_BLOCK_IDX_0	0	
BYTE	RXLEV_10	10	
BYTE	RXLEV_28	28	
BYTE	RXLEV_56	56	
BYTE	RXLEV_IDX_0	0	
BYTE	RXLEV_IDX_1	1	
BYTE	RXLEV_IDX_2	2	
BYTE	RXLEV_IDX_3	3	
BYTE	RXLEV_IDX_4	4	
BYTE	RXLEV_IDX_5	5	
BYTE	RXLEV_IDX_6	6	
BYTE	SB_FOUND	1	
BYTE	SB_NOT_FOUND	0	
BYTE	SCHEM_SIZE_1		1
BYTE	SM_WIDE_MODE		0
BYTE	SM_NARROW_MODE	1	
BYTE	T3212_36_MIN	0x06	
BYTE	TC_0	0	
BYTE	TC_1	1	
BYTE	TC_2	2	
BYTE	TC_3	3	
BYTE	TC_4	4	
BYTE	TC_5	5	
BYTE	TC_6	6	
BYTE	TC_7	7	
LONG	TIME_ALIGNMT_0		0
LONG	TIME_ALIGNMT_1		1

```

LONG TIME_ALIGNMT_14      14
LONG TIME_ALIGNMT_23      23
LONG TIME_ALIGNMT_110     110
LONG TIME_ALIGNMT_124     124
BYTE TIME_HPLMN_EMPTY     0x00
BYTE TIME_HPLMN_VALID     0x05
SHORTTMSI_142             0x0142
BYTE TV_INVALID_TIMING_INFO 0

```

/* arrays A-L */

```

BEGINARRAY (A_BCCH_CONTENT_24, 16)
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x80, 0x00, 0x00
ENDARRAY
BEGINARRAY (A_BCCH_CONTENT_EMPTY, 16)
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
ENDARRAY
BEGINARRAY (A_BCCH_CONTENT_NO_24, 16)
    0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
    0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x7F, 0xFF, 0xFF
ENDARRAY
BEGIN_SHORT_ARRAY (CHLIST_23_14_124_1_CONTENT, 4)
    ARFCN_23,
    ARFCN_14,
    ARFCN_124,
    ARFCN_1
ENDARRAY
BEGIN_SHORT_ARRAY (CHLIST_23_CONTENT, 1)
    ARFCN_23
ENDARRAY
BEGINARRAY (EMPTY_FRAME, 1)
    0x00
ENDARRAY
BEGIN_STRUCT_ARRAY(FULL_READ_ARRAY, MAX_SCHD_SIZE)
    FULL_READ,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH,
    EMPTY_SCELL_NBCCH
ENDARRAY
BEGINARRAY (IMSI_1233247114912, 14)
    0x01, 0x02, 0x03, 0x03,
    0x02, 0x04, 0x07, 0x01,
    0x01, 0x04, 0x09, 0x01,
    0x02,
    0x0F
ENDARRAY

```

```
BEGINARRAY (IMSI_0010147114912, 14)
    0x00, 0x00, 0x01, 0x00,
    0x01, 0x04, 0x07, 0x01,
    0x01, 0x04, 0x09, 0x01,
    0x02,
    0x0F
ENDARRAY
BEGINARRAY (KC_12345678, 8)
    0x01, 0x02, 0x03, 0x04,
    0x05, 0x06, 0x07, 0x08
ENDARRAY
BEGINARRAY (L2_SYS_INFO_1_ARRAY, 23)
    0x59, /* l2 pseudo length 22 bytes */
    0x06, /* protocol discriminator, transaction identifier */
    0x19, /* message type */
    0x00, 0x04, 0x00, 0x00, 0x00, 0x02, 0x00, 0x01, /* channel description */
    0x00, 0x08, 0x00, 0x00, 0x81, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x40, /* RACH control parameter */
    0x2B /* rest oktet */
ENDARRAY
BEGINARRAY (L2_SYS_INFO_1_NEW_ARRAY, 23)
    0x59, /* l2 pseudo length 22 bytes */
    0x06, /* protocol discriminator, transaction identifier */
    0x19, /* message type */
    0x00, 0x04, 0x00, 0x00, 0x00, 0x02, 0x00, 0x01, /* channel description */
    0x00, 0x18, 0x00, 0x00, 0x81, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x40, /* RACH control parameter */
    0x2B
ENDARRAY
BEGINARRAY (L2_SYS_INFO_2_ARRAY, 23)
    0x59, /* l2 pseudo length 22 bytes */
    0x06, /* protocol discriminator, transaction identifier */
    0x1A, /* message type */
    /* neighbour cell description with bit map 0 for mat (for GSM900) */
    0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, /* arfcn: 124 */
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x01, /* arfcn: 14, 1 */
    0x01, /* ncc permitted */
    0x00, 0x00, 0x40 /* rach control parameter */
ENDARRAY
BEGINARRAY (L2_SYS_INFO_3_LAI1_ARRAY, 23)
    0x59, /* l2 pseudo length (=22 Byte)*/
    0x06, /* protocol discriminator, transaction identifier */
    0x1B, /* message type */
    0x37, 0x48, /* cell identity */
    0x21, 0xF3, 0x33, 0x21, 0x47, /* location area identification MCC=123 MNC=33F */
    0x28, 0x02, 0x06, /* control channel description */
    0x5F, /* cell options BCCH */
    0x42, 0x56, /* cell selection parameter */
    0x00, 0x00, 0x40, /* rach control parameter */
    0x2B, 0x2B, 0x2B, 0x2B /* rest oktet */
ENDARRAY
```

```
BEGINARRAY (L2_SYS_INFO_3_LAI2_ARRAY, 23)
    0x59,          /* l2 pseudo length (=22 Byte)*/
    0x06,          /* protocol discriminator, transaction identifier */
    0x1B,          /* message type */
    0x37, 0x48,    /* cell identity */
    0x21, 0xF3, 0x23, 0x21, 0x47, /* location area identification MCC=123 MNC=32F */
    0x28, 0x02, 0x06, /* control channel description */
    0x5F,          /* cell options BCCH */
    0x42, 0x56,    /* cell selection parameter */
    0x00, 0x00, 0x40, /* rach control parameter */
    0x2B, 0x2B, 0x2B, 0x2B /* rest oktest */
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_4_LAI1_ARRAY, 23)
    0x59,          /* l2 pseudo length (=22 Byte)*/
    0x06,          /* protocol discriminator, transaction identifier */
    0x1C,          /* message type */
    0x21, 0xF3, 0x33, 0x21, 0x47, /* location area identification MCC=123 MNC=33F */
    0x42, 0x56,    /* cell selection parameter */
    0x00, 0x00, 0x40, /* rach control parameter */
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B, /* rest octets */
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_4_LAI2_ARRAY, 23)
    0x59,          /* l2 pseudo length (=22 Byte)*/
    0x06,          /* protocol discriminator, transaction identifier */
    0x1C,          /* message type */
    0x21, 0xF3, 0x23, 0x21, 0x47, /* location area identification MCC=123 MNC=32F */
    0x42, 0x56,    /* cell selection parameter */
    0x00, 0x00, 0x40, /* rach control parameter */
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B, /* rest octets */
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_4_ACS_ARRAY, 23)
    0x59, /* l2 pseudo length (=22 Byte)*/
    0x06, /* protocol discriminator, transaction identifier */
    0x1C, /* message type */
    0x21, 0xF3, 0x33, 0x21, 0x47, /* location area identification */
    0x42, 0xD6, /* cell selection parameter */
    0x00, 0x00, 0x40, /* rach control parameter */
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B, /* rest octets */
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_5_ARRAY, 23)
    0xFF, 0xAA,
    0x03, 0x03, 0x01,
    0x06,
    0x1D,
    0x00, 0x04, 0x00, 0x00, 0x00, 0x02, 0x00, 0x01,
    0x00, 0x08, 0x00, 0x00, 0x81, 0x00, 0x00, 0x00
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_5BIS_ARRAY,23)
    0xFF, 0xAA,
    0x03, 0x03, 0x01,
    0x06,
    0x05,
    0x00, 0x04, 0x00, 0x00, 0x00, 0x02, 0x00, 0x01,
    0x00, 0x08, 0x00, 0x00, 0x81, 0x00, 0x00, 0x00
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_6_ARRAY,23)
    0xFF, 0xAA,
    0x03, 0x03, 0x01,
    0x06,
    0x1E,
    0x00, 0x04, 0x00, 0x00, 0x00, 0x02, 0x00, 0x01,
    0x00, 0x08, 0x00, 0x00, 0x81, 0x00, 0x00, 0x00
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_7_ARRAY, 23)
```

```
    0x59,
    0x06,
    0x1F,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b,
    0x2b, 0x2b, 0x2b,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

```
BEGINARRAY (L2_SYS_INFO_8_ARRAY, 23)
```

```
    0x59,
    0x06,
    0x18,
    0x2b, 0x2b, 0x2b, 0x2b, 0x2b,
    0x2b, 0x2b,
    0x2b, 0x2b, 0x2b,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B,
    0x2B, 0x2B, 0x2B, 0x2B, 0x2B
```

ENDARRAY

/* arrays M-Z */

```
BEGINARRAY (MCC_123, 3)           /* mobile color code */
    0x01, 0x02, 0x03
```

ENDARRAY

```
BEGINARRAY (MNC_32, 2)           /* mobile network code */
    0x03, 0x02
```

ENDARRAY

```
BEGINARRAY (MNC_33, 2)           /* mobile network code */
    0x03, 0x03
```

ENDARRAY

