



Technical Document – Confidential

GSM PROTOCOL STACK
MULTILAYER TEST SPECIFICATION
SHORT MESSAGE SERVICES

Document Number:	6147.411.97.100
Version:	0.2
Status:	Draft
Approval Authority:	
Creation Date:	1998-Jan-26
Last changed:	2015-Mar-08 by XGUTTEFE
File Name:	msms.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
1998-Jan-26	LE, MS		0.1		1
2003-May-21	XGUTTEFE		0.2	Draft	

Notes:

1. Initial version

Table of Contents

1.1	References	4
1.2	Abbreviations.....	7
1.3	Terms	9
2	Overview	10
3	Parameters.....	12
4	TEST CASES.....	56
4.1	Preambles	56
4.1.1	MSMS000: Initialization (One cell).....	56
4.2	Short message service point to point.....	57
4.2.1	MSMS001: SMS mobile terminated (34.2.1)	57
4.2.2	MSMS002: SMS mobile originated (34.2.2)	64
4.2.3	MSMS003: Test of memory full condition and memory available notification (34.2.3) ..	68
4.2.4	MSMS004: Test of the status report capabilities and of SMS-COMMAND (34.2.4)	73
4.3	Test of message class 0 to 3	75
4.3.1	MSMS005: Short message class 0 (34.2.5.1).....	75
4.3.2	MSMS006: Test of class 1 short messages (34.2.5.2)	76
4.3.3	MSMS007: Test of class 2 short messages (34.2.5.3)	77
4.4	Short message service point to point.....	80
4.4.1	MSMS008: Test of the replace mechanism for SM type 1-7 (34.2.7).....	80
4.4.2	MSMS009: Test of the reply path scheme (34.2.8)	82
4.4.3	MSMS010: Cell Broadcast (34.3)	84
4.5	Additional SMS Testcases	85
4.5.1	MSMS500: Write and Read of a mobile originated SMS	85
	Appendices.....	86
A.	Acronyms	86
B.	Glossary.....	86

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

- [1] GSM 2.81, Line Identification Supplementary Services - Stage 1
ETS 300 514, ETSI, September 1994
- [2] GSM 2.82, Call Forwarding Supplementary Services - Stage 1
ETS 300 515, ETSI, September 1994
- [3] GSM 2.83, Call Waiting and Call Hold Supplementary Services - Stage 1
ETS 300 516, ETSI, September 1994
- [4] GSM 2.84, Multi Party Supplementary Services - Stage 1
ETS 300 517, ETSI, September 1994
- [5] GSM 2.85, Closed User Group Supplementary Services - Stage 1
ETS 300 518, ETSI, September 1994
- [6] GSM 2.86, Advice of Charge Supplementary Services - Stage 1
ETS 300 519, ETSI, September 1994
- [7] GSM 2.88, Call Barring Supplementary Services - Stage 1
ETS 300 520, ETSI, September 1994
- [8] GSM 3.14, Support of Dual Tone Multi Frequency Signalling via the GSM System
ETS 300 532, ETSI, April 1994
- [9] GSM 3.40, Technical Realization of the Short Message Service Point-to-Point
ETS 300 536, ETSI, January 1996
- [10] GSM 3.41, Technical Realization of Short Message Service Cell Broadcast
ETS 300 537, ETSI, June 1995
- [11] GSM 3.81, Line Identification Supplementary Services - Stage 2
ETS 300 542, ETSI, February 1995
- [12] GSM 3.82, Call Forwarding Supplementary Services - Stage 2
ETS 300 543, ETSI, February 1995
- [13] GSM 3.83, Call Waiting and Call Hold Supplementary Services - Stage 2
ETS 300 544, ETSI, November 1994
- [14] GSM 3.84, Multi Party Supplementary Services - Stage 2
ETS 300 545, ETSI, November 1994
- [15] GSM 3.85, Closed User Group Supplementary Services - Stage 2
ETS 300 546, ETSI, January 1996
- [16] GSM 3.86, Advice of Charge Supplementary Services - Stage 2
ETS 300 547, ETSI, March 1995
- [17] GSM 3.88, Call Barring Supplementary Services - Stage 2
ETS 300 548, ETSI, November 1994
- [18] GSM 4.01, MS-BSS Interface General Aspects and Principles
ETS 300 550, ETSI, September 1994
- [18a] GSM 4.03, MS-BSS Interface Channel Structures and Access Capabilities
ETS 300 552, ETSI, September 1994
- [19] GSM 4.05, Data Link Layer General Aspects
ETS 300 554, ETSI, September 1994
- [20] GSM 4.06, MS-BSS Interface Data Link Layer Specification
ETS 300 555, ETSI, September 1994
- [21] GSM 4.07, Mobile Radio Interface Signalling Layer 3 General Aspects
ETS 300 556, ETSI, February 1995

- [22] GSM 4.08, Mobile Radio Interface Layer 3 Specification
ETS 300 557, ETSI, January 1996
- [23] GSM 4.10, Mobile Radio Interface Layer 3 Supplementary Services Specification
General Aspects
ETS 300 558, ETSI, February 1995
- [24] GSM 4.11, Point-to-Point Short Message Service Support on Mobile Radio Interface
ETS 300 559, ETSI, October 1995
- [25] GSM 4.12, Short Message Service Cell Broadcast Support on Mobile Radio Interface
ETS 300 560, ETSI, January 1996
- [26] GSM 4.80, Mobile Radio Interface Supplementary Services Specification Formats and Coding
ETS 300 564, ETSI, February 1995
- [27] GSM 4.81, Line Identification Supplementary Services - Stage 3
ETS 300 565, ETSI, February 1995
- [28] GSM 4.82, Call Forwarding Supplementary Services - Stage 3
ETS 300 566, ETSI, February 1995
- [29] GSM 4.83, Call Waiting and Call Hold Supplementary Services - Stage 3
ETS 300 567, ETSI, February 1995
- [30] GSM 4.84, Multi Party Supplementary Services - Stage 3
ETS 300 568, ETSI, February 1995
- [31] GSM 4.85, Closed User Group Supplementary Services - Stage 3
ETS 300 569, ETSI, February 1995
- [32] GSM 4.86, Advice of Charge Supplementary Services - Stage 3
ETS 300 570, ETSI, February 1995
- [33] GSM 4.88, Call Barring Supplementary Services - Stage 3
ETS 300 571, ETSI, February 1995
- [34] GSM 5.01, Physical Layer on the Radio Path General Description
ETS 300 573, ETSI, October 1995
- [35] GSM 5.02, Multiplexing and Multiple Access on the Radio Path
ETS 300 574, ETSI, January 1996
- [36] GSM 5.08, Radio Sub-system Link Control
ETS 300 578, ETSI, January 1996
- [37] GSM 5.10, Radio Sub-system Synchronisation
ETS 300 579, ETSI, October 1995
- [38] Service Access Point MMREG
6147.100.96.100; Condat GmbH
- [39] Service Access Point MNCC
6147.101.96.100; Condat GmbH
- [40] Service Access Point MNSS
6147.102.96.100; Condat GmbH
- [41] Service Access Point MNSMS
6147.103.96.100; Condat GmbH
- [42] Service Access Point MMCC
6147.104.97.100; Condat GmbH
- [43] Service Access Point MMSS
6147.105.97.100; Condat GmbH
- [44] Service Access Point MMSMS
6147.106.97.100; Condat GmbH

- [45] Service Access Point RR
6147.107.97.100; Condat GmbH
- [46] Service Access Point SIM
6147.108.97.100; Condat GmbH
- [47] Service Access Point MPH
6147.109.96.100; Condat GmbH
- [48] Service Access Point DL
6147.110.96.100; Condat GmbH
- [49] Service Access Point MDL
6147.111.96.100; Condat GmbH
- [50] Service Access Point PH
6147.112.97.100; Condat GmbH
- [51] Service Access Point MMI
6147.113.96.100; Condat GmbH
- [52] Message Sequence Charts CC
6147.200.97.100; Condat GmbH
- [53] Message Sequence Charts SS
6147.201.97.100; Condat GmbH
- [54] Message Sequence Charts SMS
6147.202.97.100; Condat GmbH
- [55] Message Sequence Charts MM
6147.203.97.100; Condat GmbH
- [56] Message Sequence Charts RR
6147.204.96.100; Condat GmbH
- [57] Message Sequence Charts DL
6147.205.96.100; Condat GmbH
- [58] Users Guide
6147.300.96.100; Condat GmbH
- [59] Test Specification CC
6147.400.97.100; Condat GmbH
- [60] Test Specification SS
6147.401.97.100; Condat GmbH
- [61] Test Specification SMS
6147.402.97.100; Condat GmbH
- [62] Test Specification MM
6147.403.97.100; Condat GmbH
- [63] Test Specification RR
6147.404.97.100; Condat GmbH
- [64] Test Specification DL
6147.405.97.100; Condat GmbH
- [65] Test Specification CCD
6147.406.97.100; Condat GmbH
- [66] SDL Specification CC
6147.500.97.100; Condat GmbH
- [67] SDL Specification SS
6147.501.97.100; Condat GmbH

- [68] SDL Specification SMS
6147.502.97.100; Condat GmbH
- [69] SDL Specification MM
6147.503.97.100; Condat GmbH
- [70] SDL Specification RR
6147.504.97.100; Condat GmbH
- [71] SDL Specification DL
6147.505.97.100; Condat GmbH
- [72] Message Specification CC
6147.600.97.100; Condat GmbH
- [73] Message Specification SS
6147.601.97.100; Condat GmbH
- [74] Message Specification SMS
6147.602.97.100; Condat GmbH
- [75] Message Specification MM
6147.603.97.100; Condat GmbH
- [76] Message Specification RR
6147.604.97.100; Condat GmbH
- [77] Message Specification DL
6147.605.97.100; Condat GmbH
- [78] Technical Documentation CC
6147.700.97.100; Condat GmbH
- [79] Technical Documentation SS
6147.701.97.100; Condat GmbH
- [80] Technical Documentation SMS
6147.702.97.100; Condat GmbH
- [81] Technical Documentation MM
6147.703.97.100; Condat GmbH
- [82] Technical Documentation RR
6147.704.97.100; Condat GmbH
- [83] Technical Documentation DL
6147.705.97.100; Condat GmbH
- [84] Technical Documentation CCD
6147.706.97.100; Condat GmbH

1.2 Abbreviations

AGCH	Access Grant Channel
BCCH	Broadcast Control Channel
BS	Base Station
BSIC	Base Station Identification Code
CBCH	Cell Broadcast Channel
CBQ	Cell Bar Qualify
CC	Call Control
CCCH	Common Control Channel
CCD	Condat Coder Decoder
CKSN	Ciphering Key Sequence Number
C/R	Command / Response

C1	Path Loss Criterion
C2	Reselection Criterion
DCCH	Dedicated Control Channel
DISC	Disconnect Frame
DL	Data Link Layer
DM	Disconnected Mode Frame
EA	Extension Bit Address Field
EL	Extension Bit Length Field
EMMI	Electrical Man Machine Interface
F	Final Bit
FACCH	Fast Associated Control Channel
FHO	Forced Handover
GP	Guard Period
GSM	Global System for Mobile Communication
HPLMN	Home Public Land Mobile Network
I	Information Frame
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
Kc	Authentication Key
L	Length Indicator
LAI	Location Area Information
LPD	Link Protocol Discriminator
M	More Data Bit
MCC	Mobile Country Code
MM	Mobility Management
MMI	Man Machine Interface
MNC	Mobile Network Code
MS	Mobile Station
NCC	National Colour Code
NECI	New Establishment Causes included
N(R)	Receive Number
N(S)	Send Number
OTD	Observed Time Difference
P	Poll Bit
PCH	Paging Channel
PDU	Protocol Description Unit
P/F	Poll / Final Bit
PL	Physical Layer
PLMN	Public Land Mobile Network
RACH	Random Access Channel
REJ	Reject Frame
RNR	Receive Not Ready Frame
RR	Radio Resource Management
RR	Receive Ready Frame
RTD	Real Time Difference
SABM	Set Asynchronous Balanced Mode
SACCH	Slow Associated Control Channel
SAP	Service Access Point
SAPI	Service Access Point Identifier
SDCCH	Slow Dedicated Control Channel
SIM	Subscriber Identity Module
SMS	Short Message Service
SMSCB	Short Message Service Cell Broadcast
SS	Supplementary Services
TCH	Traffic Channel
TCH/F	Traffic Channel Full Rate
TCH/H	Traffic Channel Half Rate
TDMA	Time Division Multiple Access
TMSI	Temporary Mobile Subscriber Identity

UA	Unnumbered Acknowledgement Frame
UI	Unnumbered Information Frame
VPLMN	Visiting Public Land Mobile Network
V(A)	Acknowledgement State Variable
V(R)	Receive State Variable
V(S)	Send State Variable

1.3 Terms

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

2 Overview

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

The 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.

Fehler! Es ist nicht möglich, durch die Bearbeitung von Feldfunktionen Objekte zu erstellen.

Figure 1: Mobile-station protocol architecture

This document describes the tests for the whole protocol stack according to GSM 11.10 chapter 26.8.

3 Parameters

```
#include "P_DL.H"
#define REST_OCTET 0x2b
#define ARFCN_BCCH 122
#define BCC 0x6
/*-----*\
| GSM 11.10
| 10 Generic call set up procedure
| 10.1 Generic call setup-up procedure for mobile terminating speech calls
| 10.1.2 Definition of system information messages
/*-----*/
/*-----*\
| Information Elements
/*-----*/
/*-----*\
| BCCH Frequency list:
| Indicates seven surrounding cells on any ARFCN of the supported
| band, excluding ARFCNs in or immediately adjacent to those
| specified in section 6.2 (GSM 11.10).
| From GSM 11.10, section 6.2 the following ARFCN are given :
| 10, 14, 17, 18, 22, 24, 26, 30, 31, 34, 38, 42, 45, 46, 50,
| 52, 54, 58, 59, 62, 66, 70, 73, 74, 78, 80, 82, 86, 87, 90,
| 94, 98, 101, 102, 106, 108, 110, 114
| The following 7 cells are chosen :
| 121,117, 76, 48, 12,7,1
| Thus BA is : 01100000 00000800 00008000 00000841
/*-----*/
```

```
IE BEGIN(bcch frequency list)
BF(32,0x01100000,ACT CHECK,ANONYMOUS,"bit 128 thru 97")
BF(32,0x00000800,ACT CHECK,ANONYMOUS,"bit 96 thru 65")
BF(32,0x00008000,ACT CHECK,ANONYMOUS,"bit 64 thru 33")
BF(32,0x00000841,ACT CHECK,ANONYMOUS,"bit 32 thru 1")
IE END(bcch frequency list)
```

```
IE BEGIN(cell channel description)
BF(32,0x00000000,ACT CHECK,ANONYMOUS,"Includes the ")
BF(32,0x00000000,ACT CHECK,ANONYMOUS,"hopping sequence ")
BF(32,0x00000000,ACT CHECK,ANONYMOUS,"ARFCNs, if hopping ")
BF(32,0x00000000,ACT CHECK,ANONYMOUS,"is used. ")
IE END(cell channel description)
```

```
IE BEGIN(cell identity)
BF(16,0x0001,ACT CHECK,ANONYMOUS,"CI VALUE 0001 hex (not relevant)")
IE END(cell identity)
```

```
IE_BEGIN(channel_description)
BF( 5,M5(0,0,1,0,1),ACT_NOP, channel_type,"SDCCH/SACCH 4(1) ")
BF( 3, 0,ACT_NOP, time_slot_number,"zero ")
BF( 3, BCC,ACT_NOP,training_sequence_code,"same as BCCH ")
BF( 1, 0,ACT_NOP, hopping,"No ")
BF( 2, 0,ACT_NOP, spare,SILENT )
BF(10, ARFCN_BCCH,ACT_NOP, arfcn,"ARFCN of the BCCH")
IE_END(channel_description)
IE_BEGIN(mobile_allocation)
BF(8,0,ACT_NOP,length,"length 0 due to hopping disabled")
IE_END(mobile_allocation)
```

IE BEGIN(cell options)

BF(1,0,ACT CHECK,ANONYMOUS , "spare")
BF(1,0,ACT CHECK,pwrc , "power control not set")
BF(2,2,ACT CHECK,dtx , "MS must not use DTX")
BF(4,1,ACT CHECK,radio link time out , "8")
IE END(cell options)

IE BEGIN(cell selection parameter)

BF(3, 0,ACT CHECK,cell reselect hysteresis, "0 dB")
BF(5, 0,ACT CHECK,ms txpwr max cch , "Max. output power of MS")
BF(1, 0,ACT CHECK,acs , "no additional cell params")
BF(1, 0,ACT CHECK,neci , "New estab. cause not supp.")
BF(6,-90+111,ACT NOP,rxlev access min , "-90 dBm")
IE END(cell selection parameter)

IE BEGIN(control channel description)

BF(1,0,ACT CHECK,ANONYMOUS , "spare")
BF(1,0,ACT CHECK,att , "MS shall not apply (not relevant)")
BF(3,0,ACT CHECK,bs_ag blks res, "0 blocks reserved (not relevant)")
BF(3,1,ACT CHECK,ccch conf , "Combined CCCH/SDCCH (not relevant)")
BF(5,0,ACT CHECK,ANONYMOUS , "spare")
BF(3,3,ACT CHECK,bs_pa mfrms , "5 multiframe (not relevant)")
BF(8,0,ACT CHECK,t3212 , "Infinite")
IE END(control channel description)

IE BEGIN(I2 pseudo length 12)

BF(6,12,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 0,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 1,ACT CHECK,ANONYMOUS,SILENT)
IE END(I2 pseudo length 12)

IE BEGIN(I2 pseudo length 18)

BF(6,18,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 0,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 1,ACT CHECK,ANONYMOUS,SILENT)
IE END(I2 pseudo length 18)

IE BEGIN(I2 pseudo length 21)

BF(6,21,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 0,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 1,ACT CHECK,ANONYMOUS,SILENT)
IE END(I2 pseudo length 21)

IE BEGIN(I2 pseudo length 22)

BF(6,22,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 0,ACT CHECK,ANONYMOUS,SILENT)
BF(1, 1,ACT CHECK,ANONYMOUS,SILENT)
IE END(I2 pseudo length 22)

#define MCC 0x262 /* 262 decimal (not relevant) */
#define MNC 1 /* 01 decimal (not relevant) */
#define LAC 0x0001 /* 0001 hex (not relevant) */

IE BEGIN(location area identification)

BF(4,6 ,ACT CHECK,mcc dig 2,"digit 2 of mobile country code")
BF(4,2 ,ACT CHECK,mcc dig 1,"digit 1 of mobile country code")
BF(4, 0xF,ACT CHECK,ANONYMOUS , "end of MCC")
BF(4,2 ,ACT CHECK,mcc dig 3,"digit 3 of mobile country code")
BF(4,1 ,ACT CHECK,mnc dig 2,"digit 2 of mobile network code")

BF(4,0 ,ACT CHECK,mnc dig 1,"digit 1 of mobile network code")
BF(16, LAC,ACT CHECK,lac , "Location area code ")
IE END(location area identification)

IE BEGIN(ncc permitted)
BF(8,0xFF,ACT CHECK,ncc permit,"e.g. all NCCs permitted")
IE END(ncc permitted)

IE BEGIN(rach control parameter)
BF(2,0,ACT CHECK,max retrans,"Any Value ")
BF(4,0,ACT CHECK,tx integer,"Any Value ")
BF(1,0,ACT CHECK,cell bar access , "Not barred ")
BF(1,1,ACT CHECK,call re establishment , "Not Allowed")
BF(5,0,ACT CHECK,access control class 15 11, "None Barred")
BF(1,0,ACT CHECK,emergency call , "Allowed ")
BF(10,0,ACT CHECK,access control class 09 00, "None Barred")
IE END(rach control parameter)

IE BEGIN(rr management protocol discriminator)
BF(4, 6,ACT CHECK,ANONYMOUS,SILENT)
IE END(rr management protocol discriminator)

IE BEGIN(skip indicator)
BF(4, 0,ACT CHECK,ANONYMOUS,SILENT)
IE END(skip indicator)

IE BEGIN(si 1 rest octets)
BF(8,REST OCTET,ACT CHECK,ANONYMOUS,"Spare Octets")
IE END(si 1 rest octets)

IE BEGIN(si 3 rest octets) /* optionally contains cell (re)select params */
BF(1, 0, ACT CHECK,p1,"C2 parameters not present")
BF(7, REST OCTET & 0x7F, ACT CHECK,ANONYMOUS,SILENT)
BF(8, REST OCTET ,ACT CHECK,ANONYMOUS,SILENT)
BF(8, REST OCTET ,ACT CHECK,ANONYMOUS,SILENT)
BF(8, REST OCTET ,ACT CHECK,ANONYMOUS,SILENT)
IE END(si 3 rest octets)

IE BEGIN(si 4 rest octets) /* optionally contains cell (re)select params */
BF(1, 0, ACT CHECK, p1 , "C2 parameters not present")
BF(7,REST OCTET & 0x7F,ACT CHECK,ANONYMOUS,SILENT)
BF(8,REST OCTET, ACT CHECK,ANONYMOUS,SILENT)
BF(8,REST OCTET, ACT CHECK,ANONYMOUS,SILENT)
BF(8,REST OCTET, ACT CHECK,ANONYMOUS,SILENT)
BF(8,REST OCTET, ACT NOP,ANONYMOUS,SILENT)
IE END(si 4 rest octets)

