I

Test Suite Overview

Test Suite Structure

Suite Name : Rel4_GSM_MS_EFR_420
Standards Ref : ETS 300 557
PICS Ref :
PIXIT Ref :
Test Method(s) : Distributed Single Layer Test Method
Comments : EFR ATS 51.010−3 Version 4.2.0 after GERAN #5 (GERAN WG4 #4)

Test Group Reference	Selection Ref	Test Group Objective	Page Nr
EFR/	SelExpr_1100	Enhanced Full Rate signalling	517

Detailed Comments :

Test Case Index

Test Group Reference	Test Case Id	Selection Ref	Description	Page Nr
EFR/	TC_26_12_1	SelExpr_0001	EFR signalling/ test of the channel mode modify procedure	517
EFR/	TC_26_12_2_1_1	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	520
EFR/	TC_26_12_2_1_2	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	523
EFR/	TC_26_12_2_1_3	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	526
EFR/	TC_26_12_2_1_4	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	529
EFR/	TC_26_12_2_1_5	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	532
EFR/	TC_26_12_2_1_6	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	534
EFR/	TC_26_12_2_1_7	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	537
EFR/	TC_26_12_2_1_8	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	540
EFR/	TC_26_12_2_1_9	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	542
EFR/	TC_26_12_2_1_10	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	544
EFR/	TC_26_12_2_1_11	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	547
EFR/	TC_26_12_2_1_12	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	550
EFR/	TC_26_12_2_1_13	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	552
EFR/	TC_26_12_2_1_14	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	554
EFR/	TC_26_12_2_1_15	SelExpr_0001	EFR signalling/ Handover/ active call/ successful case	557
EFR/	TC_26_12_2_2_1	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	560
EFR/	TC_26_12_2_2_2	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	563

Continued on next page

Continued from previous page

Test Case Index

Test Group Reference	Test Case Id	Selection Ref	Description	Page Nr
EFR/	TC_26_12_2_2_3	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	566
EFR/	TC_26_12_2_2_4	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	569
EFR/	TC_26_12_2_2_5	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	572
EFR/	TC_26_12_2_2_6	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	575
EFR/	TC_26_12_2_2_7	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	578
EFR/	TC_26_12_2_2_8	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	581
EFR/	TC_26_12_2_2_9	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	584
EFR/	TC_26_12_2_2_10	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	587
EFR/	TC_26_12_2_2_11	SelExpr_0001	EFR signalling/ Handover/ successful/ call under establishment/ non−synchronized	590
EFR/	TC_26_12_3	SelExpr_1101	EFR Signalling/ Structured procedures/ MS originated call/ late assignment	593
EFR/	TC_26_12_4	SelExpr_1102	Structured procedures/ MS terminated call/ early assignment	597
EFR/	TC_26_12_5	SelExpr_0001	Structured procedures/ emergency call	601

Detailed Comments :

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
management/	Config_CBMS_A_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH, SDCCH4 and CBCH onto the physical channel which represents cell A.	604
management/	Config_FACCHF_A_1	To set one physical channel used as TCHF_ACCH's of cell A.	605
management/	Config_FACCHF_A_2	To set one physical channel used as TCHF_ACCH's for instance 2 of cell A.	606
management/	Config_FACCHF_B_1	To set one physical channel used as TCHF_ACCH's of cell B.	607
management/	Config_FACCHH_A_1	To set one physical channel used as TCHH_ACCH's of cell A.	608
management/	Config_FACCHH_A_2	To set one physical channel used as TCHH_ACCH's of cell A.	609
management/	Config_FACCHH_B_1	To set one physical channel used as TCHH_ACCH's of cell B.	610
management/	Config_BCCH_CCCH_A_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell A.	611
management/	Config_BCCH_CCCH_A_ho_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell A with controlable td and fn.	612
management/	Config_BCCH_CCCH_A_2	To set a physical channel and map the second BCCH, CCCH onto the physical channel which represents cell A.	613
management/	Config_BCCH_CCCH_A_3	To set a physical channel and map the third BCCH, CCCH onto the physical channel which represents cell A.	614
management/	Config_BCCH_CCCH_A_4	To set a physical channel and map the fourth BCCH, CCCH onto the physical channel which represents cell A.	615
management/	Config_BCCH_CCCH_B_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell B.	616
management/	Config_BCCH_CCCH_B_ho_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell B with controlable td and fn.	617
management/	Config_SDCCH4_A_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.	618
management/	Config_SDCCH4_A_ho_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A with controlable ta and fn.	619

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
management/	Config_SDCCH4_B_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.	620
management/	Config_SDCCH4_B_ho_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B with controlable ta and fn.	621
management/	Config_SDCCH4_C_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell C	622
management/	Config_SDCCH8_A_1	To set one physical channel used as SDCCH8 channel for instance 1 of cell A.	623
management/	CombinedBCCH_A_CBMS	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.	624
management/	CombinedBCCH_A	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.	624
management/	CombinedBCCH_A_ho	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.	625
management/	CombinedBCCH_B	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.	625
management/	CombinedBCCH_B_ho	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.	626
management/	CombinedBCCH_C	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell C.	626
management/	NonCombinedBCCH_A	To set one physical channel used as FCHH_SCH_BCCH_CCCH for cell A.	627
management/	NonCombinedBCCH_A_2	To set a physical channel and map the second BCCH, CCCH onto the physical channel which represents cell A.	627
management/	NonCombinedBCCH_A_3	To set a physical channel and map the third BCCH, CCCH onto the physical channel which represents cell A.	628
management/	NonCombinedBCCH_A_4	To set a physical channel and map the fourth BCCH, CCCH onto the physical channel which represents cell A.	628

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
management/	NonCombinedBCCH_A_ho	To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell A for RR testing.	629
management/	NonCombinedBCCH_B	To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell B for RR testing.	629
management/	NonCombinedBCCH_B_ho	To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell B for RR testing.	630
management/	FullRateCh_A_1	To set one physical channel used as TCHF_ACCH's for instance 1 of cell A.	630
management/	FullRateCh_A_1_nociph	To set one physical channel used as TCHF_ACCH's for instance 1 of cell A.	631
management/	FullRateCh_A_2	To set one physical channel used as TCHF_ACCH's for instance 2 of cell A for TC_26_6_13_1.	631
management/	FullRateCh_A_2_nociph	To set one physical channel used as TCHF_ACCH's for instance 2 of cell A for TC_26_6_13_1.	632
management/	FullRateCh_B_1	To set one physical channel used as TCHF_ACCH's for instance 1 of cell B.	632
management/	FullRateCh_B_1_nociph	To set one physical channel used as TCHF_ACCH's for instance 1 of cell B.	633
management/	HalfRateCh_A_1	To set one physical channel used as TCHH_ACCH's for instance 1 of cell A.	633
management/	HalfRateCh_A_1_nociph	To set one physical channel used as TCHH_ACCH's for instance 1 of cell A.	634
management/	HalfRateCh_A_2	To set one physical channel used as TCHH_ACCH's for instance 2 of cell A for TC_26_6_13_1.	634
management/	HalfRateCh_A_2_nociph	To set one physical channel used as TCHH_ACCH's for instance 2 of cell A for TC_26_6_13_1.	635
management/	HalfRateCh_B_1	To set one physical channel used as TCHH_ACCH's for instance 1 of cell B	635
management/	HalfRateCh_B_1_nociph	To set one physical channel used as TCHH_ACCH's for instance 1 of cell B	636
management/	SDCCH8_A_1_nociph	To set one physical channel used as SDCCH8 channel for instance 1 of cell A.	636
Miscellaneous/	Adjust_gsmanddcs_powerlvl		637
Miscellaneous/	AssCh_complete		637
Miscellaneous/	AssCmdGen		638

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Miscellaneous/	AssCmdGen_fh	To generate a ASSIGNMENT COMMAND message in case of hopping without starting time.	639
Miscellaneous/	Authentication	To execute the authentication procedure.	639
Miscellaneous/	CallProcGen	To generate a CALL PROCEEDING message .	640
Miscellaneous/	CCCH_group_Paging_group		642
Miscellaneous/	CC_Est_MT_Call	To perform the CC message exchange to establish a mobile station terminating call (late assignment).	643
Miscellaneous/	CC_EstMsTermCall	To perform the CC message exchange to establish a mobile station terminating call with non hopping channel (speech or data call). (Similar to EstMsTermFullRateCallNonFH but without initial RRmtcallprepare)	644
Miscellaneous/	Check_Cells_Present	To receive a specified number of measurement reports (TSPX_Msr_Cell_Present) describing a specified number of cells present.	645
Miscellaneous/	Check_Result		645
Miscellaneous/	Ciphering_on	To switch on the ciphering.	646
Miscellaneous/	EFR_ChkRcvSpchPrefs	OtherEvents	646
Miscellaneous/	EFR_SpchPrefInBCAP	OtherEvents	647
Miscellaneous/	EFR_RcvSetupOrEsetup	To check that Mobile Originated SETUP or ESETUP message contains a bearer capability IE (optional for ESETUP) with all the speech versions supported by the EFR mobile. If matching is successful then a CALL PROCEEDING message response will be sent to the EFR mobile.	648
Miscellaneous/	FnArith		650
Miscellaneous/	ImsiDetach		651
Miscellaneous/	MM_PwrOrSimOff		652
Miscellaneous/	MM_PwrOrSimOn		652
Miscellaneous/	RcvSetupOrEsetup		653
Miscellaneous/	RR_hocomp1	To finish the HO−procedure. Timing advance = 20 bits period	654
Miscellaneous/	RRmtcallprepare	To prepare a mobile terminating call establishment.	655
Miscellaneous/	SelectPagingCh		656
Miscellaneous/	SendSeqNo_chk	To check whether the sending sequence number is the same	657
Miscellaneous/	SvcSupportedChk	To check wether the Basic Service svc is supported against PICS answers.	658
Miscellaneous/	Varinit_fix		659
Miscellaneous/	ChgLAC_A	To change the LAC of cell A.	660

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Miscellaneous/	ChgLAC_B	To change the LAC of cell B.	660
Miscellaneous/	ChgLAI_C	To change the LAI of cell C to HPLMN.	661
Miscellaneous/	SysInfoSending_cbms	To send system information messages for the CBMS tests.	662
Miscellaneous/	SysInfoSending_fh	To send system information messages for the L3 tests. The following parameters specified by input parameters.	664
Miscellaneous/	SysInfoSending_nfh	To send system information messages with default parameters defined for L3 tests for which no other special parameters indicated.	666
Miscellaneous/	SysInfoSending_nfhCb	To send system information messages with default parameters defined for L3 tests except that the cell is barred.	668
Miscellaneous/	SysInfoSending_e	Sending of Systeminformation for cell A in EGSM test case TC_26_10_2_2.	670
Miscellaneous/	SysInfo_SacchSending	To send SYSTEM INFORMATION 5 and 6 messages defined by parameters 'sysinfo5_pdu' and 'sysinfo6_pdu' in the parametrised 'ch' channel.	671
Miscellaneous/	SysInfo_5bisSending	To send SYSTEM INFORMATION 5bis message defined by parameters 'sysinfo5bis_pdu' in the parametrised 'ch' channel.	671
Miscellaneous/	SetNECI	To set the NECI =1.	672
Miscellaneous/	SetATT	To set the ATT flag to "MS's in the cell should apply IMSI attach and detach procedure"	673
OperatorOP/	InitCall	To initiate a call of the basic service 'srv'.	674
OperatorOP/	InitNonCallSupp	To attempt a non call related supplementary service at the MS under test.	674
OperatorOP/	AtmpShortMsg	To attempt an MO short message service transaction at the MS under test.	674
OperatorOP/	CheckUssdStringDisplayed	To check whether the correct USSD String is displayed on the MS	675
OperatorOP/	PLMNsCHK	To check whether the MS presents a list of available PLMNs.	675
OperatorOP/	RFtransCHK	To check whether the MS transmits any radio signal.	675
Postambles/	PostLinkRelEnd	To release the RR connection and bring the MS back to Idle state. This teststep decides the final verdict and therefor it shall be used only at the end of testcases.	676
Postambles/	PostMainLinkRel	To release the main signalling link 'ch', and bring the MS back to Idle state.	676

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Postambles/	ChanRel	To release the RR connection on the channel TCV_ch and bring the MS back to Idle state.	676
Postambles/	ChanRel_P	To release the RR connection on the channel TCV_chmaindcch and bring the MS back to Idle state. This teststep decides the verdict (P).	677
Postambles/	ChanRel_end	To release the RR connection and bring the MS back to Idle state. This teststep decides the final verdict and therefor it shall be used only at the end of testcases.	677
Preambles/	Chmod	To assign correct values to the test case variables TCV_ChMod and TCV_ChModb.	678
Preambles/	EFR_BasicServiceMO	To generate a EFR Bcap and a MO SETUP/ESETUP message (to match later against received MO SETUP/ESETUP message), and an appropriate channel mode based on the service and rate, as well as on several IXIT parameters, itc, ce, ur, synchronous bit for an MO call. Bcap shall match the correct NUMBER of speech versions that the EFR mobile supports.	679
Preambles/	EFR_BasicServiceMT	To generate a EFR Bcap (to match against received CALL CONNECTED Bcap), an MT SETUP or ESETUP message (to be sent to EFR mobile) and an appropriate channel mode based on the service and rate, as well as on several IXIT parameters, itc, ce, ur, synchronous bit for an MT call. EFR Bcap shall match the correct NUMBER of speech versions that the EFR mobile supports.	681
Preambles/	Est_MO_Call_init	To initiate a mobile originating call for the supported bearer capability. The channel in use is frequency hopping channel.	683
Preambles/	Est_MT_Call_FH	To establish a mobile station terminating call with hopping channel (speech or data call).	685
Preambles/	Est_MT_CallNonFH	To establish a mobile station terminating call with non hopping channel (speech or data call).	686
Preambles/	IdleUpdated	This test step is called at the preamble of each test case.	687
Preambles/	PreEstRRConn	To establish a RR connection on SDCCH subchannel defined by TSPX_SDCCH4SubDef in cell A	698

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Preambles/	StartCellA_ho	To setup a physical channel as combined BCCH, CCCH and SDCCH4 then broadcast SYSTEM INFORMATION messages 2, 3, 4, 5 and 6 with default parameters of cell A.	699
Preambles/	StartCellB_ho	To setup a physical channel as combined BCCH, CCCH and SDCCH4 then broadcast SYSTEM INFORMATION messages 2, 3, 4, 5 and 6 with default parameters of cell B.	700

Detailed Comments :

Default Index

Default Group Reference	Default Id	Description	Page Nr
	OtherEvents	To match unexpected events and sign final verdict for preambles.	701
	OtherEventsFail	To match unexpected events and fail the test case.	701
	OtherEventsFail_01	To match irrelevant CHANNEL REQUEST msg and return or match other unexpected events and fail the test case.	702
	OtherEventsFail_02	To match unexpected events and fail the test case but ignore channel request messages that are sent before the lower tester has sent (and the MS received) the Immediate Assigment message.	703
	OtherEvents_01	To match irrelevant messages and return	703
	OtherEvents_02	To match unexpected events and sign final verdict for preambles but ignore channel request messages that are sent before the lower tester has sent (and the MS received) the Immediate Assigment message.	704
	RcvHdOvAcc	To match any HANDOVER ACCESS then return to calling tree.	704

Detailed Comments :

II

Declarations Part

Simple Type Definitions			
Type Name	**Type Definition**	**Type Encoding**	**Comments**
B_1	BITSTRING [1]		Generic type for 1 bit value
B_2	BITSTRING [2]		Generic type for 2 bits value
B_3	BITSTRING [3]		Generic type for 3 bits value
B_4	BITSTRING [4]		Generic type for 4 bits value
B_5	BITSTRING [5]		Generic type for 5 bits value
B_6	BITSTRING [6]		Generic type for 6 bits value
B_8	BITSTRING [8]		Generic type for 8 bits value
BCCHFRQ	BITSTRING [5]		position of a bcch carrier in the bcch channel list
BCC	BITSTRING [3]		BS colour code
BCDN	OCTETSTRING [1..10]		BCD numbers, GSM 04.08, 10.5.4.7, octet 4−13
BSIC	BITSTRING [6]		base station identity code
CCCH_CON	BITSTRING[3]		number of physical channels for ccch GSM 04.08, 10.5.2.11
CCSTATE	INTEGER(1,3,4,6,7,8,9,10,11,12,19,26,27)		
CellID	IA5String		Cell identifier
CHANNEL	BITSTRING [2]		needed channel type
CHMOD_VAL	BITSTRING[8]		value for channel mode GSM 04.08, 10.5.2.6, 10.5.2.7
CH_TDMA	BITSTRING[5]		channel type and TDMA offset GSM 04.08, 10.5.2.5
CI	OCTETSTRING [2]		cell identity, GSM 04.08, 10.5.1.1
CKSN	BITSTRING[3]		ciphering key sequence number(only key sequence) GSM 04.08, 10.5.1.2
CLRSUP	BITSTRING('10100001'B)		CLIR suppression GSM 04.08, 10.5.4.11a
CLRINV	BITSTRING('10100010'B)		CLIR invocation GSM 04.08, 10.5.4.11b
CMSVTYPE	BITSTRING [4]		CM service type
CP_CAU	OCTETSTRING [1]		cp cause GSM 04.11, 8.1.4.2
CPHALG	BITSTRING[3]		Ciphering algorithm identifier GSM 04.08, 10.5.2.9
CS	BITSTRING [2]		coding standard
EstMode	OCTETSTRING [1]		establish mode
EXTB	BITSTRING [1]		extension bit
FCS	HEXSTRING [2]		TP failure cause
FLWCNTL	INTEGER(0, 1, 2)		type of flow control. 0−−outband flow control; 1−−inband flow control; 2−− no flow control
HORF	BITSTRING [8]		handover reference
HSN	BITSTRING[6]		Hopping sequence number e.g.GSM 04.08 10.5.2.5

Continued on next page

Continued from previous page

Simple Type Definitions			
Type Name	**Type Definition**	**Type Encoding**	**Comments**
IDTYPE	B_4		identity type
IEI_4	BITSTRING [4]		information element identifier, type 1
IEI_8	BITSTRING [8]		information element identifier, type 2−4
L2FMTYPE	INTEGER(1..3)		L2 frame type
LENGTH	OCTETSTRING [1]		IE length or L2 pseudo length. The L2 pseudo length is composed of a L2 pseudo length value, which is 6 bits long, and of 2 additional bits. See GSM 04.08 section 10.5.2.19
LOGICCH	IA5String		
LOGCH	IA5String		
MAC	OCTETSTRING[3]		mac, e.g. GSM 04.08 10.5.2.21
MAIO	BITSTRING[6]		MAIO,GSM 04.08 10.5.2.5
MAXTXPOW	INTEGER(0..31)		Max Tx Power Level GSM 04.08, 10.5.2.4
MR	OCTETSTRING [1]		SMS RP or TP message reference
MT	BITSTRING[8]		message type, structure of it see MT0
MTI	BITSTRING [3]		SMS RP message type indicator
NCC	BITSTRING [3]		PLMN colour code
NCCP	OCTETSTRING [1]		ncc permitted
NTI	BITSTRING ('10000000'B, '10000001'B, '10000010'B)		notification indicator GSM 04.08, 10.5.4.20
PD	B_4		Protocol discriminator
PGG	BITSTRING [8]		the paging group of an MS to be paged in binary presentation
RATE	IA5String ("F", "H", "Fv2")		
RAND	BITSTRING [128]		random number
RCSD	BITSTRING [8]		reverse call setup direction GSM 04.08, 10.5.4.22a
RelMode	OCTETSTRING [1]		release mode
REJCAU	OCTETSTRING [1]		reject cause GSM 04.08, 10.5.3.6
RPI	BITSTRING [8]		repeat indicator
RRCAU	BITSTRING [8]		RR cause GSM 04.08, 10.5.2.31
RXLEV	BITSTRING [6]		received signal strength
SAPID	OCTETSTRING [1]		sap identifier
SENDINGMODE	INTEGER(1..4)		the mode of sending two consecutive messages containing paging mode IE

Continued on next page

Continued from previous page

Simple Type Definitions			
Type Name	**Type Definition**	**Type Encoding**	**Comments**
SERVICES	IA5String ("C_Telephony", "C_EmgCall", "C_AltSpchFax", "C_AutoFax", "C_Async300", "C_Async1200", "C_Async120075", "C_Async2400", "C_Async4800", "C_Async9600", "C_Sync1200", "C_Sync2400", "C_Sync4800", "C_Sync9600", "C_PAD300", "C_PAD1200", "C_PAD120075", "C_PAD2400", "C_PAD4800", "C_PAD9600", "C_Packet2400", "C_Packet4800", "C_Packet9600", "C_AltSpchData", "C_SpchData")		
SHOCT	B_4		spare half octet for type 1 information element GSM 04.08, 10.5.1.8
SKI	B_4		Skip indicator
SN	BITSTRING [3]		all possible slot number, GSM 04.08 10.5.2.5
SPB	B_1		spare bit
SP2B	B_2		2 spare bits
SP3B	B_3		3 spare bits
SP5B	B_5		5 spare bits
SP6B	B_6		6 spare bits
SRES	OCTETSTRING [4]		authentication response signature
TPCD	OCTETSTRING [1..157]		TP command data
TPSCTS	HEXSTRING [14]		TP service centre time stamp
TPUD	OCTETSTRING [1..140]		TP user data
TSC	B_3		Training sequence code,GSM 04.08 10.5.2.5
T1	BITSTRING [11]		
T1_	B_5		T1’
T2	B_5		
T3	BITSTRING [6]		
T3_	B_3		T3’
TI_V	B_3		TI value
TMSI_V	OCTETSTRING[4]		TMSI value is octetstring of length 4
TZONES	INTEGER (−79..79)		time zones in 15 minute steps
WI	HEXSTRING [2]		wait indication, unit in second

Continued on next page

Continued from previous page

Simple Type Definitions

Detailed Comments :

Structured Type Definition

Type Name : ACST

Encoding Variation :

Comments : Auxiliary states (CC information element)
GSM 04.08, 10.5.4.4

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'00100100'B
iel	LENGTH		
extb	EXTB		extension bit
sp3b	SP3B		3 spare bits
has	BITSTRING [2]		hold auxiliary state
mpas	BITSTRING [2]		multi party auxiliary state

Detailed Comments :

Structured Type Definition

Type Name : BCAP
Encoding Variation :
Comments : Bearer capability (CC information element)
GSM 04.08, 10.5.4.5

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00000100’B, octet 1
iel	LENGTH		
extb3	EXTB		extension bit (’1’B), octet 3
rchr	BITSTRING [2]		radio channel requirements ms−>n, octet 3
cs	BITSTRING [1]		coding standard, ’0’B, octet 3
tm	BITSTRING [1]		transfer mode
itc	BITSTRING [3]		information transfer capability, ’0 − 3, 7’, octet 3
oct_3a_etc_1	EFR_BCAP_3A_ETC		octet 3a etc no.1
oct_3a_etc_2	EFR_BCAP_3A_ETC		octet 3a etc no.2
oct_3a_etc_3	EFR_BCAP_3A_ETC		octet 3a etc no.3
oct_3a_etc_4	EFR_BCAP_3A_ETC		octet 3a etc no.4
oct_3a_etc_5	EFR_BCAP_3A_ETC		octet 3a etc no.5
oct_3a_etc_6	EFR_BCAP_3A_ETC		octet 3a etc no.6
extb4	EXTB		extension bit, ’1’B, octet 4
spb	SPB		spare bit , octet 4
strc	BITSTRING [2]		structure, ’0, 3’, octet 4
dplxm	BITSTRING [1]		duplex mode, octet 4
config	BITSTRING [1]		configuration, ’0’B, octet 4
nirr	BITSTRING [1]		negotiation of intermediate rate requested, octet 4
est	BITSTRING [1]		establishment, ’0’B, octet 4
extb5	EXTB		extension bit, ’1’B, octet 5
accid	BITSTRING [2]		access identify, ’00’B, octet 5
ra	BITSTRING [2]		rate adaption, ’0 − 2’, octet 5
sacp	BITSTRING [3]		signalling access protocol, ’1 − 6’, octet 5
extb6	EXTB		extension bit, ’1’B, octet 6
l1id	BITSTRING [2]		L1 identity, ’01’B, octet 6
uil1	BITSTRING [4]		user information L1 protocol, ’0000’B, octet 6
sb	BITSTRING [1]		synchronous bit, octet 6
extb6a	EXTB		extension bit, ’1’B, octet 6a
nsb	BITSTRING [1]		number of stop bits, octet 6a
nb	BITSTRING [1]		negotiation bit, ’0’B, octet 6a
ndb	BITSTRING [1]		number of data bits, octet 6a
ur	BITSTRING [4]		user rate, ’0 − 7’, octet 6a
extb6b	EXTB		extension bit, ’1’B, octet 6b

Continued on next page

Continued from previous page

Structured Type Definition			
Element Name	**Type Definition**	**Field Encoding**	**Comments**
ir	BITSTRING [2]		intermediate rate, ’2, 3’, octet 6b
nictx	BITSTRING [1]		network independent clock on transmission, octet 6b
nicrx	BITSTRING [1]		network independent clock on reception, octet 6b
pi	BITSTRING [3]		parity information, ’0 − 5’, octet 6b
extb6c	EXTB		extension bit, ’1’B, octet 6c
ce	BITSTRING [2]		connection element, octet 6c
modemt	BITSTRING [5]		modem type, ’0 − 8’, octet 6c
extb7	EXTB		extension bit, ’1’B, octet 7
l2id	BITSTRING [2]		L2 identity, ’10’B, octet 7
uil2	BITSTRING [5]		user information L2 protocol, ’6, 8 − 10, 12’, octet 7
Detailed Comments :			

Structured Type Definition			
Type Name : BLOCKTYPE			
Encoding Variation :			
Comments : Block type, GSM 04.12, 3.3.1			
Element Name	**Type Definition**	**Field Encoding**	**Comments**
spare1	SPB		
lpd	BITSTRING[2]		’01’B
lb	BITSTRING[1]		
sequence_number	BITSTRING[4]		
Detailed Comments :			

Structured Type Definition

Type Name : CAU
Encoding Variation :
Comments : Cause (CC information element)
GSM 04.08, 10.5.4.11

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00001000’B, octet 1
iel	LENGTH		
extb3	EXTB		extension bit
cs	CS		coding standard
spb	SPB		spare bit
location	BITSTRING [4]		location, ’0 − 5, 7, 12’
extb3a	EXTB		extension bit
rec	BITSTRING [7]		recommendation
extb4	EXTB		extension bit
cau_class	BITSTRING [3]		cause class
cau_va	BITSTRING [4]		cause value
cau_di	OCTETSTRING [1..28]		Diagnostics

Detailed Comments : &COMMON_U06

Structured Type Definition

Type Name : CCCAP
Encoding Variation :
Comments : Call Control Capabilities
GSM 04.08, 10.5.4.5a

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00010101’B
iel	LENGTH		length of call control capabilities contents
spr	BITSTRING [7]		spare bits ’0000000’B
dtmf	BITSTRING [1]		indication of supporting DTMF

Detailed Comments :

Structured Type Definition

Type Name : CCD

Encoding Variation :

Comments : Control channel description
GSM 04.08, 10.5.2.11

Element Name	Type Definition	Field Encoding	Comments
spr1	BITSTRING [1]		spare bit
att	BITSTRING [1]		attach−detach allowed
babr	BITSTRING [3]		base station access grant blocks reservation
ccch_con	BITSTRING [3]		number of physical channels for ccch
spr2	BITSTRING [5]		spare bits
bpm	BITSTRING [3]		number of multiframes for paging request
t3212	OCTETSTRING [1]		t3212 value

Detailed Comments :

Structured Type Definition

Type Name : CCHD

Encoding Variation :

Comments : Cell channel description
GSM 04.08, 10.5.2.1

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01100010’B
rfl	OCTETSTRING [16]		reference frequency list

Detailed Comments :

Structured Type Definition

Type Name : CD

Encoding Variation :

Comments : Cell description
GSM 04.08, 10.5.2.2

Element Name	Type Definition	Field Encoding	Comments
bcch_arfcn_h	BITSTRING [2]		bcch arfcn high part
ncc	NCC		PLMN colour code
bcc	BCC		BS colour code
bcch_arfcn_l	BITSTRING [8]		bcch arfcn low part

Detailed Comments : The info element is two octets long.

Structured Type Definition

Type Name : CDPN
Encoding Variation :
Comments : Called party BCD number (CC information element)
GSM 04.08, 10.5.4.7

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01011110'B
iel	LENGTH		
tonnpi	TON_NPI		Type of number and numbering plan
digits	BCDN		BCD numbers

Detailed Comments :

Structured Type Definition

Type Name : CDPS
Encoding Variation :
Comments : Called party subaddress (CC information element)
GSM 04.08, 10.5.4.8

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01101101'B
iel	LENGTH		
subad	SUBAD		Subaddress

Detailed Comments :

Structured Type Definition

Type Name : CE
Encoding Variation :
Comments : connection element

Element Name	Type Definition	Field Encoding	Comments
TorBothTorBothNT	B_2		
NTorBothTorBothNT	B_2		

Detailed Comments :

Structured Type Definition

Type Name : CGPN

Encoding Variation :

Comments : Calling party BCD number (CC information element)
GSM 04.08, 10.5.4.9

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01011100'B
iel	LENGTH		
tonnpi	TON_NPI		Type of number and numbering plan
pisi	PI_SI		Presentation indicator & screening indicator
digits	BCDN		BCD numbers

Detailed Comments :

Structured Type Definition

Type Name : CGPS

Encoding Variation :

Comments : Calling party subaddress (CC information element)
GSM 04.08, 10.5.4.10

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01011101'B
iel	LENGTH		
subad	SUBAD		Subaddress

Detailed Comments :

Structured Type Definition

Type Name : CHD

Encoding Variation :

Comments : Channel description
GSM 04.08, 10.5.2.5

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01100100'B
cht_schn	BITSTRING [5]		channel type and TDMA offset
tn	BITSTRING [3]		timeslot number
tsc	BITSTRING [3]		training sequence code
hch	BITSTRING [1]		hopping channel
maio	BITSTRING [6]		mobile allocation index offset − hopping
hsn	BITSTRING [6]		hopping sequence number − hopping
spr	BITSTRING [2]		'00'B − non hopping
arfcn	BITSTRING [10]		absolute RF channel number − non hopping

Detailed Comments :

Structured Type Definition

Type Name : CHMOD
Encoding Variation :
Comments : Channel mode, channel mode2
GSM 04.08, 10.5.2.6, 10.5.2.7

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01100011'B, '01100110'B
mode	B_8		

Detailed Comments :

Structured Type Definition

Type Name : CHNEED
Encoding Variation :
Comments : Channels needed
GSM 04.08, 10.5.2.8

Element Name	Type Definition	Field Encoding	Comments	
ch2	CHANNEL		needed channel 2	O
ch1	CHANNEL		needed channel 1	M

Detailed Comments :

Structured Type Definition

Type Name : CNN
Encoding Variation :
Comments : Connected number (CC information element)
GSM 04.08, 10.5.4.13

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01001100'B
iel	LENGTH		
tonnpi	TON_NPI		Type of number and numbering plan
pisi	PI_SI		Presentation indicator & screening indicator
digits	BCDN		BCD numbers

Detailed Comments :

Structured Type Definition

Type Name : CNS

Encoding Variation :

Comments : Connected subaddress (CC information element)
GSM 04.08, 10.5.4.14

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01001101’B
iel	LENGTH		
subad	SUBAD		Subaddress

Detailed Comments :

Structured Type Definition

Type Name : CO

Encoding Variation :

Comments : Cell options (BCCH / SACCH)
GSM 04.08, 10.5.2.3

Element Name	Type Definition	Field Encoding	Comments
sprb	BITSTRING [1]		spare bit for CO/BCCH, dtxh for half rate
pwrc	BITSTRING [1]		power control indicator
dtx	BITSTRING [2]		CO/BCCH: discontinuous transmission indicator, CO/SACCH: DTX−F for the full rate
rlt	BITSTRING [4]		radio link timeout

Detailed Comments :

Structured Type Definition

Type Name : Component_T
Encoding Variation :
Comments : Component for sending (downlink)
GSM 04.80, 3.6

Element Name	Type Definition	Field Encoding	Comments
comp_part1	OCTETSTRING		Octet string before the value of invoke ID
invokeId	OCTETSTRING [1]		received value of invoke id
comp_part2	OCTETSTRING		
comp_part3	OCTETSTRING		
comp_part4	OCTETSTRING		
comp_part5	OCTETSTRING		
comp_part6	OCTETSTRING		
comp_part7	OCTETSTRING		
comp_part8	OCTETSTRING		
comp_part9	OCTETSTRING		
comp_part10	OCTETSTRING		
comp_part11	OCTETSTRING		
comp_part12	OCTETSTRING		

Detailed Comments : This component is defined for the use in the downlink (from the test system to the IUT) direction.

Structured Type Definition

Type Name : CPDATA
Encoding Variation :
Comments : CP−User data element
GSM 04.11, 8.1.4.1

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'00000001'B
iel	LENGTH		
rpack	RPACK		n <−> ms, O
rpdata	RPDATA		n <−> ms, O
rperr	RPERR		n<−>ms, O
rpsmma	RPSMMA		ms −> n, O

Detailed Comments : One of the 4 types of rpdu is contained in the CPDATA.

Structured Type Definition

Type Name : CPHMS
Encoding Variation :
Comments : Cipher mode setting
GSM 04.08, 10.5.2.9

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_4		’1001’B
algid	BITSTRING [3]		algorithm identifier
sc	BITSTRING [1]		starting ciphering

Detailed Comments :

Structured Type Definition

Type Name : CPHKSN
Encoding Variation :
Comments : Ciphering key sequence number
GSM 04.08, 10.5.1.2

Element Name	Type Definition	Field Encoding	Comments
sprb	BITSTRING [1]		spare bit ’0’B
ks	BITSTRING [3]		key sequence

Detailed Comments :

Structured Type Definition

Type Name : CPH_RES
Encoding Variation :
Comments : Cipher response
GSM 04.08, 10.5.2.10

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_4		’1010’B
sprb	BITSTRING [3]		spare bits
cr	BITSTRING [1]		cipher response bit

Detailed Comments :

Structured Type Definition

Type Name : CSP
Encoding Variation :
Comments : Cell selection parameters
GSM 04.08, 10.5.2.4

Element Name	Type Definition	Field Encoding	Comments
crh	BITSTRING [3]		cell reselected hysteresis
mtmc	BITSTRING [5]		MS maximum TX power for CCH
acs	BITSTRING [1]		additional reselect parameter indicator
neci	BITSTRING [1]		half rate support
ram	BITSTRING [6]		MS minimum received signal level

Detailed Comments :

Structured Type Definition

Type Name : CST
Encoding Variation :
Comments : Call status (CC information element)
GSM 04.08, 10.5.4.6

Element Name	Type Definition	Field Encoding	Comments
cs	CS		coding standard
csv	BITSTRING [6]		call status value

Detailed Comments :

Structured Type Definition

Type Name : EFR_BCAP_3A_ETC
Encoding Variation :
Comments : Speech Versions preferences as supported by the EFR mobile − see GSM 04.08/10.5.4.5/octet 3a etc/"speech version indication"

Element Name	Type Definition	Field Encoding	Comments
extb	EXTB		Octet a1, Extension bit ('0'B or '1'B)
cod	BITSTRING[1]		Octet a1, Coding ('0'B)
spr2	BITSTRING[2]		Octet a1, Spare bits ('00'B)
svi	BITSTRING [4]		Octet a1, Speech version indication

Detailed Comments :

Structured Type Definition

Type Name : FIE

Encoding Variation :

Comments : Facility information element
GSM 04.80, 3.6

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'00011100'B
iel	LENGTH		
components_1	Components		Components of asn.1 type, for receiving uplink components
components_t	Component_T		Component of TTCN type, for sending a downlink component

Detailed Comments : When sending normally only one component is sent, but when receiving any number of components can be received even though normally we are only interested in one component during the testing.

Structured Type Definition

Type Name : FN

Encoding Variation :

Comments : The time in reference to the frame numbering corresponding to the absolute frame number modulo 42432.

Element Name	Type Definition	Field Encoding	Comments
t1_	T1_		
t3	T3		
t2	T2		

Detailed Comments : In STRT and RQR

Structured Type Definition

Type Name : FRQCHS

Encoding Variation :

Comments : Frequency channel sequence
GSM 04.08, 10.5.2.12

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01101001'B
larfcn	OCTETSTRING [1]		lowest ARFCN (1−124)
incs	OCTETSTRING [8]		increment skip ARFCN n

Detailed Comments :

Structured Type Definition

Type Name : FRQL

Encoding Variation :

Comments : Frequency list, frequency short list
GSM 04.08, 10.5.2.5, 10.5.2.6

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00000101’B, ’00000010’B
iel	LENGTH		OCTETSTRING [1]
fl	OCTETSTRING		list of the absolute radio frequency channel numbers

Detailed Comments : frequency short list has a fixed length of 10 octets and does not contain the length indicator.

Structured Type Definition

Type Name : FRQPARA

Encoding Variation :

Comments : Parameters for Description of basic physical channel in frequency domain.

Element Name	Type Definition	Field Encoding	Comments
hch	BITSTRING[1]		hopping channel
maio	BITSTRING [6]		mobile allocation index offset − hopping, if hch = 1, otherwise OMIT
hsn	BITSTRING [6]		hopping sequence number − hopping, if hch = 1, otherwise OMIT
spr	BITSTRING [2]		’00’B − non hopping, if hch = 0, otherwise OMIT
arfcn	BITSTRING [10]		absolute RF channel number − non hopping, if hch = 0, otherwise OMIT
maclength	LENGTH		’00’O for non hopping
mac_8n	BITSTRING [8]		mac64 − mac57
mac_7n	BITSTRING [8]		mac56 − mac49
mac_6n	BITSTRING [8]		mac48 − mac41
mac_5n	BITSTRING [8]		mac40 − mac33
mac_4n	BITSTRING [8]		mac32 − mac25
mac_3n	BITSTRING [8]		mac24 − mac17
mac_2n	BITSTRING [8]		mac16 − mac9
mac_1n	BITSTRING [8]		mac8 − mac1
flst	FRQL		frequency list used in hopping channel, otherwise OMIT

Detailed Comments :

Structured Type Definition

Type Name : HOACC_PARA
Encoding Variation :
Comments : Handover Access parameters

Element Name	Type Definition	Field Encoding	Comments
num_bursts	OCTETSTRING[1]		
min_mspwr	OCTETSTRING[1]		
max_mspwr	OCTETSTRING[1]		
min_msta	BITSTRING[8]		
max_msta	BITSTRING[8]		

Detailed Comments :

Structured Type Definition

Type Name : HLCMP
Encoding Variation :
Comments : High layer compatibility (CC information element)
GSM 04.08, 10.5.4.16, ITU Q.931

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01111101’B
iel	LENGTH		
extb3	EXTB		extension bit (’1’B)
cs	BITSTRING [2]		coding standard
in	BITSTRING [3]		interpretation
pmpp	BITSTRING [2]		presentation method of protocol profile
extb4	EXTB		extension bit, octet 4
hlci	BITSTRING [7]		high layer characteristics identification
extb4a	EXTB		extension bit, ’1’B, octet 4a
ehlci	BITSTRING [7]		extended high layer characteristics identification

Detailed Comments :

Structured Type Definition

Type Name : IARESTOCT
Encoding Variation :
Comments :

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		
iaroct1	OCTETSTRING[1..11]		p=00 format
iaroct2	IARESTOCT2		p=10 format used in TC_26_6_13_9, TC_26_6_13_10

Detailed Comments :

Structured Type Definition

Type Name : IARESTOCT2
Encoding Variation :
Comments :

Element Name	Type Definition	Field Encoding	Comments
p	BITSTRING[2]		'10'B p=10 format
frqparalen	BITSTRING[6]		length of frequency parameters contents
spbt1	BITSTRING[2]		spare bits '00'B
maio	BITSTRING[6]		MAIO mobile allocation index offset
ma	OCTETSTRING[0..8]		MA mobile allocation
spbt2	OCTETSTRING[0..8]		spare bits '2B'O

Detailed Comments :

Structured Type Definition

Type Name : KPF
(keypad facility)
Encoding Variation :
Comments : keypad facility
GSM 04.08 clause 10.5.4.17

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		keypad facility identity, '00101100'B
extb	EXTB		extension bit ('0'B)
kpf_info	IA5String [1]		one DTMF digit (IA5 character)

Detailed Comments : Is extb needed?

Structured Type Definition

Type Name : L1HD
Encoding Variation :
Comments : Layer 1 Header

Element Name	Type Definition	Field Encoding	Comments
spr1	BITSTRING[3]		spare bits
mspwrlvl	BITSTRING[5]		actual MS power level
spr2	BITSTRING[1]		spare bit
ta	BITSTRING[7]		actual timing advance

Detailed Comments :

Structured Type Definition

Type Name : LAI

Encoding Variation :

Comments : Location area identification
GSM 04.08, 10.5.1.3

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00010011’B
mcc	OCTETSTRING [2]		mobile country code, 3 digits
mnc	OCTETSTRING [1]		mobile network code, 2 BCD digits
lac	OCTETSTRING [2]		location area code

Detailed Comments :

Structured Type Definition

Type Name : LLCMP

Encoding Variation :

Comments : Low layer compatibility (CC information element)
GSM 04.08, 10.5.4.18 and ETS DE/SPS−5034−1 (update of ETS 300 102−1)

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01111100’B
iel	LENGTH		
extb3	EXTB		extension bit , octet 3
cs	BITSTRING [2]		coding standard, octet 3
itc	BITSTRING [5]		information transfer capability, octet 3
extb3a	EXTB		extension bit, ’1’B, octet 3a
negind	BITSTRING[1]		negotiation indicator, octet 3a
spb3a	SP6B		spare bits, ’000000’B, octet 3a
extb4	EXTB		extension bit, octet 4
tm	BITSTRING [2]		transfer mode
itr	BITSTRING [5]		information transfer rate, octet 4
extb4_1	EXTB		extension bit, ’1’B, octet 4.1
rateml	BITSTRING [7]		rate multiplier, octet 4.1
extb5	EXTB		extension bit, octet 5
l1id	BITSTRING [2]		L1 identity, ’01’B, octet 5
uil1	BITSTRING [5]		user information L1 protocol, octet 5
extb5a	EXTB		extension bit, octet 5a
sb	BITSTRING [1]		synchronous bit, octet 5a
neg	BITSTRING [1]		negotiation bit, octet 5a
ur	BITSTRING [5]		user rate, octet 5a
extb5b1	EXTB		extension bit, octet 5b1
ir	BITSTRING [2]		intermediate rate, octet 5b1
nictx	BITSTRING [1]		network independent clock on transmission, octet 5b1
nicrx	BITSTRING [1]		network independent clock on reception, octet 5b1
fctx	BITSTRING [1]		flow control on transmission, octet 5b1
fcrx	BITSTRING [1]		flow control on reception, octet 5b1
spb5b1	SPB		spare bit, ’0’B, octet 5b1
extb5b2	EXTB		extension bit, octet 5b2
hdrb	BITSTRING[1]		Hdr/no Hdr−bit, octet 5b2
mfs	BITSTRING[1]		Multi frame support, octet 5b2
mode	BITSTRING[1]		Mode, octet 5b2
llineg	BITSTRING[1]		LLI negotiation, octet 5b2
ass	BITSTRING[1]		Assignor/Assignee, octet 5b2
ibob	BITSTRING[1]		Inband/outband negotiation, octet 5b2
spb5b2	SPB		spare bit, ’0’B, octet 5b2

Continued on next page

Continued from previous page

Structured Type Definition			
Element Name	**Type Definition**	**Field Encoding**	**Comments**
extb5c	EXTB		extension bit, octet 5c
nsb	BITSTRING [2]		number of stop bits, octet 5c
ndb	BITSTRING [2]		number of data bits, octet 5c
pi	BITSTRING [3]		parity, octet 5c
extb5d	EXTB		extension bit, '1'B, octet 5d
dplxm	BITSTRING [1]		duplex mode, octet 5d
modemt	BITSTRING [6]		modem type, octet 5d
extb6	EXTB		extension bit, octet 6
l2id	BITSTRING [2]		L2 identity, '10'B, octet 6
uil2	BITSTRING [5]		user information L2 protocol, octet 6
extb6a1	EXTB		extension bit, '1'B, octet 6a1
modl2	BITSTRING[2]		mode, octet 6a1
spb6a1	SP3B		spare bits , '000'B, octet 6a1
q933	BITSTRING [2]		Q.933 use
extb6a2	EXTB		extension bit, '1'B, octet 6a2
usl2pi	BITSTRING[7]		user specified layer 2 protocol information, octet 6a2
extb6b	EXTB		extension bit, '1'B, octet 6b
ws	BITSTRING[7]		window size, octet 6b
extb7	EXTB		extension bit, octet 7
l3id	BITSTRING [2]		L3 identity, '11'B, octet 7
uil3	BITSTRING [5]		user information L3 protocol, octet 7
extb7a1	EXTB		extension bit, octet 7a1
modl3	BITSTRING[2]		mode, octet 7a1
spb7a1	SP5B		spare bits , '00000'B, octet 7a1
extb7a2	EXTB		extension bit, '1'B, octet 7a2
usl3pi	BITSTRING[7]		user specified layer 3 protocol information, octet 7a2
extb7b	EXTB		extension bit, octet 7b
spb7b	SP3B		spare bits , '000'B, octet 7b
dps	BITSTRING [4]		default packet size, octet 7b
extb7c	EXTB		extension bit, '1'B, octet 7c
pws	BITSTRING[7]		packet window size, octet 7bc
Detailed Comments :			

Structured Type Definition

Type Name : LUT
Encoding Variation :
Comments : Location updating type
GSM 04.08, 10.5.3.5

Element Name	Type Definition	Field Encoding	Comments
foreq	BITSTRING [1]		follow−on request bit
sprb	BITSTRING [1]		spare bit ’0’B
lut	BITSTRING [2]		location updating type

Detailed Comments :

Structured Type Definition

Type Name : MA
Encoding Variation :
Comments : Mobile allocation
GSM 04.08, 10.5.2.21

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01110010’B
iel	LENGTH		’00’O for non hopping
mac_8n	BITSTRING [8]		mac64 − mac57
mac_7n	BITSTRING [8]		mac56 − mac49
mac_6n	BITSTRING [8]		mac48 − mac41
mac_5n	BITSTRING [8]		mac40 − mac33
mac_4n	BITSTRING [8]		mac32 − mac25
mac_3n	BITSTRING [8]		mac24 − mac17
mac_2n	BITSTRING [8]		mac16 − mac9
mac_1n	BITSTRING [8]		mac8 − mac1

Detailed Comments :

Structured Type Definition

Type Name : MI
Encoding Variation :
Comments : Mobile identity
GSM 04.08, 10.5.1.4

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00010111’B
iel	LENGTH		
idigit_1	BITSTRING [4]		first digit
oei	BITSTRING [1]		odd / even identity length
toi	BITSTRING [3]		type of identity
idigits_other	OCTETSTRING [0..7]		other digits

Detailed Comments :

Structured Type Definition

Type Name : MODEMTYPE
Encoding Variation :
Comments :

Element Name	Type Definition	Field Encoding	Comments
V22bisOrV26ter	B_5		

Detailed Comments :

Structured Type Definition

Type Name : MSCLM1
Encoding Variation :
Comments : mobile station classmark 1
GSM 04.08, 10.5.1.5

Element Name	Type Definition	Field Encoding	Comments
spr1	BITSTRING [1]		spare bit ’0’B
rl	BITSTRING [2]		revision level
spr2	BITSTRING [1]		spare bit ’0’B
a5_1	BITSTRING [1]		a5/1 algorithm supported
rfpc	BITSTRING [3]		rf power capability

Detailed Comments : 2 octets long

Structured Type Definition

Type Name : MSCLM2
Encoding Variation :
Comments : mobile station classmark 2
GSM 04.08, 10.5.1.6

Element Name	Type Definition	Field Encoding	Comments
iel	LENGTH		OCTETSTRING [1]
spr1	BITSTRING [1]		spare bit ’0’B
rl	BITSTRING [2]		revision level
spr2	BITSTRING [1]		spare bit ’0’B
a5_1	BITSTRING [1]		a5/1 algorithm supported
rfpc	BITSTRING [3]		rf power capability
spr3	BITSTRING [1]		spare bit ’0’B
psc	BITSTRING [1]		pseudo synchronisation capability
sssi	BITSTRING [2]		SS screen indicator
smc	BITSTRING [1]		short message capability
spr4	BITSTRING [2]		spare bit ’00’B
fc	BITSTRING [1]		frequency capability
cm3	BITSTRING [1]		classmark 3 indicator
spr5	BITSTRING [5]		spare bit ’00000’B
a5_3	BITSTRING [1]		a5/3 algorithm supported
a5_2	BITSTRING [1]		a5/2 algorithm supported

Detailed Comments : 4 octets long

Structured Type Definition

Type Name : MSCLM3

Encoding Variation :

Comments : Mobile station classmark 3
GSM 04.08, 10.5.1.7

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'00100000'B
iel	LENGTH		OCTETSTRING [1]
spr1	BITSTRING [4]		spare bits
a5_7	BITSTRING [1]		a5/7 algorithm supported
a5_6	BITSTRING [1]		a5/6 algorithm supported
a5_5	BITSTRING [1]		a5/5 algorithm supported
a5_4	BITSTRING [1]		a5/4 algorithm supported
spr2	OCTETSTRING [11]		

Detailed Comments : The info element has 14 octets

Structured Type Definition

Type Name : MSRR

Encoding Variation :

Comments : measurement results
GSM 04.08, 10.5.2.20

Element Name	Type Definition	Field Encoding	Comments
ba_used	BITSTRING [1]		bcch allocation used
dtx_used	BITSTRING [1]		dtx was used
rxlev_fsc	RXLEV		received signal strength on the full serving cell
spr1	BITSTRING [1]		spare bit
meas_valid	BITSTRING [1]		measurement results valid
rxlev_ssc	RXLEV		received signal strength on a subset of the serving cell
spr2	BITSTRING [1]		spare bit
rxqual_fsc	BITSTRING [3]		received signal quality on the full serving cell
rxqual_ssc	BITSTRING [3]		received signal quality on a subset of the serving cell
no_nc	BITSTRING [3]		number of neighbouring cells (nc)
rxlev_nc1	RXLEV		received signal strength on nc1
bcchfrq_nc1	BCCHFRQ		bcch frequency position of nc1
bsic_nc1	BSIC		base station identity code of nc1
rxlev_nc2	RXLEV		received signal strength on nc2
bcchfrq_nc2	BCCHFRQ		bcch frequency position of nc2
bsic_nc2	BSIC		base station identity code of nc2
rxlev_nc3	RXLEV		received signal strength on nc3
bcchfrq_nc3	BCCHFRQ		bcch frequency position of nc3
bsic_nc3	BSIC		base station identity code of nc3
rxlev_nc4	RXLEV		received signal strength on nc4
bcchfrq_nc4	BCCHFRQ		bcch frequency position of nc4
bsic_nc4	BSIC		base station identity code of nc4
rxlev_nc5	RXLEV		received signal strength on nc5
bcchfrq_nc5	BCCHFRQ		bcch frequency position of nc5
bsic_nc5	BSIC		base station identity code nc5
rxlev_nc6	RXLEV		received signal strength on nc6
bcchfrq_nc6	BCCHFRQ		bcch frequency position of nc6

Continued on next page

Continued from previous page

Structured Type Definition			
Element Name	**Type Definition**	**Field Encoding**	**Comments**
bsic_nc6	BSIC		base station identity code of nc6
Detailed Comments : The info element has a fixed length of 16 octets.			

Structured Type Definition			
Type Name : MTDIF			
Encoding Variation :			
Comments : Mobile time difference GSM 04.08, 10.5.1.21a			
Element Name	**Type Definition**	**Field Encoding**	**Comments**
iei	IEI_8		'01110111'B
iel	LENGTH		OCTETSTRING [1]
value	OCTETSTRING [3]		one unit = 24/13 microseconds (in half bit periods)
Detailed Comments :			

Structured Type Definition			
Type Name : NCD			
Encoding Variation :			
Comments : Neighbour cells description GSM 04.08, 10.5.2.22			
Element Name	**Type Definition**	**Field Encoding**	**Comments**
rfl2	BITSTRING [2]		2 reference frequencies
extind	BITSTRING [1]		extension indication
baind	BITSTRING [1]		bcch allocation sequence number indication
rfl4	BITSTRING [4]		4 reference frequencies
rfl	OCTETSTRING [15]		remaining reference frequency list
Detailed Comments :			

Structured Type Definition

Type Name : NCD2
Encoding Variation :
Comments : Neighbour cells description2
GSM 04.08, 10.5.2.22b

Element Name	Type Definition	Field Encoding	Comments
b128	BITSTRING [1]		Bit 128, format−ID
mbrpt	BITSTRING [2]		Multiband reporting
baind	BITSTRING [1]		bcch allocation sequence number indication
b122_124	BITSTRING [3]		Bit 124 − 122, format ID next
rfl1	BITSTRING [1]		1 reference frequency
rfl	OCTETSTRING [15]		remaining reference frequency list

Detailed Comments :

Structured Type Definition

Type Name : PCMD
Encoding Variation :
Comments : Power command and access type
GSM 04.08, 10.5.2.28, 10.5.2.28a

Element Name	Type Definition	Field Encoding	Comments
sprb	BITSTRING [3]		3 spare bits for the power command, ATC (asccess type control) and 2 spare bits for the power command and access type
pl	BITSTRING [5]		power level (0−31)

Detailed Comments :

Structured Type Definition

Type Name : PI
Encoding Variation :
Comments : Progress indicator
GSM 04.08, 10.5.4.21

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’00011110’B
iel	LENGTH		’00’O for non hopping
extb3	EXTB		extension bit (’0’B)
cs	BITSTRING [2]		coding standard
spb	SPB		spare bit
loc	BITSTRING [4]		location
extb4	EXTB		extension bit (’0’B)
prd	BITSTRING [7]		progress description

Detailed Comments : total 4 octets

Structured Type Definition

Type Name : PI_SI
Encoding Variation :
Comments : Presentation indicator & screening indicator
GSM 04.08, 10.5.4.9, 10.5.4.13

Element Name	Type Definition	Field Encoding	Comments
extb	EXTB		extension bit
pi	BITSTRING [2]		presentation indicator, ’0 − 2’
sp3b	SP3B		3 spare bits
si	BITSTRING [2]		screening indicator

Detailed Comments :

Structured Type Definition

Type Name : PM
Encoding Variation :
Comments : Page mode
GSM 04.08, 10.5.2.26

Element Name	Type Definition	Field Encoding	Comments
sprb	B_2		spare bits
pgm	B_2		page mode

Detailed Comments :

Structured Type Definition

Type Name : RACHCP
Encoding Variation :
Comments : Rach control parameters
GSM 04.08, 10.5.2.29

Element Name	Type Definition	Field Encoding	Comments
maxrtx	BITSTRING [2]		maximum retransmissions
txint	BITSTRING [4]		Tx integer, number of slots to spread transmission
cba	BITSTRING [1]		cell barred for access
re	BITSTRING [1]		call reestablishment allowed
acc_2	BITSTRING [5]		access control class 15−11
ec	BITSTRING [1]		emergency call allowed
acc_1	BITSTRING [10]		access control class 9−0

Detailed Comments : The info element has a fixed length of 3 octets.

Structured Type Definition

Type Name : RPACK

Encoding Variation :

Comments : SMS RP ACKNOWLEDGEMENT ms <−> n
GSM 04.11, 7.3.3

Element Name	Type Definition	Field Encoding	Comments
sprb	SP5B		spare 5 bits M
rpmti	MTI		message type indicator M BITSTRING [3]
rpmr	MR		message reference M OCTETSTRING [1]

Detailed Comments :

Structured Type Definition

Type Name : RPCAU

Encoding Variation :

Comments : RP user data element
GSM 04.11, 8.2.5.4

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01000010’B, OMIT
iel	LENGTH		
extb2	EXTB		extension bit
rpcau_class	BITSTRING [3]		cause class
rpcau_va	BITSTRING [4]		cause value
rpcau_di	OCTETSTRING [1]		Diagnostics

Detailed Comments :

Structured Type Definition

Type Name : RPDATA

Encoding Variation :

Comments : SMS RP DATA ms <−> n
GSM 04.11, 7.3.1

Element Name	Type Definition	Field Encoding	Comments
sprb	SP5B		spare 5 bits M
rpmti	MTI		message type indicator M BITSTRING [3]
rpmr	MR		message reference M OCTETSTRING [1]
rpOaddr	CDPN		RP originator address M
rpDaddr	CDPN		RP destination address M
rpusrdat	RPUSRDAT		RP−User data element M OCTETSTRING [1..234]

Detailed Comments :

Structured Type Definition

Type Name : RPERR
Encoding Variation :
Comments : SMS RP ERROR ms <−> n
GSM 04.11, 7.3.4

Element Name	Type Definition	Field Encoding	Comments
sprb	SP5B		spare 5 bits M
rpmti	MTI		message type indicator M BITSTRING [3]
rpmr	MR		message reference M OCTETSTRING [1]
rpcau	RPCAU		RP cause 2−3 octets M
rpusrdat	RPUSRDAT		RP−User data element O OCTETSTRING [0..240]

Detailed Comments :

Structured Type Definition

Type Name : RPSMMA
Encoding Variation :
Comments : SMS RP SMMA ms −> n
GSM 04.11, 7.3.2

Element Name	Type Definition	Field Encoding	Comments
sprb	SP5B		spare 5 bits M
rpmti	MTI		message type indicator M BITSTRING [3]
rpmr	MR		message reference M OCTETSTRING [1]

Detailed Comments : SM memory available

Structured Type Definition

Type Name : RPUSRDAT
Encoding Variation :
Comments : RP user data element
GSM 04.11, 8.2.5.3, GSM 03.40, 9.2.2

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01000001'B
iel	LENGTH		
tpdeliver	SMDLVR		max. 165 bytes, n −> ms, O
tpsubmit	SMSBMT		max. 164 bytes, ms −> n, O
tpstatus_rpt	SMST_RPT		max. 29 bytes, O
tpcommand	SMCMD		max. 162 bytes, O
tpdlvr_sbmt_rpt	SMDLVR_RPT		2 bytes, n <−> ms, O

Detailed Comments : One of the six tpdu is contained in the RPUSRDATA. Since the structures of the messages SMS−Deliver−Report and SMS−Submit−Report are identical, they have been combined to tpdlvr_sbmt_rpt and therefore only five tp message types exist in this type definition.

Structured Type Definition

Type Name : RQR

Encoding Variation :

Comments : Request reference
GSM 04.08, 10.5.2.30

Element Name	Type Definition	Field Encoding	Comments
ra	BITSTRING [8]		GSM 04.08, 9.1.8
fn	FN		

Detailed Comments :

Structured Type Definition

Type Name : SI3RO

Encoding Variation :

Comments : SI3 Rest Octets
GSM 04.08, 10.5.2.34

Element Name	Type Definition	Field Encoding	Comments
pi	BITSTRING [1]		Cell reselect parameter indicator
cbq	BITSTRING [1]		Cell bar qualify
cro	BITSTRING [6]		Cell reselect offset
to	BITSTRING [3]		Temporary offset
pt	BITSTRING [5]		Penalty time
poi	BITSTRING [1]		Power offset indicator
po	BITSTRING [2]		Power offset
ti	BITSTRING [1]		SI 2ter indicator
ecsc	BITSTRING [1]		Early classmark setting control
spr1	BITSTRING [3]		spare bits
spr2	BITSTRING [8]		spare octet

Detailed Comments : The info element has a fixed length of 4 octets.

Structured Type Definition

Type Name : SI4RO

Encoding Variation :

Comments : SI4 Rest Octets
GSM 04.08, 10.5.2.35

Element Name	Type Definition	Field Encoding	Comments
pi	BITSTRING [1]		Cell reselect parameter indicator
cbq	BITSTRING [1]		Cell bar qualify
cro	BITSTRING [6]		Cell reselect offset
to	BITSTRING [3]		Temporary offset
pt	BITSTRING [5]		Penalty time
poi	BITSTRING [1]		Power offset indicator
po	BITSTRING [2]		Power offset
spr1	BITSTRING [5]		spare bits
spr2	BITSTRING [8]		spare octet
spr3	BITSTRING [8]		spare octet
spr4	BITSTRING [8]		spare octet
spr5	BITSTRING [8]		spare octet
spr6	BITSTRING [8]		spare octet
spr7	BITSTRING [8]		spare octet
spr8	BITSTRING [8]		spare octet

Detailed Comments : The info element has a fixed length of 10 octets.

Structured Type Definition

Type Name : SI7RO

Encoding Variation :

Comments : SI7 Rest Octets
GSM 04.08, 10.5.2.36

Element Name	Type Definition	Field Encoding	Comments
pi	BITSTRING [1]		Cell reselect parameter indicator
cbq	BITSTRING [1]		Cell bar qualify
cro	BITSTRING [6]		Cell reselect offset
to	BITSTRING [3]		Temporary offset
pt	BITSTRING [5]		Penalty time
poi	BITSTRING [1]		Power offset indicator
po	BITSTRING [2]		Power offset
spr1	BITSTRING [5]		spare bits
spr2	BITSTRING [8]		spare octet
spr3	BITSTRING [8]		spare octet
spr4	BITSTRING [8]		spare octet
spr5	BITSTRING [8]		spare octet
spr6	BITSTRING [8]		spare octet
spr7	BITSTRING [8]		spare octet
spr8	BITSTRING [8]		spare octet
spr9	BITSTRING [8]		spare octet
spr10	BITSTRING [8]		spare octet
spr11	BITSTRING [8]		spare octet
spr12	BITSTRING [8]		spare octet
spr13	BITSTRING [8]		spare octet
spr14	BITSTRING [8]		spare octet
spr15	BITSTRING [8]		spare octet
spr16	BITSTRING [8]		spare octet
spr17	BITSTRING [8]		spare octet
spr18	BITSTRING [8]		spare octet

Detailed Comments : The info element has a fixed length of 20 octets.