```
BEGIN_STRUCT_ARRAY (NCELL_RESULT_ARRAY1A, 8)
```

```
    NCELL_RESULTA_23,
    NCELL_RESULTA_23,
    NCELL_RESULTA_23,
    NCELL_RESULTA_23,
    NCELL_RESULTA_23,
    NCELL_RESULTA_23,
    NCELL_RESULTA_23,
    NCELL_RESULTA_23
```

ENDARRAY

```
BEGIN_STRUCT_ARRAY (NCELL_RESULT_ARRAY1B, 8)
```

```
    NCELL_RESULTB_23,
    NCELL_RESULTB_23,
    NCELL_RESULTB_23,
    NCELL_RESULTB_23,
    NCELL_RESULTB_23,
    NCELL_RESULTB_23,
    NCELL_RESULTB_23,
    NCELL_RESULTB_23
```

ENDARRAY

```
BEGIN_STRUCT_ARRAY (NCELL_RESULT_ARRAY2A, 8)
    NCELL_RESULTA_23,
    NCELL_RESULTA_14,
    NCELL_RESULTA_124,
    NCELL_RESULTA_1,
    NCELL_RESULTA_23,
    NCELL_RESULTA_14,
    NCELL_RESULTA_124,
    NCELL_RESULTA_1
ENDARRAY
BEGIN_STRUCT_ARRAY (NCELL_RESULT_ARRAY2B, 8)
    NCELL_RESULTB_23,
    NCELL_RESULTA_14,
    NCELL_RESULTA_124,
    NCELL_RESULTA_1,
    NCELL_RESULTB_23,
    NCELL_RESULTA_14,
    NCELL_RESULTA_124,
    NCELL_RESULTA_1
ENDARRAY
BEGIN_SHORT_ARRAY(STOP_BCCH_ARRAY23, 6)
    ARFCN_23,
    0,0,0,0,0
ENDARRAY
BEGIN_SHORT_ARRAY(STOP_BCCH_ARRAY14, 6)
    ARFCN_14,
    0,0,0,0,0
ENDARRAY
BEGIN_SHORT_ARRAY(STOP_BCCH_ARRAY124, 6)
    ARFCN_124,
    0,0,0,0,0
ENDARRAY
BEGIN_SHORT_ARRAY(STOP_BCCH_ARRAY1, 6)
    ARFCN_1,
    0,0,0,0,0
ENDARRAY
```

/* structures A-L */

```
BEGIN_PSTRUCT ("chan_list", CHLIST_ANY)
    SHOW_COMP("radio_freq")
ENDSTRUCT
BEGIN_PSTRUCT ("chan_list", CHLIST_23)
    SET_COMP("radio_freq", CHLIST_23_CONTENT)
ENDSTRUCT
BEGIN_PSTRUCT ("chan_list", CHLIST_23_14_124_1)
    SET_COMP("radio_freq", CHLIST_23_14_124_1_CONTENT)
ENDSTRUCT
BEGIN_PSTRUCT("schedule_array", EMPTY_SCELL_NBCCH)
    SET_COMP("modulus", 0)
    SET_COMP("relative_position", 0)
ENDSTRUCT
BEGIN_PSTRUCT("schedule_array", FULL_READ)
    SET_COMP("modulus", 1)
    SET_COMP("relative_position", 0)
ENDSTRUCT
```

```
BEGIN_PSTRUCT ("kcv", KCV_12345678)
    SET_COMP ("v_kc",      V_KC_PRES)
    SET_COMP ("kc",      KC_12345678)
ENDSTRUCT
BEGIN_PSTRUCT ("kcv", KCV_EMPTY)
    SET_COMP ("v_kc",      V_KC_NOT_PRES)
    SKIP_COMP ("kc")
ENDSTRUCT
BEGIN_PSTRUCT ("imsi", MOBILE_ID_IMSI_HPLMN)
    SET_COMP ("v_mid",      V_MID_PRES)
    SET_COMP ("id_type",    TYPE_IMSI)
    SET_COMP ("id",      IMSI_1233247114912)
    SKIP_COMP ("tmsi_dig")
ENDSTRUCT
BEGIN_PSTRUCT ("imsi", MOBILE_ID_IMSI_TEST)
    SET_COMP ("v_mid",      V_MID_PRES)
    SET_COMP ("id_type",    TYPE_IMSI)
    SET_COMP ("id",      IMSI_0010147114912)
    SKIP_COMP ("tmsi_dig")
ENDSTRUCT
BEGIN_PSTRUCT ("imsi", MOBILE_ID_NOT_SET)
    SET_COMP ("v_mid",      V_MID_PRES)
    SET_COMP ("id_type",    NOT_PRESENT_8BIT)
    SKIP_COMP ("id")
    SKIP_COMP ("tmsi_dig")
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_NO_CONTENT)
    SET_COMP ("content", EMPTY_FRAME)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_1)
    SET_COMP ("content", L2_SYS_INFO_1_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_1_NEW)
    SET_COMP ("content", L2_SYS_INFO_1_NEW_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_2)
    SET_COMP ("content", L2_SYS_INFO_2_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_3_LAI1)
    SET_COMP ("content", L2_SYS_INFO_3_LAI1_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_3_LAI2)
    SET_COMP ("content", L2_SYS_INFO_3_LAI2_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_4_LAI1)
    SET_COMP ("content", L2_SYS_INFO_4_LAI1_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_4_LAI2)
    SET_COMP ("content", L2_SYS_INFO_4_LAI2_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_4_ACS)
    SET_COMP ("content", L2_SYS_INFO_4_ACS_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_5)
    SET_COMP ("content", L2_SYS_INFO_5_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT ("l2_frame", L2_SYS_INFO_5BIS)
    SET_COMP ("content", L2_SYS_INFO_5BIS_ARRAY)
ENDSTRUCT
```

```
BEGIN_PSTRUCT("l2_frame", L2_SYS_INFO_6)
    SET_COMP("content", L2_SYS_INFO_6_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT("l2_frame", L2_SYS_INFO_7)
    SET_COMP("content", L2_SYS_INFO_7_ARRAY)
ENDSTRUCT
BEGIN_PSTRUCT("l2_frame", L2_SYS_INFO_8)
    SET_COMP("content", L2_SYS_INFO_8_ARRAY)
ENDSTRUCT
```

/* structures M-Z */

```
BEGIN_PSTRUCT ("mm_info", MM_INFO_1)
    SET_COMP ("valid",      MM_INFO_PRES)
    SET_COMP ("la",        LA_NOT_IN_FRBD_LST_INCL)
    SET_COMP ("att",       ATT_NOT_ALLOW)
    SET_COMP ("re",        REESTAB_YES)
    SET_COMP ("band",     BND_DMY_VAL)
    SKIP_COMP ("ncc")
    SKIP_COMP ("bcc")
    SET_COMP ("t3212",    T3212_36_MIN)
ENDSTRUCT
BEGIN_PSTRUCT ("mm_info", MM_INFO_2)
    SET_COMP ("valid",      MM_INFO_PRES)
    SET_COMP ("la",        LA_NOT_IN_FRBD_LST_INCL)
    SET_COMP ("att",       ATT_ALLOW)
    SET_COMP ("re",        REESTAB_YES)
    SET_COMP ("band",     BND_DMY_VAL)
    SKIP_COMP ("ncc")
    SKIP_COMP ("bcc")
    SET_COMP ("t3212",    T3212_36_MIN)
ENDSTRUCT
BEGIN_PSTRUCT ("tmsi", MOBILE_ID_TMSI)
    SET_COMP ("v_mid",     V_MID_PRES)
    SET_COMP ("id_type",  TYPE_TMSI)
    SKIP_COMP ("id")
    SET_COMP ("tmsi_dig", TMSI_142)
ENDSTRUCT
BEGIN_PSTRUCT("result", NCELL_RESULTA_1)
    SET_COMP("radio_freq", ARFCN_1)
    SET_COMP("rxlev", 12)
ENDSTRUCT
BEGIN_PSTRUCT("result", NCELL_RESULTA_14)
    SET_COMP("radio_freq", ARFCN_14)
    SET_COMP("rxlev", 44)
ENDSTRUCT
BEGIN_PSTRUCT("result", NCELL_RESULTA_124)
    SET_COMP("radio_freq", ARFCN_124)
    SET_COMP("rxlev", 25)
ENDSTRUCT
BEGIN_PSTRUCT("result", NCELL_RESULTA_23)
    SET_COMP("radio_freq", ARFCN_23)
    SET_COMP("rxlev", 56)
ENDSTRUCT
BEGIN_PSTRUCT("result", NCELL_RESULTB_23)
    SET_COMP("radio_freq", ARFCN_23)
    SET_COMP("rxlev", 10)
ENDSTRUCT
BEGIN_PSTRUCT("result", NCELL_RESULT_EMPTY)
    SKIP_COMP("radio_freq")
```

```
        SKIP_COMP("rxlev")
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_EMPTY)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_NO_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_NO_INSRT)
    SET_COMP ("func",     FUNC_LIM_SERV_ST_SRCH)
    SET_COMP ("service",  LIMITED_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_EMPTY_NO_SERV)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_NO_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_NO_INSRT)
    SET_COMP ("func",     FUNC_LIM_SERV_ST_SRCH)
    SET_COMP ("service",  NO_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_NET_SRCH_MMI)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",     FUNC_NET_SRCH_BY_MMI)
    SET_COMP ("service",  FULL_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_NET_SRCH_MMI_LIM_SRV)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",     FUNC_NET_SRCH_BY_MMI)
    SET_COMP ("service",  LIMITED_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_NET_SRCH_MMI_NO_SRV)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",     FUNC_NET_SRCH_BY_MMI)
    SET_COMP ("service",  NO_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_NO_SIM_LIM_SERV)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_NO_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_NO_INSRT)
    SET_COMP ("func",     FUNC_LIM_SERV_ST_SRCH)
    SET_COMP ("service",  LIMITED_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_NORMAL)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",       TS_NO_AVAIL)
    SET_COMP ("m",        M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",     FUNC_PLMN_SRCH)
    SET_COMP ("service",  FULL_SERVICE)
ENDSTRUCT
```