IE_BEGIN(iei_64)
BF(8,0x64,ACT_CHECK,ANONYMOUS,SILENT)
IE_END(iei_64)

IE_BEGIN(iei_72)
BF(8,0x72,ACT_CHECK,ANONYMOUS,SILENT)
IE_END(iei_72)

IE BEGIN(cp ack message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 4, ACT CHECK, ANONYMOUS, SILENT)
IE END(cp ack message type)

IE BEGIN(cp data message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 1, ACT CHECK, ANONYMOUS, SILENT)
IE END(cp data message type)

IE BEGIN(cp error message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 32, ACT CHECK, ANONYMOUS, SILENT)
IE END(cp error message type)

IE BEGIN(system information type 1 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x19, ACT CHECK, ANONYMOUS, SILENT)
IE END(system information type 1 message type)

IE BEGIN(system information type 2 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x1A, ACT CHECK, ANONYMOUS, SILENT)
IE END(system information type 2 message type)

IE BEGIN(system information type 3 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x1B, ACT CHECK, ANONYMOUS, SILENT)
IE END(system information type 3 message type)

IE BEGIN(system information type 4 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x1C, ACT CHECK, ANONYMOUS, SILENT)
IE END(system information type 4 message type)

IE BEGIN(system information type 5 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x1D, ACT CHECK, ANONYMOUS, SILENT)
IE END(system information type 5 message type)

IE BEGIN(system information type 6 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x1E, ACT CHECK, ANONYMOUS, SILENT)
IE END(system information type 6 message type)

IE BEGIN(paging request type 1 message type)
BF(1, 0, ACT CHECK, ANONYMOUS, SILENT)
BF(1, 0, ACT NOP, ANONYMOUS, SILENT)
BF(6, 0x21, ACT CHECK, ANONYMOUS, SILENT)

IE_END(paging request type 1 message type)

IE_BEGIN(immediate assignment message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x3F, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(immediate assignment message type)

IE_BEGIN(paging response message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x27, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(paging response message type)

IE_BEGIN(ciphering mode command message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x35, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(ciphering mode command message type)

IE_BEGIN(ciphering mode complete message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x32, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(ciphering mode complete message type)

IE_BEGIN(channel mode modify message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x10, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(channel mode modify message type)

IE_BEGIN(channel mode modify acknowledge message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x17, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(channel mode modify acknowledge message type)

IE_BEGIN(assignment command message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x2E, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(assignment command message type)

IE_BEGIN(assignment complete message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x29, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(assignment complete message type)

IE_BEGIN(cm service request message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x24, ACT_CHECK, ANONYMOUS, SILENT)

IE_END(cm service request message type)

IE_BEGIN(cm service abort message type)

BF(1, 0, ACT_CHECK, ANONYMOUS, SILENT)

BF(1, 0, ACT_NOP, ANONYMOUS, SILENT)

BF(6, 0x23,ACT_NOP,ANONYMOUS,SILENT)
IE_END(cm_service_abort_message_type)

IE_BEGIN(cm_service_reject_message_type)
BF(1, 0, ACT_CHECK,ANONYMOUS,SILENT)
BF(1, 0, ACT_NOP,ANONYMOUS,SILENT)
BF(6, 0x22,ACT_CHECK,ANONYMOUS,SILENT)
IE_END(cm_service_reject_message_type)

IE_BEGIN(cm_service_accept_message_type)
BF(1, 0, ACT_CHECK,ANONYMOUS,SILENT)
BF(1, 0, ACT_NOP,ANONYMOUS,SILENT)
BF(6, 0x21,ACT_CHECK,ANONYMOUS,SILENT)
IE_END(cm_service_accept_message_type)

IE_BEGIN(call_proceeding_message_type)
BF(1, 0, ACT_CHECK,ANONYMOUS,SILENT)
BF(1, 0, ACT_NOP,ANONYMOUS,SILENT)
BF(6, 0x02,ACT_CHECK,ANONYMOUS,SILENT)
IE_END(call_proceeding_message_type)

/*-----*
| Messages
/*-----*

MSG3 BEGIN(system information type 1)
IE(I2 pseudo length 21)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 1 message type)
IE(cell channel description)
IE(rach control parameter)
IE(si 1 rest octets)
MSG3 END(system information type 1)

MSG3 BEGIN(system information type 2)
IE(I2 pseudo length 22)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 2 message type)
IE(bcch frequency list)
IE(ncc permitted)
IE(rach control parameter)
MSG3 END(system information type 2)

MSG3 BEGIN(system information type 3)
IE(I2 pseudo length 18)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 3 message type)
IE(cell identity)
IE(location area identification)
IE(control channel description)
IE(cell options)
IE(cell selection parameter)
IE(rach control parameter)
IE(si 3 rest octets)
MSG3 END(system information type 3)

MSG3 BEGIN(system information type 4)

IE(l2 pseudo length 12)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 4 message type)
IE(location area identification)
IE(cell selection parameter)
IE(rach control parameter)
IE(si 4 rest octets)
MSG3 END(system information type 4)

MSG3 BEGIN(system information type 4 CB)
IE(l2 pseudo length 12)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 4 message type)
IE(location area identification)
IE(cell selection parameter)
IE(rach control parameter)
IE(iei 64)
IE(channel description)
IE(iei 72)
IE(mobile allocation)
IE(si 4 rest octets)
MSG3 END(system information type 4 CB)

MSG3 BEGIN(system information type 5)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 5 message type)
IE(bcch frequency list)
MSG3 END(system information type 5)

MSG3 BEGIN(system information type 6)
IE(skip indicator)
IE(rr management protocol discriminator)
IE(system information type 6 message type)
IE(cell identity)
IE(location area identification)
IE(cell options)
IE(ncc permitted)
MSG3 END(system information type 6)

```
/*-----*/
| GSM 11.10
| 10 Generic call set up procedure
| 10.1 Generic call setup-up procedure for mobile terminating speech calls
| 10.1.4 Specific message contents
/*-----*/
/*-----*/
| Information Elements
/*-----*/
#define NCC      0x5
#define BSIC    ((NCC<<3)|(BCC))
#define RFN     0
IE_BEGIN(authentication_parameter_rand)
  BF(32,0x80000000,ACT_NOP,rand_127_096,SILENT)
  BF(32,0x00000012,ACT_NOP,rand_095_064,SILENT)
  BF(32,0x34000000,ACT_NOP,rand_063_032,SILENT)
  BF(32,0x0000000F,ACT_NOP,rand_031_000,SILENT)
IE_END(authentication_parameter_rand)
```

IE_BEGIN(authentication_parameter_sres)
BF(32,0x0000000F,ACT_NOP,sres_031_000,SILENT)
IE_END(authentication_parameter_sres)
IE_BEGIN(bearer_capability)
BF(8,1,ACT_CHECK, length, SILENT)
BF(1,1,ACT_CHECK, ext3, SILENT)
BF(2,1,ACT_CHECK, radio_channel_requirement,SILENT)
BF(1,0,ACT_CHECK, coding_standard, SILENT)
BF(1,0,ACT_CHECK, transfer_mode, SILENT)
BF(3,0,ACT_CHECK, info_transfer_capability, SILENT)
IE_END(bearer_capability)

IE_BEGIN(bearer_capability_data)
BF(8,7,ACT_CHECK, length, SILENT)
BF(1,1,ACT_CHECK, ext3, SILENT)
BF(2,1,ACT_CHECK, radio_channel_requirement,SILENT)
BF(1,0,ACT_CHECK, coding_standard, SILENT)
BF(1,0,ACT_CHECK, transfer_mode, SILENT)
BF(3,2,ACT_CHECK, info_transfer_capability, SILENT)
BF(1,1,ACT_CHECK, ext4, SILENT)
BF(1,1,ACT_CHECK, compression, SILENT)
BF(2,0,ACT_CHECK, structure, SILENT)
BF(1,1,ACT_CHECK, duplex_mode, SILENT)
BF(1,0,ACT_CHECK, configuration, SILENT)
BF(1,0,ACT_CHECK, NIRR, SILENT)
BF(1,0,ACT_CHECK, establishment, SILENT)
BF(1,1,ACT_CHECK, ext5, SILENT)
BF(2,1,ACT_CHECK, access_identifier, SILENT)
BF(2,0,ACT_CHECK, rate_adaption, SILENT)
BF(3,1,ACT_CHECK, access_protocol, SILENT)
BF(1,0,ACT_CHECK, ext6, SILENT)
BF(2,1,ACT_CHECK, layer_1, SILENT)
BF(4,0,ACT_CHECK, default_layer_1, SILENT)
BF(1,1,ACT_CHECK, sync_async, SILENT)
BF(1,0,ACT_CHECK, ext6a, SILENT)
BF(1,0,ACT_CHECK, stop_bits, SILENT)
BF(1,0,ACT_CHECK, negotiation, SILENT)
BF(1,1,ACT_CHECK, data_bits, SILENT)
BF(4,3,ACT_CHECK, user_rate, SILENT)
BF(1,0,ACT_CHECK, ext6b, SILENT)
BF(2,3,ACT_CHECK, intermediate_rate, SILENT)
BF(1,0,ACT_CHECK, NIC_TX, SILENT)
BF(1,0,ACT_CHECK, NIC_RX, SILENT)
BF(3,3,ACT_CHECK, parity, SILENT)
BF(1,1,ACT_CHECK, ext6c, SILENT)
BF(2,1,ACT_CHECK, connection_element, SILENT)
BF(5,3,ACT_CHECK, modem_type, SILENT)
IE_END(bearer_capability_data)

IE_BEGIN(call_control_protocol_discriminator)
BF(4, 3,ACT_NOP,ANONYMOUS,SILENT)
IE_END(call_control_protocol_discriminator)

IE_BEGIN(sms_protocol_discriminator)
BF(4, 9,ACT_NOP,ANONYMOUS,SILENT)
IE_END(sms_protocol_discriminator)

IE_BEGIN(called_party_bcd_number)
BF(8,7,ACT_NOP,length , "length of IE")

```
BF( 1,1,ACT_NOP,ext           ,"Extension Bit")
BF( 3,0,ACT_NOP,type_of_number ,"Unknow")
BF( 4,1,ACT_NOP,numbering_plan ,"ISDN/telephony")
BF( 4,3,ACT_NOP,digit_2       ,"Digit 2")
BF( 4,0,ACT_NOP,digit_1       ,"Digit 1")
BF( 4,3,ACT_NOP,digit_4       ,"Digit 4")
BF( 4,0,ACT_NOP,digit_3       ,"Digit 3")
BF( 4,0,ACT_NOP,digit_6       ,"Digit 6")
BF( 4,9,ACT_NOP,digit_5       ,"Digit 5")
BF( 4,4,ACT_NOP,digit_8       ,"Digit 8")
BF( 4,9,ACT_NOP,digit_7       ,"Digit 7")
BF( 4,1,ACT_NOP,digit_10      ,"Digit 10")
BF( 4,1,ACT_NOP,digit_9       ,"Digit 9")
BF( 4,7,ACT_NOP,digit_12      ,"Digit 12")
BF( 4,0xF,ACT_NOP,digit_11    ,"Digit 11")
IE_END(called_party_bcd_number)

IE_BEGIN(channels_needed_for_mobiles_1_and_2)
  BF(2,0,ACT_NOP,second_channel,"spare, any channel")
  BF(2,0,ACT_NOP, first_channel,"spare, any channel")
IE_END(channels_needed_for_mobiles_1_and_2)
IE_BEGIN(cc_capabilities)
  BF(8,1,ACT_CHECK, length, SILENT)
  BF(8,1,ACT_CHECK, dtmf_support, SILENT)
IE_END(cc_capabilities)

IE_BEGIN(channel_description_tch)
  BF( 5,M5(0,0,0,0,1),ACT_CHECK, channel_type,"TCH Full Rate")
  BF( 3, 0,ACT_CHECK, time_slot_number,"zero ")
  BF( 3, BCC,ACT_CHECK, tsc ,"same as BCCH ")
  BF( 1, 0,ACT_CHECK, hopping,"No ")
  BF( 2, 0,ACT_CHECK, spare,SILENT )
  BF(10, ARFCN_BCCH,ACT_CHECK, arfcn,"ARFCN of the BCCH ")
IE_END(channel_description_tch)

IE_BEGIN(channel_mode_speech)
  BF( 8, 1,ACT_CHECK, mode,"Speech full rate")
IE_END(channel_mode_speech)

IE_BEGIN(cm_service_type_moc)
  BF(4,M4(0,0,0,1),ACT_NOP,service_type,"mobile originated call")
IE_END(cm_service_type_moc)

IE_BEGIN(ciphering_key_sequence_number)
  BF(1, 0,ACT_NOP, spare,SILENT)
  BF(3,M3(1,1,1),ACT_NOP,key_sequence,"no key is available (MS->BS)")
IE_END(ciphering_key_sequence_number)
IE_BEGIN(ciphering_key_sequence_number_2)
  BF(1, 0,ACT_NOP, spare,SILENT)
  BF(3,M3(1,0,1),ACT_NOP,key_sequence,"sent BS->MS")
IE_END(ciphering_key_sequence_number_2)
IE_BEGIN(ciphering_mode_setting)
  BF(3,M3(0,0,0),ACT_NOP,algorithm_identifier,"A5/1 ")
  BF(1, 1,ACT_NOP, start_ciphering,"Start ciphering")
IE_END(ciphering_mode_setting)
IE_BEGIN(cipher_response)
  BF(3,0,ACT_NOP, spare,SILENT )
  BF(1,0,ACT_NOP,cipher_response,"IMEISV shall not be included")
IE_END(cipher_response)
```

```
IE_BEGIN(description_of_the_first_channel_after_time)
BF( 5,M5(0,0,0,0,1),ACT_NOP, channel_type,"TCH ")
BF( 3, 3,ACT_NOP, time_slot_number,"three ")
BF( 3, BCC,ACT_NOP,training_sequence_code,"same as BCCH ")
BF( 1, 0,ACT_NOP, hopping,"No ")
BF( 2, 0,ACT_NOP, spare,SILENT )
BF(10, ARFCN_BCCH,ACT_NOP, arfcn,"ARFCN of the BCCH")
```

```
IE_END(description_of_the_first_channel_after_time)
IE_BEGIN(ia_rest_octets) /* maximum length (11), no hop, no start time */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 0 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 1 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 2 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 3 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 4 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 5 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 6 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 7 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 8 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 9 */
BF(8,REST_OCTET,ACT_NOP,ANONYMOUS,SILENT) /* 10 */
```

```
IE_END(ia_rest_octets)
```

```
IE_BEGIN(iei_08)
BF(8,0x08,ACT_NOP,ANONYMOUS,SILENT)
```

```
IE_END(iei_08)
```

```
IE_BEGIN(iei_1E)
BF(8,0x1E,ACT_NOP,ANONYMOUS,SILENT)
```

```
IE_END(iei_1E)
```

```
IE_BEGIN(iei_63)
BF(8,0x63,ACT_NOP,ANONYMOUS,SILENT)
```

```
IE_END(iei_63)
```

```
IE_BEGIN(iei_5E)
BF(8,0x5E,ACT_NOP,ANONYMOUS,SILENT)
```

```
IE_END(iei_5E)
```

```
IE_BEGIN(iei_15)
BF(8,0x15,ACT_CHECK,ANONYMOUS,SILENT)
```

```
IE_END(iei_15)
```

```
IE_BEGIN(iei_04)
BF(8,0x04,ACT_CHECK,ANONYMOUS,SILENT)
```

```
IE_END(iei_04)
```

```
IE_BEGIN(iei_34)
BF(8,0x34,ACT_NOP,ANONYMOUS,SILENT)
```

```
IE_END(iei_34)
```

```
IE_BEGIN(iei_2C)
BF(8,0x2C,ACT_NOP,ANONYMOUS,SILENT)
```

```
IE_END(iei_2C)
```

```
IE_BEGIN(keypad_facility_0)
BF(8, 0x30,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_0)
```

```
IE_BEGIN(keypad_facility_1)
BF(8, 0x31,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_1)
```

```
IE_BEGIN(keypad_facility_2)
BF(8, 0x32,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_2)
```

```
IE_BEGIN(keypad_facility_3)
```

BF(8, 0x33,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_3)

IE_BEGIN(keypad_facility_4)
BF(8, 0x34,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_4)

IE_BEGIN(keypad_facility_5)
BF(8, 0x35,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_5)

IE_BEGIN(keypad_facility_6)
BF(8, 0x36,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_6)

IE_BEGIN(keypad_facility_7)
BF(8, 0x37,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_7)

IE_BEGIN(keypad_facility_8)
BF(8, 0x38,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_8)

IE_BEGIN(keypad_facility_9)
BF(8, 0x39,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_9)

IE_BEGIN(keypad_facility_A)
BF(8, 0x41,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_A)

IE_BEGIN(keypad_facility_B)
BF(8, 0x42,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_B)

IE_BEGIN(keypad_facility_C)
BF(8, 0x43,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_C)

IE_BEGIN(keypad_facility_D)
BF(8, 0x44,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_D)

IE_BEGIN(keypad_facility_star)
BF(8, 0x2A,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_star)

IE_BEGIN(keypad_facility_hash)
BF(8, 0x23,ACT_NOP,ANONYMOUS,SILENT)
IE_END(keypad_facility_hash)

IE_BEGIN(I2_pseudo_length_11) /* pag req type 1 with TMSI (one mobile)*/
BF(8, 0,ACT_NOP,ANONYMOUS,SILENT)
IE_END(I2_pseudo_length_11)
IE_BEGIN(mobile_identity) /* has 8 octets */
BF(8, 7,ACT_CHECK, length,"seven octets to come")
BF(4, 2,ACT_CHECK, digit_1,SILENT)
BF(1, 1,ACT_CHECK,odd_even,SILENT)
BF(3, M3(0,0,1),ACT_CHECK, type,"IMSI")

```

BF(4, 2,ACT_CHECK, digit_3,SILENT)
BF(4, 6,ACT_CHECK, digit_2,SILENT)
BF(4, 1,ACT_CHECK, digit_4,SILENT)
BF(4, 0,ACT_CHECK, digit_3,SILENT)
BF(4, 7,ACT_CHECK, digit_6,SILENT)
BF(4, 4,ACT_CHECK, digit_5,SILENT)
BF(4, 1,ACT_CHECK, digit_8,SILENT)
BF(4, 1,ACT_CHECK, digit_7,SILENT)
BF(4, 9,ACT_CHECK, digit_10,SILENT)
BF(4, 4,ACT_CHECK, digit_9,SILENT)
BF(4, 2,ACT_CHECK, digit_12,SILENT)
BF(4, 1,ACT_CHECK, digit_11,SILENT)
IE_END(mobile_identity)
IE_BEGIN(mobility_management_protocol_discriminator)
  BF(4, 5,ACT_NOP, ANONYMOUS,SILENT)
IE_END(mobility_management_protocol_discriminator)

IE_BEGIN(identity_type)
  BF(4,1,ACT_CHECK, type_of_identity,"IMSI")
IE_END(identity_type)

IE_BEGIN(mode_of_the_first_channel)
  BF(8,1,ACT_NOP, mode,"Speech full rate")
IE_END(mode_of_the_first_channel)
IE_BEGIN(ms_classmark)
  BF(8, 3,ACT_NOP, length,SILENT)
  BF(1, 0,ACT_NOP, spare,SILENT)
  BF(2, M2(0,1),ACT_NOP, revision_level,"phase 2 MS")
  BF(1, 0,ACT_NOP, es_ind,"No 'Contr. Early Classmark Send.'")
  BF(1, 0,ACT_NOP, a5_1,"encryption algorithm A5/1 available")
  BF(3,M3(0,0,1),ACT_NOP, rf_power_capability,"class 2")
  BF(1, 0,ACT_NOP, spare2,SILENT)
  BF(1, 0,ACT_NOP, ps_capability,"no pseudo-synch capability")
  BF(2, M2(0,0),ACT_NOP, ss_screening_indicator,"default phase 1")
  BF(1, 0,ACT_NOP, sm_capability,"no point to point SMS")
  BF(1, 0,ACT_NOP, vbs,"no VBS cap. or no notific. wanted")
  BF(1, 0,ACT_NOP, vgcs,"no VGCS cap. or no notific. wanted")
  BF(1, 0,ACT_NOP, frequency_capability,"no extention band G1")
  BF(1, 0,ACT_NOP, classmark_3,"no add. MS cap. information")
  BF(5, 0,ACT_NOP, spare3,SILENT)
  BF(1, 0,ACT_NOP, a5_3,"A5/3 not available")
  BF(1, 0,ACT_NOP, a5_2,"A5/2 not available")
IE_END(ms_classmark)
IE_BEGIN(p1_rest_octets)
  /* pag. req. type1 : 22 - 11 (L2 pseud. len) = 11 bytes */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 0 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 1 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 2 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 3 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 4 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 5 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 6 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 7 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 8 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 9 */
  BF(8,REST_OCTET,ACT_NOP, ANONYMOUS,SILENT) /* 10 */
IE_END(p1_rest_octets)
IE_BEGIN(page_mode)
  BF(2,0,ACT_NOP,spare,"two spare bits ")