Structured Type Definition

Type Name : SIGNAL

Encoding Variation :

Comments : Signal (CC information element)
GSM 04.08, 10.5.4.23

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'00110100'B
sigv	BITSTRING [8]		signal value

Detailed Comments :

Structured Type Definition

Type Name : SERIAL_NUMBER
Encoding Variation :
Comments : Serial number for SMSCB,
GSM 3.41, 9.3.2

Element Name	Type Definition	Field Encoding	Comments
gs	BITSTRING[2]		
message_code	BITSTRING[10]		
update_number	BITSTRING[4]		

Detailed Comments :

Structured Type Definition

Type Name : SMCMD
Encoding Variation :
Comments : SMS COMMAND, ms − > n.
GSM 03.40, 9.2.2.4

Element Name	Type Definition	Field Encoding	Comments
sprb1	SP2B		2 spare bits
srr	BITSTRING[1]		TP Status Report Request
sprb2	SP3B		3 spare bits
mti	BITSTRING [2]		TP message type indicator
mr	MR		TP message reference
pid	TPPID		TP protocol identifier
ct	OCTETSTRING [1]		TP command type, ’0−3’
mn	OCTETSTRING [1]		TP message number
da	TPA		TP destination address
cdl	LENGTH		TP command data length
cd	TPCD		TP command data, max 157 octets

Detailed Comments :

Structured Type Definition

Type Name : SMDLVR

Encoding Variation :

Comments : SMS DELIVER, n − > ms.
GSM 03.40, 9.2.2.1

Element Name	Type Definition	Field Encoding	Comments
rp	BITSTRING [1]		TP reply path, M
udhi	BITSTRING [1]		TP User Data Header Indication
sri	BITSTRING [1]		TP status report indication, O
sprb2	SP2B		2 spare bits
mms	BITSTRING [1]		TP more message to send, M
mti	BITSTRING [2]		TP message type indicator, M
oa	TPA		TP originator address
pid	TPPID		TP protocol identifier, M
dcs	TPDCS		TP data coding scheme, M
scts	TPSCTS		TP service centre time stamp
udl	LENGTH		TP user data length
ud	TPUD		TP user data

Detailed Comments :

Structured Type Definition

Type Name : SMDLVR_RPT

Encoding Variation :

Comments : SMS DELIVER or SUBMIT REPORT contained in the RP ERROR PDU, n <−> ms.
GSM 03.40, 9.2.2.1a, 9.2.2.2a.

Element Name	Type Definition	Field Encoding	Comments
sprb1	SP6B		6 spare bits
mti	BITSTRING [2]		TP message type indicator, M
fcs	FCS		TP failure cause

Detailed Comments :

Structured Type Definition

Type Name : SMSBMT
Encoding Variation :
Comments : SMS SUBMIT, ms −> n.
GSM 03.40, 9.2.2.2

Element Name	Type Definition	Field Encoding	Comments
rp	BITSTRING [1]		TP reply path indication
udhi	BITSTRING [1]		
srr	BITSTRING [1]		TP status report request
vpf	BITSTRING [2]		TP validity period format
rd	BITSTRING [1]		
mti	BITSTRING [2]		TP message type indicator
mr	MR		TP message reference
da	TPA		TP destination address
pid	TPPID		TP protocol identifier, M
dcs	TPDCS		TP data coding scheme, M
vp1	OCTETSTRING [1]		TP validity period, 1 octet
vp7	TPSCTS		TP validity period, 7 octets
udl	LENGTH		TP user data length
ud	TPUD		TP user data

Detailed Comments :

Structured Type Definition

Type Name : SMSCBpack
Encoding Variation :
Comments : SMS cell broadcasting packing data n −> ms.
GSM 03.38, 6.1.2.2.1

Element Name	Type Definition	Field Encoding	Comments
block1	OCTETSTRING [16]		1st block
block2	OCTETSTRING [22]		2nd block
block3	OCTETSTRING [22]		3rd block
block4	OCTETSTRING [22]		4th block

Detailed Comments :

Structured Type Definition

Type Name : SMST_RPT
Encoding Variation :
Comments : SMS STATUS REPORT, n − > ms.
GSM 03.40, 9.2.2.3

Element Name	Type Definition	Field Encoding	Comments
sprb1	SP5B		5 spare bits
mms	BITSTRING [1]		TP more message to send, M
mti	BITSTRING [2]		TP message type indicator, M
mr	MR		TP message reference
ra	TPA		TP recipient address
scts	TPSCTS		TP service centre time stamp
dt	TPSCTS		TP discharge time
st	TPST		TP status

Detailed Comments :

Structured Type Definition

Type Name : SSVI
Encoding Variation :
Comments : SS version indicator
GSM 04.08, 10.5.4.24

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01111111’B
iel	LENGTH		’00’O for non hopping
ssv	OCTETSTRING [1]		ss version information

Detailed Comments :

Structured Type Definition

Type Name : STRT
Encoding Variation :
Comments : Starting time
GSM 04.08, 10.5.2.38

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01111101’B
fn	FN		

Detailed Comments :

Structured Type Definition

Type Name : SUBAD
Encoding Variation :
Comments : Subaddress
GSM 04.08, 10.5.4.8, 10.5.4.10, 10.5.4.14

Element Name	Type Definition	Field Encoding	Comments
extb	EXTB		extension bit
tos	BITSTRING [3]		Type of subaddress, ’0, 2’
oei	BITSTRING [1]		odd/even indicator
sp3b	SP3B		3 spare bits
si	OCTETSTRING [2..23]		subaddress information

Detailed Comments :

Structured Type Definition

Type Name : SYNCHI
Encoding Variation :
Comments : Synchronization indication
GSM 04.08, 10.5.2.39

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_4		’1101’B
nci	B_1		normal cell indication
rot	B_1		report observed time difference
si	B_2		synchronisation indication

Detailed Comments :

Structured Type Definition

Type Name : TA
Encoding Variation :
Comments : Timing advance
GSM 04.08, 10.5.2.40

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01111101’B
sprb	BITSTRING [2]		’00’B
value	BITSTRING [6]		one unit = 48/13 microseconds (in 1 bit periods)

Detailed Comments :

Structured Type Definition

Type Name : TDIF

Encoding Variation :

Comments : Time difference
GSM 04.08, 10.5.1.41

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01111011'B
iel	LENGTH		OCTETSTRING [1]
value	OCTETSTRING [1]		one unit = 24/13 microseconds (in half bit periods)

Detailed Comments :

Structured Type Definition

Type Name : TI

Encoding Variation :

Comments : Transaction identifier
GSM 04.08, 10.1

Element Name	Type Definition	Field Encoding	Comments
ti_f	BITSTRING [1]		Flag
ti_v	TI_V		TI value

Detailed Comments :

Structured Type Definition

Type Name : TMSI

Encoding Variation :

Comments : Temporary
GSM 04.08, 10.5.2.42

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'01111110'B
tmsi_val	OCTETSTRING [4]		

Detailed Comments :

Structured Type Definition

Type Name : TON_NPI
Encoding Variation :
Comments : Type of number and numbering plan
GSM 04.08, 10.5.4.7, 10.5.4.9, 10.5.4.13, GSM 03.40, 9.1.2.5

Element Name	Type Definition	Field Encoding	Comments
extb	EXTB		extension bit
ton	BITSTRING [3]		Type of number, ’0 − 6’
npi	BITSTRING [4]		Numbering plan id., ’0, 1, 3, 4, 8, 9, 10’

Detailed Comments :

Structured Type Definition

Type Name : TPA
Encoding Variation :
Comments : TP address
GSM 03.40, 9.1.2.5

Element Name	Type Definition	Field Encoding	Comments
iel	LENGTH		Integer representation of useful semi−octets
tonnpi	TON_NPI		Type of number and numbering plan
digits	BCDN		BCD numbers

Detailed Comments :

Structured Type Definition

Type Name : TPDCS
Encoding Variation :
Comments : SMS data coding scheme
GSM 03.38, 4, 5

Element Name	Type Definition	Field Encoding	Comments
cg	BITSTRING [4]		coding group ’0000’ or ‘1111’
code	BITSTRING [4]		usage of the bits 0−4

Detailed Comments : Identifying the coding scheme within the TP user data.

Structured Type Definition

Type Name : TPPID
Encoding Variation :
Comments : TP protocol identifier
GSM 03.40, 9.2.3.9

Element Name	Type Definition	Field Encoding	Comments
type	BITSTRING [2]		PID type
value	BITSTRING [6]		PID value

Detailed Comments :

Structured Type Definition

Type Name : TPST
Encoding Variation :
Comments : TP status
GSM 03.40, 9.2.3.15

Element Name	Type Definition	Field Encoding	Comments
sprb1	SPB		spare bit
value	BITSTRING [7]		status value/usage

Detailed Comments :

Structured Type Definition

Type Name : UNKWN
Encoding Variation :
Comments : unknown IE

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		
iel	LENGTH		
contents	OCTETSTRING		

Detailed Comments :

Structured Type Definition

Type Name : UU

Encoding Variation :

Comments : User−user
GSM 04.08, 10.5.4.25

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01111110’B
iel	LENGTH		
uupd	BITSTRING [8]		user−user protocol discriminator
uui	OCTETSTRING [1..128]		user user information

Detailed Comments : In SETUP, ALERTING, CONNECT, DISCONNECT, RELEASE and RELEASE COMPLETE messages the uui length is of 0 − 32 bytes.
In USER INFORMATION messages the uui length is of 1 − 128.

ASN.1 Type Definition

Type Name : Component

Encoding Variation :

Comments : ASN1_Encoding: BER

Type Definition

```
CHOICE {
   registerSSComponents                 RegisterSS_Components,
   eraseSSComponents                    EraseSS_Components,
   activateSSComponents                 ActivateSS_Components,
   deactivateSSComponents               DeactivateSS_Components,
   interrogateSSComponents              InterrogateSS_Components,
   notifySSComponents                   NotifySS_Components,
   registerPasswordComponents           RegisterPassword_Components,
   getPasswordComponents                GetPassword_Components,
   processUnstructuredSSDataComponents  ProcessUnstructuredSSData_Components,
   forwardCheckSSIndicationComponents   ForwardCheckSSIndication_Components,
   processUnstructuredSSRequestComponents ProcessUnstructuredSSRequest_Components,
   unstructuredSSRequestComponents      UnstructuredSSRequest_Components,
   unstructuredSSNotifyComponents       UnstructuredSSNotify_Components,
   forwardCUGInfoComponents             ForwardCUGInfo_Components,
   splitMPTYComponents                  SplitMPTY_Components,
   retrieveMPTYComponents               RetrieveMPTY_Components,
   holdMPTYComponents                   HoldMPTY_Components,
   buildMPTYComponents                  BuildMPTY_Components,
   forwardChargeAdviceComponents        ForwardChargeAdvice_Components,
   generalComponents                    General_Components                    }
```

Detailed Comments : Plural components as each type represents invoke, return result, return error etc.

ASN.1 Type Definition

Type Name : Components

Encoding Variation :

Comments : GSM 04.80, 3.6

Type Definition

```
SET OF Component
```

Detailed Comments : ASN.1 transfer encoding rules:
BER is not wholy used for the type Components. The contents of Components, without the octets encoding the tag and the length of SET OF, is carried as the Components value.

ASN.1 Type Definition

Type Name : ActivateSS_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    activateSS_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue               INTEGER (12),
                    ss_ForBS                SS_ForBS_Code },

    activateSS_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    result                  SEQUENCE {
                        localValue                  INTEGER (12),
                        ss_Info             SS_Info       }     } ,

    activateSS_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( bearerServiceNotProvisioned
                                                                                          | teleserviceNotProvisioned
                                                                                          | illegalSS_Operation
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue
                                                                                          | negativePW_Check
                                                                                          | numberOfPW_AttemptsViolation) },
        callBarredErr               [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( callBarred ),
                                                            parameter     CallBarringCause },
        ss_ErrorStatusErr           [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( ss_ErrorStatus ),
                                                            parameter     SS_Status },
        ss_SubscriptionViolationeErr [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues (ss_SubscriptionViolation),
                                                            parameter     SS_SubscriptionOption },
        ss_IncompatibilityErr       [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( ss_Incompatibility ),
                                                            parameter     SS_IncompatibilityCause },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    activateSS_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m (15 − 30 s)

ASN.1 Type Definition

Type Name : BuildMPTY_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    buildMPTY_InvokeComp          [1] IMPLICIT SEQUENCE {
                  invokeID                InvokeIDType,
                  localValue               INTEGER (124)  },

    buildMPTY_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                  invokeID              InvokeIDType,
                  result                SEQUENCE {
                       localValue                INTEGER (124) }     } ,

    buildMPTY_ReturnErrorComp   CHOICE {
        errorCodes                [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                          errorCode    ErrorLocalValues ( illegalSS_Operation
                                                                     | ss_NotAvailable
                                                                     | maxNumberOfMPTY_ParticipantsExceeded
                                                                     | resourcesNotAvailable ) },
        ss_ErrorStatusErr         [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                          errorCode    ErrorLocalValues ( ss_ErrorStatus ),
                                                          parameter     SS_Status },
        ss_IncompatibilityErr     [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                          errorCode    ErrorLocalValues ( ss_Incompatibility ),
                                                          parameter     SS_IncompatibilityCause },
        systemFailureErr          [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                          errorCode    ErrorLocalValues ( systemFailure ),
                                                          parameter     NetworkResource}   },

    buildMPTY_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer T_buildMPTY (5 − 30 s)

ASN.1 Type Definition

Type Name : DeactivateSS_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    deactivateSS_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID              InvokeIDType,
                    localValue             INTEGER (13),
                    ss_ForBS              SS_ForBS_Code },

    deactivateSS_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                     invokeID              InvokeIDType,
                     result                SEQUENCE {
                          localValue                INTEGER (13),
                          ss_Info           SS_Info       }     } ,

    deactivateSS_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( bearerServiceNotProvisioned
                                                                                          | teleserviceNotProvisioned
                                                                                          | illegalSS_Operation
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue
                                                                                          | negativePW_Check
                                                                                          | numberOfPW_AttemptsViolation) },
        callBarredErr               [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( callBarred ),
                                                            parameter     CallBarringCause },
        ss_ErrorStatusErr           [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                             errorCode    ErrorLocalValues ( ss_ErrorStatus ),
                                                             parameter     SS_Status },
        ss_SubscriptionViolationeErr [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues (ss_SubscriptionViolation),
                                                              parameter     SS_SubscriptionOption },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    deactivateSS_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : EraseSS_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    eraseSS_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue              INTEGER (11),
                    ss_ForBS                SS_ForBS_Code },

    eraseSS_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    result                  SEQUENCE {
                        localValue                  INTEGER (11),
                        ss_Info             SS_Info    }          } ,

    eraseSS_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues (  bearerServiceNotProvisioned
                                                                                          | teleserviceNotProvisioned
                                                                                          | illegalSS_Operation
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue ) },
        callBarredErr               [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( callBarred ),
                                                            parameter     CallBarringCause },
        ss_ErrorStatusErr           [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( ss_ErrorStatus ),
                                                            parameter     SS_Status },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },


    eraseSS_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : ForwardChargeAdvice_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    forwardChargeAdvice_InvokeComp          [1] IMPLICIT SEQUENCE {
                invokeID                        InvokeIDType,
                localValue                      INTEGER (125),
                forwardChargeAdviceArg          SEQUENCE {
                        ss_Code                 [0] SS_Code,
                        chargingInformation     [1]  SEQUENCE{
                                e1      [1] INTEGER (0..max10TimesUnitsPerTime)                 OPTIONAL,
                                e2      [2]  INTEGER (0..max10TimesTimeInterval)                OPTIONAL,
                                e3      [3]  INTEGER (0..max100TimesScalingFactor)              OPTIONAL,
                                e4      [4] INTEGER (0..max10TimesIncrement)                    OPTIONAL,
                                e5      [5]  INTEGER (0..max10TimesIncrementPerDataInterval)    OPTIONAL,
                                e6      [6] INTEGER (0..maxNumberOfSegmentsPerDataInterval) OPTIONAL,
                                e7      [7]  INTEGER (0..max10TimesInitialTime)                 OPTIONAL } } },

    forwardChargeAdvice_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                invokeID                InvokeIDType,
                result                  SEQUENCE {
                localValue                INTEGER (125)  }    } ,

    forwardChargeAdvice_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer T_aoc = 1 − 40 s

ASN.1 Type Definition

Type Name : ForwardCheckSSIndication_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    forwardCheckSSIndication_InvokeComp         [1] IMPLICIT SEQUENCE {
                invokeID                InvokeIDType,
                localValue                INTEGER (38)    },

    forwardCheckSSIndication_RejectComp         [4] IMPLICIT RejectComponent       }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ForwardCUGInfo_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    forwardCUGInfo_InvokeComp       [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue              INTEGER (120),
                    forwardCUGInfo_Arg          SEQUENCE{
                        cug_Index           [0]  IMPLICIT CUG_Index OPTIONAL,
                        suppressPrefCUG     [1]  IMPLICIT NULL          OPTIONAL,
                        suppressOA          [2]  IMPLICIT NULL          OPTIONAL } },

    forwardCUGInfo_RejectComp       [4] IMPLICIT RejectComponent      }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : General_Components

Encoding Variation :

Comments : Non specified components must match this type definition.

Type Definition

```
CHOICE {
  general_InvokeComp        [1] IMPLICIT General_InvokeComponent,
  general_ReturnResultComp [2] IMPLICIT General_ReturnResultComponent,
  general_ReturnErrorComp  [3] IMPLICIT General_ReturnErrorComponent,
  general_RejectComp        [4] IMPLICIT RejectComponent }


-- This is the General InvokeComponent --
General_InvokeComponent ::= SEQUENCE {
  invokeID                  InvokeIDType,
  linked_ID      [0] IMPLICIT InvokeIDType OPTIONAL,
  operation_value            Operation,
  argument                  ANY     OPTIONAL }

-- This is the General ReturnResultComponent --
General_ReturnResultComponent ::= SEQUENCE {
  invokeID      InvokeIDType,
  valueAndResult SEQUENCE {
                operation_value Operation,
                result          ANY     } OPTIONAL }

-- This is the General ReturnErrorComponent --
General_ReturnErrorComponent ::= SEQUENCE {
  invokeID InvokeIDType,
  error    ANY }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : GetPassword_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    getPassword_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    linkedID                [0] IMPLICIT InvokeIDType,
                    localValue               INTEGER (18),
                    guidanceInfo             ENUMERATED {
                                                enterPW  (0),
                                                enterNewPW  (1),
                                                enterNewPW_Again  (2) } },

    getPassword_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                     invokeID               InvokeIDType,
                     result                 SEQUENCE {
                          localValue                INTEGER (18),
                          currentPassword           Password }          } ,


    getPassword_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : HoldMPTY_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    holdMPTY_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue               INTEGER (123)  },

    holdMPTY_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                     invokeID               InvokeIDType,
                     result                 SEQUENCE {
                          localValue                INTEGER (123) }     } ,

    holdMPTY_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID       InvokeIDType,
                                                            errorCode    ErrorLocalValues ( illegalSS_Operation
                                                                       | facilityNotSupported ) },
        ss_ErrorStatusErr           [3] IMPLICIT SEQUENCE { invokeID       InvokeIDType,
                                                            errorCode    ErrorLocalValues ( ss_ErrorStatus ),
                                                            parameter     SS_Status },
        ss_IncompatibilityErr        [3] IMPLICIT SEQUENCE { invokeID       InvokeIDType,
                                                            errorCode    ErrorLocalValues ( ss_Incompatibility ),
                                                            parameter     SS_IncompatibilityCause },
        systemFailureErr             [3] IMPLICIT SEQUENCE { invokeID       InvokeIDType,
                                                            errorCode    ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    holdMPTY_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer T_holdMPTY (5 − 30 s)

ASN.1 Type Definition

Type Name : InterrogateSS_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    interrogateSS_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID              InvokeIDType,
                    localValue             INTEGER (14),
                    ss_ForBS              SS_ForBS_Code },

    interrogateSS_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                     invokeID              InvokeIDType,
                     result                SEQUENCE {
                          localValue                INTEGER (14),
                          interrogateSS_Res         CHOICE {
                              ss_Status             [0] IMPLICIT SS_Status,
                              basicServiceGroupList   [2] IMPLICIT BasicServiceGroupList,
                              forwardingFeatureList   [3] IMPLICIT ForwardingFeatureList,
                              cli_RestrictionInfo     [4] IMPLICIT SEQUENCE {
                                        ss_Status             SS_Status,
                                        cliRestrictionOption    CliRestrictionOption    OPTIONAL } } } } ,

    interrogateSS_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( bearerServiceNotProvisioned
                                                                                          | teleserviceNotProvisioned
                                                                                          | illegalSS_Operation
                                                                                          | ss_NotAvailable
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue) },
        callBarredErr               [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( callBarred ),
                                                            parameter     CallBarringCause },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },


    interrogateSS_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : NotifySS_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    notifySS_InvokeComp         [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue               INTEGER (16),
                    notifySS_Arg            SEQUENCE{
                        ss_Code                 [1] IMPLICIT SS_Code                    OPTIONAL,
                        ss_Status               [4] IMPLICIT SS_Status                  OPTIONAL,
                        ss_Notification         [5] IMPLICIT OCTET STRING (SIZE (1)) OPTIONAL,
                        callIsWaiting_Indicator [14] IMPLICIT NULL                      OPTIONAL,
                        callOnHold_Indicator    [15] IMPLICIT ENUMERATED {
                                                        callRetrieved (0),
                                                        callOnHold (1) }        OPTIONAL,
                        mpty_Indicator          [16] IMPLICIT NULL                      OPTIONAL,
                        cug_Index               [17]  IMPLICIT CUG_Index                OPTIONAL,
                        clirSuppressionRejected [18]  IMPLICIT NULL                     OPTIONAL }    },

    notifySS_RejectComp         [4] IMPLICIT RejectComponent       }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ProcessUnstructuredSSData_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    processUnstructuredSSData_InvokeComp        [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue               INTEGER (19),
                    ss_UserData             SS_UserData },

    processUnstructuredSSData_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                     invokeID               InvokeIDType,
                     result                 SEQUENCE {
                        localValue                  INTEGER (19),
                        ss_UserData                 SS_UserData   } } ,

    processUnstructuredSSData_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues (unexpectedDataValue) },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    processUnstructuredSSData_RejectComp        [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : ProcessUnstructuredSSRequest_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    processUnstructuredSSRequest_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                  InvokeIDType,
                    localValue                 INTEGER (59),
                    ussd_Arg                  USSD_Arg },

    processUnstructuredSSRequest_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                     invokeID                 InvokeIDType,
                     result                   SEQUENCE {
                          localValue                  INTEGER (59),
                          ussd_Res                    USSD_Res}          } ,

    processUnstructuredSSRequest_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues (dataMissing
                                                                                         | unexpectedDataValue
                                                                                         | unknownAlphabet) },
        callBarredErr               [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( callBarred ),
                                                            parameter     CallBarringCause },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode     ErrorLocalValues ( systemFailure ),
                                                            parameter      NetworkResource}   },

    processUnstructuredSSRequest_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : RegisterPassword_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    registerPassword_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                 InvokeIDType,
                    localValue                INTEGER (17),
                    ss_Code                  SS_Code     },

    registerPassword_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                 InvokeIDType,
                    result                   SEQUENCE {
                        localValue                   INTEGER (17),
                        newPassword                  Password    }       } ,

    registerPassword_ReturnErrorComp   CHOICE {
        errorCodes                   [3] IMPLICIT SEQUENCE { invokeID    InvokeIDType,
                                                             errorCode     ErrorLocalValues ( dataMissing
                                                                                            | unexpectedDataValue
                                                                                            | negativePW_Check
                                                                                            | numberOfPW_AttemptsViolation) },
        callBarredErr                [3] IMPLICIT SEQUENCE { invokeID    InvokeIDType,
                                                             errorCode     ErrorLocalValues ( callBarred ),
                                                             parameter     CallBarringCause },
        ss_SubscriptionViolationeErr [3] IMPLICIT SEQUENCE { invokeID    InvokeIDType,
                                                             errorCode     ErrorLocalValues
(ss_SubscriptionViolation),
                                                             parameter     SS_SubscriptionOption },
        systemFailureErr             [3] IMPLICIT SEQUENCE { invokeID    InvokeIDType,
                                                              errorCode    ErrorLocalValues ( systemFailure ),
                                                              parameter    NetworkResource},
        pw_RegistrationFailureErr    [3] IMPLICIT SEQUENCE { invokeID    InvokeIDType,
                                                             errorCode     ErrorLocalValues ( pw_RegistrationFailure ),
                                                             parameter     PW_RegistrationFailureCause }   },

    registerPassword_RejectComp      [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : RegisterSS_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    registerSS_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue              INTEGER (10),
                    registerSS_Arg          SEQUENCE{
                        ss_Code                 SS_Code,
                        basicService            BasicServiceCode       OPTIONAL,
                        forwardedToNumber       [4] IMPLICIT AddressString      OPTIONAL,
                        forwardedToSubaddress   [6] IMPLICIT ISDN_SubaddressString      OPTIONAL,
                        noReplyConditionTime    [5] IMPLICIT NoReplyConditionTime       OPTIONAL    }   },

    registerSS_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    result                  SEQUENCE {
                        localValue              INTEGER (10),
                        ss_Info             SS_Info         }       } ,

    registerSS_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( bearerServiceNotProvisioned
                                                                                          | teleserviceNotProvisioned
                                                                                          | illegalSS_Operation
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue ) },
        callBarredErr               [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( callBarred ),
                                                            parameter    CallBarringCause },
        ss_IncompatibilityErr       [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( ss_Incompatibility ),
                                                            parameter    SS_IncompatibilityCause },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( systemFailure ),
                                                            parameter    NetworkResource}   },


    registerSS_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : RejectComponent

Encoding Variation :

Comments : ITU−T Q.767

Type Definition

```
SEQUENCE {
 invokedID CHOICE {
              derivable          InvokeIDType,
              notDerivable       NULL },
 problem   CHOICE {
              generalProblem     [0] IMPLICIT GeneralProblem,
              invokeProblem      [1] IMPLICIT InvokeProblem,
              returnResultProblem [2] IMPLICIT ReturnResultProblem,
              returnErrorProblem  [3] IMPLICIT ReturnErrorProblem } }
```

Detailed Comments : Reject Component is not specific to any particular operation. The invokeID may be used to identify a specific operation.

ASN.1 Type Definition

Type Name : RetrieveMPTY_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    retrieveMPTY_InvokeComp          [1] IMPLICIT SEQUENCE {
                  invokeID               InvokeIDType,
                  localValue              INTEGER (122)  },

    retrieveMPTY_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                  invokeID               InvokeIDType,
                  result                 SEQUENCE {
                       localValue                INTEGER (122) }     } ,

    retrieveMPTY_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode   ErrorLocalValues ( illegalSS_Operation
                                                                      | facilityNotSupported ) },
        ss_ErrorStatusErr           [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode   ErrorLocalValues ( ss_ErrorStatus ),
                                                            parameter    SS_Status },
        ss_IncompatibilityErr       [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode   ErrorLocalValues ( ss_Incompatibility ),
                                                            parameter    SS_IncompatibilityCause },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode   ErrorLocalValues ( systemFailure ),
                                                            parameter    NetworkResource}   },

    retrieveMPTY_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer T_retrieveMPTY (5 − 30 s)

ASN.1 Type Definition

Type Name : SplitMPTY_Components

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
CHOICE {
    splitMPTY_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    localValue              INTEGER (121)  },

    splitMPTY_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                InvokeIDType,
                    result                  SEQUENCE {
                        localValue                  INTEGER (121) }     } ,

    splitMPTY_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( illegalSS_Operation
                                                                       | facilityNotSupported ) },
        ss_ErrorStatusErr           [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( ss_ErrorStatus ),
                                                            parameter     SS_Status },
        ss_IncompatibilityErr        [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( ss_Incompatibility ),
                                                            parameter     SS_IncompatibilityCause },
        systemFailureErr             [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    splitMPTY_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer T_splitMPTY (5 − 30 s)

ASN.1 Type Definition

Type Name : UnstructuredSSNotify_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    unstructuredSSNotify_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                 InvokeIDType,
                    localValue               INTEGER (61),
                    ussd_Arg                 USSD_Arg                },

    unstructuredSSNotify_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                 InvokeIDType,
                    result                   SEQUENCE {
                        localValue                   INTEGER (61) }  } ,

    unstructuredSSNotify_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( illegalSubscriber
                                                                                          | illegalEquipment
                                                                                          | absentSubscriber
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue
                                                                                          | unknownAlphabet
                                                                                          | ussd_Busy) },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    unstructuredSSNotify_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : UnstructuredSSRequest_Components

Encoding Variation :

Comments : GSM 09.02, 14.5, 14.6.4, 14.6.6, 14.7.4, 14.7.7

Type Definition

```
CHOICE {
    unstructuredSSRequest_InvokeComp          [1] IMPLICIT SEQUENCE {
                    invokeID                  InvokeIDType,
                    localValue                INTEGER (60),
                    ussd_Arg                  USSD_Arg                },

    unstructuredSSRequest_ReturnResultComp  [2] IMPLICIT SEQUENCE {
                    invokeID                  InvokeIDType,
                    result                    SEQUENCE {
                        localValue                    INTEGER (60),
                        ussd_Res                      USSD_Res      } } ,

    unstructuredSSRequest_ReturnErrorComp   CHOICE {
        errorCodes                  [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( illegalSubscriber
                                                                                          | illegalEquipment
                                                                                          | absentSubscriber
                                                                                          | dataMissing
                                                                                          | unexpectedDataValue
                                                                                          | unknownAlphabet
                                                                                          | ussd_Busy) },
        systemFailureErr            [3] IMPLICIT SEQUENCE { invokeID      InvokeIDType,
                                                            errorCode    ErrorLocalValues ( systemFailure ),
                                                            parameter     NetworkResource}   },

    unstructuredSSRequest_RejectComp          [4] IMPLICIT RejectComponent       }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : AddressString

Encoding Variation :

Comments : GSM 09.02, 14.7.8

Type Definition

```
OCTET STRING (SIZE ( 1 .. maxAddressLength ))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : Asn1Integer

Encoding Variation :

Comments : INTEGER of ASN.1 type

Type Definition

```
INTEGER
```

Detailed Comments :

ASN.1 Type Definition

Type Name : BasicServiceCode
Encoding Variation :
Comments : GSM 09.02, 14.7.8

Type Definition

```
CHOICE {
    bearerService   [2] IMPLICIT BearerServiceCode,
    teleservice     [3] IMPLICIT TeleserviceCode    }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : BasicServiceGroupList
Encoding Variation :
Comments : GSM 09.02, 14.7.8

Type Definition

```
SEQUENCE SIZE ( 1 .. maxNumOfBasicServiceGroups ) OF BasicServiceCode
```

Detailed Comments :

ASN.1 Type Definition

Type Name : BearerServiceCode
Encoding Variation :
Comments : GSM 09.02, 14.7.10

Type Definition

```
OCTET STRING (SIZE (1))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CallBarringCause
Encoding Variation :
Comments : GSM 09.02, 14.7.7

Type Definition

```
ENUMERATED {
    barringServiceActive        (0),
    operatorBarring             (1) }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CallBarringFeature
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    basicService    BasicServiceCode    OPTIONAL,
    ss_Status       [4] IMPLICIT SS_Status          OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CallBarringFeatureList
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE SIZE (1..maxNumOfBasicServiceGroups) OF CallBarringFeature
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CallBarringInfo
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    ss_Code                 SS_Code         OPTIONAL,
    callBarringFeatureList  CallBarringFeatureList      }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CliRestrictionOption
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
ENUMERATED {
    permanent                   (0),
    temporaryDefaultRestricted  (1),
    temporaryDefaultAllowed     (2) }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CUG_Feature
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    basicService                BasicServiceCode        OPTIONAL,
    preferentialCUG_Indicator   CUG_Index               OPTIONAL,
    interCUG_Restrictions       InterCUG_Restrictions               }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CUG_FeatureList
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE SIZE (1.. maxNumOfBasicServiceGroups ) OF CUG_Feature
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CUG_Index
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
INTEGER (0..32767)
```

Detailed Comments : The internal structure is defined in ETS 300 138.

ASN.1 Type Definition

Type Name : CUG_Interlock
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
OCTET STRING (SIZE (4))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CUG_Info

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    cug_SubscriptionList    CUG_SubscriptionList,
    cug_FeatureList         CUG_FeatureList   OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CUG_Subscription

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    cug_Index               CUG_Index,
    cug_Interlock           CUG_Interlock,
    intraCUG_Options        IntraCUG_Options,
    basicServiceGroupList   BasicServiceGroupList   OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : CUG_SubscriptionList

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE SIZE (1.. maxNumOfCUG ) OF CUG_Subscription
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ErrorLocalValues

Encoding Variation :

Comments : GSM 04.80, 4.5

Type Definition

```
INTEGER {
  unknownSubscriber                    (1),
  illegalSubscriber                    (9),
  bearerServiceNotProvisioned          (10),
  teleserviceNotProvisioned            (11),
  illegalEquipment                     (12),
  callBarred                           (13),
  illegalSS_Operation                  (16),
  ss_ErrorStatus                       (17),
  ss_NotAvailable                      (18),
  ss_SubscriptionViolation             (19),
  ss_Incompatibility                   (20),
  facilityNotSupported                 (21),
  absentSubscriber                     (27),
  systemFailure                        (34),
  dataMissing                          (35),
  unexpectedDataValue                  (36),
  pw_RegistrationFailure               (37),
  negativePW_Check                     (38),
  numberOfPW_AttemptsViolation         (43),
  unknownAlphabet                      (71),
  ussd_Busy                            (72),
  maxNumberOfMPTY_ParticipantsExceeded (126),
  resourcesNotAvailable                (127)
 }
```

Detailed Comments : Elements of INTEGER are global for the ATS.

ASN.1 Type Definition

Type Name : ForwardingFeature

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
      basicService              BasicServiceCode                 OPTIONAL,
      ss_Status                 [4] IMPLICIT SS_Status           OPTIONAL,
      forwardedToNumber         [5] IMPLICIT ISDN_AddressString  OPTIONAL,
      forwardedToSubaddress     [8] IMPLICIT ISDN_SubaddressString OPTIONAL,
      forwardingOptions         [6] IMPLICIT ForwardingOptions   OPTIONAL,
      noReplyConditionTime      [7] IMPLICIT NoReplyConditionTime OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ForwardingFeatureList
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE SIZE (1..maxNumOfBasicServiceGroups) OF ForwardingFeature
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ForwardingInfo
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    ss_Code                 SS_Code         OPTIONAL,
    forwardingFeatureList   ForwardingFeatureList }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ForwardingOptions
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
OCTET STRING (SIZE (1))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : GeneralProblem
Encoding Variation :
Comments : ITU−T Q.767

Type Definition

```
TCAP_Problems ( unrecognizedComponent
          | mistypedComponent
          | badlyStructuredComponent )
```

Detailed Comments : Type restricted to these three.

ASN.1 Type Definition

Type Name : InterCUG_Restrictions

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

OCTET STRING (SIZE (1))

Detailed Comments : bits: 876543: 000000 (unused)
bits 21: 00 CUG only facilities
01 CUG with outgoing access
10 CUG with incoming access
11 CUG with both outgoing and incoming access

ASN.1 Type Definition

Type Name : IntraCUG_Options

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

ENUMERATED {
noCUG_Restrictions (0),
cugIC_CallBarred (1),
cugOG_CallBarred (2) }

Detailed Comments :

ASN.1 Type Definition

Type Name : InvokeIDType

Encoding Variation :

Comments : ITU−T, Q.773

Type Definition

INTEGER (−128 .. 127)

Detailed Comments : Values:
Sending Components: If it is an invoke component then use Test Case Variable (with default) to set value. If another invoke component is sent the TCV should be incremented beforehand. If it is a return result, error or reject component in response to a received invoke component then use TCV also, making sure the value is set to the value of the received component beforehand.

Receiving Components: If it is an invoke comp then use '?'. If it is a return result, error or reject component in response to a sent invoke component then use TCV value (as used in sent invoke component).

ASN.1 Type Definition

Type Name : InvokeProblem
Encoding Variation :
Comments : ITU−T Q.767

Type Definition

```
TCAP_Problems ( duplicateInvokeID
        | unrecognizedOperation
        | mistypedArgument
        | resourceLimitation
        | initiatingRelease
        | unrecognizedLinkedID
        | linkedResponseUnexpected
        | unexpectedLinkedOperation )
```

Detailed Comments : Type restricted to these 8.

ASN.1 Type Definition

Type Name : ISDN_AddressString
Encoding Variation :
Comments : GSM 09.02, 14.7.8

Type Definition

```
AddressString (SIZE ( 1 .. maxISDN_AddressLength ))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ISDN_SubaddressString
Encoding Variation :
Comments : GSM 09.02, 14.7.8

Type Definition

```
OCTET STRING (SIZE ( 1 .. maxISDN_SubaddressLength ))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : NetworkResource

Encoding Variation :

Comments : GSM 09.02, 14.7.8,

Type Definition

```
ENUMERATED {
    plmn                (0),
    hlr                 (1),
    vlr                 (2),
    pvlr                (3),
    controllingMSC      (4),
    vmsc                (5),
    eir                 (6),
    rss                 (7)  }
```

Detailed Comments : Elements of INTEGER are global for the ATS.

ASN.1 Type Definition

Type Name : NoReplyConditionTime

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
INTEGER (5 .. 30 )
```

Detailed Comments :

ASN.1 Type Definition

Type Name : Operation

Encoding Variation :

Comments : ITU−T Q.767

Type Definition

```
CHOICE {
  localValue  INTEGER,
  globalValue OBJECT IDENTIFIER}
```

Detailed Comments :

ASN.1 Type Definition

Type Name : OverrideCategory

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
ENUMERATED {
    overrideEnabled   (0),
    overrideDisabled  (1) }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : Password
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
NumericString (FROM ("0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9"))  (SIZE (4))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : PW_RegistrationFailureCause
Encoding Variation :
Comments : GSM 09.02, 14.7.7

Type Definition

```
ENUMERATED {
      undetermined              (0),
      invalidFormat             (1),
      newPasswordsMismatch   (2) }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : ReturnErrorProblem
Encoding Variation :
Comments : ITU−T Q.767

Type Definition

```
TCAP_Problems ( unrecognizedInvokeID
          | returnErrorUnexpected
          | unrecognizedError
          | unexpectedError
          | mistypedParameter )
```

Detailed Comments : Type restricted to these 5.

ASN.1 Type Definition

Type Name : ReturnResultProblem
Encoding Variation :
Comments : ITU−T Q.767

Type Definition

```
TCAP_Problems ( unrecognizedInvokeID
          | returnResultUnexpected
          | mistypedResult )
```

Detailed Comments : Type restricted to these three.

ASN.1 Type Definition

Type Name : SS_Code

Encoding Variation :

Comments : GSM 09.02, 14.7.5

Type Definition

```
OCTET STRING (SIZE (1))
```

Detailed Comments : group (bits 8765), and specific service (bits 4321)

ASN.1 Type Definition

Type Name : SS_Data

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    ss_Code                 SS_Code                 OPTIONAL,
    ss_Status               [4] IMPLICIT SS_Status          OPTIONAL,
    ss_SubscriptionOption   SS_SubscriptionOption   OPTIONAL,
    basicServiceGroupList   BasicServiceGroupList   OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : SS_ForBS_Code

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    ss_Code       SS_Code,
    basicService  BasicServiceCode      OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : SS_IncompatibilityCause

Encoding Variation :

Comments : GSM 09.02, 14.7.8

Type Definition

```
SEQUENCE {
    ss_Code       [1] IMPLICIT SS_Code OPTIONAL,
    basicService  BasicServiceCode     OPTIONAL,
    ss_Status     [4] SS_Status        OPTIONAL }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : SS_Info

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
CHOICE {
    forwardingInfo  [0] IMPLICIT ForwardingInfo,
    callBarringInfo  [1] IMPLICIT CallBarringInfo,
    cug_Info        [2] IMPLICIT CUG_Info,
    ss_Data         [3] IMPLICIT SS_Data        }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : SS_Status

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

OCTET STRING (SIZE (1))

Detailed Comments : bits 8765: 0000 unused,
bits 4: SS state information Q bit,
bits 3: SS state information P bit,
bits 2: SS state information R bit,
bits 1: SS state information A bit.

ASN.1 Type Definition

Type Name : SS_SubscriptionOption

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
CHOICE {
    overrideCategory       [1] IMPLICIT OverrideCategory,
    cliRestrictionOption    [2] IMPLICIT CliRestrictionOption }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : SS_UserData

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

IA5String (SIZE (1 .. maxSignalInfoLength))

Detailed Comments :

ASN.1 Type Definition

Type Name : TCAP_Problems

Encoding Variation :

Comments : ITU−T Q.767

Type Definition

```
INTEGER {
  unrecognizedComponent      (0),    −− GeneralProblem
  unrecognizedInvokeID       (0),    −− ReturnResultProblem, ReturnErrorProblem
  duplicateInvokeID          (0),    −− InvokeProblem
  mistypedComponent          (1),    −− GeneralProblem
  returnErrorUnexpected      (1),    −− ReturnErrorProblem
  returnResultUnexpected     (1),    −− ReturnResultProblem
  unrecognizedOperation      (1),    −− InvokeProblem
  badlyStructuredComponent   (2),    −− GeneralProblem
  unrecognizedError          (2),    −− ReturnErrorProblem
  mistypedArgument           (2),    −− InvokeProblem, originally called mistypedParameter in TCAP
  mistypedResult             (2),    −− ReturnResultProblem
  resourceLimitation         (3),    −− InvokeProblem
  unexpectedError            (3),    −− ReturnErrorProblem
  mistypedParameter          (4),    −− ReturnErrorProblem
  initiatingRelease          (4),    −− InvokeProblem
  unrecognizedLinkedID       (5),    −− InvokeProblem
  linkedResponseUnexpected   (6),    −− InvokeProblem
  unexpectedLinkedOperation  (7)     −− InvokeProblem
  }
```

Detailed Comments : Errors of the same integer value are distinguished by their different parent types (General, Invoke, ReturnResult, ReturnError).

ASN.1 Type Definition

Type Name : TeleserviceCode

Encoding Variation :

Comments : GSM 09.02, 14.7.9

Type Definition

```
OCTET STRING (SIZE (1))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : USSD_Arg

Encoding Variation :

Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
      ussd_DataCodingScheme    USSD_DataCodingScheme,
      ussd_String              USSD_String }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : USSD_Res
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
SEQUENCE {
    ussd_DataCodingScheme    USSD_DataCodingScheme,
    ussd_String              USSD_String }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : USSD_DataCodingScheme
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
OCTET STRING (SIZE (1))
```

Detailed Comments :

ASN.1 Type Definition

Type Name : USSD_String
Encoding Variation :
Comments : GSM 09.02, 14.7.4

Type Definition

```
OCTET STRING (SIZE (1..maxUSSD_StringLength))
```

Detailed Comments : The structure of the contents of the USSD−String is dependent on the USSD−DataCodingScheme as described in TS GSM03.38.

Test Suite Operation Definition

Operation Name : OC_Asn1intToOct(n : Asn1Integer; l: INTEGER)

Result Type : OCTETSTRING

Comments :

Description

OC_Asn1intToOct converts the ASN.1 INTEGER ‘n‘ into OCTETSTRING with length = ‘l‘.

for example:

OC_Asn1intToOct(14,1) = ’0E’O;
OC_Asn1intToOct(18,1) = ’12’O;
OC_Asn1intToOct(18,2) = ’0012’O;
OC_Asn1intToOct(−128,1) = ’80’O (MSB (position p out of 1...p) represents −2exp(p−1));
OC_Asn1intToOct(−32768,2) = ’8000’O (MSB (position p out of 1...p) represents −2exp(p−1)).

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_Bcap(setup : SETUP_MO_PDU; callproc : CALL_PROC_PDU; mem : INTEGER)

Result Type : BCAP

Comments :

Description

OC_Bcap operation returns a bearer capability IE according the following rule:

- for ‘mem‘ = 1:
  - if bearer capability IE were presented in ‘callproc‘ the returned BC is the bearer capability 1 of the ‘callproc‘;
  - otherwise, the returned bearer capability is the bearer capability 1 of the ‘setup‘.

- for ‘mem‘ = 2:
  - if bearer capability IE were presented in ‘callproc‘ the returned BC is the bearer capability 2 of the ‘callproc‘;
  - otherwise, the returned bearer capability is the bearer capability 2 of the ‘setup‘.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_BinAdd(bitstr1 : BITSTRING; bitstr2 : BITSTRING)

Result Type : BITSTRING

Comments :

Description

OC_BinAdd operation performs binary addition of two input parameters ‘bitstr1‘ and ‘bitstr2‘, then returns the result of the addition. These two input parameters shall have the same length, the result of the operation has the same length as the input parameters.

for example:

OC_BinAdd(’01000’B, ’00110’B) = ’01110’B;
OC_BinAdd(’01000’B, ’00011’B) = ’01011’B;
OC_BinAdd(’00100’B, ’00010’B) = ’00110’B.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_Bit7(bstring: B_8)

Result Type : BITSTRING

Comments : The input parameter bstring is of type BITSTRING[8].

Description

OC_Bit7(bstring) returns the value of bit 7 in the ‘bstring‘.

for example:

OC_Bit7('01010101'B) = '1'B,
OC_Bit7('10101010'B) = '0'B

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_BCDtoInt(bcdstring:HEXSTRING; relevant_digits: INTEGER)

Result Type : INTEGER

Comments :

Description

The operation OC_BCDtoInt converts last 'relevant_digits' of an HEXSTRING containing BCD coded digits to an integer representation of these relevant digits.

Example: OC_BCDtoInt('12345'H, 3) := 345

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_CalledNumCHK(callednum : OCTETSTRING; dialnum : IA5String)

Result Type : BOOLEAN

Comments :

Description

the operation OC_CalledNumCHK to check whether the called party number ‘callednum‘, which is represented by OCTETSTRING, is the same as the dialed number 'dialnum‘, which is represented by IA5String. It returns TRUE if they are the same, otherwise FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_ChkSRES(sres : OCTETSTRING; ki: BITSTRING; rand : BITSTRING)

Result Type : BOOLEAN

Comments : sres is 32 bits value.

Description

OC_ChkSRES checks the input parameter ‘sres‘ according to the authentication algorithm defined in the following procedure. It returns TRUE if the ‘sres‘ is correct, otherwise it returns FALSE.

- firstly the ‘ki‘ XOR to the ‘rand‘ results in RES1;
- then compare the most significant 32 bits of the RES1 with the ‘sres‘;
- if they are equal, the ‘sres‘ is correct and the operation returns TRUE;
- if they are not equal, the ‘sres‘ is wrong and the operation returns FALSE.

NOTE: this procedure is the test algorithm for authentication defined by GSM.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_CnvtMax(max : INTEGER)

Result Type : B_2

Comments :

Description

The allowed integer values for max are 1, 2, 4, 7 (maximum number of retransmissions). The result BITSTRING is 2 bits long. OC_CnvtMax converts the input integer ‘max‘ into a BITSTRING according to the following rule:

1. the result is ’00’B if the ‘max‘ = 1,
2. the result is ’01’B if the ‘max‘ = 2,
3. the result is ’10’B if the ‘max‘ = 4,
4. the result is ’11’B if the ‘max‘ = 7.

for example:

OC_CnvtMax(1) = ’00’B,
OC_CvntMax(7) = ’11’B.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_CnvtTx(tx : INTEGER)

Result Type : B_4

Comments : The result BITSTRING is 4 bits long.

Description

OC_CnvtTx converts the input integer ‘tx‘ into a BITSTRING according the following rule:

1. the result is ’0000’B if the ‘tx‘ = 3,
2. the result is ’0001’B if the ‘tx‘ = 4,
3. the result is ’0010’B if the ‘tx‘ = 5,
4. the result is ’0011’B if the ‘tx‘ = 6,
5. the result is ’0100’B if the ‘tx‘ = 7,
6. the result is ’0101’B if the ‘tx‘ = 8,
7. the result is ’0110’B if the ‘tx‘ = 9,
8. the result is ’0111’B if the ‘tx‘ = 10,
9. the result is ’1000’B if the ‘tx‘ = 11,
10. the result is ’1001’B if the ‘tx‘ = 12,
11. the result is ’1010’B if the ‘tx‘ = 14,
12. the result is ’1011’B if the ‘tx‘ = 16,
13. the result is ’1100’B if the ‘tx‘ = 20,
14. the result is ’1101’B if the ‘tx‘ = 25,
15. the result is ’1110’B if the ‘tx‘ = 32,
16. the result is ’1111’B if the ‘tx‘ = 50.

for example:

OC_CnvtTx(3) = ’0000’B,
OC_CvntTx(5) = ’0010’B.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_CodeSMSCBMessage(firstoct: INTEGER; lastoct: INTEGER)

Result Type : OCTETSTRING

Comments :

Description

The operation codes a part of the contents for a cell broadcast short message. The cell broadcast short message, is 82 octets long, i.e. 93 characters, each represented by 7 bits. As many as possible different characters are sent, the characters are those corresponding to the 7−bit representation of the integers 0, 1, 2, ..., 92. The bits are arranged acc. to GSM 03.38, clause 6.1.2.1.1. The result of this operation is the octetstring of the octets ’firstoct’ to ’lastoct’ (16 octets for the first message block, 22 octets for the 2nd, 3rd and 4th blocks), with the octets of the cell broadcast short message being numbered from 1 to 82.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_CodingOfUssdString(text: IA5String)

Result Type : OCTETSTRING

Comments :

Description

This operation provides the coding for a USSD String ’text’ in the default alpabet, each character represented by 7 bits. The bits are arranged acc. to GSM 03.38, clause 6.1.2.2.1.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_ComputeSMContents(NumberOfCharacters: INTEGER)

Result Type : OCTETSTRING

Comments : max. 160 characters, i.e. 140 octets.

Description

This operation provides a short message's contents with a specified number of characters 'NumberOfCharacters', each represented by 7 bits. As possible different characters are sent, the characters are those corresponding to the 7−bit representation of 0, 1, 2, ... up to ('NumberOfCharacters' − 1). If more than 128 characters are sent, the rest of the characters is the corresponding to 0, 1, ... up to (NumberOfCharacters − 128 − 1), e.g. for 160 characters: 0, 1, ..., 127, 0, 1, ..., 31. The bits are arranged acc. to GSM 03.38, clause 6.1.2.1.1.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_ComputeSMContentsSpecText(NumbOfIA5chara: INTEGER; text: IA5String)

Result Type : OCTETSTRING

Comments : max. 160 characters, i.e. 140 octets.

Description

This operation provides a short message's contents with a specified number of characters 'NumbOfIA5chara', each represented by 7 bits. 'text' is used as contents of the short message. If "text' contains less than 'NumberOfCharacters' characters, 'text' is repeated until the short message reaches the 'NumberOfCharacters' characters long. The bits packing is arranged according. to GSM 03.38, clause 6.1.2.1.1.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_CphKeyGen(ki : BITSTRING; rand : BITSTRING)

Result Type : BITSTRING

Comments : both ki and rand are 128 bits values, the result of the operation is 64 bits value

Description

OC_CphKeyGen generates the ciphering key from the input parameters according to the following procedure:

− firstly the 'ki' XOR to the 'rand' results in RES1;
− then discard the most significant 32 bits of the RES;
− the next 64 bits of RES1 are the ciphering key, the operation returns this value.
− the 32 least singnificant bits are not used.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_FirstDigi(bcddigits : HEXSTRING)

Result Type : B_4

Comments :

Description

The input parameter bcddigits shall be a BCD string (subset of HEXSTRING), the resut is a BITSTRING[4] of a binary representation of one BCD digit.
The function of the OC_FirstDigi is to return the first (most significant) digit of the input parameter ‘bcddigits‘.

for example:

OC_FirstDigi(’12345’) = ’0001’B,
OC_FirstDigi(’012345678’) = ’0000’B.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_FnInc(fn : FN; inc : INTEGER)

Result Type : FN

Comments :

Description

The function of OC_FnInc operation to increase frame number with ‘inc‘. The frame number to be incremented is the input parameter ‘fn‘ in FN type and the increment ‘inc‘ is in INTEGER type, the incremented frame number is returned in FN type.

```
OC_FnInc(fn, inc)
FN     fn;
INTEGER  inc;
{
int     fmrin;
FN    frmout;

    frmin = 51 * ((fn.t3 − fn.t2) MOD 26) + fn.t3 + 1326 * fn.t1_ ;

    frmout.t1_ = ((frmin + inc) DIV 1326) MOD 32;
    frmout.t2   = (frmin + inc) MOD 26;
    frmout.t3   = (frmin + inc) MOD 51;
    return (frmout);
}
```

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_GetSCTimeStamp(timezone : TZONES)

Result Type : TPSCTS

Comments : TPSCTS is HEXSTRING[14]

Description

This Operation provides the hexstring containing the service center time stamp (SCTS) according to GSM 03.40, clauses 9.2.2.1 and 9.2.3.11. The TSO reads the current time of the test systems clock and transforms the time in conbination with the input parameter 'timezone' into a service center time stamp.
Example:
1996 April 18, 15:32:46, timezone=4
OC_GetSCTimeStamp returns 69408151236440

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_IncTmsi(tmsi : TMSI_V; inc : OCTETSTRING)

Result Type : TMSI_V

Comments : the 'tmsi' is 4 OCTETs long

Description

OC_IncTmsi operation adds the two input parameters and returns the result. An overflow of addition is allowed.

For example :

OC_IncTmsi('33542140'O + '01'O) = '33542141'O;
OC_IncTmsi('21322140'O + '08'O) = '21322148'O.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_InRang(tx: INTEGER; maxret : INTEGER; m: INTEGER)

Result Type : BOOLEAN

Comments :

Description

The operation returns TRUE if (m DIV (maxret * ((230+maxret −1)/maxret))) is inside the following interval:
[0.8 − ((tx+1)/2) DIV tx; 1.2 − ((tx+1)/2) DIV tx]

where / is integer division, DIV is float division, m, tx and maxret are input parameters of the operation.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_IntToOct(n : INTEGER; l: INTEGER)

Result Type : OCTETSTRING

Comments :

Description

OC_IntToOct converts the INTEGER ‘n‘ into OCTETSTRING with length = ‘l‘.

for example:

OC_IntToOct(14,1) = ’0E’O;
OC_IntToOct(18,1) = ’12’O;
OC_IntToOct(18,2) = ’0012’O.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_LeastBits(bstring : BITSTRING; lg : INTEGER)

Result Type : BITSTRING

Comments :

Description

OC_LeastBits operation returns the ‘lg‘ least significant bits of the original ‘bstring‘.

for example:

OC_LeastBits(’110011000101010’B, 3) = ’010’B,
OC_LeastBits(’110011000101010’B, 6) = ’101010’B.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_LengthOf(identity_field :MI)

Result Type : INTEGER

Comments :

Description

OC_LengthOf operation returns the actual length of the ‘identity_field‘, the unit of length is in OCTET.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_LengthOfBCDN(bcdn :BCDN)

Result Type : INTEGER

Comments :

Description

OC_LengthOfBCDN operation returns the actual length of an IE of type BCDN ’bcdn’, the unit of length is in OCTET.

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_LengthOfComp(comp: Component_T)
Result Type	: LENGTH
Comments	:
Description	
OC_LengthOfComp operation returns the actual length of the Component_T ‘comp‘, the unit of length is in OCTET.	
Detailed Comments	:

Test Suite Operation Definition	
Operation Name	: OC_LengthOfString(strg: IA5String)
Result Type	: INTEGER
Comments	:
Description	
OC_LengthOfString operation returns the actual length (number of characters) of the string ‘strg‘.	
Detailed Comments	:

Test Suite Operation Definition

Operation Name : OC_LookupS(t: INTEGER; combined :BOOLEAN)

Result Type : INTEGER

Comments : the algorithm is derived from Table 3.1/GSM 04.08 for values of parameter S.

Description

This operation returns an INTEGER according the following algorithm:

```
OC_LookupS( t, comb)
INTEGER t;
BOOLEAN combined;
{
    switch (t)
        {  case 3:
           case 8:
           case 14:
           case 50:
              if (combined) then
                      return(41)
                 else   return(55);
              break;
            case 4:
            case 9:
            case 16:
               if (combined) then
                       return(52)
                  else  return(76);
               break;
            case 5:
            case 10:
            case 20:
               if (combined) then
                       return(58)
                  else  return(109);
               break;
            case 6:
            case 11:
            case 25:
               if (combined) then
                       return(86)
                  else  return(163);
               break;
            case 7:
            case 12:
            case 32:
               if (combined) then
                       return(115)
                  else  return(217);
               break;
     }
}
```

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_MostBits(bstring : BITSTRING; lg :INTEGER)

Result Type : BITSTRING

Comments :

Description

OC_MostBits operation returns the ‘lg‘ most significant bits of the original ‘bstring‘.

for example:

OC_LeastBits(’110011000101010’B, 3) = ’110’B,
OC_LeastBits(’110011000101010’B, 6) = ’110011’B.

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_MsrReptChk(msrres: MSRR; index : INTEGER)
Result Type	: BOOLEAN
Comments	: parameter index is used for selecting different match pattern

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_MsrReptChk operation checks whether the received measurement report contains correct values:

1. when the ‘index‘ = 1:
   if set { { msrres.bcchfrq_nc1, msrres.bsic_nc1}, { msrres.bcchfrq_nc2, msrres.bsic_nc2},
   { msrres.bcchfrq_nc3, msrres.bsic_nc3}, { msrres.bcchfrq_nc4, msrres.bsic_nc4},
   { msrres.bcchfrq_nc5, msrres.bsic_nc5}, { msrres.bcchfrq_nc6, msrres.bsic_nc6}
   }
   equals to
   set { {’00000’B, ’001011’B}, {’01111’B, ’001001’B},
   {’10101’B, ’001011’B}, {’10110’B, ’001101’B},
   {’11100’B, ’001111’B}, {’11111’B, ’001001’B}
   }
   then the operation returns TRUE otherwise FALSE.

2. when the ‘index‘ = 2:
   if set { { msrres.bcchfrq_nc1, msrres.bsic_nc1}, { msrres.bcchfrq_nc2, msrres.bsic_nc2},
   { msrres.bcchfrq_nc3, msrres.bsic_nc3}, { msrres.bcchfrq_nc4, msrres.bsic_nc4},
   { msrres.bcchfrq_nc5, msrres.bsic_nc5}, { msrres.bcchfrq_nc6, msrres.bsic_nc6}
   }
   equals to
   set { {’00000’B, ’001011’B}, {’00100’B, ’001001’B},
   {’00110’B, ’001011’B}, {’00101’B, ’001101’B},
   {’00010’B, ’001111’B}, {’00111’B, ’001001’B}
   }
   then the operation returns TRUE otherwise FALSE.

3. when the ‘index‘ = 3:
   if set { { msrres.bcchfrq_nc1, msrres.bsic_nc1}, { msrres.bcchfrq_nc2, msrres.bsic_nc2},
   { msrres.bcchfrq_nc3, msrres.bsic_nc3}, { msrres.bcchfrq_nc4, msrres.bsic_nc4},
   { msrres.bcchfrq_nc5, msrres.bsic_nc5}, { msrres.bcchfrq_nc6, msrres.bsic_nc6}
   }
   equals to
   set { {’00001’B, ’001011’B}, {’00101’B, ’001001’B},
   {’00111’B, ’001011’B}, {’00110’B, ’001101’B},
   {’00011’B, ’001111’B}, {’01000’B, ’001001’B}
   }
   then the operation returns TRUE otherwise FALSE.

4. when the ‘index‘ = 4 :
   if set { { msrres.bcchfrq_nc1, msrres.bsic_nc1}, { msrres.bcchfrq_nc2, msrres.bsic_nc2},
   { msrres.bcchfrq_nc3, msrres.bsic_nc3}, { msrres.bcchfrq_nc4, msrres.bsic_nc4},
   }
   equals to
   set { {’00000’B, ’001011’B}, {’00010’B, ’001111’B},
   {’00001’B, ’001111’B}, {’00011’B, ’001001’B}
   }
   then the operation returns TRUE otherwise FALSE.

5. when the ‘index‘ = 5 :
   if set { { msrres.bcchfrq_nc1, msrres.bsic_nc1}, { msrres.bcchfrq_nc2, msrres.bsic_nc2},
   { msrres.bcchfrq_nc3, msrres.bsic_nc3}, { msrres.bcchfrq_nc4, msrres.bsic_nc4},
   }
   equals to
   set { {’00000’B, ’001011’B}, {’00010’B, ’001111’B},
   {’00001’B, ’001111’B}, {’00100’B, ’001001’B}
   }
   then the operation returns TRUE otherwise FALSE.

6. when the ‘index‘ = 6:
   if set { { msrres.bcchfrq_nc1, msrres.bsic_nc1}, { msrres.bcchfrq_nc2, msrres.bsic_nc2},
   { msrres.bcchfrq_nc3, msrres.bsic_nc3}, { msrres.bcchfrq_nc4, msrres.bsic_nc4},
   { msrres.bcchfrq_nc5, msrres.bsic_nc5}, { msrres.bcchfrq_nc6, msrres.bsic_nc6}
   }
   equals to
   set { {’00000’B, ’001011’B}, {’00011’B, ’001001’B},
   {’00100’B, ’001011’B}, {’00101’B, ’001101’B},
   {’00110’B, ’001111’B}, {’00111’B, ’001001’B}
   }

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_OeBit(bcddigits : HEXSTRING)
Result Type	: BITSTRING
Comments	: The input parameter ‘bcddigits‘ is really BCD string (subset of HEXSTRING), the result is BITSTRING[1].

Description

The function of the OC_OeBit is as the following:

1. it returns ’1’B, if the length of the ‘bcddigits‘ is odd,
2. it returns ’0’B, if the length of the ‘bcddigits‘ is even.

for example:

OC_OeBit(’12583’) = ’1’B,
OC_OeBit(’87259957’) =’0’B.