```
BEGIN_PSTRUCT ("op", OP_MODE_TEST_SIM)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",      TS_AVAIL)
    SET_COMP ("m",      M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",    FUNC_PLMN_SRCH)
    SET_COMP ("service",  FULL_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_TEST_SIM_LIM_SERV)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",      TS_AVAIL)
    SET_COMP ("m",      M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",    FUNC_PLMN_SRCH)
    SET_COMP ("service",  LIMITED_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("op", OP_MODE_TEST_SIM_NO_SERV)
    SET_COMP ("v_op",      V_OP_PRES)
    SET_COMP ("ts",      TS_AVAIL)
    SET_COMP ("m",      M_AUTO)
    SET_COMP ("sim_ins",  SIM_INSRT)
    SET_COMP ("func",    FUNC_PLMN_SRCH)
    SET_COMP ("service",  NO_SERVICE)
ENDSTRUCT
BEGIN_PSTRUCT ("plmn", PLMN_ID_123)
    SET_COMP ("v_plmn",   V_PLMN_PRES)
    SET_COMP ("mcc",     MCC_123)
    SET_COMP ("mnc",     MNC_32)
ENDSTRUCT
BEGIN_PSTRUCT ("plmn", PLMN_ID_EMPTY)
    SET_COMP ("v_plmn",   V_PLMN_NOT_PRES)
    SKIP_COMP ("mcc")
    SKIP_COMP ("mnc")
ENDSTRUCT
BEGIN_PSTRUCT ("bcch_info", S_BCCH_INFO_24)
    SET_COMP ("v_bcch",   V_BCCH_PRES)
    SET_COMP ("bcch",     A_BCCH_CONTENT_24)
ENDSTRUCT
BEGIN_PSTRUCT ("bcch_info", S_BCCH_INFO_EMPTY)
    SET_COMP ("v_bcch",   V_BCCH_NOT_PRES)
    SET_COMP ("bcch",     A_BCCH_CONTENT_EMPTY)
ENDSTRUCT
BEGIN_PSTRUCT ("bcch_info", S_BCCH_INFO_NO_24)
    SET_COMP ("v_bcch",   V_BCCH_PRES)
    SET_COMP ("bcch",     A_BCCH_CONTENT_NO_24)
ENDSTRUCT
```

/*

History:	31-Jul-01	MSB	Initial
	17-Oct-01	MSB	Different color codes for sys info 3 and 4
			Constants changed for TIME_ALIGNMENT

and FN_OFFSET

*/

4 Test Cases

4.1 Configuration, Suite Members

4.1.1 AIR000: Configuration (Filter, Reset, Redirect)

Description: Routing with RR-DL-ALR as a *block* of entities. With the command DUPLICATE it is possible to test the primitive communication inside the block.

Preamble: none

MM	RR/DL/ALR	L1
COMMAND (TAP RESET)		
COMMAND (MMI RESET)		
COMMAND (CC RESET)		
COMMAND (SS RESET)		
COMMAND (SMS RESET)		
COMMAND (MM RESET)		
COMMAND (RR RESET)		
COMMAND (DL RESET)		
COMMAND (SIM RESET)		
COMMAND (PL RESET)		
COMMAND (TAP REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CLEAR)		
COMMAND (CC REDIRECT CLEAR)		
COMMAND (SS REDIRECT CLEAR)		
COMMAND (SMS REDIRECT CLEAR)		
COMMAND (MM REDIRECT CLEAR)		
COMMAND (RR REDIRECT CLEAR)		
COMMAND (DL REDIRECT CLEAR)		
COMMAND (SIM REDIRECT CLEAR)		
COMMAND (PL REDIRECT CLEAR)		
COMMAND (MMI REDIRECT MM NULL)		
COMMAND (MMI REDIRECT CC NULL)		
COMMAND (MMI REDIRECT SS NULL)		
COMMAND (MMI REDIRECT SMS NULL)		
COMMAND (MMI REDIRECT PL NULL)		
COMMAND (CC REDIRECT MMI NULL)		
COMMAND (CC REDIRECT MM NULL)		
COMMAND (SS REDIRECT MMI NULL)		
COMMAND (SS REDIRECT MM NULL)		
COMMAND (SMS REDIRECT MMI NULL)		
COMMAND (SMS REDIRECT MM NULL)		
COMMAND (MM REDIRECT MMI NULL)		
COMMAND (MM REDIRECT CC NULL)		
COMMAND (MM REDIRECT SS NULL)		
COMMAND (MM REDIRECT SMS NULL)		
COMMAND (MM REDIRECT SIM NULL)		
COMMAND (SIM REDIRECT MM NULL)		

```

COMMAND (RR REDIRECT MM TAP)
COMMAND (PL REDIRECT L1 TAP)
|
|
|
COMMAND (TAP REDIRECT TAP **000110***** RR) /* SAP RR 0x0600, 0x4600
*/
COMMAND (TAP REDIRECT TAP **000100***** DL) /* SAP MDL 0x0400, 0x4400
*/
COMMAND (TAP REDIRECT TAP **000000***** PL) /* SAP MPH 0x0000 */
COMMAND (TAP REDIRECT TAP **000001***** PL) /* SAP PH 0x0100, 0x4100
*/
|
|
|
COMMAND (RR TRACECLASS 0F)
COMMAND (PL TRACECLASS 0F)
|
|
|

```

Parametrization

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