```

```
BF(2,0,ACT_NOP,pm , "Normal Paging")
IE_END(page_mode)
IE_BEGIN(power_command)
  BF(8,10,ACT_NOP,power,SILENT)
IE_END(power_command)
IE_BEGIN(rach)
  BF(3,M3(1,0,0),ACT_CHECK ,establishment_cause,"paging ind. any chan")
  BF(5,M5(1,1,1,1,1),ACT_NOP, random_reference,"ignore Random Ref.")
IE_END(rach)
IE_BEGIN(rach_moc)
  BF(3,M3(1,1,1),ACT_CHECK ,establishment_cause,"MOC & TCH/F")
  BF(5,M5(1,1,1,1,1),ACT_NOP, random_reference,"ignore Random Ref.")
IE_END(rach_moc)
IE_BEGIN(rach_111)
  BF(3,M3(1,1,1),ACT_CHECK ,establishment_cause,"SDCCH Assignment")
  BF(5,M5(1,1,1,1,1),ACT_NOP, random_reference,"ignore Random Ref.")
IE_END(rach_111)
IE_BEGIN(request_reference)
  BF(3, M3(1,0,0),ACT_NOP,random_access_info,"As in CHAN REQ")
  BF(5,M5(1,1,1,1,1),ACT_NOP, random_reference,SILENT)
  BF(5, 0,ACT_NOP, t1,SILENT)
  BF(6, 0,ACT_NOP, t3,SILENT)
  BF(5, 0,ACT_NOP, t2,SILENT)
IE_END(request_reference)

IE_BEGIN(rr_cause)
  BF(8,0,ACT_CHECK,rr_cause,"normal event")
IE_END(rr_cause)

IE_BEGIN(reject_cause)
  BF(8,0x20,ACT_CHECK,reject_cause,"service not available")
IE_END(reject_cause)

IE_BEGIN(progress_indicator_32)
  BF(8, 2,ACT_CHECK, length,"two octets")
  BF(1, 1,ACT_CHECK, ext_1,SILENT)
  BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
  BF(1, 0,ACT_SHOW, spare,SILENT)
  BF(4, 0,ACT_CHECK, location,"User")
  BF(1, 1,ACT_CHECK, ext_2,SILENT)
  BF(7, 32,ACT_CHECK, progress_description,"call is end-to-end PLMN/SDN")
IE_END(progress_indicator_32)

IE_BEGIN(progress_indicator_8)
  BF(8, 2,ACT_CHECK, length,"two octets")
  BF(1, 1,ACT_CHECK, ext_1,SILENT)
  BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
  BF(1, 0,ACT_SHOW, spare,SILENT)
  BF(4, 0,ACT_CHECK, location,"User")
  BF(1, 1,ACT_CHECK, ext_2,SILENT)
  BF(7, 8,ACT_CHECK, progress_description,"in band tones available")
IE_END(progress_indicator_8)

IE_BEGIN(cause_03)
  BF(8, 2,ACT_CHECK, length,"two octets")
  BF(1, 1,ACT_CHECK, ext_1,SILENT)
  BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
  BF(1, 0,ACT_SHOW, spare,SILENT)
  BF(4, 0,ACT_CHECK, location,"User")
```

BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 3,ACT_CHECK, cause,"no route to destination")
IE_END(cause_03)

IE_BEGIN(cause_17)
BF(8, 2,ACT_CHECK, length,"two octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 17,ACT_CHECK, cause,"user busy")
IE_END(cause_17)

IE_BEGIN(cause_31)
BF(8, 2,ACT_CHECK, length,"two octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 31,ACT_CHECK, cause,"normal, unspecified")
IE_END(cause_31)

IE_BEGIN(cause_81)
BF(8, 2,ACT_CHECK, length,"two octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 81,ACT_CHECK, cause,"invalid transaction identifier")
IE_END(cause_81)

IE_BEGIN(cause_88)
BF(8, 2,ACT_CHECK, length,"two octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 88,ACT_CHECK, cause,"incompatible destination")
IE_END(cause_88)

IE_BEGIN(cause_102_303)
BF(8, 5,ACT_CHECK, length,"five octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 102,ACT_CHECK, cause,"recovery on timer expiry")
BF(8, 0x33,ACT_CHECK, diag_0,"3")
BF(8, 0x30,ACT_CHECK, diag_1,"0")
BF(8, 0x33,ACT_CHECK, diag_2,"3")
IE_END(cause_102_303)

IE_BEGIN(cause_102_310)

BF(8, 5,ACT_CHECK, length,"five octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 102,ACT_CHECK, cause,"recovery on timer expiry")
BF(8, 0x33,ACT_CHECK, diag_0,"3")
BF(8, 0x31,ACT_CHECK, diag_1,"1")
BF(8, 0x30,ACT_CHECK, diag_2,"0")
IE_END(cause_102_310)

IE_BEGIN(cause_102_313)
BF(8, 5,ACT_CHECK, length,"five octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 102,ACT_CHECK, cause,"recovery on timer expiry")
BF(8, 0x33,ACT_CHECK, diag_0,"3")
BF(8, 0x31,ACT_CHECK, diag_1,"1")
BF(8, 0x33,ACT_CHECK, diag_2,"3")
IE_END(cause_102_313)

IE_BEGIN(cause_97_20)
BF(8, 3,ACT_CHECK, length,"three octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 97,ACT_CHECK, cause,"message not implemented")
BF(8, 0x20,ACT_CHECK, diag_0,"message type 0x20")
IE_END(cause_97_20)

IE_BEGIN(cause_16)
BF(8, 2,ACT_CHECK, length,"two octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 16,ACT_CHECK, cause,"normal clearing")
IE_END(cause_16)

IE_BEGIN(cause_30)
BF(8, 2,ACT_CHECK, length,"two octets")
BF(1, 1,ACT_CHECK, ext_1,SILENT)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(1, 0,ACT_SHOW, spare,SILENT)
BF(4, 0,ACT_CHECK, location,"User")
BF(1, 1,ACT_CHECK, ext_2,SILENT)
BF(7, 30,ACT_CHECK, cause,"response to enquiry")
IE_END(cause_30)

IE_BEGIN(call_state_1)
BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 1,ACT_CHECK, call_state_value,"U1 Call Initiated")

IE_END(call_state_1)

IE_BEGIN(call_state_3)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 3,ACT_CHECK, call_state_value,"U3 Call Proceeding")

IE_END(call_state_3)

IE_BEGIN(call_state_4)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 4,ACT_CHECK, call_state_value,"U4 Call Delivered")

IE_END(call_state_4)

IE_BEGIN(call_state_7)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 7,ACT_CHECK, call_state_value,"U7 Call Receiving")

IE_END(call_state_7)

IE_BEGIN(call_state_8)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 8,ACT_CHECK, call_state_value,"U8 Connect Request")

IE_END(call_state_8)

IE_BEGIN(call_state_9)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 9,ACT_CHECK, call_state_value,"U9 Call Confirmed")

IE_END(call_state_9)

IE_BEGIN(call_state_10)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 10,ACT_CHECK, call_state_value,"U10 Call Active")

IE_END(call_state_10)

IE_BEGIN(call_state_11)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 11,ACT_CHECK, call_state_value,"U11 Disconnect Request")

IE_END(call_state_11)

IE_BEGIN(call_state_12)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 12,ACT_CHECK, call_state_value,"U12 Disconnect Ind")

IE_END(call_state_12)

IE_BEGIN(call_state_19)

BF(2, 3,ACT_CHECK, coding_standard, "GSM Standard")
BF(6, 19,ACT_CHECK, call_state_value,"U19 Release Request")

IE_END(call_state_19)

IE_BEGIN(signal_call_waiting)

BF(8,M8(0,0,0,0,1,1),ACT_NOP,signal_value,"(Any non-res. value)")

IE_END(signal_call_waiting)

IE_BEGIN(spare_half_octet)

BF(4, 0,ACT_NOP,ANONYMOUS,SILENT)

IE_END(spare_half_octet)

IE_BEGIN(timing_advance)

BF(2,0,ACT_NOP, spare,SILENT)
BF(6,0,ACT_NOP,timing_advance,"0")

IE_END(timing_advance)
IE_BEGIN(transaction_identifier_source)
BF(4,M4(0,0,0,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source)

IE_BEGIN(transaction_identifier_source_1)
BF(4,M4(0,0,0,1),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source_1)

IE_BEGIN(transaction_identifier_source_2)
BF(4,M4(0,0,1,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source_2)

IE_BEGIN(transaction_identifier_source_3)
BF(4,M4(0,0,1,1),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source_3)

IE_BEGIN(transaction_identifier_source_4)
BF(4,M4(0,1,0,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source_4)

IE_BEGIN(transaction_identifier_source_5)
BF(4,M4(0,1,0,1),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source_5)

IE_BEGIN(transaction_identifier_source_6)
BF(4,M4(0,1,1,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_source_6)

IE_BEGIN(transaction_identifier_dest)
BF(4,M4(1,0,0,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest)

IE_BEGIN(transaction_identifier_dest_1)
BF(4,M4(1,0,0,1),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest_1)

IE_BEGIN(transaction_identifier_dest_2)
BF(4,M4(1,0,1,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest_2)

IE_BEGIN(transaction_identifier_dest_3)
BF(4,M4(1,0,1,1),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest_3)

IE_BEGIN(transaction_identifier_dest_4)
BF(4,M4(1,1,0,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest_4)

IE_BEGIN(transaction_identifier_dest_5)
BF(4,M4(1,1,0,1),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest_5)

IE_BEGIN(transaction_identifier_dest_6)
BF(4,M4(1,1,1,0),ACT_NOP,ANONYMOUS,SILENT)
IE_END(transaction_identifier_dest_6)

IE_BEGIN(authentication_request_message_type)
BF(8,0x12,ACT_NOP,ANONYMOUS,SILENT)

IE_END(authentication_request_message_type)

IE_BEGIN(identity_request_message_type)
BF(8, 0x18,ACT_NOP,ANONYMOUS,SILENT)
IE_END(identity_request_message_type)

IE_BEGIN(identity_response_message_type)
BF(8, 0x19,ACT_NOP,ANONYMOUS,SILENT)
IE_END(identity_response_message_type)

IE_BEGIN(authentication_response_message_type)
BF(8, 0x14,ACT_NOP,ANONYMOUS,SILENT)
IE_END(authentication_response_message_type)
IE_BEGIN(setup_message_type)
BF(8, 0x05,ACT_NOP,ANONYMOUS,SILENT)
IE_END(setup_message_type)
IE_BEGIN(call_confirmed_message_type)
BF(8, 0x08,ACT_NOP,ANONYMOUS,SILENT)
IE_END(call_confirmed_message_type)
IE_BEGIN(connect_message_type)
BF(8, 0x07,ACT_NOP,ANONYMOUS,SILENT)
IE_END(connect_message_type)
IE_BEGIN(connect_acknowledge_message_type)
BF(8, 0x0F,ACT_NOP,ANONYMOUS,SILENT)
IE_END(connect_acknowledge_message_type)
IE_BEGIN(alerting_message_type)
BF(8, 0x01,ACT_NOP,ANONYMOUS,SILENT)
IE_END(alerting_message_type)

IE_BEGIN(progress_message_type)
BF(8, 0x03,ACT_NOP,ANONYMOUS,SILENT)
IE_END(progress_message_type)

IE_BEGIN(status_message_type)
BF(8, 0x3D,ACT_NOP,ANONYMOUS,SILENT)
IE_END(status_message_type)
IE_BEGIN(status_enquiry_message_type)
BF(8, 0x34,ACT_NOP,ANONYMOUS,SILENT)
IE_END(status_enquiry_message_type)

IE_BEGIN(unknown_message_type)
BF(8, 0x20,ACT_NOP,ANONYMOUS,SILENT)
IE_END(unknown_message_type)

IE_BEGIN(disconnect_message_type)
BF(8, 0x25,ACT_NOP,ANONYMOUS,SILENT)
IE_END(disconnect_message_type)

IE_BEGIN(release_complete_message_type)
BF(8, 0x2A,ACT_NOP,ANONYMOUS,SILENT)
IE_END(release_complete_message_type)

IE_BEGIN(release_message_type)
BF(8, 0x2D,ACT_NOP,ANONYMOUS,SILENT)
IE_END(release_message_type)

IE_BEGIN(channel_release_message_type)
BF(8, 0x0D,ACT_NOP,ANONYMOUS,SILENT)
IE_END(channel_release_message_type)

IE_BEGIN(start_dtmf_message_type)
BF(8, 0x35, ACT_NOP, ANONYMOUS, SILENT)
IE_END(start_dtmf_message_type)

IE_BEGIN(start_dtmf_acknowledge_message_type)
BF(8, 0x36, ACT_NOP, ANONYMOUS, SILENT)
IE_END(start_dtmf_acknowledge_message_type)

IE_BEGIN(stop_dtmf_message_type)
BF(8, 0x31, ACT_NOP, ANONYMOUS, SILENT)
IE_END(stop_dtmf_message_type)

IE_BEGIN(stop_dtmf_acknowledge_message_type)
BF(8, 0x32, ACT_NOP, ANONYMOUS, SILENT)
IE_END(stop_dtmf_acknowledge_message_type)

IE_BEGIN(start_dtmf_reject_message_type)
BF(8, 0x37, ACT_NOP, ANONYMOUS, SILENT)
IE_END(start_dtmf_reject_message_type)

IE BEGIN(cp user data sms submit)
BF(8,0x1d, ACT_CHECK, len, "length of cp user data")
BF(5, 0, ACT_CHECK, spare, "spare bits")
BF(3, 0, ACT_CHECK, mti, "rp data mobile to mobile")
BF(8,0, ACT_CHECK, msg_ref, "message reference")
BF(8, 0, ACT_CHECK, rp_orig_len, "rp originator address")
BF(8, 5, ACT_CHECK, rp_dest_len, "rp destination address")
BF(1, 1, ACT_CHECK, rp_dest_ext, "Extension Bit")
BF(3, 1, ACT_CHECK, rp_dest_type_of_number, "Unknown")
BF(4, 1, ACT_CHECK, rp_dest_numbering_plan, "ISDN/telephony")
BF(4,2, ACT_CHECK, rp_dest_digit_2, "Digit 2")
BF(4,1, ACT_CHECK, rp_dest_digit_1, "Digit 1")
BF(4,4, ACT_CHECK, rp_dest_digit_4, "Digit 4")
BF(4,3, ACT_CHECK, rp_dest_digit_3, "Digit 3")
BF(4,6, ACT_CHECK, rp_dest_digit_6, "Digit 6")
BF(4,5, ACT_CHECK, rp_dest_digit_5, "Digit 5")
BF(4,8, ACT_CHECK, rp_dest_digit_8, "Digit 8")
BF(4,7, ACT_CHECK, rp_dest_digit_7, "Digit 7")
BF(8,19, ACT_CHECK, rp_user_data_len, "rp user data len")
BF(1, 0, ACT_CHECK, tp_rp, "reply path")
BF(1, 0, ACT_CHECK, tp_udhi, "no user data header indicator")
BF(1, 0, ACT_CHECK, tp_srr, "no status report request")
BF(2, 2, ACT_CHECK, tp_vpf, "relative validity period format")
BF(1, 0, ACT_CHECK, tp_rd, "no reject duplicates")
BF(2, 1, ACT_CHECK, tp_mti, "sms submit")
BF(8, 0, ACT_CHECK, tp_msg_ref, "tp message reference")
BF(8, 4, ACT_CHECK, tp_da_len, "tp da with four digits")
BF(1, 1, ACT_CHECK, tp_da_ext, "Extension Bit")
BF(3, 1, ACT_CHECK, tp_da_type_of_number, "Unknown")
BF(4, 1, ACT_CHECK, tp_da_numbering_plan, "ISDN/telephony")
BF(4,6, ACT_CHECK, tp_da_digit_2, "Digit 2")
BF(4,5, ACT_CHECK, tp_da_digit_1, "Digit 1")
BF(4,8, ACT_CHECK, tp_da_digit_4, "Digit 4")
BF(4,7, ACT_CHECK, tp_da_digit_3, "Digit 3")
BF(8, 0, ACT_CHECK, tp_pd, "protocol identifier")
BF(8, 0, ACT_CHECK, tp_dcs, "data coding scheme")
BF(8, 3, ACT_CHECK, tp_vp, "relative time 03")

BF(8, 0x0A, ACT CHECK, tp ud len, "characters in sms")
BF(8, 0xC8, ACT CHECK, tp ud 1, "SMS Halli Hallo")
BF(8, 0x20, ACT CHECK, tp ud 2, SILENT)
BF(8, 0x93, ACT CHECK, tp ud 3, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 4, SILENT)
BF(8, 0x44, ACT CHECK, tp ud 5, SILENT)
BF(8, 0x06, ACT CHECK, tp ud 6, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 7, SILENT)
BF(8, 0xCC, ACT CHECK, tp ud 8, SILENT)
BF(8, 0x27, ACT CHECK, tp ud 9, SILENT)
IE END(cp user data sms submit)

IE BEGIN(cp user data sms deliver)
BF(8,38, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 1, ACT CHECK, mti, "rp data network to mobile")
BF(8, 22, ACT CHECK, msg_ref, "message reference")
BF(8, 7, ACT CHECK, rp_orig_len, "rp originator address")
BF(1, 1, ACT CHECK, rp_orig_ext, "Extension Bit")
BF(3, 0, ACT CHECK, rp_orig_type_of_number, "Unknown")
BF(4, 1, ACT CHECK, rp_orig_numbering_plan, "ISDN/telephony")
BF(4,3, ACT CHECK, rp_orig_digit_2, "Digit 2")
BF(4,0, ACT CHECK, rp_orig_digit_1, "Digit 1")
BF(4,3, ACT CHECK, rp_orig_digit_4, "Digit 4")
BF(4,0, ACT CHECK, rp_orig_digit_3, "Digit 3")
BF(4,0, ACT CHECK, rp_orig_digit_6, "Digit 6")
BF(4,9, ACT CHECK, rp_orig_digit_5, "Digit 5")
BF(4,4, ACT CHECK, rp_orig_digit_8, "Digit 8")
BF(4,9, ACT CHECK, rp_orig_digit_7, "Digit 7")
BF(4,1, ACT CHECK, rp_orig_digit_10, "Digit 10")
BF(4,1, ACT CHECK, rp_orig_digit_9, "Digit 9")
BF(4,0xF, ACT CHECK, rp_orig_digit_12, "Digit 12")
BF(4,0x7, ACT CHECK, rp_orig_digit_11, "Digit 11")
BF(8, 0, ACT CHECK, rp_dest_len, "rp destination address")
BF(8, 26, ACT CHECK, rp_user_data_len, "rp user data len")
BF(1, 1, ACT CHECK, tp_rp, "reply path")
BF(1, 0, ACT CHECK, tp_udhi, "no user data header indicator")
BF(1, 0, ACT CHECK, tp_sri, "no status report indication")
BF(2, 0, ACT CHECK, tp_spare, "spare bits")
BF(1, 0, ACT CHECK, tp_mms, "no more messages to send")
BF(2, 0, ACT CHECK, tp_mti, "sms deliver")
BF(8, 8, ACT CHECK, tp_ua_len, "tp ua with eight digits")
BF(1, 1, ACT CHECK, tp_ua_ext, "Extension Bit")
BF(3, 0, ACT CHECK, tp_ua_type_of_number, "Unknown")
BF(4, 1, ACT CHECK, tp_ua_numbering_plan, "ISDN/telephony")
BF(4,3, ACT CHECK, tp_ua_digit_2, "Digit 2")
BF(4,0, ACT CHECK, tp_ua_digit_1, "Digit 1")
BF(4,3, ACT CHECK, tp_ua_digit_4, "Digit 4")
BF(4,0, ACT CHECK, tp_ua_digit_3, "Digit 3")
BF(4,0, ACT CHECK, tp_ua_digit_6, "Digit 6")
BF(4,9, ACT CHECK, tp_ua_digit_5, "Digit 5")
BF(4,4, ACT CHECK, tp_ua_digit_8, "Digit 8")
BF(4,9, ACT CHECK, tp_ua_digit_7, "Digit 7")
BF(8,0x1F, ACT CHECK, tp_pd, "protocol identifier")
BF(8, 0xF2, ACT CHECK, tp_dcs, "data coding scheme")
BF(8, 0x89, ACT CHECK, tp_scts_year, "year 98")
BF(8, 0x40, ACT CHECK, tp_scts_month, "month 04")
BF(8, 0x61, ACT CHECK, tp_scts_day, "day 16")
BF(8, 0x01, ACT CHECK, tp_scts_hour, "hour 10")

BF(8, 0x51, ACT CHECK, tp scts minute, "minute 15")
BF(8, 0x43, ACT CHECK, tp scts second, "second 34")
BF(8, 0x00, ACT CHECK, tp scts timezone, "zone 00")
BF(8, 0x0A, ACT CHECK, tp ud len, "characters in sms")
BF(8, 0xC8, ACT CHECK, tp ud 1, "SMS Halli Hallo")
BF(8, 0x20, ACT CHECK, tp ud 2, SILENT)
BF(8, 0x93, ACT CHECK, tp ud 3, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 4, SILENT)
BF(8, 0x44, ACT CHECK, tp ud 5, SILENT)
BF(8, 0x06, ACT CHECK, tp ud 6, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 7, SILENT)
BF(8, 0xCC, ACT CHECK, tp ud 8, SILENT)
BF(8, 0x27, ACT CHECK, tp ud 9, SILENT)
IE END(cp user data sms deliver)

IE BEGIN(cp user data sms deliver class1)