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_OctToInt(ostr: OCTETSTRING)
Result Type	: INTEGER
Comments	:

Description

OC_OctToInt converts the OCTETSTRING ‘ostr‘ into INTEGER.

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_OctToInvokeIDType(o : OCTETSTRING)
Result Type	: InvokeIDType
Comments	:

Description

OC_OctToInvokeIDType converts the OCTETSTRING ‘o‘ into InvokeIDType, with the MSB of ’o’ representing the negative value −2exp(n−1), for example ’80’O−>’10000000’B−> −128. The rest of the bits can code positive values up to +2exp(n−2)−1, for example ’4F’O−>’01111111’B−> +127.

for example:

OC_OctToInvokeIDType(’80’O) = −128;
OC_OctToInvokeIDType(’81’O) = −128+1 = −127;
OC_OctToInvokeIDType(’40’O) = +64.
OC_OctToInvokeIDType(’7F’O) = +127.
OC_OctToInvokeIDType(’C0’O) = −128+64 = −64.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_OtherDigi(bcddigits : HEXSTRING)

Result Type : OCTETSTRING

Comments : The input parameter‘ bcddigits‘ is really BCD string (subset of HEXSTRING), the result is an even BCD digits except that the next to last may either be ’F’H or a BCD digit.

Description

The function of the OC_OtherDigi is as the following:

1. If the number of the ‘bcddigits‘ is odd, the operation removes the most significant digit, and then reverses the order of each pair of digits;
2. If the number of the ‘bcddigits‘ is even, first the operation suffixes the ‘bcddigits‘ with ’F’H, then removes the most significant digit, and then reverses the order of each pair of digits.

for example:

OC_OtherDigi(’12345’) = ’3254’,
OC_OtherDigi(’12345678’) =’325476F8’.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_PosinSet(set: Components; comp: Component)

Result Type : INTEGER

Comments :

Description

The operation OC_PosinSet returns the position of component ‘comp‘ within the SET ‘set‘.

for example:

if the set = { registerSSComponents, eraseSSComponents, activateSSComponents, deactivateSSComponents }

OC_PosinSet(set, registerSSComponents) = 0,
OC_PosinSet(set, activateSSComponents) = 2.

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_RachSlots(fn1 : FN; fn2 : FN; cmbn: BOOLEAN; mode : INTEGER)
Result Type	: INTEGER
Comments	:

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_RachSlots calculates the number of RACH slots between frame number ‘fn1‘ and ‘fn2‘ excluding the slots in ‘fn1‘ and ‘fn2‘, then return the result of the calculation.
If mode=0, ’fn1’ denotes the (first) frame number for sending Paging Request, while ’fn2’ denotes the (first) frame number for the received Channel Request. If mode=1 ’fn1’ and ’fn2’ denote the (first) frame number for the received two subsequent Channel Requests. Any other values for mode is not allowed.
Since the number of RACH slots depends on the RACH being combined or not with dedicated channels, the parameter ’cmbn’ is needed: TRUE −−> combined, FALSE−−> not combined.
It is assumed that the distance between fn1 and fn2 is not more than one group of 42432 TDMA frames (modulo 42432 is used to calculate t1’ of fn1 and fn2). This is equivalent to approximately 196 seconds. The 51 TDMA frames in a multiframe are numbered 0 to 50.
fn1’ := fn1 mod 51
fn2’ := fn2 mod 51

CASE 1: cmbn is FALSE (’not combined’)
When ’not combined’, all slots are RACH slots.

1.1 mode=0 (sending paging message at fn1)
The paging uses the numbered frames 6 − 9, 12 −19, 22−29, 32−39 and 42−49.

1.1.1 If fn1 is not in the paging TDMA frame mapping, it leads testing to a test system error.
IF (fn1’ < 6) OR (9 < fn1’ < 12) OR (19 < fn1’ < 22) OR (29 < fn1’ < 32) OR (39 < fn1’ < 42) OR (fn1’ >49),
A test system error! The calling test case shall re−run again.
OTHERWISE

1.1.2 The number of RACH slots is equal to the number of frames between fn1 + 3 and fn2. The fn1 is added by 3 because a page message occupies 4 slots. The fn1 indicates the first slot of the paging message.
IF (fn1 + 3) < fn2, OC_RachSlots := fn2 − (fn1 + 3) −1 = fn2 − fn1 − 4,

1.1.3 If fn1+3 is equal to or greater than fn2, then this is due to fn2 being in the next group of 42432 frames. In this case 42432 frames have to be added.
IF fn2 <= (fn1 + 3), OC_RachSlots := fn2 −fn1 − 3 + 42432 −1 = fn2 −fn1 + 42428

1.2 mode=1 (receiving channel request at fn1)
The number of RACH slots is equal to the number of frames between fn1 and fn2

1.2.1 IF fn1 < fn2, OC_RachSlots := fn2 − fn1 − 1,
1.2.2 IF fn2 <= fn1, OC_RachSlots := fn2 −fn1 + 42432 −1 = fn2 −fn1 + 42431

CASE 2: cmbn is TRUE (’combined’)
When combined only the slots of the numbered frames 4, 5, 14 to 36, 45 and 46 in each multiframe are RACH slots, i.e.total 27 frames per multiframe.

2.1 mode=0 (sending paging message at fn1)
The paging uses the numbered frames 6 − 9 and 12 −19.

2.1.1 If fn1 is not in the paging TDMA frame mapping, it leads testing to a test system error. If fn2 is not in the RACH TDMA frame mapping, it leads testing to fail.

IF (fn2‘ < 4) OR (5 < fn2‘ < 14) OR (36 < fn2‘ < 45) OR (fn2’ > 46), OC_RachSlots := −9999
IF (fn1’ < 6) OR (9 < fn1’ < 12) OR (fn1’ >19), A test system error! The calling test case shall re−run again.
OTHERWISE

2.1.2 Calculation of the number ’c’ multiframes between fn1 + 3 and fn2. If fn1 + 3 is equal to or greater than fn2, then this is due to fn2 being in the next group of 42432 frames. In this case 42432 frames have to be added. ’/’ shall be the integer division, i.e. the result is also integer. Fractions are discarded.

2.1.2.1 IF (fn1 + 3) < fn2, c := fn2 / 51 − (fn1 + 3) / 51
2.1.2.2 IF fn2 <= (fn1 + 3), c := fn2 / 51 − (fn1 + 3) / 51 + 42432 / 51 = fn2 / 51 − (fn1 + 3) / 51 + 832

2.1.3 Calculation of the number of frames ’a’ to be subtracted according to the position of fn1’ within the multiframe
IF (5 < fn1‘ < 9) , a := 2
IF (fn1‘ =9), a := 3
IF (11<fn1‘ <20), a := fn1’−8

2.2 mode=1 (receiving channel request at fn1)

2.2.1 If fn1 or fn2 are not in the RACH TDMA frame mapping, it leads testing to fail.
IF (fn2‘ < 4) OR (5 < fn2‘ < 14) OR (36 < fn2‘ < 45) OR (fn2’ > 46)
OR (fn1’ < 6) OR (9 < fn1’ < 12) OR (fn1’ >19), OC_RachSlots := −9999

2.2.2 Calculation of the number ’c’ multiframes between fn1 and fn2

Continued from previous page

Test Suite Operation Definition
Detailed Comments : TC_26_2_1_1 uses mode=0. TC_26_2_1_2 uses mode=1.

Test Suite Operation Definition	
Operation Name	: OC_Random(n1 : INTEGER; n2 : INTEGER)
Result Type	: INTEGER
Comments	:
Description	
This operation randomly returns one number from the following candidates: 'n1', 'n1'+1, ..., 'n2'	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: OC_RcsdPresent(msg : MODIFY_PDU)
Result Type	: BOOLEAN
Comments	: To check if RCSD IE is present or not in Modify PDU
Description	
IF RCSD IE is present in the PDU passed on input paremeter return TRUE ELSE return FALSE.	
Detailed Comments :	

Test Suite Operation Definition

Operation Name : OC_SaveAndProc1(val : INTEGER; mode :INTEGER; cnt : INTEGER; cmbnd : BOOLEAN)

Result Type : BOOLEAN

Comments : This operation is used for storing and analysing the CHANNEL REQUEST message distribution.

Description

The function of this operation is defined as an pseudo C code:

```
#define SAVE 0;
#define PROC 1;

OC_SaveAndProc(val, mode,cnt)
INTEGER      val, mode,cnt;
BOOLEAN     combined;
{
  static INTEGER   buf[200];
   int i, j, n;
   if (mode == SAVE)
       { cnt = cnt mod 200;
         if (cmbnd == C_NotCombined) AND (val<(151 + 8)) then
                 {  buf[cnt] = val;
                   return (TRUE);
                  }
         if (cmbnd == C_Combined) AND (val<(81 + 8)) then
                 {  buf[cnt] = val;
                   return (TRUE);
                  }
         else return (FALSE);
        }
   if (mode == PROC)
        {
           for (i=0,  i<200, i++)
                {  n=0;
                   for (j=0, j<200, j++)
                       if (buf[j] == buf[i])
                           n = n+1;
                   if (n > 41) return (FALSE);
                }
             return(TRUE);
         }
}
```

Detailed Comments : This test suite operation has two operation modes :

− when the mode is SAVE and the val is less than 89 for combined or 159 for non−combined, it saves the value val into internal buffer and returns TRUE, otherwise returns FALSE. 200 values will be stored in the internal buffer when the operation is invoked 200 times.

− When the mode is PROC, it analyses the values stored in the internal buffer, if no more than 41 of them are equal the operation returns TRUE, otherwise returns FALSE. (i.e. for all n, CARD{ k|f(k) = n } <= 41)

Test Suite Operation Definition

Operation Name : OC_SaveAndProc3(val : BITSTRING; mode :INTEGER ; cnt : INTEGER)

Result Type : BOOLEAN

Comments : This operation is used for storing and analysing the random reference.

Description

The function of this operation is defined as an pseudo C code:

```
#define SAVE 0;
#define PROC 1;

OC_SaveAndProc(val, mode,cnt)
BITSTRING[8] val;
INTEGER        mode,cnt;
{
  static BITSTRING[8]   buf[7];
   int i, j, n=7;
   if (mode == SAVE)
       { cnt = cnt mod 7;
         buf[cnt] = val;
         return(TRUE);
        }
   if (mode == PROC)
        {
          for (i=0,  i<6, i++)
              { for (j=i+1, j<7, j++)
                    if (buf[j] == buf[i])
                      { n = n−1;
                        break;
                      }
              }
           if (n>=4) then
                    return(TRUE)
           else    return (FALSE);
         }

}
```

Detailed Comments : The function of this test suite operation is :

− it saves the value val into internal buffer when the mode = SAVE. 7 values will be stored in the internal buffer when the operation is invoked 7 times.

− it compares the values stored in the internal buffer when the mode = PROC, if 4 or more than 4 of them are different the operation returns TRUE, otherwise returns FALSE.

Test Suite Operation Definition

Operation Name : OC_SaveAndRetrv(val : RQR; mode, idx, ex: INTEGER)

Result Type : RQR

Comments : This operation is used for storing and retrieving the random reference.

Description

The function of this test suite operation is (in prose):
− it saves the value val into internal buffer buf[idx] when the mode = SAVE and ex = 0. Maximum 9 values can be stored. The idx is ranged between 1−9.
− it returns the value stored in the internal buffer buf[idx] when the mode = RETRV and ex = 0.
− it returns a value which is different from any value stored in the buf and also different from any values which have been returned by the consecutive invocation of this operation with ex <> 0, if ex <> 0. The idx is ranged between 10−13.

The function of this operation is defined as an pseudo C code:

```
#define SAVE 0;
#define RETRV 1;

OC_SaveAndRetrv(val, mode, idx, ex)
RQR             val;
INTEGER         mode, idx, ex;
{
  int i, j;
  static RQR   buf[14];
   if ((mode == SAVE) AND (ex == 0))
       {
         buf[idx] = val;
         buf[10].ra = '00000000'B;
         buf[10].fn = '00'O;
         buf[11].ra = '00000000'B;
         buf[11].fn = '00'O;
         buf[12].ra = '00000000'B;
         buf[12].fn = '00'O;
         buf[13].ra = '00000000'B;
         buf[13].fn = '00'O;
         return (buf[10];
        }
   if ((mode == RETRV) AND (ex == 0))
         return (buf[idx]);
   if (ex != 0)
         {
           buf[0].ra = buf[1].ra + '00000001'B;
           buf[0].fn = buf[1].fn;
           for (j=1, j <13, j++)
              {
                for (i=1, i<14, i++)
                   if( buf[0] == buf[i])
                       { buf[0].ra = buf[i].ra + '00000001'B;
                         break;
                        }
                if (i == 14)
                 {
                   buf[idx].ra = buf[0].ra:
                   return (buf[0]);
                 }
              }
          buf[idx].ra = buf[i].ra+'00000001'B;
          return (buf[idx]);
         }
}
```

Detailed Comments : 0. The pseudo C code gives one of the possible implementations for the OC description in prose.
1. '00'O should be understood as equivalence of Fn_01, a TTCN structured type constraint.

Continued on next page

Continued from previous page

Test Suite Operation Definition

Detailed Comments : ...
2. The OC can save maximum 13 RQR values. In SAVE mode, idx is ranged between 1−9.
3. For ex<>0, there are maximum 13 scans. In each scan a new value is assigned to buf[0].ra. If there is a 'non−match' in one of the 13 scans the buf[0] is returned. In case of 'all−match' for the 13 scans, the pair of (buf[i].ra + '00000001' at the last scan, buf[idx].fn is returned (idx is ranged between 10 − 13).

Test Suite Operation Definition

Operation Name : OC_StartTime(frmn : FN; t, i : INTEGER)
Result Type : STRT
Comments :

Description

OC_StartTime operation generates the STARTING TIME IE according to the input parameters.

```
(* frmn is the current frame number,
   t is the "delay" in applying the new frequencies,
   i is the contents of the starting time IE. *)
OC_StartTime(frmn, t, i)
INTEGER  t, i;
FN          frmn;
{
int tmp;
STRT strt;

tmp = 51 * ((frmn.t3 − frmn.t2) MOD 26) + frmn.t3 + 1326 * frmn.t1_ ;
tmp = (tmp + t) MOD 42432;
strt.fn.t1_ = (tmp DIV 1326) MOD 32;
strt.fn.t2   = tmp MOD 26;
strt.fn.t3   = tmp MOD 51;
strt.iei = OMIT;
if (i == 1) strt.iei := '01111100'B  /* if i=1 the information element identifier shall be included */
return(strt);
}
```

Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_SubOctet(src : OCTETSTRING; len :INTEGER)
Result Type : OCTETSTRING
Comments :

Description

OC_SubOctet(src, len) is the octetstring of length 'len' starting from the leftmost position of the source octetstring 'src'.

For example : OC_SubOctet('123456789ABCDEF'O, 4) = '12345678'O

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_SubchOfFacchh(subch: B_1; cell : CellID; inst : INTEGER)
Result Type	: LOGICCH
Comments	: subch is of type BITSTRING[1]

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_SubchOfFacchh operation returns a logic channel identifier for FACCHH subchannel indicated by the input parameters ‘subch‘, ‘cell‘ and ‘inst‘, where ’subch’ is the subchannel number, ‘cell‘ is the cell identifier which the channel belongs to, ‘inst‘ is the instance of the channel.

for example:

OC_SubchOfFacchh(’0’B, C_CellA, 1) = C_FACCHH0_A_1;
OC_SubchOfFacchh(’1’B, C_CellA, 1) = C_FACCHH1_A_1;
OC_SubchOfFacchh(’0’B, C_CellB, 1) = C_FACCHH0_B_1;
OC_SubchOfFacchh(’1’B, C_CellB, 1) = C_FACCHH1_B_1.

pseudo C code definition for the operation as following :

```
OC_SubchOfFacchh(subch, cell, inst)
BITSTRING   subch;
CellID     cell;
INTEGER   inst;
{
  LOGICCH    logch;
  if (subch==’0’B && cell == C_CellA)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_A_1; break;
         case 2 : logch = C_FACCHH0_A_2; break;
         case 3 : logch = C_FACCHH0_A_3; break;
       }
 if (subch==’0’B && cell == C_CellB)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_B_1; break;
         case 2 : logch = C_FACCHH0_B_2; break;
         case 3 : logch = C_FACCHH0_B_3; break;
       }
 if (subch==’0’B && cell == C_CellC)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_C_1; break;
         case 2 : logch = C_FACCHH0_C_2; break;
         case 3 : logch = C_FACCHH0_C_3; break;
       }
 if (subch==’0’B && cell == C_CellD)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_D_1; break;
         case 2 : logch = C_FACCHH0_D_2; break;
         case 3 : logch = C_FACCHH0_D_3; break;
       }
 if (subch==’0’B && cell == C_CellE)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_E_1; break;
         case 2 : logch = C_FACCHH0_E_2; break;
         case 3 : logch = C_FACCHH0_E_3; break;
       }
 if (subch==’0’B && cell == C_CellF)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_F_1; break;
         case 2 : logch = C_FACCHH0_F_2; break;
         case 3 : logch = C_FACCHH0_F_3; break;
       }
 if (subch==’0’B && cell == C_CellG)
     switch (inst)
       {
         case 1 : logch = C_FACCHH0_G_1; break;
         case 2 : logch = C_FACCHH0_G_2; break;
         case 3 : logch = C_FACCHH0_G_3; break;
       }
 if (subch==’0’B && cell == C_CellH)
     switch (inst)
       {
```

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_SubchOfSacchh(subch: B_1; cell : CellID; inst : INTEGER)
Result Type	: LOGICCH
Comments	: subch is of type BITSTRING[1]

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_SubchOfSacchh operation returns a logic channel identifier for SACCHH subchannel indicated by the input parameters ‘subch‘, ‘cell‘ and ‘inst‘, where ’subch’ is the subchannel number, ‘cell‘ is the cell identifier which the channel belongs to, ‘inst‘ is the instance of the channel.

for example:

OC_SubchOfSacchh(’0’B, C_CellA, 1) = C_SACCHH0_A_1;
OC_SubchOfSacchh(’1’B, C_CellA, 1) = C_SACCHH1_A_1;
OC_SubchOfSacchh(’0’B, C_CellB, 1) = C_SACCHH0_B_1;
OC_SubchOfSacchh(’1’B, C_CellB, 1) = C_SACCHH1_B_1.

pseudo C code definition for the operation as following :

```
OC_SubchOfSacchh(subch, cell, inst)
BITSTRING   subch;
CellID     cell;
INTEGER   inst;
{
  LOGICCH    logch;
  if (subch==’0’B && cell == C_CellA)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_A_1; break;
          case 2 : logch = C_SACCHH0_A_2; break;
          case 3 : logch = C_SACCHH0_A_3; break;
        }
  if (subch==’0’B && cell == C_CellB)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_B_1; break;
          case 2 : logch = C_SACCHH0_B_2; break;
          case 3 : logch = C_SACCHH0_B_3; break;
        }
  if (subch==’0’B && cell == C_CellC)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_C_1; break;
          case 2 : logch = C_SACCHH0_C_2; break;
          case 3 : logch = C_SACCHH0_C_3; break;
        }
  if (subch==’0’B && cell == C_CellD)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_D_1; break;
          case 2 : logch = C_SACCHH0_D_2; break;
          case 3 : logch = C_SACCHH0_D_3; break;
        }
  if (subch==’0’B && cell == C_CellE)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_E_1; break;
          case 2 : logch = C_SACCHH0_E_2; break;
          case 3 : logch = C_SACCHH0_E_3; break;
        }
  if (subch==’0’B && cell == C_CellF)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_F_1; break;
          case 2 : logch = C_SACCHH0_F_2; break;
          case 3 : logch = C_SACCHH0_F_3; break;
        }
  if (subch==’0’B && cell == C_CellG)
      switch (inst)
        {
          case 1 : logch = C_SACCHH0_G_1; break;
          case 2 : logch = C_SACCHH0_G_2; break;
          case 3 : logch = C_SACCHH0_G_3; break;
        }
  if (subch==’0’B && cell == C_CellH)
      switch (inst)
        {
```

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_SubchOfSacch4(subch: B_2; cell : CellID)

Result Type : LOGICCH

Comments :

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_SubchOfSacch4 operation returns a logic channel identifier for SACCHC4 subchannel indicated by the input parameters ‘subch ‘ and ‘cell‘, where ‘subch‘ is the TDMA offset, ‘cell‘ is the cell identifier which the channel belongs to.

for example:

OC_SubchOfSacch4(’00’B, C_CellA) = C_SACCHC40_A;
OC_SubchOfSacch4(’01’B, C_CellA) = C_SACCHC41_A;
OC_SubchOfSacch4(’10’B, C_CellA) = C_SACCHC42_A;
OC_SubchOfSacch4(’11’B, C_CellA) = C_SACCHC43_A;
OC_SubchOfSacch4(’00’B, C_CellB) = C_SACCHC40_B;
OC_SubchOfSacch4(’01’B, C_CellB) = C_SACCHC41_B.

pseudo C code definition for the operation as following :

```
OC_SubchOfSacch4(subch, cell)
BITSTRING   subch;
CellID     cell;
{
  LOGICCH    logch;
  if (cell == C_CellA)
     switch (subch)
       {
         case ’00’B : logch = C_SACCHC40_A; break;
         case ’01’B : logch = C_SACCHC41_A; break;
         case ’10’B : logch = C_SACCHC42_A; break;
         case ’11’B : logch = C_SACCHC43_A; break;
       }
  if (cell == C_CellB)
     switch (subch)
       {
         case ’00’B : logch = C_SACCHC40_B; break;
         case ’01’B : logch = C_SACCHC41_B; break;
         case ’10’B : logch = C_SACCHC42_B; break;
         case ’11’B : logch = C_SACCHC43_B; break;
       }
  if (cell == C_CellC)
     switch (subch)
       {
         case ’00’B : logch = C_SACCHC40_C; break;
         case ’01’B : logch = C_SACCHC41_C; break;
         case ’10’B : logch = C_SACCHC42_C; break;
         case ’11’B : logch = C_SACCHC43_C; break;
       }
  if (cell == C_CellD)
     switch (subch)
       {
         case ’00’B : logch = C_SACCHC40_D; break;
         case ’01’B : logch = C_SACCHC41_D; break;
         case ’10’B : logch = C_SACCHC42_D; break;
         case ’11’B : logch = C_SACCHC43_D; break;
       }
  if (cell == C_CellE)
     switch (subch)
       {
         case ’00’B : logch = C_SACCHC40_E; break;
         case ’01’B : logch = C_SACCHC41_E; break;
         case ’10’B : logch = C_SACCHC42_E; break;
         case ’11’B : logch = C_SACCHC43_E; break;
       }
  if (cell == C_CellF)
     switch (subch)
       {
         case ’00’B : logch = C_SACCHC40_F; break;
         case ’01’B : logch = C_SACCHC41_F; break;
         case ’10’B : logch = C_SACCHC42_F; break;
         case ’11’B : logch = C_SACCHC43_F; break;
       }
  if (cell == C_CellG)
     switch (subch)
       {
```

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_SubchOfSacch8(subch: B_3; cell : CellID; inst : INTEGER)
Result Type	: LOGICCH
Comments	: offset is of type BITSTRING[3]

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_SubchOfSacch8 operation returns a logic channel identifier for SACCHC8 subchannel indicated by the input parameters ‘subch‘, ‘cell‘ and ‘inst‘, where ‘subch‘ is the TDMA offset, ‘cell‘ is the cell identifier which the channel belongs to, ‘inst‘ is the instance of the channel.

for example:

OC_SubchOfSacch8(’000’B, C_CellA, 1) = C_SACCHC80_A_1;
OC_SubchOfSacch8(’001’B, C_CellA, 1) = C_SACCHC81_A_1;
OC_SubchOfSacch8(’010’B, C_CellA, 1) = C_SACCHC82_A_1;
OC_SubchOfSacch8(’011’B, C_CellA, 1) = C_SACCHC83_A_1;
OC_SubchOfSacch8(’100’B, C_CellA, 1) = C_SACCHC84_A_1;
OC_SubchOfSacch8(’101’B, C_CellA, 1) = C_SACCHC85_A_1;
OC_SubchOfSacch8(’110’B, C_CellA, 1) = C_SACCHC86_A_1;
OC_SubchOfSacch8(’111’B, C_CellA, 1) = C_SACCHC87_A_1;
OC_SubchOfSacch8(’000’B, C_CellB, 1) = C_SACCHC80_B_1;
OC_SubchOfSacch8(’001’B, C_CellB, 1) = C_SACCHC81_B_1;
OC_SubchOfSacch8(’000’B, C_CellC, 2) = C_SACCHC80_C_2;
OC_SubchOfSacch8(’001’B, C_CellC, 2) = C_SACCHC81_C_2.

pseudo C code definition for the operation as following :

```
OC_SubchOfSacch8(subch, cell, inst)
BITSTRING   subch;
CellID     cell;
INTEGER   inst;
{
  LOGICCH    logch;
  if (inst==1 && cell == C_CellA)
     switch (subch)
       {
         case ’000’B : logch = C_SACCHC80_A_1; break;
         case ’001’B : logch = C_SACCHC81_A_1; break;
         case ’010’B : logch = C_SACCHC82_A_1; break;
         case ’011’B : logch = C_SACCHC83_A_1; break;
         case ’100’B : logch = C_SACCHC84_A_1; break;
         case ’101’B : logch = C_SACCHC85_A_1; break;
         case ’110’B : logch = C_SACCHC86_A_1; break;
         case ’111’B : logch = C_SACCHC87_A_1; break;
       }
  if (inst==2 && cell == C_CellA)
     switch (subch)
       {
         case ’000’B : logch = C_SACCHC80_A_2; break;
         case ’001’B : logch = C_SACCHC81_A_2; break;
         case ’010’B : logch = C_SACCHC82_A_2; break;
         case ’011’B : logch = C_SACCHC83_A_2; break;
         case ’100’B : logch = C_SACCHC84_A_2; break;
         case ’101’B : logch = C_SACCHC85_A_2; break;
         case ’110’B : logch = C_SACCHC86_A_2; break;
         case ’111’B : logch = C_SACCHC87_A_2; break;
       }
  if (inst==3 && cell == C_CellA)
     switch (subch)
       {
         case ’000’B : logch = C_SACCHC80_A_3; break;
         case ’001’B : logch = C_SACCHC81_A_3; break;
         case ’010’B : logch = C_SACCHC82_A_3; break;
         case ’011’B : logch = C_SACCHC83_A_3; break;
         case ’100’B : logch = C_SACCHC84_A_3; break;
         case ’101’B : logch = C_SACCHC85_A_3; break;
         case ’110’B : logch = C_SACCHC86_A_3; break;
         case ’111’B : logch = C_SACCHC87_A_3; break;
       }
  if (inst==1 && cell == C_CellB)
     switch (subch)
       {
         case ’000’B : logch = C_SACCHC80_B_1; break;
         case ’001’B : logch = C_SACCHC81_B_1; break;
         case ’010’B : logch = C_SACCHC82_B_1; break;
         case ’011’B : logch = C_SACCHC83_B_1; break;
         case ’100’B : logch = C_SACCHC84_B_1; break;
```

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition

Operation Name : OC_SubchOfSdcch4(subch: B_2; cell : CellID)

Result Type : LOGICCH

Comments :

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_SubchOfSdcch4 operation returns a logic channel identifier for SDCCH4 subchannel indicated by the input parameters ‘subch ‘ and ‘cell‘, where ‘subch‘ is the TDMA offset, ‘cell‘ is the cell identifier which the channel belongs to.

for example:

OC_SubchOfSdcch4(’00’B, C_CellA) = C_SDCCH40_A;
OC_SubchOfSdcch4(’01’B, C_CellA) = C_SDCCH41_A;
OC_SubchOfSdcch4(’10’B, C_CellA) = C_SDCCH42_A;
OC_SubchOfSdcch4(’11’B, C_CellA) = C_SDCCH43_A;
OC_SubchOfSdcch4(’00’B, C_CellB) = C_SDCCH40_B;
OC_SubchOfSdcch4(’01’B, C_CellB) = C_SDCCH41_B.

pseudo C code definition for the operation as following :

```
OC_SubchOfSdcch4(subch, cell)
BITSTRING   subch;
CellID     cell;
{
  LOGICCH    logch;
  if (cell == C_CellA)
     switch (subch)
       {
         case ’00’B : logch = C_SDCCH40_A; break;
         case ’01’B : logch = C_SDCCH41_A; break;
         case ’10’B : logch = C_SDCCH42_A; break;
         case ’11’B : logch = C_SDCCH43_A; break;
       }
  if (cell == C_CellB)
     switch (subch)
       {
         case ’00’B : logch = C_SDCCH40_B; break;
         case ’01’B : logch = C_SDCCH41_B; break;
         case ’10’B : logch = C_SDCCH42_B; break;
         case ’11’B : logch = C_SDCCH43_B; break;
       }
  if (cell == C_CellC)
     switch (subch)
       {
         case ’00’B : logch = C_SDCCH40_C; break;
         case ’01’B : logch = C_SDCCH41_C; break;
         case ’10’B : logch = C_SDCCH42_C; break;
         case ’11’B : logch = C_SDCCH43_C; break;
       }
  if (cell == C_CellD)
     switch (subch)
       {
         case ’00’B : logch = C_SDCCH40_D; break;
         case ’01’B : logch = C_SDCCH41_D; break;
         case ’10’B : logch = C_SDCCH42_D; break;
         case ’11’B : logch = C_SDCCH43_D; break;
       }
  if (cell == C_CellE)
     switch (subch)
       {
         case ’00’B : logch = C_SDCCH40_E; break;
         case ’01’B : logch = C_SDCCH41_E; break;
         case ’10’B : logch = C_SDCCH42_E; break;
         case ’11’B : logch = C_SDCCH43_E; break;
       }
  if (cell == C_CellF)
     switch (subch)
       {
         case ’00’B : logch = C_SDCCH40_F; break;
         case ’01’B : logch = C_SDCCH41_F; break;
         case ’10’B : logch = C_SDCCH42_F; break;
         case ’11’B : logch = C_SDCCH43_F; break;
       }
  if (cell == C_CellG)
     switch (subch)
       {
```

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OC_SubchOfSdcch8(subch: B_3; cell : CellID; inst : INTEGER)
Result Type	: LOGICCH
Comments	: offset is of type BITSTRING[3]

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OC_SubchOfSdcch8 operation returns a logic channel identifier for SDCCH8 subchannel indicated by the input parameters ‘subch‘, ‘cell‘ and ‘inst‘, where ‘subch‘ is the TDMA offset, ‘cell‘ is the cell identifier which the channel belongs to, ‘inst‘ is the instance of the channel.

for example:

OC_SubchOfSdcch8(’000’B, C_CellA, 1) = C_SDCCH80_A_1;
OC_SubchOfSdcch8(’001’B, C_CellA, 1) = C_SDCCH81_A_1;
OC_SubchOfSdcch8(’010’B, C_CellA, 1) = C_SDCCH82_A_1;
OC_SubchOfSdcch8(’011’B, C_CellA, 1) = C_SDCCH83_A_1;
OC_SubchOfSdcch8(’100’B, C_CellA, 1) = C_SDCCH84_A_1;
OC_SubchOfSdcch8(’101’B, C_CellA, 1) = C_SDCCH85_A_1;
OC_SubchOfSdcch8(’110’B, C_CellA, 1) = C_SDCCH86_A_1;
OC_SubchOfSdcch8(’111’B, C_CellA, 1) = C_SDCCH87_A_1;
OC_SubchOfSdcch8(’000’B, C_CellB, 1) = C_SDCCH80_B_1;
OC_SubchOfSdcch8(’001’B, C_CellB, 1) = C_SDCCH81_B_1;
OC_SubchOfSdcch8(’000’B, C_CellC, 2) = C_SDCCH80_C_2;
OC_SubchOfSdcch8(’001’B, C_CellC, 2) = C_SDCCH81_C_2.

pseudo C code definition for the operation as following :

```
OC_SubchOfSdcch8(subch, cell, inst)
BITSTRING   subch;
CellID     cell;
INTEGER   inst;
{
  LOGICCH    logch;
  if (inst==1 && cell == C_CellA)
     switch (subch)
       {
         case ’000’B : logch = C_SDCCH80_A_1; break;
         case ’001’B : logch = C_SDCCH81_A_1; break;
         case ’010’B : logch = C_SDCCH82_A_1; break;
         case ’011’B : logch = C_SDCCH83_A_1; break;
         case ’100’B : logch = C_SDCCH84_A_1; break;
         case ’101’B : logch = C_SDCCH85_A_1; break;
         case ’110’B : logch = C_SDCCH86_A_1; break;
         case ’111’B : logch = C_SDCCH87_A_1; break;
       }
  if (inst==2 && cell == C_CellA)
     switch (subch)
       {
         case ’000’B : logch = C_SDCCH80_A_2; break;
         case ’001’B : logch = C_SDCCH81_A_2; break;
         case ’010’B : logch = C_SDCCH82_A_2; break;
         case ’011’B : logch = C_SDCCH83_A_2; break;
         case ’100’B : logch = C_SDCCH84_A_2; break;
         case ’101’B : logch = C_SDCCH85_A_2; break;
         case ’110’B : logch = C_SDCCH86_A_2; break;
         case ’111’B : logch = C_SDCCH87_A_2; break;
       }
  if (inst==3 && cell == C_CellA)
     switch (subch)
       {
         case ’000’B : logch = C_SDCCH80_A_3; break;
         case ’001’B : logch = C_SDCCH81_A_3; break;
         case ’010’B : logch = C_SDCCH82_A_3; break;
         case ’011’B : logch = C_SDCCH83_A_3; break;
         case ’100’B : logch = C_SDCCH84_A_3; break;
         case ’101’B : logch = C_SDCCH85_A_3; break;
         case ’110’B : logch = C_SDCCH86_A_3; break;
         case ’111’B : logch = C_SDCCH87_A_3; break;
       }
  if (inst==1 && cell == C_CellB)
     switch (subch)
       {
         case ’000’B : logch = C_SDCCH80_B_1; break;
         case ’001’B : logch = C_SDCCH81_B_1; break;
         case ’010’B : logch = C_SDCCH82_B_1; break;
         case ’011’B : logch = C_SDCCH83_B_1; break;
         case ’100’B : logch = C_SDCCH84_B_1; break;
```

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition
Operation Name : OM_2Msgs(ch : LOGICCH; pgg : PGG; mode : SENDINGMODE)
Result Type : BOOLEAN
Comments :
Description
OM_2Msgs puts the layer 2 emulator into a special operation mode then returns. In this special mode the layer 2 emulator sends the next two consecutive messages in the following way: − send the first message on the paging subchannel indicated by the ‘pgg‘ and ’ch’; − if the ‘mode‘ = C_NxtButOne, send the second message in the next but one paging sub block; − if the ‘mode‘ = C_FmrAGB, send the second message in a former access grant block; − if the ‘mode‘ = C_BfReOcc, send the second message before the MS’s original paging subchannel re−occurs but later than the next paging block of that CCCH (paging block not belong to the MS); − if the ‘mode‘ = C_NxtButOneNxt, nothing is sent in the next but one paging sub block, then send the second message in the next paging subblock of the MS’s paging subchannel.
Detailed Comments :

Test Suite Operation Definition

Operation Name	:	OM_Assoc(lgch1 : LOGICCH; sublgch1 : LOGICCH; sublgch2 : LOGICCH; sublgch3 : LOGICCH; sublgch4 : LOGICCH; sublgch5 : LOGICCH; sublgch6 : LOGICCH; sublgch7: LOGICCH; sublgch8 : LOGICCH; lgch2 : LOGICCH; sublgch9 : LOGICCH; sublgch10 : LOGICCH; sublgch11 : LOGICCH; sublgch12 : LOGICCH; sublgch13 : LOGICCH; sublgch14 : LOGICCH; sublgch15: LOGICCH; sublgch16 : LOGICCH)
Result Type	:	BOOLEAN
Comments	:	

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

OM_Assoc operation associates the sub logic channel identifiers ‘sublgch1‘ to ‘sublgch8‘ and ‘sublgch9‘ to ‘sublgch16‘ with the generic "parent" channel identifier ‘ lgch1‘ and ‘lgch2‘ respectively, therefore the subchannel identifiers can refer to the corresponding channels configured by OM_ChConf operation.

The ‘lgch1‘ can be an identifier for SDCCH4, SDCCH8, FACCHH and the ‘lgch2‘ can be SACCHC4, SACCHC8, and SACCHH.

"dummy" in the actual parameter list means the corresponding parameter is not used.

Naming convention for logical channel constants:

C_<channel type>[<rate>][<subchannel>][_<cell>][_<instance>]
e.g.: C_FACCH H 0 _C _3 => C_FACCHH0_C_3

channel type: FCCH, SCH, BCCH, CBCH, PCH, RACH, FACCH, SACCH, SDCCH, DCCH

rate: F(ull), H(alf), 4 (fourths, C4 for SACCHC4), 8 (eighths, C8 for SACCHC8).

subchannel:
H: 0, 1
(C)4: 0, ..., 3
(C)8: 0, ..., 7

cell: A, ..., H

instance: 1, ..., 3 When several channels of the same type can exist in one cell simultaneously (in different time slots and / or on different frequencies) they are identified by their instance. The parameters of ChConf are used to specify the frequency and slot of this instance of the channel type.

Not all possible combinations of this naming syntax have been used. According to the elements that are used / omitted the name can have a special meaning.

Examples:	Comments:
C_FCCH_A	FCCH of cell A.
C_SCH_A	SCH of cell A.
C_BCCH_A_1	First instance of a BCCH in cell A.
C_CBCH_A	CBCH of cell A.
C_PCH_A_1	First instance of a PCH in cell A (there is one instance of the PCH per BCCH in the cell).
C_AGCH_A_1	First instance of a AGCH in cell A (there is one instance of the AGCH per BCCH in the cell).
C_RACH_A_1	First instance of a RACH in cell A.
C_FACCHF_A_1	First instance of a full rate FACCH in cell A.
C_FACCHH_A_1	First instance of a half rate FACCH in cell A (all subchannels).
C_FACCHH0_A_1	Subchannel 0 of the first instance of a half rate FACCH in cell A.
C_SACCHF_A_1	First instance of a full rate SACCH in cell A.
C_SACCHH_A_1	First instance of a half rate SACCH in cell A (all subchannels).
C_SACCHH1_A_1	Subchannel 1 of the first instance of a half rate SACCH in cell A.
C_SACCHC4_A	SACCH/4 of cell A (all subchannels).
C_SACCHC43_A	Subchannel 3 of the SACCH/4 of cell A.
C_SACCHC8_A_1	First instance of a SACCH/8 in cell A (all subchannels).
C_SACCHC87_A_1	Subchannel 7 of the first instance of a SACCH/8 in cell A.
C_SDCCH4_A	SDCCH/4 of cell A (all subchannels).
C_SDCCH42_A	Subchannel 2 of the SDCCH/4 of cell A.
C_SDCCH87_A_1	First instance of a SDCCH/8 in cell A (all subchannels).
C_SDCCH87_A_1	Subchannel 7 of the first instance of a SDCCH/8 in cell A.

This list covers all types of combinations used in the TTCN for the time being.

When the meaning refers to ’all’ subchannels of a channel (eg. C_FACCHH_A_1), it means that all active subchannels of this channel activated previously by OM_ChConf, are associated with a logical channel identifier in the OM_Assoc.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_BmInfo(ch : LOGICCH; mode : B_8)

Result Type : BOOLEAN

Comments : mode is of type BITSTRING[8], imode is a part of the channel mode IE

Description

OM_BmInfo operation checks whether the MS starts transmiting Bm channel information according to the new channel mode 'mode'. The operation returns TRUE if the result of the check is correct otherwise the TSO returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_ChangeRFOf2Cells(cellid1 : CellID; bspwr1 : INTEGER; cellid2 : CellID; bspwr2 : INTEGER)

Result Type : BOOLEAN

Comments :

Description

This TSO is used to change the RF level of the cell 'cellid1' to the RF level 'bspwr1' and the RF level of the cell 'cellid2' to the RF level 'bspwr2' [dBuVemf], then the TSO returns TRUE.

The main use of this TSO is to force the Mobile Station(MS) to select a new cell according to the GSM 05.08 and GSM 03.22.

For the MS to select the new cell then the path loss criterion parameter (C1) of the old will have to be less than zero (C1 < 0).
This criterion is defined as:

C1 = (A − max(B, 0))

when

A − Received Level Average − (RXLEV_ACCESS_MIN)
B − MS_TXPWR_MAX_CCH − P {non DCS 1800}
MS_TXPWR_MAX_CCH + POWEROFFSET − P {DCS 1800}

RXLEV_ACCESS_MIN − Minimum received level at the MS required for access to the system.
MS_TXPWR_MAX_CCH − Maximum TX power level an MS may use when accessing the system until otherwise commanded.
POWER OFFSET − The power offset to be used in conjunction with the MS TXPWR MAX CCH parameter by the class 3 DCS 1 800 MS.
P − Maximum RF output power of the MS.

All values are expressed in dBm.

In most test cases MS_TXPWR_MAX_CHH and the POWER_OFFSET will be set at their minimum values thus C1 = A (B always being negative). Suitable levels of cells will be either 63 dBuV(C_E_default) ,53 dBuV(C_E_neighbourdefault) or 45 dBuV(C_E_suitable). Assuming that the MS uses a RXLEV_ACCESS_MIN between −110 and −115 dbM then to ensure that C1 < 0 the test system's output needs to be set to ensure the level input detected by the MS is less than −123 dBm or lower (−115 plus an 8dB for discrimination between C1 or C2 values and 0, GSM 11.10−1 section 20). This value of −123 dBm corresponds to −10 dBuV (C_E_notsuitable)

Detailed Comments :

Test Suite Operation Definition	
Operation Name	: OM_ChConf(bspwr: INTEGER; mspwr : INTEGER; acttype : BITSTRING; chmod : CHMOD; ta : TA; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; td : INTEGER; fn : INTEGER; babr, cch_con, bpm : B_3; pgfil : PG1_RQ_PDU; dtxu : BITSTRING; dtxd : BITSTRING; cell : CellID; lgch1 : LOGICCH; lgch2 : LOGICCH; lgch3 : LOGICCH; lgch4 : LOGICCH; lgch5 : LOGICCH; lgch6 : LOGICCH; lgch7 : LOGICCH; lgch8 : LOGICCH; lgch9 : LOGICCH)
Result Type	: BOOLEAN
Comments	:

Continued on next page

Continued from previous page

Test Suite Operation Definition

Description

The operation sets the configuration of a basic physical channel according to the input parameters and map logic channel combination onto it:

− bspwr: base station power level in dBuv;
− mspwr: mobile station power level in "power control level";
− acttype: type of activation:
  − ’000’B activation for intra−cell channel change −−−− immediate assignment;
  − ’001’B activation for intra−cell channel change −−−− assignment;
  − ’010’B activation for inter−cell channel change −−−− asynchronous handover;
  − ’011’B activation for inter−cell channel change −−−− synchronous handover;
  − ’100’B activation for additional assignment;
    NOTE: the types defined above are all the same, no real difference
  − ’101’B activation for receiving only;
− chmod: channel mode;
− ta: timing advance −−−− receive timing ta bit periods in advance of the normal receive timing if the channel is a dedicated channel ;
− slot: time slot;
− tsc: Training sequence code for normal bursts;
− rf: ARFCN or hopping parameters;
− chcmbn: logic channel combination mapping to the physical channel;
− td: timing difference between the cell and conceptual timing reference −−−− to make 2 cells with different timebase;
− fn: initial frame number offset to the timing base counter −−−− to make 2 cells with different frame numbers;
− babr number of blocks reserved for access grant
− cch−con CCCH/SDCCHs configuration
− bpm multiframe period for transmission of PAGING REQUEST
− pgfil: paging filling contents;
− dtxu: mobile station discontinuous transmission:
  − ’0’B mobile station discontinuous transmission is not applied;
  − ’1’B mobile station discontinuous transmission is applied;
− dtxd: base station discontinuous transmission:
  − ’0’B base station discontinuous transmission is not applied;
  − ’1’B base station discontinuous transmission is applied;
− cell: cell identifier;
− lgch1: logic channel identifier for the channel FCCH;
− lgch2: logic channel identifier for the channel SCH;
− lgch3: logic channel identifier for the channel BCCH;
− lgch4: logic channel identifier for the channel PCH;
− lgch5: logic channel identifier for the channel AGCH;
− lgch6: logic channel identifier for the channel RACH;
− lgch7: logic channel identifier for the channel FACCH or SDCCH;
− lgch8: logic channel identifier for the channel SACCH;
− lgch9: logic channel identifier for the channel CBCH;

For parameters lgch1 to lgch9 "dummy" in the actual parameter list means that the corresponding parameters are not used.

In the test cases the logic channel identifiers lgch1 to lgch9 are used to refer the logic channels configured by the operation. There are only generic identifiers for SDCCH4 or SDCCH8 or FACCHH channel and SACCHC4 or SACCHC8 or SACCHH, identifiers for subchannels of SDCCH4 or SDCCH8 or FACCHH and SACCHC4 or SACCHC8 or SACCHH are linked to the generic identifiers by OM_Assoc operation and in turn refer to the sub logic channels configured by the operation.

Channel re−use whithiin OM_ChConf:
When an OM_ChConf calls is made then the following rules can be applied for deciding whether or not to re−use existing physical channel.
Decode the ’chcmbn’ (channel combination) parameter to determine the new channel type.
If new channel = "C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4" (combined) or
if new channel = "C_CBCH_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4" (combined + SMS) then re−use any existing control channel configuration with same CellID.
If new channel = "C_FCCH_SCH_BCCH_CCCH" (non−combined) then use any combined, combined + SMS or non−combined control channel configuration with the same CellID.
If new channel = "C_BCCH_CCCH_2", "C_BCCH_CCCH_3" or "C_BCCH_CCCH_4" (ccch−group) then re−use any existing ccch−group control channel configuration with the same CellID and the slot, or re−use any existing combined, combined + SMS with the same CellID.
If new channel = "C_TCHF_ACCHF_1", "C_TCHF_ACCHF_2", "C_TCHH_ACCHH_1", "C_TCHH_ACCHH_2" (traffic) or "C_SDCCH8_SACCHC8_1", "C_SDCCH8_SACCHC8_2", "C_SDCCH8_SACCHC8_3" (dedicated) then re−use any existing traffic or dedicated channel with the same channel description.

Continued on next page

Continued from previous page

Test Suite Operation Definition
Detailed Comments :

Test Suite Operation Definition
Operation Name : OM_ChMdModi(ch : LOGICCH; chmod: CHMOD)
Result Type : BOOLEAN
Comments :
Description
OM_ChMdModi operation passes channel mode IE for channel ‘ch‘ to lower layer emulator and requests the emulator to set the ‘ch‘ to the mode ‘chmod‘. After the mode changed, it returns.
Detailed Comments :

Test Suite Operation Definition
Operation Name : OM_CphMdChg(ch : LOGICCH; cphmod: CPHMS; key : BITSTRING)
Result Type : BOOLEAN
Comments :
Description
OM_CphMdChg operation passes ciphering parameters for channel ‘ch‘ to lower layer and sets the lower layer emulator into a special operation mode in which lower layer starts the 3−step ciphering mode setting sequence when the next L3 message on the channel ‘ch‘ arrives.
Detailed Comments :

Test Suite Operation Definition
Operation Name : OM_CphMd(ch : LOGICCH; cphmod: CPHMS; key : BITSTRING)
Result Type : BOOLEAN
Comments :
Description
OM_CphMd operation passes ciphering parameters for channel ‘ch‘ to lower layer and sets the channel ‘ch‘ in the ciphering mode ‘cphmod‘. This operation puts the channel ‘ch‘ into specified ciphering mode ‘cphmod‘ immediately without the 3−step ciphering mode setting sequence.
Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_ComingFn(ch : LOGICCH)

Result Type : FN

Comments :

Description

OM_ComingFn operation returns the frame number (FN modulo 42432) which is about 5 seconds later than current frame number and is able to carry L3 message on the channel ‘ch‘.

Detailed Comments : The delay of about 5 seconds ensures that there is still enough time left for higher layer controller to prepare next TTCN send event after the OM_ComingFn returns. the exact delay value is up to the implementor.

Test Suite Operation Definition

Operation Name : OM_Deactivate(ch1, ch2 : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

OM_Deactivate operation deactivates the logical channels ‘ch1‘ and ’ch2’, then returns TRUE.

The ch1 can be a DCCH or a TCH, the ch2 is the associated SACCH.

Deactivation of an active SDCCH channel can be done by stopping transmission; this means that for the bursts on the BCCH carriers, dummy bursts shall be transmitted, for other bursts nothing shall be transmitted. Alternatively the deactivation of a SDCCH can be done by transmitting bad blocks on the SACCH with the parity bits set incorrectly. This will lead to a radio link failure in the MS.

Deactivation of a TCH channel can be done by stopping transmission if it is non−BCCH.

Deactivation of a SACCH_T can be done by transmitting bad blocks on the SACCH_T with the parity bits set incorrectly. The wrong channel coding on the SACCH will lead to a radio link failure in the MS after several seconds depending on the system information messages..

Detailed Comments : The TSO is called in
TC_26_7_3_2, TC_26_7_4_3_2, TC_26_7_4_3_3, TC_26_7_4_3_4 on a SDCCH/4/SACCH/C4 channel in downlink,
TC_26_2_4_1, TC_26_2_4_2, TC_26_8_2_1, TC_26_8_2_2, TC_26_8_2_3 on TCH/SACCH_T channel in both directions.

Test Suite Operation Definition

Operation Name : OM_FHCHK(ch : LOGICCH; ca : CCHD; ma : MA; cd : CHD; t :INTEGER; fn : FN)

Result Type : BOOLEAN

Comments :

Description

After the SS has been configured to start / change frequency hopping at a particular start time, this TSO instructs the SS to check whether the MS correctly uses the hopping frequencies on and after the expected start time according to the frequency parameters.

If ‘t‘ = 0 or 32024 <= ‘t‘ <= 42431 the operation checks that the new frequencies starts "without delay".
If 0 < ‘t‘ <= 31000 the operation checks that the new frequencies starts "on" the frame number of (X + t) mod 42432, where x = 51 * ((fn.t3 − fn.t2) MOD 26) + fn.t3 + 1326 * fn.t1.
If 31000 < ‘t‘ < 32024, or ‘t‘ < 0 or ‘t‘ > 42431
then the test has been configured incorrectly and should be re−run with a permitted value for ‘t‘.

This check is performed at the uplink RF burst level by sampling for a period of 3 seconds from the start time and should therefore be called immediately after the SS has been configured to start / change hopping. The operation returns TRUE if the MS behaviour is correct, otherwise it returns FALSE.
Note: before this TSO is called, a layer 3 message must be sent to the MS to command the MS to start / change hopping using the same frequency hopping parameters.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_FreeResource(ch :LOGICCH)

Result Type : BOOLEAN

Comments :

Description

OM_FreeResource operation instructs the lower layer emulator to release the resources which have been allocated to the logical channel "ch". After the resources are released the operation returns TRUE.

Detailed Comments : In some test cases (most of them are hanhover test cases) diffirent channels are needed while the test goes on. the purpose of the operation is to reuse lower layer resources (e.g. tranceivers), which have been allocated to the previous channel, for configing another channel required in the next testing step.

Test Suite Operation Definition

Operation Name : OM_FreqDef(st : STRT; ma : MA; ch :LOGICCH; chd: CHD; ca : CCHD; frmn: FN)

Result Type : BOOLEAN

Comments :

Description

OM_FreqDef operation passes the frequency hopping parameters for channel ‘ch‘ to lower layer emulator. The emulator starts using these frequency parameters at the frame number indicated by starting time ‘st‘. The parameter ’frmn’ indicates a reference frame number at which the first burst of the message, FREQUENCY REDEFINITION, ASSIGNMENT COMMAND or HANDOVER COMMAND, containing starting time is sent in the concerned cell.
If 0 <= (st − frmn) mod 42432 <= 31623, the indicated time is the next time when frame number mod 42432 is equal to st.
If 32024 <= (st − frmn) mod 42432 <= 42431, the indicated time has already elapsed. After completely sending out the message containing the parameter starting time the tester starts frequency hopping.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_GetHoaccPara(ch : LOGICCH)

Result Type : HOACC_PARA

Comments :

Description

OM_GetHoaccPara instructs the lower layer emulator to return the total number of the handover access bursts received on channel ‘ch‘, the observed minimum and maximum power levels in dBm modulo 256 and the observed minimum and maximum timing advance in bits modulo 256.The lower layer emulator starts the observation of handover access bursts as soon as sending out a HANDOVER COMMAND message. The TSO returns an accumulative result of the observation when it is called.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_GetL1Hd(ch : LOGICCH)

Result Type : L1HD

Comments :

Description

OM_GetL1Hd instructs the L 2 emulator to read the L 1 header of the next SACCH frame received on channel ‘ch‘ and return this value.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_LowerLayerFail(ch : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

This operation is to force the lower layer failure, then it returns.

The lower layer failure can be any one of layer 1 failures or layer 2 failures.

- layer 1 failure, see GSM 05.08;
- layer 2 failure, see GSM 04.06.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_NoL2Ack(i : INTEGER; ch: LOGICCH)

Result Type : BOOLEAN

Comments :

Description

When the parameter i>0 the layer 2 emulator is put into a special operation mode then the TSO returns. In this special mode the layer 2 emulator shall not aknowledge the 'i'th occurence of the L2 I frame which has the more data bit "M" = 0 on the channel 'ch'.
The layer 2 emulator resumes normal operation when
− either the TSO is called again with i = 0 on the same 'ch',
− or a data link SABM message is received on the 'ch'.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_NotAckSetup(ch : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

This operation puts the layer 2 emulator into a special operation mode then returns. In this special mode the layer 2 emulator will not acknowledge the last L 2 frame which carries the SETUP message, and the layer 2 emulator resumes normal operation after this L 2 frame.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_NoUAforDISC(ch : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

OM_NoUAforDISC and OM_ResumUAforDISC are a pair of TSOs. OM_NoUAforDISC forces the L2 emulator of the tester to enter a specific mode. OM_ResumUAforDISC resumes the L2 emulator to a normal L2 operational mode.
In the specific mode on receiving a DISC on the channel 'ch', the L2 emulator
− does not respond with a UA,
− remains in the multiple−frame established state,
− indicates that the DISC has been received (by means of subsequent DL_RELEASE_INDICATION ASPs).
Then the TSO returns TRUE.

Detailed Comments : The TSO is called in TC_26_6_12_2 and TC_26_6_12_4.

Test Suite Operation Definition

Operation Name : OM_PgFill(cellid : CellID ; msg : PG1_RQ_PDU)

Result Type : BOOLEAN

Comments : The result returned by the operation is not used.

Description

OM_PgFill operation sets up the contents of paging filling frame to be sent on all paging subchannels continuously. If the contents of paging filling frame in cell ‘cellid‘ have not been set up, the operation sets the contents to ‘msg‘. If paging filling frame in cell ‘cellid‘ has been set up, the operation changes the paging filling message to ‘msg‘. After the required action has been taken it returns.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_Reactivate(ch1, ch2 : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

The TSO OM_Reactivate reactivates the logical channels ‘ch1‘ and 'ch2' which were previously deactivated by the OM_Deactivate. Then the TSO returns TRUE. The ch1 is an SDCCH, the ch2 is the associated SACCH.

Reactivation of deactivated SDCCH/SACCH channels means to restart normal transmissions both on the SDCCH and the associated SACCH as specified in the ETS 300 574 (GSM 05.02).

Detailed Comments : The TSO is called in TC_26_7_3_2, TC_26_7_4_3_2, TC_26_7_4_3_3, TC_26_7_4_3_4 on the SDCCH/4 / SACCH/C4 channels.

Test Suite Operation Definition

Operation Name : OM_ReturnFn(ch : LOGICCH)

Result Type : FN

Comments :

Description

OM_ReturnFn operation returns the frame number (FN modulo 42432) on which the last L3 message was sent on logic channel ‘ch‘.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_ResumUAforDISC(ch : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

OM_NoUAforDISC and OM_ResumUAforDISC are a pair of TSOs. OM_NoUAforDISC forces the L2 emulator to enter a specific mode. OM_ResumUAforDISC resumes the L2 emulator to a normal L2 operational mode.
OM_ResumeUAforDISC cancels the effect of OM_NoUAforDISC, returning L2 on the channel ’ch’ to normal operation in the multiple−frame established state.
Then the TSO returns TRUE.

Detailed Comments : The TSO is called in TC_26_6_12_2 and TC_26_6_12_4.

Test Suite Operation Definition

Operation Name : OM_SendSMSCBWhilePaging(ch : LOGICCH)

Result Type : BOOLEAN

Comments : The result returned by the operation is not used.

Description

The TSO makes the lower layers send the second SMSCB message at the same time as the MS is paged.
This shall be achieved by paging the MS immediately after the first block of the CB message has been sent. The SS shall ensure that the page is transmitted on the radio interface prior to the transmission of the 4th block of the CB message.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_SendNextOn(ch : LOGICCH; fn : FN)

Result Type : BOOLEAN

Comments :

Description

OM_SendNextOn operation sets the lower layer emulator into a special operation mode then returns. In the special mode the lower layer emulator sends the next L3 message on the indicated frame number ‘fn‘ on the ’ch’ channel. After the next L3 message is sent, the lower layer resumes normal operation automatically.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_SendNextOnHO(ch : LOGICCH; cnt : INTEGER)

Result Type : BOOLEAN

Comments : Used in non−synchronised handover tests

Description

OM_SendNextOnHO operation sets the lower layer emulator into a special operation mode then returns. In the special mode the lower layer emulator waits for the number of Handover bursts specified by the parameter ’cnt’ on channel ’ch’ before sending the next L3 message. After the message is sent, the lower layer resumes normal operation automatically.The TSO does not affect any TTCN receiving event.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_StartMsrReport(ch : LOGICCH)

Result Type : BOOLEAN

Comments : The result returned by the operation is not used.

Description

The operation starts the reporting of received MEASUREMENT REPORT message to layer 3 emulator, then it returns with value TRUE.
Measurement Reports shall be disable by default. By default means that the tester filters them out from the TTCN in−buffer. This operation enables the reports entering the TTCN in−buffer. The OM_StopMsrReport operation disables the reports entering the TTCN in−buffer. When starting each test case, the measurement reports are filtered out by default.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_StopAllBCCH(cell1, cell2, cell3, cell4, cell5, cell6, cell7, cell8 : CellID)

Result Type : BOOLEAN

Comments : The result returned by the operation is not used.

Description

The TSO stops the RF transmission on all BCCH channels including dummy bursts in the indicated cells, but keeps the uplink CCCH reception active. After all transmission have been stopped the TSO returns TRUE.
If a cell in the parameter list is not in use a "dummy" value needs to be assigned to it by the calling test case.
The TSO is called in TC_26_3_3.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_StopCell(cell : CellID)

Result Type : BOOLEAN

Comments : The result returned by the operation is not used.

Description

The TSO stops completely all RF transmission of the specific cell ‘cell‘, including the dummy bursts on other slots. After all RF transmission have been stopped the TSO returns TRUE.
The cell shall be deleted from the lower layer tester and therefore if the cell ’cell’ is required again by the calling test case it needs to be re−configured.
The TSO is called in TC_26_3_4, TC_26_6_6_1, TC_26_7_4_2_2_1, TC_26_7_4_2_2_2, TC_26_7_4_6, TC_31_6_2_3.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_StopMsrReport(ch : LOGICCH)

Result Type : BOOLEAN

Comments : The result returned by the operation is not used.

Description

The operation stops the reporting of received MEASUREMENT REPORT message to layer 3 emulator, after measurement report stops it returns with value TRUE.
Measurement Reports shall be disable by default. By default means that the tester filters them out from the TTCN in−buffer. This operation disables the reports entering the TTCN in−buffer. The OM_StartMsrReport operation enables the reports entering the TTCN in−buffer. When starting each test case, the measurement reports are filtered out by default.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_ACMIncCHK(para : IA5String)

Result Type : BOOLEAN

Comments :

Description

the OO_ACMIncCHK operation requests (e.g. from the control console of the test system) the test operator to read the value of ACM on SIM and check the increment of the value. Then the operator informs the test system whether the increment is as description ‘para‘, if it is as the description the operation returns TRUE, otherwise returns FALSE.