```

History:      31-Jul-01      MSB      Initial
/*
COMMAND (RR DUPLICATE PL TAP)
COMMAND (PL DUPLICATE RR TAP)
COMMAND (RR DUPLICATE DL TAP)
COMMAND (DL DUPLICATE RR TAP)
COMMAND (DL DUPLICATE PL TAP)
COMMAND (PL DUPLICATE DL TAP)
COMMAND (TAP REDIRECT TAP **000011***** DL) /* SAP DL 0x0300, 0x4300
*/
COMMAND (RR CONFIG NO_SYS_TIME)
*/

```

4.1.2 AIR001: Power Measurement campaign

Description: Five power measurement requests with the resulting measurement indications. Because the primitive are too big to be forwarded via the test interface, we use in the windows simulation an array (T_POWER_MEAS tap_rxlev_response, alr_pei.c) with predefined values and set shared_ptr to the index of the choosen measurement values.

Preamble: none

Variants: <A>..<G>

MM	RR/DL/ALR	L1
(1)	MPHC_RXLEV_REQ	
(2)	MPHC_RXLEV_IND	
(3)	MPHC_RXLEV_REQ	
(4)	MPHC_RXLEV_IND	
(5)	MPHC_RXLEV_REQ	
(6)	MPHC_RXLEV_IND	
(7)	MPHC_RXLEV_REQ	
(8)	MPHC_RXLEV_IND	
(9)	MPHC_RXLEV_REQ	
(10)	MPHC_RXLEV_IND	

Parametrization

Primitive	Parameter	Value
(1) MPHC_RXLEV_REQ	shared_ptr	NOT_USED
(2) MPHC_RXLEV_IND	shared_ptr	RXLEV_IDX_0
<A>	shared_ptr	RXLEV_IDX_1
	shared_ptr	RXLEV_IDX_2
<C>	shared_ptr	RXLEV_IDX_3
<D>	shared_ptr	RXLEV_IDX_4
<E>	shared_ptr	RXLEV_IDX_5
<F>	shared_ptr	RXLEV_IDX_6
<G>	shared_ptr	
(3) MPHC_RXLEV_REQ	shared_ptr	NOT_USED
(4) MPHC_RXLEV_IND	shared_ptr	RXLEV_IDX_0
<A>	shared_ptr	RXLEV_IDX_1
	shared_ptr	RXLEV_IDX_2
<C>	shared_ptr	RXLEV_IDX_3
<D>	shared_ptr	RXLEV_IDX_4
<E>	shared_ptr	RXLEV_IDX_5
<F>	shared_ptr	RXLEV_IDX_6
<G>	shared_ptr	
(5) MPHC_RXLEV_REQ	shared_ptr	NOT_USED
(6) MPHC_RXLEV_IND	shared_ptr	RXLEV_IDX_0
<A>	shared_ptr	RXLEV_IDX_1
	shared_ptr	RXLEV_IDX_2
<C>	shared_ptr	RXLEV_IDX_3
<D>	shared_ptr	RXLEV_IDX_4
<E>	shared_ptr	RXLEV_IDX_5
<F>	shared_ptr	RXLEV_IDX_6
<G>	shared_ptr	

(7) MPH_C_RXLEV_REQ	shared_ptr	NOT_USED
(8) MPH_C_RXLEV_IND		
<A>	shared_ptr	RXLEV_IDX_0
	shared_ptr	RXLEV_IDX_1
<C>	shared_ptr	RXLEV_IDX_2
<D>	shared_ptr	RXLEV_IDX_3
<E>	shared_ptr	RXLEV_IDX_4
<F>	shared_ptr	RXLEV_IDX_5
<G>	shared_ptr	RXLEV_IDX_6
(9) MPH_C_RXLEV_REQ	shared_ptr	NOT_USED
(10) MPH_C_RXLEV_IND		
<A>	shared_ptr	RXLEV_IDX_0
	shared_ptr	RXLEV_IDX_1
<C>	shared_ptr	RXLEV_IDX_2
<D>	shared_ptr	RXLEV_IDX_3
<E>	shared_ptr	RXLEV_IDX_4
<F>	shared_ptr	RXLEV_IDX_5
<G>	shared_ptr	RXLEV_IDX_6

History: 31-Jul-01 MSB Initial

4.1.3 AIR005: BCCH Valid Detection

Description: After RR requests BCCH detection (block intern) ALR requests synchronisation to frequency corrections and synchron bursts. Layer 1 acknowledges that by indication. The sb_flag indicates whether layer 1 should read the bursts correct or the bursts has failed.

- Variant A: arfcn = 23
- Variant B: arfcn = 14
- Variant C: arfcn = 124
- Variant D: arfcn = 1

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)		MPH_C_NETWORK_SYNC_REQ	
		*=====	
(2)		MPH_C_NETWORK_SYNC_IND	
		*<=====	

Parametrization

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

(1) MPHC_NETWORK_SYNC_REQ		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	fn_offset	NOT_USED
	time_alignment	NOT_USED
	timing_validity	TV_INVALID_TIMING_INFO
	search_mode	SM_WIDE_MODE
(2) MPHC_NETWORK_SYNC_IND		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	sb_flag	SB_FOUND
	fn_offset	FN_OFFSET_0
	time_alignment	TIME_ALIGNMT_0
	bsic	BSIC_1

History: 3-Aug-01 MSB Initial

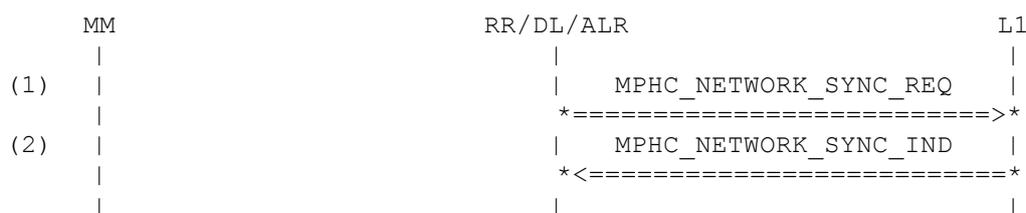
4.1.4 AIR006: BCCH Invalid Detection

Description: After RR requests BCCH detection (block intern) ALR requests synchronisation to frequency corrections and synchron bursts. Layer 1 acknowledges that by indication. The sb_flag indicates whether layer 1 should read the bursts correct or the bursts has failed.

- Variant A: arfcn = 23
- Variant B: arfcn = 14
- Variant C: arfcn = 124
- Variant D: arfcn = 1

Preamble: none

Variants: <A>..<>D>



Parametrization

Primitive	Parameter	Value
(3) MPHC_NETWORK_SYNC_REQ		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	fn_offset	NOT_USED
	time_alignment	NOT_USED
	timing_validity	TV_INVALID_TIMING_INFO
	search_mode	SM_WIDE_MODE

(4) MPHC_NETWORK_SYNC_IND

<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	sb_flag	SB_NOT_FOUND
	fn_offset	FN_OFFSET_0
	time_alignment	TIME_ALIGNMT_0
	bsic	BSIC_1

History: 3-Aug-01 MSB Initial

4.1.5 AIR010: Timeout

Description: Wait 1 second (Timeout).
 Adds 1 second to the default waiting time of 10 seconds.

Preamble: none

MM	RR/DL/ALR	L1
TIMEOUT (1000)		

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
------------------	------------------	--------------

History: 3-Aug-01 MSB Initial

4.1.6 AIR011: Timeout Wait

Description: Wait 1 second (Timeout Wait).
 No primitives are expected by the entity for 1 second. The TAP pauses for one second.

Preamble: none

MM	RR/DL/ALR	L1
TIMEOUT_WAIT (1000)		

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
------------------	------------------	--------------

History: 3-Aug-01 MSB Initial

4.1.7 AIR012: Mute

Description: Wait 1 second (Mute).
 The test will fail if the entity sends any primitives during the time of 1 second.

Preamble: none

MM	RR/DL/ALR	L1
MUTE (1000)		

Parametrization

Primitive	Parameter	Value
History:	3-Aug-01	MSB Initial

4.1.8 AIR013: DummyWait

Description: Wait 10 second for invalid Message. The Primitive MPH_C_RA_CON will be never appear in direction to layer 1. Therefore the test will fail after the default waiting time of 10 seconds.

Preamble: none

MM	RR/DL/ALR	L1
(1)	MPHC_RA_CON	
	=====	

Parametrization

Primitive	Parameter	Value
(1) MPH_C_RA_CON	fn	NOT_USED
	channel_request	NOT_USED

History: 3-Aug-01 MSB Initial

4.1.9 AIR020: Serving Cell Selection

Description: ALR selects the serving cell and layer 1 acknowledges it.
 Variant A-D: several arfcn's with fn_offset and time_alignment set to 0
 Variant E: new serving cell arfcn=14, with relevant fn_offset and time_alignment
 Variant F: new serving cell arfcn=124, with relevant fn_offset and time_alignment

Preamble: none

Variants: <A>..<<F>

MM	RR/DL/ALR	L1
(1)	MPHC_NEW_SCELL_REQ	
	=====	
(2)	MPHC_NEW_SCELL_CON	
	<=====	

Parametrization

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

(1) MPH_C_NEW_SCELL_REQ		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
<E>	radio_freq	ARFCN_14
<F>	radio_freq	ARFCN_124
<A>	fn_offset	FN_OFFSET_0
	fn_offset	FN_OFFSET_0
<C>	fn_offset	FN_OFFSET_0
<D>	fn_offset	FN_OFFSET_0
<E>	fn_offset	FN_OFFSET_114
<F>	fn_offset	FN_OFFSET_110
<A>	time_alignment	TIME_ALIGNMT_0
	time_alignment	TIME_ALIGNMT_0
<C>	time_alignment	TIME_ALIGNMT_0
<D>	time_alignment	TIME_ALIGNMT_0
<E>	time_alignment	TIME_ALIGNMT_14
<F>	time_alignment	TIME_ALIGNMT_110
	tsc	BSIC_1

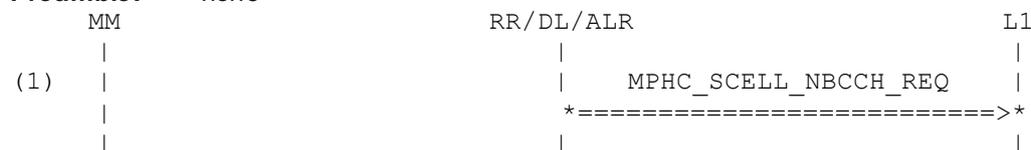
(2) MPH_C_NEW_SCELL_CON		
	param	NOT_USED

History: 7-Aug-01 MSB Initial
 17-Oct-01 MSB New variant F

4.1.10 AIR030: Read Serving Cell BCCH

Description: ALR requests full reading of the normal BCCH of the serving cell

Preamble: none



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) MPH_C_SCELL_NBCCH_REQ	schedule_array_size	SCHED_SIZE_1
	schedule_array	FULL_READ_ARRAY

History: 7-Aug-01 MSB Initial

4.1.11 AIR035: Serving Cell System Informations

Description: After the request of ALR for full reading of the normal serving cell BCCH , layer 1 forwards subsequent received data blocks (system info's) on BCCH.

Variant(s): several arfcn's

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)		MPHC_DATA_IND	
		<=====	
(2)		MPHC_DATA_IND	
		<=====	
(3)		MPHC_DATA_IND	
		<=====	
(4)		MPHC_DATA_IND	
		<=====	

Parametrization

Primitive	Parameter	Value
(1) MPHC_DATA_IND		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	l2_channel	L2_CHANNEL_NBCCH
	error_flag	VALID_BLOCK
	l2_frame	L2_SYS_INFO_1
	tc	TC_0
	ccch_lev	NOT_USED
	fn	FN_OFFSET_0
(2) MPHC_DATA_IND		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	l2_channel	L2_CHANNEL_NBCCH
	error_flag	VALID_BLOCK
	l2_frame	L2_SYS_INFO_2
	tc	TC_1
	ccch_lev	NOT_USED
	fn	FN_OFFSET_0

(3) MPH_C_DATA_IND

<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	l2_channel	L2_CHANNEL_NBCCH
	error_flag	VALID_BLOCK
<A>	l2_frame	L2_SYS_INFO_3_LAI2
	l2_frame	L2_SYS_INFO_3_LAI1
<C>	l2_frame	L2_SYS_INFO_3_LAI2
<D>	l2_frame	L2_SYS_INFO_3_LAI1
	tc	TC_2
	ccch_lev	NOT_USED
	fn	FN_OFFSET_0

(4) MPH_C_DATA_IND

<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	l2_channel	L2_CHANNEL_NBCCH
	error_flag	VALID_BLOCK
<A>	l2_frame	L2_SYS_INFO_4_LAI2
	l2_frame	L2_SYS_INFO_4_LAI1
<C>	l2_frame	L2_SYS_INFO_4_LAI2
<D>	l2_frame	L2_SYS_INFO_4_LAI1
	tc	TC_3
	ccch_lev	NOT_USED
	fn	FN_OFFSET_0

History: 7-Aug-01 MSB Initial

4.1.12 AIR036: Invalid Serving Cell System Informations

Description: After the request of ALR for full reading of the normal BCCH, Layer 1 forwards subsequent received data blocks (system info's) on BCCH. In this test case the received frame is invalid.

Variant(s): several arfcn's

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)		MPH_C_DATA_IND	
		<=====	

Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
------------------	------------------	--------------

(5) MPHC_DATA_IND

<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	l2_channel	L2_CHANNEL_NBCCH
	error_flag	INVALID_BLOCK
	l2_frame	L2_NO_CONTENT
	tc	TC_0
	cch_lev	NOT_USED
	fn	FN_OFFSET_0

History: 15-Aug-01 MSB Initial

4.1.13 AIR039: Stop Serving Cell Reading BCCH

Description: Stop of BCCH is requested and acknowledged.

Preamble: none

	MM	RR/DL/ALR	L1
(1)		MPHC_STOP_SCELL_BCCH_REQ	
		*=====	
(2)		MPHC_STOP_SCELL_BCCH_CON	
		*<=====	

Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1)	MPHC_STOP_SCELL_BCCH_REQ	param	NOT_USED
(2)	MPHC_STOP_SCELL_BCCH_CON	param	NOT_USED

History: 7-Aug-01 MSB Initial

4.1.14 AIR040: Read Neighbour Cell BCCH

Description: Read non-serving cell BCCH.

Variant(s): several arfcn's