BF(8,38, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 1, ACT CHECK, mti, "rp data network to mobile")
BF(8, 22, ACT CHECK, msg ref, "message reference")
BF(8, 7, ACT CHECK, rp orig len, "rp originator address")
BF(1, 1, ACT CHECK, rp orig ext, "Extension Bit")
BF(3, 0, ACT CHECK, rp orig type of number, "Unknown")
BF(4, 1, ACT CHECK, rp orig numbering plan, "ISDN/telephony")
BF(4,3, ACT CHECK, rp orig digit 2, "Digit 2")
BF(4,0, ACT CHECK, rp orig digit 1, "Digit 1")
BF(4,3, ACT CHECK, rp orig digit 4, "Digit 4")
BF(4,0, ACT CHECK, rp orig digit 3, "Digit 3")
BF(4,0, ACT CHECK, rp orig digit 6, "Digit 6")
BF(4,9, ACT CHECK, rp orig digit 5, "Digit 5")
BF(4,4, ACT CHECK, rp orig digit 8, "Digit 8")
BF(4,9, ACT CHECK, rp orig digit 7, "Digit 7")
BF(4,1, ACT CHECK, rp orig digit 10, "Digit 10")
BF(4,1, ACT CHECK, rp orig digit 9, "Digit 9")
BF(4,0xF, ACT CHECK, rp orig digit 12, "Digit 12")
BF(4,0x7, ACT CHECK, rp orig digit 11, "Digit 11")
BF(8, 0, ACT CHECK, rp dest len, "rp destination address")
BF(8, 26, ACT CHECK, rp user data len, "rp user data len")
BF(1, 1, ACT CHECK, tp rp, "reply path")
BF(1, 0, ACT CHECK, tp udhi, "no user data header indicator")
BF(1, 0, ACT CHECK, tp sri, "no status report indication")
BF(2, 0, ACT CHECK, tp spare, "spare bits")
BF(1, 0, ACT CHECK, tp mms, "no more messages to send")
BF(2, 0, ACT CHECK, tp mti, "sms deliver")
BF(8, 8, ACT CHECK, tp oa len, "tp oa with eight digits")
BF(1, 1, ACT CHECK, tp oa ext, "Extension Bit")
BF(3, 0, ACT CHECK, tp oa type of number, "Unknown")
BF(4, 1, ACT CHECK, tp oa numbering plan, "ISDN/telephony")
BF(4,3, ACT CHECK, tp oa digit 2, "Digit 2")
BF(4,0, ACT CHECK, tp oa digit 1, "Digit 1")
BF(4,3, ACT CHECK, tp oa digit 4, "Digit 4")
BF(4,0, ACT CHECK, tp oa digit 3, "Digit 3")
BF(4,0, ACT CHECK, tp oa digit 6, "Digit 6")
BF(4,9, ACT CHECK, tp oa digit 5, "Digit 5")
BF(4,4, ACT CHECK, tp oa digit 8, "Digit 8")
BF(4,9, ACT CHECK, tp oa digit 7, "Digit 7")
BF(8,0x1F, ACT CHECK, tp pd, "protocol identifier")
BF(8, 0xF1, ACT CHECK, tp dcs, "data coding scheme")
BF(8, 0x89, ACT CHECK, tp scts year, "year 98")

BF(8, 0x40, ACT CHECK, tp scts month, "month 04")
BF(8, 0x61, ACT CHECK, tp scts day, "day 16")
BF(8, 0x01, ACT CHECK, tp scts hour, "hour 10")
BF(8, 0x51, ACT CHECK, tp scts minute, "minute 15")
BF(8, 0x43, ACT CHECK, tp scts second, "second 34")
BF(8, 0x00, ACT CHECK, tp scts timezone, "zone 00")
BF(8, 0x0A, ACT CHECK, tp ud len, "characters in sms")
BF(8, 0xC8, ACT CHECK, tp ud 1, "SMS Halli Hallo")
BF(8, 0x20, ACT CHECK, tp ud 2, SILENT)
BF(8, 0x93, ACT CHECK, tp ud 3, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 4, SILENT)
BF(8, 0x44, ACT CHECK, tp ud 5, SILENT)
BF(8, 0x06, ACT CHECK, tp ud 6, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 7, SILENT)
BF(8, 0xCC, ACT CHECK, tp ud 8, SILENT)
BF(8, 0x27, ACT CHECK, tp ud 9, SILENT)
IE END(cp user data sms deliver class1)

IE BEGIN(cp user data sms deliver def)
BF(8,38, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 1, ACT CHECK, mti, "rp data network to mobile")
BF(8, 22, ACT CHECK, msg ref, "message reference")
BF(8, 7, ACT CHECK, rp orig len, "rp originator address")
BF(1, 1, ACT CHECK, rp orig ext, "Extension Bit")
BF(3, 0, ACT CHECK, rp orig type of number, "Unknown")
BF(4, 1, ACT CHECK, rp orig numbering plan, "ISDN/telephony")
BF(4,3, ACT CHECK, rp orig digit 2, "Digit 2")
BF(4,0, ACT CHECK, rp orig digit 1, "Digit 1")
BF(4,3, ACT CHECK, rp orig digit 4, "Digit 4")
BF(4,0, ACT CHECK, rp orig digit 3, "Digit 3")
BF(4,0, ACT CHECK, rp orig digit 6, "Digit 6")
BF(4,9, ACT CHECK, rp orig digit 5, "Digit 5")
BF(4,4, ACT CHECK, rp orig digit 8, "Digit 8")
BF(4,9, ACT CHECK, rp orig digit 7, "Digit 7")
BF(4,1, ACT CHECK, rp orig digit 10, "Digit 10")
BF(4,1, ACT CHECK, rp orig digit 9, "Digit 9")
BF(4,0xF, ACT CHECK, rp orig digit 12, "Digit 12")
BF(4,0x7, ACT CHECK, rp orig digit 11, "Digit 11")
BF(8, 0, ACT CHECK, rp dest len, "rp destination address")
BF(8, 26, ACT CHECK, rp user data len, "rp user data len")
BF(1, 1, ACT CHECK, tp rp, "reply path")
BF(1, 0, ACT CHECK, tp udhi, "no user data header indicator")
BF(1, 0, ACT CHECK, tp sri, "no status report indication")
BF(2, 0, ACT CHECK, tp spare, "spare bits")
BF(1, 0, ACT CHECK, tp mms, "no more messages to send")
BF(2, 0, ACT CHECK, tp mti, "sms deliver")
BF(8, 8, ACT CHECK, tp oa len, "tp oa with eight digits")
BF(1, 1, ACT CHECK, tp oa ext, "Extension Bit")
BF(3, 0, ACT CHECK, tp oa type of number, "Unknown")
BF(4, 1, ACT CHECK, tp oa numbering plan, "ISDN/telephony")
BF(4,3, ACT CHECK, tp oa digit 2, "Digit 2")
BF(4,0, ACT CHECK, tp oa digit 1, "Digit 1")
BF(4,3, ACT CHECK, tp oa digit 4, "Digit 4")
BF(4,0, ACT CHECK, tp oa digit 3, "Digit 3")
BF(4,0, ACT CHECK, tp oa digit 6, "Digit 6")
BF(4,9, ACT CHECK, tp oa digit 5, "Digit 5")
BF(4,4, ACT CHECK, tp oa digit 8, "Digit 8")
BF(4,9, ACT CHECK, tp oa digit 7, "Digit 7")

BF(8,0x1F, ACT CHECK, tp pd, "protocol identifier")
BF(8, 0x00, ACT CHECK, tp dcs, "data coding scheme")
BF(8, 0x89, ACT CHECK, tp scts year, "year 98")
BF(8, 0x40, ACT CHECK, tp scts month, "month 04")
BF(8, 0x61, ACT CHECK, tp scts day, "day 16")
BF(8, 0x01, ACT CHECK, tp scts hour, "hour 10")
BF(8, 0x51, ACT CHECK, tp scts minute, "minute 15")
BF(8, 0x43, ACT CHECK, tp scts second, "second 34")
BF(8, 0x00, ACT CHECK, tp scts timezone, "zone 00")
BF(8, 0x0A, ACT CHECK, tp ud len, "characters in sms")
BF(8, 0xC8, ACT CHECK, tp ud 1, "SMS Halli Hallo")
BF(8, 0x20, ACT CHECK, tp ud 2, SILENT)
BF(8, 0x93, ACT CHECK, tp ud 3, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 4, SILENT)
BF(8, 0x44, ACT CHECK, tp ud 5, SILENT)
BF(8, 0x06, ACT CHECK, tp ud 6, SILENT)
BF(8, 0x99, ACT CHECK, tp ud 7, SILENT)
BF(8, 0xCC, ACT CHECK, tp ud 8, SILENT)
BF(8, 0x27, ACT CHECK, tp ud 9, SILENT)
IE END(cp user data sms deliver def)

IE BEGIN(cp user data rp ack to bs)
BF(8, 2, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 2, ACT CHECK, mti, "rp ack mobile to network")
BF(8, 22, ACT CHECK, msg ref, "message reference")
IE END(cp user data rp ack to bs)

IE BEGIN(cp user data rp ack to ms)
BF(8, 2, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 2, ACT CHECK, mti, "rp ack network to mobile")
BF(8, 22, ACT CHECK, msg ref, "message reference")
IE END(cp user data rp ack to ms)

IE BEGIN(cp user data rp error 111 to bs)
BF(8, 4, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 4, ACT CHECK, mti, "rp error mobile to network")
BF(8, 22, ACT CHECK, msg ref, "message reference")
BF(8, 1, ACT CHECK, cause len, "rp error cause len")
BF(8, 111, ACT CHECK, cause, "rp error cause")
IE END(cp user data rp error 111 to bs)

IE BEGIN(cp user data rp error 22 to bs)
BF(8, 4, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 4, ACT CHECK, mti, "rp error mobile to network")
BF(8, 22, ACT CHECK, msg ref, "message reference")
BF(8, 1, ACT CHECK, cause len, "rp error cause len")
BF(8, 22, ACT CHECK, cause, "rp error cause")
IE END(cp user data rp error 22 to bs)

IE BEGIN(cp user data rp smma to bs)
BF(8, 2, ACT CHECK, len, "length of cp user data")
BF(5, 0, ACT CHECK, spare, "spare bits")
BF(3, 6, ACT CHECK, mti, "rp smma mobile to network")
BF(8, 0, ACT CHECK, msg ref, "message reference")
IE END(cp user data rp smma to bs)

IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(immediate_assignment_message_type)
IE(spare_half_octet)
IE(page_mode)
IE(channel_description)
IE(request_reference)
IE(timing_advance)
IE(mobile_allocation)
IE(ia_rest_octets)
MSG3_END(immediate_assignment)
MSG3_BEGIN(immediate_assignment_tch)
IE(l2_pseudo_length_21)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(immediate_assignment_message_type)
IE(spare_half_octet)
IE(page_mode)
IE(channel_description_tch)
IE(request_reference)
IE(timing_advance)
IE(mobile_allocation)
IE(ia_rest_octets)
MSG3_END(immediate_assignment_tch)
MSG3_BEGIN(paging_response)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(paging_response_message_type)
IE(spare_half_octet)
IE(ciphering_key_sequence_number)
IE(ms_classmark)
IE(mobile_identity)
MSG3_END(paging_response)
MSG3_BEGIN(authentication_request)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(authentication_request_message_type)
IE(spare_half_octet)
IE(ciphering_key_sequence_number_2)
IE(authentication_parameter_rand)
MSG3_END(authentication_request)
MSG3_BEGIN(authentication_response)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(authentication_response_message_type)
IE(authentication_parameter_sres)
MSG3_END(authentication_response)

MSG3_BEGIN(identity_request)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(identity_request_message_type)
IE(spare_half_octet)
IE(identity_type)
MSG3_END(identity_request)

MSG3_BEGIN(identity_response)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(identity_response_message_type)
IE(mobile_identity)
MSG3_END(identity_response)

MSG3_BEGIN(cm_service_request)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(cm_service_request_message_type)
IE(ciphering_key_sequence_number)
IE(cm_service_type_moc)
IE(ms_classmark)
IE(mobile_identity)
MSG3_END(cm_service_request)

MSG3_BEGIN(cm_service_abort)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(cm_service_abort_message_type)
MSG3_END(cm_service_abort)

MSG3_BEGIN(cm_service_reject)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(cm_service_reject_message_type)
IE(reject_cause)
MSG3_END(cm_service_reject)

MSG3_BEGIN(cm_service_accept)
IE(skip_indicator)
IE(mobility_management_protocol_discriminator)
IE(cm_service_accept_message_type)
MSG3_END(cm_service_accept)

MSG3_BEGIN(ciphering_mode_command)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(ciphering_mode_command_message_type)
IE(ciphering_mode_setting)
IE(cipher_response)
MSG3_END(ciphering_mode_command)

MSG3_BEGIN(ciphering_mode_complete)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(ciphering_mode_complete_message_type)
MSG3_END(ciphering_mode_complete)

MSG3_BEGIN(channel_mode_modify)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(channel_mode_modify_message_type)
IE(channel_description_tch)
IE(channel_mode_speech)
MSG3_END(channel_mode_modify)

MSG3_BEGIN(channel_mode_modify_acknowledge)
IE(skip_indicator)

IE(rr_management_protocol_discriminator)
IE(channel_mode_modify_acknowledge_message_type)
IE(channel_description_tch)
IE(channel_mode_speech)
MSG3_END(channel_mode_modify_acknowledge)

MSG3_BEGIN(setup) /* contains 'signal' but no 'Bearer Cap' */
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(setup_message_type)
IE(iei_34)
IE(signal_call_waiting)
MSG3_END(setup)

MSG3_BEGIN(setup_no_signal)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(setup_message_type)
MSG3_END(setup_no_signal)

MSG3_BEGIN(setup_moc)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(setup_message_type)
IE(iei_04)
IE(bearer_capability)
IE(iei_5E)
IE(called_party_bcd_number)
MSG3_END(setup_moc)

MSG3_BEGIN(setup_data)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(setup_message_type)
IE(iei_04)
IE(bearer_capability_data)
IE(iei_5E)
IE(called_party_bcd_number)
MSG3_END(setup_data)

MSG3_BEGIN(call_confirmed) /* contains bearer capability */
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(call_confirmed_message_type)
IE(iei_04)
IE(bearer_capability)
IE(iei_15)
IE(cc_capabilities)
MSG3_END(call_confirmed)

MSG3_BEGIN(call_confirmed_user_busy) /* contains cause user busy */
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(call_confirmed_message_type)
IE(iei_04)
IE(bearer_capability)
IE(iei_08)
IE(cause_17)
IE(iei_15)

IE(cc_capabilities)
MSG3_END(call_confirmed_user_busy)

MSG3_BEGIN(call_proceeding)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(call_proceeding_message_type)
MSG3_END(call_proceeding)

MSG3_BEGIN(connect)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(connect_message_type)
MSG3_END(connect)

MSG3_BEGIN(unknown_message)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(unknown_message_type)
MSG3_END(unknown_message)

MSG3_BEGIN(unknown_message_mtc)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(unknown_message_type)
MSG3_END(unknown_message_mtc)

MSG3_BEGIN(alerting)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(alerting_message_type)
MSG3_END(alerting)

MSG3_BEGIN(progress)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(progress_message_type)
IE(progress_indicator_32)
MSG3_END(progress)

MSG3_BEGIN(progress_1)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(progress_message_type)
IE(progress_indicator_8)
MSG3_END(progress_1)

MSG3_BEGIN(assignment_command)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(assignment_command_message_type)
IE(description_of_the_first_channel_after_time)
IE(power_command)
IE(iei_63)
IE(mode_of_the_first_channel)
MSG3_END(assignment_command)
MSG3_BEGIN(assignment_complete)
IE(skip_indicator)

IE(rr_management_protocol_discriminator)
IE(assignment_complete_message_type)
IE(rr_cause)
MSG3_END(assignment_complete)
MSG3_BEGIN(connect_acknowledge)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(connect_acknowledge_message_type)
MSG3_END(connect_acknowledge)

MSG3_BEGIN(status_enquiry_0)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_0)

MSG3_BEGIN(status_enquiry_1)
IE(transaction_identifier_dest_1)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_1)

MSG3_BEGIN(status_enquiry_2)
IE(transaction_identifier_dest_2)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_2)

MSG3_BEGIN(status_enquiry_3)
IE(transaction_identifier_dest_3)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_3)

MSG3_BEGIN(status_enquiry_4)
IE(transaction_identifier_dest_4)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_4)

MSG3_BEGIN(status_enquiry_5)
IE(transaction_identifier_dest_5)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_5)

MSG3_BEGIN(status_enquiry_6)
IE(transaction_identifier_dest_6)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_6)

MSG3_BEGIN(status_enquiry_0_mtc)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_0_mtc)

MSG3_BEGIN(status_enquiry_1_mtc)

IE(transaction_identifier_source_1)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_1_mtc)

MSG3_BEGIN(status_enquiry_2_mtc)
IE(transaction_identifier_source_2)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_2_mtc)

MSG3_BEGIN(status_enquiry_3_mtc)
IE(transaction_identifier_source_3)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_3_mtc)

MSG3_BEGIN(status_enquiry_4_mtc)
IE(transaction_identifier_source_4)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_4_mtc)

MSG3_BEGIN(status_enquiry_5_mtc)
IE(transaction_identifier_source_5)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_5_mtc)

MSG3_BEGIN(status_enquiry_6_mtc)
IE(transaction_identifier_source_6)
IE(call_control_protocol_discriminator)
IE(status_enquiry_message_type)
MSG3_END(status_enquiry_6_mtc)

MSG3_BEGIN(disconnect_t303)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_102_303)
MSG3_END(disconnect_t303)

MSG3_BEGIN(disconnect_t310)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_102_310)
MSG3_END(disconnect_t310)

MSG3_BEGIN(disconnect_t313)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_102_313)
MSG3_END(disconnect_t313)

MSG3_BEGIN(disconnect_8)

IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_16)
IE(iei_1E)
IE(progress_indicator_8)
MSG3_END(disconnect_8)

MSG3_BEGIN(disconnect_8_mtc)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_16)
IE(iei_1E)
IE(progress_indicator_8)
MSG3_END(disconnect_8_mtc)

MSG3_BEGIN(disconnect_ms)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_16)
MSG3_END(disconnect_ms)

MSG3_BEGIN(disconnect)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(disconnect_message_type)
IE(cause_16)
MSG3_END(disconnect)

MSG3_BEGIN(release_complete)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_03)
MSG3_END(release_complete)

MSG3_BEGIN(release_complete_empty)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
MSG3_END(release_complete_empty)

MSG3_BEGIN(release_complete_ms)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
MSG3_END(release_complete_ms)

MSG3_BEGIN(release)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(release_message_type)
MSG3_END(release)

MSG3_BEGIN(release_ms)
IE(transaction_identifier_dest)

IE(call_control_protocol_discriminator)
IE(release_message_type)
MSG3_END(release_ms)

MSG3_BEGIN(release_mtc)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(release_message_type)
MSG3_END(release_mtc)

MSG3_BEGIN(release_t305)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(release_message_type)
IE(iei_08)
IE(cause_16)
MSG3_END(release_t305)

MSG3_BEGIN(release_bs)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(release_message_type)
IE(iei_08)
IE(cause_31)
MSG3_END(release_bs)

MSG3_BEGIN(release_complete_88)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_88)
MSG3_END(release_complete_88)

MSG3_BEGIN(release_complete_bs)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
MSG3_END(release_complete_bs)

MSG3_BEGIN(release_complete_0)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_0)

MSG3_BEGIN(release_complete_1)
IE(transaction_identifier_source_1)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_1)

MSG3_BEGIN(release_complete_2)
IE(transaction_identifier_source_2)
IE(call_control_protocol_discriminator)

IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_2)

MSG3_BEGIN(release_complete_3)
IE(transaction_identifier_source_3)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_3)

MSG3_BEGIN(release_complete_4)
IE(transaction_identifier_source_4)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_4)

MSG3_BEGIN(release_complete_5)
IE(transaction_identifier_source_5)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_5)

MSG3_BEGIN(release_complete_6)
IE(transaction_identifier_source_6)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_6)

MSG3_BEGIN(release_complete_0_mtc)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_0_mtc)

MSG3_BEGIN(release_complete_1_mtc)
IE(transaction_identifier_dest_1)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)
MSG3_END(release_complete_1_mtc)

MSG3_BEGIN(release_complete_2_mtc)
IE(transaction_identifier_dest_2)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)

MSG3_END(release_complete_2_mtc)

MSG3_BEGIN(release_complete_3_mtc)
IE(transaction_identifier_dest_3)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)

MSG3_END(release_complete_3_mtc)

MSG3_BEGIN(release_complete_4_mtc)
IE(transaction_identifier_dest_4)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)

MSG3_END(release_complete_4_mtc)

MSG3_BEGIN(release_complete_5_mtc)
IE(transaction_identifier_dest_5)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)

MSG3_END(release_complete_5_mtc)

MSG3_BEGIN(release_complete_6_mtc)
IE(transaction_identifier_dest_6)
IE(call_control_protocol_discriminator)
IE(release_complete_message_type)
IE(iei_08)
IE(cause_81)

MSG3_END(release_complete_6_mtc)

MSG3_BEGIN(channel_release)
IE(skip_indicator)
IE(rr_management_protocol_discriminator)
IE(channel_release_message_type)
IE(rr_cause)

MSG3_END(channel_release)

MSG3_BEGIN(status_30_1)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_1)

MSG3_END(status_30_1)

MSG3_BEGIN(status_30_3)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_3)

MSG3_END(status_30_3)

MSG3_BEGIN(status_30_4)

IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_4)
MSG3_END(status_30_4)