The value of ACM can be read either via MMI or by removing the SIM and using SIM reader.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_ACMReading

Result Type : BOOLEAN

Comments :

Description

the OO_ACMReading operation requests (e.g. from the control console of the test system) the test operator to read and note the value of ACM on SIM. After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

The value of ACM can be read either via MMI or by removing the SIM and using SIM reader.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_ACMSetting

Result Type : BOOLEAN

Comments :

Description

the OO_ACMSetting operation requests (e.g. from the control console of the test system) the test operator to reset the ACM to zero and set the ACMmax to 2 units. After the operator finishes the action and informs the test system it returns TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_AddPwrAmp

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to add power amplification at the Mobile Station under test.
After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_AltIndCHK
Result Type : BOOLEAN
Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the alerting indication is generated. If the operator informs (e.g. from the control console of the test system) the test system that the alerting indication is generated, the operation returns TRUE. If the operator informs the test system that the alerting indication is not generated, the operation returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CalledNumCHK
Result Type : BOOLEAN
Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the called party number is correctly displayed on the mobile station. If the operator informs (e.g. from the control console of the test system) the test system that the display is correct, the operation returns TRUE. If the operator informs the test system that the display is wrong, the operation returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CalledPtyNumCHK(num : CDPN)
Result Type : BOOLEAN
Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the called party number ‘num‘ is the one that was entered into the MS. If the operator informs (e.g. from the control console of the test system) the test system that the number is correct, the operation returns TRUE. If the operator informs the test system that the number is wrong, the operation returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CheckAllSMPresentBut4th
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check if the mobile station under test has stored all short messages during the test case but not the 4th one which should have been replaced by the 5th(cf. GSM 11.10, clause 34.2.7.4). The messages are displayed. Then the operation returns.
TRUE: All SM present but not the 4th
FALSE: 4th SM present or 1st, 2nd, 3rd or 5th missing.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CheckCBSMReceived(NumbOfIA5chara: INTEGER; msg:SMSCBpack)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check whether the MS under test has correctly received the cell broadcast short message represented by 'msg' . Then the TSO returns
TRUE: if the CBSM has been correctly received,
FALSE: if non or an incorrect CBSM has been received.
The TSO needs to convert the 'msg' into IA5 string according to the packing character rules specified in the section 6.1.2 of ETS 300 628 (GSM 03.38). The 'NumbOfIA5chara' indicates the number of IA5 characters in the 'msg'.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CheckMCEFOnSIM

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check if the Memory Capacity Exceeded Flag has been set on the SIM simulator and to inform the test system of the result of the checking. If the checking succeds the operation returns TRUE, FALSE otherwise.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CheckMCEFOnSIMUnset

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check if the Memory Capacity Exceeded Flag has been unset on the SIM simulator and to inform the test system of the result of the checking. If the checking succeds the operation returns TRUE, FALSE otherwise.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CheckMessageDisplayed(NumbOfIA5chara:INTEGER; msg:OCTETSTRING)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check if the mobile station under test indicates that an SM has arrived. If the MS provides the functionality to display MT messages, it is checked that the message contents represented by 'msg' is correctly displayed. Then the operation returns
TRUE: if the MS indicates SM reception (and displays correct message),
FALSE: if the MS does not indicate SM reception (or displays incorrect message).
The TSO needs to convert the 'msg' into IA5 string according to the packing character rules specified in the section 6.1.2 of ETS 300 628 (GSM 03.38). The 'NumbOfIA5chara' indicates the number of IA5 characters in the 'msg'.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_CheckUssdStringDisplayed(strg: IA5String)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check if the mobile station under test displays the Ussd String 'strg'. Then the operation returns.
TRUE: MS displays the correct Ussd String.
FALSE: MS does not display the correct Ussd String.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_ConnectSIMSimulator

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to connect the SIM simulator to the mobile station under test.
For the contents of the SIM simulator the operator refers to initial conditions of the test case in GSM 11.10−1.

After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_DepressEndKey

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to depress the 'END' key of the Mobile Stations keypad, then returns.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_DialCalledNum
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to dial the called party number at the Mobile Station under test but not press the "SEND" key (i.e. not initiate the call setup).
After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_DisplaySMAndSendReplySM(n, NumbOfIA5chara: INTEGER; msg:OCTETSTRING)
Result Type : OCTETSTRING
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to display the 'n'−th short message received in the arriving order and to check whether the short message contents represented by 'msg' is correctly displayed.
Then the TSO asks the operator to send a reply short message from the mobile.
After the operator finishes the action, the operation returns with the message contents sent in OCTETSTRING.
The TSO needs to convert the 'msg' into IA5 string and the IA5 text entered by the operator into OCTETSTRING according to the packing character rules specified in the section 6.1.2 of ETS 300 628 (GSM 03.38). The 'NumbOfIA5chara' indicates the number of IA5 characters in the 'msg'.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_DTMFIndCHK(character : IA5String)
Result Type : BOOLEAN
Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the 'character' is indicated by the DTMF indicator of the MS. If the operator informs (e.g. from the control console of the test system) the test system that the MS does not give the indication of 'character', the operation returns FALSE. If the operator informs the test system that the MS gives the indication, the operation returns TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_EmptyMessageStorage
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to empty the message storage of the mobile station under test.
After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_EnterPswd(pswd : IA5String)

Result Type : BOOLEAN

Comments :

Description

This TSO displays a message on the control console to inform the operator that he is to enter a given password (the actual password is given by 'pswd') on the mobile. The operation waits for the operator to acknowledge the request, which may be done before or after entering the password into the mobile, and then returns the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_HookOff

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to answer the mobile terminating call at the Mobile Station under test.
This operation waits for the operator to acknowledge the request which may be done before or after the call has been answered and then returns the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_IFsetup_Telephony

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or configuration of the MS in such a way that the MS is able to successfully receive the call for telephony service. If the MS supports both telephony without immediate connection mode and telephony with immediate connection mode, set the MS in telephony without immediate connection mode. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

This TSO is used in MO call test cases except TC_11_2, where this TSO is call for single−numbering scheme MT testing.

Detailed Comments : used for TS11 and TS12

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS2xorBS3x_MO(srv : SERVICES; itc, sacp : B_3; ce : B_2; rate : IA5String)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or the MS configuration in such a way that the MS, when trigged, is able to initiate the call for the BS2x or BS3x service 'srv' with the specified itc, sacp, ce values. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

This TSO is used in MO call test cases except TC_11_2, where this TSO is call for single−numbering scheme MT testing.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT paramters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

The possible values are:
1) srv −−−−− basic service
− C_Async300 Basic Service 21, Asynchronous 300 bps
− C_Async1200 Basic Service 22, Asynchronous 1.2 kbps
− C_Async120075 Basic Service 23, Asynchronous 1200/75 bps
− C_Async2400 Basic Service 24, Asynchronous 2.4 kbps
− C_Async4800 Basic Service 25, Asynchronous 4.8 kbps
− C_Async9600 Basic Service 26, Asynchronous 9.6 kbps
− C_Sync1200 Basic Service 31, Synchronous 1.2 kbps
− C_Sync2400 Basic Service 32, Synchronous 2.4 kbps
− C_Sync4800 Basic Service 33, Synchronous 4.8 kbps
− C_Sync9600 Basic Service 34, Synchronous 9.6 kbps

2) itc −−−−−− information transfer capability
− '001'B unrestricted digital information
− '010'B 3.1 kHz audio, ex PLMN

3) sacp −−−− signalling access protocol
− '001'B I.440/450
− '010'B X.21
− '101'B X.28 − non dedicated PAD
− '110'B X.32

4) ce −−−−−− connection element
− '00'B transparent
− '01'B non transparent

5) rate −−−−− channel rate
− C_Full full rate channel
− C_Half half rate channel

Detailed Comments : Used for BS21, BS22, BS23, BS24, BS25, BS26, BS31, BS32, BS33, BS34

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS4x_MO(srv : SERVICES; ce : B_2; rate : IA5String)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, orthe MS configuration in such a way that the MS, when trigged, is able to initiate the call for the BS4x service ‘srv‘ with specified ur and ce values. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

The possible values are:
1) srv −−−−− basic service
− C_PAD300 Basic Service 41, PAD access 300 bps
− C_PAD1200 Basic Service 42, PAD access 1.2 kbps
− C_PAD120075 Basic Service 43, PAD access 1200/75 bps
− C_PAD2400 Basic Service 44, PAD access 2.4 kbps
− C_PAD4800 Basic Service 45, PAD access 4.8 kbps
− C_PAD9600 Basic Service 46, PAD access 9.6 kbps

2) ce −−−−−− connection element
− ’00’B transparent
− ’01’B non transparent

3) rate −−−−− channel rate
− C_Full full rate channel
− C_Half half rate channel

Detailed Comments : Used for BS41, BS42, BS43, BS44, BS45, BS46

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS5x_MO(srv : SERVICES; ur : B_4; rate : IA5String)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, orthe MS configuration in such a way that the MS, when trigged, is able to initiate the call for the BS5x service ‘srv‘ with the specifyied ur value. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

The possible values are:
1) srv −−−−− basic service
− C_Packet2400 Basic Service 51, Packet access 2.4 kbps
− C_Packet4800 Basic Service 52, Packet access 4.8 kbps
− C_Packet9600 Basic Service 53, Packet access 9.6 kbps

2) ur −−−−−− user rate
− ’0011’B 2400 bit/s
− ’0100’B 4800 bit/s
− ’0101’B 9600 bit/s

3) rate −−−−− channel rate
− C_Full full rate channel
− C_Half half rate channel

Detailed Comments : Used for BS51, BS52, BS53

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS61orBS81_MO(srv : SERVICES; ur : B_4; sa : B_1; ce : B_2; rate : IA5String)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or the MS configuration in such a way that the MS, when trigged, is able to initiate the call for BS61 or BS81 with the specified ur, ce and sa values, and the first bearer capability is speech. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

This TSO is used in MO call test cases except TC_11_2, where this TSO is call for single−numbering scheme MT testing.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

The possible values are:
1) srv −−−−− basic service
− C_AltSpchData Basic Service 61, Alternatie Speech/data
− C_SpchData Basic Service 81, Speech followed by data

2) ur −−−−−− user rate
− ’0001’B 300 bit/s
− ’0010’B 1200 bit/s
− ’0111’B 1200/75 bit/s
− ’0011’B 2400 bit/s
− ’0100’B 4800 bit/s
− ’0101’B 9600 bit/s

3) sa −−−−−− synchronous/asynchronous
− ’0’B synchronous
− ’1’B asynchronous

4) ce −−−−−− connection element
− ’00’B transparent
− ’01’B non transparent

5) rate −−−−− channel rate
− C_Full full rate channel
− C_Half half rate channel

Detailed Comments : used for BS61 and BS81

Test Suite Operation Definition

Operation Name : OO_IFsetup_TS6x_MO(srv : SERVICES; ur : B_4; ce : B_2; rate : IA5String)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or the MS configuration in such a way that the MS, when trigged, is able to initiate the call for TS61 and TS62 with the specified ur and ce values, and the first bearer capability is speech. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

This TSO is used in MO call test cases except TC_11_2, where this TSO is call for single−numbering scheme MT testing.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

The possible values are:
1) srv −−−−− basic service
− C_AltSpchFax Teleservice 61, Alternatie Speech and Facsimile group 3
− C_AutoFax Teleservice 62, Automatic Facsimile group 3

2) ur −−−−−− user rate
− ’0011’B 2400 bit/s
− ’0100’B 4800 bit/s
− ’0101’B 9600 bit/s

3) ce −−−−−− connection element
− ’00’B transparent
− ’01’B non transparent

4) rate −−−−− channel rate
− C_Full full rate channel
− C_Half half rate channel

Detailed Comments : used for TS61 and TS62

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS2x_MT(srv : SERVICES; itc : B_3)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or the MS configuration in such a way that the MS is able to successfully receive the call for the BS2x service ‘srv‘ with the specified itc value. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT paramters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

Detailed Comments : Used for BS21, BS22, BS24, BS25, BS26

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS3x_MT(srv : SERVICES; itc, sacp : B_3)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, orthe MS configuration in such a way that the MS is able to successfully receive the call for the BS3x service ‘srv‘ with specified itc and sacp values. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

Detailed Comments : Used for BS31, BS32, BS33, BS34

Test Suite Operation Definition

Operation Name : OO_IFsetup_BS61orBS81_MT(srv : SERVICES; ur : B_4; sa : B_1)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or the MS configuration in such a way that the MS is able to successfully receive the call for BS61 or BS81 with the specified ur and sa values. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

Detailed Comments : used for BS61 and BS81

Test Suite Operation Definition

Operation Name : OO_IFsetup_TS6x_MT(srv : SERVICES; ur : B_4)

Result Type : BOOLEAN

Comments :

Description

This TSO requests (e.g. from the control console of the test system) the test operator to setup the external interface, or the MS configuration in such a way that the MS is able to successfully receive the call for TS61 and TS62 with the specified ur value. After the operator finishes the required action and informs the test system, the TSO returns with value TRUE.

Three negotiable parameters, number of stop bit, number of data bit, parity information, are asked as global IXIT parameters for all TE configurations. If necessary, the test operator refers to the manual provided by manufacturer to configure the MS.

Detailed Comments : used for TS61 and TS62

Test Suite Operation Definition

Operation Name : OO_InCallModi(srv : SERVICES)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to initiate a in−call modification at the Mobile Station under test, which is triggered by the calling tone identification (CNG) received by the MS, for the basic service 'srv', then returns TRUE. The calling test step can pass the 'srv' value either as TSPX or constant to the TSO.

The possible service values include (see also test suite constant declarations):
1) Teleservices
− C_Telephony
− C_EmgCall
− C_AltSpchFax
− C_AutoFax

2) Bearer services
− C_Async300, C_Async1200, C_Async120075, C_Async2400, C_Async4800, C_Async9600
− C_Sync1200, C_Sync2400, C_Sync4800, C_Sync9600
− C_PAD300, C_PAD1200, C_PAD120075, C_PAD2400, C_PAD4800, C_PAD9600
− C_Packet2400, C_Packet4800, C_Packet9600
− C_AltSpchData
− C_SpchData

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_InitCall(srv : SERVICES)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to initiate an MS originating call for the basic service 'srv', then returns TRUE. The calling test step can pass the 'srv' value either as TSPX or constant to the TSO. The necessary MS configuration has been done in the corresponding TSO OO_IFsetup_xx...xx_MO before this TSO is called.

The possible service values include (see also test suite constant declarations):

1) Teleservices

- C_Telephony ----- TS11
- C_EmgCall ----- TS12
- C_AltSpchFax ----- TS61
- C_AutoFax ----- TS62

2) Bearer services

- C_Async300 ----- BS21
- C_Async1200 ----- BS22
- C_Async120075 ----- BS23
- C_Async2400 ----- BS24
- C_Async4800 ----- BS25
- C_Async9600 ----- BS26
- C_Sync1200 ----- BS31
- C_Sync2400 ----- BS32
- C_Sync4800 ----- BS33
- C_Sync9600 ----- BS34
- C_PAD300 ----- BS41
- C_PAD1200 ----- BS42
- C_PAD120075 ----- BS43
- C_PAD2400 ----- BS44
- C_PAD4800 ----- BS45
- C_PAD9600 ----- BS46
- C_Packet2400 ----- BS51
- C_Packet4800 ----- BS52
- C_Packet9600 ----- BS53
- C_AltSpchData ----- BS61
- C_SpchData ----- BS81

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_InitNonCallSS

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to initiate a non call related supplementary service which is supported by the MS.
After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_InitSS(action: IA5String)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to initiate the required supplementary service by the MMI sequence ‘action‘.
After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_InServiceCHK

Result Type : BOOLEAN

Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the Mobile Station under test is in normal service state ("Idle, updated")−− listening to CCCH and BCCH and with U1 UPDATED status. If the operator informs (e.g. from the control console of the test system) the test system that the MS does not give any service indication, the operation returns FALSE. If the operator informs the test system that the MS gives service indication, the operation returns TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_Key

Result Type : BOOLEAN

Comments :

Description

This operation returns TRUE if a key of the SS’ keyboard has been pressed. Otherwise it returns FALSE.
After OO_PressKeyWhenInService was called, the TSO is then continuously called until the test operator hits a key or a timer expires.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_MptyCall

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to enter MultiParty MMI command at the Mobile Station under test, then it returns with the value TRUE after the operator finishes the action and informs the test system.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_MSSetupStoreClass1SMInMEMemory

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to setup the MS under test to store class 1 SM in the ME memory (by way of MMI, as described in PICS/PIXIT statement).
After the operator finishes the action, he informs the test system and the operation returns with the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_PLMNsCHK

Result Type : BOOLEAN

Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the Mobile Station presents a list of available PLMNs. The list shall include the MNC and MCC of cells 1 to 7, but not cell 8 (for GSM900) or cells 1 to 6, but not cell 7 (for DCS1800). If the operator informs (e.g. from the control console of the test system) the test system that the MS correctly presents the list, the operation returns TRUE. If the operator informs the test system that the list is incorrect, the operation returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_PLMNselModeAuto

Result Type : BOOLEAN

Comments :

Description

This operation switches the PLMN selection mode of the MS to automatic selection. After correctly switching of the mode the operation returns TRUE otherwise FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_PLMNselModeMan

Result Type : BOOLEAN

Comments :

Description

This operation switches the PLMN selection mode of the MS to manual selection. After correctly switching of the mode the operation returns TRUE otherwise FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_PowerUp
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to power up the Mobile Station under test and to inform after the test system, then it returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_PowerDown
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to power down the Mobile Station under test and to inform after the test system, then it returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_PressKeyWhenInService
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to press any key when the MS shows the service indicator, then returns. The return value is always TRUE.
Working with OO_Key together, the TSO displays a prompt to the test operator and then immediately returns.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_RecallAndDisplaySM(NumbOfIA5chara: INTEGER; msg: OCTETSTRING)
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to recall and display an SM stored in the ME of the MS under test e.g. by means of the MMI. Then the operator checks whether the short message contents represented by 'msg' is correctly displayed. The TSO returns
TRUE: if SM can be recalled and displayed correctly in comparison with the 'msg',
FALSE: if SM can not be recalled or displayed, or displayed incorrectly in comparison with the 'msg'.
The TSO needs to convert the 'msg' into IA5 string according to the packing character rules specified in the section 6.1.2 of ETS 300 628 (GSM 03.38). The 'NumbOfIA5chara' indicates the number of IA5 characters in the 'msg'.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_RemoveOneSM

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to remove one of the SM from the message store and then to inform the test system. It does not matter which short message has been removed. After that the operation returns with value TRUE. The returned value is not used in the ATS.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_RemvPwrAmp

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to remove the added power amplification at the Mobile Station under test and to inform after the test system, then the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_RFoutputCHK

Result Type : BOOLEAN

Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the Mobile Station produces any RF output. If the operator informs (e.g. from the control console of the test system) the test system that the MS does not produce any RF output, the operation returns FALSE. If the operator informs the test system that the MS produces RF output, the operation returns TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SelPLMN(par_plmn:OCTETSTRING)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to select the given PLMN in par_plmn manually, and to inform after the test system then the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SendMOShortMessage
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to send a short message from the mobile station under test.
The operation waits for the operator to acknowldge the request which may be done before or after sending the MO Short Message, and then returns the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SendSMSCOMMANDDel
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to send a SMS COMMAND message from the mobile station under test containing requiring to delete the previouly submited SM and then to inform the test system. Then, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SendSMSCOMMANDEnq
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to send a SMS COMMAND message from the mobile station under test containing an enquiry about the previouly submited SMand then to inform the test system. Then, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SetRefuseCall
Result Type : BOOLEAN
Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to enable call refusal on the MS. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_ShortKeyDepr(ch : IA5String)

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to cause a DTMF tone to be generated, e.g. by short depression of the key ‘ch‘ on the Mobile Station under test. The operation waits for the operator to acknowldge the request which may be done before or after generating the DTMF tone, and then returns the value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SIMIns

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to insert the SIM card into the Mobile Station under test. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SIM2Ins

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to insert the SIM card 2 into the Mobile Station under test. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, the operation returns with value TRUE.
The SIM Card 2 shall contain the following parameter values, which are different from dafault values(SIM Card 1):
IMSI= ’001011234’
HPLMN_search_period=6min.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SIM3Ins

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to insert the SIM card 3 into the Mobile Station under test and turn the MS power on, after the test operator finishes the action and informs the test system it returns with value TRUE.
The SIM Card 3 shall contain fixed dialling number allocated and activated.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SIMRmv
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to remove the SIM card from the Mobile Station under test. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SIMSimulAttIndOK
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check whether the SIM simulator indicates an attempt made by the ME to store the short message in the SIM. The SIM simulator returns the status response "OK" ('90 00"). Then returns.
Attempt indicated: TRUE
Attempt NOT indicated: FALSE

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SIMSimulAttIndMemProblem
Result Type : BOOLEAN
Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to check whether the SIM simulator indicates an attempt made by the ME to store the short message in the SIM. The SIM simulator returns the status response "Memory Problem" ('92 40"). Then returns.
Attempt indicated: TRUE
Attempt NOT indicated: FALSE

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SSresultCHK(svc : INTEGER)
Result Type : BOOLEAN
Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check the user indication of the result of the supplementary service ' svc'. If the operator informs (e.g. from the control console of the test system) the test system that the indication is correct, the operation returns TRUE. If the operator informs the test system that the indication is wrong, the operation returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SwitchOn

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to switch on (or if there is no switch then to restore the power to) the Mobile Station under test. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_SwitchOff

Result Type : BOOLEAN

Comments :

Description

This operation requests (e.g. from the control console of the test system) the test operator to switch off (or if there is no switch then to remove the power from) the Mobile Station under test. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, the operation returns with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_TCHThroConnCHK

Result Type : BOOLEAN

Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the TCH is through connected. The SS has to generate a noise pattern so that the operator can check this. If the operator informs (e.g. from the control console of the test system) the test system that the TCH is through connected, the operation returns TRUE. If the operator informs the test system that the TCH is not through connected, the operation returns FALSE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_TermCall

Result Type : BOOLEAN

Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to terminate the ongoing call. After the operator completes the action and informs (e.g. from the control console of the test system) the test system, it returns TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_TguardTimeOut

Result Type : BOOLEAN

Comments :

Description

This operation informs (e.g. from the control console of the test system) the test operator that the T_guard timer has expired during the test case leading to an inconclusive or fail verdict and returns immediately with value TRUE.

Detailed Comments :

Test Suite Operation Definition

Operation Name : OO_ToneStopCHK

Result Type : BOOLEAN

Comments :

Description

This operation first requests (e.g. from the control console of the test system) the test operator to check whether the MS stops the tone generation If the operator informs (e.g. from the control console of the test system) the test system that the MS does not stop tone generation, the operation returns FALSE. If the operator informs the test system that the MS stops tone generation, the operation returns TRUE.

Definition of "tone generation":

−During MO call: MS generates alerting tone/presents inband information sent by the network.
−During MT call: MS generates ringing tone.

Detailed Comments :

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPC_24DataF	BOOLEAN	PICS Table A.25 Item 12	2.4 k full rate data mode supported − must be TRUE if TSPC_24DataH = TRUE
TSPC_24DataH	BOOLEAN	PICS Table A.25 Item 13	2.4 k half rate data mode supported
TSPC_48DataF	BOOLEAN	PICS Table A.25 Item 14	4.8 k full rate data mode supported − must be TRUE if TSPC_48DataH = TRUE
TSPC_48DataH	BOOLEAN	PICS Table A.25 Item 15	4.8 k half rate data mode supported
TSPC_96Data	BOOLEAN	PICS Table A.25 Item 16	9.6 k full rate data mode supported
TSPC_Feat_A51	BOOLEAN	PICS Table A.2 Item 17	ciphering algorithm A5/1 supported
TSPC_Feat_A52	BOOLEAN	PICS Table A.2 Item 18	ciphering algorithm A5/2 supported
TSPC_AddCharSet	BOOLEAN	PICS Table A.25 Item 47	A, B, C, D chars supported
TSPC_AddInfo_PseudoSynch	BOOLEAN	PICS	Pseudo synchronised handover supported
TSPC_AlertInd	BOOLEAN	PICS Table A.25 Item 49	alerting indication to the user supported
TSPC_SvcOnTCH	BOOLEAN	PICS Table A.25 Item 22	at least one service on traffic channel supported
TSPC_SMS	BOOLEAN	PICS Table A.25 Item 31	at least one short message service supported
TSPC_SS	BOOLEAN	PICS Table A.25 Item 29	at least one supplementary service supported
TSPC_AutoAutoMode	BOOLEAN	PICS Table A.25 Item 48	automatically enter automatic selection of PLMN mode supported
TSPC_BasCharSet	BOOLEAN	PICS Table A.25 Item 46	Chars 0−9, *, # supported
TSPC_CalledNumDisp	BOOLEAN	PICS Table A.2 Item 1	called number display supported
TSPC_DCS	BOOLEAN	PICS Table A.1 Item 3	testing DCS1800
TSPC_DetachOnPwrDn	BOOLEAN	PICS Table A.25 Item 38	detach on power down supported
TSPC_DetachOnSIMRmv	BOOLEAN	PICS Table A.25 Item 39	detach on SIM remove supported
TSPC_DispRcvSMS	BOOLEAN	PICS Table A.25 Item 34	display of received SMS supported
TSPC_DualRate	BOOLEAN	PICS Table A.25 Item 23	dual rate channel types supported
TSPC_EGSM	BOOLEAN	PICS Table A.1 Item 2	both standard (PGSM) and extended GSM band supported
TSPC_EmgOnly	BOOLEAN	PICS Table A.25 Item 27	the only circuit switched basic service is emergency call
TSPC_followOnReq	BOOLEAN	PICS Table A.25 Item 53	follow−on request procedure supported
TSPC_HalfRateData	BOOLEAN	PICS Table A.25 Item 6	at least one half rate data service supported
TSPC_HalfRateSpeech	BOOLEAN	PICS Table A.25 Item 3	half rate speech mode supported

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPC_InCallMod	BOOLEAN	PICS Table A.25 Item 52	In−Call modification supported
TSPC_NoimmConn	BOOLEAN	PICS Table A.25 Item 51	at least one service not support immediate connection
TSPC_Feat_FND	BOOLEAN	PICS Table A.2 Item 21	FND feature supported
TSPC_NonCallSS	BOOLEAN	PICS Table A.25 Item 30	non call related supplementary service supported
TSPC_MTsvc	BOOLEAN	PICS Table A.25 Item 19	at least one MT circuit switched basic service supported
TSPC_MOsvc	BOOLEAN	PICS Table A.25 Item 20	at least one MO circuit switched basic service supported
TSPC_PGSM	BOOLEAN	PICS Table A.1 Item 1	only standard GSM band supported (not EGSM)
TSPC_RefusalCall	BOOLEAN	PICS Table A.25 Item 54	refusal of call supported
TSPC_ReplaceSMS	BOOLEAN	PICS Table A.25 Item 33	replace SMS supported
TSPC_ReplyProc	BOOLEAN	PICS Table A.25 Item 32	(SMS) reply procedures supported
TSPC_RFAmp	BOOLEAN	PICS Table A.25 Item 55	RF amplification supported
TSPC_SDCCHOnly	BOOLEAN	PICS Table A.25 Item 21	only SDCCH supported
TSPC_Serv_SS_AoCC	BOOLEAN	PICS Table A.5 Item 14	Advice of Charge (Charging) SS supported
TSPC_Serv_SS_BAIC	BOOLEAN	PICS Table A.5 Item 18	Barring of All Incoming Calls SS supported
TSPC_Serv_SS_BI	BOOLEAN	PICS	BI SS supported
TSPC_Serv_SS_BICRoam	BOOLEAN	PICS Table A.5 Item 19	Barring of Incoming Calls when Roaming Outside the Home PLMN Country SS supported
TSPC_Serv_SS_BAOC	BOOLEAN	PICS Table A.5 Item 15	Barring of All Outgoing Calls SS supported
TSPC_Serv_SS_BOIC	BOOLEAN	PICS Table A.5 Item 16	Barring of Outgoing International Calls SS supported
TSPC_Serv_SS_BOICexHC	BOOLEAN	PICS Table A.5 Item 17	Barring of Outgoing International Calls except those directed to the Home PLMN Country SS supported
TSPC_Serv_SS_CFB	BOOLEAN	PICS Table A.5 Item 6	Call Forwarding on Mobile Subscriber Busy SS supported
TSPC_Serv_SS_CFNRy	BOOLEAN	PICS Table A.5 Item 7	Call Forwarding on No Reply SS supported
TSPC_Serv_SS_CFNRc	BOOLEAN	PICS Table A.5 Item 8	Call Forwarding on Mobile Subscriber Not Reachable SS supported
TSPC_Serv_SS_CFU	BOOLEAN	PICS Table A.5 Item 5	Call Forwarding Unconditional SS supported
TSPC_Serv_SS_HOLD	BOOLEAN	PICS Table A.5 Item 10	Call Hold SS supported
TSPC_Serv_SS_MPTY	BOOLEAN	PICS Table A.5 Item 11	multiparty SS supported

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPC_Serv_SS_unstruct	BOOLEAN	PICS	USSD supported
TSPC_Serv_TS11	BOOLEAN	PICS Table A.3 Item 1	telephony supported
TSPC_Serv_TS12	BOOLEAN	PICS Table A.3 Item 2	emergency call supported
TSPC_Serv_TS21	BOOLEAN	PICS Table A.3 Item 3	SMS MT/PP supported
TSPC_Serv_TS22	BOOLEAN	PICS Table A.3 Item 4	SMS MO/PP supported
TSPC_Serv_TS23	BOOLEAN	PICS Table A.3 Item 5	SMS cell broadcast supported
TSPC_Serv_TS61	BOOLEAN	PICS	alternate speech and G3 fax (TS61) supported
TSPC_Serv_TS62	BOOLEAN	PICS	automatic G3 fax (TS62) supported
TSPC_Serv_BS21	BOOLEAN	PICS Table A.4 Item 1	data circuit duplex async 300 bit/s supported
TSPC_Serv_BS22	BOOLEAN	PICS Table A.4 Item 2	data circuit duplex async 1200 bit/s supported
TSPC_Serv_BS23	BOOLEAN	PICS Table A.4 Item 3	data circuit duplex async 1200/75 bit/s supported
TSPC_Serv_BS24	BOOLEAN	PICS Table A.4 Item 4	data circuit duplex async 2400 bit/s supported
TSPC_Serv_BS25	BOOLEAN	PICS Table A.4 Item 5	data circuit duplex async 4800 bit/s supported
TSPC_Serv_BS26	BOOLEAN	PICS Table A.4 Item 6	data circuit duplex async 9600 bit/s supported
TSPC_Serv_BS31	BOOLEAN	PICS Table A.4 Item 7	data circuit duplex sync 1200 bit/s supported
TSPC_Serv_BS32	BOOLEAN	PICS Table A.4 Item 8	data circuit duplex sync 2400 bit/s supported
TSPC_Serv_BS33	BOOLEAN	PICS Table A.4 Item 9	data circuit duplex sync 4800 bit/s supported
TSPC_Serv_BS34	BOOLEAN	PICS Table A.4 Item 10	data circuit duplex sync 9600 bit/s supported
TSPC_Serv_BS41	BOOLEAN	PICS Table A.4 Item 11	PAD access 300 bit/s supported
TSPC_Serv_BS42	BOOLEAN	PICS Table A.4 Item 12	PAD access 1200 bit/s supported
TSPC_Serv_BS43	BOOLEAN	PICS Table A.4 Item 13	PAD access 1200/75 bit/s supported
TSPC_Serv_BS44	BOOLEAN	PICS Table A.4 Item 14	PAD access 2400 bit/s supported
TSPC_Serv_BS45	BOOLEAN	PICS Table A.4 Item 15	PAD access 4800 bit/s supported
TSPC_Serv_BS46	BOOLEAN	PICS Table A.4 Item 16	PAD access 9600 bit/s supported
TSPC_Serv_BS51	BOOLEAN	PICS Table A.4 Item 17	packet access 2400 bit/s supported
TSPC_Serv_BS52	BOOLEAN	PICS Table A.4 Item 18	packet access 4800 bit/s supported
TSPC_Serv_BS53	BOOLEAN	PICS Table A.4 Item 19	packet access 9600 bit/s supported
TSPC_Serv_BS61	BOOLEAN	PICS	alternate speech/data (BS61) supported
TSPC_Serv_BS81	BOOLEAN	PICS	speech followed data (BS81) supported

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPC_SMSStatusRepCap	BOOLEAN	PICS Table A.25 Item 35	SMS status report capabilities supported
TSPC_StoreRcvSMSME	BOOLEAN	PICS Table A.25 Item 37	Storage of received SMS in ME supported
TSPC_StoreRcvSMSSIM	BOOLEAN	PICS Table A.25 Item 36	Storage of received SMS in SIM supported
TSPC_SIMRmv	BOOLEAN	PICS Table A.25 Item 40	SIM removable without power down supported
TSPC_SwitchOnOff	BOOLEAN	PICS Table A.2 Item 15	switch on/off supported
TSPC_TranspDataOnly	BOOLEAN	PICS Table A.25 Item 9	only transparent data service supported
TSPC_CC	BOOLEAN	PICS Table A.25 Item 26	CC protocol for at least one BC supported
TSPC_TeleSvc	BOOLEAN	PICS Table A.25 Item 25	at least one teleservice supported
TSPC_EFR	BOOLEAN	PICS	Set to TRUE for EFR MS which supports any EFR features, and FALSE otherwise.
TSPC_EFR_Speech_v2	BOOLEAN	PICS	Set to TRUE for EFR MS which supports Speech Version 2 and FALSE otherwise.
TSPC_EFR_Speech_v3	BOOLEAN	PICS	Set to TRUE for EFR MS which supports Speech Version 3 and FALSE otherwise.
TSPC_EFR_EmgCallBcap	BOOLEAN	PICS	Set to TRUE if received ESETUP msg contains bearer capability IE, otherwise FALSE.
TSPC_FullRateSpeech	BOOLEAN	PICS Table A.25 Item 2	full rate speech mode supported − must be TRUE if TSPC_HalfRateSpeech = TRUE
TSPC_FullRateOnly	BOOLEAN	PICS Table A.25 Item 24	only full rate channel type supported
TSPX_AltNb	BOOLEAN	PIXIT TC	use alternative neighbour cells description. default : FALSE
TSPX_TE_stopbit	B_1	PIXIT MS	Terminal Equipment configuration number stop bits. ’0’, 1 bit; ’1’, 2 bits. default value : ’0’B, 1 bit
TSPX_TE_databit	B_1	PIXIT MS	Terminal Equipment configuration number data bits. ’0’, 7 bits; ’1’, 8 bits. default value : ’1’B, 8 bits
TSPX_TE_FLCT	FLWCNTL	PIXIT MS	Terminal Equipment flow control. 0−outband, 1−inband, 2−no flow control. default : 0, outband flow control

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_TE_parity	B_3	PIXIT MS	Terminal Equipment configuration parity. 000=odd, 010=even, 011=none, 100=forced to 0, 101=forced to1. default : ’001’B, no parity
TSPX_BS_21_itc1	B_3	PIXIT MS	Information Transfer Capability value supported for BS21 (async data 300bits/s). default value : ’001’B, Unrestricted Digital Information
TSPX_BS_21_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS21 (async data 300bits/s). default value : FALSE, only one itc
TSPX_BS_21_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported by BS21 (async data 300bits/s). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_21_ce	B_2	PIXIT MS	BS21 (async data 300bits/s) connection element value used in MO call and if the answer to TSPX_BS_21_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’00’B, transparent mode
TSPX_BS_21_sacp	B_3	PIXIT MS	BS21 (async data 300bits/s)Signalling Access Protocol value used in MO call. default value : ’001’B, I440/450
TSPX_BS_21_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS21 (async data 300bits/s), it shall be different from the previous one if the answer to TSPX_BS_21_more_itc is TRUE. default value :’001’B, Unrestricted Digital Information
TSPX_BS_22_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS22 (async data 1200bits/s). default value : ’010’B, 3.1kHz audio, exPLMN

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_22_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS22 (async data 1200bits/s). default value : FALSE, only one itc
TSPX_BS_22_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported by BS22 (async data 1200bits/s). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_22_ce	B_2	PIXIT MS	BS22 (async data 1200bits/s). Connection element value used in MO call and if the answer to TSPX_BS_22_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’01’B, non transparent
TSPX_BS_22_sacp	B_3	PIXIT MS	BS22 (async data 1200bits/s). Signalling Access Protocol value used in MO call. default value : ’001’B, I440/450
TSPX_BS_22_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS22 (async data 1200bits/s), it shall be different from the previous one if the answer to TSPX_BS_22_more_itc is TRUE. default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_23_itc	B_3	PIXIT MS	Information Transfer Capability value supported by BS23 (async data 1200/75bits/s). default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_23_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for BS23 (async data 1200/75bits/s). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_23_ce	B_2	PIXIT MS	BS23 (async data 1200/75bits/s) Connection element value used in MO call. default value : ’00’B, transparent
TSPX_BS_23_sacp	B_3	PIXIT MS	BS23 (async data 1200/75bits/s) Signalling Access Protocol value used in MO call. default value : ’001’B, I440/450

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_24_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS24 (async data 2400bits/s). default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_24_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS24 (async data 2400bits/s). default value : FALSE, only one itc
TSPX_BS_24_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported by BS24 (async data 2400bits/s). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_24_ce	B_2	PIXIT MS	BS24 (async data 2400bits/s). Connection element value used in MO call and if the answer to TSPX_BS_24_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’00’B, transparent
TSPX_BS_24_sacp	B_3	PIXIT MS	BS24 (async data 2400bits/s) Signalling Access Protocol value used in MO call. default value : ’001’B, I440/450
TSPX_BS_24_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS24 (async data 2400bits/s), it shall be different from the previous one if the answer to TSPX_BS_24_more_itc is TRUE. default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_25_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS25 (async data 4800bits/s). default value : ’001’B, Unrestricted Digital Information
TSPX_BS_25_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS25 (async data 4800bits/s). default value : FALSE, only one itc

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_25_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported by BS25 (async data 4800bits/s). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_25_ce	B_2	PIXIT MS	BS25 (async data 4800bits/s). Connection element value used in MO call and if the answer to TSPX_BS_25_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’00’B, transparent
TSPX_BS_25_sacp	B_3	PIXIT MS	BS25 (async data 4800bits/s) Signalling Access Protocol value used in MO call. default value : ’001’B, I440/450
TSPX_BS_25_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS25 (async data 4800bits/s), it shall be different from the previous one if the answer to TSPX_BS_25_more_itc is TRUE. default value : ’001’B, Unrestricted Digital Information
TSPX_BS_26_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS26 (async data 9600bits/s). default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_26_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS26 (async data 9600bits/s). default value : FALSE, only one itc
TSPX_BS_26_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported by BS26 (async data 9600bits/s). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_26_ce	B_2	PIXIT MS	BS26 (async data 9600bits/s). Connection element value used in MO call and if the answer to TSPX_BS_26_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’00’B, transparent

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_26_sacp	B_3	PIXIT MS	BS26 (async data 9600bits/s) Signalling Access Protocol value used in MO call. default value : '001'B, I440/450
TSPX_BS_26_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS26 (async data 9600bits/s), it shall be different from the previous one if the answer to TSPX_BS_26_more_itc is TRUE. default value : '010'B, 3.1kHz audio, exPLMN
TSPX_BS_31_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS31 (sync data 1200bits/s). default value : FALSE, only one itc
TSPX_BS_31_more_sacp	BOOLEAN	PIXIT MS	more than one Signalling Access Protocol value supported by BS31(sync data 1200bits/s). default value : FALSE, only one sacp
TSPX_BS_31_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS31(sync data 1200bits/s). default value : '001'B, Unrestricted Digital Information
TSPX_BS_31_sacp1	B_3	PIXIT MS	Signalling Access Protocol value supported by BS31(sync data 1200bits/s). default value : '001'B, I440/450
TSPX_BS_31_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS31(sync data 1200bits/s), it shall be different from the previous one if the answer to TSPX_BS_31_more_itc is TRUE. default value : '001'B, Unrestricted Digital Information
TSPX_BS_31_sacp2	B_3	PIXIT MS	other Signalling Access Protocol value supported by BS31(sync data 1200bits/s), it shall be different from the previous one if the answer to TSPX_BS_31_more_sacp is TRUE. default value : '001'B, I440/450

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_32_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS32 (sync data 2400bits/s). default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_32_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS32 (sync data 2400bits/s). default value : FALSE, only one itc
TSPX_BS_32_more_sacp	BOOLEAN	PIXIT MS	more than one Signalling Access Protocol value supported by BS32 (sync data 2400bits/s). default value : FALSE, only one sacp
TSPX_BS_32_sacp1	B_3	PIXIT MS	Signalling Access Protocol value supported by BS32 (sync data 2400bits/s). default value : ’110’B, X.32
TSPX_BS_32_X32_T_NT	BOOLEAN	PIXIT MS	BS 32 (sync data 2400bits/s) both Transparent and Non−Transparent supported for X32 3.1kHz. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_32_X32_ce	B_2	PIXIT MS	BS32 (sync data 2400bits/s). If the answer to TSPX_BS_32_X32_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. Connection Element default value : ’01’B, non transparent
TSPX_BS_32_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS32 (sync data 2400bits/s), it shall be different from the previous one if the answer to TSPX_BS_32_more_itc is TRUE. default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_32_sacp2	B_3	PIXIT MS	other Signalling Access Protocol value supported by BS32 (sync data 2400bits/s), it shall be different from the previous one if the answer to TSPX_BS_32_more_sacp is TRUE. default value : ’110’B, X.32

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_33_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS33 (sync data 4800bits/s). default value : ’001’B, Unrestricted Digital Information
TSPX_BS_33_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS33 (sync data 4800bits/s). default value : FALSE, only one itc
TSPX_BS_33_more_sacp	BOOLEAN	PIXIT MS	more than one Signalling Access Protocol value supported by BS33 (sync data 4800bits/s). default value : FALSE, only one sacp
TSPX_BS_33_sacp1	B_3	PIXIT MS	Signalling Access Protocol value supported by BS33 (sync data 4800bits/s). default value : ’010’B, X.21
TSPX_BS_33_X32_T_NT	BOOLEAN	PIXIT MS	BS 33 (sync data 4800bits/s) both Transparent and Non−Transparent supported for X32 3.1kHz. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_33_X32_ce	B_2	PIXIT MS	BS33 (sync data 4800bits/s). If the answer to TSPX_BS_33_X32_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. Connection element default value : ’00’B, transparent
TSPX_BS_33_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS33 (sync data 4800bits/s), it shall be different from the previous one if the answer to TSPX_BS_33_more_itc is TRUE. default value : ’001’B, Unrestricted Digital Information
TSPX_BS_33_sacp2	B_3	PIXIT MS	other Signalling Access Protocol value supported by BS33 (sync data 4800bits/s), it shall be different from the previous one if the answer to TSPX_BS_33_more_sacp is TRUE. default value : ’010’B, X.21

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_34_itc1	B_3	PIXIT MS	Information Transfer Capability value supported by BS34 (sync data 9600bits/s). default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_34_more_itc	BOOLEAN	PIXIT MS	more than one Information Transfer Capability value supported by BS34 (sync data 9600bits/s). default value : FALSE, only one itc
TSPX_BS_34_more_sacp	BOOLEAN	PIXIT MS	more than one Signalling Access Protocol value supported by BS34 (sync data 9600bits/s). default value : FALSE, only one sacp
TSPX_BS_34_sacp1	B_3	PIXIT MS	Signalling Access Protocol value supported by BS34 (sync data 9600bits/s). default value : ’001’B, I440/450
TSPX_BS_34_X32_T_NT	BOOLEAN	PIXIT MS	BS 34 (sync data 9600bits/s) both Transparent and Non−Transparent supported for X32 3.1kHz. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_34_X32_ce	B_2	PIXIT MS	BS34 (sync data 9600bits/s). If the answer to TSPX_BS_34_X32_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’00’B, transparent
TSPX_BS_34_itc2	B_3	PIXIT MS	other Information Transfer Capability value supported by BS34 (sync data 9600bits/s), it shall be different from the previous one if the answer to TSPX_BS_34_more_itc is TRUE. default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_34_sacp2	B_3	PIXIT MS	other Signalling Access Protocol value supported by BS34 (sync data 9600bits/s), it shall be different from the previous one if the answer to TSPX_BS_34_more_sacp is TRUE. default value : ’001’B, I440/450

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_41_T_NT	BOOLEAN	PIXIT MS	BS41 (PAD access 300bit/s) both Transparent and Non−Tranparent supported . default value : FALSE, not support both
TSPX_BS_41_ce	B_2	PIXIT MS	Connection element value used in BS41 (PAD access 300bit/s) MO call. default value : ’00’B, transparent
TSPX_BS_42_T_NT	BOOLEAN	PIXIT MS	BS42 (PAD access 1200bit/s) both Transparent and Non−Transparent supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_42_ce	B_2	PIXIT MS	Connection element value used in BS42 (PAD access 1200bit/s) MO call. default value : ’010’B, non transparent
TSPX_BS_43_T_NT	BOOLEAN	PIXIT MS	BS43 (PAD access 1200/75bit/s) both Transparent and Non−Transparent supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_43_ce	B_2	PIXIT MS	Connection element value used in BS43 (PAD access 1200/75bit/s) MO call. default value : ’00’B, transparent
TSPX_BS_44_T_NT	BOOLEAN	PIXIT MS	BS44 (PAD access 2400bit/s) both Transparent and Non−Transparent supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_44_ce	B_2	PIXIT MS	Connection element value used in BS44 (PAD access 2400bit/s) MO call. default value : ’00’B, transparent
TSPX_BS_45_T_NT	BOOLEAN	PIXIT MS	BS45 (PAD access 4800bit/s) both Transparent and Non−Transparent supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_45_ce	B_2	PIXIT MS	Connection element value used in BS45 (PAD access 4800bit/s) MO call. default value : ’01’B, non transparent

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_46_T_NT	BOOLEAN	PIXIT MS	BS46 (PAD access 9600bit/s) both Transparent and Non−Transparent supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_46_ce	B_2	PIXIT MS	Connection element value used in BS46 (PAD access 300bit/s) MO call. default value : ’00’B, transparent
TSPX_BS_61_S	BOOLEAN	PIXIT MS	synchronous data supported by BS 61 (alternate speech/data). default value : TRUE, synchronous mode
TSPX_BS_61_S_more_ur	BOOLEAN	PIXIT MS	more than one user rate for synchronous data service supported by BS61 (alternate speech/data). default value : FALSE, only ine user rate
TSPX_BS_61_S_ur1	B_4	PIXIT MS	usrer rate value supported for synchronous data service of BS61 (alternate speech/data). default value : ’0100’B, 4.8 kbit/s
TSPX_BS_61_S_ur2	B_4	PIXIT MS	another usrer rate value supported for synchronous data service of BS61 (alternate speech/data). it shall be different from the previous one if the answer to TSPX_BS_61_S_more_ur is TRUE. default value : ’0100’B, 4.8 kbit/s
TSPX_BS_61_A	BOOLEAN	PIXIT MS	asynchronous data service supported by BS61 (alternate speech/data) . default value : TRUE, asynchronous
TSPX_BS_61_A_ur1	B_4	PIXIT MS	usrer rate value supported for asynchronous data service of BS61 (alternate speech/data). default value : ’0101’B, 9.6 kbit/s
TSPX_BS_61_A_ur1_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for the rate TSPX_BS_61_A_ur1 of BS61 (alternate speech/data). default value : FALSE, not support both transparent and non transparent modes

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_61_A_ur1_ce	B_2	PIXIT MS	BS61 (alternate speech/data). If the answer to TSPX_BS_61_A_ur1_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. Connection element default value : ’00’B, transparent
TSPX_BS_61_A_more_ur	BOOLEAN	PIXIT MS	more than one user rate for asynchronous data service supported by BS61 (alternate speech/data). default value : FALSE, only one user rate
TSPX_BS_61_A_ur2	B_4	PIXIT MS	another user rate value supported for asynchronous data service of BS61 (alternate speech/data). it shall be different from the previous one if the answer to TSPX_BS_61_A_more_ur is TRUE. default value : ’0101’B, 9.6 kbit/s
TSPX_BS_61_A_ur2_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for the rate TSPX_BS_61_A_ur2 of BS61 (alternate speech/data). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_81_S	BOOLEAN	PIXIT MS	synchronous data supported by BS 81 (speech followed by data). default value : TRUE, synchronous mode
TSPX_BS_81_S_more_ur	BOOLEAN	PIXIT MS	more than one user rate for synchronous data service supported by BS81 (speech followed by data). default value : FALSE, only one user rate
TSPX_BS_81_S_ur1	B_4	PIXIT MS	user rate value supported for synchronous data service of BS81 (speech followed by data). default value : ’0101’B, 9.6 kbit/s
TSPX_BS_81_S_ur2	B_4	PIXIT MS	another user rate value supported for synchronous data service of BS81 (speech followed by data). it shall be different from the previous one if the answer to TSPX_BS_81_S_more_ur is TRUE. default value : ’0101’B, 9.6 kbit/s

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_81_A	BOOLEAN	PIXIT MS	asynchronous data service supported by BS81 (speech followed by data). default value : TRUE, asynchronous mode
TSPX_BS_81_A_ur1	B_4	PIXIT MS	user rate value supported for asynchronous data service of BS81 (speech followed by data). default value : '0100'B, 4.8 kbit/s
TSPX_BS_81_A_ur1_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for the rate TSPX_BS_81_A_ur1 of BS81 (speech followed by data). default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_81_A_ur1_ce	B_2	PIXIT MS	BS81 (speech followed by data) If the answer to TSPX_BS_81_A_ur1_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : Connection element '01'B, non transparent
TSPX_BS_81_A_more_ur	BOOLEAN	PIXIT MS	more than one user rate for asynchronous data service supported by BS81 (speech followed by data). default value : FALSE, only one user rate
TSPX_BS_81_A_ur2	B_4	PIXIT MS	another user rate value supported for asynchronous data service of BS81 (speech followed by data). it shall be different from the previous one if the answer to TSPX_BS_81_A_more_ur is TRUE. default value : '0100'B, 4.8 kbit/s
TSPX_BS_81_A_ur2_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for the rate TSPX_BS_81_A_ur2 of BS81 (speech followed by data). default value : FALSE, not support both transparent and non transparent modes

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_TS_61_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for TS61(alternate speech and G3 fax). default value : FALSE, not support both transparent and non transparent modes
TSPX_TS_61_ce	B_2	PIXIT MS	TS61(alternate speech and G3 fax) if the answer to TSPX_TS_61_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. Connection element default value : ’01’B, non transparent
TSPX_TS_61_ur1	B_4	PIXIT MS	user rate for TS61(alternate speech and G3 fax). default value : ’0101’B, 9.6 kbit/s
TSPX_TS_61_more_ur	BOOLEAN	PIXIT MS	more than one user rate supported for TS61(alternate speech and G3 fax). default value : FALSE, only one user rate
TSPX_TS_61_ur2	B_4	PIXIT MS	another user rate for TS61 (alternate speech and G3 fax), it shall be different from the previous one if the answer to TSPX_TS_61_more_ur is TRUE. default value : ’0101’B, 9.6 kbit/s
TSPX_TS_62_T_NT	BOOLEAN	PIXIT MS	both Transparent and Non−Transparent supported for TS62 (automatic G3 fax). default value : FALSE, not support both transparent and non transparent modes
TSPX_TS_62_ce	B_2	PIXIT MS	TS62 (automatic G3 fax). If the answer to TSPX_TS_62_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. Connection element default value : ’00’B, transparent
TSPX_TS_62_ur1	B_4	PIXIT MS	user rate for TS62 (automatic G3 fax). default value : ’0100’B, 4.8 kbit/s
TSPX_TS_62_more_ur	BOOLEAN	PIXIT MS	more than one user rate supported by TS62 (automatic G3 fax). default value : FALSE, only one user rate

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_TS_62_ur2	B_4	PIXIT MS	another user rate for TS62 (automatic G3 fax), it shall be different from the previous one if the answer to TSPX_TS_62_more_ur is TRUE. default value : ’0100’B, 4.8 kbit/s
TSPX_DTMF	B_1	PIXIT MS	call control capabilities: value ’1’B means MS supports DTMF
TSPX_modF	CHMOD_VAL	PIXIT TC	any non signalling full rate channel mode for TC_26_6_4_1
TSPX_modH	CHMOD_VAL	PIXIT TC	any non signalling half channel mode for TC_26_6_4_1
TSPX_anymod	CHMOD_VAL	PIXIT TC	Any supported channel mode value except signalling and Full rate speech. possible values are ’00000011’B’ 12k data, ’00001011’B 6k data, ’0010011’B 3.6k data
TSPX_CKSNA	CKSN	PIXIT TC	cipher key sequence number, GSM 04.08, 10.5.1.2
TSPX_CKSNB	CKSN	PIXIT TC	cipher key sequence number, GSM 04.08, 10.5.1.2
TSPX_CKSNC	CKSN	PIXIT TC	cipher key sequence number, GSM 04.08, 10.5.1.2
TSPX_CKSNDef	CKSN	PIXIT TC	default cipher key sequence number, GSM 04.08, 10.5.1.2
TSPX_RfPwrCap	B_3	PIXIT MS	RF power capability − GSM ’000’B Class 1 to ’100’B Class 5 DCS ’000’B Class 1 to ’010’B Class 3
TSPX_RevLevel	B_2	PIXIT MS	Revision level of classmark 1 − ’00’B Phase 1, ’01’B Phase 2
TSPX_CiphAlgA5_1	B_1	PIXIT MS	A5/1 algorithm: ’0’B : available, ’1’B: not available
TSPX_CiphAlgA5_2	B_1	PIXIT MS	A5/2 algorithm: ’1’B : available, ’0’B: not available
TSPX_CiphAlgA5_3	B_1	PIXIT MS	A5/3 algorithm: ’1’B : available, ’0’B: not available
TSPX_CiphAlgA5_4	B_1	PIXIT MS	A5/4 algorithm: ’1’B : available, ’0’B: not available
TSPX_CiphAlgA5_5	B_1	PIXIT MS	A5/5 algorithm: ’1’B : available, ’0’B: not available
TSPX_CiphAlgA5_6	B_1	PIXIT MS	A5/6 algorithm: ’1’B : available, ’0’B: not available
TSPX_CiphAlgA5_7	B_1	PIXIT MS	A5/7 algorithm: ’1’B : available, ’0’B: not available
TSPX_pSyncCap	B_1	PIXIT MS	pseudo synchronisation capability, ’0’B: not present ’1’B: present

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_SSscrnInd	B_2	PIXIT MS	SS screen indicator −GSM 04.80
TSPX_SMCap	B_1	PXIIT MS	Support of mobile terminated point to point short messages, ’0’B: not present ’1’B: present
TSPX_frqCap	B_1	PIXIT MS	EGSM frequency capability. DCS: reserved, GSM: ’0’B not support extension band, ’1’B support extension band
TSPX_ClsMk3	B_1	PIXIT MS	classmark 3 indicator, ’0’B no addition capability info, ’1’B additional capability info in class mark 3
TSPX_RfPwrCapAmp	B_3	PIXIT MS	RF power capability with external RF amplifier − GSM ’000’B Class 1 to ’100’B Class 5 DCS ’000’B Class 1 to ’010’B Class 3
TSPX_CphAlgA	CPHALG	PIXIT TC	ciphering algorithm identifier − ’000’B A5/1 to ’110’ A5/7
TSPX_CphAlgB	CPHALG	PIXIT TC	ciphering algorithm identifier − ’000’B A5/1 to ’110’ A5/7
TSPX_CphAlgC	CPHALG	PIXIT TC	ciphering algorithm identifier − ’000’B A5/1 to ’110’ A5/7
TSPX_CphAlgD	CPHALG	PIXIT TC	ciphering algorithm identifier − ’000’B A5/1 to ’110’ A5/7
TSPX_CphAlgE	CPHALG	PIXIT TC	ciphering algorithm identifier − ’000’B A5/1 to ’110’ A5/7
TSPX_CphAlgDef	CPHALG	PIXIT TC	default ciphering algorithm identifier − ’000’B A5/1 to ’110’ A5/7
TSPX_DTMFInd	BOOLEAN	PIXIT MS	DTMF indication to user supported
TSPX_NoOfHoAccA	INTEGER	PIXIT TC	number of handover access bursts, value range 10 − 20 TC_26_6_5_1(M=1,M=8) TC_26_6_5_2_1 26_6_5_2_8 TC_26_6_5_4_1 TC_26_10_2_4_1 TC_26_10_2_4_2 TC_26_12_2_1_1 TC_26_12_2_1_4 TC_26_12_2_1_7 TC_26_12_2_1_11 TC_26_12_2_2_1 TC_26_12_2_2_4 TC_26_12_2_2_7 TC_26_12_2_2_8 TC_26_12_2_2_11 TC_26_11_2_2_1 TC_26_11_5_1 TC_26_11_5_2

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_NoOfHoAccB	INTEGER	PIXIT TC	number of handover access bursts, value range 10 − 20 TC_26_6_5_1(M=2) TC_26_6_5_2_6 TC_26_6_5_2_9 TC_26_12_2_1_2 TC_26_12_2_1_5 TC_26_12_2_1_8 TC_26_12_2_1_13 TC_26_12_2_2_2 TC_26_12_2_2_5 TC_26_12_2_2_9 TC_26_11_2_2_1
TSPX_NoOfHoAccC	INTEGER	PIXIT TC	number of handover access bursts, value range 10 − 20 TC_26_6_5_1(M=3) TC_26_6_5_2_7 TC_26_12_2_1_3 TC_26_12_2_1_6 TC_26_12_2_1_9 TC_26_12_2_1_15 TC_26_12_2_2_3 TC_26_12_2_2_6 TC_26_12_2_2_10 TC_26_11_2_2_1
TSPX_NoOfHoAccD	INTEGER	PIXIT TC	number of handover access bursts, value range 5 − 10 TC_26_6_5_1(M=4,M=7) TC_26_6_5_2_2 TC_26_12_2_1_10
TSPX_NoOfHoAccE	INTEGER	PIXIT TC	number of handover access bursts, value range 5 − 10 TC_26_6_5_1(M=5) TC_26_6_5_2_5 TC_26_12_2_1_12
TSPX_NoOfHoAccF	INTEGER	PIXIT TC	number of handover access bursts, value range 5 − 10 TC_26_6_5_1(M=6) TC_26_6_5_2_10 TC_26_12_2_1_14
TSPX_NoOfHoAccG	INTEGER	PIXIT TC	number of handover access bursts, value range 2 − 5 TC_26_6_5_2_3
TSPX_NoOfHoAccH	INTEGER	PIXIT TC	number of handover access bursts, value range 2 − 5 TC_26_6_5_2_4
TSPX_NoOfHoAccI	INTEGER	PIXIT TC	number of handover access bursts, value range 2 − 5 TC_26_10_2_4_1

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_HoRefA	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=1) TC_26_6_5_2_1 TC_26_6_5_4_1 TC_26 10_2_4_2 TC_26_12_2_1_1 TC_26_12_2_1_11 TC_26_11_2_2_1 TC_26_11_5_1 TC_26_11_5_2
TSPX_HoRefB	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=2,M=8) TC_26_6_5_2_2 TC_26_12_2_1_2 TC_26_12_2_1_12 TC_26_11_2_2_1
TSPX_HoRefC	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=3) TC_26_6_5_2_3 TC_26_12_2_1_3 TC_26_12_2_1_13 TC_26_12_2_2_3 TC_26_12_2_2_6 TC_26_12_2_2_10 TC_26_11_2_2_1
TSPX_HoRefD	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=4) TC_26_6_5_2_4 TC_26_12_2_1_4 TC_26_12_2_1_14
TSPX_HoRefE	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=5) TC_26_6_5_2_5 TC_26_12_2_1_5 TC_26_12_2_1_15
TSPX_HoRefF	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=6) TC_26_6_5_2_6 TC_26_12_2_1_6
TSPX_HoRefG	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_1(M=7) TC_26_6_5_2_7 TC_26_12_2_1_7
TSPX_HoRefH	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_2_8 TC_26_12_2_1_8
TSPX_HoRefI	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_2_9 TC_26_12_2_1_9

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_HoRefJ	HORF	PIXIT TC	Hand over reference, GSM 04.08, 10.5.2.15 TC_26_6_5_2_10 TC_26_12_2_1_10
TSPX_HSN	INTEGER	PIXIT TC	Hopping sequence number value range: 0 − 63. 0=cyclic hopping. Refer to GSM 11.10 for the value to be used in a particular test case
TSPX_IMSI	HEXSTRING	PIXIT MS	IMSI of the MS − Phase 2 Test SIM value: 001010123456063
TSPX_IMEI	HEXSTRING	PIXIT MS	IMEI of the MS.
TSPX_IMEISV	HEXSTRING	PIXIT MS	IMEISV of the MS. Used in: TC_26_6_8_5, TC_26_7_3_1
TSPX_Ki	BITSTRING	PIXIT MS	default authentication key used in testing − Phase 2 test SIM value: 0101111001001010101100 1101011000100100010011 0111010111010010101011 1011101000000100101110 0110011111000011000010 011010011000101001
TSPX_MAIO	INTEGER	PIXIT TC	mobile allocation index offset, value range: 0 − 63
TSPX_MaxRetrans	INTEGER	PIXIT TC	Max−Retrans − Maximum number of re−transmissions of Channel Requests as defined in System Information (values 1, 2, 4 or 7 retransmissions)
TSPX_MSTxpwrMax	MAXTXPOW	PIXIT TC	maximum output power from MS GSM 05.05, 4.11
TSPX_PwrlvlA	INTEGER	PIXIT TC	power level, value between 2 to 15 GSM 05.05, 4.11
TSPX_PwrlvlB	INTEGER	PIXIT TC	power level, value between 2 to 15 GSM 05.05, 4.11
TSPX_PwrlvlC	INTEGER	PIXIT TC	power level, value between 2 to 15 GSM 05.05, 4.11
TSPX_PwrlvlD	INTEGER	PIXIT TC	power level, value between 2 to 15 GSM 05.05, 4.11
TSPX_RANDA	RAND	PIXIT TC	challenge RAND
TSPX_RANDB	RAND	PIXIT TC	challenge RAND
TSPX_RANDC	RAND	PIXIT TC	challenge RAND
TSPX_RANDDef	RAND	PIXIT TC	default challenge RAND
TSPX_SDCCH4SubA	B_2	PIXIT TC	TDMA offset of SDCCH/4 subchannel
TSPX_SDCCH4SubB	B_2	PIXIT TC	TDMA offset of SDCCH/4 subchannel
TSPX_SDCCH4SubC	B_2	PIXIT TC	TDMA offset of SDCCH/4 subchannel
TSPX_SDCCH4SubDef	B_2	PIXIT TC	TDMA offset of default SDCCH/4 subchannel