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)		MPHC_NCELL_BCCH_REQ	
		*=====	

Parametrization

	<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
--	------------------	------------------	--------------

(1) MPMC_NCELL_BCCH_REQ		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
<A>	fn_offset	FN_OFFSET_23
	fn_offset	FN_OFFSET_114
<C>	fn_offset	FN_OFFSET_224
<D>	fn_offset	FN_OFFSET_101
<A>	time_alignment	TIME_ALIGNMT_23
	time_alignment	TIME_ALIGNMT_14
<C>	time_alignment	TIME_ALIGNMT_124
<D>	time_alignment	TIME_ALIGNMT_1
	tsc	BSIC_1
	bcch_blocks_required	NCELL_BCCH_SI_3_4
	gprs_prio	NOT_USED

History: 15-Aug-01 MSB Initial

4.1.15 AIR045: Neighbour Cell System Information

Description: Read non-serving cell BCCH system information.
 Variant(s): several arfcn's

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)			
		MPMC_NCELL_BCCH_IND	
		* <=====*	

Parametrization

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

(1) MPMC_NCELL_BCCH_IND		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	l2_channel	L2_CHANNEL_NBCCH
	error_flag	VALID_BLOCK
<A>	l2_frame	L2_SYS_INFO_3_LAI2
	l2_frame	L2_SYS_INFO_3_LAI1
<C>	l2_frame	L2_SYS_INFO_3_LAI2
<D>	l2_frame	L2_SYS_INFO_3_LAI1
	tc	TC_2
<A>	fn	FN_OFFSET_23
	fn	FN_OFFSET_114
<C>	fn	FN_OFFSET_224
<D>	fn	FN_OFFSET_101

History: 15-Aug-01 MSB Initial

4.1.16 AIR048: Stop Neighbour Cell Synchronization

Description: L1 must remove the given radio frequencies from its neighbour synchronization pending list.

Variant(s): several arfcn's

Preamble: none

Variants: <A>..<<E>

MM	RR/DL/ALR	L1
(1)	MPHC_STOP_NCELL_SYNC_REQ	
	=====>*	

Parametrization

Primitive	Parameter	Value
(1) MPHC_STOP_NCELL_SYNC_REQ		
<A>	radio_freq_array_size	CHANNELS_1
	radio_freq_array_size	CHANNELS_1
<C>	radio_freq_array_size	CHANNELS_1
<D>	radio_freq_array_size	CHANNELS_1
<E>	radio_freq_array_size	NOT_USED
<A>	radio_freq_array	STOP_BCCH_ARRAY23
	radio_freq_array	STOP_BCCH_ARRAY14
<C>	radio_freq_array	STOP_BCCH_ARRAY124
<D>	radio_freq_array	STOP_BCCH_ARRAY1
<E>	radio_freq_array	NOT_USED

History: 16-Aug-01 MSB Initial

4.1.17 AIR049: Stop Neighbour Cell Reading BCCH

Description: Stop of non-serving cell BCCH reading is requested and acknowledged.

Variant(s): several arfcn's

Preamble: none

Variants: <A>..<<E>

MM	RR/DL/ALR	L1
(1)	MPHC_STOP_NCELL_BCCH_REQ	
	=====>*	
(2)	MPHC_STOP_NCELL_BCCH_CON	
	<=====	

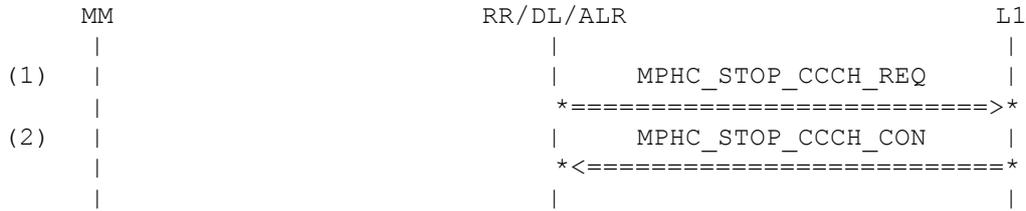
Parametrization

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

4.1.19 AIR061: Stop CCCH Reading

Description: Layer 1 is stopping the read of the common channel info CCCH.

Preamble: none



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
(1) MPHC_STOP_CCCH_REQ	param	NOT_USED
(2) MPHC_STOP_CCCH_CON	param	NOT_USED

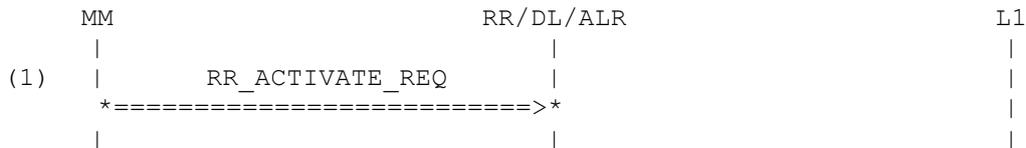
History: 16-Okt-01 MSB Initial

4.2 Cell Selection and Reselection

4.2.1 AIR100: Start cell selection

Description: A cell selection without BCCH information is started.
 Variant A: no SIM is inserted.
 Variant B: SIM is inserted, test SIM card, IMSI in the HPLMN, TMSI
 Variant C: SIM is inserted, normal SIM card, IMSI in the HPLMN, no TMSI
 Variant D: SIM is inserted, test SIM card, IMSI in the HPLMN, no TMSI
 Variant E: SIM is inserted, normal SIM card, IMSI in the HPLMN, TMSI

Preamble: none
Variants: <A>..<<E>



Parametrization

<u>Primitive</u>	<u>Parameter</u>	<u>Value</u>
------------------	------------------	--------------

(1) RR_ACTIVATE_REQ

<A>	plmn	PLMN_ID_EMPTY
	plmn	PLMN_ID_123
<C>	plmn	PLMN_ID_123
<D>	plmn	PLMN_ID_123
<E>	plmn	PLMN_ID_123
<A>	op	OP_MODE_EMPTY
	op	OP_MODE_TEST_SIM
<C>	op	OP_MODE_NORMAL
<D>	op	OP_MODE_TEST_SIM
<E>	op	OP_MODE_NORMAL
<A>	cksn	CKSN_NOT_PRES
	cksn	CKSN_NOT_PRES
<C>	cksn	CKSN_NOT_PRES
<D>	cksn	CKSN_NOT_PRES
<E>	cksn	CKSN_6
<A>	kcv	KCV_EMPTY
	kcv	KCV_12345678
<C>	kcv	KCV_12345678
<D>	kcv	KCV_12345678
<E>	kcv	KCV_12345678
<A>	acc	ACC_CTRL_CLASS_0000
	acc	ACC_CTRL_CLASS_0008
<C>	acc	ACC_CTRL_CLASS_0008
<D>	acc	ACC_CTRL_CLASS_4000
<E>	acc	ACC_CTRL_CLASS_0000
<A>	imsi	MOBILE_ID_NOT_SET
	imsi	MOBILE_ID_IMSI_HPLMN
<C>	imsi	MOBILE_ID_IMSI_HPLMN
<D>	imsi	MOBILE_ID_IMSI_HPLMN
<E>	imsi	MOBILE_ID_IMSI_HPLMN
<A>	tmsi	MOBILE_ID_NOT_SET
	tmsi	MOBILE_ID_TMSI
<C>	tmsi	MOBILE_ID_NOT_SET
<D>	tmsi	MOBILE_ID_NOT_SET
<E>	tmsi	MOBILE_ID_TMSI
<A>	thplmn	TIME_HPLMN_EMPTY
	thplmn	TIME_HPLMN_VALID
<C>	thplmn	TIME_HPLMN_VALID
<D>	thplmn	TIME_HPLMN_VALID
<E>	thplmn	TIME_HPLMN_VALID
	bcch_info	S_BCCH_INFO_EMPTY
	cell_test	CELL_TEST_DISABLE
	gprs_indic	NOT_USED

History: 1-Aug-01

MSB

Initial

4.2.2 AIR101: Start cell selection with or without BCCH information (std 6)

Description: A cell selection without BCCH information is started.
 Variant A: no BCCH information on SIM, ignore PCM.
 Variant B: BCCH information on SIM GSM 900 channel not set, PCM sets channel
 Variant C: BCCH information on SIM GSM 900 channel set, PCM sets channel

Preamble: none

Variants: <A>..<C>

MM	RR/DL/ALR	L1
COMMAND (PL CONFIG STD=6)		
(1) RR_ACTIVATE_REQ		
----->		

Parametrization

Primitive	Parameter	Value
(1) RR_ACTIVATE_REQ	plmn	PLMN_ID_123
	op	OP_MODE_NORMAL
	cksn	CKSN_NOT_PRES
	kcv	KCV_12345678
	acc	ACC_CTRL_CLASS_0008
	imsi	MOBILE_ID_IMSI_HPLMN
	tmsi	MOBILE_ID_TMSI
	thplmn	TIME_HPLMN_VALID
<A>	bcch_info	S_BCCH_INFO_EMPTY
	bcch_info	S_BCCH_INFO_NO_24
<C>	bcch_info	S_BCCH_INFO_24
	cell_test	CELL_TEST_DISABLE
	gprs_indic	NOT_USED

History: 1-Aug-01 MSB Initial

4.2.3 AIR105: Cell Selection Confirmation

Description: Confirmation of activation request
 Variant A: no SIM is inserted.
 Variant B: Test SIM card is inserted
 Variant C: Normal SIM card is inserted

Preamble: none

Variants: <A>..<C>

MM	RR/DL/ALR	L1
(1) RR_ACTIVATE_CNF		
<-----		

Parametrization

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

(1) RR_ACTIVATE_CNF			
<A>	op		OP_MODE_EMPTY
	op		OP_MODE_TEST_SIM
<C>	op		OP_MODE_NORMAL
	mm_info		MM_INFO_1
	cid		NOT_USED
	plmn		NOT_USED
	lac		NOT_USED
	power		NOT_USED
	gprs_indic		GPRS_NO

History: 14-Aug-01 MSB Initial

4.2.4 AIR110: Synchronization of RR and MM

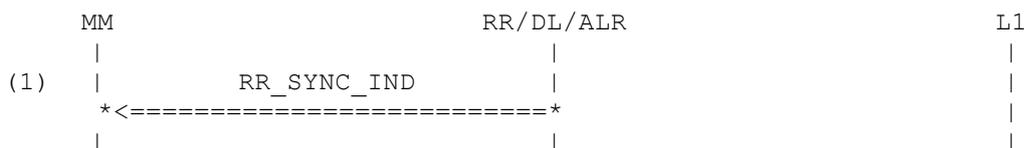
Description: Synchronization of RR and MM.

Variant A: synnccs: not present

Variant B: synnccs: SYNCCS_IDLE_SELECTION

Preamble: none

Variants: <A>..



Parametrization

Primitive	Parameter	Value
(2) RR_SYNC_IND		
	ciph	NOT_PRESENT_8BIT
	mm_info	NOT_USED
	bcch_info	NOT_USED
<A>	synnccs	NOT_PRESENT_16BIT
	synnccs	SYNCCS_IDLE_SELECTION
	chm	NOT_USED

History: 14-Aug-01 MSB Initial
 17-Oct-01 MSB New variant B

4.3 Idle Mode

4.3.1 AIR200: Periodically Measurement Report Request

Description: Request for measurement of cell signal strengths during dedicated mode.

Variant A: Only serving cell in channel list, ba_id=1

Variant B: Serving cell and 3 neighbour cells (14,124,1) in channel list, ba_id=2

Variant C: Any cell allowed in channel list, ba_id=not used

Preamble: none

Variants: <A>..<C>

MM	RR/DL/ALR	L1
(1)	MPHC_RXLEV_PERIODIC_REQ	
	=====>	

Parametrization

Primitive	Parameter	Value
(1) MPHC_RXLEV_PERIODIC_REQ		
<A>	chan_list	CHLIST_23
	chan_list	CHLIST_23_14_124_1
<C>	chan_list	CHLIST_ANY
<A>	num_of_chans	CHANNELS_1
	num_of_chans	CHANNELS_4
<C>	num_of_chans	NOT_USED
<A>	ba_id	BA_ID_1
	ba_id	BA_ID_2
<C>	ba_id	NOT_USED
	next_radio_freq_measured	CHAN_LIST_IDX_0

History: 10-Aug-01 MSB Initial
 17-Oct-01 MSB Variant C changed

4.3.2 AIR201: Stop Periodically Measurements

Description: Stops the periodically report of cell signal strength. Only Trigger, no parameter (only dummy).

Preamble: none

MM	RR/DL/ALR	L1
(1)	MPHC_STOP_RXLEV_PERIODIC_REQ	
	=====>	
(1)	MPHC_STOP_RXLEV_PERIODIC_CON	
	<=====	

Parametrization

Primitive	Parameter	Value
(1) MPHC_STOP_RXLEV_PERIODIC_REQ	param	NOT_USED
(2) MPHC_STOP_RXLEV_PERIODIC_CON	param	NOT_USED

History: 17-Aug-01 NN Initial

4.3.3 AIR205: Periodically Measurement Report Indication

Description: Measurement indication of cell signal strengths during dedicated mode.

Variant A: Only serving cell measurements, ba_id = 0

Variant B: Measurement of serving cell (23) and 3 neighbour cells (14,124,1), ba_id=0

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)		MPHC_RXLEV_PERIODIC_IND	
		<=====>	

Parametrization

Primitive	Parameter	Value
(1) MPHC_RXLEV_PERIODIC_IND		
<A>	result	NCELL_RESULT_ARRAY1A
	result	NCELL_RESULT_ARRAY2A
<C>	result	NCELL_RESULT_ARRAY1B
<D>	result	NCELL_RESULT_ARRAY2B
	nbr_of_carriers	CHANNELS_8
<A>	s_rxlev	RXLEV_56
	s_rxlev	RXLEV_56
<C>	s_rxlev	RXLEV_10
<D>	s_rxlev	RXLEV_10
	ba_id	BA_ID_0

History: 10-Aug-01 MSB Initial

4.3.4 AIR210: Read Neighbour Cell in Idle mode

Description: Read Non-Serving Cell SCH in Idle mode

Variant(s): several ARFCN's

Preamble: none

Variants: <A>..<D>

	MM	RR/DL/ALR	L1
(1)		MPHC_NCELL_SYNC_REQ	
		=====>	

Parametrization

Primitive	Parameter	Value
(1) MPHC_NCELL_SYNC_REQ		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
	fn_offset	NOT_USED
	time_alignment	NOT_USED
	timing_validity	TV_INVALID_TIMING_INFO

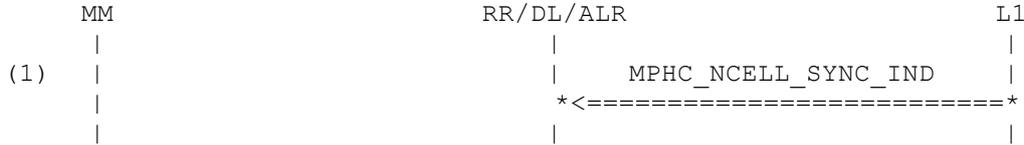
History: 15-Aug-01 MSB Initial

4.3.5 AIR215: Answer to the Neighbour Cell Synchronization Request

Description: Indication message to answer the individual neighbour cell synchronization request.
 Variant(s): several ARFCN's

Preamble: none

Variants: <A>..<D>



Parametrization

Primitive	Parameter	Value
(1) MPHC_NCELL_SYNC_IND		
<A>	radio_freq	ARFCN_23
	radio_freq	ARFCN_14
<C>	radio_freq	ARFCN_124
<D>	radio_freq	ARFCN_1
<A>	sb_flag	SB_FOUND
	fn_offset	FN_OFFSET_23
<C>	fn_offset	FN_OFFSET_114
<D>	fn_offset	FN_OFFSET_224
<A>	fn_offset	FN_OFFSET_101
	time_alignment	TIME_ALIGNMT_23
<C>	time_alignment	TIME_ALIGNMT_14
<D>	time_alignment	TIME_ALIGNMT_124
<D>	time_alignment	TIME_ALIGNMT_1
	bsic	BSIC_1

History: 15-Aug-01 MSB Initial

5 Suites

/* Internal Suites, Named Test Cases */

/* Configuration of TAP */