MSG3_BEGIN(status_30_7)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_7)
MSG3_END(status_30_7)

MSG3_BEGIN(status_30_8)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_8)
MSG3_END(status_30_8)

MSG3_BEGIN(status_30_9)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_9)
MSG3_END(status_30_9)

MSG3_BEGIN(status_30_10)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_10)
MSG3_END(status_30_10)

MSG3_BEGIN(status_30_10_mtc)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_10)
MSG3_END(status_30_10_mtc)

MSG3_BEGIN(status_30_11)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_11)
MSG3_END(status_30_11)

MSG3_BEGIN(status_30_11_mtc)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)

IE(cause_30)
IE(call_state_11)
MSG3_END(status_30_11_mtc)

MSG3_BEGIN(status_30_12)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_12)
MSG3_END(status_30_12)

MSG3_BEGIN(status_30_12_mtc)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_12)
MSG3_END(status_30_12_mtc)

MSG3_BEGIN(status_30_19)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_19)
MSG3_END(status_30_19)

MSG3_BEGIN(status_30_19_mtc)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_30)
IE(call_state_19)
MSG3_END(status_30_19_mtc)

MSG3_BEGIN(status_97_1)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_1)
MSG3_END(status_97_1)

MSG3_BEGIN(status_97_3)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_3)
MSG3_END(status_97_3)

MSG3_BEGIN(status_97_4)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_4)
MSG3_END(status_97_4)

MSG3_BEGIN(status_97_7)

IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_7)

MSG3_END(status_97_7)

MSG3_BEGIN(status_97_8)

IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_8)

MSG3_END(status_97_8)

MSG3_BEGIN(status_97_9)

IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_9)

MSG3_END(status_97_9)

MSG3_BEGIN(status_97_11)

IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_11)

MSG3_END(status_97_11)

MSG3_BEGIN(status_97_12)

IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(status_message_type)
IE(cause_97_20)
IE(call_state_12)

MSG3_END(status_97_12)

MSG3_BEGIN(stop_dtmf)

IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(stop_dtmf_message_type)

MSG3_END(stop_dtmf)

MSG3_BEGIN(stop_dtmf_acknowledge)

IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(stop_dtmf_acknowledge_message_type)

MSG3_END(stop_dtmf_acknowledge)

MSG3_BEGIN(start_dtmf_reject)

IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_reject_message_type)
IE(cause_03)

MSG3_END(start_dtmf_reject)

MSG3_BEGIN(start_dtmf_0)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_0)
MSG3_END(start_dtmf_0)

MSG3_BEGIN(start_dtmf_1)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_1)
MSG3_END(start_dtmf_1)

MSG3_BEGIN(start_dtmf_2)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_2)
MSG3_END(start_dtmf_2)

MSG3_BEGIN(start_dtmf_3)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_3)
MSG3_END(start_dtmf_3)

MSG3_BEGIN(start_dtmf_4)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_4)
MSG3_END(start_dtmf_4)

MSG3_BEGIN(start_dtmf_5)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_5)
MSG3_END(start_dtmf_5)

MSG3_BEGIN(start_dtmf_6)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_6)
MSG3_END(start_dtmf_6)

MSG3_BEGIN(start_dtmf_7)
IE(transaction_identifier_dest)

IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_7)
MSG3_END(start_dtmf_7)

MSG3_BEGIN(start_dtmf_8)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_8)
MSG3_END(start_dtmf_8)

MSG3_BEGIN(start_dtmf_9)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_9)
MSG3_END(start_dtmf_9)

MSG3_BEGIN(start_dtmf_A)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_A)
MSG3_END(start_dtmf_A)

MSG3_BEGIN(start_dtmf_B)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_B)
MSG3_END(start_dtmf_B)

MSG3_BEGIN(start_dtmf_C)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_C)
MSG3_END(start_dtmf_C)

MSG3_BEGIN(start_dtmf_D)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_D)
MSG3_END(start_dtmf_D)

MSG3_BEGIN(start_dtmf_star)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)

IE(keypad_facility_star)
MSG3_END(start_dtmf_star)

MSG3_BEGIN(start_dtmf_hash)
IE(transaction_identifier_dest)
IE(call_control_protocol_discriminator)
IE(start_dtmf_message_type)
IE(iei_2C)
IE(keypad_facility_hash)
MSG3_END(start_dtmf_hash)

MSG3_BEGIN(start_dtmf_acknowledge_0)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_0)
MSG3_END(start_dtmf_acknowledge_0)

MSG3_BEGIN(start_dtmf_acknowledge_1)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_1)
MSG3_END(start_dtmf_acknowledge_1)

MSG3_BEGIN(start_dtmf_acknowledge_2)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_2)
MSG3_END(start_dtmf_acknowledge_2)

MSG3_BEGIN(start_dtmf_acknowledge_3)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_3)
MSG3_END(start_dtmf_acknowledge_3)

MSG3_BEGIN(start_dtmf_acknowledge_4)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_4)
MSG3_END(start_dtmf_acknowledge_4)

MSG3_BEGIN(start_dtmf_acknowledge_5)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_5)
MSG3_END(start_dtmf_acknowledge_5)

MSG3_BEGIN(start_dtmf_acknowledge_6)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_6)
MSG3_END(start_dtmf_acknowledge_6)

MSG3_BEGIN(start_dtmf_acknowledge_7)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_7)
MSG3_END(start_dtmf_acknowledge_7)

MSG3_BEGIN(start_dtmf_acknowledge_8)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_8)
MSG3_END(start_dtmf_acknowledge_8)

MSG3_BEGIN(start_dtmf_acknowledge_9)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_9)
MSG3_END(start_dtmf_acknowledge_9)

MSG3_BEGIN(start_dtmf_acknowledge_A)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_A)
MSG3_END(start_dtmf_acknowledge_A)

MSG3_BEGIN(start_dtmf_acknowledge_B)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_B)
MSG3_END(start_dtmf_acknowledge_B)

MSG3_BEGIN(start_dtmf_acknowledge_C)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_C)
MSG3_END(start_dtmf_acknowledge_C)

MSG3_BEGIN(start_dtmf_acknowledge_D)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)

IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_D)
MSG3_END(start_dtmf_acknowledge_D)

MSG3_BEGIN(start_dtmf_acknowledge_star)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_star)
MSG3_END(start_dtmf_acknowledge_star)

MSG3_BEGIN(start_dtmf_acknowledge_hash)
IE(transaction_identifier_source)
IE(call_control_protocol_discriminator)
IE(start_dtmf_acknowledge_message_type)
IE(iei_2C)
IE(keypad_facility_hash)
MSG3_END(start_dtmf_acknowledge_hash)

MSG3 BEGIN(cp data sms submit)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data sms submit)
MSG3 END(cp data sms submit)

MSG3 BEGIN(cp data sms deliver)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data sms deliver)
MSG3 END(cp data sms deliver)

MSG3 BEGIN(cp data sms deliver class1)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data sms deliver class1)
MSG3 END(cp data sms deliver class1)

MSG3 BEGIN(cp data sms deliver def)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data sms deliver def)
MSG3 END(cp data sms deliver def)

MSG3 BEGIN(cp data rp ack to bs)
IE(transaction identifier dest)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data rp ack to bs)
MSG3 END(cp data rp ack to bs)

MSG3 BEGIN(cp data rp error 111 to bs)
IE(transaction identifier dest)
IE(sms protocol discriminator)

IE(cp data message type)
IE(cp user data rp error 111 to bs)
MSG3 END(cp data rp error 111 to bs)

MSG3 BEGIN(cp data rp error 22 to bs)
IE(transaction identifier dest)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data rp error 22 to bs)
MSG3 END(cp data rp error 22 to bs)

MSG3 BEGIN(cp data rp smma to bs)
IE(transaction identifier dest)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data rp smma to bs)
MSG3 END(cp data rp smma to bs)

MSG3 BEGIN(cp ack to bs)
IE(transaction identifier dest)
IE(sms protocol discriminator)
IE(cp ack message type)
MSG3 END(cp ack to bs)

MSG3 BEGIN(cp ack to bs mo)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp ack message type)
MSG3 END(cp ack to bs mo)

MSG3 BEGIN(cp ack to ms)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp ack message type)
MSG3 END(cp ack to ms)

MSG3 BEGIN(cp ack to ms mo)
IE(transaction identifier dest)
IE(sms protocol discriminator)
IE(cp ack message type)
MSG3 END(cp ack to ms mo)

MSG3 BEGIN(cp data rp ack to bs mo)
IE(transaction identifier source)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data rp ack to bs)
MSG3 END(cp data rp ack to bs mo)

MSG3 BEGIN(cp data rp ack to ms mo)
IE(transaction identifier dest)
IE(sms protocol discriminator)
IE(cp data message type)
IE(cp user data rp ack to ms)
MSG3 END(cp data rp ack to ms mo)

MSG3 BEGIN(cp error to ms mo)
IE(transaction identifier dest)
IE(sms protocol discriminator)

IE(cp error message type)
MSG3 END(cp error to ms mo)

MSG3 BEGIN(sms cbch message)
IE(cbch message)
MSG3 END(sms cbch message)

4 TEST CASES

4.1 Preambles

4.1.1 MSMS000: Initialization (One cell)

Description:

Preamble: None

Script:

```
ISS_INIT          ( 1 )

BS_SET_SYS_INFO  ( 0, system_information_type_1 )
BS_SET_SYS_INFO  ( 0, system_information_type_2 )
BS_SET_SYS_INFO  ( 0, system_information_type_3 )
BS_SET_SYS_INFO  ( 0, system_information_type_4 )
BS_SET_SYS_INFO_SACCH( 0, system_information_type_5 )
BS_SET_SYS_INFO_SACCH( 0, system_information_type_6 )

BS_SET_SCH        ( 0,BSIC , RFN                )
BS_SET_ARFCN      ( 0,ARFCN_BCCH                )
BS_SET_POWER      ( 0,-73                       )
BS_ON_OFF         ( 0,TRUE                      )

COMMAND          ( "MMI CONFIG KEY_SEQUENCE=<#*91*1#>" )

ISS_DELAY        ( 20000 ) /* wait until the MS has detected the cell */

SET_TIMEOUT      (40000)
```

History: 15.12.97 VK Initial

4.2 Short message service point to point

4.2.1 MSMS001: SMS mobile terminated (34.2.1)

Description: To verify the ability of a MS to receive and decode the SMS where provided for the point to point service.

Preamble: MSMS000

Script:

```

BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT    ( 0, channel_request,      "2: Establishment cause is Answer to paging"
                  )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND      ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT    ( 0, paging_response,      "4: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "5" )
BS_MSG3_AWAIT    ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
                  message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )

NOT_IMPLEMENTED  ( "9: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND     ( 0, sabm_(sapi=3),        "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT    ( 0, ua_(sapi=3),        "11: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND     ( 0, cp_data_sms_deliver,  "12: Contains RP-DATA RPDU (SMS DELIVER
                  TPDU)" )
NOT_IMPLEMENTED  ( "13: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT    ( 0, cp_ack_to_bs,        "14" )
NOT_IMPLEMENTED  ( "15: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT    ( 0, cp_data_rp_ack_to_bs, "16: Contains RP-ACK RPDU" )
BS_MSG3_SEND     ( 0, cp_ack_to_ms,        "17: Within TCIM after step 16, no further
                  CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND     ( 0, channel_release,      "18: The main signalling link is released."
                  )
NOT_IMPLEMENTED  ( "19: The MS shall indicate that an SM has
                  arrived. If the MS provides the
                  functionality to display MT messages, it is
                  checked that the correct message is
                  displayed" )

COMMAND          ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)
COMMAND          ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)

BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "20" )
BS_RACH_AWAIT    ( 0, channel_request,      "21: Establishment cause is Answer to
                  paging" )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND      ( 0, immediate_assignment, "22: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT    ( 0, paging_response,      "23: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "24" )
BS_MSG3_AWAIT    ( 0, authentication_response, "25: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "26: SS starts deciphering after sending the
                  message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "27: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )

NOT_IMPLEMENTED  ( "28: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND     ( 0, sabm_(sapi=3),        "29: SS establishes SAPI 3" )
BS_MSG3_AWAIT    ( 0, ua_(sapi=3),        "30: The MS shall respond to the SABM" )
*/
    
```

BS_CONFIG_CHANNEL	(0, SDCCH, ACK, SAPI_3)	
BS_MSG3_SEND	(0, cp_data_sms_deliver,	"31: Contains RP-DATA RPDU (SMS DELIVER TPDU)")
NOT_IMPLEMENTED	("32: Waits max 25 seconds for CP-ACK")
BS_MSG3_AWAIT	(0, cp_ack_to_bs,	"33")
NOT_IMPLEMENTED	("34: Waits max 60 seconds for RP-ACK RPDU")
BS_MSG3_AWAIT	(0, cp_data_rp_ack_to_bs,	"35: First CP-DATA from MS, contains RP-ACK RPDU")
NOT_IMPLEMENTED	("36: First CP-DATA message not acknowledged by SS")
ISS_DELAY (5000)		
BS_MSG3_AWAIT	(0, cp_data_rp_ack_to_bs,	"37: Retransmitted CP-DATA from MS within twice TC1M, after step 35, contains RP-ACK RPDU")
BS_MSG3_SEND	(0, cp_ack_to_ms,	"38: Second CP-DATA message is acknowledged within TC1M after step 37, no further CP-DATA messages")
BS_CONFIG_CHANNEL	(0, SDCCH, ACK, SAPI_0)	
BS_MSG3_SEND	(0, channel_release,	"39: The main signalling link is released.")
NOT_IMPLEMENTED	("40: The MS shall indicate that an SM has arrived. If the MS provides the functionality to display MT messages, it is checked that the correct message is displayed")
ISS_DELAY (5000)		
COMMAND	("MMI CONFIG AT+CMGR=1")	
ISS_DELAY (5000)		
COMMAND	("MMI CONFIG AT+CMGD=1")	
ISS_DELAY (2000)		
BS_CONFIG_CHANNEL	(0, PCH, UNACK, SAPI_0)	
BS_MSG3_SEND	(0, paging_request_type_1,	"41")
BS_RACH_AWAIT	(0, channel_request,	"42: Establishment cause is Answer to paging")
BS_CONFIG_CHANNEL	(0, AGCH, UNACK, SAPI_0)	
BS_STORE_RACH_PARAMS	(0, 0)	
BS_MSG3_SEND	(0, immediate_assignment,	"43: SS assigns an SDCCH")
BS_CONFIG_CHANNEL	(0, SDCCH, ACK, SAPI_0)	
BS_MSG3_AWAIT	(0, paging_response,	"44: Message is contained in SABM.")
BS_MSG3_SEND	(0, authentication_request,	"45")
BS_MSG3_AWAIT	(0, authentication_response,	"46: SRES specifies correct value.")
BS_MSG3_SEND	(0, ciphering_mode_command,	"47: SS starts deciphering after sending the message.")
BS_MSG3_AWAIT	(0, ciphering_mode_complete,	"48: Shall be sent enciphered. All following messages shall be sent enciphered.")
NOT_IMPLEMENTED	("49: SS starts ciphering.")
/*		
not implemented		
BS_MSG3_SEND	(0, sabm(sapi=3),	"50: SS establishes SAPI 3")
BS_MSG3_AWAIT	(0, ua(sapi=3),	"51: The MS shall respond to the SABM")
*/		
BS_CONFIG_CHANNEL	(0, SDCCH, ACK, SAPI_3)	
BS_MSG3_SEND	(0, cp_data_sms_deliver,	"52: Contains RP-DATA RPDU (SMS DELIVER TPDU)")
NOT_IMPLEMENTED	("53: Waits max 25 seconds for CP-ACK")
BS_MSG3_AWAIT	(0, cp_ack_to_bs,	"54")
NOT_IMPLEMENTED	("55: Waits max 60 seconds for RP-ACK RPDU")
BS_MSG3_AWAIT	(0, cp_data_rp_ack_to_bs,	"56: Contains RP-ACK RPDU")
NOT_IMPLEMENTED	("57: First CP-DATA message not acknowledged by SS")
ISS_DELAY (5000)		
BS_MSG3_AWAIT	(0, cp_data_rp_ack_to_bs,	"58: Retransmitted CP-DATA from MS within twice TC1M after step 56, contains RP-ACK RPDU")
NOT_IMPLEMENTED	("59: Retransmitted CP-DATA message not acknowledged by SS")
NOT_IMPLEMENTED	("60: Depending upon the maximum number of CP-DATA retransmissions implemented, step 58 and 59 may be repeated.")
BS_CONFIG_CHANNEL	(0, SDCCH, ACK, SAPI_0)	
BS_MSG3_SEND	(0, channel_release,	"61: The main signalling link is released after a duration of TC1M + 5 seconds after the last CP-DATA retransmission..")
ISS_DELAY (5000)		

```

NOT_IMPLEMENTED      (
                                "62: The MS shall indicate that an SM has
                                arrived. If the MS provides the
                                functionality to display MT messages, it is
                                checked that the correct message is
                                displayed" )

COMMAND              ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)
COMMAND              ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)

COMMAND ("MMI CONFIG KEY_SEQUENCE=03039094117")

BS_RACH_AWAIT       ( 0,channel_request_moc,      SILENT)
BS_CONFIG_CHANNEL   ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND        ( 0,immediate_assignment,     SILENT)
BS_CONFIG_CHANNEL   ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT       ( 0, cm_service_request,      SILENT)
BS_MSG3_SEND        ( 0, cipherring_mode_command, SILENT)
BS_MSG3_AWAIT       ( 0, cipherring_mode_complete, SILENT)

BS_MSG3_AWAIT       ( 0, setup_moc,                SILENT)

BS_MSG3_SEND        ( 0, call_proceeding,          SILENT)
BS_MSG3_SEND        ( 0, alerting,                 SILENT)
BS_MSG3_SEND        ( 0, assignment_command,        SILENT)
BS_CONFIG_CHANNEL   ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_AWAIT       ( 0, assignment_complete,      SILENT)

BS_MSG3_SEND        ( 0, connect,                  SILENT)
BS_MSG3_AWAIT       ( 0, connect_acknowledge,      SILENT)

NOT_IMPLEMENTED      (
                                "63: A data or speech call is established on
                                a TCH and the state U10 of call control is
                                entered." )

/*
not implemented

BS_MSG3_SEND        ( 0, sabm_(sapi=3),            "64: Sent on SACCH associated with TCH" )
BS_MSG3_AWAIT       ( 0, ua_(sapi=3),              "65: The MS shall respond to the SABM" )
*/
BS_CONFIG_CHANNEL   ( 0, SACCH, ACK,SAPI_3)
BS_MSG3_SEND        ( 0, cp_data_sms_deliver,      "66: Contains RP-DATA RPDU (SMS DELIVER
                                TPDU)" )
NOT_IMPLEMENTED      (
                                "67: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT       ( 0, cp_ack_to_bs,              "68" )
NOT_IMPLEMENTED      (
                                "69: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT       ( 0, cp_data_rp_ack_to_bs,     "70: Contains RP-ACK RPDU" )
BS_MSG3_SEND        ( 0, cp_ack_to_ms,              "71: Within TC1M after step 70, no further
                                CP-DATA messages" )

BS_CONFIG_CHANNEL   ( 0, FACCH, ACK,SAPI_0)
BS_MSG3_SEND        ( 0, channel_release,          "72: The main signalling link is released."
                                )

ISS_DELAY (5000)
NOT_IMPLEMENTED      (
                                "73: The MS shall indicate that an SM has
                                arrived. If the MS provides the
                                functionality to display MT messages, it is
                                checked that the correct message is
                                displayed" )

NOT_IMPLEMENTED      (
                                "74: Clear the SMS message store" )

COMMAND              ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)
COMMAND              ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)

COMMAND ("MMI CONFIG KEY_SEQUENCE=03039094117")

BS_RACH_AWAIT       ( 0,channel_request_moc,      SILENT)
BS_CONFIG_CHANNEL   ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND        ( 0,immediate_assignment,     SILENT)
BS_CONFIG_CHANNEL   ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT       ( 0, cm_service_request,      SILENT)
BS_MSG3_SEND        ( 0, cipherring_mode_command, SILENT)
    
```

```

BS_MSG3_AWAIT      ( 0, ciphering_mode_complete, SILENT)

BS_MSG3_AWAIT      ( 0, setup_moc,                SILENT)

BS_MSG3_SEND       ( 0, call_proceeding,          SILENT)
BS_MSG3_SEND       ( 0, alerting,                 SILENT)
BS_MSG3_SEND       ( 0, assignment_command,       SILENT)
BS_CONFIG_CHANNEL  ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_AWAIT      ( 0, assignment_complete,      SILENT)

BS_MSG3_SEND       ( 0, connect,                  SILENT)
BS_MSG3_AWAIT      ( 0, connect_acknowledge,      SILENT)

NOT_IMPLEMENTED    (                               "75: A data or speech call is established on
                                                         a TCH and the state U10 of call control is
                                                         entered." )

/*
not implemented
BS_MSG3_SEND       ( 0, sabm_(sapi=3),            "76: Sent on SACCH associated with TCH" )
BS_MSG3_AWAIT      ( 0, ua_(sapi=3),             "77: The MS shall respond to the SABM" )
*/
BS_CONFIG_CHANNEL  ( 0, SACCH, ACK, SAPI_3)
BS_MSG3_SEND       ( 0, cp_data_sms_deliver,      "78: Contains RP-DATA RPDU (SMS DELIVER
                                                         TPDU)" )

NOT_IMPLEMENTED    (                               "79: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT      ( 0, cp_ack_to_bs,             "80" )
NOT_IMPLEMENTED    (                               "81: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT      ( 0, cp_data_rp_ack_to_bs,     "82: First CP-DATA from MS, contains RP-ACK
                                                         RPDU" )

NOT_IMPLEMENTED    (                               "83: First CP-DATA message not acknowledged
                                                         by SS" )

ISS_DELAY (5000)
BS_MSG3_AWAIT      ( 0, cp_data_rp_ack_to_bs,     "84: Retransmitted CP-DATA message within
                                                         twice TC1M after step 82, contains RP-ACK
                                                         RPDU" )

BS_MSG3_SEND       ( 0, cp_ack_to_ms,            "85: Second CP-DATA message is acknowledged
                                                         within TC1M after step 84, no further CP-
                                                         DATA messages" )

BS_CONFIG_CHANNEL  ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_SEND       ( 0, channel_release,          "86: The main signalling link is released."
                                                         )

ISS_DELAY (5000)
NOT_IMPLEMENTED    (                               "87: The MS shall indicate that an SM has
                                                         arrived. If the MS provides the
                                                         functionality to display MT messages, it is
                                                         checked that the correct message is
                                                         displayed" )

NOT_IMPLEMENTED    (                               "88: Clear the SMS message store" )

COMMAND            ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)
COMMAND            ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)

COMMAND ("MMI CONFIG KEY_SEQUENCE=03039094117")

BS_RACH_AWAIT      ( 0, channel_request_moc,      SILENT)
BS_CONFIG_CHANNEL  ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND       ( 0, immediate_assignment,     SILENT)
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT      ( 0, cm_service_request,       SILENT)
BS_MSG3_SEND       ( 0, ciphering_mode_command,   SILENT)
BS_MSG3_AWAIT      ( 0, ciphering_mode_complete,   SILENT)

BS_MSG3_AWAIT      ( 0, setup_moc,                SILENT)

BS_MSG3_SEND       ( 0, call_proceeding,          SILENT)
BS_MSG3_SEND       ( 0, alerting,                 SILENT)
BS_MSG3_SEND       ( 0, assignment_command,       SILENT)
BS_CONFIG_CHANNEL  ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_AWAIT      ( 0, assignment_complete,      SILENT)

BS_MSG3_SEND       ( 0, connect,                  SILENT)
BS_MSG3_AWAIT      ( 0, connect_acknowledge,      SILENT)