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_SDCCH8SubA	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubB	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubC	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubD	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubE	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubF	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubG	B_3	PIXIT TC	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubDef	B_3	PIXIT TC	TDMA offset of default SDCCH/8 subchannel
TSPX_TC1M	INTEGER	PIXIT MS	timer value for GSM timer TC1M (for SMS)
TSPX_MaxCPDataRetx	INTEGER	PIXIT MS	max. number of CP data retransmissions for SMS
TSPX_Immconn	BOOLEAN	PIXIT MS	Immediate connect supported
TSPX_TimadvA	INTEGER	PIXIT TC	timing advance in bit periods
TSPX_TimadvB	INTEGER	PIXIT TC	timing advance in bit periods
TSPX_TimadvC	INTEGER	PIXIT TC	timing advance in bit periods
TSPX_TmSltA	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltB	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltC	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltD	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltE	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltF	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltG	SN	PIXIT TC	time slot GSM 04.08, 10.5.2.5
TSPX_TmSltDef	SN	PIXIT TC	default time slot GSM 04.08, 10.5.2.5
TSPX_TmSltNotZero	SN	PIXIT TC	time slot, arbitrarily value, but not zero.
TSPX_TmSltNotZero1	SN	PIXIT TC	time slot, arbitrarily value, but not zero and not TSPX_TmSltNotZero
TSPX_Txint	INTEGER	PIXIT TC	Tx−Integer, no. of slots to spread transmission (values 3,..,12,14,16,20,25,32 or 50)
TSPX_TscA	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_TscB	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5
TSPX_TscC	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5
TSPX_TscD	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5
TSPX_TscE	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5
TSPX_TscF	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5
TSPX_TscG	TSC	PIXIT TC	training sequence code GSM 04.08, 10.5.2.5
TSPX_TscDef	TSC	PIXIT TC	default training sequence code GSM 04.08, 10.5.2.5
TSPX_T3122	WI	PIXIT TC	value of wait indication (before mobile may attempt another Channel Request) timer T3122, hexstring representation of value in seconds
TSPX_TCHcarrierA_ho	INTEGER	PIXIT TC	the value can be choosen arbitrarily from cell allocation of cell A (GSM), but not BCCH carrier .possible values are : 10, 17, 26, 34, 42, 45, 46, 52, 59, 66, 73, 74, 75, 76, 108, 114
TSPX_TCHcarrierA_hod	INTEGER	PIXIT TC	the value can be choosen arbitrarily from cell allocation of cell A (DCS), but not BCCH carrier . possible values are : 734,741,754,759,762,766,767,773,775,779,782,791,798,829,832,844
TSPX_TCHcarrierB	INTEGER	PIXIT TC	TCH and SDCCH channel frequency number of cell B (GSM)
TSPX_TCHcarrierBd	INTEGER	PIXIT TC	TCH and SDCCH channel frequency number of cell B (DCS)
TSPX_TCHcarrierB_ho	INTEGER	PIXIT TC	the value can be choosen arbitrarily from cell allocation of cell B (GSM), but not BCCH carrier . possible values are : 14,18,22,24,30,31,38,60,66,73,74,75,76,108,114
TSPX_TCHcarrierB_hod	INTEGER	PIXIT TC	not BCCH carrier of cell B. the value can be choosen arbitrarily from cell allocation of cell B (DCS), which is (739,743,746,749,756,758,761,771,779,782,791,798,829,832,844)

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_TCHcarrierB2_ho	INTEGER	PIXIT TC	Choosen arbitrarly from cell allocation B for GSM HO cases, but not BCCH carrier!
TSPX_TCHcarrierB2_hod	INTEGER	PIXIT TC	Choosen arbitrarly from cell allocation B for DCS HO cases, but not BCCH carrier!
TSPX_TCHHSubA	B_1	PIXIT TC	TDMA offset of half rate subchannel
TSPX_TCHHSubDef	B_1	PIXIT TC	TDMA offset of default half rate subchannel
TSPX_TMSI	TMSI_V	PIXIT TC	TMSI of the MS used in test − standard value is 12345678
TSPX_TMSI1	TMSI_V	PIXIT TC	another TMSI used in test which shall differ from TSPX_TMSI, TSPX_TMSI + '01'O TSPX_TMSI + '02'O TSPX_TMSI + '03'O
TSPX_Uupd	B_8	PIXIT TC	user−user protocol discriminator for TC 26_8_3
TSPX_UuInfo	OCTETSTRING	PIXIT TC	user−user information for TC 26_8_3
TSPX_k	INTEGER	PIXIT TC	timing of cell A before cell B k bit periods for TC_26_6_5_1, TC_26_6_5_2_? TC_26_6_5_3_?, TC_26_6_5_4_?.
TSPX_y	INTEGER	PIXIT TC	timing advance for TC_26_6_5_3_?, TC_26_6_5_4_?, TC_26_6_5_5_? . The values of TSPX_k (or TSPX_k1) and TSPX_y: 0 < (2*TSPX_k + TSPX_y) MOD 256 < 60.
TSPX_k1	INTEGER	PIXIT TC	timing of cell A before cell B k bit periods for TC_26_6_5_5_?
TSPX_k2	INTEGER	PIXIT TC	timing of cell A before cell B k bit periods for TC_26_6_5_6
TSPX_y2	INTEGER	PIXIT TC	timing advance for TC_26_6_5_6, value range : 11 − 62.
TSPX_k3	INTEGER	PIXIT TC	timing of cell A before cell B k bit periods for TC_26_6_5_7
TSPX_y3	INTEGER	PIXIT TC	timing advance for TC_26_6_5_7

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_nPara	INTEGER	PIXIT TC	the n'th ChReq for TC_26_6_1_2, shall be [1..8], MAXRETRANS for TC_26_6_1_2 is 7 (See 11.10 for more explaination).
TSPX_kPara	INTEGER	PIXIT TC	the k'th ChReq for TC_26_6_1_2 (range 4..8)
TSPX_rPara	INTEGER	PIXIT TC	the r'th ChReq for TC_26_6_1_2 (range 4..8)
TSPX_i1Para	INTEGER	PIXIT TC	for TC_26_6_1_2 (TSPX_nPara−3)<TSPX_i1Para<=TSPX_nPara
TSPX_i2Para	INTEGER	PIXIT TC	for TC_26_6_1_2 0<TSPX_i2Para<=(TSPX_kPara − 3)
TSPX_i3Para	INTEGER	PIXIT TC	for TC_26_6_1_2 (TSPX_rPara − 3)< TSPX_i3Para <= TSPX_rPara
TSPX_n1Para	INTEGER	PIXIT TC	the n'th ChReq for TC_26_6_1_3 (range 1...8)
TSPX_i4Para	INTEGER	PIXIT TC	for TC_26_6_1_3 (TSPX_n1Para−3)<TSPX_i4Para<=TSPX_n1Para
TSPX_xPara	INTEGER	PIXIT TC	t3122 for TC_26_6_1_3 (range 5...255)
TSPX_AGBLKS1	INTEGER	PIXIT TC	BS−AG−BLKS−RES for TC_26_6_2_3_1, TC_26_6_2_1_1
TSPX_PAMFRMS1	INTEGER	PIXIT TC	BS−PA−MFRMS for TC_26_6_2_3_1 (range 2..8), TC_26_6_2_1_1
TSPX_CcchConf1	CCCH_CON	PIXIT TC	CCCH Configuration for TC_26_6_2_3_1, TC_26_6_2_1_1
TSPX_PgSubch	INTEGER	PIXIT TC	paging subchannel for TC_26_6_2_3_1
TSPX_AGBLKS2	INTEGER	PIXIT TC	BS−AG−BLKS−RES for TC_26_6_2_3_2 (range 0..2), TC_26_6_2_1_2
TSPX_PAMFRMS2	INTEGER	PIXIT TC	BS−PA−MFRMS for TC_26_6_2_3_2, TC_26_6_2_1_2
TSPX_CcchConf2	CCCH_CON	PIXIT TC	CCCH Configuration for TC_26_6_2_3_2, TC_26_6_2_1_2
TSPX_AGBLKS3	INTEGER	PIXIT TC	BS−AG−BLKS−RES for TC_26_6_2_4, TC_26_6_2_2, TC_26_6_2_1_3
TSPX_PAMFRMS3	INTEGER	PIXIT TC	BS−PA−MFRMS for TC_26_6_2_4, TC_26_6_2_2, TC_26_6_2_1_3

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_CcchConf3	CCCH_CON	PIXIT TC	CCCH Configuration for TC_26_6_2_4, TC_26_6_2_2, TC_26_6_2_1_3
TSPX_AGBLKS4	INTEGER	PIXIT TC	BS−AG−BLKS−RES for TC_26_6_2_5
TSPX_PAMFRMS4	INTEGER	PIXIT TC	BS−PA−MFRMS for TC_26_6_2_5
TSPX_CcchConf4	CCCH_CON	PIXIT TC	CCCH Configuration for TC_26_6_2_5 (shall be in the set ('010', '100', '110'))
TSPX_Chtp1	CH_TDMA	PIXIT TC	channel type and TDMA offset for TC_26_6_13_1, any value supported by the MS
TSPX_ChMod1	CHMOD_VAL	PIXIT TC	channel mode for TC_26_6_13_1, any value for the channel type of TSPX_Chtp1
TSPX_Maio1	INTEGER	PIXIT TC	Mobile allocation index offset hopping parameter for TC_26_6_13_1, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma1, default : 2
TSPX_Ma1	MAC	PIXIT TC	mobile allocation (GSM 04.08 10.5.2.21) for TC_26_6_13_1, its value shall indicate number of frequencies between 1 and 17. default : '010101'O
TSPX_Hsn1	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_1. default: 1
TSPX_Maio2	INTEGER	PIXIT TC	Mobile allocation index offset hopping parameter for TC_26_6_13_1, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma2, default : 8
TSPX_Ma2	MAC	PIXIT TC	mobile allocation for TC_26_6_13_1, its value shall indicate number of frequencies between 1 and 17. default :'01362A5'O
TSPX_Hsn2	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_1, default : 6
TSPX_Maio3	INTEGER	PIXIT TC	Mobile allocation index offset hopping parameter for TC_26_6_13_1, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma3, default : 14

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Ma3	MAC	PIXIT TC	mobile allocation for TC_26_6_13_1, its value shall indicate number of frequencies between 1 and 17. default :'01FFFA'O
TSPX_Hsn3	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_1, default : 0
TSPX_Chtp2	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_2, any value supported by the MS
TSPX_ChMod2	CHMOD_VAL	PIXIT TC	channel mode for TC_26_6_13_2
TSPX_Maio4	INTEGER	PIXIT TC	Mobile allocation index offset hopping parameter for TC_26_6_13_2, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma4, default : 3
TSPX_Ma4	MAC	PIXIT TC	mobile allocation for TC_26_6_13_2, its value shall indicate number of frequencies between 1 and 17. default : '001141'O
TSPX_Hsn4	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_2, default : 0
TSPX_Maio5	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_2, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma5, default : 9
TSPX_Ma5	MAC	PIXIT TC	mobile allocation for TC_26_6_13_2, its value shall indicate number of frequencies between 1 and 17. default : '01ABCB'O
TSPX_Hsn5	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_2, default : 16
TSPX_Chtp3	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_3, any value supported by the MS
TSPX_Maio6	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_3, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma6, default : 4
TSPX_Ma6	MAC	PIXIT TC	mobile allocation for TC_26_6_13_3, its value shall indicate number of frequencies between 1 and 17. default :'006248'O

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Hsn6	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_3, default : 4
TSPX_Maio7	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_3, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma7, default : 10
TSPX_Ma7	MAC	PIXIT TC	mobile allocation for TC_26_6_13_3, its value shall indicate number of frequencies between 2 and 17. default : ’009AFB’O
TSPX_Chtp4	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_3, any value supported by the MS
TSPX_Maio8	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_3, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma8, default : 1
TSPX_Ma8	MAC	PIXIT TC	mobile allocation for TC_26_6_13_3, its value shall indicate number of frequencies between 2 and 17. default : ’002800’O
TSPX_Hsn8	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_3, default : 40
TSPX_Maio9	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_3, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma9, default : 15
TSPX_Ma9	MAC	PIXIT TC	mobile allocation for TC_26_6_13_3, its value shall indicate number of frequencies between 2 and 17. default : ’01FFFB’O
TSPX_Hsn9	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_3, default : 8
TSPX_Chtp5	CH_TDMA	PIXIT TC	SDCCH8 subchannel immediately assigned, TC_26_6_13_4
TSPX_Maio10	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_4, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma10, default : 5

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Ma10	MAC	PIXIT TC	mobile allocation for TC_26_6_13_4, its value shall indicate number of frequencies between 1 and 17. default : '01D082'O
TSPX_Hsn10	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_4, default: 0
TSPX_Maio11	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_4, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma11, default : 11
TSPX_Ma11	MAC	PIXIT TC	mobile allocation for TC_26_6_13_4, its value shall indicate number of frequencies between 2 and 17. default : '00CFF3'O
TSPX_Chtp6	CH_TDMA	PIXIT TC	channel type of a non−existing channel in Assignment Cmd, TC_26_6_13_4
TSPX_Maio12	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_4, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma12, default : 1
TSPX_Ma12	MAC	PIXIT TC	mobile allocation for TC_26_6_13_4, its value shall indicate number of frequencies between 1 and 17. default : '010100'O
TSPX_Hsn12	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_4, default : 39
TSPX_Maio13	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_4, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma13, default : 6
TSPX_Ma13	MAC	PIXIT TC	mobile allocation for TC_26_6_13_4, its value shall indicate number of frequencies between 1 and 17. default : '00E690'O
TSPX_Hsn13	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_4, default: 42
TSPX_Chtp7	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_5
TSPX_ChMod4	CHMOD_VAL	PIXIT TC	channel mode for TC_26_6_13_5

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Maio14	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_5, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma14, default : 5
TSPX_Ma14	MAC	PIXIT TC	mobile allocation for TC_26_6_13_5, its value shall indicate number of frequencies between 1 and 17. default : '004A28'O
TSPX_Hsn14	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_5, default : 50
TSPX_Maio15	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_5, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma15, default : 12
TSPX_Ma15	MAC	PIXIT TC	mobile allocation for TC_26_6_13_5, its value shall indicate number of frequencies between 1 and 16. default : '00FF79'O
TSPX_Hsn15	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_5, default : 33
TSPX_Maio16	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_5, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma16, default : 2
TSPX_Ma16	MAC	PIXIT TC	mobile allocation for TC_26_6_13_5, its value shall indicate number of frequencies between 1 and 16. default : '001110'O
TSPX_Hsn16	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_5, default : 21
TSPX_Chtp8	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_6
TSPX_ChMod5	CHMOD_VAL	PIXIT TC	channel mode for TC_26_6_13_6
TSPX_Maio17	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_6, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma17, default : 7

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Ma17	MAC	PIXIT TC	mobile allocation for TC_26_6_13_6, its value shall indicate number of frequencies between 1 and 17. default : ’00E6A1’O
TSPX_Hsn17	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_6, default : 22
TSPX_Maio18	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_6, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma18, default : 13
TSPX_Ma18	MAC	PIXIT TC	mobile allocation for TC_26_6_13_6, its value shall indicate number of frequencies between 1 and 16. default : ’00FFF9’O
TSPX_Hsn18	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_6, default : 9
TSPX_Maio19	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_6, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma19, default : 3
TSPX_Ma19	MAC	PIXIT TC	mobile allocation for TC_26_6_13_6, its value shall indicate number of frequencies between 1 and 16. default : ’001111’O
TSPX_Hsn19	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_6, default : 44
TSPX_Chtp9	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_7
TSPX_ChMod6	CHMOD_VAL	PIXIT TC	channel mode for TC_26_6_13_7
TSPX_Maio20	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_7, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma20, default : 8
TSPX_Ma20	MAC	PIXIT TC	mobile allocation for TC_26_6_13_7, its value shall indicate number of frequencies between 1 and 17. default : ’016699’O
TSPX_Hsn20	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_7, default : 30

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Maio21	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_7, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma21, default : 14
TSPX_Ma21	MAC	PIXIT TC	mobile allocation for TC_26_6_13_7, its value shall indicate number of frequencies between 1 and 17. default : '01FEFB'O
TSPX_Chtp10	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_7
TSPX_Maio22	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_7, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma22, default : 4
TSPX_Ma22	MAC	PIXIT TC	mobile allocation for TC_26_6_13_7, its value shall indicate number of frequencies between 1 and 16. default : '006241'O
TSPX_Hsn22	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_7, default : 11
TSPX_Maio23	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_7, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma23, default : 11
TSPX_Ma23	MAC	PIXIT TC	mobile allocation for TC_26_6_13_7, its value shall indicate number of frequencies between 1 and 16. default : '00FF1B'O
TSPX_Hsn23	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_7, default : 60
TSPX_Chtp11	CH_TDMA	PIXIT TC	SDCCH8 subchannel immediately assigned, TC_26_6_13_8
TSPX_ChMod7	CHMOD_VAL	PIXIT TC	channel mode for TC_26_6_13_8
TSPX_Maio24	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_8, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma24, default : 9

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Ma24	MAC	PIXIT TC	mobile allocation for TC_26_6_13_8, its value shall indicate number of frequencies between 1 and 17. default : '00E6E9'O
TSPX_Hsn24	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_8, default : 7
TSPX_Maio25	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_8, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma25, default : 15
TSPX_Ma25	MAC	PIXIT TC	mobile allocation for TC_26_6_13_8, its value shall indicate number of frequencies between 1 and 17. default : '01FFFB'O
TSPX_Chtp12	CH_TDMA	PIXIT TC	channel type of a non−existing channel in Assignment Cmd, TC_26_6_13_8
TSPX_Maio26	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_8, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma26, default : 5
TSPX_Ma26	MAC	PIXIT TC	mobile allocation for TC_26_6_13_8, its value shall indicate number of frequencies between 1 and 16. default : '009168'O
TSPX_Hsn26	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_8, default : 9
TSPX_Maio27	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_8, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma27, default : 1
TSPX_Ma27	MAC	PIXIT TC	mobile allocation for TC_26_6_13_8, its value shall indicate number of frequencies between 1 and 16. default : '004080'O
TSPX_Hsn27	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_8, default : 38
TSPX_Chtp13	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_9

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Tm3	INTEGER	PIXIT TC	starting time, in frames, included in the Immediate Assignment message in TC_26_6_13_9, value between 60 −100
TSPX_Maio28	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_9, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma28, default : 10
TSPX_Ma28	MAC	PIXIT TC	mobile allocation for TC_26_6_13_9, its value shall indicate number of frequencies between 1 and 17. default : ’00EE7A’O
TSPX_Hsn28	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_9, default : 11
TSPX_Maio29	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_9, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma29, default : 6
TSPX_Ma29	MAC	PIXIT TC	mobile allocation for TC_26_6_13_9, its value shall indicate number of frequencies between 1 and 17. default : ’00F070’O
TSPX_Chtp14	CH_TDMA	PIXIT TC	channel type for TC_26_6_13_10
TSPX_Maio30	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_10, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma30, default : 11
TSPX_Ma30	MAC	PIXIT TC	mobile allocation for TC_26_6_13_10, its value shall indicate number of frequencies between 1 and 17. default : ’017F69’O
TSPX_Hsn30	INTEGER	PIXIT TC	hopping sequence number for TC_26_6_13_10, default : 62
TSPX_Maio31	INTEGER	PIXIT TC	mobile allocation index offset hopping parameter for TC_26_6_13_10, its value between 0 and (the number of frequencies) −1, which is defined in TSPX_Ma31, default : 1

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_Ma31	MAC	PIXIT TC	mobile allocation for TC_26_6_13_10, its value shall indicate number of frequencies between 1 and 17. default : '000101'O
TSPX_MOBscSvcA	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcB	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcC	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcD	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcE	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcF	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcG	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcH	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcI	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MOBscSvcJ	SERVICES	PIXIT TC	any supported MO basic service
TSPX_MT_DualModSvc	SERVICES	PIXIT TC	any supported MT dual mode service
TSPX_MT_NonSptSvc	SERVICES	PIXIT TC	any not supported MT basic service
TSPX_MTBscSvcA	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcB	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcC	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcD	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcE	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcF	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcG	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcH	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcI	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTBscSvcJ	SERVICES	PIXIT TC	any supported MT basic service
TSPX_MTNIC_BscSvcA	SERVICES	PIXIT TC	any supported MT basic service without immediate connection

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_MTNIC_BscSvcB	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcC	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcD	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcE	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcF	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcG	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcH	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcI	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_MTNIC_BscSvcJ	SERVICES	PIXIT TC	any supported MT basic service without immediate connection
TSPX_Telephony_Rate	RATE	PIXIT TC	channel rate for TS11, default value : "F"
TSPX_MOChRateA	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcA, default value : "F"
TSPX_MOChRateB	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcB, default value : "F"
TSPX_MOChRateC	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcC, default value : "F"
TSPX_MOChRateD	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcD, default value : "F"
TSPX_MOChRateE	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcE, default value : "F"
TSPX_MOChRateF	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcF, default value : "F"
TSPX_MOChRateG	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcG, default value : "F"
TSPX_MOChRateH	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcH, default value : "F"
TSPX_MOChRateI	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcI, default value : "F"

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_MOChRateJ	RATE	PIXIT TC	channel rate for TSPX_MOBscSvcJ, default value : "F"
TSPX_MTChRateA	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcA, default value : "F"
TSPX_MTChRateB	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcB, default value : "F"
TSPX_MTChRateC	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcC, default value : "F"
TSPX_MTChRateD	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcD, default value : "F"
TSPX_MTChRateE	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcE, default value : "F"
TSPX_MTChRateF	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcF, default value : "F"
TSPX_MTChRateG	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcG, default value : "F"
TSPX_MTChRateH	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcH, default value : "F"
TSPX_MTChRateI	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcI, default value : "F"
TSPX_MTChRateJ	RATE	PIXIT TC	channel rate for TSPX_MTBscSvcJ, default value : "F"
TSPX_MTNIC_ChRateA	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcA, default value : "F"
TSPX_MTNIC_ChRateB	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcB, default value : "F"
TSPX_MTNIC_ChRateC	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcC, default value : "F"
TSPX_MTNIC_ChRateD	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcD, default value : "F"
TSPX_MTNIC_ChRateE	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcE, default value : "F"
TSPX_MTNIC_ChRateF	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcF, default value : "F"
TSPX_MTNIC_ChRateG	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcG, default value : "F"

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_MTNIC_ChRateH	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcH, default value : "F"
TSPX_MTNIC_ChRateI	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcI, default value : "F"
TSPX_MTNIC_ChRateJ	RATE	PIXIT TC	channel rate for TSPX_MTNIC_BscSvcJ, default value : "F"
TSPX_EmgCallRate	RATE	PIXIT TC	Rate for the basic service supported for MO emergency calls
TSPX_MO_NonCallSS	SERVICES	PIXIT TC	any supported MO non−call releated supplementary service
TSPX_MO_BscSvc_SMS	SERVICES	PIXIT TC	any supported MO SMS calls
TSPX_MO_DualModSvc	SERVICES	PIXIT TC	any supported MO dual mode call
TSPX_MO_DualModRate	RATE	PIXIT TC	Rate for the supported MO dual mode call in TSPX_MO_DualModSvc
TSPX_EFR_SpchPrefsNo	INTEGER	PIXIT MS	Total no. of Speech Versions supported
TSPX_EFR_SpchPref1	B_4	PIXIT MS	Speech Version supported, No.1, ’0000’B full rate version 1, ’0010’B full rate version 2, ’0001’B half rate version 1
TSPX_EFR_SpchPref2	B_4	PIXIT MS	Speech Version supported, No.2, ’0000’B full rate version 1, ’0010’B full rate version 2, ’0001’B half rate version 1
TSPX_EFR_SpchPref3	B_4	PIXIT MS	Speech Version supported, No.3, ’0000’B full rate version 1, ’0010’B full rate version 2, ’0001’B half rate version 1
TSPX_ESIND	B_1	PIXIT MS	Controlled early ClassMark sending option implementation − default 0, not implemented
TSPX_MsrCellPresent	INTEGER	PIXIT MS	The number of measurement reports to be received − which include measurements of the second cell − prior to handover to that cell. Range 1 to 4 inclusive. Default = 4
Detailed Comments :			

Test Case Selection Expression Definitions		
Expression Name	**Selection Expression**	**Comments**
SelExpr_1100	TSPC_EFR	
SelExpr_1101	TSPC_MOsvc	
SelExpr_1102	TSPC_MTsvc	
SelExpr_0001	TRUE	
Detailed Comments :		

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_arfcnA	INTEGER	20	ARFCN 20
C_arfcn_1	INTEGER	1	ARFCN 1
C_arfcn_2	INTEGER	2	ARFCN 2
C_arfcn_1015	INTEGER	1015	ARFCN 1015
C_arfcn_10	INTEGER	10	ARFCN 10
C_arfcn_34	INTEGER	34	ARFCN 34
C_arfcnB	INTEGER	10	ARFCN of cell B
C_arfcnC	INTEGER	30	ARFCN 30
C_arfcnH	INTEGER	44	ARFCN 44
C_arfcnAd	INTEGER	590	ARFCN 590
C_arfcnAd_1	INTEGER	520	ARFCN 520
C_arfcnAd_2	INTEGER	514	ARFCN 514
C_arfcnAd_3	INTEGER	715	ARFCN 715
C_arfcnAd_4	INTEGER	747	ARFCN 747
C_arfcnAd_5	INTEGER	734	ARFCN 734
C_arfcnAd_6	INTEGER	759	ARFCN 759
C_arfcnBd	INTEGER	520	ARFCN 520
C_arfcnCd	INTEGER	700	ARFCN 700
C_arfcnHd	INTEGER	810	ARFCN 810
C_arfcn_tchA	INTEGER	30	ARFCN f30
C_arfcn_tchAd	INTEGER	650	ARFCN 650
C_arfcn_tchB	INTEGER	114	ARFCN 114
C_arfcn_tchBd	INTEGER	844	ARFCN 844
C_arfcnEgsm	INTEGER	990	ARFCN 990
C_arfcnEgsm_iacmd	INTEGER	20	ARFCN 20
C_arfcnEgsm_asscmd	INTEGER	20	ARFCN 20
C_BCCHcarrierA_ho	INTEGER	20	BCCH frequency number of cell A for Ho cases (GSM: 20)
C_BCCHcarrierA_hod	INTEGER	747	BCCH frequency number of cell A for Ho cases (DCS:747)
C_BCCHcarrierB_ho	INTEGER	40	BCCH frequency number of cell B for Ho cases (GSM: 40)
C_BCCHcarrierB_hod	INTEGER	764	BCCH frequency number of cell B for Ho cases (DCS:764)
C_Noarfcn	INTEGER	65534	No arfcn defined (hopping channel)
C_ATT_0	INTEGER	0	Attach value 0
C_ATT_1	INTEGER	1	Attach value 1
C_NCC	NCC	’001’B	Network color code
C_NCC_0	NCC	’000’B	Network color code value ’000’B
C_NCC_1	NCC	’001’B	Network color code value ’001’B
C_NCC_2	NCC	’010’B	Network color code value ’010’B
C_NCC_3	NCC	’011’B	Network color code value ’011’B

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_NCC_4	NCC	’100’B	Network color code value ’100’B
C_NCC_5	NCC	’101’B	Network color code value ’101’B
C_NCCP_2	NCCP	’02’O	Network color code permitted value ’02’O
C_NCCP_10	NCCP	’0A’O	Network color code permitted value ’0A’O
C_BCC	BCC	’101’B	Base Station color code
C_BCC_0	BCC	’000’B	Base Station color code value 0
C_BCC_1	BCC	’001’B	Base Station color code value 1
C_BCC_5	BCC	’101’B	Base Station color code value 5
C_cch_1nonComb	B_3	’000’B	cch_con = 1 basic ch non combined
C_cch_1Comb	B_3	’001’B	cch_con = 1 basic ch combined
C_cch_2Comb	B_3	’011’B	cch_con = 2 basic ch combined
C_cch_3Comb	B_3	’101’B	cch_con = 3 basic ch combined
C_cch_4Comb	B_3	’111’B	cch_con = 4 basic ch combined
C_cch_2nonComb	B_3	’010’B	cch_con = 2 basic chs non combined
C_cch_3nonComb	B_3	’100’B	cch_con = 3 basic chs non combined
C_cch_4nonComb	B_3	’110’B	cch_con = 4 basic chs non combined
C_cch_XnonComb	B_3	’??0’B	cch_con = non combined all number value permitted
C_cch_NotAllowed	B_3	’??1’B	cch_con = value reserved after testing ’001’B
C_cchd_e_407	OCTETSTRING	’8DEA0DF4CC6C4AFAFB10000000000000’O	f−list for cellchdescr using of 128 format. The coded set is {980, 981, 982, 983, 990, 991, 992, 993, 994, 1000, 1005, 1010, 1015}. Length = 16 TC_26_10_4, k=2, c=1 TC_26.10.6, c=1
C_cchd_e_412	OCTETSTRING	’00020800000007F020000008000080000’O	f−list for cellchdescr using of bit map 0 format. The coded set is {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114,115} Length = 16 TC_26_10_4, k=2, c=6
C_cchd_e_414	OCTETSTRING	’00000000000000000200002000020000000’O	f−list for cellchdescr using of bit map 0 format. Length = 16 f_list: 30, 50 ,70 TC: 26.10.6 c=6

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_cchd_e_415	OCTETSTRING	'841EEA893EF9814380000 00000000000'O	f−list for cellchdescr using range 1024 format. The coded set is {0, 30, 40, 66, 80, 1005, 1010, 1015}. Length = 16 TC_26_10_4, k=2, c=4 TC: 26.10.6 c=4
C_ChMod_s	CHMOD_VAL	'00000000'B	Mode : signalling only.
C_ChMod_r	CHMOD_VAL	'00000001'B	Mode : speech full or half rate.
C_ChMod_r2	CHMOD_VAL	'00100001'B	Mode : enhanced speech full or half rate − version 2.
C_ChMod_r3	CHMOD_VAL	'01000001'B	Mode : enhanced speech full or half rate − version 3.
C_ChMod_12k	CHMOD_VAL	'00000011'B	Mode : 12k radio rate.
C_ChMod_6k	CHMOD_VAL	'00001011'B	Mode : 6k radio rate.
C_ChMod_3k	CHMOD_VAL	'00010011'B	Mode : 3.6k radio rate.
C_Sap0	SAPID	'00'O	service access point 0
C_Sap3	SAPID	'03'O	service access point 3
C_CellA	CellID	"C_CellA"	cell A −− cell 1
C_CellB	CellID	"C_CellB"	cell B −− cell 2
C_CellC	CellID	"C_CellC"	cell C −− cell 3
C_CellD	CellID	"C_CellD"	cell D −− cell 4
C_CellE	CellID	"C_CellE"	cell E −− cell 5
C_CellF	CellID	"C_CellF"	cell F −− cell 6
C_CellG	CellID	"C_CellG"	cell G −− cell 7
C_CellH	CellID	"C_CellH"	cell H −− cell 8
C_ci_cellA	CI	'0001'O	Cell Id for cell A
C_ci_cellB	CI	'0002'O	Cell Id for cell B
C_ci_cellC	CI	'0003'O	Cell Id for cell C
C_ci_cellD	CI	'0004'O	Cell Id for cell D
C_ci_cellE	CI	'0005'O	Cell Id for cell E
C_ci_cellF	CI	'0006'O	Cell Id for cell F
C_ci_cellG	CI	'0007'O	Cell Id for cell G
C_ci_cellH	CI	'0008'O	Cell Id for cell H
C_cksn_nokey	BITSTRING	'111'B	No key available
C_E_default	INTEGER	63	Default field strength
C_E_neighbourdefault	INTEGER	53	Default field strength of neighbouring cells.
C_E_suitable	INTEGER	45	Field strength of a suitable cell
C_E_notsuitable	INTEGER	−10	Field strength of a not suitable cell for reselection
C_E_notsuitable_fSel	INTEGER	24	Field strength of a not suitable cell for selection

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_flist_e_401	OCTETSTRING	'8DF68AEC00'O	It includes the list of f's in EGSM test cases. The coded set is {1005, 1010, 1015}. range 128 format Length = 5 TC_26_10_4, k=1, c=1 TC: 26.10.5.1 k=1/2,c=1
C_flist_e_402	OCTETSTRING	'8A2481FF03F8'O	It includes the list of f's in EGSM test cases. The coded set is {73, 74, 75, 76, 77}, range 256 format. Length = 6 TC_26_10_4, k=1, c=2 TC: 26.10.5.1 k=1/2,c=2
C_flist_e_403	OCTETSTRING	'89EA00BFC040'O	It includes the list of f's in EGSM test cases. The coded set is {980, 981, 982, 983}, range 512 format. Length = 6 TC_26_10_4, k=1, c=3 TC: 26.10.5.1 k=1/2,c=3
C_flist_e_404	OCTETSTRING	'801EED02BEC0'O	It includes the list of f's in EGSM test cases. The coded set is {30, 40, 1010, 1015}, range 1024 format. Length = 6 TC_26_10_4, k=1, c=4 TC: 26.10.5.1 k=1/2,c=4
C_flist_e_405	OCTETSTRING	'8FEA001E000010'O	It includes the list of f's in EGSM test cases. The coded set is {980, 991, 992, 993, 994, 1015}. Length = 7 TC_26_10_4, k=1, c=5 TC: 26.10.5.1 k=1/2,c=5
C_flist_e_406	OCTETSTRING	'00000000000000002000008000080000'O	It includes the list of f's in EGSM test cases. The coded set is {20, 40, 66}, bitmap 0 format. Length = 16 TC_26_10_4, k=1, c=6 TC: 26.10.5.1 k=2,c=6
C_flist_e_407	OCTETSTRING	'8DEA0DF4CC6C4AFAFB10000000000000'O	It includes the list of f's in EGSM test cases. The coded set is {980, 981, 982, 983, 990, 991, 992, 993, 994, 1000, 1005, 1010, 1015}, range 128 format. Length = 16 TC_26_10_4, k=2, c=1 TC: 26.10.5.1 k=3,c=1 TC: 26.10.6 c=1

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_flist_e_408	OCTETSTRING	'8A0A1CFD3EF4610E2FFFFA0000'O	It includes the list of f's in EGSM test cases. The coded set is {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114, 115}, range 256 format. Length = 13 TC_26_10_4, k=2, c=2 TC: 26.10.5.1 k=3,c=2 TC: 26.10.6 c=2
C_flist_e_409	OCTETSTRING	'89EA037F433C7B042BFEBFEC10000000'O	It includes the list of f's in EGSM test cases. The coded set is {980, 981, 982, 983, 990, 991, 992, 993, 994, 1000, 1005, 1010, 1015}, range 512 format. Length = 16 TC_26_10_4, k=2, c=3 TC: 26.10.5.2 k=3,c=3 TC: 26.10.6 c=3
C_flist_e_410	OCTETSTRING	'841EEA893EF98143B161'O	Length = 10 TC: 26.10.5.2 k=3,c=4
C_flist_e_411	OCTETSTRING	'8FEA703E084210'O	It includes the list of f's in EGSM test cases. The coded set is {980, 981, 982, 983, 990, 991, 992, 993, 994, 1000, 1005, 1010, 1015}, variable bitmap format. Length = 7 TC_26_10_4, k=2, c=5 TC: 26.10.5.1 k=3,c=5 TC: 26.10.6 c=5
C_flist_e_412	OCTETSTRING	'00020800000007F020000008000080000'O	It includes the list of f's in EGSM test cases. The coded set is {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114, 115}, bitmap 0 format. Length = 16 TC_26_10_4, k=2, c=6 TC: 26.10.5.1 k=3,c=6
C_flist_e_415	OCTETSTRING	'841EEA893EF9814380'O	It includes the list of f's in EGSM test cases. The coded set is {0, 30, 40, 66, 80, 1005, 1010, 1015}, range 1024 format. Length = 9 Format: 1024 TC_26_10_4, k=2, c=4 TC: 26.10.6 c=4
C_flist_e_416	OCTETSTRING	'00020800000007F020000008000080000'O	bit map 0 format. The coded set is {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114} Length = 16
C_BABR_0	B_3	'000'B	BABR value 0
C_BABR_1	B_3	'001'B	BABR value 1
C_BABR_2	B_3	'010'B	BABR value 2
C_BPM_0	B_3	'000'B	BPM value 0, 2 multiframes

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_BPM_2	B_3	'010'B	BPM value 2, 4 multiframes
C_BPM_3	B_3	'011'B	BPM value 3, 5 multiframes
C_BPM_7	B_3	'111'B	BPM value 7, 9 multiframes
C_T3212_0	OCTETSTRING	'00'O	T3212 infinite
C_T3212_1	OCTETSTRING	'01'O	T3212 6 mn / 1 deci hours
C_T3212_2	OCTETSTRING	'02'O	T3212 12 mn / 2 deci hours
C_T3212_5	OCTETSTRING	'05'O	T3212 30 mn / 5 deci hours
C_IMSI	INTEGER	0	MI type: IMSI
C_TMSI	INTEGER	1	MI type: TMSI
C_IMEI	INTEGER	2	MI type: IMEI
C_IMEISV	INTEGER	3	MI type: IMEISV
C_shortIMSI	HEXSTRING	'001011234'H	A short IMSI
C_LAC_1	OCTETSTRING	'0001'O	lac 1.
C_LAC_2	OCTETSTRING	'0002'O	lac 2.
C_LAC_3	OCTETSTRING	'0003'O	lac 3.
C_LAC_4	OCTETSTRING	'0004'O	lac 4.
C_LAC_5	OCTETSTRING	'0005'O	lac 5.
C_LAC_6	OCTETSTRING	'0006'O	lac 6.
C_LAC_7	OCTETSTRING	'0007'O	lac 7.
C_LAC_8	OCTETSTRING	'0008'O	lac 8.
C_LAC_spe	OCTETSTRING	'5344'O	lac '5344'H.
C_lacdeleted	OCTETSTRING	'FFFE'O	lac value deleted.
C_MCC_1	OCTETSTRING	'00F1'O	MCC 1
C_MCC_2	OCTETSTRING	'00F2'O	MCC 2
C_MCC_3	OCTETSTRING	'00F3'O	MCC 3
C_MCC_4	OCTETSTRING	'00F4'O	MCC 4
C_MCC_5	OCTETSTRING	'00F5'O	MCC 5
C_MCC_6	OCTETSTRING	'00F6'O	MCC 6
C_MCC_7	OCTETSTRING	'00F7'O	MCC 7
C_MCC_8	OCTETSTRING	'00F8'O	MCC 8
C_MCC_9	OCTETSTRING	'013B'O	MCC 9
C_Max_1	INTEGER	1	Max Retrans value 1
C_Max_2	INTEGER	2	Max Retrans value 2
C_Max_7	INTEGER	7	Max Retrans value 7
C_NECI_0	INTEGER	0	NECI value 0, not supported
C_NECI_1	INTEGER	1	NECI value 1, supported
C_PLMN_1	OCTETSTRING	'10'O	PLMN 1 (mnc=01)
C_PLMN_2	OCTETSTRING	'20'O	PLMN 2 (mnc=02)
C_PLMN_3	OCTETSTRING	'30'O	PLMN 3 (mnc=03)
C_PLMN_4	OCTETSTRING	'F0'O	PLMN 4
C_PLMN_5	OCTETSTRING	'F2'O	PLMN 5
C_PLMN_6	OCTETSTRING	'F3'O	PLMN 6
C_PLMN_7	OCTETSTRING	'F4'O	PLMN 7
C_PLMN_8	OCTETSTRING	'F5'O	PLMN 8
C_PLMN_9	OCTETSTRING	'F6'O	PLMN 9
C_PLMN_10	OCTETSTRING	'F7'O	PLMN 10
C_PLMN_Home	OCTETSTRING	'10'O	PLMN 1 (mnc=01)

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_TxInt_5	INTEGER	5	Tx Integer value 5
C_TxInt_7	INTEGER	7	Tx Integer value 7
C_TxInt_10	INTEGER	10	Tx Integer value 10
C_NotCombined	BOOLEAN	FALSE	CCCH not combined with SDCCH
C_Combined	BOOLEAN	TRUE	CCCH combined with SDCCH
C_FCCH_A	LOGICCH	"C_FCCH_A"	frequency correction channel of cell A(1)
C_FCCH_B	LOGICCH	"C_FCCH_B"	frequency correction channel of cell B(2)
C_FCCH_C	LOGICCH	"C_FCCH_C"	frequency correction channel of cell C(3)
C_FCCH_D	LOGICCH	"C_FCCH_D"	frequency correction channel of cell D(4)
C_FCCH_E	LOGICCH	"C_FCCH_E"	frequency correction channel of cell E(5)
C_FCCH_F	LOGICCH	"C_FCCH_F"	frequency correction channel of cell F(6)
C_FCCH_G	LOGICCH	"C_FCCH_G"	frequency correction channel of cell G(7)
C_FCCH_H	LOGICCH	"C_FCCH_H"	frequency correction channel of cell H(8)
C_SCH_A	LOGICCH	"C_SCH_A"	sync channel of cell A(1)
C_SCH_B	LOGICCH	"C_SCH_B"	sync channel of cell B(2)
C_SCH_C	LOGICCH	"C_SCH_C"	sync channel of cell C(3)
C_SCH_D	LOGICCH	"C_SCH_D"	sync channel of cell D(4)
C_SCH_E	LOGICCH	"C_SCH_E"	sync channel of cell E(5)
C_SCH_F	LOGICCH	"C_SCH_F"	sync channel of cell F(6)
C_SCH_G	LOGICCH	"C_SCH_G"	sync channel of cell G(7)
C_SCH_H	LOGICCH	"C_SCH_H"	sync channel of cell H(8)
C_BCCH_A_1	LOGICCH	"C_BCCH_A_1"	1st broadcast channel of cell A
C_BCCH_A_2	LOGICCH	"C_BCCH_A_2"	2nd broadcast channel of cell A
C_BCCH_A_3	LOGICCH	"C_BCCH_A_3"	3rd broadcast channel of cell A
C_BCCH_A_4	LOGICCH	"C_BCCH_A_4"	4th broadcast channel of cell A
C_BCCH_B_1	LOGICCH	"C_BCCH_B_1"	1st broadcast channel of cell B
C_BCCH_C_1	LOGICCH	"C_BCCH_C_1"	1st broadcast channel of cell C
C_BCCH_D_1	LOGICCH	"C_BCCH_D_1"	1st broadcast channel of cell D
C_BCCH_E_1	LOGICCH	"C_BCCH_E_1"	1st broadcast channel of cell E
C_BCCH_F_1	LOGICCH	"C_BCCH_F_1"	1st broadcast channel of cell F
C_BCCH_G_1	LOGICCH	"C_BCCH_G_1"	1st broadcast channel of cell G

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_BCCH_H_1	LOGICCH	"C_BCCH_H_1"	1st broadcast channel of cell H
C_CBCH_A	LOGICCH	"C_CBCH_A"	Cell broadcast channel in cell A
C_AGCH_A_1	LOGICCH	"C_AGCH_A_1"	1st down link CCCH (AGCH) of cell A
C_AGCH_A_2	LOGICCH	"C_AGCH_A_2"	2nd down link CCCH (AGCH) of cell A
C_AGCH_A_3	LOGICCH	"C_AGCH_A_3"	3rd down link CCCH (AGCH) of cell A
C_AGCH_A_4	LOGICCH	"C_AGCH_A_4"	4th down link CCCH (AGCH) of cell A
C_AGCH_B_1	LOGICCH	"C_AGCH_B_1"	1st down link CCCH (AGCH) of cell B
C_AGCH_B_2	LOGICCH	"C_AGCH_B_2"	2nd down link CCCH (AGCH) of cell B
C_AGCH_B_3	LOGICCH	"C_AGCH_B_3"	3rd down link CCCH (AGCH) of cell B
C_AGCH_B_4	LOGICCH	"C_AGCH_B_4"	4th down link CCCH (AGCH) of cell B
C_AGCH_C_1	LOGICCH	"C_AGCH_C_1"	1st down link CCCH (AGCH) of cell C
C_AGCH_C_2	LOGICCH	"C_AGCH_C_2"	2nd down link CCCH (AGCH) of cell C
C_AGCH_C_3	LOGICCH	"C_AGCH_C_3"	3rd down link CCCH (AGCH) of cell C
C_AGCH_C_4	LOGICCH	"C_AGCH_C_4"	4th down link CCCH (AGCH) of cell C
C_AGCH_D_1	LOGICCH	"C_AGCH_D_1"	1st down link CCCH (AGCH) of cell D
C_AGCH_E_1	LOGICCH	"C_AGCH_E_1"	1st down link CCCH (AGCH) of cell E
C_AGCH_F_1	LOGICCH	"C_AGCH_F_1"	1st down link CCCH (AGCH) of cell F
C_AGCH_G_1	LOGICCH	"C_AGCH_G_1"	1st down link CCCH (AGCH) of cell G
C_AGCH_H_1	LOGICCH	"C_AGCH_H_1"	1st down link CCCH (AGCH) of cell H
C_PCH_A_1	LOGICCH	"C_PCH_A_1"	1st down link CCCH (PCH) of cell A
C_PCH_A_2	LOGICCH	"C_PCH_A_2"	2nd down link CCCH (PCH) of cell A
C_PCH_A_3	LOGICCH	"C_PCH_A_3"	3rd down link CCCH (PCH) of cell A
C_PCH_A_4	LOGICCH	"C_PCH_A_4"	4th down link CCCH (PCH) of cell A
C_PCH_B_1	LOGICCH	"C_PCH_B_1"	1st down link CCCH (PCH) of cell B
C_PCH_B_2	LOGICCH	"C_PCH_B_2"	2nd down link CCCH (PCH) of cell B
C_PCH_B_3	LOGICCH	"C_PCH_B_3"	3rd down link CCCH (PCH) of cell B

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_PCH_B_4	LOGICCH	"C_PCH_B_4"	4th down link CCCH (PCH) of cell B
C_PCH_C_1	LOGICCH	"C_PCH_C_1"	1st down link CCCH (PCH) of cell C
C_PCH_C_2	LOGICCH	"C_PCH_C_2"	2nd down link CCCH (PCH) of cell C
C_PCH_C_3	LOGICCH	"C_PCH_C_3"	3rd down link CCCH (PCH) of cell C
C_PCH_C_4	LOGICCH	"C_PCH_C_4"	4th down link CCCH (PCH) of cell C
C_PCH_D_1	LOGICCH	"C_PCH_D_1"	1st down link CCCH (PCH and AGCH) of cell D
C_PCH_E_1	LOGICCH	"C_PCH_E_1"	1st down link CCCH (PCH) of cell E
C_PCH_F_1	LOGICCH	"C_PCH_F_1"	1st down link CCCH (PCH) of cell F
C_PCH_G_1	LOGICCH	"C_PCH_G_1"	1st down link CCCH (PCH) of cell G
C_PCH_H_1	LOGICCH	"C_PCH_H_1"	1st down link CCCH (PCH) of cell H
C_RACH_A_1	LOGICCH	"C_RACH_A_1"	1st uplink CCCH (RACH) of cell A
C_RACH_A_2	LOGICCH	"C_RACH_A_2"	2nd uplink CCCH (RACH) of cell A
C_RACH_A_3	LOGICCH	"C_RACH_A_3"	3rd uplink CCCH (RACH) of cell A
C_RACH_A_4	LOGICCH	"C_RACH_A_4"	4th uplink CCCH (RACH) of cell A
C_RACH_B_1	LOGICCH	"C_RACH_B_1"	1st uplink CCCH (RACH) of cell B
C_RACH_B_2	LOGICCH	"C_RACH_B_1"	2nd uplink CCCH (RACH) of cell B
C_RACH_B_3	LOGICCH	"C_RACH_B_3"	3rd uplink CCCH (RACH) of cell B
C_RACH_B_4	LOGICCH	"C_RACH_B_4"	4th uplink CCCH (RACH) of cell B
C_RACH_C_1	LOGICCH	"C_RACH_C_1"	1st uplink CCCH (RACH) of cell C
C_RACH_C_2	LOGICCH	"C_RACH_C_2"	2nd uplink CCCH (RACH) of cell C
C_RACH_C_3	LOGICCH	"C_RACH_C_3"	3rd uplink CCCH (RACH) of cell C
C_RACH_C_4	LOGICCH	"C_RACH_C_4"	4th uplink CCCH (RACH) of cell C
C_RACH_D_1	LOGICCH	"C_RACH_D_1"	1st uplink CCCH (RACH) of cell D
C_RACH_E_1	LOGICCH	"C_RACH_E_1"	1st uplink CCCH (RACH) of cell E
C_RACH_F_1	LOGICCH	"C_RACH_F_1"	1st uplink CCCH (RACH) of cell F
C_RACH_G_1	LOGICCH	"C_RACH_G_1"	1st uplink CCCH (RACH) of cell G

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_RACH_H_1	LOGICCH	"C_RACH_H_1"	1st uplink CCCH (RACH) of cell H
C_FACCHF_A_1	LOGICCH	"C_FACCHF_A_1"	FACCH associated with 1st TCH/F of cell A
C_FACCHF_A_2	LOGICCH	"C_FACCHF_A_2"	FACCH associated with 2nd TCH/F of cell A
C_FACCHF_A_3	LOGICCH	"C_FACCHF_A_3"	FACCH associated with 3rd TCH/F of cell A
C_FACCHF_B_1	LOGICCH	"C_FACCHF_B_1"	FACCH associated with 1st TCH/F of cell B
C_FACCHF_B_2	LOGICCH	"C_FACCHF_B_2"	FACCH associated with 2nd TCH/F of cell B
C_FACCHF_B_3	LOGICCH	"C_FACCHF_B_3"	FACCH associated with 3rd TCH/F of cell B
C_FACCHF_C_1	LOGICCH	"C_FACCHF_C_1"	FACCH associated with 1sd TCH/F of cell C
C_FACCHF_C_2	LOGICCH	"C_FACCHF_C_2"	FACCH associated with 2nd TCH/F of cell C
C_FACCHF_C_3	LOGICCH	"C_FACCHF_C_3"	FACCH associated with 3rd TCH/F of cell C
C_FACCHF_H_1	LOGICCH	"C_FACCHF_H_1"	FACCH associated with 1st TCH/F of cell H
C_FACCHH_A_1	LOGICCH	"C_FACCHH_A_1"	FACCH associated with 1st TCH/H of cell A
C_FACCHH_A_2	LOGICCH	"C_FACCHH_A_2"	FACCH associated with 2nd TCH/H of cell A
C_FACCHH_A_3	LOGICCH	"C_FACCHH_A_3"	FACCH associated with 3rd TCH/H of cell A
C_FACCHH_B_1	LOGICCH	"C_FACCHH_B_1"	FACCH associated with 1st TCH/H of cell B
C_FACCHH_B_2	LOGICCH	"C_FACCHH_B_2"	FACCH associated with 2nd TCH/H of cell B
C_FACCHH_B_3	LOGICCH	"C_FACCHH_B_3"	FACCH associated with 3rd TCH/H of cell B
C_FACCHH_C_1	LOGICCH	"C_FACCHH_C_1"	FACCH associated with 1st TCH/H of cell C
C_FACCHH_C_2	LOGICCH	"C_FACCHH_C_2"	FACCH associated with 2nd TCH/H of cell C
C_FACCHH_C_3	LOGICCH	"C_FACCHH_C_3"	FACCH associated with 3rd TCH/H of cell C
C_FACCHH0_A_1	LOGICCH	"C_FACCHH0_A_1"	FACCH associated with 1st TCH/H0 of cell A
C_FACCHH1_A_1	LOGICCH	"C_FACCHH1_A_1"	FACCH associated with 1st TCH/H1 of cell A
C_FACCHH0_A_2	LOGICCH	"C_FACCHH0_A_2"	FACCH associated with 2nd TCH/H0 of cell A
C_FACCHH1_A_2	LOGICCH	"C_FACCHH1_A_2"	FACCH associated with 2nd TCH/H1 of cell A
C_FACCHH0_A_3	LOGICCH	"C_FACCHH0_A_3"	FACCH associated with 3rd TCH/H0 of cell A
C_FACCHH1_A_3	LOGICCH	"C_FACCHH1_A_3"	FACCH associated with 3rd TCH/H1 of cell A

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_FACCHH0_B_1	LOGICCH	"C_FACCHH0_B_1"	FACCH associated with 1st TCH/H0 of cell B
C_FACCHH1_B_1	LOGICCH	"C_FACCHH1_B_1"	FACCH associated with 1st TCH/H1 of cell B
C_FACCHH0_B_2	LOGICCH	"C_FACCHH0_B_2"	FACCH associated with 2nd TCH/H0 of cell B
C_FACCHH1_B_2	LOGICCH	"C_FACCHH1_B_2"	FACCH associated with 2nd TCH/H1 of cell B
C_FACCHH0_B_3	LOGICCH	"C_FACCHH0_B_3"	FACCH associated with 3rd TCH/H0 of cell B
C_FACCHH1_B_3	LOGICCH	"C_FACCHH1_B_3"	FACCH associated with 3rd TCH/H1 of cell B
C_FACCHH0_C_1	LOGICCH	"C_FACCHH0_C_1"	FACCH associated with 1st TCH/H0 of cell C
C_FACCHH1_C_1	LOGICCH	"C_FACCHH1_C_1"	FACCH associated with 1st TCH/H1 of cell C
C_FACCHH0_C_2	LOGICCH	"C_FACCHH0_C_2"	FACCH associated with 2nd TCH/H0 of cell C
C_FACCHH1_C_2	LOGICCH	"C_FACCHH1_C_2"	FACCH associated with 2nd TCH/H1 of cell C
C_FACCHH0_C_3	LOGICCH	"C_FACCHH0_C_3"	FACCH associated with 3rd TCH/H0 of cell C
C_FACCHH1_C_3	LOGICCH	"C_FACCHH1_C_3"	FACCH associated with 3rd TCH/H1 of cell C
C_FACCHH0_D_1	LOGICCH	"C_FACCHH0_D_1"	FACCH associated with 1st TCH/H0 of cell D
C_FACCHH1_D_1	LOGICCH	"C_FACCHH1_D_1"	FACCH associated with 1st TCH/H1 of cell D
C_FACCHH0_D_2	LOGICCH	"C_FACCHH0_D_2"	FACCH associated with 2nd TCH/H0 of cell D
C_FACCHH1_D_2	LOGICCH	"C_FACCHH1_D_2"	FACCH associated with 2nd TCH/H1 of cell D
C_FACCHH0_D_3	LOGICCH	"C_FACCHH0_D_3"	FACCH associated with 3rd TCH/H0 of cell D
C_FACCHH1_D_3	LOGICCH	"C_FACCHH1_D_3"	FACCH associated with 3rd TCH/H1 of cell D
C_FACCHH0_E_1	LOGICCH	"C_FACCHH0_E_1"	FACCH associated with 1st TCH/H0 of cell E
C_FACCHH1_E_1	LOGICCH	"C_FACCHH1_E_1"	FACCH associated with 1st TCH/H1 of cell E
C_FACCHH0_E_2	LOGICCH	"C_FACCHH0_E_2"	FACCH associated with 2nd TCH/H0 of cell E
C_FACCHH1_E_2	LOGICCH	"C_FACCHH1_E_2"	FACCH associated with 2nd TCH/H1 of cell E
C_FACCHH0_E_3	LOGICCH	"C_FACCHH0_E_3"	FACCH associated with 3rd TCH/H0 of cell E
C_FACCHH1_E_3	LOGICCH	"C_FACCHH1_E_3"	FACCH associated with 3rd TCH/H1 of cell E
C_FACCHH0_F_1	LOGICCH	"C_FACCHH0_F_1"	FACCH associated with 1st TCH/H0 of cell F
C_FACCHH1_F_1	LOGICCH	"C_FACCHH1_F_1"	FACCH associated with 1st TCH/H1 of cell F

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_FACCHH0_F_2	LOGICCH	"C_FACCHH0_F_2"	FACCH associated with 2nd TCH/H0 of cell F
C_FACCHH1_F_2	LOGICCH	"C_FACCHH1_F_2"	FACCH associated with 2nd TCH/H1 of cell F
C_FACCHH0_F_3	LOGICCH	"C_FACCHH0_F_3"	FACCH associated with 3rd TCH/H0 of cell F
C_FACCHH1_F_3	LOGICCH	"C_FACCHH1_F_3"	FACCH associated with 3rd TCH/H1 of cell F
C_FACCHH0_G_1	LOGICCH	"C_FACCHH0_G_1"	FACCH associated with 1st TCH/H0 of cell G
C_FACCHH1_G_1	LOGICCH	"C_FACCHH1_G_1"	FACCH associated with 1st TCH/H1 of cell G
C_FACCHH0_G_2	LOGICCH	"C_FACCHH0_G_2"	FACCH associated with 2nd TCH/H0 of cell G
C_FACCHH1_G_2	LOGICCH	"C_FACCHH1_G_2"	FACCH associated with 2nd TCH/H1 of cell G
C_FACCHH0_G_3	LOGICCH	"C_FACCHH0_G_3"	FACCH associated with 3rd TCH/H0 of cell G
C_FACCHH1_G_3	LOGICCH	"C_FACCHH1_G_3"	FACCH associated with 3rd TCH/H1 of cell G
C_FACCHH0_H_1	LOGICCH	"C_FACCHH0_H_1"	FACCH associated with 1st TCH/H0 of cell H
C_FACCHH1_H_1	LOGICCH	"C_FACCHH1_H_1"	FACCH associated with 1st TCH/H1 of cell H
C_FACCHH0_H_2	LOGICCH	"C_FACCHH0_H_2"	FACCH associated with 2nd TCH/H0 of cell H
C_FACCHH1_H_2	LOGICCH	"C_FACCHH1_H_2"	FACCH associated with 2nd TCH/H1 of cell H
C_FACCHH0_H_3	LOGICCH	"C_FACCHH0_H_3"	FACCH associated with 3rd TCH/H0 of cell H
C_FACCHH1_H_3	LOGICCH	"C_FACCHH1_H_3"	FACCH associated with 3rd TCH/H1 of cell H
C_SACCH_A	LOGICCH	"C_SACCH_A"	all SACCHs of cell A
C_SACCH_B	LOGICCH	"C_SACCH_B"	all SACCHs of cell B
C_SACCH_C	LOGICCH	"C_SACCH_C"	all SACCHs of cell C
C_SACCH_D	LOGICCH	"C_SACCH_D"	all SACCHs of cell D
C_SACCH_E	LOGICCH	"C_SACCH_E"	all SACCHs of cell E
C_SACCH_F	LOGICCH	"C_SACCH_F"	all SACCHs of cell F
C_SACCH_G	LOGICCH	"C_SACCH_G"	all SACCHs of cell G
C_SACCH_H	LOGICCH	"C_SACCH_H"	all SACCHs of cell H
C_SACCHF_A_1	LOGICCH	"C_SACCHF_A_1"	SACCH associated with 1st TCH/F of cell A
C_SACCHF_A_2	LOGICCH	"C_SACCHF_A_2"	SACCH associated with 2nd TCH/F of cell A
C_SACCHF_B_1	LOGICCH	"C_SACCHF_B_1"	SACCH associated with 1st TCH/F of cell B
C_SACCHF_B_2	LOGICCH	"C_SACCHF_B_2"	SACCH associated with 2nd TCH/F of cell B
C_SACCHF_H_1	LOGICCH	"C_SACCHF_H_1"	SACCH associated with 1st TCH/F of cell H
C_SACCHH_A_1	LOGICCH	"C_SACCHH_A_1"	SACCH associated with 1st TCH/H of cell A

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHH_A_2	LOGICCH	"C_SACCHH_A_2"	SACCH associated with 2nd TCH/H of cell A
C_SACCHH_A_3	LOGICCH	"C_SACCHH_A_3"	SACCH associated with 3rd TCH/H of cell A
C_SACCHH_B_1	LOGICCH	"C_SACCHH_B_1"	SACCH associated with 1st TCH/H of cell B
C_SACCHH_B_2	LOGICCH	"C_SACCHH_B_2"	SACCH associated with 2nd TCH/H of cell B
C_SACCHH_B_3	LOGICCH	"C_SACCHH_B_3"	SACCH associated with 3rd TCH/H of cell B
C_SACCHH_C_1	LOGICCH	"C_SACCHH_C_1"	SACCH associated with 1st TCH/H of cell C
C_SACCHH_C_2	LOGICCH	"C_SACCHH_C_2"	SACCH associated with 2nd TCH/H of cell C
C_SACCHH_C_3	LOGICCH	"C_SACCHH_C_3"	SACCH associated with 3rd TCH/H of cell C
C_SACCHH0_A_1	LOGICCH	"C_SACCHH0_A_1"	SACCH associated with 1st TCH/H_0 of cell A
C_SACCHH1_A_1	LOGICCH	"C_SACCHH1_A_1"	SACCH associated with 1st TCH/H_1 of cell A
C_SACCHH0_A_2	LOGICCH	"C_SACCHH0_A_2"	SACCH associated with 2nd TCH/H_0 of cell A
C_SACCHH1_A_2	LOGICCH	"C_SACCHH1_A_2"	SACCH associated with 2nd TCH/H_1 of cell A
C_SACCHH0_A_3	LOGICCH	"C_SACCHH0_A_3"	SACCH associated with 3rd TCH/H_0 of cell A
C_SACCHH1_A_3	LOGICCH	"C_SACCHH1_A_3"	SACCH associated with 3rd TCH/H_1 of cell A
C_SACCHH0_B_1	LOGICCH	"C_SACCHH0_B_1"	SACCH associated with 1st TCH/H_0 of cell B
C_SACCHH1_B_1	LOGICCH	"C_SACCHH1_B_1"	SACCH associated with 1st TCH/H_1 of cell B
C_SACCHH0_B_2	LOGICCH	"C_SACCHH0_B_2"	SACCH associated with 2nd TCH/H_0 of cell B
C_SACCHH1_B_2	LOGICCH	"C_SACCHH1_B_2"	SACCH associated with 2nd TCH/H_1 of cell B
C_SACCHH0_B_3	LOGICCH	"C_SACCHH0_B_3"	SACCH associated with 3rd TCH/H_0 of cell B
C_SACCHH1_B_3	LOGICCH	"C_SACCHH1_B_3"	SACCH associated with 3rd TCH/H_1 of cell B
C_SACCHH0_C_1	LOGICCH	"C_SACCHH0_C_1"	SACCH associated with 1st TCH/H_0 of cell C
C_SACCHH1_C_1	LOGICCH	"C_SACCHH1_C_1"	SACCH associated with 1st TCH/H_1 of cell C
C_SACCHH0_C_2	LOGICCH	"C_SACCHH0_C_2"	SACCH associated with 2nd TCH/H_0 of cell C
C_SACCHH1_C_2	LOGICCH	"C_SACCHH1_C_2"	SACCH associated with 2nd TCH/H_1 of cell C
C_SACCHH0_C_3	LOGICCH	"C_SACCHH0_C_3"	SACCH associated with 3rd TCH/H_0 of cell C
C_SACCHH1_C_3	LOGICCH	"C_SACCHH1_C_3"	SACCH associated with 3rd TCH/H_1 of cell C