```
INT_Routing: AIR000;
INT_Wait1s: AIR010;
INT_Pause1s: AIR011;
INT_Mute1s: AIR012;
INT_DummyWait: AIR013;
```

/* Start cell selection */

```
INT_ActivateReqLim: AIR100A; /* RR_ACTIVATE_REQ */
INT_ActivateReqFull: AIR100B; /* RR_ACTIVATE_REQ */
INT_ActivateCnfNoSIM: AIR105A; /* RR_ACTIVATE_CNF */
INT_ActivateCnfTestSIM: AIR105B; /* RR_ACTIVATE_CNF */
INT_ActivateCnfSIM: AIR105C; /* RR_ACTIVATE_CNF */
INT_SyncInd: AIR110A; /* RR_SYNC_IND */
INT_SyncIndB: AIR110B; /* RR_SYNC_IND */
```

/* Measurements */

```
INT_Meas0: AIR001A; /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
```

```

INT_Meas1: AIR001B;          /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
INT_Meas2: AIR001C;          /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
INT_Meas3: AIR001D;          /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
INT_Meas4: AIR001E;          /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
INT_Meas5: AIR001F;          /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
INT_Meas6: AIR001G;          /* 5* MPHC_RXLEV_REQ, MPHC_RXLEV_IND */
INT_PeriodicMeasA: AIR200A; /* MPHC_RXLEV_PERIODIC_REQ */
INT_PeriodicMeasB: AIR200B; /* MPHC_RXLEV_PERIODIC_REQ */
INT_PeriodicMeas: AIR200C; /* MPHC_RXLEV_PERIODIC_REQ */
INT_MeasReportA1: AIR205A; /* MPHC_RXLEV_PERIODIC_IND */
INT_MeasReportA2: AIR205B; /* MPHC_RXLEV_PERIODIC_IND */
INT_MeasReportB1: AIR205C; /* MPHC_RXLEV_PERIODIC_IND */
INT_MeasReportB2: AIR205D; /* MPHC_RXLEV_PERIODIC_IND */
INT_StopMeasReport: AIR201; /* MPHC_STOP_RXLEV_PERIODIC_REQ/CON */

```

/* Network synchronisation */