```

```

NOT_IMPLEMENTED ( "89: A data or speech call is established on
a TCH and the state U10 of call control is
entered." )

/*
not implemented
BS_MSG3_SEND ( 0, sabm_(sapi=3), "90: Sent on SACCH associated with TCH" )
BS_MSG3_AWAIT ( 0, ua_(sapi=3), "91: The MS shall respond to the SABM" )
*/
BS_CONFIG_CHANNEL ( 0, SACCH, ACK, SAPI_3)
BS_MSG3_SEND ( 0, cp_data_sms_deliver, "92: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )

NOT_IMPLEMENTED ( "93: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT ( 0, cp_ack_to_bs, "94" )
NOT_IMPLEMENTED ( "95: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_data_rp_ack_to_bs, "96: First CP-DATA from MS, contains RP-ACK
RPDU" )

NOT_IMPLEMENTED ( "97: First CP-DATA message not acknowledged
by SS" )

ISS_DELAY (5000)
BS_MSG3_AWAIT ( 0, cp_data_rp_ack_to_bs, "98: Transmitted CP-DATA message within
twice TC1M after step 96, contains RP-ACK
RPDU" )

NOT_IMPLEMENTED ( "99: Retransmitted CP-DATA message not
acknowledged by SS" )

NOT_IMPLEMENTED ( "100: Depending on the maximum number of CP-
DATA retransmissions implemented, step 98-
99 may be repeated. The maximum number of
retransmissions may however not exceed
three." )

BS_CONFIG_CHANNEL ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "101: The main signalling link is released
after a duration of TC1M + 15 seconds after
the last CP-DATA retransmission." )

ISS_DELAY (5000)
NOT_IMPLEMENTED ( "102: The MS shall indicate that an SM has
arrived. If the MS provides the
functionality to display MT messages, it is
checked that the correct message is
displayed" )

NOT_IMPLEMENTED ( "103: Clear the SMS message store" )

COMMAND ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)
COMMAND ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)

COMMAND ("MMI CONFIG KEY_SEQUENCE=03039094117")

BS_RACH_AWAIT ( 0, channel_request_moc, SILENT)
BS_CONFIG_CHANNEL ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, SILENT)
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT ( 0, cm_service_request, SILENT)
BS_MSG3_SEND ( 0, ciphering_mode_command, SILENT)
BS_MSG3_AWAIT ( 0, ciphering_mode_complete, SILENT)

BS_MSG3_AWAIT ( 0, setup_moc, SILENT)

BS_MSG3_SEND ( 0, call_proceeding, SILENT)
BS_MSG3_SEND ( 0, alerting, SILENT)
BS_MSG3_SEND ( 0, assignment_command, SILENT)
BS_CONFIG_CHANNEL ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_AWAIT ( 0, assignment_complete, SILENT)

BS_MSG3_SEND ( 0, connect, SILENT)
BS_MSG3_AWAIT ( 0, connect_acknowledge, SILENT)

NOT_IMPLEMENTED ( "104: A data or speech call is established
on a TCH and the state U10 of call control
is entered." )

/*
not implemented
BS_MSG3_SEND ( 0, sabm_(sapi=3), "105: Sent on SACCH associated with TCH" )

```

```

BS_MSG3_AWAIT      ( 0, ua_(sapi=3),          "106: The MS shall respond to the SABM" )
*/
BS_CONFIG_CHANNEL  ( 0, FACCH, ACK,SAPI_0)
BS_MSG3_SEND       ( 0, disconnect,          "107: The speech call is cleared by the SS.
The call clearing is continued on the FACCH
in parallel to the following exchange of
messages related to SMS." )

BS_MSG3_AWAIT      ( 0, release,              SILENT)
BS_CONFIG_CHANNEL  ( 0, SACCH, ACK,SAPI_3)
BS_MSG3_SEND       ( 0, cp_data_sms_deliver, "108: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )

NOT_IMPLEMENTED    (
BS_MSG3_AWAIT      ( 0, cp_ack_to_bs,        "109: Waits max 25 seconds for CP-ACK" )
NOT_IMPLEMENTED    (
BS_MSG3_SEND       ( 0, cp_ack_to_ms,        "110" )
NOT_IMPLEMENTED    (
BS_MSG3_AWAIT      ( 0, cp_data_rp_ack_to_bs, "111: Waits max 60 seconds for RP-ACK RPDU"
BS_MSG3_SEND       ( 0, cp_ack_to_ms,        )
"112: Contains RP-ACK RPDU" )
"113: Within TC1M after step 112, no further
CP-DATA" )

BS_CONFIG_CHANNEL  ( 0, FACCH, ACK,SAPI_0)
BS_MSG3_SEND       ( 0, channel_release,     "114: The main signalling link is released."
)

ISS_DELAY (5000)
NOT_IMPLEMENTED    (
"115: The MS shall indicate that an SM has
arrived. If the MS provides the
functionality to display MT messages, it is
checked that the correct message is
displayed" )

NOT_IMPLEMENTED    (
"116: Clear the SMS message store" )

COMMAND            ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)
COMMAND            ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)

COMMAND ("MMI CONFIG KEY_SEQUENCE=03039094117")

BS_RACH_AWAIT      ( 0,channel_request_moc,   SILENT)
BS_CONFIG_CHANNEL  ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS (0, 0);
BS_MSG3_SEND       ( 0,immediate_assignment,  SILENT)
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT      ( 0, cm_service_request,   SILENT)
BS_MSG3_SEND       ( 0, ciphering_mode_command, SILENT)
BS_MSG3_AWAIT      ( 0, ciphering_mode_complete, SILENT)

BS_MSG3_AWAIT      ( 0, setup_moc,           SILENT)

BS_MSG3_SEND       ( 0, call_proceeding,     SILENT)
BS_MSG3_SEND       ( 0, alerting,           SILENT)
BS_MSG3_SEND       ( 0, assignment_command,   SILENT)
BS_CONFIG_CHANNEL  ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_AWAIT      ( 0, assignment_complete,  SILENT)

BS_MSG3_SEND       ( 0, connect,             SILENT)
BS_MSG3_AWAIT      ( 0, connect_acknowledge,  SILENT)

NOT_IMPLEMENTED    (
"117: A data or speech call is established
on a TCH and the state U10 of call control
is entered." )

/*
not implemented
BS_MSG3_SEND       ( 0, sabm_(sapi=3),        "118: Sent on SACCH associated with TCH" )
BS_MSG3_AWAIT      ( 0, ua_(sapi=3),        "119: The MS shall respond to the SABM" )
*/
NOT_IMPLEMENTED    (
"120: Waits 15 seconds" )

COMMAND ("MMI CONFIG KEY_SEQUENCE=<#*43*1#>"); /* Hook On */
BS_MSG3_AWAIT      ( 0, disconnect_ms,       SILENT);

NOT_IMPLEMENTED    (
"121: During the SS wait time in step 120
the speech call shall be cleared from the
MS. The call clearing is continued on the
FACCH in parallel to the following exchange
of messages related to SMS." )

```

```

BS_MSG3_SEND      ( 0, release_ms,          SILENT)
BS_MSG3_AWAIT     ( 0, release_complete_ms, SILENT)

BS_CONFIG_CHANNEL ( 0, SACCH, ACK, SAPI_3)
BS_MSG3_SEND      ( 0, cp_data_sms_deliver, "122: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )
NOT_IMPLEMENTED   (                               "123: Waits max 25 seconds for CP-ACK" )
BS_CONFIG_CHANNEL ( 0, SACCH, ACK, SAPI_3)
BS_MSG3_AWAIT     ( 0, cp_ack_to_bs,        "124" )
NOT_IMPLEMENTED   (                               "125: Waits max 60 seconds for RP-ACK RPDU"
)
BS_MSG3_AWAIT     ( 0, cp_data_rp_ack_to_bs, "126: Contains RP-ACK RPDU" )
BS_MSG3_SEND      ( 0, cp_ack_to_ms,        "127: Within TC1M after step 126, no further
CP-DATA" )

BS_CONFIG_CHANNEL ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_SEND      ( 0, channel_release,     "128: The main signalling link is released."
)
NOT_IMPLEMENTED   (                               "129: The MS shall indicate that an SM has
arrived. If the MS provides the
functionality to display MT messages, it is
checked that the correct message is
displayed" )

NOT_IMPLEMENTED   (                               "130: Clear the SMS message store" )
COMMAND           ( "MMI CONFIG AT+CMGR=1" )
ISS_DELAY (5000)  (                               )
COMMAND           ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)  (                               )
    
```

History:

26.01.98

MS

Initial

4.2.2 MSMS002: SMS mobile originated (34.2.2)

Description: To verify that the MS is able to correctly send a short message where the SMS is provided for the point to point service. The test also verifies that the MS is capable of simultaneously receive a network originated SM whilst sending a mobile originated SM

Preamble: MSMS000

Script:

```

COMMAND ("MMI CONFIG AT+CSCA=12345678*145")
COMMAND ("MMI CONFIG AT+CSMP=18*03*0*0")
COMMAND ("MMI CONFIG AT+CMGS=5678*145*HALLIHALLO")

BS_RACH_AWAIT      ( 0, channel_request_111,      "1: Establishment cause is Other procedures
                  which can be completed with an SDDCH" NECI
                  = 0" )

BS_CONFIG_CHANNEL  ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND       ( 0, immediate_assignment,      "2: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_AWAIT      ( 0, cm_service_request,        "3: Message is contained in SABM on SAPI 0.
                  CM service type set to Short message
                  transfer" )

BS_MSG3_SEND       ( 0, authentication_request,    "4" )
BS_MSG3_AWAIT      ( 0, authentication_response,   "5: SRES specifies correct value." )
BS_MSG3_SEND       ( 0, ciphering_mode_command,    "6: SS starts deciphering after sending the
                  message." )

BS_MSG3_AWAIT      ( 0, ciphering_mode_complete,   "7: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )

NOT_IMPLEMENTED    (
NOT_IMPLEMENTED    (
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_3);
BS_MSG3_AWAIT      ( 0, cp_data_sms_submit,        "11: Contains RP-DATA RPDU (SMS SUBMIT
                  TPDU)" )

BS_MSG3_SEND       ( 0, cp_ack_to_ms_mo,           "12: Sent within TC1M after step 11" )
BS_MSG3_SEND       ( 0, cp_data_rp_ack_to_ms_mo,   "13: Contains RP-ACK RPDU" )
NOT_IMPLEMENTED    (
BS_MSG3_AWAIT      ( 0, cp_ack_to_bs_mo,           "14: Waits max 25 seconds for CP-ACK" )
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_SEND       ( 0, channel_release,           "15" )
                  "16: The main signalling link is released."
                  )

ISS_DELAY (10000)

COMMAND ("MMI CONFIG AT+CMGS=5678*145*HALLIHALLO")

BS_RACH_AWAIT      ( 0, channel_request_111,      "18: Establishment cause is Other procedures
                  which can be completed with an SDDCH"
                  (NECI=0)" )

BS_CONFIG_CHANNEL  ( 0, AGCH, UNACK, SAPI_0);
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND       ( 0, immediate_assignment,      "19: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_AWAIT      ( 0, cm_service_request,        "20: Message is contained in SABM on SAPI
                  0." )

BS_MSG3_SEND       ( 0, authentication_request,    "21" )
BS_MSG3_AWAIT      ( 0, authentication_response,   "22: SRES specifies correct value." )
BS_MSG3_SEND       ( 0, ciphering_mode_command,    "23: SS starts deciphering after sending the
                  message." )

BS_MSG3_AWAIT      ( 0, ciphering_mode_complete,   "24: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )

NOT_IMPLEMENTED    (
NOT_IMPLEMENTED    (
BS_CONFIG_CHANNEL  ( 0, SDCCH, ACK, SAPI_3);
BS_MSG3_AWAIT_BEGIN ( 0, cp_data_sms_submit,        "28: Contains RP-DATA RPDU (SMS SUBMIT
                  TPDU)" )

        BF_SET_VAL      (msg_ref, 1, SILENT)
        BF_SET_VAL      (tp_msg_ref, 1, SILENT)
BS_MSG3_AWAIT_END  ( )
NOT_IMPLEMENTED    (
BS_MSG3_AWAIT_BEGIN ( 0, cp_data_sms_submit,        "29: SS configured not to send CP-ACK" )
                  "30: Retransmitted CP-DATA message within
                  twice TC1M after step 28" )

        BF_SET_VAL      (msg_ref, 1, SILENT)
        BF_SET_VAL      (tp_msg_ref, 1, SILENT)
BS_MSG3_AWAIT_END  ( )
    
```

```

NOT_IMPLEMENTED ( "31: Depending on the maximum number of CP-
DATA retransmissions implemented, step 30
may be repeated. The maximum number of
retransmissions may however not exceed
three." )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_SEND ( 0, channel_release, "32: The main signalling link is released
after a duration of TC1M + 5 seconds after
the last CP-DATA retransmission." )

ISS_DELAY (10000)

COMMAND ("MMI CONFIG AT+CMGS=5678*145*HALLIHALLO")

BS_RACH_AWAIT ( 0, channel_request_111, "33: Establishment cause is Other procedures
which can be completed with an SDDCH"
(NECI=0)" )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK, SAPI_0);
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "34: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_AWAIT ( 0, cm_service_request, "35: Message is contained in SABM. CM
service type set to short message transfer"
)

BS_MSG3_SEND ( 0, authentication_request, "36" )
BS_MSG3_AWAIT ( 0, authentication_response, "37: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "38: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "39: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "40 SS starts ciphering." )
NOT_IMPLEMENTED ( "41: MS establishes SAPI 3" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_3);
BS_MSG3_AWAIT_BEGIN ( 0, cp_data_sms_submit, "43: Contains RP-DATA RPDU (SMS SUBMIT
TPDU)" )

BF_SET_VAL (msg_ref, 2, SILENT)
BF_SET_VAL (tp_msg_ref,2 , SILENT)
BS_MSG3_AWAIT_END ( )
BS_MSG3_SEND ( 0, cp_error_to_ms_mo, "44: Sent within TC1M containing Network
Failure cause." )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_SEND ( 0, channel_release, "45: The main signalling link is released."
)

ISS_DELAY (10000);

COMMAND ("MMI CONFIG KEY_SEQUENCE=03039094117")

BS_RACH_AWAIT ( 0,channel_request_moc, SILENT)
BS_CONFIG_CHANNEL ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0,immediate_assignment, SILENT)

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT ( 0, cm_service_request, SILENT)
BS_MSG3_SEND ( 0, ciphering_mode_command, SILENT)
BS_MSG3_AWAIT ( 0, ciphering_mode_complete, SILENT)

BS_MSG3_AWAIT ( 0, setup_moc, SILENT)

BS_MSG3_SEND ( 0, call_proceeding, SILENT)
BS_MSG3_SEND ( 0, alerting, SILENT)
BS_MSG3_SEND ( 0, assignment_command, SILENT)
BS_CONFIG_CHANNEL ( 0, FACCH, ACK, SAPI_0)
BS_MSG3_AWAIT ( 0, assignment_complete, SILENT)

BS_MSG3_SEND ( 0, connect, SILENT)
BS_MSG3_AWAIT ( 0, connect_acknowledge, SILENT)

NOT_IMPLEMENTED ( "46: A data or speech call is established on
a TCH and the state U10 of call control is
entered." )

NOT_IMPLEMENTED ( "47: The MS is set up to send an SM" )

COMMAND ("MMI CONFIG AT+CMGS=5678*145*HALLIHALLO")

```



```

BS_MSG3_SEND      ( 0, channel_release,      "68: The main signalling link is released
ISS_DELAY (10000)
COMMAND ("MMI CONFIG AT+CMGS=5678*145*HALLIHALLO")

BS_RACH_AWAIT     ( 0, channel_request_111,  "70: Establishment cause is Other procedures
BS_STORE_RACH_PARAMS ( 0, 0)
BS_CONFIG_CHANNEL (0, AGCH, UNACK, SAPI_0);
BS_MSG3_SEND      ( 0, immediate_assignment, "71: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL (0, SDCCH, ACK, SAPI_0);
BS_MSG3_AWAIT     ( 0, cm_service_request,   "72: Message is contained in SABM. CM
BS_MSG3_SEND      ( 0, authentication_request, "73" )
BS_MSG3_AWAIT     ( 0, authentication_response, "74: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "75: SS starts deciphering after sending the
BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "76: Shall be sent enciphered. All following
NOT_IMPLEMENTED   (                               "77: SS starts ciphering." )
NOT_IMPLEMENTED   (                               "62: MS establishes SAPI 3" )
BS_CONFIG_CHANNEL (0, SDCCH, ACK, SAPI_3);
BS_MSG3_AWAIT_BEGIN ( 0, cp_data_sms_submit, "80: Contains RP-DATA RPDU (SMS SUBMIT
        BF_SET_VAL      (msg_ref, 5, SILENT)
        BF_SET_VAL      (tp_msg_ref,5 , SILENT)
BS_MSG3_AWAIT_END  ( )
NOT_IMPLEMENTED   (                               "81: The SS sends an SM to the MS triggered
BS_MSG3_SEND      ( 0, cp_data_sms_deliver,  "82: Contains RP-DATA RPDU (SMS DELIVER
NOT_IMPLEMENTED   (                               "83: !!!!!!! The MS shall correctly receive
BS_MSG3_AWAIT     ( 0, cp_ack_to_bs,        SILENT)
BS_MSG3_AWAIT     ( 0, cp_data_rp_ack_to_bs, SILENT)
BS_MSG3_SEND      ( 0, cp_ack_to_ms_mo,     SILENT)
BS_MSG3_SEND      ( 0, cp_ack_to_ms,       SILENT)
BS_CONFIG_CHANNEL (0, SDCCH, ACK, SAPI_0);
BS_MSG3_SEND      ( 0, channel_release,     SILENT)
ISS_DELAY (10000)
COMMAND ("MMI CONFIG AT+CMGS=5678*145*HALLIHALLO")

BS_RACH_AWAIT     ( 0, channel_request_111,  "84: Establishment cause is Other procedures
BS_STORE_RACH_PARAMS ( 0, 0)
BS_CONFIG_CHANNEL (0, AGCH, UNACK, SAPI_0);
BS_MSG3_SEND      ( 0, immediate_assignment, "85: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL (0, SDCCH, ACK, SAPI_0);
BS_MSG3_AWAIT     ( 0, cm_service_request,   "86: Message is contained in SABM. CM
BS_MSG3_SEND      ( 0, cm_service_reject,    "87: Reject cause set to Service Option not
NOT_IMPLEMENTED   (                               "88: The MS shall not establish SAPI-3" )
BS_MSG3_SEND      ( 0, channel_release,     "89: Sent 5 seconds after CM SERVICE REJ" )

```

History: 26.01.98

MS

Initial

4.2.3 MSMS003: Test of memory full condition and memory available notification (34.2.3)

Description: 1. To verify that the MS sends the correct acknowledgement when its memory in the SIM becomes full.
 2. To verify that the MS sends the correct acknowledgement when its memory in the ME and the SIM becomes full, and sets the "memory exceeded" notification flag in the SIM. 3. To verify that the MS performs the "memory available" procedure when its message store becomes available for receiving short messages, and only at this moment.