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHH0_D_1	LOGICCH	"C_SACCHH0_D_1"	SACCH associated with 1st TCH/H_0 of cell D
C_SACCHH1_D_1	LOGICCH	"C_SACCHH1_D_1"	SACCH associated with 1st TCH/H_1 of cell D
C_SACCHH0_D_2	LOGICCH	"C_SACCHH0_D_2"	SACCH associated with 2nd TCH/H_0 of cell D
C_SACCHH1_D_2	LOGICCH	"C_SACCHH1_D_2"	SACCH associated with 2nd TCH/H_1 of cell D
C_SACCHH0_D_3	LOGICCH	"C_SACCHH0_D_3"	SACCH associated with 3rd TCH/H_0 of cell D
C_SACCHH1_D_3	LOGICCH	"C_SACCHH1_D_3"	SACCH associated with 3rd TCH/H_1 of cell D
C_SACCHH0_E_1	LOGICCH	"C_SACCHH0_E_1"	SACCH associated with 1st TCH/H_0 of cell E
C_SACCHH1_E_1	LOGICCH	"C_SACCHH1_E_1"	SACCH associated with 1st TCH/H_1 of cell E
C_SACCHH0_E_2	LOGICCH	"C_SACCHH0_E_2"	SACCH associated with 2nd TCH/H_0 of cell E
C_SACCHH1_E_2	LOGICCH	"C_SACCHH1_E_2"	SACCH associated with 2nd TCH/H_1 of cell E
C_SACCHH0_E_3	LOGICCH	"C_SACCHH0_E_3"	SACCH associated with 3rd TCH/H_0 of cell E
C_SACCHH1_E_3	LOGICCH	"C_SACCHH1_E_3"	SACCH associated with 3rd TCH/H_1 of cell E
C_SACCHH0_F_1	LOGICCH	"C_SACCHH0_F_1"	SACCH associated with 1st TCH/H_0 of cell F
C_SACCHH1_F_1	LOGICCH	"C_SACCHH1_F_1"	SACCH associated with 1st TCH/H_1 of cell F
C_SACCHH0_F_2	LOGICCH	"C_SACCHH0_F_2"	SACCH associated with 2nd TCH/H_0 of cell F
C_SACCHH1_F_2	LOGICCH	"C_SACCHH1_F_2"	SACCH associated with 2nd TCH/H_1 of cell F
C_SACCHH0_F_3	LOGICCH	"C_SACCHH0_F_3"	SACCH associated with 3rd TCH/H_0 of cell F
C_SACCHH1_F_3	LOGICCH	"C_SACCHH1_F_3"	SACCH associated with 3rd TCH/H_1 of cell F
C_SACCHH0_G_1	LOGICCH	"C_SACCHH0_G_1"	SACCH associated with 1st TCH/H_0 of cell G
C_SACCHH1_G_1	LOGICCH	"C_SACCHH1_G_1"	SACCH associated with 1st TCH/H_1 of cell G
C_SACCHH0_G_2	LOGICCH	"C_SACCHH0_G_2"	SACCH associated with 2nd TCH/H_0 of cell G
C_SACCHH1_G_2	LOGICCH	"C_SACCHH1_G_2"	SACCH associated with 2nd TCH/H_1 of cell G
C_SACCHH0_G_3	LOGICCH	"C_SACCHH0_G_3"	SACCH associated with 3rd TCH/H_0 of cell G
C_SACCHH1_G_3	LOGICCH	"C_SACCHH1_G_3"	SACCH associated with 3rd TCH/H_1 of cell G
C_SACCHH0_H_1	LOGICCH	"C_SACCHH0_H_1"	SACCH associated with 1st TCH/H_0 of cell H
C_SACCHH1_H_1	LOGICCH	"C_SACCHH1_H_1"	SACCH associated with 1st TCH/H_1 of cell H

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHH0_H_2	LOGICCH	"C_SACCHH0_H_2"	SACCH associated with 2nd TCH/H_0 of cell H
C_SACCHH1_H_2	LOGICCH	"C_SACCHH1_H_2"	SACCH associated with 2nd TCH/H_1 of cell H
C_SACCHH0_H_3	LOGICCH	"C_SACCHH0_H_3"	SACCH associated with 3rd TCH/H_0 of cell H
C_SACCHH1_H_3	LOGICCH	"C_SACCHH1_H_3"	SACCH associated with 3rd TCH/H_1 of cell H
C_SACCHC4_A	LOGICCH	"C_SACCHC4_A"	SACCHC4 channel of cell A
C_SACCHC4_B	LOGICCH	"C_SACCHC4_B"	SACCHC4 channel of cell B
C_SACCHC4_C	LOGICCH	"C_SACCHC4_C"	SACCHC4 channel of cell C
C_SACCHC4_D	LOGICCH	"C_SACCHC4_D"	SACCHC4 channel of cell D
C_SACCHC4_E	LOGICCH	"C_SACCHC4_E"	SACCHC4 channel of cell E
C_SACCHC4_F	LOGICCH	"C_SACCHC4_F"	SACCHC4 channel of cell F
C_SACCHC4_G	LOGICCH	"C_SACCHC4_G"	SACCHC4 channel of cell G
C_SACCHC4_H	LOGICCH	"C_SACCHC4_H"	SACCHC4 channel of cell H
C_SACCHC40_A	LOGICCH	"C_SACCHC40_A"	SACCH/C4(0) associated with SDCCH/4(0) of cell A
C_SACCHC41_A	LOGICCH	"C_SACCHC41_A"	SACCH/C4(1) associated with SDCCH/4(1) of cell A
C_SACCHC42_A	LOGICCH	"C_SACCHC42_A"	SACCH/C4(2) associated with SDCCH/4(2) of cell A
C_SACCHC43_A	LOGICCH	"C_SACCCH43_A"	SACCH/C4(3) associated with SDCCH/4(3) of cell A
C_SACCHC40_B	LOGICCH	"C_SACCHC40_B"	SACCH/C4(0) associated with SDCCH/4(0) of cell B
C_SACCHC41_B	LOGICCH	"C_SACCHC41_B"	SACCH/C4(1) associated with SDCCH/4(1) of cell B
C_SACCHC42_B	LOGICCH	"C_SACCHC42_B"	SACCH/C4(2) associated with SDCCH/4(2) of cell B
C_SACCHC43_B	LOGICCH	"C_SACCHC43_B"	SACCH/C4(3) associated with SDCCH/4(3) of cell B
C_SACCHC40_C	LOGICCH	"C_SACCHC40_C"	SACCH/C4(0) associated with SDCCH/4(0) of cell C
C_SACCHC41_C	LOGICCH	"C_SACCHC41_C"	SACCH/C4(1) associated with SDCCH/4(1) of cell C
C_SACCHC42_C	LOGICCH	"C_SACCHC42_C"	SACCH/C4(2) associated with SDCCH/4(2) of cell C
C_SACCHC43_C	LOGICCH	"C_SACCHC43_C"	SACCH/C4(3) associated with SDCCH/4(3) of cell C
C_SACCHC40_D	LOGICCH	"C_SACCHC40_D"	SACCH/C4(0) associated with SDCCH/4(0) of cell D
C_SACCHC41_D	LOGICCH	"C_SACCHC41_D"	SACCH/C4(1) associated with SDCCH/4(1) of cell D
C_SACCHC42_D	LOGICCH	"C_SACCHC42_D"	SACCH/C4(2) associated with SDCCH/4(2) of cell D
C_SACCHC43_D	LOGICCH	"C_SACCCH43_D"	SACCH/C4(3) associated with SDCCH/4(3) of cell D
C_SACCHC40_E	LOGICCH	"C_SACCHC40_E"	SACCH/C4(0) associated with SDCCH/4(0) of cell E
C_SACCHC41_E	LOGICCH	"C_SACCHC41_E"	SACCH/C4(1) associated with SDCCH/4(1) of cell E

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC42_E	LOGICCH	"C_SACCHC42_E"	SACCH/C4(2) associated with SDCCH/4(2) of cell E
C_SACCHC43_E	LOGICCH	"C_SACCHC43_E"	SACCH/C4(3) associated with SDCCH/4(3) of cell E
C_SACCHC40_F	LOGICCH	"C_SACCHC40_F"	SACCH/C4(0) associated with SDCCH/4(0) of cell F
C_SACCHC41_F	LOGICCH	"C_SACCHC41_F"	SACCH/C4(1) associated with SDCCH/4(1) of cell F
C_SACCHC42_F	LOGICCH	"C_SACCHC42_F"	SACCH/C4(2) associated with SDCCH/4(2) of cell F
C_SACCHC43_F	LOGICCH	"C_SACCHC43_F"	SACCH/C4(3) associated with SDCCH/4(3) of cell F
C_SACCHC40_G	LOGICCH	"C_SACCHC40_G"	SACCH/C4(0) associated with SDCCH/4(0) of cell G
C_SACCHC41_G	LOGICCH	"C_SACCHC41_G"	SACCH/C4(1) associated with SDCCH/4(1) of cell G
C_SACCHC42_G	LOGICCH	"C_SACCHC42_G"	SACCH/C4(2) associated with SDCCH/4(2) of cell G
C_SACCHC43_G	LOGICCH	"C_SACCHC43_G"	SACCH/C4(3) associated with SDCCH/4(3) of cell G
C_SACCHC40_H	LOGICCH	"C_SACCHC40_C"	SACCH/C4(0) associated with SDCCH/4(0) of cell H
C_SACCHC41_H	LOGICCH	"C_SACCHC41_C"	SACCH/C4(1) associated with SDCCH/4(1) of cell H
C_SACCHC42_H	LOGICCH	"C_SACCHC42_C"	SACCH/C4(2) associated with SDCCH/4(2) of cell H
C_SACCHC43_H	LOGICCH	"C_SACCHC43_C"	SACCH/C4(3) associated with SDCCH/4(3) of cell H
C_SACCHC8_A_1	LOGICCH	"C_SACCHC8_A_1"	1st SACCH/8 channel of cell A
C_SACCHC8_A_2	LOGICCH	"C_SACCHC8_A_2"	2nd SACCH/8 channel of cell A
C_SACCHC8_A_3	LOGICCH	"C_SACCHC8_A_3"	3rd SACCH/8 channel of cell A
C_SACCHC8_B_1	LOGICCH	"C_SACCHC8_B_1"	1st SACCH/8 channel of cell B
C_SACCHC8_B_2	LOGICCH	"C_SACCHC8_B_2"	2nd SACCH/8 channel of cell B
C_SACCHC8_B_3	LOGICCH	"C_SACCHC8_B_3"	3rd SACCH/8 channel of cell B
C_SACCHC8_C_1	LOGICCH	"C_SACCHC8_C_1"	1st SACCH/8 channel of cell C
C_SACCHC8_C_2	LOGICCH	"C_SACCHC8_C_2"	2nd SACCH/8 channel of cell C
C_SACCHC8_C_3	LOGICCH	"C_SACCHC8_C_3"	3rd SACCH/8 channel of cell C
C_SACCHC80_A_1	LOGICCH	"C_SACCHC80_A_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell A
C_SACCHC81_A_1	LOGICCH	"C_SACCHC81_A_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell A

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC82_A_1	LOGICCH	"C_SACCHC82_A_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell A
C_SACCHC83_A_1	LOGICCH	"C_SACCHC83_A_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell A
C_SACCHC84_A_1	LOGICCH	"C_SACCHC84_A_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell A
C_SACCHC85_A_1	LOGICCH	"C_SACCHC85_A_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell A
C_SACCHC86_A_1	LOGICCH	"C_SACCHC86_A_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell A
C_SACCHC87_A_1	LOGICCH	"C_SACCHC87_A_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell A
C_SACCHC80_A_2	LOGICCH	"C_SACCHC80_A_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell A
C_SACCHC81_A_2	LOGICCH	"C_SACCHC81_A_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell A
C_SACCHC82_A_2	LOGICCH	"C_SACCHC82_A_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell A
C_SACCHC83_A_2	LOGICCH	"C_SACCHC83_A_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell A
C_SACCHC84_A_2	LOGICCH	"C_SACCHC84_A_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell A
C_SACCHC85_A_2	LOGICCH	"C_SACCHC85_A_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell A
C_SACCHC86_A_2	LOGICCH	"C_SACCHC86_A_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell A
C_SACCHC87_A_2	LOGICCH	"C_SACCHC87_A_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell A
C_SACCHC80_A_3	LOGICCH	"C_SACCHC80_A_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell A
C_SACCHC81_A_3	LOGICCH	"C_SACCHC81_A_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell A
C_SACCHC82_A_3	LOGICCH	"C_SACCHC82_A_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell A
C_SACCHC83_A_3	LOGICCH	"C_SACCHC83_A_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell A

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC84_A_3	LOGICCH	"C_SACCHC84_A_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell A
C_SACCHC85_A_3	LOGICCH	"C_SACCHC85_A_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell A
C_SACCHC86_A_3	LOGICCH	"C_SACCHC86_A_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell A
C_SACCHC87_A_3	LOGICCH	"C_SACCHC87_A_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell A
C_SACCHC80_B_1	LOGICCH	"C_SACCHC80_B_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell B
C_SACCHC81_B_1	LOGICCH	"C_SACCHC81_B_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell B
C_SACCHC82_B_1	LOGICCH	"C_SACCHC82_B_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell B
C_SACCHC83_B_1	LOGICCH	"C_SACCHC83_B_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell B
C_SACCHC84_B_1	LOGICCH	"C_SACCHC84_B_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell B
C_SACCHC85_B_1	LOGICCH	"C_SACCHC85_B_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell B
C_SACCHC86_B_1	LOGICCH	"C_SACCHC86_B_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell B
C_SACCHC87_B_1	LOGICCH	"C_SACCHC87_B_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell B
C_SACCHC80_B_2	LOGICCH	"C_SACCHC80_B_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell B
C_SACCHC81_B_2	LOGICCH	"C_SACCHC81_B_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell B
C_SACCHC82_B_2	LOGICCH	"C_SACCHC82_B_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell B
C_SACCHC83_B_2	LOGICCH	"C_SACCHC83_B_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell B
C_SACCHC84_B_2	LOGICCH	"C_SACCHC84_B_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell B
C_SACCHC85_B_2	LOGICCH	"C_SACCHC85_B_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell B

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC86_B_2	LOGICCH	"C_SACCHC86_B_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell B
C_SACCHC87_B_2	LOGICCH	"C_SACCHC87_B_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell B
C_SACCHC80_B_3	LOGICCH	"C_SACCHC80_B_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell B
C_SACCHC81_B_3	LOGICCH	"C_SACCHC81_B_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell B
C_SACCHC82_B_3	LOGICCH	"C_SACCHC82_B_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell B
C_SACCHC83_B_3	LOGICCH	"C_SACCHC83_B_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell B
C_SACCHC84_B_3	LOGICCH	"C_SACCHC84_B_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell B
C_SACCHC85_B_3	LOGICCH	"C_SACCHC85_B_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell B
C_SACCHC86_B_3	LOGICCH	"C_SACCHC86_B_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell B
C_SACCHC87_B_3	LOGICCH	"C_SACCHC87_B_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell B
C_SACCHC80_C_1	LOGICCH	"C_SACCHC80_C_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell C
C_SACCHC81_C_1	LOGICCH	"C_SACCHC81_C_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell C
C_SACCHC82_C_1	LOGICCH	"C_SACCHC82_C_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell C
C_SACCHC83_C_1	LOGICCH	"C_SACCHC83_C_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell C
C_SACCHC84_C_1	LOGICCH	"C_SACCHC84_C_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell C
C_SACCHC85_C_1	LOGICCH	"C_SACCHC85_C_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell C
C_SACCHC86_C_1	LOGICCH	"C_SACCHC86_C_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell C
C_SACCHC87_C_1	LOGICCH	"C_SACCHC87_C_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell C

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC80_C_2	LOGICCH	"C_SACCHC80_C_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell C
C_SACCHC81_C_2	LOGICCH	"C_SACCHC81_C_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell C
C_SACCHC82_C_2	LOGICCH	"C_SACCHC82_C_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell C
C_SACCHC83_C_2	LOGICCH	"C_SACCHC83_C_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell C
C_SACCHC84_C_2	LOGICCH	"C_SACCHC84_C_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell C
C_SACCHC85_C_2	LOGICCH	"C_SACCHC85_C_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell C
C_SACCHC86_C_2	LOGICCH	"C_SACCHC86_C_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell C
C_SACCHC87_C_2	LOGICCH	"C_SACCHC87_C_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell C
C_SACCHC80_C_3	LOGICCH	"C_SACCHC80_C_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell C
C_SACCHC81_C_3	LOGICCH	"C_SACCHC81_C_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell C
C_SACCHC82_C_3	LOGICCH	"C_SACCHC82_C_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell C
C_SACCHC83_C_3	LOGICCH	"C_SACCHC83_C_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell C
C_SACCHC84_C_3	LOGICCH	"C_SACCHC84_C_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell C
C_SACCHC85_C_3	LOGICCH	"C_SACCHC85_C_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell C
C_SACCHC86_C_3	LOGICCH	"C_SACCHC86_C_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell C
C_SACCHC87_C_3	LOGICCH	"C_SACCHC87_C_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell C
C_SACCHC80_D_1	LOGICCH	"C_SACCHC80_D_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell D
C_SACCHC81_D_1	LOGICCH	"C_SACCHC81_D_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell D

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC82_D_1	LOGICCH	"C_SACCHC82_D_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell D
C_SACCHC83_D_1	LOGICCH	"C_SACCHC83_D_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell D
C_SACCHC84_D_1	LOGICCH	"C_SACCHC84_D_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell D
C_SACCHC85_D_1	LOGICCH	"C_SACCHC85_D_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell D
C_SACCHC86_D_1	LOGICCH	"C_SACCHC86_D_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell D
C_SACCHC87_D_1	LOGICCH	"C_SACCHC87_D_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell D
C_SACCHC80_D_2	LOGICCH	"C_SACCHC80_D_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell D
C_SACCHC81_D_2	LOGICCH	"C_SACCHC81_D_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell D
C_SACCHC82_D_2	LOGICCH	"C_SACCHC82_D_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell D
C_SACCHC83_D_2	LOGICCH	"C_SACCHC83_D_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell D
C_SACCHC84_D_2	LOGICCH	"C_SACCHC84_D_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell D
C_SACCHC85_D_2	LOGICCH	"C_SACCHC85_D_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell D
C_SACCHC86_D_2	LOGICCH	"C_SACCHC86_D_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell D
C_SACCHC87_D_2	LOGICCH	"C_SACCHC87_D_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell D
C_SACCHC80_D_3	LOGICCH	"C_SACCHC80_D_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell D
C_SACCHC81_D_3	LOGICCH	"C_SACCHC81_D_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell D
C_SACCHC82_D_3	LOGICCH	"C_SACCHC82_D_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell D
C_SACCHC83_D_3	LOGICCH	"C_SACCHC83_D_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell D

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC84_D_3	LOGICCH	"C_SACCHC84_D_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell D
C_SACCHC85_D_3	LOGICCH	"C_SACCHC85_D_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell D
C_SACCHC86_D_3	LOGICCH	"C_SACCHC86_D_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell D
C_SACCHC87_D_3	LOGICCH	"C_SACCHC87_D_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell D
C_SACCHC80_E_1	LOGICCH	"C_SACCHC80_E_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell E
C_SACCHC81_E_1	LOGICCH	"C_SACCHC81_E_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell E
C_SACCHC82_E_1	LOGICCH	"C_SACCHC82_E_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell E
C_SACCHC83_E_1	LOGICCH	"C_SACCHC83_E_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell E
C_SACCHC84_E_1	LOGICCH	"C_SACCHC84_E_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell E
C_SACCHC85_E_1	LOGICCH	"C_SACCHC85_E_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell E
C_SACCHC86_E_1	LOGICCH	"C_SACCHC86_E_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell E
C_SACCHC87_E_1	LOGICCH	"C_SACCHC87_E_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell E
C_SACCHC80_E_2	LOGICCH	"C_SACCHC80_E_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell E
C_SACCHC81_E_2	LOGICCH	"C_SACCHC81_E_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell E
C_SACCHC82_E_2	LOGICCH	"C_SACCHC82_E_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell E
C_SACCHC83_E_2	LOGICCH	"C_SACCHC83_E_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell E
C_SACCHC84_E_2	LOGICCH	"C_SACCHC84_E_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell E
C_SACCHC85_E_2	LOGICCH	"C_SACCHC85_E_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell E

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC86_E_2	LOGICCH	"C_SACCHC86_E_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell E
C_SACCHC87_E_2	LOGICCH	"C_SACCHC87_E_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell E
C_SACCHC80_E_3	LOGICCH	"C_SACCHC80_E_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell E
C_SACCHC81_E_3	LOGICCH	"C_SACCHC81_E_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell E
C_SACCHC82_E_3	LOGICCH	"C_SACCHC82_E_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell E
C_SACCHC83_E_3	LOGICCH	"C_SACCHC83_E_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell E
C_SACCHC84_E_3	LOGICCH	"C_SACCHC84_E_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell E
C_SACCHC85_E_3	LOGICCH	"C_SACCHC85_E_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell E
C_SACCHC86_E_3	LOGICCH	"C_SACCHC86_E_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell E
C_SACCHC87_E_3	LOGICCH	"C_SACCHC87_E_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell E
C_SACCHC80_F_1	LOGICCH	"C_SACCHC80_F_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell F
C_SACCHC81_F_1	LOGICCH	"C_SACCHC81_F_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell F
C_SACCHC82_F_1	LOGICCH	"C_SACCHC82_F_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell F
C_SACCHC83_F_1	LOGICCH	"C_SACCHC83_F_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell F
C_SACCHC84_F_1	LOGICCH	"C_SACCHC84_F_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell F
C_SACCHC85_F_1	LOGICCH	"C_SACCHC85_F_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell F
C_SACCHC86_F_1	LOGICCH	"C_SACCHC86_F_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell F
C_SACCHC87_F_1	LOGICCH	"C_SACCHC87_F_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell F

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC80_F_2	LOGICCH	"C_SACCHC80_F_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell F
C_SACCHC81_F_2	LOGICCH	"C_SACCHC81_F_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell F
C_SACCHC82_F_2	LOGICCH	"C_SACCHC82_F_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell F
C_SACCHC83_F_2	LOGICCH	"C_SACCHC83_F_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell F
C_SACCHC84_F_2	LOGICCH	"C_SACCHC84_F_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell F
C_SACCHC85_F_2	LOGICCH	"C_SACCHC85_F_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell F
C_SACCHC86_F_2	LOGICCH	"C_SACCHC86_F_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell F
C_SACCHC87_F_2	LOGICCH	"C_SACCHC87_F_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell F
C_SACCHC80_F_3	LOGICCH	"C_SACCHC80_F_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell F
C_SACCHC81_F_3	LOGICCH	"C_SACCHC81_F_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell F
C_SACCHC82_F_3	LOGICCH	"C_SACCHC82_F_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell F
C_SACCHC83_F_3	LOGICCH	"C_SACCHC83_F_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell F
C_SACCHC84_F_3	LOGICCH	"C_SACCHC84_F_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell F
C_SACCHC85_F_3	LOGICCH	"C_SACCHC85_F_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell F
C_SACCHC86_F_3	LOGICCH	"C_SACCHC86_F_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell F
C_SACCHC87_F_3	LOGICCH	"C_SACCHC87_F_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell F
C_SACCHC80_G_1	LOGICCH	"C_SACCHC80_G_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell G
C_SACCHC81_G_1	LOGICCH	"C_SACCHC81_G_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell G

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC82_G_1	LOGICCH	"C_SACCHC82_G_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell G
C_SACCHC83_G_1	LOGICCH	"C_SACCHC83_G_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell G
C_SACCHC84_G_1	LOGICCH	"C_SACCHC84_G_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell G
C_SACCHC85_G_1	LOGICCH	"C_SACCHC85_G_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell G
C_SACCHC86_G_1	LOGICCH	"C_SACCHC86_G_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell G
C_SACCHC87_G_1	LOGICCH	"C_SACCHC87_G_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell G
C_SACCHC80_G_2	LOGICCH	"C_SACCHC80_G_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell G
C_SACCHC81_G_2	LOGICCH	"C_SACCHC81_G_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell G
C_SACCHC82_G_2	LOGICCH	"C_SACCHC82_G_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell G
C_SACCHC83_G_2	LOGICCH	"C_SACCHC83_G_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell G
C_SACCHC84_G_2	LOGICCH	"C_SACCHC84_G_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell G
C_SACCHC85_G_2	LOGICCH	"C_SACCHC85_G_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell G
C_SACCHC86_G_2	LOGICCH	"C_SACCHC86_G_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell G
C_SACCHC87_G_2	LOGICCH	"C_SACCHC87_G_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell G
C_SACCHC80_G_3	LOGICCH	"C_SACCHC80_G_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell G
C_SACCHC81_G_3	LOGICCH	"C_SACCHC81_G_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell G
C_SACCHC82_G_3	LOGICCH	"C_SACCHC82_G_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell G
C_SACCHC83_G_3	LOGICCH	"C_SACCHC83_G_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell G

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC84_G_3	LOGICCH	"C_SACCHC84_G_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell G
C_SACCHC85_G_3	LOGICCH	"C_SACCHC85_G_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell G
C_SACCHC86_G_3	LOGICCH	"C_SACCHC86_G_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell G
C_SACCHC87_G_3	LOGICCH	"C_SACCHC87_G_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell G
C_SACCHC80_H_1	LOGICCH	"C_SACCHC80_H_1"	SACCH/C8(0) associated with 1st SDCCH/8(0) of cell H
C_SACCHC81_H_1	LOGICCH	"C_SACCHC81_H_1"	SACCH/C8(1) associated with 1st SDCCH/8(1) of cell H
C_SACCHC82_H_1	LOGICCH	"C_SACCHC82_H_1"	SACCH/C8(2) associated with 1st SDCCH/8(2) of cell H
C_SACCHC83_H_1	LOGICCH	"C_SACCHC83_H_1"	SACCH/C8(3) associated with 1st SDCCH/8(3) of cell H
C_SACCHC84_H_1	LOGICCH	"C_SACCHC84_H_1"	SACCH/C8(4) associated with 1st SDCCH/8(4) of cell H
C_SACCHC85_H_1	LOGICCH	"C_SACCHC85_H_1"	SACCH/C8(5) associated with 1st SDCCH/8(5) of cell H
C_SACCHC86_H_1	LOGICCH	"C_SACCHC86_H_1"	SACCH/C8(6) associated with 1st SDCCH/8(6) of cell H
C_SACCHC87_H_1	LOGICCH	"C_SACCHC87_H_1"	SACCH/C8(7) associated with 1st SDCCH/8(7) of cell H
C_SACCHC80_H_2	LOGICCH	"C_SACCHC80_H_2"	SACCH/C8(0) associated with 2nd SDCCH/8(0) of cell H
C_SACCHC81_H_2	LOGICCH	"C_SACCHC81_H_2"	SACCH/C8(1) associated with 2nd SDCCH/8(1) of cell H
C_SACCHC82_H_2	LOGICCH	"C_SACCHC82_H_2"	SACCH/C8(2) associated with 2nd SDCCH/8(2) of cell H
C_SACCHC83_H_2	LOGICCH	"C_SACCHC83_H_2"	SACCH/C8(3) associated with 2nd SDCCH/8(3) of cell H
C_SACCHC84_H_2	LOGICCH	"C_SACCHC84_H_2"	SACCH/C8(4) associated with 2nd SDCCH/8(4) of cell H
C_SACCHC85_H_2	LOGICCH	"C_SACCHC85_H_2"	SACCH/C8(5) associated with 2nd SDCCH/8(5) of cell H

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SACCHC86_H_2	LOGICCH	"C_SACCHC86_H_2"	SACCH/C8(6) associated with 2nd SDCCH/8(6) of cell H
C_SACCHC87_H_2	LOGICCH	"C_SACCHC87_H_2"	SACCH/C8(7) associated with 2nd SDCCH/8(7) of cell H
C_SACCHC80_H_3	LOGICCH	"C_SACCHC80_H_3"	SACCH/C8(0) associated with 3rd SDCCH/8(0) of cell H
C_SACCHC81_H_3	LOGICCH	"C_SACCHC81_H_3"	SACCH/C8(1) associated with 3rd SDCCH/8(1) of cell H
C_SACCHC82_H_3	LOGICCH	"C_SACCHC82_H_3"	SACCH/C8(2) associated with 3rd SDCCH/8(2) of cell H
C_SACCHC83_H_3	LOGICCH	"C_SACCHC83_H_3"	SACCH/C8(3) associated with 3rd SDCCH/8(3) of cell H
C_SACCHC84_H_3	LOGICCH	"C_SACCHC84_H_3"	SACCH/C8(4) associated with 3rd SDCCH/8(4) of cell H
C_SACCHC85_H_3	LOGICCH	"C_SACCHC85_H_3"	SACCH/C8(5) associated with 3rd SDCCH/8(5) of cell H
C_SACCHC86_H_3	LOGICCH	"C_SACCHC86_H_3"	SACCH/C8(6) associated with 3rd SDCCH/8(6) of cell H
C_SACCHC87_H_3	LOGICCH	"C_SACCHC87_H_3"	SACCH/C8(7) associated with 3rd SDCCH/8(7) of cell H
C_SDCCH4_A	LOGICCH	"C_SDCCH4_A"	SDCCH/4 channel of cell A
C_SDCCH4_B	LOGICCH	"C_SDCCH4_B"	SDCCH/4 channel of cell B
C_SDCCH4_C	LOGICCH	"C_SDCCH4_C"	SDCCH/4 channel of cell C
C_SDCCH4_D	LOGICCH	"C_SDCCH4_D"	SDCCH/4 channel of cell D
C_SDCCH4_E	LOGICCH	"C_SDCCH4_E"	SDCCH/4 channel of cell E
C_SDCCH4_F	LOGICCH	"C_SDCCH4_F"	SDCCH/4 channel of cell F
C_SDCCH4_G	LOGICCH	"C_SDCCH4_G"	SDCCH/4 channel of cell G
C_SDCCH4_H	LOGICCH	"C_SDCCH4_H"	SDCCH/4 channel of cell H
C_SDCCH40_A	LOGICCH	"C_SDCCH40_A"	SDCCH/4(0) of cell A
C_SDCCH41_A	LOGICCH	"C_SDCCH41_A"	SDCCH/4(1) of cell A
C_SDCCH42_A	LOGICCH	"C_SDCCH42_A"	SDCCH/4(2) of cell A
C_SDCCH43_A	LOGICCH	"C_SDCCH43_A"	SDCCH/4(3) of cell A
C_SDCCH40_B	LOGICCH	"C_SDCCH40_B"	SDCCH/4(0) of cell B
C_SDCCH41_B	LOGICCH	"C_SDCCH41_B"	SDCCH/4(1) of cell B
C_SDCCH42_B	LOGICCH	"C_SDCCH42_B"	SDCCH/4(2) of cell B
C_SDCCH43_B	LOGICCH	"C_SDCCH43_B"	SDCCH/4(3) of cell B
C_SDCCH40_C	LOGICCH	"C_SDCCH40_C"	SDCCH/4(0) of cell C
C_SDCCH41_C	LOGICCH	"C_SDCCH41_C"	SDCCH/4(1) of cell C
C_SDCCH42_C	LOGICCH	"C_SDCCH42_C"	SDCCH/4(2) of cell C
C_SDCCH43_C	LOGICCH	"C_SDCCH43_C"	SDCCH/4(3) of cell C
C_SDCCH40_D	LOGICCH	"C_SDCCH40_D"	SDCCH/4(0) of cell D
C_SDCCH41_D	LOGICCH	"C_SDCCH41_D"	SDCCH/4(1) of cell D

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SDCCH42_D	LOGICCH	"C_SDCCH42_D"	SDCCH/4(2) of cell D
C_SDCCH43_D	LOGICCH	"C_SDCCH43_D"	SDCCH/4(3) of cell D
C_SDCCH40_E	LOGICCH	"C_SDCCH40_E"	SDCCH/4(0) of cell E
C_SDCCH41_E	LOGICCH	"C_SDCCH41_E"	SDCCH/4(1) of cell E
C_SDCCH42_E	LOGICCH	"C_SDCCH42_E"	SDCCH/4(2) of cell E
C_SDCCH43_E	LOGICCH	"C_SDCCH43_E"	SDCCH/4(3) of cell E
C_SDCCH40_F	LOGICCH	"C_SDCCH40_F"	SDCCH/4(0) of cell F
C_SDCCH41_F	LOGICCH	"C_SDCCH41_F"	SDCCH/4(1) of cell F
C_SDCCH42_F	LOGICCH	"C_SDCCH42_F"	SDCCH/4(2) of cell F
C_SDCCH43_F	LOGICCH	"C_SDCCH43_F"	SDCCH/4(3) of cell F
C_SDCCH40_G	LOGICCH	"C_SDCCH40_G"	SDCCH/4(0) of cell G
C_SDCCH41_G	LOGICCH	"C_SDCCH41_G"	SDCCH/4(1) of cell G
C_SDCCH42_G	LOGICCH	"C_SDCCH42_G"	SDCCH/4(2) of cell G
C_SDCCH43_G	LOGICCH	"C_SDCCH43_G"	SDCCH/4(3) of cell G
C_SDCCH40_H	LOGICCH	"C_SDCCH40_H"	SDCCH/4(0) of cell H
C_SDCCH41_H	LOGICCH	"C_SDCCH41_H"	SDCCH/4(1) of cell H
C_SDCCH42_H	LOGICCH	"C_SDCCH42_H"	SDCCH/4(2) of cell H
C_SDCCH43_H	LOGICCH	"C_SDCCH43_H"	SDCCH/4(3) of cell H
C_SDCCH8_A_1	LOGICCH	"C_SDCCH8_A_1"	1st SDCCH/8 channel of cell A
C_SDCCH8_A_2	LOGICCH	"C_SDCCH8_A_2"	2nd SDCCH/8 channel of cell A
C_SDCCH8_A_3	LOGICCH	"C_SDCCH8_A_3"	3rd SDCCH/8 channel of cell A
C_SDCCH8_B_1	LOGICCH	"C_SDCCH8_B_1"	1st SDCCH/8 channel of cell B
C_SDCCH8_B_2	LOGICCH	"C_SDCCH8_B_2"	2nd SDCCH/8 channel of cell B
C_SDCCH80_A_1	LOGICCH	"C_SDCCH80_A_1"	1st SDCCH/8(0) of cell A
C_SDCCH81_A_1	LOGICCH	"C_SDCCH81_A_1"	1st SDCCH/8(1) of cell A
C_SDCCH82_A_1	LOGICCH	"C_SDCCH82_A_1"	1st SDCCH/8(2) of cell A
C_SDCCH83_A_1	LOGICCH	"C_SDCCH83_A_1"	1st SDCCH/8(3) of cell A
C_SDCCH84_A_1	LOGICCH	"C_SDCCH84_A_1"	1st SDCCH/8(4) of cell A
C_SDCCH85_A_1	LOGICCH	"C_SDCCH85_A_1"	1st SDCCH/8(5) of cell A
C_SDCCH86_A_1	LOGICCH	"C_SDCCH86_A_1"	1st SDCCH/8(6) of cell A
C_SDCCH87_A_1	LOGICCH	"C_SDCCH87_A_1"	1st SDCCH/8(7) of cell A
C_SDCCH80_A_2	LOGICCH	"C_SDCCH80_A_2"	2nd SDCCH/8(0) of cell A
C_SDCCH81_A_2	LOGICCH	"C_SDCCH81_A_2"	2nd SDCCH/8(1) of cell A
C_SDCCH82_A_2	LOGICCH	"C_SDCCH82_A_2"	2nd SDCCH/8(2) of cell A
C_SDCCH83_A_2	LOGICCH	"C_SDCCH83_A_2"	2nd SDCCH/8(3) of cell A
C_SDCCH84_A_2	LOGICCH	"C_SDCCH84_A_2"	2nd SDCCH/8(4) of cell A
C_SDCCH85_A_2	LOGICCH	"C_SDCCH85_A_2"	2nd SDCCH/8(5) of cell A
C_SDCCH86_A_2	LOGICCH	"C_SDCCH86_A_2"	2nd SDCCH/8(6) of cell A
C_SDCCH87_A_2	LOGICCH	"C_SDCCH87_A_2"	2nd SDCCH/8(7) of cell A
C_SDCCH80_A_3	LOGICCH	"C_SDCCH80_A_3"	3rd SDCCH/8(0) of cell A
C_SDCCH81_A_3	LOGICCH	"C_SDCCH81_A_3"	3rd SDCCH/8(1) of cell A
C_SDCCH82_A_3	LOGICCH	"C_SDCCH82_A_3"	3rd SDCCH/8(2) of cell A
C_SDCCH83_A_3	LOGICCH	"C_SDCCH83_A_3"	3rd SDCCH/8(3) of cell A

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SDCCH84_A_3	LOGICCH	"C_SDCCH84_A_3"	3rd SDCCH/8(4) of cell A
C_SDCCH85_A_3	LOGICCH	"C_SDCCH85_A_3"	3rd SDCCH/8(5) of cell A
C_SDCCH86_A_3	LOGICCH	"C_SDCCH86_A_3"	3rd SDCCH/8(6) of cell A
C_SDCCH87_A_3	LOGICCH	"C_SDCCH87_A_3"	3rd SDCCH/8(7) of cell A
C_SDCCH80_B_1	LOGICCH	"C_SDCCH80_B_1"	1st SDCCH/8(0) of cell B
C_SDCCH81_B_1	LOGICCH	"C_SDCCH81_B_1"	1st SDCCH/8(1) of cell B
C_SDCCH82_B_1	LOGICCH	"C_SDCCH82_B_1"	1st SDCCH/8(2) of cell B
C_SDCCH83_B_1	LOGICCH	"C_SDCCH83_B_1"	1st SDCCH/8(3) of cell B
C_SDCCH84_B_1	LOGICCH	"C_SDCCH84_B_1"	1st SDCCH/8(4) of cell B
C_SDCCH85_B_1	LOGICCH	"C_SDCCH85_B_1"	1st SDCCH/8(5) of cell B
C_SDCCH86_B_1	LOGICCH	"C_SDCCH86_B_1"	1st SDCCH/8(6) of cell B
C_SDCCH87_B_1	LOGICCH	"C_SDCCH87_B_1"	1st SDCCH/8(7) of cell B
C_SDCCH80_B_2	LOGICCH	"C_SDCCH80_B_2"	2nd SDCCH/8(0) of cell B
C_SDCCH81_B_2	LOGICCH	"C_SDCCH81_B_2"	2nd SDCCH/8(1) of cell B
C_SDCCH82_B_2	LOGICCH	"C_SDCCH82_B_2"	2nd SDCCH/8(2) of cell B
C_SDCCH83_B_2	LOGICCH	"C_SDCCH83_B_2"	2nd SDCCH/8(3) of cell B
C_SDCCH84_B_2	LOGICCH	"C_SDCCH84_B_2"	2nd SDCCH/8(4) of cell B
C_SDCCH85_B_2	LOGICCH	"C_SDCCH85_B_2"	2nd SDCCH/8(5) of cell B
C_SDCCH86_B_2	LOGICCH	"C_SDCCH86_B_2"	2nd SDCCH/8(6) of cell B
C_SDCCH87_B_2	LOGICCH	"C_SDCCH87_B_2"	2nd SDCCH/8(7) of cell B
C_SDCCH80_B_3	LOGICCH	"C_SDCCH80_B_3"	3rd SDCCH/8(0) of cell B
C_SDCCH81_B_3	LOGICCH	"C_SDCCH81_B_3"	3rd SDCCH/8(1) of cell B
C_SDCCH82_B_3	LOGICCH	"C_SDCCH82_B_3"	3rd SDCCH/8(2) of cell B
C_SDCCH83_B_3	LOGICCH	"C_SDCCH83_B_3"	3rd SDCCH/8(3) of cell B
C_SDCCH84_B_3	LOGICCH	"C_SDCCH84_B_3"	3rd SDCCH/8(4) of cell B
C_SDCCH85_B_3	LOGICCH	"C_SDCCH85_B_3"	3rd SDCCH/8(5) of cell B
C_SDCCH86_B_3	LOGICCH	"C_SDCCH86_B_3"	3rd SDCCH/8(6) of cell B
C_SDCCH87_B_3	LOGICCH	"C_SDCCH87_B_3"	3rd SDCCH/8(7) of cell B
C_SDCCH80_C_1	LOGICCH	"C_SDCCH80_C_1"	1st SDCCH/8(0) of cell C
C_SDCCH81_C_1	LOGICCH	"C_SDCCH81_C_1"	1st SDCCH/8(1) of cell C
C_SDCCH82_C_1	LOGICCH	"C_SDCCH82_C_1"	1st SDCCH/8(2) of cell C
C_SDCCH83_C_1	LOGICCH	"C_SDCCH83_C_1"	1st SDCCH/8(3) of cell C
C_SDCCH84_C_1	LOGICCH	"C_SDCCH84_C_1"	1st SDCCH/8(4) of cell C
C_SDCCH85_C_1	LOGICCH	"C_SDCCH85_C_1"	1st SDCCH/8(5) of cell C
C_SDCCH86_C_1	LOGICCH	"C_SDCCH86_C_1"	1st SDCCH/8(6) of cell C
C_SDCCH87_C_1	LOGICCH	"C_SDCCH87_C_1"	1st SDCCH/8(7) of cell C
C_SDCCH80_C_2	LOGICCH	"C_SDCCH80_C_2"	2nd SDCCH/8(0) of cell C
C_SDCCH81_C_2	LOGICCH	"C_SDCCH81_C_2"	2nd SDCCH/8(1) of cell C
C_SDCCH82_C_2	LOGICCH	"C_SDCCH82_C_2"	2nd SDCCH/8(2) of cell C
C_SDCCH83_C_2	LOGICCH	"C_SDCCH83_C_2"	2nd SDCCH/8(3) of cell C
C_SDCCH84_C_2	LOGICCH	"C_SDCCH84_C_2"	2nd SDCCH/8(4) of cell C
C_SDCCH85_C_2	LOGICCH	"C_SDCCH85_C_2"	2nd SDCCH/8(5) of cell C
C_SDCCH86_C_2	LOGICCH	"C_SDCCH86_C_2"	2nd SDCCH/8(6) of cell C
C_SDCCH87_C_2	LOGICCH	"C_SDCCH87_C_2"	2nd SDCCH/8(7) of cell C
C_SDCCH80_C_3	LOGICCH	"C_SDCCH80_C_3"	3rd SDCCH/8(0) of cell C
C_SDCCH81_C_3	LOGICCH	"C_SDCCH81_C_3"	3rd SDCCH/8(1) of cell C
C_SDCCH82_C_3	LOGICCH	"C_SDCCH82_C_3"	3rd SDCCH/8(2) of cell C

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SDCCH83_C_3	LOGICCH	"C_SDCCH83_C_3"	3rd SDCCH/8(3) of cell C
C_SDCCH84_C_3	LOGICCH	"C_SDCCH84_C_3"	3rd SDCCH/8(4) of cell C
C_SDCCH85_C_3	LOGICCH	"C_SDCCH85_C_3"	3rd SDCCH/8(5) of cell C
C_SDCCH86_C_3	LOGICCH	"C_SDCCH86_C_3"	3rd SDCCH/8(6) of cell C
C_SDCCH87_C_3	LOGICCH	"C_SDCCH87_C_3"	3rd SDCCH/8(7) of cell C
C_SDCCH80_D_1	LOGICCH	"C_SDCCH80_D_1"	1st SDCCH/8(0) of cell D
C_SDCCH81_D_1	LOGICCH	"C_SDCCH81_D_1"	1st SDCCH/8(1) of cell D
C_SDCCH82_D_1	LOGICCH	"C_SDCCH82_D_1"	1st SDCCH/8(2) of cell D
C_SDCCH83_D_1	LOGICCH	"C_SDCCH83_D_1"	1st SDCCH/8(3) of cell D
C_SDCCH84_D_1	LOGICCH	"C_SDCCH84_D_1"	1st SDCCH/8(4) of cell D
C_SDCCH85_D_1	LOGICCH	"C_SDCCH85_D_1"	1st SDCCH/8(5) of cell D
C_SDCCH86_D_1	LOGICCH	"C_SDCCH86_D_1"	1st SDCCH/8(6) of cell D
C_SDCCH87_D_1	LOGICCH	"C_SDCCH87_D_1"	1st SDCCH/8(7) of cell D
C_SDCCH80_D_2	LOGICCH	"C_SDCCH80_D_2"	2nd SDCCH/8(0) of cell D
C_SDCCH81_D_2	LOGICCH	"C_SDCCH81_D_2"	2nd SDCCH/8(1) of cell D
C_SDCCH82_D_2	LOGICCH	"C_SDCCH82_D_2"	2nd SDCCH/8(2) of cell D
C_SDCCH83_D_2	LOGICCH	"C_SDCCH83_D_2"	2nd SDCCH/8(3) of cell D
C_SDCCH84_D_2	LOGICCH	"C_SDCCH84_D_2"	2nd SDCCH/8(4) of cell D
C_SDCCH85_D_2	LOGICCH	"C_SDCCH85_D_2"	2nd SDCCH/8(5) of cell D
C_SDCCH86_D_2	LOGICCH	"C_SDCCH86_D_2"	2nd SDCCH/8(6) of cell D
C_SDCCH87_D_2	LOGICCH	"C_SDCCH87_D_2"	2nd SDCCH/8(7) of cell D
C_SDCCH80_D_3	LOGICCH	"C_SDCCH80_D_3"	3rd SDCCH/8(0) of cell D
C_SDCCH81_D_3	LOGICCH	"C_SDCCH81_D_3"	3rd SDCCH/8(1) of cell D
C_SDCCH82_D_3	LOGICCH	"C_SDCCH82_D_3"	3rd SDCCH/8(2) of cell D
C_SDCCH83_D_3	LOGICCH	"C_SDCCH83_D_3"	3rd SDCCH/8(3) of cell D
C_SDCCH84_D_3	LOGICCH	"C_SDCCH84_D_3"	3rd SDCCH/8(4) of cell D
C_SDCCH85_D_3	LOGICCH	"C_SDCCH85_D_3"	3rd SDCCH/8(5) of cell D
C_SDCCH86_D_3	LOGICCH	"C_SDCCH86_D_3"	3rd SDCCH/8(6) of cell D
C_SDCCH87_D_3	LOGICCH	"C_SDCCH87_D_3"	3rd SDCCH/8(7) of cell D
C_SDCCH80_E_1	LOGICCH	"C_SDCCH80_E_1"	1st SDCCH/8(0) of cell E
C_SDCCH81_E_1	LOGICCH	"C_SDCCH81_E_1"	1st SDCCH/8(1) of cell E
C_SDCCH82_E_1	LOGICCH	"C_SDCCH82_E_1"	1st SDCCH/8(2) of cell E
C_SDCCH83_E_1	LOGICCH	"C_SDCCH83_E_1"	1st SDCCH/8(3) of cell E
C_SDCCH84_E_1	LOGICCH	"C_SDCCH84_E_1"	1st SDCCH/8(4) of cell E
C_SDCCH85_E_1	LOGICCH	"C_SDCCH85_E_1"	1st SDCCH/8(5) of cell E
C_SDCCH86_E_1	LOGICCH	"C_SDCCH86_E_1"	1st SDCCH/8(6) of cell E
C_SDCCH87_E_1	LOGICCH	"C_SDCCH87_E_1"	1st SDCCH/8(7) of cell E
C_SDCCH80_E_2	LOGICCH	"C_SDCCH80_E_2"	2nd SDCCH/8(0) of cell E
C_SDCCH81_E_2	LOGICCH	"C_SDCCH81_E_2"	2nd SDCCH/8(1) of cell E
C_SDCCH82_E_2	LOGICCH	"C_SDCCH82_E_2"	2nd SDCCH/8(2) of cell E
C_SDCCH83_E_2	LOGICCH	"C_SDCCH83_E_2"	2nd SDCCH/8(3) of cell E
C_SDCCH84_E_2	LOGICCH	"C_SDCCH84_E_2"	2nd SDCCH/8(4) of cell E
C_SDCCH85_E_2	LOGICCH	"C_SDCCH85_E_2"	2nd SDCCH/8(5) of cell E
C_SDCCH86_E_2	LOGICCH	"C_SDCCH86_E_2"	2nd SDCCH/8(6) of cell E
C_SDCCH87_E_2	LOGICCH	"C_SDCCH87_E_2"	2nd SDCCH/8(7) of cell E
C_SDCCH80_E_3	LOGICCH	"C_SDCCH80_E_3"	3rd SDCCH/8(0) of cell E
C_SDCCH81_E_3	LOGICCH	"C_SDCCH81_E_3"	3rd SDCCH/8(1) of cell E

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SDCCH82_E_3	LOGICCH	"C_SDCCH82_E_3"	3rd SDCCH/8(2) of cell E
C_SDCCH83_E_3	LOGICCH	"C_SDCCH83_E_3"	3rd SDCCH/8(3) of cell E
C_SDCCH84_E_3	LOGICCH	"C_SDCCH84_E_3"	3rd SDCCH/8(4) of cell E
C_SDCCH85_E_3	LOGICCH	"C_SDCCH85_E_3"	3rd SDCCH/8(5) of cell E
C_SDCCH86_E_3	LOGICCH	"C_SDCCH86_E_3"	3rd SDCCH/8(6) of cell E
C_SDCCH87_E_3	LOGICCH	"C_SDCCH87_E_3"	3rd SDCCH/8(7) of cell E
C_SDCCH80_F_1	LOGICCH	"C_SDCCH80_F_1"	1st SDCCH/8(0) of cell F
C_SDCCH81_F_1	LOGICCH	"C_SDCCH81_F_1"	1st SDCCH/8(1) of cell F
C_SDCCH82_F_1	LOGICCH	"C_SDCCH82_F_1"	1st SDCCH/8(2) of cell F
C_SDCCH83_F_1	LOGICCH	"C_SDCCH83_F_1"	1st SDCCH/8(3) of cell F
C_SDCCH84_F_1	LOGICCH	"C_SDCCH84_F_1"	1st SDCCH/8(4) of cell F
C_SDCCH85_F_1	LOGICCH	"C_SDCCH85_F_1"	1st SDCCH/8(5) of cell F
C_SDCCH86_F_1	LOGICCH	"C_SDCCH86_F_1"	1st SDCCH/8(6) of cell F
C_SDCCH87_F_1	LOGICCH	"C_SDCCH87_F_1"	1st SDCCH/8(7) of cell F
C_SDCCH80_F_2	LOGICCH	"C_SDCCH80_F_2"	2nd SDCCH/8(0) of cell F
C_SDCCH81_F_2	LOGICCH	"C_SDCCH81_F_2"	2nd SDCCH/8(1) of cell F
C_SDCCH82_F_2	LOGICCH	"C_SDCCH82_F_2"	2nd SDCCH/8(2) of cell F
C_SDCCH83_F_2	LOGICCH	"C_SDCCH83_F_2"	2nd SDCCH/8(3) of cell F
C_SDCCH84_F_2	LOGICCH	"C_SDCCH84_F_2"	2nd SDCCH/8(4) of cell F
C_SDCCH85_F_2	LOGICCH	"C_SDCCH85_F_2"	2nd SDCCH/8(5) of cell F
C_SDCCH86_F_2	LOGICCH	"C_SDCCH86_F_2"	2nd SDCCH/8(6) of cell F
C_SDCCH87_F_2	LOGICCH	"C_SDCCH87_F_2"	2nd SDCCH/8(7) of cell F
C_SDCCH80_F_3	LOGICCH	"C_SDCCH80_F_3"	3rd SDCCH/8(0) of cell F
C_SDCCH81_F_3	LOGICCH	"C_SDCCH81_F_3"	3rd SDCCH/8(1) of cell F
C_SDCCH82_F_3	LOGICCH	"C_SDCCH82_F_3"	3rd SDCCH/8(2) of cell F
C_SDCCH83_F_3	LOGICCH	"C_SDCCH83_F_3"	3rd SDCCH/8(3) of cell F
C_SDCCH84_F_3	LOGICCH	"C_SDCCH84_F_3"	3rd SDCCH/8(4) of cell F
C_SDCCH85_F_3	LOGICCH	"C_SDCCH85_F_3"	3rd SDCCH/8(5) of cell F
C_SDCCH86_F_3	LOGICCH	"C_SDCCH86_F_3"	3rd SDCCH/8(6) of cell F
C_SDCCH87_F_3	LOGICCH	"C_SDCCH87_F_3"	3rd SDCCH/8(7) of cell F
C_SDCCH80_G_1	LOGICCH	"C_SDCCH80_G_1"	1st SDCCH/8(0) of cell G
C_SDCCH81_G_1	LOGICCH	"C_SDCCH81_G_1"	1st SDCCH/8(1) of cell G
C_SDCCH82_G_1	LOGICCH	"C_SDCCH82_G_1"	1st SDCCH/8(2) of cell G
C_SDCCH83_G_1	LOGICCH	"C_SDCCH83_G_1"	1st SDCCH/8(3) of cell G
C_SDCCH84_G_1	LOGICCH	"C_SDCCH84_G_1"	1st SDCCH/8(4) of cell G
C_SDCCH85_G_1	LOGICCH	"C_SDCCH85_G_1"	1st SDCCH/8(5) of cell G
C_SDCCH86_G_1	LOGICCH	"C_SDCCH86_G_1"	1st SDCCH/8(6) of cell G
C_SDCCH87_G_1	LOGICCH	"C_SDCCH87_G_1"	1st SDCCH/8(7) of cell G
C_SDCCH80_G_2	LOGICCH	"C_SDCCH80_G_2"	2nd SDCCH/8(0) of cell G
C_SDCCH81_G_2	LOGICCH	"C_SDCCH81_G_2"	2nd SDCCH/8(1) of cell G
C_SDCCH82_G_2	LOGICCH	"C_SDCCH82_G_2"	2nd SDCCH/8(2) of cell G
C_SDCCH83_G_2	LOGICCH	"C_SDCCH83_G_2"	2nd SDCCH/8(3) of cell G
C_SDCCH84_G_2	LOGICCH	"C_SDCCH84_G_2"	2nd SDCCH/8(4) of cell G
C_SDCCH85_G_2	LOGICCH	"C_SDCCH85_G_2"	2nd SDCCH/8(5) of cell G
C_SDCCH86_G_2	LOGICCH	"C_SDCCH86_G_2"	2nd SDCCH/8(6) of cell G
C_SDCCH87_G_2	LOGICCH	"C_SDCCH87_G_2"	2nd SDCCH/8(7) of cell G
C_SDCCH80_G_3	LOGICCH	"C_SDCCH80_G_3"	3rd SDCCH/8(0) of cell G

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SDCCH81_G_3	LOGICCH	"C_SDCCH81_G_3"	3rd SDCCH/8(1) of cell G
C_SDCCH82_G_3	LOGICCH	"C_SDCCH82_G_3"	3rd SDCCH/8(2) of cell G
C_SDCCH83_G_3	LOGICCH	"C_SDCCH83_G_3"	3rd SDCCH/8(3) of cell G
C_SDCCH84_G_3	LOGICCH	"C_SDCCH84_G_3"	3rd SDCCH/8(4) of cell G
C_SDCCH85_G_3	LOGICCH	"C_SDCCH85_G_3"	3rd SDCCH/8(5) of cell G
C_SDCCH86_G_3	LOGICCH	"C_SDCCH86_G_3"	3rd SDCCH/8(6) of cell G
C_SDCCH87_G_3	LOGICCH	"C_SDCCH87_G_3"	3rd SDCCH/8(7) of cell G
C_SDCCH80_H_1	LOGICCH	"C_SDCCH80_H_1"	1st SDCCH/8(0) of cell H
C_SDCCH81_H_1	LOGICCH	"C_SDCCH81_H_1"	1st SDCCH/8(1) of cell H
C_SDCCH82_H_1	LOGICCH	"C_SDCCH82_H_1"	1st SDCCH/8(2) of cell H
C_SDCCH83_H_1	LOGICCH	"C_SDCCH83_H_1"	1st SDCCH/8(3) of cell H
C_SDCCH84_H_1	LOGICCH	"C_SDCCH84_H_1"	1st SDCCH/8(4) of cell H
C_SDCCH85_H_1	LOGICCH	"C_SDCCH85_H_1"	1st SDCCH/8(5) of cell H
C_SDCCH86_H_1	LOGICCH	"C_SDCCH86_H_1"	1st SDCCH/8(6) of cell H
C_SDCCH87_H_1	LOGICCH	"C_SDCCH87_H_1"	1st SDCCH/8(7) of cell H
C_SDCCH80_H_2	LOGICCH	"C_SDCCH80_H_2"	2nd SDCCH/8(0) of cell H
C_SDCCH81_H_2	LOGICCH	"C_SDCCH81_H_2"	2nd SDCCH/8(1) of cell H
C_SDCCH82_H_2	LOGICCH	"C_SDCCH82_H_2"	2nd SDCCH/8(2) of cell H
C_SDCCH83_H_2	LOGICCH	"C_SDCCH83_H_2"	2nd SDCCH/8(3) of cell H
C_SDCCH84_H_2	LOGICCH	"C_SDCCH84_H_2"	2nd SDCCH/8(4) of cell H
C_SDCCH85_H_2	LOGICCH	"C_SDCCH85_H_2"	2nd SDCCH/8(5) of cell H
C_SDCCH86_H_2	LOGICCH	"C_SDCCH86_H_2"	2nd SDCCH/8(6) of cell H
C_SDCCH87_H_2	LOGICCH	"C_SDCCH87_H_2"	2nd SDCCH/8(7) of cell H
C_SDCCH80_H_3	LOGICCH	"C_SDCCH80_H_3"	3rd SDCCH/8(0) of cell H
C_SDCCH81_H_3	LOGICCH	"C_SDCCH81_H_3"	3rd SDCCH/8(1) of cell H
C_SDCCH82_H_3	LOGICCH	"C_SDCCH82_H_3"	3rd SDCCH/8(2) of cell H
C_SDCCH83_H_3	LOGICCH	"C_SDCCH83_H_3"	3rd SDCCH/8(3) of cell H
C_SDCCH84_H_3	LOGICCH	"C_SDCCH84_H_3"	3rd SDCCH/8(4) of cell H
C_SDCCH85_H_3	LOGICCH	"C_SDCCH85_H_3"	3rd SDCCH/8(5) of cell H
C_SDCCH86_H_3	LOGICCH	"C_SDCCH86_H_3"	3rd SDCCH/8(6) of cell H
C_SDCCH87_H_3	LOGICCH	"C_SDCCH87_H_3"	3rd SDCCH/8(7) of cell H
C_CHSDCCH8_FH	OCTETSTRING	'00'O	ch.type SDCCH8 with FH
C_CHSDCCH8_NonFH	OCTETSTRING	'01'O	ch.type SDCCH8 No FH
C_CHSDCCH4_NonFH	OCTETSTRING	'03'O	ch.type SDCCH4 NoFH
C_CHTCHF_FH	OCTETSTRING	'04'O	ch.type TCHF FH
C_CHTCHF_NonFH	OCTETSTRING	'05'O	ch.type TCHF NonFH
C_CHTCHH_FH	OCTETSTRING	'06'O	ch.type TCHH FH
C_CHTCHH_NonFH	OCTETSTRING	'07'O	ch.type TCHH NonFH
C_TCHF_ACCHF_1	LOGCH	"C_TCHF_ACCHF_1"	first TCH/F + ACCHs type channel
C_TCHF_ACCHF_2	LOGCH	"C_TCHF_ACCHF_2"	second TCH/F + ACCHs type channel
C_TCHH_ACCHH_1	LOGCH	"C_TCHH_ACCHH_1"	first TCH/H + ACCHs type channel
C_TCHH_ACCHH_2	LOGCH	"C_TCHH_ACCHH_2"	second TCH/H + ACCHs type channel
C_FCCH_SCH_BCCH_CCCH	LOGCH	"C_FCCH_SCH_BCCH_CCCH"	FCCH_SCH_BCCH_CCCH type channel

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_BCCH_CCCH_2	LOGCH	"C_BCCH_CCCH_2"	second BCCH_CCCH type channel
C_BCCH_CCCH_3	LOGCH	"C_BCCH_CCCH_3"	third BCCH_CCCH type channel
C_BCCH_CCCH_4	LOGCH	"C_BCCH_CCCH_4"	fourth BCCH_CCCH type channel
C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4	LOGCH	"C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4"	combined CCCH type channel
C_CBCH_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4	LOGCH	"C_CBCH_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4"	combined CCCH type channel for cell broadcast SM
C_SDCCH8_SACCHC8_1	LOGCH	"C_SDCCH8_SACCHC8_1"	first SDCCH/8 type channel
C_SDCCH8_SACCHC8_2	LOGCH	"C_SDCCH8_SACCHC8_2"	second SDCCH/8 type channel
C_SDCCH8_SACCHC8_3	LOGCH	"C_SDCCH8_SACCHC8_3"	3rd SDCCH/8 type channel
C_S0	SN	’000’B	time slot 0
C_S2	SN	’010’B	time slot 2
C_S3	SN	’011’B	time slot 3
C_S4	SN	’100’B	time slot 4
C_S6	SN	’110’B	time slot 6
C_SAVE	INTEGER	0	the OC_SaveAndProc saves input values
C_PROC	INTEGER	1	the OC_SvaAndProc processes saved values
C_RETRV	INTEGER	1	
C_U1	CCSTATE	1	CC state U1
C_U3	CCSTATE	3	CC state U3
C_U4	CCSTATE	4	CC state U4
C_U6	CCSTATE	6	CC state U6
C_U7	CCSTATE	7	CC state U7
C_U8	CCSTATE	8	CC state U8
C_U9	CCSTATE	9	CC state U9
C_U10	CCSTATE	10	CC state U10
C_U11	CCSTATE	11	CC state U11
C_U12	CCSTATE	12	CC state U12
C_U19	CCSTATE	19	CC state U19
C_U26	CCSTATE	26	CC state U26
C_NxtButOne	SENDINGMODE	1	send the second message on the next but one paging subblock
C_FmrAGB	SENDINGMODE	2	send the second message on the former access grant block
C_BfReOcc	SENDINGMODE	3	send the second message before the MS’s original paging subchannel re−occurs, but later than the next paging block of that CCCH