```

INT_NetworkSync23: AIR005A; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSync14: AIR005B; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSync124: AIR005C; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSync1: AIR005D; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSyncFail23: AIR006A; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSyncFail14: AIR006B; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSyncFail124: AIR006C; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */
INT_NetworkSyncFail1: AIR006D; /* MPHC_NETWORK_SYNC_REQ,
MPHC_NETWORK_SYNC_IND */

```

/* Serving Cell BCCH Read & Selection */

```

INT_SCReadBCCH: AIR030; /* MPHC_SCELL_NBCCH_REQ */
INT_SCSysInfos23: AIR035A; /* 4* MPHC_DATA_IND */
INT_SCSysInfos14: AIR035B; /* 4* MPHC_DATA_IND */
INT_SCSysInfos124: AIR035C; /* 4* MPHC_DATA_IND */
INT_SCSysInfos1: AIR035D; /* 4* MPHC_DATA_IND */
INT_SCSysInfoInvalid23: AIR036A; /* MPHC_DATA_IND */
INT_SCSysInfoInvalid14: AIR036B; /* MPHC_DATA_IND */
INT_SCSysInfoInvalid124: AIR036C; /* MPHC_DATA_IND */
INT_SCSysInfoInvalid1: AIR036D; /* MPHC_DATA_IND */
INT_SCBCCH23: INT_SCReadBCCH, AIR035A; /* MPHC_SCELL_NBCCH_REQ, 4*
MPHC_DATA_IND */
INT_SCBCCH14: INT_SCReadBCCH, AIR035B; /* MPHC_SCELL_NBCCH_REQ, 4*
MPHC_DATA_IND */
INT_SCBCCH124: INT_SCReadBCCH, AIR035C; /* MPHC_SCELL_NBCCH_REQ, 4*
MPHC_DATA_IND */
INT_SCBCCH1: INT_SCReadBCCH, AIR035D; /* MPHC_SCELL_NBCCH_REQ, 4*
MPHC_DATA_IND */
INT_SCSelect23: AIR020A; /* MPHC_NEW_SCELL_REQ,
MPHC_NEW_SCELL_CON */
INT_SCSelect14: AIR020B; /* MPHC_NEW_SCELL_REQ,
MPHC_NEW_SCELL_CON */
INT_SCSelect14B: AIR020E; /* MPHC_NEW_SCELL_REQ, MPHC_NEW_SCELL_CON */
INT_SCSelect124: AIR020C; /* MPHC_NEW_SCELL_REQ, MPHC_NEW_SCELL_CON */

```

```

INT_SCSelect124B: AIR020F;          /* MPH_C_NEW_SCELL_REQ, MPH_C_NEW_SCELL_CON */
INT_SCSelect1: AIR020D;             /* MPH_C_NEW_SCELL_REQ,
MPH_C_NEW_SCELL_CON */
INT_SCStopBCCH: AIR039;            /* MPH_C_STOP_SCELL_BCCH_REQ,
MPH_C_STOP_SCELL_BCCH_CON */
INT_CCCHStartA: AIR060A;           /* MPH_C_START_CCCH_REQ */
INT_CCCHStartB: AIR060B;           /* MPH_C_START_CCCH_REQ */
INT_CCCHStartC: AIR060C;           /* MPH_C_START_CCCH_REQ */
INT_StopCCCH: AIR061;              /* MPH_C_STOP_CCCH_REQ/CON */
INT_BCCHConfigA: INT_SCStopBCCH,
INT_CCCHStartA;                    /* MPH_C_STOP_SCELL_BCCH_REQ/CON,
MPH_C_START_CCCH_REQ */
INT_BCCHConfigB: INT_SCStopBCCH,
INT_CCCHStartB;                    /* MPH_C_STOP_SCELL_BCCH_REQ/CON,
MPH_C_START_CCCH_REQ */
INT_BCCHConfigC: INT_SCStopBCCH,
INT_CCCHStartC;                    /* MPH_C_STOP_SCELL_BCCH_REQ/CON,
MPH_C_START_CCCH_REQ */

```

/* NC process */

```

INT_NC23: INT_NetworkSync23, INT_SCSelect23, INT_SCBCCH23;
INT_NC14: INT_NetworkSync14, INT_SCSelect14, INT_SCBCCH14;
INT_NC124: INT_NetworkSync124, INT_SCSelect124, INT_SCBCCH124;
INT_NC1: INT_NetworkSync1, INT_SCSelect1, INT_SCBCCH1;
INT_NCPProcessOld: INT_NC23, INT_SyncInd, INT_SCStopBCCH,
INT_NC14, INT_SCStopBCCH,
INT_NC124, INT_SCStopBCCH,
INT_NC1, INT_SCStopBCCH;
INT_NCPProcess: INT_NC23, INT_SyncInd, INT_SCStopBCCH, INT_CCCHStartA,
INT_SCBCCH14, INT_SCStopBCCH,
INT_NC124, INT_SCStopBCCH,
INT_NC1, INT_SCStopBCCH;
INT_NCRReadBCCH23: AIR040A;         /* MPH_C_NCELL_BCCH_REQ */
INT_NCRReadBCCH14: AIR040B;        /* MPH_C_NCELL_BCCH_REQ */
INT_NCRReadBCCH124: AIR040C;       /* MPH_C_NCELL_BCCH_REQ */
INT_NCRReadBCCH1: AIR040D;         /* MPH_C_NCELL_BCCH_REQ */
INT_NCSysInfo23: AIR045A;           /* MPH_C_NCELL_BCCH_IND */
INT_NCSysInfo14: AIR045B;           /* MPH_C_NCELL_BCCH_IND */
INT_NCSysInfo124: AIR045C;         /* MPH_C_NCELL_BCCH_IND */
INT_NCSysInfo1: AIR045D;           /* MPH_C_NCELL_BCCH_IND */
INT_NCStopBCCH23: AIR049A;          /* MPH_C_STOP_NCELL_BCCH_REQ,
MPH_C_STOP_NCELL_BCCH_CON */
INT_NCStopBCCH14: AIR049B;          /* MPH_C_STOP_NCELL_BCCH_REQ,
MPH_C_STOP_NCELL_BCCH_CON */
INT_NCStopBCCH124: AIR049C;        /* MPH_C_STOP_NCELL_BCCH_REQ,
MPH_C_STOP_NCELL_BCCH_CON */
INT_NCStopBCCH1: AIR049D;           /* MPH_C_STOP_NCELL_BCCH_REQ,
MPH_C_STOP_NCELL_BCCH_CON */
INT_NCStopBCCH: AIR049E;            /* MPH_C_STOP_NCELL_BCCH_REQ,
MPH_C_STOP_NCELL_BCCH_CON */
INT_NCStopSync23: AIR048A;          /* MPH_C_STOP_NCELL_SYNC_REQ */
INT_NCStopSync14: AIR048B;          /* MPH_C_STOP_NCELL_SYNC_REQ */
INT_NCStopSync124: AIR048C;         /* MPH_C_STOP_NCELL_SYNC_REQ */
INT_NCStopSync1: AIR048D;           /* MPH_C_STOP_NCELL_SYNC_REQ */
INT_NCStopSync: AIR048E;            /* MPH_C_STOP_NCELL_SYNC_REQ */

```

/* Cell selection (error free)*/