Preamble: MSMS000

Script:

```

COMMAND          ( "MMI CONFIG AT+CPMS=1" )
COMMAND          ( "MMI CONFIG AT+CNMI=1" )
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND     ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT   ( 0, channel_request,      "2: Establishment cause is Answer to paging"
                  )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND     ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT   ( 0, paging_response,      "4: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "5" )
BS_MSG3_AWAIT   ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
                  message." )
BS_MSG3_AWAIT   ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )
NOT_IMPLEMENTED ( "9: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND     ( 0, sabm_(sapi=3),        "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT   ( 0, ua_(sapi=3),        "11: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND     ( 0, cp_data_sms_deliver, "12: Contains RP-DATA RPDU (SMS DELIVER
                  TPDU)" )
NOT_IMPLEMENTED ( "13: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT   ( 0, cp_ack_to_bs,        "14" )
NOT_IMPLEMENTED ( "15: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT   ( 0, cp_data_rp_ack_to_bs, "16: Contains RP-ACK RPDU" )
BS_MSG3_SEND     ( 0, cp_ack_to_ms,        "17: Within TC1M after step 16, no further
                  CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND     ( 0, channel_release,     "18: The main signalling link is released."
                  )

ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND     ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT   ( 0, channel_request,      "2: Establishment cause is Answer to paging"
                  )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND     ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT   ( 0, paging_response,      "4: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "5" )
BS_MSG3_AWAIT   ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
                  message." )
BS_MSG3_AWAIT   ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )
NOT_IMPLEMENTED ( "9: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND     ( 0, sabm_(sapi=3),        "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT   ( 0, ua_(sapi=3),        "11: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND     ( 0, cp_data_sms_deliver, "12: Contains RP-DATA RPDU (SMS DELIVER
                  TPDU)" )
NOT_IMPLEMENTED ( "13: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT   ( 0, cp_ack_to_bs,        "14" )
    
```

```

NOT_IMPLEMENTED ( "15: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_data_rp_ack_to_bs, "16: Contains RP-ACK RPDU" )
BS_MSG3_SEND ( 0, cp_ack_to_ms, "17: Within TC1M after step 16, no further
CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "18: The main signalling link is released."
)

ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT ( 0, channel_request, "2: Establishment cause is Answer to paging"
)

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "5" )
BS_MSG3_AWAIT ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "9: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND ( 0, sabm_(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT ( 0, ua_(sapi=3), "11: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND ( 0, cp_data_sms_deliver, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )
NOT_IMPLEMENTED ( "13: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT ( 0, cp_ack_to_bs, "14" )
NOT_IMPLEMENTED ( "15: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_data_rp_ack_to_bs, "16: Contains RP-ACK RPDU" )
BS_MSG3_SEND ( 0, cp_ack_to_ms, "17: Within TC1M after step 16, no further
CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "18: The main signalling link is released."
)

ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT ( 0, channel_request, "2: Establishment cause is Answer to paging"
)

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "5" )
BS_MSG3_AWAIT ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "9: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND ( 0, sabm_(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT ( 0, ua_(sapi=3), "11: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND ( 0, cp_data_sms_deliver, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )
NOT_IMPLEMENTED ( "13: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT ( 0, cp_ack_to_bs, "14" )
NOT_IMPLEMENTED ( "15: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_data_rp_error_111_to_bs, "16: Contains RP-ACK RPDU" )
BS_MSG3_SEND ( 0, cp_ack_to_ms, "17: Within TC1M after step 16, no further
CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "18: The main signalling link is released."
)

```

```

ISS_DELAY (10000)

BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "19" )
BS_RACH_AWAIT    ( 0, channel_request,      "20: Establishment cause is Answer to
                                     paging" )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND      ( 0, immediate_assignment, "21: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT    ( 0, paging_response,      "22: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "23" )
BS_MSG3_AWAIT    ( 0, authentication_response, "24: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "25: SS starts deciphering after sending the
                                     message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "26: Shall be sent enciphered. All following
                                     messages shall be sent enciphered." )
NOT_IMPLEMENTED  (                                     "27: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND      ( 0, sabm_(sapi=3),        "28: SS establishes SAPI 3" )
BS_MSG3_AWAIT    ( 0, ua_(sapi=3),          "29: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND      ( 0, cp_data_sms_deliver_class1, "30: Contains RP-DATA RPDU (SMS DELIVER
                                     TPDU)" )
NOT_IMPLEMENTED  (                                     "31: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT    ( 0, cp_ack_to_bs,          "32" )
NOT_IMPLEMENTED  (                                     "33: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT    ( 0, cp_data_rp_ack_to_bs,  "34: Contains RP-ACK RPDU" )
BS_MSG3_SEND      ( 0, cp_ack_to_ms,        "35: Within TCIM after step 16, no further
                                     CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND      ( 0, channel_release,      "36: The main signalling link is released."
                                     )

ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "19" )
BS_RACH_AWAIT    ( 0, channel_request,      "20: Establishment cause is Answer to
                                     paging" )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND      ( 0, immediate_assignment, "21: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT    ( 0, paging_response,      "22: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "23" )
BS_MSG3_AWAIT    ( 0, authentication_response, "24: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "25: SS starts deciphering after sending the
                                     message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "26: Shall be sent enciphered. All following
                                     messages shall be sent enciphered." )
NOT_IMPLEMENTED  (                                     "27: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND      ( 0, sabm_(sapi=3),        "28: SS establishes SAPI 3" )
BS_MSG3_AWAIT    ( 0, ua_(sapi=3),          "29: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND      ( 0, cp_data_sms_deliver_class1, "30: Contains RP-DATA RPDU (SMS DELIVER
                                     TPDU)" )
NOT_IMPLEMENTED  (                                     "31: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT    ( 0, cp_ack_to_bs,          "32" )
NOT_IMPLEMENTED  (                                     "33: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT    ( 0, cp_data_rp_ack_to_bs,  "34: Contains RP-ACK RPDU" )
BS_MSG3_SEND      ( 0, cp_ack_to_ms,        "35: Within TCIM after step 16, no further
                                     CP-DATA messages" )

BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND      ( 0, channel_release,      "36: The main signalling link is released."
                                     )

ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "19" )
BS_RACH_AWAIT    ( 0, channel_request,      "20: Establishment cause is Answer to
                                     paging" )

```

```

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "21: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT ( 0, paging_response, "22: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "23" )
BS_MSG3_AWAIT ( 0, authentication_response, "24: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "25: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "26: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED ( "27: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND ( 0, sabm_(sapi=3), "28: SS establishes SAPI 3" )
BS_MSG3_AWAIT ( 0, ua_(sapi=3), "29: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND ( 0, cp_data_sms_deliver_class1, "30: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )
NOT_IMPLEMENTED ( "31: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT ( 0, cp_ack_to_bs, "32" )
NOT_IMPLEMENTED ( "33: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_data_rp_error_22_to_bs, "34: Contains RP-ACK RPDU" )
BS_MSG3_SEND ( 0, cp_ack_to_ms, "35: Within TC1M after step 16, no further
CP-DATA messages" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "36: The main signalling link is released."
)
ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND ( 0, paging_request_type_1, "37" )
BS_RACH_AWAIT ( 0, channel_request, "38: Establishment cause is Answer to
paging" )
BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "39: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT ( 0, paging_response, "40: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "41" )
BS_MSG3_AWAIT ( 0, authentication_response, "42: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "43: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "44: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED ( "45: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND ( 0, sabm_(sapi=3), "46: SS establishes SAPI 3" )
BS_MSG3_AWAIT ( 0, ua_(sapi=3), "47: MS shall respond to SABM in step 10" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND ( 0, cp_data_sms_deliver_def, "48: Contains RP-DATA RPDU (SMS DELIVER
TPDU)" )
NOT_IMPLEMENTED ( "49: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT ( 0, cp_ack_to_bs, "50" )
NOT_IMPLEMENTED ( "51: Waits max 60 seconds for RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_data_rp_error_22_to_bs, "52: Contains RP-ACK RPDU" )
BS_MSG3_SEND ( 0, cp_ack_to_ms, "53: Within TC1M after step 16, no further
CP-DATA messages" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "54: The main signalling link is released."
)
ISS_DELAY (10000)
NOT_IMPLEMENTED ( "55: Delete one message from SIM")
COMMAND ( "MMI CONFIG AT+CPMS=3" )
COMMAND ( "MMI CONFIG AT+CMGD=1" )
ISS_DELAY (2000)
BS_RACH_AWAIT ( 0, channel_request_111, "57: Establishment cause is Other procedures
which can be completed with an SDDCH" NECI
= 0" )
BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)

```

```

BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "58: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_AWAIT ( 0, cm_service_request, "59: Message is contained in SABM on SAPI 0.
CM service type set to Short message
transfer" )

BS_MSG3_SEND ( 0, cm_service_accept, "60" )
NOT_IMPLEMENTED ( "61: MS establishes SAPI 3" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_3);
BS_MSG3_AWAIT ( 0, cp_data_rp_smma_to_bs, "63: Contains RP-DATA RPDU (SMS SUBMIT
TPDU)" )

BS_MSG3_SEND ( 0, cp_ack_to_ms_mo, "64: Sent within TC1M after step 11" )
BS_MSG3_SEND ( 0, cp_data_rp_ack_to_ms_mo, "65: Contains RP-ACK RPDU" )
BS_MSG3_AWAIT ( 0, cp_ack_to_bs_mo, "66" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0);
BS_MSG3_SEND ( 0, channel_release, "67: The main signalling link is released."
)

ISS_DELAY (10000)
NOT_IMPLEMENTED ( "68: Delete one message from SIM" )
COMMAND ( "MMI CONFIG AT+CMGD=1" )
BS_RACH_EXPECT_TIMEOUT (1, 10000)
    
```

History: 26.01.98 MS Initial

4.2.4 MSMS004: Test of the status report capabilities and of SMS-COMMAND (34.2.4)

Description: 1) To verify that the MS is able to accept a SMS-STATUS-REPORT TPDU. 2) To verify that the MS is able to use the SMS-COMMAND functionality correctly and sends an SMS-COMMAND TPDU with the correct TP-Message-Reference.

Preamble: MSMS000

Script:

```

BS_RACH_AWAIT ( 0, channel_request, "1: Establishment cause is Other procedures
which can be completed with an SDCCH
(NECI=0)" )

BS_MSG3_SEND ( 0, immediate_assignment, "2: SS assigns an SDCCH" )
BS_MSG3_AWAIT ( 0, cm_service_request, "3: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "4" )
BS_MSG3_AWAIT ( 0, authentication_response, "5: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "6: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "7: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "8: SS starts ciphering." )
BS_MSG3_AWAIT ( 0, sabm(sapi=3), "9: MS establishes SAPI 3" )
BS_MSG3_SEND ( 0, ua(sapi=3), "10" )
BS_MSG3_AWAIT ( 0, cp-data, "11: Contains RP-DATA RPDU (SMS SUBMIT
TPDU)" )

BS_MSG3_SEND ( 0, cp-ack, "12: Sent within TCIM after step 11" )
BS_MSG3_SEND ( 0, cp-data, "13: Contains RP-ACK RPDU" )
NOT_IMPLEMENTED ( "14: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT ( 0, cp-ack, "15" )
BS_MSG3_SEND ( 0, channel_release, "16: The main signalling link is released."
)

BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND ( 0, paging_request_type_1, "17" )
BS_RACH_AWAIT ( 0, channel_request, "18: Establishment cause is Answer to
paging" )

BS_MSG3_SEND ( 0, immediate_assignment, "19: SS assigns an SDCCH" )
BS_MSG3_AWAIT ( 0, paging_response, "20: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "22" )
BS_MSG3_AWAIT ( 0, authentication_response, "23: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "24: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "25: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "26: SS starts ciphering." )
BS_MSG3_SEND ( 0, sabm(sapi=3), "27: SS establishes SAPI 3" )
BS_MSG3_AWAIT ( 0, ua(sapi=3), "28" )
BS_MSG3_SEND ( 0, cp-data, "29: Contains RP-DATA RPDU (SMS-STATUS-
REPORT TPDU)" )

BS_MSG3_AWAIT ( 0, cp-ack, "30" )
BS_MSG3_AWAIT ( 0, cp-data, "31: Contains RP-ACK RPDU" )
BS_MSG3_SEND ( 0, cp-ack, "32" )
BS_MSG3_SEND ( 0, channel_release, "33" )
NOT_IMPLEMENTED ( "34: The MS is made to send an SMS-COMMAND
message enquiring about the previously
submitted SM" )

BS_RACH_AWAIT ( 0, channel_request, "35: Establishment cause Other services
which can be completed with an SDCCH." )

BS_MSG3_SEND ( 0, immediate_assignment, "36: SS assigns an SDCCH" )
BS_MSG3_AWAIT ( 0, cm_service_request, "37: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "38" )
BS_MSG3_AWAIT ( 0, authentication_response, "39: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "40: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "41: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "42: SS starts ciphering." )
BS_MSG3_AWAIT ( 0, sabm(sapi=3), "43: MS establishes SAPI 3" )
BS_MSG3_SEND ( 0, ua(sapi=3), "44" )
BS_MSG3_AWAIT ( 0, cp-data, "45: Contains RP-DATA RPDU (SMS-COMMAND
TPDU) which shall contain the correct TP-
MR" )

BS_MSG3_SEND ( 0, cp-ack, "46" )

```

```

BS_MSG3_SEND      ( 0, cp-data,          "47: Contains RP-ACK RPDU" )
BS_MSG3_AWAIT     ( 0, cp-ack,          "48" )
BS_MSG3_SEND      ( 0, channel_release,  "49" )
NOT_IMPLEMENTED   ( 0, the_ms_is_made_to_send_an_sms-command "50: message requiring to
                  delete the previously submitted SM." )
BS_RACH_AWAIT     ( 0, channel_request,  "51: Establishment cause Other services
                  which can be completed with an SDCCH." )
BS_MSG3_SEND      ( 0, immediate_assignment, "52: SS assigns an SDCCH" )
BS_MSG3_AWAIT     ( 0, cm_service_request, "53: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "54" )
BS_MSG3_AWAIT     ( 0, authentication_response, "55: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "56: SS starts deciphering after sending the
                  message." )
BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "57: Shall be sent enciphered. All following
                  messages shall be sent enciphered." )
NOT_IMPLEMENTED   ( "58: SS starts ciphering." )
BS_MSG3_AWAIT     ( 0, sabm_(sapi=3),      "59: MS establishes SAPI 3" )
BS_MSG3_SEND      ( 0, ua_(sapi=3),       "60" )
BS_MSG3_AWAIT     ( 0, cp-data,          "61: Contains RP-DATA RPDU (SMS-COMMAND
                  TPDU) which shall contain the correct TP-
                  MR" )
BS_MSG3_SEND      ( 0, cp-ack,          "62" )
BS_MSG3_SEND      ( 0, cp-data,          "63: Contains RP-ACK RPDU" )
BS_MSG3_AWAIT     ( 0, cp-ack,          "64" )
BS_MSG3_SEND      ( 0, channel_release,  "65" )

```

History: 26.01.98 MS Initial

4.3 Test of message class 0 to 3

4.3.1 MSMS005: Short message class 0 (34.2.5.1)

Description: To verify that the MS will accept and display but not store a class 0 message, and that it will accept and display a class 0 message if its message store is full. NOTE: failure of this test in a mobile could cause it to reject a class 0 message when its SMS memory becomes full. This could lead to unwanted repetitions between the MS and the service centre.

Preamble: MSMS000

Script:

```

BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT    ( 0, channel_request, "2: Establishment cause is Answer to paging"
)
BS_MSG3_SEND      ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_MSG3_AWAIT    ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "5" )
BS_MSG3_AWAIT    ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED  ( "9: SS starts ciphering." )
BS_MSG3_SEND      ( 0, sabm_(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT    ( 0, ua_(sapi=3), "11" )
BS_MSG3_SEND      ( 0, cp-data, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 0 Short Message" )
BS_MSG3_AWAIT    ( 0, cp-ack, "13" )
BS_MSG3_AWAIT    ( 0, cp-data, "14: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp-ack, "15" )
BS_MSG3_SEND      ( 0, channel_release, "16" )
NOT_IMPLEMENTED  ( "17: The content of the short message shall
be displayed by the ME. The MS shall not
store the message. This can be checked by
verifying that it is impossible to retrieve
any short messages from the MS message
store." )
NOT_IMPLEMENTED  ( "18: The MS message store shall be filled
(for example by using the method of 34.2.3)
with Class 1 SMS-DELIVER TPDU." )
BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "19" )
BS_RACH_AWAIT    ( 0, channel_request, "20: Establishment cause is Answer to
paging" )
BS_MSG3_SEND      ( 0, immediate_assignment, "21: SS assigns an SDCCH" )
BS_MSG3_AWAIT    ( 0, paging_response, "22: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "23" )
BS_MSG3_AWAIT    ( 0, authentication_response, "24: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "25: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "26: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED  ( "27: SS starts ciphering." )
BS_MSG3_SEND      ( 0, sabm_(sapi=3), "28: SS establishes SAPI 3" )
BS_MSG3_AWAIT    ( 0, ua_(sapi=3), "29" )
BS_MSG3_SEND      ( 0, cp-data, "30: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 0 Short Message" )
BS_MSG3_AWAIT    ( 0, cp-ack, "31" )
BS_MSG3_AWAIT    ( 0, cp-data, "32: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp-ack, "33" )
BS_MSG3_SEND      ( 0, channel_release, "34" )
NOT_IMPLEMENTED  ( "35: The content of the short message shall
be displayed by the ME." )
    
```

History: 26.01.98 MS Initial

4.3.2 MSMS006: Test of class 1 short messages (34.2.5.2)

Description: This procedure verifies that the MS acts correctly on receiving a class 1 message, i.e. that it stores the message in the ME or SIM and sends an acknowledgement (at RP and CP-Layer).

Preamble: MSMS

Script:

```

BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT     ( 0, channel_request, "2: Establishment cause is Answer to paging"
)
BS_MSG3_SEND      ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_MSG3_AWAIT     ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "5" )
BS_MSG3_AWAIT     ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED   ( "9: SS starts ciphering." )
BS_MSG3_SEND      ( 0, sabm_(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3), "11" )
BS_MSG3_SEND      ( 0, cp-data, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 1 Short Message" )
BS_MSG3_AWAIT     ( 0, cp-ack, "13" )
BS_MSG3_AWAIT     ( 0, cp-data, "14: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp-ack, "15" )
BS_MSG3_SEND      ( 0, channel_release, "16" )
NOT_IMPLEMENTED   ( "17: The short message shall be recalled and
displayed at the MS." )
    
```

History: 26.01.98 MS Initial

4.3.3 MSMS007: Test of class 2 short messages (34.2.5.3)

Description: This procedure verifies that the MS acts correctly on receiving a class 2 message, i.e. that it stores the message correctly in the SIM, and if this is not possible, returns a protocol error message, with the correct error cause, to the network. There are 2 cases: 1) If the MS supports storing of short messages in the SIM and in the ME, and storage in the ME is not full, and the short message cannot be stored in the SIM, the error cause shall be "protocol error, unspecified". 2) If the MS supports storing of short messages in the SIM and not in the ME, and storage in the ME is not full, and the short message cannot be stored in the SIM, the error cause shall be "memory capacity exceeded". NOTE: If the MS supports storing of short messages in the SIM and the ME, and storage in the ME is full, and the short message cannot be stored in the SIM, the error cause shall be "memory capacity exceeded". This case is not tested in this test.