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_NxtButOneNxt	SENDINGMODE	4	nothing is sent in the next but one paging sub block, then send the second message in the next paging subblock of the MS's paging subchannel
C_CMServiceTypeE	CMSVTYPE	'0010'B	CM Service Type for emergency call.
C_Telephony	IA5String	"C_Telephony"	telephony service (TS11)
C_EmgCall	IA5String	"C_EmgCall"	emergency call service (TS12)
C_AltSpchFax	IA5String	"C_AltSpchFax"	alternate speech and G3 fax service (TS61)
C_AutoFax	IA5String	"C_AutoFax"	automatic G3 fax service (TS62)
C_Async300	IA5String	"C_Async300"	data circuit duplex async. 300 bit/s service (BS21)
C_Async1200	IA5String	"C_Async1200"	data circuit duplex async. 1200 bit/s service (BS22)
C_Async120075	IA5String	"C_Async120075"	data circuit duplex async. 1200/75 bit/s service (BS23)
C_Async2400	IA5String	"C_Async2400"	data circuit duplex async. 2400 bit/s service (BS24)
C_Async4800	IA5String	"C_Async4800"	data circuit duplex async. 4800 bit/s service (BS25)
C_Async9600	IA5String	"C_Async9600"	data circuit duplex async 9600 bit/s service (BS26)
C_Sync1200	IA5String	"C_Sync1200"	data circuit duplex sync. 1200 bit/s service (BS31)
C_Sync2400	IA5String	"C_Sync2400"	data circuit duplex sync. 2400 bit/s service (BS32)
C_Sync4800	IA5String	"C_Sync4800"	data circuit duplex sync. 4800 bit/s service (BS33)
C_Sync9600	IA5String	"C_Sync9600"	data circuit duplex sync 9600 bit/s service (BS34)
C_PAD300	IA5String	"C_PAD300"	PAD access 300 bit/s service (BS41)
C_PAD1200	IA5String	"C_PAD1200"	PAD access 1200 bit/s service (BS42)
C_PAD120075	IA5String	"C_PAD120075"	PAD access 1200/75 bit/s service (BS43)
C_PAD2400	IA5String	"C_PAD2400"	PAD access 2400 bit/s service (BS44)
C_PAD4800	IA5String	"C_PAD4800"	PAD access 4800 bit/s service (BS45)
C_PAD9600	IA5String	"C_PAD9600"	PAD access 9600 bit/s service (BS46)
C_Packet2400	IA5String	"C_Packet2400"	packt access 2400 bit/s service (BS51)
C_Packet4800	IA5String	"C_Packet4800"	packt access 4800 bit/s service (BS52)

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_Packet9600	IA5String	"C_Packet9600"	packt access 9600 bit/s service (BS53)
C_AltSpchData	IA5String	"C_AltSpchData"	alternate speech/data service (BS61)
C_SpchData	IA5String	"C_SpchData"	speech followed data service (BS81)
C_I	INTEGER	1	I command of L 2
C_one	INTEGER	1	one frequency, no hopping
C_Immass	BITSTRING	’000’B	activation for immediate assignment
C_Ass	BITSTRING	’001’B	activation for assignment
C_Asynho	BITSTRING	’010’B	activation for non synchronous handover
C_Synho	BITSTRING	’011’B	activation for synchronous handover
C_Rcv	BITSTRING	’101’B	activation for receiving only
C_LocEndRel	RelMode	’01’O	local end release
C_MaxPwrLvlD	INTEGER	15	Max power level for MS DCS
C_MaxPwrLvlG	INTEGER	19	Max power level for MS GSM
C_rc_conditIEerror	REJCAU	’64’O	reject cause: Conditional IE error
C_rc_congestion	REJCAU	’16’O	reject cause: congestion
C_rc_illegal_ms	REJCAU	’03’O	reject cause: illegal MS
C_rc_illegal_me	REJCAU	’06’O	reject cause: illegal ME
C_rc_imsiunknownhlr	REJCAU	’02’O	reject cause: IMSI unknown in HLR
C_rc_imsiunknownvlr	REJCAU	’04’O	reject cause: IMSI unknown in VLR
C_rc_invalidmaninfo	REJCAU	’60’O	reject cause: invalid mandatory information
C_rc_LAnotallowed	REJCAU	’0C’O	reject cause: LocationArea is not allowed
C_rc_networkfailure	REJCAU	’11’O	reject cause: network failure
C_rc_notidentified	REJCAU	’26’O	reject cause: can not be identified
C_rc_plmn_not	REJCAU	’0B’O	reject cause: PLMN not allowed
C_rc_protocolerror	REJCAU	’6F’O	reject cause: Protocol error unspecified
C_rc_reqservoptnotsub	REJCAU	’21’O	reject cause: requested service option not subscribed
C_rc_roamingnot	REJCAU	’0D’O	reject cause: Roaming not allowed
C_Restablishment	B_1	’0’B	Restablishment allowed
C_noRestablishment	B_1	’1’B	No restablishment allowed

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_SlotNU	SN	'111'B	Slot Number not used
C_CellReselectHys12	INTEGER	6	Cell Reselection Hysteresis value 12db
C_CellReselectHys4	INTEGER	2	Cell Reselection Hysteresis value 4db
C_RI_alternate	B_8	'11010001'B	Bcap repeat indicator for alternate (circular selection)
C_RI_follow	B_8	'11010011'B	Bcap repeat indicator for successive (sequential selection)
C_SIMIn	BOOLEAN	FALSE	SIM is always inserted in a test case
C_SIMneedRmv	BOOLEAN	TRUE	Removal of SIM is needed in a test case (SIM is not always inserted.)
C_Yes	BOOLEAN	TRUE	
C_No	BOOLEAN	FALSE	
C_Start_Imsi	INTEGER	1	Test case body starts without TMSI
C_Start_Tmsi	INTEGER	2	Test case body starts with Tmsi_01
C_Start_TmsiOff	INTEGER	5	Test case body starts with Tmsi_01. MS has been deactivated.
C_Start_Rej	INTEGER	3	Test case body starts with LAI deleted. MS has been deactivated.
C_Start_PLMNnot	INTEGER	4	Test case body starts with PLMN not allowed
C_Test_fh	INTEGER	1	Test case with possibly frequency hopping
C_Test_nfh	INTEGER	2	Test case with no frequency hopping
C_Test_egsm	INTEGER	3	Test case for egsm
C_Test_cbms	INTEGER	4	Test case for CBMS
C_normal_updating	B_2	'00'B	normal location updating
C_periodic_updating	B_2	'01'B	periodic location updating
C_normalOrperiodic	B_2	'0?'B	periodic or normal location updating
C_imsi_attach	B_2	'10'B	location updating IMSI attach
C_norm_period_attach	B_2	'??'B	imsi attach or periodic location updating or normal location updating
C_normal_paging	B_2	'00'B	normal page mode
C_extended_paging	B_2	'01'B	extended paging
C_reorg_paging	B_2	'10'B	paging reorganisation
C_sab_paging	B_2	'11'B	same as before paging
C_non_synchronised	B_2	'00'B	non−synchronised
C_synchronised	B_2	'01'B	synchronised
C_pre_synchronised	B_2	'10'B	pre−synchronised
C_pseudo_synchronised	B_2	'11'B	pseudo−synchronised

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_not_report_otd	B_1	'0'B	not report observed time difference
C_report_otd	B_1	'1'B	report observed time difference
C_Tzone0	TZONES	0	Time zone 0 (used in TC_34_2_1, TC_34_2_5_3)
C_Tzone1	TZONES	4	Time zone 4 (used in TC_34_2_2, TC_34_2_7)
C_Tzone2	TZONES	10	Time zone 10 (used in TC_34_2_3)
C_Tzone3	TZONES	15	Time zone 15 (used in TC_34_2_4)
C_Tzone4	TZONES	25	Time zone 25 (used in TC_34_2_5_1)
C_Tzone5	TZONES	35	Time zone 35 (used in TC_34_2_5_2)
C_Tzone6	TZONES	45	Time zone 45 (used in TC_34_2_8)
C_T_AssCmd	INTEGER	900	600ms + 300ms(tolerance)
C_T_Wait1stChReq	INTEGER	35000	max time for full BCCH data decoding
C_T_mrsrp	INTEGER	1500	3 times time btw 2 msr rpt.
C_T_T3124	INTEGER	675	T3124(675ms)
C_T_T3211plus	INTEGER	20000	20s
C_T_T3212	INTEGER	360000	T3212(6min)
C_T_T3212min	INTEGER	345000	T3212(6min) − 15sec
C_T_T3212dif	INTEGER	15000	Difference between T3212 and T3212min
C_T_T3210	INTEGER	20000	timer T3210 in ms
C_T_T3211min	INTEGER	13500	90% T3211 in ms
C_T_T3230min	INTEGER	13500	90% T3230 in ms
C_T_T3240min	INTEGER	9000	90% T3240 in ms
C_T_T3240tol	INTEGER	2000	20% T3240 in ms
C_T_Wait	INTEGER	5000	5s. T3126max
C_T_Wait1	INTEGER	1000	1s
C_T_Wait_FAC	INTEGER	1500	1.5s Facility response time (1s) + tolerance (0.5s)
C_release_time	INTEGER	11	release timer value
C_RegCFNRy	INTEGER	1	registration of call forwarding service for CFNRy (speech)
C_RegCFU	INTEGER	2	registration of call forwarding service for CFU (all facsimile)
C_RegCF	INTEGER	3	registration of call forwarding service for CF (all facsimile)
C_ErsCFC	INTEGER	4	erasure of call forwarding service for CFC (all facsimile)

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_ErsCFNRc	INTEGER	5	erasure of call forwarding service for CFNRc (all basic services)
C_ErsCFNRy	INTEGER	6	erasure of call forwarding service for CFNRy (all facsimile)
C_ActCF	INTEGER	7	activation of call forwarding service for CF (all synchronous services)
C_ActCFU	INTEGER	8	activation of call forwarding service for CFU (all basic services)
C_DeactCFC	INTEGER	9	deactivation of call forwarding service for CFC (speech)
C_DeactCFNRc	INTEGER	10	deactivation of call forwarding service for CFNRc (all facsimile)
C_InterrogCFC	INTEGER	11	interrogation of call forwarding service for CFC (Speech)
C_InterrogCFB	INTEGER	12	interrogation of call forwarding service for CFB (all basic services)
C_NotifyCFB	INTEGER	13	notification of call forwarding service for CFB (incoming call is forwarded)
C_NotifyCFU	INTEGER	14	notification of call forwarding service for CFU (provisioned, registered, active)
C_NotifyCFC	INTEGER	15	notification of call forwarding service for CFC (provisioned, registered, active)
C_NotifyCFNRc	INTEGER	16	notification of call forwarding service for CFNRc (Ms not reachable)
C_RegPswd	INTEGER	17	Registration of password for all call barring services
C_ActBOAC	INTEGER	18	Activation of BOAC
C_ActBICRoam	INTEGER	19	Activation of BICRoam
C_ActBOIC	INTEGER	20	Activation of BOIC
C_ActBAIC	INTEGER	21	Activation of BAIC
C_DeactBO	INTEGER	22	Deactivation of BO
C_DeactBI	INTEGER	23	Deactivation of BI
C_DeactBOICExHC	INTEGER	24	Deactivation of BOICExHC
C_NotifyBI	INTEGER	25	Notify of BI
C_InterrogBOIC	INTEGER	26	Interrogation of BOIC
C_InterrogBOICExHC	INTEGER	27	Interrogation of BOICExHC
C_Full	IA5String	"F"	full rate channel (any speech version or data)
C_Full_v2	IA5String	"Fv2"	full rate speech channel, version 2

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_Half	IA5String	"H"	half rate channel
C_nirr_dontcare	B_1	'?'B	NIRR no meaning or requested
C_Rchr_dontcare	B_2	'??'B	radio channel requirement not checked
C_Rchr_Full	B_2	'01'B	radio channel requirement : full rate
C_Rchr_DualForDualH	B_2	'1?'B	radio channel requirement : dual rate support/half rate preferred or dual rate support/full rate preferred
C_Rchr_DualForDualH0	B_2	'10'B	radio channel requirement : dual rate support/half rate preferred or dual rate support/full rate preferred
C_StartingTimeHO	INTEGER	238	Starting time 238 (1.1s), in TC_26_6_5_1, TC_26_6_5_4_2
C_StrT_1000	INTEGER	1000	Starting time 1000, in TC_26_6_13_1
C_Norm	OCTETSTRING	'00'O	Establish mode: normal
C_Speech	B_3	'000'B	Info. Transfer Cap. speech
C_3100Hz	B_3	'010'B	Info. Transfer Cap. 3.1 kHz
C_UDI	B_3	'001'B	Info. Transfer Cap. UDI
C_FAX3	B_3	'011'B	Info. Transfer Cap. FAX3
C_SDUintegrity	B_2	'00'B	Structure SDU integrity
C_Unstructured	B_2	'11'B	Structure Unstructured
C_struc_dontchk	B_2	'??'B	Structure not checked
C_nirr_nomeaning	B_1	'0'B	NIRR no meaning
C_no_rate_adaption	B_2	'00'B	Rate adaption no
C_rate_adaption_V110	B_2	'01'B	Rate adaption V110
C_rate_adaption_X31	B_2	'10'B	Rate adaption X31 flag
C_sacp_NA	B_3	'001'B	Signaling access NA (sec.11.8.1.5.1)
C_I440_450	B_3	'001'B	Signaling access I440/450
C_X21	B_3	'010'B	Signaling access X21
C_X28_unui	B_3	'100'B	Signaling access X28 U NUI
C_X28_nond	B_3	'101'B	Signaling access X28 non dedicated
C_X32	B_3	'110'B	Signaling access X32
C_Synchronous	B_1	'0'B	Synchronous
C_Asynchronous	B_1	'1'B	Asynchronous
C_stopbit_NA	B_1	'0'B	Number of Stop bits NA
C_stopbit_dontcare	B_1	'?'B	Number of Stop bits not care
C_databit_NA	B_1	'1'B	Number of Data bits not care
C_databit_dontcare	B_1	'?'B	Number of Data bits NA
C_300bs	B_4	'0001'B	User rate 300 bs
C_1200bs	B_4	'0010'B	User rate 1200 bs
C_120075bs	B_4	'0111'B	User rate 1200/75 bs

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_2400bs	B_4	'0011'B	User rate 2400 bs
C_4800bs	B_4	'0100'B	User rate 4800 bs
C_9600bs	B_4	'0101'B	User rate 9600 bs
C_parity_NA	B_3	'011'B	Parity NA
C_parity_dontcare	B_3	'???'B	Parity not care
C_ir_8kbs	B_2	'10'B	Intermediate rate 8 Kbs
C_ir_16kbs	B_2	'11'B	Intermediate rate 16 Kbs
C_ir_8or16kbs	B_2	'1?'B	Intermediate rate 8 or 16 Kbs
C_transparent	B_2	'00'B	Connection element T
C_notransparent	B_2	'01'B	Connection element NT
C_BothT	B_2	'10'B	Connection element both T
C_BothNT	B_2	'11'B	Connection element both NT
C_modemt_none	B_5	'00000'B	no modem
C_modemt_V21	B_5	'00001'B	modem V21
C_modemt_V22	B_5	'00010'B	modem V22
C_modemt_V22bis	B_5	'00011'B	modem V22bis
C_modemt_V23	B_5	'00100'B	modem V23
C_modemt_V32	B_5	'00110'B	modem V32
C_ISO6429	B_5	'01000'B	Inband flow control
C_COPnoFLCT	B_5	'01100'B	No flow control
C_X25_flct	B_5	'00110'B	X25 flow control
C_Outband	FLWCNTL	0	outband flow control
C_Inband	FLWCNTL	1	outband flow control
C_Nocontrol	FLWCNTL	2	outband flow control
C_uil2p_dontcare	B_5	'?????'B	not checked
C_HSN_0	INTEGER	0	cyclic hopping
maxAddressLength	Asn1Integer	20	
maxISDN_AddressLength	Asn1Integer	9	
maxISDN_SubaddressLength	Asn1Integer	21	
maxNumOfBasicServiceGroups	Asn1Integer	13	
maxNumOfCUG	Asn1Integer	10	
maxNumberOfSegmentsPerDataInterval	Asn1Integer	8191	
maxSignalInfoLength	Asn1Integer	200	
maxUSSD_StringLength	Asn1Integer	160	
max10TimesIncrement	Asn1Integer	8191	
max10TimesIncrementPerDataInterval	Asn1Integer	8191	
max10TimesInitialTime	Asn1Integer	8191	
max100TimesScalingFactor	Asn1Integer	8191	
max10TimesTimeInterval	Asn1Integer	8191	
max10TimesUnitsPerTime	Asn1Integer	8191	
Detailed Comments :			

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_CC	BOOLEAN	TSPC_Serv_TS11 OR TSPC_Serv_TS12 OR TSPC_Serv_TS61 OR TSPC_Serv_TS62 OR TSPC_Serv_BS21 OR TSPC_Serv_BS22 OR TSPC_Serv_BS23 OR TSPC_Serv_BS24 OR TSPC_Serv_BS25 OR TSPC_Serv_BS26 OR TSPC_Serv_BS31 OR TSPC_Serv_BS32 OR TSPC_Serv_BS33 OR TSPC_Serv_BS34 OR TSPC_Serv_BS41 OR TSPC_Serv_BS42 OR TSPC_Serv_BS43 OR TSPC_Serv_BS44 OR TSPC_Serv_BS45 OR TSPC_Serv_BS46 OR TSPC_Serv_BS51 OR TSPC_Serv_BS52 OR TSPC_Serv_BS53 OR TSPC_Serv_BS61 OR TSPC_Serv_BS81	
TCV_AssCmd	ASS_CMD_PDU		to hold ASSIGNMENT COMMAND PDU
TCV_Bcap1	BCAP		to hold bearer capability
TCV_BcapMT1	BCAP		to hold BC for MT call
TCV_BcapMO1	BCAP		to hold BC for MO call
TCV_Bcap2	BCAP		to hold bearer capability
TCV_BcapMT2	BCAP		to hold BC for MT call
TCV_bs_ag_blks_res	INTEGER		
TCV_bs_cc_chans	INTEGER		
TCV_bs_pa_mfrms	INTEGER		
TCV_CalledNum	CDPN		to hold called party number
TCV_CallProc	CALL_PROC_PDU		to hold CALL PROCEEDING PDU
TCV_CallCfm	CALL_CO_PDU		to hold CALL COMFIRM PDU
TCV_Cau	CAU		to hold cause
TCV_Cau0	CAU		to hold cause
TCV_CAgsm	CCHD		To compute Cell Channel Descriptor for gsm
TCV_CAdcs	CCHD		To compute Cell Channel Descriptor for dcs
TCV_CB1	OCTETSTRING		store SMSCB block data
TCV_CB2	OCTETSTRING		store SMSCB block data
TCV_CB3	OCTETSTRING		store SMSCB block data
TCV_CB4	OCTETSTRING		store SMSCB block data
TCV_CE	CE		
TCV_Ccchg	INTEGER		CCCH_GROUP
TCV_Ccd0A	CCD		a copy of control channel description in use for cell A

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_Ccd0B	CCD		a copy of control channel description in use for cell B
TCV_Ccd0C	CCD		a copy of control channel description in use for cell C
TCV_Ccd0D	CCD		a copy of control channel description in use for cell D
TCV_Ccd0E	CCD		a copy of control channel description in use for cell E
TCV_Ccd0F	CCD		a copy of control channel description in use for cell F
TCV_Ccd0G	CCD		a copy of control channel description in use for cell G
TCV_Ccd0H	CCD		a copy of control channel description in use for cell H
TCV_cchdescr	OCTETSTRING		Frequency list of Cell Channel Descr, used in EGSM
TCV_CCst	CST		to hold call status
TCV_cchannel	B_3		to hold the type of the control channel in IdleUpdated test step
TCV_ch	LOGICCH	"dummy"	to hold logic channel
TCV_ch1	LOGICCH	"dummy"	to hold logic channel
TCV_sacch	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_B	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_C	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_D	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_E	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_F	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_G	LOGICCH	"dummy"	to hold logic channel
TCV_sacch_H	LOGICCH	"dummy"	to hold logic channel
TCV_sacch8	LOGICCH	"dummy"	to hold logic channel
TCV_sacchTch	LOGICCH	"dummy"	to hold logic channel
TCV_sacchTch1	LOGICCH	"dummy"	to hold logic channel
TCV_sacchTch_H	LOGICCH	"dummy"	to hold logic channel
TCV_chTch	LOGICCH	"dummy"	to hold logic channel
TCV_chTch1	LOGICCH	"dummy"	to hold logic channel
TCV_chSms	LOGICCH	"dummy"	to hold logic channel
TCV_agch	LOGICCH	"dummy"	downlink access grant channel
TCV_WorkingCh	LOGICCH	"dummy"	to hold logic channel
TCV_cellid	CellID		to hold the cell identifier
TCV_Char	IA5String		to hold the IA5 char. corresponding to a DTMF tone
TCV_chd1	CHD		to hold channel description IE
TCV_cchd1	CCHD		to hold cell channel description IE
TCV_ChMod	CHMOD		to hold channel mode IE

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_ChModb	CHMOD		to hold channel mode IE for second mode in dual mode services
TCV_Tchtype	CH_TDMA	’00001’B	Traffic channel type and TDMA offset in channel description IE
TCV_chtype	CH_TDMA		to channel type and TDMA offset in channel description IE
TCV_cksn	BITSTRING		to hold the CKSN.
TCV_Cnt	INTEGER		general loop counter
TCV_Cnt1	INTEGER		general loop counter
TCV_Cnt2	INTEGER		general loop counter
TCV_Cntstart	BOOLEAN	FALSE	general loop counter start decision
TCV_Cntref	INTEGER		Reference for general loop counter
TCV_CphAlg	BITSTRING		ciphering algorithm
TCV_CphKey	BITSTRING		ciphering key
TCV_CphMd	CPHMS		ciphering mode setting
TCV_CPDataRetx	INTEGER		Number of CP Data retransmissions for SMS
TCV_ChRate	IA5String	C_Full	to hold channel rate
TCV_ChRate1	IA5String	C_Full	to hold channel rate
TCV_counter_c	INTEGER		loop counter c
TCV_counter_k	INTEGER		loop counter k
TCV_Fk	INTEGER		to hold an interval between events
TCV_flist	OCTETSTRING		Frequency List, used in EGSM for IE "frequency list" or "frequency short list"
TCV_flistl	LENGTH		Length of Frequency List, used in EGSM for IE "frequency list" or "frequency short list"
TCV_Fn	FN		to hold the frame number (T1’ , T2, T3)
TCV_Fn1	FN		to hold the frame number (T1’ , T2, T3)
TCV_FnAss	FN		to hold the frame number in AssCmd statement
TCV_FnBurst	FN		to hold the frame number of first burst of handover access
TCV_FollowingOctets	OCTETSTRING		Contains the octets following the invoke id
TCV_HoaccPara	HOACC_PARA		Handover access timing and power
TCV_Horf	HORF		Variable for HO reference in HO−Messages of HO cases.
TCV_HoCMD	HO_CMD_PDU		to hold handover PDU
TCV_Hrf	HORF		to hold handover reference

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_iel	LENGTH		Length of IE
TCV_kcnt	INTEGER		loop counter
TCV_K	INTEGER		
TCV_L1Head0	L1HD		to hold L 1 header
TCV_L1Head	L1HD		to hold L 1 header
TCV_lac	OCTETSTRING		to hold the lac. 2 octets
TCV_lacunknown	OCTETSTRING	'??'O	to hold a wild card lac when the lac is unknown
TCV_n	INTEGER		
TCV_Neci	BITSTRING		
TCV_M	INTEGER		counter
TCV_maio	MAIO		to hold MAIO
TCV_mae1	BITSTRING		Mobile Allocation in EGSM. It includes the 1.octet of MA
TCV_mae2	BITSTRING		Mobile Allocation in EGSM. It includes the 2.octet of MA
TCV_Max	BITSTRING		to hold Max−retrans bitstring
TCV_MemCapExcd	BOOLEAN		RP error Memory Capacity Exceeded was sent
TCV_MsrRes	MSRR		to hold measurement results
TCV_Mt	BITSTRING		to hold message type
TCV_Mt1	BITSTRING		to hold message type
TCV_Modify	MODIFY_PDU		To hold Modify message
TCV_Null	BOOLEAN		to collect useless result from some test suite operations
TCV_PgCh	LOGICCH	"dummy"	logical channel for paging
TCV_Pgg	PGG		paging group
TCV_PreviousOctets	OCTETSTRING		Contains the octets previous to the invoke id
TCV_ProtErrorUnspec	BOOLEAN		RP error Protocol Error Unspecified was sent
TCV_Pwrlvl	BITSTRING		to hold the power level
TCV_Pwrlvl_ho	BITSTRING		power level for HO_CMD, used in HO cases.
TCV_RaCh	LOGICCH		random access channel parameter.
TCV_Rchr	B_2		to hold Radio channel requirement
TCV_Res	BOOLEAN		to hold the result of a test suite operation
TCV_Rpmr	MR		RP message reference for SMS
TCV_Rqr	RQR		to hold the request reference
TCV_Rqr9	RQR		to hold the request reference differing from TCV_Rqr
TCV_Rqr10	RQR		to hold the request reference differing from TCV_Rqr, TCV_Rqr9

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_Rqr11	RQR		to hold the request reference differing from TCV_Rqr, TCV_Rqr9, TCV_Rqr10
TCV_Rr	BITSTRING		to hold the random request reference
TCV_Rr1	BITSTRING		to hold the random reference
TCV_Service	SERVICES		to hold basicservice selected
TCV_Service1	SERVICES		to hold basicservice selected
TCV_Setup_mo	SETUP_MO_PDU		to hold the SETUP PDU (MO)
TCV_Setup_mo1	SETUP_MO_PDU		to hold the SETUP PDU (MO)
TCV_Setup_mt	SETUP_MT_PDU		to hold the SETUP PDU (MT)
TCV_Setup_mt1	SETUP_MT_PDU		to hold the 2nd SETUP PDU (MT)
TCV_2ndtest	BOOLEAN		2nd test required in TC_11_1_1
TCV_EFR_svi1	B_4		Holds 1st Speech Verson Indication from received BCAP
TCV_EFR_svi2	B_4		Holds 2nd Speech Verson Indication from received BCAP
TCV_EFR_svi3	B_4		Holds 3rd Speech Verson Indication from received BCAP
TCV_Esetup	ESETUP_PDU		to hold the emergency SETUP PDU
TCV_Esetup1	ESETUP_PDU		to hold the emergency SETUP PDU
TCV_Ecall	BOOLEAN	FALSE	it is set to TRUE, if the service under test is emergency call
TCV_slot	SN		To hold the default slot used during the entire TC.
TCV_slt2	SN		To hold a second working slot number.
TCV_Sres	OCTETSTRING		to hold the SRES returned
TCV_sysinf5	SYSINFO5_PDU		to hold the SysInfo5 PDU
TCV_sysinfo5	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell A
TCV_sysinfo5bis	SYSINFO5bis_PDU		to hold the SysInfo5bis PDU for cell A
TCV_sysinfo5_B	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell B
TCV_sysinfo5_C	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell c
TCV_sysinfo5_D	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell D
TCV_sysinfo5_E	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell E

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_sysinfo5_F	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell F
TCV_sysinfo5_G	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell G
TCV_sysinfo5_H	SYSINFO5_PDU		to hold the SysInfo5 PDU for cell H
TCV_sysinfo5bis_A	SYSINFO5bis_PDU		to hold the SysInfo5bis PDU for cell A(TC_26_10_2_4_1)
TCV_sysinfo5bis_B	SYSINFO5bis_PDU		to hold the SysInfo5bis PDU for cell B
TCV_sysinf6	SYSINFO6_PDU		to hold the SysInfo6 PDU
TCV_sysinfo6	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell A
TCV_sysinfo6_B	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell B
TCV_sysinfo6_C	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell C
TCV_sysinfo6_D	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell D
TCV_sysinfo6_E	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell E
TCV_sysinfo6_F	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell F
TCV_sysinfo6_G	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell G
TCV_sysinfo6_H	SYSINFO6_PDU		to hold the SysInfo6 PDU for cell H
TCV_temp	INTEGER		to hold results of expressions that are to be passed as parameters
TCV_ti_v	TI_V		to hold the transaction id. value
TCV_ti_v_2	TI_V		to hold the transaction id. value for another transaction
TCV_T	INTEGER		to hold Tx−integer chosen randomly
TCV_T_TwiceTC1M	INTEGER	TSPX_TC1M * 2000	Twice TC1M in ms
TCV_chdescr_arfcn	INTEGER		ARFCN carrier of actual cell
TCV_tch_arfcn	INTEGER		ARFCN carrier for traffic channel
TCV_asscmd_ts	BITSTRING		Time slot for channel description in assignment cmd.
TCV_ts	BITSTRING		Time slot for channel description in handover cmd.
TCV_ia_ts	BITSTRING		Time slot for channel description in immediate assignment
TCV_tsc	TSC		To hold the default training sequence used during the entire TC.

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_Td	OCTETSTRING		to hold timing difference
TCV_Time	INTEGER		to hold the measured value of T310 or T305 or T308
TCV_TI	TI		to hold the transaction ID for MO calls in sending evets
TCV_TI0	TI		to hold the transaction ID for MO calls in receiving evets
TCV_TI1	TI		to hold the transaction ID
TCV_TI2	TI		to hold the transaction ID
TCV_TI3	TI		to hold the transaction ID
TCV_tmp	INTEGER		temporary integer variable
TCV_Tx	BITSTRING		to hold Tx bitstring
TCV_S	INTEGER		to hold the S parameter
TCV_Upd	BOOLEAN		to hold the information whether the MS is updated or not,.
TCV_UssdString	IA5String		String for USS Data
TCV_UssdString1	IA5String		String for USS Data
TCV_UssdString2	IA5String		String for USS Data
TCV_InvkId	OCTETSTRING		to hold SS transaction ID
TCV_InvkId0	OCTETSTRING		to hold SS transaction ID
TCV_InvkId1	OCTETSTRING		to hold SS transaction ID
TCV_Comp	Components		to hold SS Components
TCV_Strt	STRT		to hold starting time
TCV_CBch	LOGICCH	"dummy"	Cell Broadcast channel
TCV_Tpmr	MR		TP message reference
TCV_TPOA1	BCDN		TP originating address digits
TCV_TPOA2	BCDN		TP originating address digits
TCV_TPDA	BCDN		TP destination address digits
TCV_RPOA_MT	BCDN		RP originating address digits for MT short messages
TCV_RPOA_MO	BCDN		RP originating address digits for MO short messages
TCV_RPOA1	BCDN		RP originating address digits for MT short messages
TCV_RPOA2	BCDN		RP originating address digits for MT short messages
TCV_RPDA_MT	BCDN		RP destination address digits for MT short messages
TCV_RPDA_MO	BCDN		RP destination address digits for MO short messages
TCV_SMTypeM	INTEGER		Replace short message type
TCV_SMTypeN	INTEGER		Replace short message type
TCV_SMcntns	OCTETSTRING		short message contents
TCV_SMSCBpack	SMSCBpack		CB message contents
TCV_freq	FRQPARA		to hold Freq type constraints

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_srv	SERVICES		to hold MT service
TCV_ur	B_4		to hold user rate
TCV_itc	B_3		to hold information transfer capability
TCV_sacp	B_3		to hold service access protocol
TCV_ce	B_2		to hold connection element
TCV_ce1	B_2		to hold connection element
TCV_ce3	B_2		to hold connection element
TCV_sa	B_1		to hold synchronous/asynchronous
TCV_ir	B_2		to hold intermediate rate
TCV_uil1p	B_2		to hold user information layer 1 protocol
TCV_modemt	B_5		to hold modem type
TCV_MODEM	MODEMTYPE		
TCV_ur2	B_4		to hold user rate for second configuration
TCV_itc2	B_3		to hold information transfer capability for second configuration
TCV_sacp2	B_3		to hold service access protocol for second configuration
TCV_strc	B_2		to hold structure
TCV_sa2	B_1		to hold synchronous/asynchronous for second configuration
TCV_supported	BOOLEAN	FALSE	used for indicating a selected service is declared supported or not
TCV_SpecialCase	BOOLEAN	FALSE	used for indicating whether a test case tests non−supported service
TCV_nc	B_3		Number of neighbour cells in a Measurement Report to be reported on
TCV_No_of_cells	B_3		To hold no. of neighbouring cells from measurement report.
TCV_Measure_Count	INTEGER	0	To hold number of Measurement reports.
TCV_CellPresent	INTEGER		
Detailed Comments :			

PCO Type Declarations		
PCO Type	**Role**	**Comments**
SAP0_3	LT	
Detailed Comments :		

PCO Declarations			
PCO Name	**PCO Type**	**Role**	**Comments**
L	SAP0_3	LT	
Detailed Comments :	1. The PCO consists of multiple SAPs: the SAP0 and SAP3. 2. The lower tester (LT) is the user of the data link layer service. 3. The SAP0 at the lower tester controlling and observing the exchange of CC, MM, RR and SS PDUs (messages) on the DCCH, SACCH and/or RACH, BCCH, CCCH channels. 4. The SAP3 at the lower tester controlling and observing the exchange of SMS PDUs (messages) on the DCCH or SACCH channels.		

Timer Declarations			
Timer Name	**Duration**	**Unit**	**Comments**
T_dly		ms	general purpose delay timer
T_dly1		ms	general purpose delay timer
T_dly2		ms	general purpose delay timer
T_dlyAss	5000	ms	AssCmd timer, 1)
T_guard	300	s	guard timer
T_release	C_release_time	s	release timer

Detailed Comments : 1) In order to give to the lower tester enough time to dequeue and to send out the call SETUP and the other call establishment related messages in the sending buffer before starting sending of ASSINGMENT COMMAND message, the default value 5s is used in the test step AssCh_complete. In the test step AssCh_failure the timer value is set to 4s. For the TC_26_6_4_1, 600ms is used.

ASP Type Definition

ASP Name : DL_EstRq
(DL_ESTABLISH_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to request the establishment of multiple frame operation (L3 −> L2). The normal establishment procedure is initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0, 3
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	Norm

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstIn
(DL_ESTABLISH_INDICATION)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the establishment of multiple frame operation (L2 −> L3). The normal establishment procedure has been initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0, 3
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	Norm
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstInCmsRq
(DL_ESTABLISH_INDICATION_CM_SERVICE_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the establishment of multiple frame operation due to the receipt of the MM CM SERVICE REQUEST message (L2 −> L3). The contention resolution establishment procedure has been initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	CoRes
msg	CMS_RQ_PDU	CM SERVICE REQUEST
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstInLupRq
(DL_ESTABLISH_INDICATION_LOCATION_UPDATING_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the establishment of multiple frame operation due to the receipt of the MM LOCATION UPDATING REQUEST message (L2 −> L3). The contention resolution establishment procedure has been initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	CoRes
msg	LUP_RQ_PDU	LOCATION UPDATING REQUEST
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstInImsidIn
(DL_ESTABLISH_INDICATION_IMSI_DETACH_INDICATION)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the establishment of multiple frame operation due to the receipt of the MM IMSI DETACH INDICATION message (L2 −> L3). The contention resolution establishment procedure has been initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	CoRes
msg	IMSID_IN_PDU	IMSI DETACH
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstInPgRes
(DL_ESTABLISH_INDICATION_PAGING_RESPONSE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the establishment of multiple frame operation due to the receipt of the RR PAGING RESPONSE message (L2 −> L3). The contention resolution establishment procedure has been initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	CoRes
msg	PG_RES_PDU	PAGING RESPONSE
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstInCmreRq
(DL_ESTABLISH_INDICATION_CM_REESTABLISHMENT_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the establishment of multiple frame operation due to the receipt of the MM CM REESTABLISHMENT REQUEST message (L2 −> L3). The contention resolution establishment procedure has been initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	CoRes
msg	CMRE_RQ_PDU	CM REESTABLISHMENT REQUEST
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_EstCo
(DL_ESTABLISH_CONFIRM)

PCO Type : SAP0_3

Comments : The ASP is used by the L2 to inform the L3 about the establishment of multiple frame link (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0, 3
logic_ch	LOGICCH	DCCH, SACCH
establish_mode	EstMode	Norm

Detailed Comments :

ASP Type Definition

ASP Name : DL_RelIn
(DL_RELEASE_INDICATION)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the termination of an established multiple frame operation or to report an unsuccessful establishment attempt (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0, 3
logic_ch	LOGICCH	DCCH, SACCH
release_mode	RelMode	normal, local end release
outstanding_indicator	BOOLEAN	
fn	FN	frame number of the first frame carrying the message

Detailed Comments : The outstanding indicator indicates whether or not there are outstanding acknowledgements or unsolved DL_DATA_REQUEST primitives.

ASP Type Definition

ASP Name : DL_RacInChRq
(DL_RANDOM_ACCESS_INDICATION_CHANNEL_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the arrival of an RR CHANNEL REQUEST message (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0,
logic_ch	LOGICCH	RACH
fn	FN	frame number when the access burst was received
msg	CH_RQ_PDU	REQUEST REFERENCE

Detailed Comments :

ASP Type Definition

ASP Name : DL_RacInHoacc
(DL_RADOM_ACCESS_INDICATION_HANDOVER_ACCESS)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the arrival of an RR HANDOVER ACCESS message (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0,
logic_ch	LOGICCH	DCCH
msg	HOACC_PDU	HANDOVER REFERENCE
fn	FN	frame number of the frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqImmass
(DL_UNIT_DATA_REQUEST_IMMEDIATE_ASSIGNMENT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR IMMEDIATE ASSIGNMENT message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	RACH (CCCH)
msg	IMMASS_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqImmass_sp
(DL_UNIT_DATA_REQUEST_IMMEDIATE_ASSIGNMENT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR IMMEDIATE ASSIGNMENT message using unacknowledged operation (L3 −> L2) on the MS paging channel.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	PGCH
pgg	PGG	Paging group of the mobile to be paged.
msg	IMMASS_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqImmassx
(DL_UNIT_DATA_REQUEST_IMMEDIATE_ASSIGNMENT_EXTENDED)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR IMMEDIATE ASSIGNMENT EXTENDED message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	AGCH (CCCH)
msg	IMMASSX_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqImmassx_sp
(DL_UNIT_DATA_REQUEST_IMMEDIATE_ASSIGNMENT_EXTENDED)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR IMMEDIATE ASSIGNMENT EXTENDED message using unacknowledged operation (L3 −> L2) on the MS paging channel.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	PGCH
pgg	PGG	Paging group of the mobile to be paged.
msg	IMMASSX_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqImmassRej
(DL_UNIT_DATA_REQUEST_IMMEDIATE_ASSIGNMENT_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR IMMEDIATE ASSIGNMENT REJECT message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	AGCH (CCCH)
msg	IMMASS_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqImmassRej_sp
(DL_UNIT_DATA_REQUEST_IMMEDIATE_ASSIGNMENT_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR IMMEDIATE ASSIGNMENT REJECT message using unacknowledged operation (L3 −> L2) on the MS paging channel.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	PGCH
pgg	PGG	Paging group of the mobile to be paged.
msg	IMMASS_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatInMsrRpt
(DL_DATA_INDICATION_MEASUREMENT_REPORT)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR MEASUREMENT REPORT message using unacknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	SACCH
msg	MSR_RPT_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments : The ASPs are continuously received during the testing.

ASP Type Definition

ASP Name : DL_UdatIn

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the filler frame message using unacknowledged operation

Parameter Name	Parameter Type	Comments
sapi	SAPID	
logic_ch	LOGICCH	
fn	FN	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSMSCBData
(DL_UNIT_DATA_REQUEST_SMSCB_DATA)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the SMSCB data using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	3
logic_ch	LOGICCH	CBCH
msg	SMSCB_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqPg1Rq
(DL_UNIT_DATA_REQUEST_PAGING_REQUEST_TYPE1)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR PAGING REQUEST TYPE 1 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	CCCH
pgg	PGG	the paging group of an MS to be paged
msg	PG1_RQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqPg2Rq
(DL_UNIT_DATA_REQUEST_PAGING_REQUEST_TYPE2)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR PAGING REQUEST TYPE 2 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	CCCH
pgg	PGG	the paging group of an MS to be paged
msg	PG2_RQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqPg3Rq
(DL_UNIT_DATA_REQUEST_PAGING_REQUEST_TYPE3)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR PAGING REQUEST TYPE 3 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	CCCH
pgg	PGG	the paging group of an MS to be paged
msg	PG3_RQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSchinfo
(DL_UNIT_DATA_REQUEST_SYNCHRONIZATION_CHANNEL_INFORMATION)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYNCHRONIZATION CHANNEL INFORMATION message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	SCH (BCCH)
msg	SCHINFO_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo1
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE1)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 1 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO1_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo1_nh
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE1)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 3 message in case of non hopping

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO3_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo2
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE2)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 2 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO2_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo2bis
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE2bis)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 2bis message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO2bis_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo2ter
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE2ter)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 2ter message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO2ter_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo3
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE3)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 3 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO3_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo4
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE4)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 4 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO4_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo5
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE5)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 5 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	SACCH
msg	SYSINFO5_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo5bis
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE5bis)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 5bis message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	SACCH
msg	SYSINFO5bis_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo5ter
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE5ter)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 5ter message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	SACCH
msg	SYSINFO5ter_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo6
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE6)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 6 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	SACCH
msg	SYSINFO6_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo7
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE7)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 7 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO7_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatRqSysinfo8
(DL_UNIT_DATA_REQUEST_SYSTEM_INFORMATION_TYPE8)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR SYSTEM INFORMATION TYPE 8 message using unacknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	BCCH
msg	SYSINFO8_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_UdatInClmChn
(DL_UdatINDICATION_CLASSMARK_CHANGE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR CLASSMARK CHANGE message using unacknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CLM_CHN_PDU	20 octets
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqAssCmd
(DL_DATA_REQUEST_ASSIGNMENT_COMMAND)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR ASSIGNMENT COMMAND message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ASS_CMD_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInAssCom
(DL_DATA_INDICATION_ASSIGNMENT_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR ASSIGNMENT COMPLETE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ASS_COM_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments : The ASP is a result of the ASP DL_RESUME_REQUEST of the SUT sending the ASSIGNMENT COMPLETE message.

ASP Type Definition

ASP Name : DL_DatInAssfl
(DL_DATA_INDICATION_ASSIGNMENT_FAILURE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR ASSIGNMENT FAILURE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH (old channel)
msg	ASSFL_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments : The ASP is a result of the ASP DL_RECONNECT_REQUEST of the SUT sending the ASSIGNMEN FAILURE message.

ASP Type Definition

ASP Name : DL_DatInCmsRq
(DL_DATA_INDICATION_CM_SERVICE_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the MM CM SERVICE REQUEST message (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH, SACCH
msg	CMS_RQ_PDU	CM SERVICE REQUEST
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqChmmo
(DL_DATA_REQUEST_CHANNEL_MODE_MODIFY)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR CHANNEL MODE MODIFY message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CHMMO_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInChmmoAck
(DL_DATA_INDICATION_CHANNEL_MODE_MODIFY_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR CHANNEL MODE MODIFY ACKNOWLEDGE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CHMMO_ACK_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqChRel
(DL_DATA_REQUEST_CHANNEL_RELEASE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR CHANNEL_RELEASE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CH_REL_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqChRelErr
(DL_DATA_REQUEST_CHANNEL_RELEASE_ERROR)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR CHANNEL_RELEASE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CH_REL_PDU_ERR	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCphmCmd
(DL_DATA_REQUEST_CIPHERING_MODE_COMMAND)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR CIPHERING_MODE_COMMAND message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CPHM_CMD_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCphmCmdErr
(DL_DATA_REQUEST_CIPHERING_MODE_COMMAND_ERROR)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the invalid RR CIPHERING_MODE_COMMAND message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CPHM_CMD_PDU_ERR	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInCphmCom
(DL_DATA_INDICATION_CIPHERING_MODE_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR CIPHERING MODE COMPLETE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CPHM_COM_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInClmChn
(DL_DAT_INDICATION_CLASSMARK_CHANGE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR CLASSMARK CHANGE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CLM_CHN_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqClmEnq
(DL_DATA_REQUEST_CLASSMARK_ENQUIRY)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR CLASSMARK ENQUIRY message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CLM_ENQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqFrqre
(DL_DATA_REQUEST_FREQUENCY_REDEFINITION)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR FREQUENCY REDEFINITION message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	FRQRE_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqHoCmd
(DL_DATA_REQUEST_HANDOVER_COMMAND)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR HANDOVER COMMAND message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	HO_CMD_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInHoCom
(DL_DATA_INDICATION_HANDOVER_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR HANDOVER COMPLETE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	HO_COM_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments : The ASP is a result of the ASP DL_RESUME_REQUEST of the SUT sending the HANDOVER COMPLETE message.

ASP Type Definition

ASP Name : DL_DatInHofl
(DL_DATA_INDICATION_HANDOVER_FAILURE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR HANDOVER FAILURE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH (old channel)
msg	HOFL_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments : The ASP is a result of the ASP DL_RECONNECT_REQUEST of the SUT sending the HANDOVER FAILURE message.

ASP Type Definition

ASP Name : DL_DatRqPhyinfo
(DL_DATA_REQUEST_PHYSICAL_INFORMATION)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the RR PHYSICAL INFORMATION message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	PHYINFO_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInRrst
(DL_DATA_INDICATION_RR_STATUS)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the RR STATUS message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	RRST_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqAuthRej
(DL_DATA_REQUEST_AUTHENTICATION_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM AUTHENTICATION REJECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	AUTH_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqAuthRq
(DL_DATA_REQUEST_AUTHENTICATION_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM AUTHENTICATION REQUEST message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	AUTH_RQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInAuthRes
(DL_DATA_INDICATION_AUTHENTICATION_RESPONSE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the MM AUTHENTICATION RESPONSE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	AUTH_RES_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCmsAcp
(DL_DATA_REQUEST_CM_SERVICE_ACCEPT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM CM SERVICE ACCEPT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CMS_ACP_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCmsRej
(DL_DATA_REQUEST_CM_SERVICE_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM CM SERVICE REJECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CMS_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqAbrt
(DL_DATA_REQUEST_ABORT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM ABORT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ABRT_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqIdRq
(DL_DATA_REQUEST_IDENTITY_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM IDENTITY REQUEST message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ID_RQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInIdRes
(DL_DATA_INDICATION_IDENTIFY_RESPONSE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the MM IDENTITY RESPONSE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ID_RES_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqLupAcp
(DL_DATA_REQUEST_LOCATION_UPDATING_ACCEPT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM LOCATION UPDATING ACCEPT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	LUP_ACP_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqLupAcpErr
(DL_DATA_REQUEST_LOCATION_UPDATING_ACCEPT_ERR)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM LOCATION UPDATING ACCEPT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	LUP_ACP_PDU_ERR	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqLupRej
(DL_DATA_REQUEST_LOCATION_UPDATING_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM LOCATION UPDATING REJECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	LUP_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInMmst
(DL_DATA_INDICATION_MM_STATUS)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the MM STATUS message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	MMST_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqTmsireCmd
(DL_DATA_REQUEST_TMSI_REALLOCATION_COMMAND)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the MM TMSI REALLOCATION COMMAND message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	TMSIRE_CMD_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInTmsireCom
(DL_DATA_INDICATION_TMSI_REALLOCATION_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the MM TMSI REALLOCATION COMPLETE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	TMSIRE_COM_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqAlert
(DL_DATA_REQUEST_ALERTING)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC ALERTING message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ALERT_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInAlert
(DL_DATA_INDICATION_ALERTING)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC ALERTING message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ALERT_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInCallCo
(DL_DATA_INDICATION_CALL_CONFIRMED)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC CALL CONFIRMED message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CALL_CO_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCallProc
(DL_DATA_REQUEST_CALL_PROCEEDING)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC CALL PROCEEDING message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CALL_PROC_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqConn
(DL_DATA_REQUEST_CONNECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC CONNECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CONN_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqConnErr
(DL_DATA_REQUEST_CONNECT_ERR)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC CONNECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CONN_PDU_ERR	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInConn
(DL_DATA_INDICATION_CONNECT)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC CONNECT message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CONN_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqConnAck
(DL_DATA_REQUEST_CONNECT_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC CONNECT ACKNOWLEDGE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CONN_ACK_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInConnAck
(DL_DATA_INDICATION_CONNECT_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the reception of the CC CONNECT ACKNOWLEDGE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CONN_ACK_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqDisc
(DL_DATA_REQUEST_DISCONNECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC DISCONNECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	DISC_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqDiscErr
(DL_DATA_REQUEST_DISCONNECT_ERR)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC DISCONNECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	DISC_PDU_ERR	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInDisc
(DL_DATA_INDICATION_DISCONNECT)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC DISCONNECT message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	DISC_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInESetup
(DL_DATA_INDICATION_EMERGENCY_SETUP)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC EMERGENCY_SETUP message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ESETUP_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqFac
(DL_DATA_REQUEST_FACILITY)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC FACILITY message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	FAC_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInFac
(DL_DATA_INDICATION_FACILITY)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC FACILITY message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	FAC_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInHold
(DL_DATA_INDICATION_HOLD)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC HOLD message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	HOLD_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqHoldAck
(DL_DATA_REQUEST_HOLD_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC HOLD ACKNOWLEDGE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	HOLD_ACK_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqModify
(DL_DATA_REQUEST_MODIFY)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC MODIFY message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	MODIFY_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInModify
(DL_DATA_INDICATION_MODIFY)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC MODIFY message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	MODIFY_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqModifyCom
(DL_DATA_REQUEST_MODIFY_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC MODIFY COMPLETE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	MODIFY_COM_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqModifyRej
(DL_DATA_REQUEST_MODIFY_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC MODIFY REJECT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	MODIFY_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInModifyRej
(DL_DATA_INDICATION_MODIFY_REJECT)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC MODIFY REJECT message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	MODIFY_REJ_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqNotify
(DL_DATA_REQUEST_NOTIFY)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC NOTIFY message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	NOTIFY_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqProg
(DL_DATA_REQUEST_PROGRESS)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC PROGRESS message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	PROG_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqRegister
(DL_DATA_REQUEST_REGISTER)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the SS REGISTER message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	REGISTER_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInRegister
(DL_DATA_INDICATION_REGISTER)

PCO Type : SAP0_3

Comments : The ASP is used to receive the transmission of the SS REGISTER message using acknowledged operation.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	REGISTER_PDU	
fn	FN	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqRel
(DL_DATA_REQUEST_RELEASE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC RELEASE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	REL_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInRel
(DL_DATA_INDICATION_RELEASE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC RELEASE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	REL_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqRelCmp
(DL_DATA_REQUEST_RELEASE_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC RELEASE COMPLETE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	REL_COM_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInRelCmp
(DL_DATA_INDICATION_RELEASE_COMPLETE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC RELEASE COMPLETE message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	REL_COM_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqSetup
(DL_DATA_REQUEST_SETUP)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC SETUP message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	SETUP_MT_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInSetup
(DL_DATA_INDICATION_SETUP)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC SETUP message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	SETUP_MO_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInStartDtmf
(DL_DATA_INDICATION_START_DTMF)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC START_DTMF message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	START_DTMF_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqStartDtmfAck
(DL_DATA_REQUEST_START_DTMF_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC START DTMF ACKNOWLEDGE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	START_DTMF_ACK_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqStartDtmfRej
(DL_DATA_REQUEST_START_DTMF_REJCT)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC START DTMF REJCT message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	START_DTMF_REJ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInStopDtmf
(DL_DATA_INDICATION_STOP_DTMF)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC STOP_DTMF message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	STOP_DTMF_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqStopDtmfAck
(DL_DATA_REQUEST_STOP_DTMF_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC STOP DTMF ACKNOWLEDGE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	STOP_DTMF_ACK_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCcst
(DL_DATA_REQUEST_CC_STATUS)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC STATUS message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CCST_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInCcst
(DL_DATA_INDICATION_CC_STATUS)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the CC STATUS message using acknowledged operation (L2 −> L3).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CCST_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCcstEnq
(DL_DATA_REQUEST_CC_STATUS_ENQ)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CC STATUS_ENQ message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CCST_ENQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqUndefCC
(DL_DATA_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to transmit any message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CONN_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqUndefMM
(DL_DATA_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to transmit any message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	ID_RES_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqUndefRR
(DL_DATA_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to transmit any message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	PART_REL_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqUnknown
(DL_DATA_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to transmit any message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	0
logic_ch	LOGICCH	DCCH
msg	CCST_ENQ_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInCpData
(DL_DATA_INDICATION_CP_DATA)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the SMS CP DATA message using acknowledged operation (L2 −> L3) for MT.

Parameter Name	Parameter Type	Comments
sapi	SAPID	3
logic_ch	LOGICCH	DCCH
msg	CP_DATA_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCpData
(DL_DATA_REQUEST_CP_DATA)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the SMS CP DATA message using acknowledged operation (L3 −> L2) for MO.

Parameter Name	Parameter Type	Comments
sapi	SAPID	3
logic_ch	LOGICCH	DCCH
msg	CP_DATA_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatInCpDataAck
(DL_DATA_INDICATION_CPDATA_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to indicate the receipt of the SMS CP DATA message using acknowledged operation (L2 −> L3) .

Parameter Name	Parameter Type	Comments
sapi	SAPID	3
logic_ch	LOGICCH	DCCH
msg	CPDATA_ACK_PDU	
fn	FN	frame number of the first frame carrying the message

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCpDataAck
(DL_DATA_REQUEST_CPDATA_ACKNOWLEDGE)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the CP DATA ACKNOWLEDGE message using acknowledged operation (L3 −> L2).

Parameter Name	Parameter Type	Comments
sapi	SAPID	3
logic_ch	LOGICCH	DCCH
msg	CPDATA_ACK_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : DL_DatRqCpError
(DL_DATA_REQUEST_CP_ERROR)

PCO Type : SAP0_3

Comments : The ASP is used to request the transmission of the SMS CP ERROR message using acknowledged operation (L3 −> L2) for MO.

Parameter Name	Parameter Type	Comments
sapi	SAPID	3
logic_ch	LOGICCH	DCCH
msg	CPERR_PDU	

Detailed Comments :

ASP Type Definition

ASP Name : MDL_RelRq
(MDL_RELEASE_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to request the local end termiination of a previous established acknowledged mode operation (L3 −> L2). The local end termination procedure is initiated.

Parameter Name	Parameter Type	Comments
sapi	SAPID	0, 3
logic_ch	LOGICCH	DCCH, SACCH
release_mode	RelMode	local end release

Detailed Comments :

PDU Type Definition

PDU Name : ASS_CMD_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR ASSSIGNMENT COMMAND n –> ms
GSM 04.08, 9.1.2

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
ch1d_at	CHD		description of the 1st channel, after time M OCTETSTRING [3]
pcmd	PCMD		power command M OCTETSTRING [1]
frql_at	FRQL		frequency list, after time C OCTETSTRING [4..132]
cchd	CCHD		cell channel description O OCTETSTRING [17]
ch1mod	CHMOD		mode of the 1st channel O OCTETSTRING [2]
ch2d_at	CHD		description of the 2nd channel, after time O OCTETSTRING [4]
ch2mod	CHMOD		mode 2 of the 2nd channel O OCTETSTRING [2]
ma_at	MA		mobile allocation after time C OCTETSTRING [3..10]
strt	STRT		starting time O OCTETSTRING [3]
frql_bt	FRQL		frequency list, before time C OCTETSTRING [4..132]
ch1d_bt	CHD		description of the 1st channel, before time O OCTETSTRING [4]
ch2d_bt	CHD		description of the 2nd channel, before time O OCTETSTRING [4]
frqchs_bt	FRQCHS		frequency channel sequence before time C OCTETSTRING [10]
ma_bt	MA		mobile allocation before time C OCTETSTRING [3..10]
cphms	CPHMS		cipher mode setting O OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : ASS_COM_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR ASSSIGNMENT COMPLETE ms −> n
GSM 04.08, 9.1.3

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cuase M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : ASSFL_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR ASSSIGNMENT FAILURE ms −> n
GSM 04.08, 9.1.4

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cuase M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : CHMMO_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CHANNEL MODE MODIFY n −> ms
GSM 04.08, 9.1.5

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chd	CHD		channel description M OCTETSTRING [3]
chmod	CHMOD		channel mode M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : CHMMO_ACK_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CHANNEL MODE MODIFY ACKNOWLEDGE ms −> n
GSM 04.08, 9.1.6

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chd	CHD		channel description M OCTETSTRING [3]
chmod	CHMOD		channel mode M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : CH_REL_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CHANNEL RELEASE n −> ms
GSM 04.08, 9.1.7

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cuase M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : CH_REL_PDU_ERR
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CHANNEL RELEASE containing additional unknown IE n −> ms
GSM 04.08, 9.1.7

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cuase M BITSTRING [8]
add	OCTETSTRING		additional unknown IE

Detailed Comments :

PDU Type Definition

PDU Name : CH_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CHANNEL REQUEST ms −> n
GSM 04.08, 9.1.8

Field Name	Field Type	Field Encoding	Comments
ecau_rrf	BITSTRING [8]		establishment cause / random reference M

Detailed Comments :

PDU Type Definition

PDU Name : CPHM_CMD_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CIPHERING MODE COMMAND n −> ms
GSM 04.08, 9.1.9

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
cph_res	CPH_RES		cipher response M BITSTRING [4]
cphms	CPHMS		cipher mode setting M BITSTRING [4]

Detailed Comments :

PDU Type Definition

PDU Name : CPHM_CMD_PDU_ERR
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CIPHERING MODE COMMAND with additional unknown IE
GSM 04.08, 9.1.9

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
cph_res	CPH_RES		cipher response M BITSTRING [4]
cphms	CPHMS		cipher mode setting M BITSTRING [4]
add	OCTETSTRING		additional unknown IE

Detailed Comments :

PDU Type Definition

PDU Name : CPHM_COM_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CIPHERING MODE COMPLETE ms −> n
GSM 04.08, 9.1.10

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
mei	MI		mobile equipment identity O OCTETSTRING [10], 15 digits, BCD

Detailed Comments :

PDU Type Definition

PDU Name : CLM_CHN_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CLASSMARK CHANGE ms −> n
GSM 04.08, 9.1.11

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
msclm	MSCLM2		mobile station classmark M OCTETSTRING [4]
msclm_adi	MSCLM3		addtional mobile station classmark information C OCTETSTRING [14]

Detailed Comments :

PDU Type Definition

PDU Name : CLM_ENQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR CLASSMARK ENQUIRY n −> ms
GSM 04.08, 9.1.12

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : FRQRE_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR FREQUENCY REDEFINITIONY n −> ms
GSM 04.08, 9.1.13

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chd	CHD		channel description M OCTETSTRING [3]
ma	MA		mobile allocation M OCTETSTRING [2..9]
strt	STRT		starting time M OCTETSTRING [2]
cchd	CCHD		cell channel description O OCTETSTRING [17]

Detailed Comments :

PDU Type Definition

PDU Name : HOACC_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR HANDOVER ACCESS ms −> n
GSM 04.08, 9.1.14

Field Name	Field Type	Field Encoding	Comments
horf	HORF		handover reference M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : HO_CMD_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR HANDOVER COMMAND n −> ms
GSM 04.08, 9.1.15

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
cd	CD		cell description M OCTETSTRING [2]
ch1d_at	CHD		description of the 1st channel, after time M OCTETSTRING [3]
horf	HORF		handover reference M BITSTRING [8]
pcmd	PCMD		power command M OCTETSTRING [1]
synchi	SYNCHI		synchronization indication O OCTETSTRING [1]
frqsl_at	FRQL		frequency short list, after time C OCTETSTRING [10]
frql_at	FRQL		frequency list, after time C OCTETSTRING [4..132]
cchd	CCHD		cell channel description C OCTETSTRING [17]
ch1mod	CHMOD		mode of the 1st channel O OCTETSTRING [2]
ch2d_at	CHD		description of the 2nd channel, after time O OCTETSTRING [4]
ch2mod	CHMOD		mode of the 2nd channel O OCTETSTRING [2]
frqchs_at	FRQCHS		frequency channel sequence, after time C OCTETSTRING [10]
ma_at	MA		mobile allocation, after time C OCTETSTRING [3..10]
strt	STRT		starting time O OCTETSTRING [3]
rtdif	TDIF		real time difference C OCTETSTRING [3]
ta	TA		timing advance C OCTETSTRING [2]
frqsl_bt	FRQL		frequency short list, before time C OCTETSTRING [10]

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	**Field Type**	**Field Encoding**	**Comments**
frql_bt	FRQL		frequency list, before time C OCTETSTRING [4..132]
ch1d_bt	CHD		description of the 1st channel, before time O OCTETSTRING [4]
ch2d_bt	CHD		description of the 2nd channel, before time O OCTETSTRING [4]
frqchs_bt	FRQCHS		frequency channel sequence, before time C OCTETSTRING [10]
ma_bt	MA		mobile allocation, before time C OCTETSTRING [3..10]
cphms	CPHMS		cipher mode setting O OCTETSTRING [1]
Detailed Comments :			

PDU Type Definition			
PDU Name : HO_COM_PDU			
PCO Type : SAP0_3			
Encoding Rule Name :			
Encoding Variation :			
Comments : RR HANDOVER COMPLETE ms −> n GSM 04.08, 9.1.16			
Field Name	**Field Type**	**Field Encoding**	**Comments**
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cuase M OCTETSTRING [1]
motdif	MTDIF		mobile observed time difference O OCTETSTRING [5]
Detailed Comments :			

PDU Type Definition

PDU Name : HOFL_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR HANDOVER FAILURE ms −> n
GSM 04.08, 9.1.17

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cuase M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : IMMASS_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR IMMEDIATE ASSIGNMENT n −> ms
GSM 04.08, 9.1.18

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
chd	CHD		channel description M OCTETSTRING [3]
rqr	RQR		request reference M OCTETSTRING [3]
ta	TA		timing advance M OCTETSTRING [1]
ma	MA		mobile allocation, M OCTETSTRING [1..9]
strt	STRT		starting time O OCTETSTRING [3]
iaroct	IARESTOCT		IA rest octets / frequency parameter, before time M

Detailed Comments : The message has a fixed length of 23 octets.