```

INT_CSLim4: INT_Routing, INT_ActivateReqLim, INT_Meas0;

```

INT_CSLim2: INT_Routing, INT_ActivateReqLim, INT_Meas5;
INT_CSLim1: INT_Routing, INT_ActivateReqLim, INT_Meas6;
INT_CSFull4: INT_Routing, INT_ActivateReqFull, INT_Meas0;

/* Idle Mode */

INT_NCSyncReq23: AIR210A; /* MPH_C_NCELL_SYNC_REQ */
INT_NCSyncReq14: AIR210B; /* MPH_C_NCELL_SYNC_REQ */
INT_NCSyncReq124: AIR210C; /* MPH_C_NCELL_SYNC_REQ */
INT_NCSyncReq1: AIR210D; /* MPH_C_NCELL_SYNC_REQ */
INT_NCSyncInd23: AIR215A; /* MPH_C_NCELL_SYNC_IND */
INT_NCSyncInd14: AIR215B; /* MPH_C_NCELL_SYNC_IND */
INT_NCSyncInd124: AIR215C; /* MPH_C_NCELL_SYNC_IND */
INT_NCSyncInd1: AIR215D; /* MPH_C_NCELL_SYNC_IND */

/* SUI_CellSelectionLim */

/* SUI_CellSelectionLim:
Power measurement (MPHC_RXLEV_REQ/IND),
Cell selection network synchronization (MPHC_NETWORK_SYNC_REQ/IND) ,
New serving cell selection (MPHC_NEW_SCELL_REQ/CON),
Read serving cell BCCH (MPHC_SCELL_NBCCH_REQ, 4* MPH_C_DATA_IND) ,
Read common control channel info, configuration of serving cell
(MPHC_START_CCCH_REQ),
Read serving cell BCCH (MPHC_SCELL_NBCCH_REQ) */

SUI_CellSelectionLim:

INT_CSLim4,
INT_NetworkSync23, /* MPH_C_NETWORK_SYNC_REQ/IND */
INT_SCSelect23, /* MPH_C_NEW_SCELL_REQ/CON */
INT_SCBCCH23, /* MPH_C_SCELL_NBCCH_REQ, 4* MPH_C_DATA_IND */
INT_SyncInd, /* RR_SYNC_IND */
INT_BCCHConfigA, /* MPH_C_STOP_SCELL_BCCH_REQ, MPH_C_START_CCCH_REQ
*/
INT_SCReadBCCH, /* MPH_C_SCELL_NBCCH_REQ */
INT_ActivateCnfNoSIM; /* RR_ACTIVATE_CNF */

/* SUI_CellSelectionLim4:
Power measurement (MPHC_RXLEV_REQ/IND),
Cell selection network synchronization (MPHC_NETWORK_SYNC_REQ/IND) 1st failed,
Cell selection network synchronization (MPHC_NETWORK_SYNC_REQ/IND) 2nd is valid,
New serving cell selection (MPHC_NEW_SCELL_REQ/CON),
Read serving cell BCCH (MPHC_SCELL_NBCCH_REQ, 4* MPH_C_DATA_IND) ,
Read common control channel info, configuration of serving cell
(MPHC_START_CCCH_REQ),
Read serving cell BCCH (MPHC_SCELL_NBCCH_REQ) */

SUI_CellSelectionLim4:

INT_CSLim4,
INT_NetworkSyncFail23, /* MPH_C_NETWORK_SYNC_REQ/IND */
INT_NetworkSync14, /* MPH_C_NETWORK_SYNC_REQ/IND */
INT_SCSelect14, /* MPH_C_NEW_SCELL_REQ/CON */
INT_SCReadBCCH, /* MPH_C_SCELL_NBCCH_REQ */
INT_SCSysInfoInvalid14, /* MPH_C_DATA_IND */
INT_SCSysInfos14, /* 4*MPH_C_DATA_IND */
INT_SyncInd, /* RR_SYNC_IND */
INT_BCCHConfigA, /* MPH_C_STOP_SCELL_BCCH_REQ, MPH_C_START_CCCH_REQ
*/
INT_SCReadBCCH, /* MPH_C_SCELL_NBCCH_REQ */
INT_ActivateCnfNoSIM; /* RR_ACTIVATE_CNF */

/* SUI_CellSelectionFull */

```
/* SUI_CellSelectionFull:
   Power measurement (MPHC_RXLEV_REQ/IND),
   Neighbour cell process for 1 cell (if suitable),
   Serving cell selection
   Read common control channel info, configuration of serving cell
(MPHC_START_CCCH_REQ),
   Read serving cell BCCH (MPHC_SCELL_NBCCH_REQ) */
SUI_CellSelectionFull:
   INT_CSFull4,
   INT_NC23,
   INT_SyncInd,          /* RR_SYNC_IND */
   INT_BCCHConfigB,     /* MPHC_STOP_SCELL_BCCH_REQ, MPHC_START_CCCH_REQ
*/
   INT_SCReadBCCH,     /* MPHC_SCELL_NBCCH_REQ */
   INT_ActivateCnfTestSIM; /* RR_ACTIVATE_CNF */
```

/* SUI_Idle */

SUI_IdleLim:

```
SUI_CellSelectionLim,
INT_PeriodicMeas, /* MPHC_RXLEV_PERIODIC_REQ */
INT_MeasReportA2, /* MPHC_RXLEV_PERIODIC_IND */
INT_NCSyncReq14, /* MPHC_NCELL_SYNC_REQ */
INT_NCSyncReq124, /* MPHC_NCELL_SYNC_REQ */
INT_NCSyncReq1, /* MPHC_NCELL_SYNC_REQ */
INT_MeasReportA2, /* MPHC_RXLEV_PERIODIC_IND */
INT_NCSyncInd14, /* MPHC_NCELL_SYNC_IND */
INT_NCRReadBCCH14, /* MPHC_NCELL_BCCH_REQ */
INT_NCSysInfo14, /* MPHC_NCELL_BCCH_IND */
INT_NCStopBCCH14, /* MPHC_STOP_BCCH_REQ */
INT_NCSyncInd124, /* MPHC_NCELL_SYNC_IND */
INT_NCRReadBCCH124, /* MPHC_NCELL_BCCH_REQ */
INT_NCSysInfo124, /* MPHC_NCELL_BCCH_IND */
INT_NCStopBCCH124, /* MPHC_STOP_BCCH_REQ */
INT_MeasReportA2, /* MPHC_RXLEV_PERIODIC_IND */
INT_NCSyncInd1, /* MPHC_NCELL_SYNC_IND */
INT_NCRReadBCCH1, /* MPHC_NCELL_BCCH_REQ */
INT_NCSysInfo1, /* MPHC_NCELL_BCCH_IND */
INT_NCStopBCCH1, /* MPHC_STOP_BCCH_REQ */
INT_MeasReportA2, /* MPHC_RXLEV_PERIODIC_IND */
INT_Pause1s;
```

SUI_Idle:

```
SUI_CellSelectionFull,
INT_PeriodicMeas, /* MPHC_RXLEV_PERIODIC_REQ */
INT_MeasReportA2, /* MPHC_RXLEV_PERIODIC_IND */
INT_NCSyncReq14, /* MPHC_NCELL_SYNC_REQ */
INT_NCSyncReq124, /* MPHC_NCELL_SYNC_REQ */
INT_NCSyncReq1, /* MPHC_NCELL_SYNC_REQ */
INT_MeasReportA2, /* MPHC_RXLEV_PERIODIC_IND */
INT_NCSyncInd14, /* MPHC_NCELL_SYNC_IND */
INT_NCRReadBCCH14, /* MPHC_NCELL_BCCH_REQ */
INT_NCSysInfo14, /* MPHC_NCELL_BCCH_IND */
INT_NCStopBCCH14, /* MPHC_STOP_BCCH_REQ */
INT_NCSyncInd124, /* MPHC_NCELL_SYNC_IND */
```

```
INT_NCReadBCCH124,      /* MPMC_NCELL_BCCH_REQ */
INT_NCSysInfo124,      /* MPMC_NCELL_BCCH_IND */
INT_NCStopBCCH124,    /* MPMC_STOP_BCCH_REQ */
INT_MeasReportA2,     /* MPMC_RXLEV_PERIODIC_IND */
INT_NCSyncInd1,       /* MPMC_NCELL_SYNC_IND */
INT_NCReadBCCH1,      /* MPMC_NCELL_BCCH_REQ */
INT_NCSysInfo1,       /* MPMC_NCELL_BCCH_IND */
INT_NCStopBCCH1,     /* MPMC_STOP_BCCH_REQ */
INT_MeasReportA2;     /* MPMC_RXLEV_PERIODIC_IND */
```

/* SUI_CellReselection */

INT_SCReduceLev10:

```
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2,
INT_MeasReportB2;
```

SUI_CellReselection:

```
/* get idle mode with serving cell 23 */
SUI_Idle,
/* reduce level of current serving cell 23 */
INT_SCReduceLev10,
/* start cell reselection to cell 14
   (cell 14 has the highest filed strength but don't has the right color code) */
INT_StopMeasReport, /* MPMC_STOP_RXLEV_PERIODIC_REQ/CON */
INT_NCStopSync,     /* MPMC_STOP_NCELL_SYNC_REQ */
INT_NCStopBCCH,    /* MPMC_STOP_NCELL_BCCH_REQ,
MPMC_STOP_NCELL_BCCH_CON */
INT_SCStopBCCH,    /* MPMC_STOP_SCELL_BCCH_REQ,
MPMC_STOP_SCELL_BCCH_CON */
INT_SCSelect14B,   /* MPMC_NEW_SCELL_REQ, MPMC_NEW_SCELL_CON with fn
and ta */
INT_CCCHStartC,   /* MPMC_START_CCCH_REQ */
INT_SCReadBCCH,   /* MPMC_SCELL_NBCCH_REQ */
INT_SCSysInfos14, /* 4*MPMC_DATA_IND */
/* start cell reselection to cell 124 (the next highest field strength and the right color code) */
INT_SCSelect124B, /* MPMC_NEW_SCELL_REQ, MPMC_NEW_SCELL_CON with fn
and ta */
INT_CCCHStartC,   /* MPMC_START_CCCH_REQ */
INT_SCReadBCCH,   /* MPMC_SCELL_NBCCH_REQ */
INT_SCSysInfos124, /* 4*MPMC_DATA_IND */
INT_SCStopBCCH,   /* MPMC_STOP_SCELL_BCCH_REQ */
INT_SyncInd,      /* RR_SYNC_IND */
INT_CCCHStartB,   /* MPMC_START_CCCH_REQ */
INT_SCReadBCCH,   /* MPMC_SCELL_NBCCH_REQ */
INT_SyncIndB,     /* RR_SYNC_IND */
INT_PeriodicMeas, /* MPMC_RXLEV_PERIODIC_REQ */
INT_MeasReportA2; /* MPMC_RXLEV_PERIODIC_IND */
/*INT_DummyWait;*/
```

/*

History:	16-Aug-01	MSB	Initial
	17-Oct-01	MSB	Cell Reselection

*/

/* end of suites

ENDSUITES

*/

Appendices

A. Acronyms

DS-WCDMA Direct Sequence/Spread Wideband Code Division Multiple Access

B. Glossary

International Mobile Telecommunication 2000 (IMT-2000/ITU-2000) Formerly referred to as FPLMTS (Future Public Land-Mobile Telephone System), this is the ITU's specification/family of standards for 3G. This initiative provides a global infrastructure through both satellite and terrestrial systems, for fixed and mobile phone users. The family of standards is a framework comprising a mix/blend of systems providing global roaming. <URL: <http://www.imt-2000.org/>>