Preamble: MSMS000

Script:

```

BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT ( 0, channel_request, "2: Establishment cause is Answer to paging"
)

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "5" )
BS_MSG3_AWAIT ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED ( "9: SS starts ciphering." )
/*
not implemented
BS_MSG3_SEND ( 0, sabm(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT ( 0, ua(sapi=3), "11" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_3)
BS_MSG3_SEND ( 0, cp_data_sms_deliver, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 2 Short Message" )

BS_MSG3_AWAIT ( 0, cp_ack_to_bs, "13" )
/*NOT_IMPLEMENTED ( "14: The ME shall correctly store the short
message in a free record of EFSMS in the
SIM, i.e. - the ME shall use a free
record - the first byte of the record
shall indicate message received by MS from
network - the TS-Service-Centre-
Address shall be correctly stored - the
TPDU shall be identical to that sent by the
SS - bytes following the TPDU shall be
set to FF" )

NOT_IMPLEMENTED ( "15: The SIM simulator returns the status
response OK (90 00). The SIM simulator
shall indicate if an attempt was made by
the ME to store the short message in the
SIM." )*/

BS_MSG3_AWAIT ( 0, cp_data_rp_ack_to_bs, "16: Contains RP-ACK RPDU." )
BS_MSG3_SEND ( 0, cp_ack_to_ms, "16A" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_SEND ( 0, channel_release, "17" )
ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT ( 0, channel_request, "2: Establishment cause is Answer to paging"
)

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK, SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK, SAPI_0)
BS_MSG3_AWAIT ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND ( 0, authentication_request, "5" )
BS_MSG3_AWAIT ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )

```

```

BS_MSG3_AWAIT      ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
NOT_IMPLEMENTED    (                               messages shall be sent enciphered." )
/*                               "9: SS starts ciphering." )
not implemented
BS_MSG3_SEND      ( 0, sabm_(sapi=3),           "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3),           "11" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND      ( 0, cp_data_sms_deliver,    "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 2 Short Message" )

BS_MSG3_AWAIT     ( 0, cp_ack_to_bs,          "13" )
/*NOT_IMPLEMENTED (                               "14: The ME shall correctly store the short
message in a free record of EFSMS in the
SIM, i.e. - the ME shall use a free
record - the first byte of the record
shall indicate message received by MS from
network - the TS-Service-Centre-
Address shall be correctly stored - the
TPDU shall be identical to that sent by the
SS - bytes following the TPDU shall be
set to FF" )

NOT_IMPLEMENTED   (                               "15: The SIM simulator returns the status
response OK (90 00). The SIM simulator
shall indicate if an attempt was made by
the ME to store the short message in the
SIM." )*/

BS_MSG3_AWAIT     ( 0, cp_data_rp_ack_to_bs,   "16: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp_ack_to_ms,          "16A" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND      ( 0, channel_release,        "17" )
ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1,  "1" )
BS_RACH_AWAIT     ( 0, channel_request,        "2: Establishment cause is Answer to paging"
)

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND      ( 0, immediate_assignment,   "3: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT     ( 0, paging_response,        "4: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "5" )
BS_MSG3_AWAIT     ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
NOT_IMPLEMENTED   (                               messages shall be sent enciphered." )
/*                               "9: SS starts ciphering." )
not implemented
BS_MSG3_SEND      ( 0, sabm_(sapi=3),           "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3),           "11" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND      ( 0, cp_data_sms_deliver,    "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 2 Short Message" )

BS_MSG3_AWAIT     ( 0, cp_ack_to_bs,          "13" )
/*NOT_IMPLEMENTED (                               "14: The ME shall correctly store the short
message in a free record of EFSMS in the
SIM, i.e. - the ME shall use a free
record - the first byte of the record
shall indicate message received by MS from
network - the TS-Service-Centre-
Address shall be correctly stored - the
TPDU shall be identical to that sent by the
SS - bytes following the TPDU shall be
set to FF" )

NOT_IMPLEMENTED   (                               "15: The SIM simulator returns the status
response OK (90 00). The SIM simulator
shall indicate if an attempt was made by
the ME to store the short message in the
SIM." )*/

BS_MSG3_AWAIT     ( 0, cp_data_rp_ack_to_bs,   "16: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp_ack_to_ms,          "16A" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)

```

```

BS_MSG3_SEND      ( 0, channel_release,      "17" )
ISS_DELAY (10000)
BS_CONFIG_CHANNEL ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "18" )
BS_RACH_AWAIT     ( 0, channel_request,      "19: Establishment cause is Answer to
paging" )

BS_CONFIG_CHANNEL ( 0, AGCH, UNACK,SAPI_0)
BS_STORE_RACH_PARAMS ( 0, 0);
BS_MSG3_SEND      ( 0, immediate_assignment, "20: SS assigns an SDCCH" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_AWAIT     ( 0, paging_response,      "21: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "22" )
BS_MSG3_AWAIT     ( 0, authentication_response, "23: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "24: SS starts deciphering after sending the
message." )

BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "25: Shall be sent enciphered. All following
messages shall be sent enciphered." )

NOT_IMPLEMENTED   (
/*
not implemented
BS_MSG3_SEND      ( 0, sabm_(sapi=3),        "27: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3),        "28" )
*/
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_3)
BS_MSG3_SEND      ( 0, cp_data_sms_deliver,  "29: Contains RP-DATA RPDU (SMS DELIVER
TPDU), Class 2 Short Message" )

BS_MSG3_AWAIT     ( 0, cp_ack_to_bs,        "30" )
NOT_IMPLEMENTED   (
/*NOT_IMPLEMENTED (
"31: The ME shall attempt to store the short
message in a free record of EFSMS in the
SIM." )
"32: The SIM simulator returns the status
response memory problem (92 40). The SIM
simulator shall indicate if an attempt was
made by the ME to store the short message
in the SIM." )*/
BS_MSG3_AWAIT     ( 0, cp_data_rp_error_111_to_bs, "33: Contains RP-ERROR RPDU with
error cause protocol error, unspecified" )

BS_MSG3_SEND      ( 0, cp_ack_to_ms,        "33A" )
BS_CONFIG_CHANNEL ( 0, SDCCH, ACK,SAPI_0)
BS_MSG3_SEND      ( 0, channel_release,      "34" )

```

History: 26.01.98 MS Initial

4.4 Short message service point to point

4.4.1 MSMS008: Test of the replace mechanism for SM type 1-7 (34.2.7)

Description: This procedure verifies the correct implementation of the replace mechanism for Replace Short Messages.

Preamble: MSMS000

Script:

```

BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT     ( 0, channel_request, "2: Establishment cause is Answer to paging"
)
BS_MSG3_SEND      ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_MSG3_AWAIT     ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "5" )
BS_MSG3_AWAIT     ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED   ( "9: SS starts ciphering." )
BS_MSG3_SEND      ( 0, sabm_(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3), "11" )
BS_MSG3_SEND      ( 0, cp-data, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU) TP-PID is Replace Short Message Type
n, TP-OA is TPOA1 and RP-OA is RPOA1" )
BS_MSG3_AWAIT     ( 0, cp-ack, "13" )
BS_MSG3_AWAIT     ( 0, cp-data, "14: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp-ack, "14A" )
BS_MSG3_SEND      ( 0, channel_release, "15" )
BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "16" )
BS_RACH_AWAIT     ( 0, channel_request, "17: Establishment cause is Answer to
paging" )
BS_MSG3_SEND      ( 0, immediate_assignment, "18: SS assigns an SDCCH" )
BS_MSG3_AWAIT     ( 0, paging_response, "19: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "20" )
BS_MSG3_AWAIT     ( 0, authentication_response, "21: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "22: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "23: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED   ( "24: SS starts ciphering." )
BS_MSG3_SEND      ( 0, sabm_(sapi=3), "25: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3), "26" )
BS_MSG3_SEND      ( 0, cp-data, "27: Contains RP-DATA RPDU (SMS DELIVER
TPDU) TP-PID is Replace Short Message Type
n, TP-OA is TPOA2 and RP-OA is RPOA1, TP-UD
different from step 12" )
BS_MSG3_AWAIT     ( 0, cp-ack, "28" )
BS_MSG3_AWAIT     ( 0, cp-data, "29: Contains RP-ACK RPDU." )
BS_MSG3_SEND      ( 0, cp-ack, "29A" )
BS_MSG3_SEND      ( 0, channel_release, "30" )
BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "31" )
BS_RACH_AWAIT     ( 0, channel_request, "32: Establishment cause is Answer to
paging" )
BS_MSG3_SEND      ( 0, immediate_assignment, "33: SS assigns an SDCCH" )
BS_MSG3_AWAIT     ( 0, paging_response, "34: Message is contained in SABM." )
BS_MSG3_SEND      ( 0, authentication_request, "35" )
BS_MSG3_AWAIT     ( 0, authentication_response, "36: SRES specifies correct value." )
BS_MSG3_SEND      ( 0, ciphering_mode_command, "37: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT     ( 0, ciphering_mode_complete, "38: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED   ( "39: SS starts ciphering." )
BS_MSG3_SEND      ( 0, sabm_(sapi=3), "40: SS establishes SAPI 3" )
BS_MSG3_AWAIT     ( 0, ua_(sapi=3), "41" )
BS_MSG3_SEND      ( 0, cp-data, "42: Contains RP-DATA RPDU (SMS DELIVER
TPDU) TP-PID is Replace Short Message Type

```

```

n, TP-OA is TPOA2 and RP-OA is RPOA2, TP-UD
different from step 12 and 27" )
BS_MSG3_AWAIT      ( 0, cp-ack,          "43" )
BS_MSG3_AWAIT      ( 0, cp-data,        "44: Contains RP-ACK RPDU." )
BS_MSG3_SEND       ( 0, cp-ack,          "44A" )
BS_MSG3_SEND       ( 0, channel_release, "45" )
BS_CONFIG_CHANNEL  ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND       ( 0, paging_request_type_1, "46" )
BS_RACH_AWAIT      ( 0, channel_request, "47: Establishment cause is Answer to
                    paging" )
BS_MSG3_SEND       ( 0, immediate_assignment, "48: SS assigns an SDCCH" )
BS_MSG3_AWAIT      ( 0, paging_response,  "49: Message is contained in SABM." )
BS_MSG3_SEND       ( 0, authentication_request, "50" )
BS_MSG3_AWAIT      ( 0, authentication_response, "51: SRES specifies correct value." )
BS_MSG3_SEND       ( 0, ciphering_mode_command, "52: SS starts deciphering after sending the
                    message." )
BS_MSG3_AWAIT      ( 0, ciphering_mode_complete, "53: Shall be sent enciphered. All following
                    messages shall be sent enciphered." )
NOT_IMPLEMENTED    (
                    "54: SS starts ciphering." )
BS_MSG3_SEND       ( 0, sabm_(sapi=3),    "55: SS establishes SAPI 3" )
BS_MSG3_AWAIT      ( 0, ua_(sapi=3),     "56" )
BS_MSG3_SEND       ( 0, cp-data,          "57: Contains RP-DATA RPDU (SMS DELIVER
                    TPDU) TP-PID is Replace Short Message Type
                    m, TP-OA is TPOA2 and RP-OA is RPOA2, TP-UD
                    different from step 12, 27 and 42" )
BS_MSG3_AWAIT      ( 0, cp-ack,          "58" )
BS_MSG3_AWAIT      ( 0, cp-data,          "59: Contains RP-ACK RPDU." )
BS_MSG3_SEND       ( 0, cp-ack,          "59A" )
BS_MSG3_SEND       ( 0, channel_release,  "60" )
BS_CONFIG_CHANNEL  ( 0, PCH, UNACK,SAPI_0)
BS_MSG3_SEND       ( 0, paging_request_type_1, "61" )
BS_RACH_AWAIT      ( 0, channel_request,  "62: Establishment cause is Answer to
                    paging" )
BS_MSG3_SEND       ( 0, immediate_assignment, "63: SS assigns an SDCCH" )
BS_MSG3_AWAIT      ( 0, paging_response,  "64: Message is contained in SABM." )
BS_MSG3_SEND       ( 0, authentication_request, "65" )
BS_MSG3_AWAIT      ( 0, authentication_response, "66: SRES specifies correct value." )
BS_MSG3_SEND       ( 0, ciphering_mode_command, "67: SS starts deciphering after sending the
                    message." )
BS_MSG3_AWAIT      ( 0, ciphering_mode_complete, "68: Shall be sent enciphered. All following
                    messages shall be sent enciphered." )
NOT_IMPLEMENTED    (
                    "69: SS starts ciphering." )
BS_MSG3_SEND       ( 0, sabm_(sapi=3),    "70: SS establishes SAPI 3" )
BS_MSG3_AWAIT      ( 0, ua_(sapi=3),     "71" )
BS_MSG3_SEND       ( 0, cp-data,          "72: Contains RP-DATA RPDU (SMS DELIVER
                    TPDU) TP-PID is Replace Short Message Type
                    m, TP-OA is TPOA2 and RP-OA is RPOA2, TP-UD
                    different from step 57" )
BS_MSG3_AWAIT      ( 0, cp-ack,          "73" )
BS_MSG3_AWAIT      ( 0, cp-data,          "74: Contains RP-ACK RPDU." )
BS_MSG3_SEND       ( 0, cp-ack,          "74A" )
BS_MSG3_SEND       ( 0, channel_release,  "75" )
NOT_IMPLEMENTED    (
                    "76: Prompts the operator to display the
                    Short Messages stored in the MS. Only the
                    Short Messages delivered in step 12, 27, 42
                    and 72 shall be retrievable and displayed"
                    )

```

History: 26.01.98

MS

Initial

4.4.2 MSMS009: Test of the reply path scheme (34.2.8)

Description: This procedure verifies that the MS is able to send a Reply Short Message back to the correct originating SME even if in the meantime it receives another Short Message.

Preamble: MSMS000

Script:

```

BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND      ( 0, paging_request_type_1, "1" )
BS_RACH_AWAIT    ( 0, channel_request, "2: Establishment cause is Answer to paging"
)
BS_MSG3_SEND     ( 0, immediate_assignment, "3: SS assigns an SDCCH" )
BS_MSG3_AWAIT   ( 0, paging_response, "4: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "5" )
BS_MSG3_AWAIT   ( 0, authentication_response, "6: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "7: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT   ( 0, ciphering_mode_complete, "8: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED (
BS_MSG3_SEND     ( 0, sabm_(sapi=3), "9: SS starts ciphering." )
BS_MSG3_AWAIT   ( 0, ua_(sapi=3), "10: SS establishes SAPI 3" )
BS_MSG3_SEND     ( 0, cp-data, "11" )
BS_MSG3_SEND     ( 0, cp-data, "12: Contains RP-DATA RPDU (SMS DELIVER
TPDU) TP-RP set to 1" )
BS_MSG3_AWAIT   ( 0, cp-ack, "13: Sent within TC1M after step 12" )
BS_MSG3_AWAIT   ( 0, cp-data, "14: Contains RP-ACK RPDU." )
BS_MSG3_SEND     ( 0, cp-ack, "14A" )
BS_MSG3_SEND     ( 0, channel_release, "15" )
BS_CONFIG_CHANNEL ( 0, PCH, UNACK, SAPI_0)
BS_MSG3_SEND     ( 0, paging_request_type_1, "16" )
BS_RACH_AWAIT    ( 0, channel_request, "17: Establishment cause is Answer to
paging" )
BS_MSG3_SEND     ( 0, immediate_assignment, "18: SS assigns an SDCCH" )
BS_MSG3_AWAIT   ( 0, paging_response, "19: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "20" )
BS_MSG3_AWAIT   ( 0, authentication_response, "21: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "22: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT   ( 0, ciphering_mode_complete, "23: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED (
BS_MSG3_SEND     ( 0, sabm_(sapi=3), "24: SS starts ciphering." )
BS_MSG3_AWAIT   ( 0, ua_(sapi=3), "25: SS establishes SAPI 3" )
BS_MSG3_SEND     ( 0, cp-data, "26" )
BS_MSG3_SEND     ( 0, cp-data, "27: Contains RP-DATA RPDU (SMS DELIVER
TPDU) TP-OA, RP-OA and TP-UD different from
step 12" )
BS_MSG3_AWAIT   ( 0, cp-ack, "28: Sent within TC1M after step 12" )
BS_MSG3_AWAIT   ( 0, cp-data, "29: Contains RP-ACK RPDU." )
BS_MSG3_SEND     ( 0, cp-ack, "29A" )
BS_MSG3_SEND     ( 0, channel_release, "30" )
NOT_IMPLEMENTED (
BS_RACH_AWAIT    ( 0, channel_request, "31: One of the two Short Messages is
displayed and the Reply Short Message is
submitted." )
BS_RACH_AWAIT    ( 0, channel_request, "32" )
BS_MSG3_SEND     ( 0, immediate_assignment, "33: SS assigns an SDCCH" )
BS_MSG3_AWAIT   ( 0, cm_service_request, "34: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "35" )
BS_MSG3_AWAIT   ( 0, authentication_response, "36: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "37: SS starts deciphering after sending the
message." )
BS_MSG3_AWAIT   ( 0, ciphering_mode_complete, "38: Shall be sent enciphered. All following
messages shall be sent enciphered." )
NOT_IMPLEMENTED (
BS_MSG3_AWAIT   ( 0, sabm_(sapi=3), "39: SS starts ciphering." )
BS_MSG3_SEND     ( 0, ua_(sapi=3), "40: MS establishes SAPI 3" )
BS_MSG3_AWAIT   ( 0, cp-data, "41" )
BS_MSG3_AWAIT   ( 0, cp-data, "42: Contains RP-DATA RPDU (SMS SUBMIT TPDU)
RP-DA = RP-OA corresponding to the message
displayed TP-DA = TP-OA corresponding to
the message displayed" )
BS_MSG3_SEND     ( 0, cp-ack, "43: Sent within TC1M after step 42" )
BS_MSG3_SEND     ( 0, cp-data, "44: Contains RP-ACK RPDU" )
NOT_IMPLEMENTED (
BS_MSG3_AWAIT   ( 0, cp-ack, "45: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT   ( 0, cp-ack, "46" )
    
```

```

BS_MSG3_SEND      ( 0, channel_release,      "47: The main signalling link is released.
                  " )
NOT_IMPLEMENTED   (                          "48: The other Short Message is displayed
                  and the Reply Short Message is submitted."
                  )
BS_RACH_AWAIT    ( 0, channel_request,      "49" )
BS_MSG3_SEND     ( 0, immediate_assignment, "50: SS assigns an SDCCH" )
BS_MSG3_AWAIT    ( 0, cm_service_request,  "51: Message is contained in SABM." )
BS_MSG3_SEND     ( 0, authentication_request, "52" )
BS_MSG3_AWAIT    ( 0, authentication_response, "53: SRES specifies correct value." )
BS_MSG3_SEND     ( 0, ciphering_mode_command, "54: SS starts deciphering after sending the
                  message." )
BS_MSG3_AWAIT    ( 0, ciphering_mode_complete, "55: Shall be sent enciphered. All
                  following messages shall be sent
                  enciphered." )
NOT_IMPLEMENTED   (                          "56: SS starts ciphering." )
BS_MSG3_AWAIT    ( 0, sabm(sapi=3),        "57: MS establishes SAPI 3" )
BS_MSG3_SEND     ( 0, ua(sapi=3),         "58" )
BS_MSG3_AWAIT    ( 0, cp-data,            "59: Contains RP-DATA RPDU (SMS SUBMIT TPDU)
                  RP-DA = RP-OA corresponding to the Message
                  displayed TP-DA = TP-OA corresponding to
                  the message displayed" )
BS_MSG3_SEND     ( 0, cp-ack,              "60: Sent within TC1M after step 59" )
BS_MSG3_SEND     ( 0, cp-data,            "61: Contains RP-ACK RPDU" )
NOT_IMPLEMENTED   (                          "62: Waits max 25 seconds for CP-ACK" )
BS_MSG3_AWAIT    ( 0, cp-ack,              "63" )
BS_MSG3_SEND     ( 0, channel_release,      "64: The main signalling link is released."
                  )
    
```

History:

26.01.98

MS

Initial

4.4.3 MSMS010: Cell Broadcast (34.3)

Description: This procedure checks SMS cell broadcast

Preamble: none

Script:

```

ISS_INIT                ( 1 )

BS_SET_SYS_INFO        ( 0, system_information_type_1 )
BS_SET_SYS_INFO        ( 0, system_information_type_2 )
BS_SET_SYS_INFO        ( 0, system_information_type_3 )
BS_SET_SYS_INFO        ( 0, system_information_type_4_CB )
BS_SET_SYS_INFO_SACCH( 0, system_information_type_5 )
BS_SET_SYS_INFO_SACCH( 0, system_information_type_6 )

BS_SET_SCH              ( 0,BSIC , RFN                )
BS_SET_ARFCN            ( 0,ARFCN_BCCH                )
BS_SET_POWER            ( 0,-73                        )
BS_ON_OFF               ( 0,TRUE                       )

COMMAND                 ( "MMI CONFIG KEY_SEQUENCE=<#*91*1#>" )

ISS_DELAY               ( 20000 ) /* wait until the MS has detected the cell */

BS_CONFIG_CHANNEL       ( 0, CBCH, UNACK,SAPI_0)
BS_MSG3_SEND            ( 0, sms_cbch_message,         "1: SS sends a CBCH message" )

ISS_DELAY               ( 10000 )

COMMAND                 ( "MMI CONFIG AT+CPMS=0" )
COMMAND                 ( "MMI CONFIG AT+CMGR=1" )

ISS_DELAY               ( 5000 )

COMMAND                 ( "MMI CONFIG AT+CMGD=1" )

ISS_DELAY               ( 5000 )
    
```

History: 26.01.98 MS Initial

4.5 Additional SMS Testcases

4.5.1 MSMS500: Write and Read of a mobile originated SMS

Description: A mobile originated SMS is stored on the SIM card and read again.

Preamble: none

Script:

```
ISS_INIT          ( 1 )

BS_SET_SYS_INFO ( 0, system_information_type_1 )
BS_SET_SYS_INFO ( 0, system_information_type_2 )
BS_SET_SYS_INFO ( 0, system_information_type_3 )
BS_SET_SYS_INFO ( 0, system_information_type_4 )
BS_SET_SYS_INFO_SACCH( 0, system_information_type_5 )
BS_SET_SYS_INFO_SACCH( 0, system_information_type_6 )

BS_SET_SCH        ( 0,BSIC , RFN                )
BS_SET_ARFCN      ( 0,ARFCN_BCCH                )
BS_SET_POWER      ( 0,-73                        )
BS_ON_OFF         ( 0,TRUE                       )

COMMAND           ( "MMI CONFIG KEY_SEQUENCE=<#*91*1#>" )

ISS_DELAY         ( 20000 ) /* wait until the MS has detected the cell */

AT_SEND ("AT+CMGW=\"654321\",,\r\n",SILENT);    /* start entering MO-SMS */
AT_RECEIVE ("\n\r>", SILENT);
AT_SEND ("ABCDEFGHIJKLMNPOQRSTUVWXYZ\r\n",SILENT);
AT_RECEIVE ("+CMGW: 1", SILENT);
AT_SEND ("AT+CMGR=1", SILENT);    /* read MO-SMS */
AT_RECEIVE ("+CMGR: STO UNSENT", SILENT);
```

History: 26.01.98 MS Initial

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/>>