PDU Type Definition

PDU Name : IMMASSX_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR IMMEDIATE ASSIGNMENT n −> ms
GSM 04.08, 9.1.19

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
chd1	CHD		channel description 1 M OCTETSTRING [3]
rqr1	RQR		request reference 1 M OCTETSTRING [3]
ta1	TA		timing advance 1 M OCTETSTRING [1]
chd2	CHD		channel description 2 M OCTETSTRING [3]
rqr2	RQR		request reference 2 M OCTETSTRING [3]
ta2	TA		timing advance 2 M OCTETSTRING [1]
ma	MA		mobile allocation, M OCTETSTRING [1..5]
strt	STRT		starting time O OCTETSTRING [3]
iaxroct	OCTETSTRING [1..4]		IAX rest octets M

Detailed Comments : The message has a fixed length of 23 octets.

PDU Type Definition

PDU Name : IMMASS_REJ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR IMMEDIATE REJECT n −> ms
GSM 04.08, 9.1.20

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
rqr1	RQR		request reference 1 M OCTETSTRING [3]
wi1	WI		wait indication 1 M HEXSTRING [2]
rqr2	RQR		request reference 2 M OCTETSTRING [3]
wi2	WI		wait indication 2 M HEXSTRING [2]
rqr3	RQR		request reference 3 M OCTETSTRING [3]
wi3	WI		wait indication 3 M HEXSTRING [2]
rqr4	RQR		request reference 4 M OCTETSTRING [3]
wi4	WI		wait indication 4 M HEXSTRING [2]
iarroct	OCTETSTRING [3]		IAR rest octets M

Detailed Comments : The message has a fixed length of 23 octets.

PDU Type Definition

PDU Name : MSR_RPT_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR MEASUREMENT REPORT ms −> n
GSM 04.08, 9.1.21

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
msrr	MSRR		measurement results M OCTETSTRING [16]

Detailed Comments :

PDU Type Definition

PDU Name : PG1_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR PAGING REQUEST_TYPE1 n −> ms
GSM 04.08, 9.1.22

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chn_m1_2	CHNEED		channels needed for mobiles 1 and 2 M BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
mi1	MI		mobility identity 1 M OCTETSTRING [2..9]
mi2	MI		mobility identity 2 O OCTETSTRING [3..10]
p1roct	OCTETSTRING [0..17]		P1 rest octets '2B2B2B...'O

Detailed Comments : The message has a fixed length of 23 octets.

PDU Type Definition

PDU Name : PG2_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR PAGING REQUEST TYPE2 n −> ms
GSM 04.08, 9.1.23

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chn_m1_2	CHNEED		channels needed M BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
mi1	TMSI		mobility identity 1 M OCTETSTRING [4]
mi2	TMSI		mobility identity 2 M OCTETSTRING [4]
mi3	MI		mobility identity 3 O OCTETSTRING [3..10]
p2roct	OCTETSTRING [1..11]		p2 rest octets M

Detailed Comments : The message has a fixed length of 23 octets.

PDU Type Definition

PDU Name : PG3_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR PAGING REQUEST TYPE3 n –> ms
GSM 04.08, 9.1.24

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chn_m1_2	CHNEED		channels needed for mobiles 1 and 2 M BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
mi1	TMSI		mobility identity 1 M OCTETSTRING [4]
mi2	TMSI		mobility identity 2 M OCTETSTRING [4]
mi3	TMSI		mobility identity 2 M OCTETSTRING [4]
mi4	TMSI		mobility identity 4 M OCTETSTRING [4]
p3roct	OCTETSTRING [3]		P3 rest octets M

Detailed Comments : The message has an L2 pseudo length of 19 octets and a total length of 23 octets.

PDU Type Definition

PDU Name : PG_RES_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR PAGING RESPONSE ms −> n
GSM 04.08, 9.1.25

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
cphksn	CPHKSN		ciphering key sequence number M BITSTRING [4]
msclm	MSCLM2		mobile station classmark information M OCTETSTRING [4]
mi	MI		mobility identity M OCTETSTRING [2..9]

Detailed Comments :

PDU Type Definition

PDU Name : PART_REL_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR PARTIAL RELEASE n −> ms
GSM 04.08, 9.1.26

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
chd	MAC		wrong channel description M OCTETSTRING [3]

Detailed Comments :

PDU Type Definition

PDU Name : PHYINFO_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR PHYSICAL INFORMATION n −> ms
GSM 04.08 clause 9.1.28

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
ta	TA		timing advance M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : RRST_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR STATUS n <−> ms
GSM 04.08, 9.1.29

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rrcau	RRCAU		RR cause M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : SCHINFO_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYNCHRONIZATION CHANNEL INFORMATION n −> ms
GSM 04.08, 9.1.30

Field Name	Field Type	Field Encoding	Comments
ncc	NCC		PLMN colour code
bcc	BCC		BS colour code
t1	T1		part of reduced TDMA frame number
t2	T2		part of reduced TDMA frame number
t3_	T3_		part of reduced TDMA frame number

Detailed Comments : SCHINFO_PDU has a total lengh of 25 bits (GSM 04.04).

PDU Type Definition

PDU Name : SYSINFO1_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE1 n −> ms
GSM 04.08, 9.1.31

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
cchd	CCHD		cell channel description M OCTETSTRING [16]
rachcp	RACHCP		rach control parameters M OCTETSTRING [3]
si1roct	OCTETSTRING [1]		SI1 rest octets M

Detailed Comments : The message has an L2 pseudo length of 21 octets and a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO2_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE2 n −> ms
GSM 04.08, 9.1.32

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcchfl	NCD		bcch frequency list M OCTETSTRING [16]
nccp	NCCP		ncc permitted M OCTETSTRING [1]
rachcp	RACHCP		rach control parameters M OCTETSTRING [3]

Detailed Comments : The message has an L2 pseudo length of 22 octets and a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO2bis_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE2bis or 2ter n −> ms
GSM 04.08, 9.1.33, 9.1.33a

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
xbcchfl	NCD		extended bcch frequency list M OCTETSTRING [16]
rachcp	RACHCP		rach control parameters M for 2bis, N/A for 2ter OCTETSTRING [3]
si2bisroct	OCTETSTRING [1..4]		SI2bis rest octets M 1 octet for 2bis, 4 octets for 2ter

Detailed Comments : This PDU type is used both for SYSINFO2bis, as well as for 2ter.
The 2bis message has an L2 pseudo length of 21 octets and a total length of 23 octets.
The 2ter message has an L2 pseudo length of 18 octets and a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO2ter_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION 2ter n −> ms
GSM 04.08, 9.1.34

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
xbcchfl	NCD2		extended bcch frequency list M OCTETSTRING [16]
si2terroct	OCTETSTRING[4]		SI2ter rest octets 4 octets for 2ter

Detailed Comments : The 2ter message has an L2 pseudo length of 18 octets and a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO3_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE3 n –> ms
GSM 04.08, 9.1.34

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
ci	CI		cell identity M OCTETSTRING [2]
lai	LAI		location area identification M OCTETSTRING [5]
ccd	CCD		control channel description M OCTETSTRING [3]
co	CO		cell options/BCCH M, 1 octet
csp	CSP		cell selection parameters M OCTETSTRING [2]
rachcp	RACHCP		rach control parameters M OCTETSTRING [3]
si3roct	SI3RO		SI3 rest octets M

Detailed Comments : The message has an L2 pseudo length of 18 octets and a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO4_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE4 n −> ms
GSM 04.08, 9.1.35

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
lai	LAI		location area identification M OCTETSTRING [5]
csp	CSP		cell selection parameters M OCTETSTRING [2]
rachcp	RACHCP		rach control parameters M OCTETSTRING [3]
cbchd	CHD		cbch channel description M OCTETSTRING [4]
cbchma	MA		mobile allocation C OCTETSTRING [3..6]
si4roct	SI4RO		SI4 rest octets M

Detailed Comments : The message has a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO5_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE5 n −> ms
GSM 04.08, 9.1.36

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcchfl	NCD		bcch frequency list M OCTETSTRING [16]

Detailed Comments :

PDU Type Definition

PDU Name : SYSINFO5bis_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE5bis n −> ms
GSM 04.08, 9.1.38

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
xbcchfl	NCD		Extension of bcch frequency list M OCTETSTRING [16]

Detailed Comments :

PDU Type Definition

PDU Name : SYSINFO5ter_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE5ter n −> ms
GSM 04.08, 9.1.39

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
xbcchfl	NCD2		Extension of bcch frequency list M OCTETSTRING [16]

Detailed Comments :

PDU Type Definition

PDU Name : SYSINFO6_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE6 n −> ms
GSM 04.08, 9.1.38

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
ci	CI		cell identity M OCTETSTRING [2]
lai	LAI		location area identification M OCTETSTRING [5]
co	CO		cell options /SACCH M , 1 octet
nccp	NCCP		ncc permitted M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : SYSINFO7_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE7 n −> ms
GSM 04.08, 9.1.41

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
si7roct	SI7RO		SI7 rest octets M

Detailed Comments : The message has a total length of 23 octets.

PDU Type Definition

PDU Name : SYSINFO8_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : RR SYSTEM INFORMATION TYPE8 n −> ms
GSM 04.08, 9.1.42

Field Name	Field Type	Field Encoding	Comments
l2_pl	LENGTH		L2 pseudo length M OCTETSTRING [1]
ski	SKI		skip identifier M BITSTRING [4]
rrpd	PD		RR protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
si8roct	SI7RO		SI8 rest octets M

Detailed Comments : The message has a total length of 23 octets.

PDU Type Definition

PDU Name : ABRT_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM ABORT n −> ms
GSM 04.08, 9.2.8

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rejcau	REJCAU		reject cuase M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : AUTH_REJ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM AUTHENTICATION REJECT n −> ms
GSM 04.08, 9.2.1

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : AUTH_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM AUTHENTICATION REQUEST n −> ms
GSM 04.08, 9.2.2

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
cphksn	CPHKSN		ciphering key sequence number M BITSTRING [4]
rand	RAND		authentication parameter rand M BITSTRING [128]

Detailed Comments :

PDU Type Definition

PDU Name : AUTH_RES_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM AUTHENTICATION RESPONSE ms −> n
GSM 04.08, 9.2.3

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
sres	SRES		authentication response signature M OCTETSTRING [4]

Detailed Comments :

PDU Type Definition

PDU Name : CMRE_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM CM REESTABLISHMENT REQUEST ms −> n
GSM 04.08, 9.2.4

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
cphksn	CPHKSN		ciphering key sequence number M BITSTRING [4]
msclm	MSCLM2		mobile station classmark M OCTETSTRING [4]
mi	MI		mobility identity M OCTETSTRING [2..9]
lai	LAI		location area identification C OCTETSTRING [6]

Detailed Comments :

PDU Type Definition

PDU Name : CMS_ACP_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM CM SERVICE ACCEPT n −> ms
GSM 04.08, 9.2.5

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : CMS_REJ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM CM SERVICE REJECT n −> ms
GSM 04.08, 9.2.6

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
mmcau	REJCAU		Reject Cause M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : CMS_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM CM SERVICE REQUEST ms −> n
GSM 04.08, 9.2.9

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
cphksn	CPHKSN		ciphering key sequence number M BITSTRING [4]
svtype	CMSVTYPE		CM service type M BITSTRING [4]
msclm	MSCLM2		mobile station classmark M OCTETSTRING [4]
mi	MI		mobility identity M OCTETSTRING [2..9]

Detailed Comments :

PDU Type Definition

PDU Name : ID_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM IDENTITY REQUEST n −> ms
GSM 04.08, 9.2.10

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
shoct	SHOCT		spare half octet M BITSTRING [4]
idtype	IDTYPE		identity type M BITSTRING [4]

Detailed Comments :

PDU Type Definition

PDU Name : ID_RES_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM IDENTITY RESPONSE ms −> n
GSM 04.08, 9.2.11

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
mi	MI		mobility identity M OCTETSTRING [2..9]

Detailed Comments :

PDU Type Definition

PDU Name : IMSID_IN_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM IMSI DETACH INDICATION ms −> n
GSM 04.08, 9.2.12

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
msclm	MSCLM1		mobile station classmark M OCTETSTRING [1]
mi	MI		mobility identity M OCTETSTRING [2..9]

Detailed Comments :

PDU Type Definition

PDU Name : LUP_ACP_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM LOCATION UPDATING ACCEPT n −> ms
GSM 04.08, 9.2.13

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
lai	LAI		location area identification M OCTETSTRING [5]
mi	MI		mobility identity O OCTETSTRING [2..9]
fop	IEI_8		follow−on proceed O BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : LUP_ACP_PDU_ERR
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM LOCATION UPDATING ACCEPT n −> ms
GSM 04.08, 9.2.13

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
lai	LAI		location area identification C OCTETSTRING [5]
mi	MI		mobility identity M OCTETSTRING [2..9]
dupmi	MI		duplicated mobile identifier

Detailed Comments :

PDU Type Definition

PDU Name : LUP_REJ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM LOCATION UPDATING REJECT n −> ms
GSM 04.08, 9.2.14

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rejcau	REJCAU		reject cuase M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : LUP_RQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM LOCATION UPDATING REQUEST ms −> n
GSM 04.08, 9.2.15

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
cphksn	CPHKSN		ciphering key sequence number M BITSTRING [4]
lutype	LUT		location updating type M BITSTRING [4]
lai	LAI		location area identification C OCTETSTRING [6]
msclm	MSCLM1		mobile station classmark 1 M OCTETSTRING [4]
mi	MI		mobility identity M OCTETSTRING [2..9]

Detailed Comments :

PDU Type Definition

PDU Name : MMST_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM STATUS n <–> ms
GSM 04.08, 9.2.16

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
rejcau	REJCAU		reject cuase M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : TMSIRE_CMD_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM TMSI REALLOCATION COMMAND n –> ms
GSM 04.08, 9.2.17

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
lai	LAI		location area identification C OCTETSTRING [6]
mi	MI		mobility identity M OCTETSTRING [2..9]

Detailed Comments :

PDU Type Definition

PDU Name : TMSIRE_COM_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : MM TMSI REALLOCATION COMPLETE ms −> n
GSM 04.08, 9.2.18

Field Name	Field Type	Field Encoding	Comments
ski	SKI		skip identifier M BITSTRING [4]
mmpd	PD		MM protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : ALERT_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC ALERTING ms <−> n
GSM 04.08, 9.3.1

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
fie	FIE		facility O
pi	PI		progress indicator n −> ms O OCTETSTRING [4]
uu	UU		user−user O OCTETSTRING [3..35]
ssvi	SSVI		SS version indicator ms −> n O OCTETSTRING [2..3]

Detailed Comments :

PDU Type Definition

PDU Name : CALL_CO_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC CALL CONFIRMED ms −> n
GSM 04.08, 9.3.2

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
bcri	RPI		BC repeat indicator C BITSTRING [8]
bcap1	BCAP		bearer capab. O OCTETSTRING [3..10]
bcap2	BCAP		bearer capab. O
cau	CAU		cause O OCTETSTRING [4..32]
cccap	CCCAP		cc capabilities O

Detailed Comments :

PDU Type Definition

PDU Name : CALL_PROC_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC CALL PROCEEDING n −> ms
GSM 04.08, 9.3.3

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcri	RPI		BC repeat indicator C BITSTRING [8]
bcap1	BCAP		bearer capab. O OCTETSTRING [3..10]
bcap2	BCAP		bearer capab. O
fie	FIE		facility O
pi	PI		progress indicator O OCTETSTRING [4]

Detailed Comments :

PDU Type Definition

PDU Name : CONN_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC CONNECT n <−> ms
GSM 04.08, 9.3.5

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
fie	FIE		facility O
pi	PI		progress indicator n −> ms O OCTETSTRING [4]
cnn	CNN		connected number n −> ms O OCTETSTRING [3..14]
cns	CNS		connected subaddress O OCTETSTRING [2..23]
uu	UU		user−user information O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n O OCTETSTRING [2..3]

Detailed Comments :

PDU Type Definition

PDU Name : CONN_PDU_ERR
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC CONNECT n <−> ms
Used as invalid message

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
unknown	UNKWN		unknown IE

Detailed Comments :

PDU Type Definition

PDU Name : CONN_ACK_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC CONNECT ACKNOWLEDGE ms <−> n
GSM 04.08, 9.3.6

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : DISC_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC DISCONNECT ms <−> n (both directions)
GSM 04.08, 9.3.7

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
cau	CAU		cause M OCTETSTRING [4..32]
fie	FIE		facility O
pi	PI		progress indicator n −> ms O OCTETSTRING [4]
uu	UU		user−user information O OCTETSTRING [3..35]
ssvi	SSVI		SS version indicator O OCTETSTRING [3]

Detailed Comments :

PDU Type Definition

PDU Name : DISC_PDU_ERR
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments :

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
cau	CAU		cause M OCTETSTRING [4..32]
unknown	UNKWN		facility O

Detailed Comments :

PDU Type Definition

PDU Name : ESETUP_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC EMERGENCY SETUP ms −> n
GSM 04.08, 9.3.8

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcap	BCAP		bearer capab. O OCTETSTRING [3..10]

Detailed Comments :

PDU Type Definition

PDU Name : FAC_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC FACILITY n <−> ms
GSM 04.08, 9.3.9

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
cc_sspd	PD		CC or SS protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
fie	FIE		facility M
ssvi	SSVI		SS version ms −> n O OCTETSTRING [2..3]

Detailed Comments :

PDU Type Definition

PDU Name : HOLD_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC HOLD ms −> n
GSM 04.08, 9.3.10

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : HOLD_ACK_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC HOLD ACKNOWLEDGE n −> ms
GSM 04.08, 9.3.11

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : MODIFY_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC MODIFY ms <−> n
GSM 04.08, 9.3.13

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcap	BCAP		bearer capab. M OCTETSTRING [3..10]
llcmp	LLCMP		low layer compatib. O OCTETSTRING [2..15]
hlcmp	HLCMP		high layer compat. O OCTETSTRING [2..5]
rcsd	RCSD		reverse call setup direction O BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : MODIFY_COM_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC MODIFY COMPLETE ms <−> n
GSM 04.08, 9.3.14

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcap	BCAP		bearer capab. M OCTETSTRING [3..10]
llcmp	LLCMP		low layer compatib. O OCTETSTRING [2..15]
hlcmp	HLCMP		high layer compat. O OCTETSTRING [2..5]
rcsd	RCSD		reverse call setup direction O OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : MODIFY_REJ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC MODIFY REJECT ms <−> n
GSM 04.08, 9.3.15

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcap	BCAP		bearer capab. M OCTETSTRING [3..10]
cau	CAU		cause M OCTETSTRING [4..32]
llcmp	LLCMP		low layer compatib. O OCTETSTRING [2..15]
hlcmp	HLCMP		high layer compat. O OCTETSTRING [2..5]

Detailed Comments :

PDU Type Definition

PDU Name : NOTIFY_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC NOTIFY ms <−> n
GSM 04.08, 9.3.16

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
nti	NTI		notification indicator M OCTETSTRING [1]

Detailed Comments :

PDU Type Definition

PDU Name : REGISTER_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : SS REGISTER ms <−> n
GSM 04.80, 2.4

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
sspd	PD		SMS protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
fie	FIE		facility MV[2−?]
ssvi	SSVI		SS version indicator[3] ms −> n

Detailed Comments :

PDU Type Definition

PDU Name : PROG_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC PROGRESS n −> ms
GSM 04.08, 9.3.17

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
pi	PI		progress indicator n −> ms M OCTETSTRING [4]
uu	UU		user−user information O OCTETSTRING [3..35]

Detailed Comments :

PDU Type Definition

PDU Name : REL_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC RELEASE n <−> ms
GSM 04.08, 9.3.18

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
cau	CAU		cause O OCTETSTRING [4..32]
cau2	CAU		second cause
fie	FIE		facility O
uu	UU		user−user information O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n O OCTETSTRING [2..3]

Detailed Comments :

PDU Type Definition

PDU Name : REL_COM_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC or SS RELEASE COMPLETE n <−> ms
GSM 04.08, 9.3.19; GSM 04.80, 2.5

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
cc_sspd	PD		protocol discriminator M
mt	MT		message type M BITSTRING [8]
cau	CAU		cause O OCTETSTRING [4..32]
fie	FIE		facility O
uu	UU		user−user information O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n O OCTETSTRING [2..3]

Detailed Comments :

PDU Type Definition

PDU Name : SETUP_MO_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC SETUP n <− ms
GSM 04.08, 9.3.23.2

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcri	RPI		BC repeat indicator C BITSTRING [8]
bcap1	BCAP		bearer capab. n −> ms O; ms −> n M
bcap2	BCAP		bearer capab. O OCTETSTRING [3..10]
fie	FIE		facility O
cgps	CGPS		calling party subaddr. O OCTETSTRING [2..23]
cdpn	CDPN		called party BCD number n −> ms O; ms −> n M OCTETSTRING[2..23]
cdps	CDPS		called party subaddr. O OCTETSTRING [2..23]
llcri	RPI		LLC repeat indicator O OCTETSTRING [1]
llcmp1	LLCMP		low layer compatib.1 O OCTETSTRING [2..15]
llcmp2	LLCMP		low layer compatib.2 n −> ms C; ms −> n O
hlcri	RPI		HLC repeat indicator O OCTETSTRING [1]
hlcmp1	HLCMP		high layer compat.1 O OCTETSTRING [2..5]
hlcmp2	HLCMP		high layer compat. 2 n −> ms C; ms −> n O
uu	UU		user−user information O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n O OCTETSTRING [2..3]
clirsup	CLRSUP		CLIR suppression ms −> n C
clirinv	CLRINV		CLIR invocation ms −> n O
cccap	CCCAP		call control capabilities ms−>n O

Detailed Comments :

PDU Type Definition

PDU Name : SETUP_MT_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC SETUP n −> ms
GSM 04.08, 9.3.23.1

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcri	RPI		BC repeat indicator C BITSTRING [8]
bcap1	BCAP		bearer capab. n −> ms O; ms −> n M
bcap2	BCAP		bearer capab. O OCTETSTRING [3..10]
fie	FIE		facility O
pi	PI		progress indicator n −> ms O OCTETSTRING [4]
sig	SIGNAL		signal (n −>ms) O OCTETSTRING[2..34]
cgpn	CGPN		calling party BCD number (n −>ms) O OCTETSTRING [2..24]
cgps	CGPS		calling party subaddr. O OCTETSTRING [2..23]
cdpn	CDPN		called party BCD number n −> ms O; ms −> n M OCTETSTRING[2..23]
cdps	CDPS		called party subaddr. O OCTETSTRING [2..23]
llcri	RPI		LLC repeat indicator O OCTETSTRING [1]
llcmp1	LLCMP		low layer compatib.1 O OCTETSTRING [2..15]
llcmp2	LLCMP		low layer compatib.2 n −> ms C; ms −> n O
hlcri	RPI		HLC repeat indicator O OCTETSTRING [1]
hlcmp1	HLCMP		high layer compat.1 O OCTETSTRING [2..5]
hlcmp2	HLCMP		high layer compat. 2 n −> ms C; ms −> n O
uu	UU		user−user information O OCTETSTRING [3..35]

Detailed Comments :

PDU Type Definition

PDU Name : SMSCB_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : SMSCB message
GSM 03.41, 9.3

Field Name	Field Type	Field Encoding	Comments
blocktype	BLOCKTYPE		
serial_number	SERIAL_NUMBER		Only for the first block
message_id	OCTETSTRING[2]		Only for the first block
dcs	TPDCS		Only for the first block
page_param	BITSTRING[8]		Only for the first block
message_contents	OCTETSTRING		

Detailed Comments :

PDU Type Definition

PDU Name : START_DTMF_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC START DTMF ms −> n
GSM 04.08, 9.3.24

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
kpf	KPF		keypad facility M OCTETSTRING [2]

Detailed Comments :

PDU Type Definition

PDU Name : START_DTMF_ACK_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC START DTMF ACKNOWLEDGE n −> ms
GSM 04.08, 9.3.25

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
kpf	KPF		keypad facility M OCTETSTRING [2]

Detailed Comments :

PDU Type Definition

PDU Name : START_DTMF_REJ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC START DTMF REJECT n −> ms
GSM 04.08, 9.3.26

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
cau	CAU		cause M OCTETSTRING [4..32]

Detailed Comments :

PDU Type Definition

PDU Name : CCST_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC STATUS n <−> ms
GSM 04.08, 9.3.27

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]
cau	CAU		cause M OCTETSTRING [4..32]
cst	CST		call state M OCTETSTRING[1]
acst	ACST		auxiliary status O OCTETSTRING[3]

Detailed Comments :

PDU Type Definition

PDU Name : CCST_ENQ_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC STATUS ENQUIRY ms <−> n
GSM 04.08, 9.3.28

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : STOP_DTMF_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC STOP DTMF ms −> n
GSM 04.08, 9.3.29

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : STOP_DTMF_ACK_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CC STOP DTMF acknowledge n −> ms
GSM 04.08, 9.3.30

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
ccpd	PD		CC protocol discriminator M
mt	MT		message type M BITSTRING [8]

Detailed Comments :

PDU Type Definition

PDU Name : CP_DATA_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : SMS CP DATA ms <−> n
GSM 04.11, 7.2.1

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
smspd	PD		SMS protocol discriminator M BITSTRING [4], ’1001’B
mt	MT		message type M BITSTRING [8], ’0000 0001’
CPdata	CPDATA		CP−User data element M OCTETSTRING [1..249]

Detailed Comments : CPDATA contains RPDU − RP_ACK or RP_ERROR, either returning to the MS in case of MO, or reporting the outcome of a MT messaging attempt in case of MT .

PDU Type Definition

PDU Name : CPDATA_ACK_PDU
PCO Type : SAP0_3
Encoding Rule Name :
Encoding Variation :
Comments : CP DATA ACKNOWLEDGE ms <−> n
GSM 04.11, 7.2.2

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
smspd	PD		SMS protocol discriminator M
mt	MT		message type M BITSTRING [8], ’0000 0100’

Detailed Comments :

PDU Type Definition

PDU Name : CPERR_PDU

PCO Type : SAP0_3

Encoding Rule Name :

Encoding Variation :

Comments : CP_ERROR n <−> ms
GSM 04.11, 7.2.3

Field Name	Field Type	Field Encoding	Comments
ti	TI		transaction identifier M BITSTRING [4]
smspd	PD		SMS protocol discriminator M
mt	MT		message type M BITSTRING [8], '0001 0000'
cp_cause	CP_CAU		cp cause (RPDU) M OCTETSTRING[1..255]

Detailed Comments : CP_UDAT contains RPDU, RP_ACK or RP_ERROR, either returning to the MS in case of MO, or reporting the outcome of a MT messaging attempt in case of MT .

III

Constraints Part

Structured Type Constraint Declaration

Constraint Name : EFR_Bcap_Speech_MO2(rcr : B_2)
Structured Type : BCAP
Derivation Path :
Encoding Variation :
Comments : speech bearer capability for direction ms−>n. An EFR ms MUST send two 3a_etc (speech preference) octets.

Element Name	Element Value	Element Encoding	Comments
iei	'00000100'B		IEI
iel	'03'O		length of the IE contents
extb3	'0'B		extension
rchr	rcr		radio channel reqirement
cs	'0'B		coding standard : GSM
tm	'0'B		circuit mode
itc	'000'B		speech
oct_3a_etc_1	EFR_Bcap_3a_etc		
oct_3a_etc_2	EFR_Bcap_3a_etc		
oct_3a_etc_3	−		
oct_3a_etc_4	−		
oct_3a_etc_5	−		
oct_3a_etc_6	−		
extb4	−		
spb	−		
strc	−		
dplxm	−		
config	−		
nirr	−		
est	−		
extb5	−		
accid	−		
ra	−		
sacp	−		
extb6	−		
l1id	−		
uil1	−		
sb	−		
extb6a	−		
nsb	−		
nb	−		
ndb	−		
ur	−		
extb6b	−		
ir	−		
nictx	−		
nicrx	−		
pi	−		
extb6c	−		
ce	−		
modemt	−		
extb7	−		

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
l2id	–		
uil2	–		
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : EFR_Bcap_Speech_MO3(rcr : B_2)
Structured Type : BCAP
Derivation Path :
Encoding Variation :
Comments : speech bearer capability for direction ms−>n. An EFR ms must send THREE 3a_etc (speech preference) octets.

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	’04’O		length of the IE contents
extb3	’0’B		extension
rchr	rcr		radio channel reqirement
cs	’0’B		coding standard : GSM
tm	’0’B		circuit mode
itc	’000’B		speech
oct_3a_etc_1	EFR_Bcap_3a_etc		
oct_3a_etc_2	EFR_Bcap_3a_etc		
oct_3a_etc_3	EFR_Bcap_3a_etc		
oct_3a_etc_4	−		
oct_3a_etc_5	−		
oct_3a_etc_6	−		
extb4	−		
spb	−		
strc	−		
dplxm	−		
config	−		
nirr	−		
est	−		
extb5	−		
accid	−		
ra	−		
sacp	−		
extb6	−		
l1id	−		
uil1	−		
sb	−		
extb6a	−		
nsb	−		
nb	−		
ndb	−		
ur	−		
extb6b	−		
ir	−		
nictx	−		
nicrx	−		
pi	−		
extb6c	−		
ce	−		
modemt	−		
extb7	−		

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
l2id	–		
uil2	–		
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : EFR_Bcap_Speech_MT
Structured Type : BCAP
Derivation Path :
Encoding Variation :
Comments : speech bearer capability for direction n−>ms

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	’01’O		length of the IE contents
extb3	’1’B		no extension
rchr	’01’B		spare bits for n−> ms
cs	’0’B		coding standard : GSM
tm	’0’B		circuit mode
itc	’000’B		speech
oct_3a_etc_1	−		
oct_3a_etc_2	−		
oct_3a_etc_3	−		
oct_3a_etc_4	−		
oct_3a_etc_5	−		
oct_3a_etc_6	−		
extb4	−		
spb	−		
strc	−		
dplxm	−		
config	−		
nirr	−		
est	−		
extb5	−		
accid	−		
ra	−		
sacp	−		
extb6	−		
l1id	−		
uil1	−		
sb	−		
extb6a	−		
nsb	−		
nb	−		
ndb	−		
ur	−		
extb6b	−		
ir	−		
nictx	−		
nicrx	−		
pi	−		
extb6c	−		
ce	−		
modemt	−		
extb7	−		

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
l2id	–		
uil2	–		
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : EFR_Bcap_Speech_MT2
Structured Type : BCAP
Derivation Path :
Encoding Variation :
Comments : speech bearer capability for direction n−>ms

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	’01’O		length of the IE contents
extb3	’0’B		extension
rchr	’01’B		spare bits for n−> ms
cs	’0’B		coding standard : GSM
tm	’0’B		circuit mode
itc	’000’B		speech
oct_3a_etc_1	EFR_Bcap_3a_etc		
oct_3a_etc_2	EFR_Bcap_3a_etc		
oct_3a_etc_3	−		
oct_3a_etc_4	−		
oct_3a_etc_5	−		
oct_3a_etc_6	−		
extb4	−		
spb	−		
strc	−		
dplxm	−		
config	−		
nirr	−		
est	−		
extb5	−		
accid	−		
ra	−		
sacp	−		
extb6	−		
l1id	−		
uil1	−		
sb	−		
extb6a	−		
nsb	−		
nb	−		
ndb	−		
ur	−		
extb6b	−		
ir	−		
nictx	−		
nicrx	−		
pi	−		
extb6c	−		
ce	−		
modemt	−		
extb7	−		

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
l2id	–		
uil2	–		
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : EFR_Bcap_Speech_MT3
Structured Type : BCAP
Derivation Path :
Encoding Variation :
Comments : speech bearer capability for direction n−>ms

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	’01’O		length of the IE contents
extb3	’0’B		extension
rchr	’01’B		spare bits for n−> ms
cs	’0’B		coding standard : GSM
tm	’0’B		circuit mode
itc	’000’B		speech
oct_3a_etc_1	EFR_Bcap_3a_etc		
oct_3a_etc_2	EFR_Bcap_3a_etc		
oct_3a_etc_3	EFR_Bcap_3a_etc		
oct_3a_etc_4	−		
oct_3a_etc_5	−		
oct_3a_etc_6	−		
extb4	−		
spb	−		
strc	−		
dplxm	−		
config	−		
nirr	−		
est	−		
extb5	−		
accid	−		
ra	−		
sacp	−		
extb6	−		
l1id	−		
uil1	−		
sb	−		
extb6a	−		
nsb	−		
nb	−		
ndb	−		
ur	−		
extb6b	−		
ir	−		
nictx	−		
nicrx	−		
pi	−		
extb6c	−		
ce	−		
modemt	−		
extb7	−		

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
l2id	–		
uil2	–		
Detailed Comments :			

Structured Type Constraint Declaration			
Constraint Name : EFR_Bcap_3a_etc			
Structured Type : EFR_BCAP_3A_ETC			
Derivation Path :			
Encoding Variation :			
Comments : speech bearer capability for direction ms−>n.			
Element Name	**Element Value**	**Element Encoding**	**Comments**
extb	’?’B		
cod	’0’B		
spr2	’00’B		
svi	’0???’B		Must be full/half rate speech v1/v2 or v3
Detailed Comments :			

Structured Type Constraint Declaration			
Constraint Name : BcchFreqLst_Omit			
Structured Type : NCD			
Derivation Path :			
Encoding Variation :			
Comments : Omit Bcch Frequency list (Neighbour cells description)			
Element Name	**Element Value**	**Element Encoding**	**Comments**
rfl2	–		
extind	–		
baind	–		
rfl4	–		
rfl	–		
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_01
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {10, 20, 40, 80, 90, 100, 110, 120}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0 format
extind	'0'B		complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'0000'B		
rfl	'802008020080000000080000080200'O		ARFCN: 10, 20, 40, 80, 90, 100, 110, 120

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_02
Structured Type : NCD
Derivation Path : BcchFreqLst_01.
Encoding Variation :
Comments : Alternative neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900. These are ARFCNs 15, 85, 95, 105, 115, and 122

Element Name	Element Value	Element Encoding	Comments
rfl4	'0010'B		
rfl	'040100401000000000000000004000'O		ARFCN: 15, 85, 95, 105, 115, 122

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_03
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 for DCS1800 for cell B with the ARFCN list = {520, 600, 700, 780, 810, 870}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		range 512 format
extind	'0'B		extension indication
baind	'0'B		bcch allocation sequence number indication
rfl4	'1001'B		
rfl	'04412C168E44000000000000000000'O		ARFCN: 520, 600, 700, 780, 810, 870

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_04
Structured Type : NCD
Derivation Path : BcchFreqLst_01.
Encoding Variation :
Comments : Alternative neighbour cells description for SYSTEM INFORMATION 2 and 5 for DCS1800.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		range 512 format
extind	'0'B		extension indication
baind	'0'B		bcch allocation sequence number indication
rfl4	'1001'B		
rfl	'09412C168E44000000000 000000000'O		ARFCN: 530, 610, 710, 790, 820, 880

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_45
Structured Type : NCD
Derivation Path : BcchFreqLst_01.
Encoding Variation :
Comments : Default neighbour cells description of cell B for SYSTEM INFORMATION 2 and 5 under GSM900.

Element Name	Element Value	Element Encoding	Comments
rfl	'802008020080000000000 000000200'O		ARFCN: 10, 80, 90, 100, 110, 120

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_47
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis and 5bis in cell B under EGSMwith the ARFCN list = {1005, 1010, 1015}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		256 format
extind	'1'B		not complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'1101'B		
rfl	'F68AEC000000000000000 000000000'O		ARFCN: 1005, 1010, 1015

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_48
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 for DCS1800 for cell B in RR, MM and SMS test cases with the ARFCN list = {520, 590, 600, 700,747, 764, 780, 810, 870}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		range 512 format
extind	'0'B		extension indication
baind	'0'B		bcch allocation sequence number indication
rfl4	'1001'B		
rfl	'043D170B8A62C8F23000 0000000000'O		ARFCN: 520, 590, 600, 700,747, 764, 780, 810, 870

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CallCntrlCap
Structured Type : CCCAP
Derivation Path :
Encoding Variation :
Comments : CC state 'st'

Element Name	Element Value	Element Encoding	Comments
iei	'00010101'B		
iel	'01'O		
spr	'0000000'B		spare bits '0000000'B
dtmf	TSPX_DTMF		indication of supporting DTMF

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Cause_01
Structured Type : CAU
Derivation Path :
Encoding Variation :
Comments : cause information element with cause value = #16, location = user.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'02'O		
extb3	'1'B		no extension
cs	'11'B		standard coding
spb	'0'B		
location	'0000'B		user
extb3a	–		no octet 3a
rec	–		
extb4	'1'B		no extension
cau_class	'001'B		normal event
cau_va	'0000'B		normal call clearing
cau_di	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Cause_Def
Structured Type : CAU
Derivation Path :
Encoding Variation :
Comments : cause information element with any valid cause value, any valid location.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	?		
extb3	'1'B		no extension
cs	'11'B		GSM coding
spb	'0'B		
location	('0000'B, '0001'B, '0010'B, '0011'B, '0100'B, '0101'B, '0111'B, '1010'B)		
extb3a	–		no octet 3a
rec	–		
extb4	?		no extension
cau_class	?		
cau_va	?		
cau_di	*		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_Omit
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Omit descriptor

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_02
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell A in RR testing of GSM 900 with the ARFCN_list={20,30,50,70}.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'00000000000000020000200020080000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_03
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell A in RR testing of DCS1800 with the ARFCN list = {590, 650, 750, 850}..

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'892728271900000000000000000000'O		ARFCNs : 590, 650, 750, 850 512 range

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_04
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell B in RR testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'00000000000000000000 00000000200'O		ARFCNs : 10

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_20_Aman
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Cell Channel Description for any cell in RR testing of GSM 900 HO cases with the ARFCN_list={10,17,20,26,34,42,45,46,52,59,66,73,74,75,76,108,114}.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'0002080000000F0204083 20202090200'O		bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_201_Ad
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Cell Channel Description for cell A in RR testing of DCS1800 HO cases with the ARFCN_list={734,741,747,754,759,762,766,767,773,775,779,782,791,798,829,832,844} using 256 format.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'8B6F14F32FC602C59EFF A54999400000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_202_Adman
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Cell Channel Description for cell A in RR testing of DCS1800 HO cases with the Complete Cell Allocation of Cell A in HO cases using 512 format. ARFCN_list={734,741,747,754,759,762,766,767,773,775,779,782,791,798,829,832,844}

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'896F0A7CC5FC700A8B9F 7FF452463340'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_20_Bman
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell B in RR testing of GSM 900 HO cases with the complete Cell Allocation ARFCN_list={14,18,22,24,30,31,38,40,60,66,73,74,75,76,108,114}.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'0002080000000F0208000 0A060A22000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_201_Bd
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Cell Channel Description for cell B in RR testing of DCS1800 HO cases with the ARFCN_list={739,743,746,749,756,758,761,764,771,779,782,791,798,829,832,844} using 256 format.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'8B71907137B602C5DEF7 A348D8000000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_202_Bd
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Cell Channel Description for cell B in RR testing of DCS1800 HO cases with the ARFCN_list={739,743,746,749,756,758,761,764,771,779,782,791,798,829,832,844} using 512 format.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'8971883C46FB700A8BDF 7DF432433000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellDescrp(ncc1, bcc1 : B_3; par_arfcn: INTEGER)
Structured Type : CD
Derivation Path :
Encoding Variation :
Comments : BCCH channel for Cell A

Element Name	Element Value	Element Encoding	Comments
bcch_arfcn_h	OC_MostBits(INT_TO_BIT(par_arfcn, 10), 2)		bcch arfcn high part
ncc	ncc1		PLMN colour code
bcc	bcc1		BS colour code
bcch_arfcn_l	OC_LeastBits(INT_TO_BIT(par_arfcn, 10), 8)		bcch arfcn low part

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellOpt_01
Structured Type : CO
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
sprb	'0'B		
pwrc	'0'B		power control indicator not set
dtx	'10'B		MS shall not use uplink discontinuous transmission.
rlt	'0001'B		radio link timeout, 8 SACCH blocks

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellSelPara(crh, mtmc : INTEGER; neci : B_1)
Structured Type : CSP
Derivation Path :
Encoding Variation :
Comments : Default value.

Element Name	Element Value	Element Encoding	Comments
crh	INT_TO_BIT(crh, 3)		
mtmc	INT_TO_BIT(mtmc, 5)		
acs	'0'B		no addtional cell parameters are presented in SI msg 7 and 8
neci	neci		new establishment causes are not supported
ram	'000000'B		minimum

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChDescrp_fh(par_chtype: CH_TDMA; ts_ccch: SN; tsc : TSC; maio : MAIO; hsn : HSN)
Structured Type : CHD
Derivation Path :
Encoding Variation :
Comments : A derived constraint for <<Channel description >>ie with TCH/F freq. hopping in any CELL.

Element Name	Element Value	Element Encoding	Comments
iei	–		'01100100'B
cht_schn	par_chtype		
tn	ts_ccch		
tsc	tsc		
hch	'1'B		hopping
maio	maio		
hsn	hsn		
spr	–		not used
arfcn	–		not used

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChDescrp_nfh(type : CH_TDMA; slot : SN; tsc : TSC; par_arfcn: INTEGER)
Structured Type : CHD
Derivation Path :
Encoding Variation :
Comments : Channel description for CC testing, non hopping, no starting time.

Element Name	Element Value	Element Encoding	Comments
iei	–		
cht_schn	type		channel type and TDMA offset
tn	slot		Time slot
tsc	tsc		training sequence code
hch	'0'B		
maio	–		
hsn	–		
spr	'00'B		
arfcn	INT_TO_BIT(par_arfcn, 10)		ARFCN

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod(mode : B_8)
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
mode	mode		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_omit
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments : parametrised mode.

Element Name	Element Value	Element Encoding	Comments
iei	–		
mode	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_EFR2
Structured Type : CHMOD
Derivation Path : ChMod_sign.
Encoding Variation :
Comments : enhanced full/half rate speech − version 2.

Element Name	Element Value	Element Encoding	Comments
mode	C_ChMod_r2		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_EFR2_iei
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments : Version 2 full rate speech.

Element Name	Element Value	Element Encoding	Comments
iei	’01100011’B		
mode	C_ChMod_r2		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_EFR3
Structured Type : CHMOD
Derivation Path : ChMod_sign.
Encoding Variation :
Comments : enhanced full/half rate speech − version 3.

Element Name	Element Value	Element Encoding	Comments
mode	C_ChMod_r3		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_sign
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments : signalling only

Element Name	Element Value	Element Encoding	Comments
iei	−		
mode	C_ChMod_s		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_sign_iei
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments : signalling only

Element Name	Element Value	Element Encoding	Comments
iei	'01100011'B		
mode	C_ChMod_s		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_speech
Structured Type : CHMOD
Derivation Path : ChMod_sign.
Encoding Variation :
Comments : speech full or half rate.

Element Name	Element Value	Element Encoding	Comments
mode	C_ChMod_r		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_speech_iei
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments : speech full or half rate.

Element Name	Element Value	Element Encoding	Comments
iei	'01100011'B		
mode	C_ChMod_r		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_12k
Structured Type : CHMOD
Derivation Path : ChMod_sign.
Encoding Variation :
Comments : data 9.6 kb/s full rate, (12.0 kb/s air interface)

Element Name	Element Value	Element Encoding	Comments
mode	C_ChMod_12k		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_6k
Structured Type : CHMOD
Derivation Path : ChMod_sign.
Encoding Variation :
Comments : data 4.8 kb/s full rate, (6.0 kb/s air interface)

Element Name	Element Value	Element Encoding	Comments
mode	C_ChMod_6k		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChMod_3k
Structured Type : CHMOD
Derivation Path : ChMod_sign.
Encoding Variation :
Comments : data 2.4 kb/s full rate, (3.6 kb/s air interface)

Element Name	Element Value	Element Encoding	Comments
mode	C_ChMod_3k		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Chneed_01
Structured Type : CHNEED
Derivation Path :
Encoding Variation :
Comments : any channel

Element Name	Element Value	Element Encoding	Comments
ch2	'00'B		
ch1	'00'B		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CiphRes_01
Structured Type : CPH_RES
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
sprb	'000'B		spare bits
cr	'0'B		IMEISV not included

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ClassMark2
Structured Type : MSCLM2
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iel	'03'O		length
spr1	'0'B		spare bit '0'B
rl	TSPX_RevLevel		revision level
spr2	TSPX_ESIND		early ClassMark sending indication
a5_1	TSPX_CiphAlgA5_1		a5/1 algorithm supported
rfpc	TSPX_RfPwrCap		rf power capability
spr3	'0'B		spare bit '0'B
psc	TSPX_pSyncCap		pseudo synchronisation capability
sssi	TSPX_SSscrnInd		SS screen indicator
smc	TSPX_SMCap		short message capability
spr4	'00'B		spare bit '00'B
fc	TSPX_frqCap		frequency capability
cm3	TSPX_ClsMk3		classmark 3 indicator
spr5	'00000'B		spare bit '00000'B
a5_3	TSPX_CiphAlgA5_3		a5/3 algorithm supported
a5_2	TSPX_CiphAlgA5_2		a5/2 algorithm supported

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CntrlChDscrp(att: INTEGER; babr, cch_con, bpm : B_3; timer : OCTETSTRING)
Structured Type : CCD
Derivation Path :
Encoding Variation :
Comments : Generic Control channel descriptor.

Element Name	Element Value	Element Encoding	Comments
spr1	'0'B		
att	INT_TO_BIT(att,1)		
babr	babr		
ccch_con	cch_con		
spr2	'00000'B		
bpm	bpm		
t3212	timer		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CphKeySN_01(cksn : B_3)
Structured Type : CPHKSN
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
sprb	'0'B		spare bit '0'B
ks	cksn		default key sequence number

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CphKeySN_07(par: BITSTRING)
Structured Type : CPHKSN
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
sprb	'0'B		spare bit '0'B
ks	par		default key sequence number

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CphMod_omit
Structured Type : CPHMS
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
algid	–		
sc	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CphMod_01
Structured Type : CPHMS
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
algid	TSPX_CphAlgDef		algorithm identifier
sc	’1’B		starting ciphering

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CphMod_02
Structured Type : CPHMS
Derivation Path : CphMod_01.
Encoding Variation :
Comments : No ciphering

Element Name	Element Value	Element Encoding	Comments
sc	’0’B		no ciphering

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqBCCH(arfcn : INTEGER)
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Broadcast channel.

Element Name	Element Value	Element Encoding	Comments
hch	’0’B		non hopping
maio	−		mobile allocation index offset − hopping, if hch = 1, otherwise −
hsn	−		hopping sequence number − hopping, if hch = 1, otherwise −
spr	’00’B		’00’B − non hopping, if hch = 0, otherwise −
arfcn	INT_TO_BIT(arfcn, 10)		ARFCN
maclength	’00’O		’00’O for non hopping
mac_8n	−		mac64 − mac57
mac_7n	−		mac56 − mac49
mac_6n	−		mac48 − mac41
mac_5n	−		mac40 − mac33
mac_4n	−		mac32 − mac25
mac_3n	−		mac24 − mac17
mac_2n	−		mac16 − mac9
mac_1n	−		mac8 − mac1
flst	−		frequency list for hopping

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Freqchseq_omit
Structured Type : FRQCHS
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	−		
larfcn	−		
incs	−		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Freqchseq_03
Structured Type : FRQCHS
Derivation Path :
Encoding Variation :
Comments : Frequency channel sequence with the List_ARFCN={14,114}

Element Name	Element Value	Element Encoding	Comments
iei	'01101001'B		
larfcn	'0E'O		
incs	'000000A000000000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Freqchseq_06
Structured Type : FRQCHS
Derivation Path :
Encoding Variation :
Comments : Frequency channel sequence with the
List_ARFCN={14,18,22,24,30,31,38,53,66,73,74,75,76,108,114}
see Frql_20_B5

Element Name	Element Value	Element Encoding	Comments
iei	'01101001'B		
larfcn	'0E'O		
incs	'442617FD71110026'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Freqchseq_22
Structured Type : FRQCHS
Derivation Path :
Encoding Variation :
Comments : Frequency channel sequence with the List_ARFCN={10,17,20,26,59,66,73,74,75,76,108,114}

Element Name	Element Value	Element Encoding	Comments
iei	'01101001'B		
larfcn	'0A'O		
incs	'7360037711100260'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCH(arfcn : INTEGER)
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Default Traffic channel non frequency hopping

Element Name	Element Value	Element Encoding	Comments
hch	'0'B		non hopping
maio	–		
hsn	–		
spr	'00'B		'00'B for no hopping
arfcn	INT_TO_BIT(arfcn, 10)		ARFCN
maclength	'00'O		'00'O for non hopping
mac_8n	–		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCH_omit
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Omitted FRQPARA when not needed in parameters list.

Element Name	Element Value	Element Encoding	Comments
hch	–		
maio	–		
hsn	–		
spr	–		
arfcn	–		
maclength	–		
mac_8n	–		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHa_hof1
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping traffic channel of cell A for GSM.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 12), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'02'O		
mac_8n	'00001111'B		(75,76,108,114)
mac_7n	'11111111'B		(10,17,20,26,59,66,73,74)
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A		{10,17,20,26,59,66,73,74,75,76,108,114}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHa_hof1d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping channel of cell A for DCS1800

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 9), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'02'O		
mac_8n	'00000001'B		(844)
mac_7n	'11111111'B		(747, 775, 779, 782, 791, 798, 829, 832)
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_Ad		{747, 775, 779, 782, 791, 798, 829, 832, 844}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHa_hof_ind3
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping traffic channel of cell A for GSM which indicates three frequences (73, 74, 75) used in TC_26_12_2_2, M=7.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 3), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'03'O		
mac_8n	'00000000'B		
mac_7n	'00111000'B		75, 74, 73
mac_6n	'00000000'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A0		10, 17, 20, 26, 34, 42, 45, 46, 52, 59, 66, 73, 74, 75, 76, 108, 114

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHa_hof_ind3d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping channel of cell A for DCS1800 which indicates only 3 frequencies (734, 762, 791) used in TC_26_12_2_2, M=7

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 3), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'03'O		
mac_8n	'00000000'B		
mac_7n	'00010000'B		(791)
mac_6n	'00100001'B		(762, 734)
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A0d		{734, 741, 747, 754, 759, 762, 766, 767, 773, 775, 779, 782, 791, 798, 829, 832, 844}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHa_hof_ind5
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping traffic channel of cell A for GSM which indicates five frequences (17, 34, 46, 52, 73) used in TC_26_12_2_2, M=6.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 5), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'03'O		
mac_8n	'00000000'B		
mac_7n	'00001001'B		73,52
mac_6n	'10010010'B		46, 34, 17
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A0		10, 17, 20, 26, 34, 42, 45, 46, 52, 59, 66, 73, 74, 75, 76, 108, 114

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHa_hof_ind5d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping channel of cell A for DCS1800 which indicates only 5 frequencies (734, 741, 762, 791, 832) used in TC_26_12_2_2, M=6

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 5), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'03'O		
mac_8n	'00000000'B		
mac_7n	'10010000'B		(832, 791)
mac_6n	'00100011'B		(762, 741, 734)
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A0d		{734, 741, 747, 754, 759, 762, 766, 767, 773, 775, 779, 782, 791, 798, 829, 832, 844}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof6
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 2), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'02'O		
mac_8n	'10000000'B		(114)
mac_7n	'00000001'B		(14)
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_B0		{14,18,22,24,30,31,38,40,60,66,73,74,75,76,108,114}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof6d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 3), 6)		
hsn	'000000'B		
spr	–		
arfcn	–		
maclength	'01'O		
mac_8n	'00000111'B		(764, 779, 782)
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_B10d		(764,779,782)

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof1
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Broadcast channel of cell B.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 12), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'02'O		
mac_8n	'00001111'B		(75,76,108,114)
mac_7n	'11111111'B		(14,18,22,24,60,66,73,74)
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_B3		{14,18,22,24,60,66,73,74,75,76,108,114}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof1d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Broadcast channel of cell B for hopping DCS1800

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 12), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'02'O		
mac_8n	'00001111'B		(798, 829, 832, 844)
mac_7n	'11111111'B		(749, 758, 761, 764, 771, 779, 782, 791)
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_B3d		{749, 758, 761, 764, 771, 779, 782, 791, 798, 829, 832, 844}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof3
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
hch	’1’B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 15), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	−		
arfcn	−		
maclength	’02’O		
mac_8n	’01111111’B		(66, 73, 74, 75, 76, 108, 114)
mac_7n	’11111111’B		(14, 18, 22, 24, 30, 31, 38, 53)
mac_6n	−		mac48 − mac41
mac_5n	−		mac40 − mac33
mac_4n	−		mac32 − mac25
mac_3n	−		mac24 − mac17
mac_2n	−		mac16 − mac9
mac_1n	−		mac8 − mac1
flst	Frql_20_B5		frequency list, after time for cell B {14,18,22,24,30,31,38,53,66,73,74,75,76,108,114}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof3d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 11), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	−		
arfcn	−		
maclength	'02'O		
mac_8n	'00000111'B		(829,832,844)
mac_7n	'11111111'B		(756,758,761,771,779,782, 791,798)
mac_6n	−		mac48 − mac41
mac_5n	−		mac40 − mac33
mac_4n	−		mac32 − mac25
mac_3n	−		mac24 − mac17
mac_2n	−		mac16 − mac9
mac_1n	−		mac8 − mac1
flst	Frql_20_B5d		frequency list, after time for cell B (756, 758, 761, 771, 779, 782, 791, 798, 829, 832, 844)

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof_ind8
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping traffic channel (TC_26_12_2_2, M = 6 and 10)

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 8), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	−		
arfcn	−		
maclength	'01'O		
mac_8n	'11111111'B		mac64 − mac57
mac_7n	−		mac56 − mac49
mac_6n	−		mac48 − mac41
mac_5n	−		mac40 − mac33
mac_4n	−		mac32 − mac25
mac_3n	−		mac24 − mac17
mac_2n	−		mac16 − mac9
mac_1n	−		mac8 − mac1
flst	Frql_ind8B		frequency list after time for cell B that uses bit map 0 to allocate the following 8 frequencies: (14, 18, 24, 30, 38, 60, 73, 74)

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCHb_hof_ind8d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Broadcast channel of cell B for hopping DCS1800 (TC_26_12_2_2, M=6 and 10)

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 8), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'01'O		
mac_8n	'11111111'B		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_ind8Bd		frequency list that uses range 256 to alocate the following 8 frequencies: (739, 743, 746, 749, 761, 771, 782, 791)

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqSDCCHa_hof1
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : hopping with {73,74,75} of cell allocation A.

Element Name	Element Value	Element Encoding	Comments
hch	’1’B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 3), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	’03’O		’00O for non hopping
mac_8n	’00000000’B		
mac_7n	’00111000’B		(73,74,75)
mac_6n	’00000000’B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A0		{10,17,20,26,34,42,45,46,52,59,66,73,74,75,76,108,114}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqSDCCHa_hof1d
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : a hopping channel of cell A with {773,775,779} of cell allocation A.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 3), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'03'O		
mac_8n	'00000000'B		
mac_7n	'00000111'B		(773, 775, 779)
mac_6n	'00000000'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_A0d		{734,741,747,754,759,762, 766,767,773,775,779,782,7 91,798,829,832,844}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_omit
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	–		
fl	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_A
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time used in HO_case in cell A.
List_ARFCN={10,17,20,26,59,66,73,74,75,76,108,114}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0002080000000F0204000 00002090200'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_A0
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Complete Cell Allocation of cell A in HO cases.
ARFCN_list={10,17,20,26,34,42,45,46,52,59,66,73,74,75,76,108,114}.

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0002080000000F0204083 20202090200'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_A0d
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Complete Cell Allocation of Cell A in HO cases using 512 format.
frequency list, after time for a target cell used in HO_case in cell A for DCS1800.
List_ARFCN={734,741,747,754,759,762,766,767,773,775,779,782,791,798,829,832,844}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		
fl	'896F0A7CC5FC700A8B9F 7FF452463340'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_Ad
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list used in HO_case in cell A for DCS1800 using 256 format for the List_ARFCN={747, 775, 779, 782, 791, 798, 829, 832, 844}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'09'O		
fl	'8B7599F045EFA499C0'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frqls_20_Ad
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list used in HO_case in cell A for DCS1800 using 256 format for the List_ARFCN={747, 775, 779, 782, 791, 798, 829, 832, 844}

Element Name	Element Value	Element Encoding	Comments
iei	'00000010'B		
iel	–		
fl	'8B7599F045EFA499C0'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B0
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Complete Cell Allocation of cell B in HO cases.
all of the CA: ARFCN_list={14,18,22,24,30,31,38,40,60,66,73,74,75,76,108,114}.

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0002080000000F02080000A060A22000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B1
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time for cell B.
List_ARFCN={14,18,22,24,30,31,38,53,66,73,74,75,76,108,114}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0002080000000F0200100 02060222000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B3
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time for cell B. List_ARFCN={14,18,22,24,60,66,73,74,75,76,108,114}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0002080000000F0208000 00000A22000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B3d
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time for cell B using 1024 format.
List_ARFCN={749, 758, 761, 764, 771, 779, 782, 791, 798, 829, 832, 844}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'0D'O		
fl	'830EF70BFEB843C3FBF9 F1D018'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B5
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time for cell B.
List_ARFCN={14,18,22,24,30,31,38,53,66,73,74,75,76,108,114}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0002080000000F0200100 02060A22000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B5d
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time for cell B.
List_ARFCN={756, 758, 761, 771, 779, 782, 791, 798, 829, 832, 844}

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'09'O		
fl	'8D7A23B07B3D7FD4D0'O		range 128 format

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B8d
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency list, after time for cell B using 1024 format.
List_ARFCN= complete cell allocation of cell B in HO cases.
739,743,746,749,756,758,761,764,771,779,782,791,798,829,832,844

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		
fl	'8303F886FDBC0148BEFD FBF41890633C'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_B10d
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : frequency short list, after time for cell B using range 256 format, ARFCN={764,779,782}

Element Name	Element Value	Element Encoding	Comments
iei	'00000010'B		
iel	–		
fl	'8B7E097D0000000000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_ind8B
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Uses bit map 0 to allocate the following 8 frequencies:(14, 18, 24, 30, 38, 60, 73, 74)

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'00000000000030008000 02020422000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_ind8Bd
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Uses range 256 to allocate the following 8 frequencies:(739, 743, 746, 749, 761, 771, 782, 791)

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'08'O		length = 8
fl	'8B718B712BEEA192'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Hlcmp_AnyOrOmit
Structured Type : HLCMP
Derivation Path :
Encoding Variation :
Comments : High Layer Compatibility IE with parameter "any'" or "omit".

Element Name	Element Value	Element Encoding	Comments
iei	'01111101'B IF_PRESENT		'01111101'B
iel	*		
extb3	*		extension bit ('1'B)
cs	*		CCITT
in	*		first HL characteristics identification to be used in the call.
pmpp	*		High layer protocol profile
extb4	*		extension bit, octet 4
hlci	*		telephony
extb4a	*		extension bit, '1'B, octet 4a
ehlci	*		extended high layer characteristics identification

Detailed Comments : used in test cases where the HLC is not checked

Structured Type Constraint Declaration

Constraint Name : Hlcmp_NotApplicable
Structured Type : HLCMP
Derivation Path :
Encoding Variation :
Comments : High Layer Compatibility IE wich is not applicable for a specific service (empty IE).

Element Name	Element Value	Element Encoding	Comments
iei	'01111101'B		'01111101'B
iel	'00'O		empty IE −> length is 0
extb3	−		
cs	−		
in	−		
pmpp	−		
extb4	−		
hlci	−		
extb4a	−		
ehlci	−		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : IaRestOct(iel : LENGTH)
Structured Type : IARESTOCT
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	−		
iaroct1	OC_SubOctet('2B2B2B2B2B2B2B2B2B2B 2B2B2B2B2B2B2B'O, (11 − OC_OctToInt(iel)))		p=00 format
iaroct2	−		p=10 format

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : IaRestOct_01
Structured Type : IARESTOCT
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	−		
iaroct1	'2B2B2B2B2B2B2B2B2B2B 2B'O		p=00 format, 11 octets
iaroct2	−		p=10 format used in TC_26_6_13_9, TC_26_6_13_10

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Llcmp_AnyOrOmit
Structured Type : LLCMP
Derivation Path :
Encoding Variation :
Comments : High Layer Compatibility IE which has any or omit value in all fields.

Element Name	Element Value	Element Encoding	Comments
iei	'01111100'B IF_PRESENT		'01111100'B
iel	*		empty IE −> length is 0
extb3	*		extension bit, octet 3
cs	*		coding standard, octet 3
itc	*		information transfer capability, octet 3
extb3a	*		extension bit, '1'B, octet 3a
negind	*		negotiation indicator
spb3a	*		spare bits, '000000'B
extb4	*		extension bit, octet 4
tm	*		transfer mode
itr	*		information transfer rate, octet 4
extb4_1	*		extension bit, octet 4.1
rateml	*		not checked
extb5	*		extension bit, octet 5
l1id	*		L1 identity, '01'B, octet 5
uil1	*		user information L1 protocol, octet 5
extb5a	*		extension bit, octet 5a
sb	*		synchronous bit, octet 5a
neg	*		negotiation bit, octet 5a
ur	*		user rate, octet 5a
extb5b1	*		extension bit, octet 5b1
ir	*		intermediate rate, octet 5b1
nictx	*		network independent clock on transmission, octet 5b1
nicrx	*		network independent clock on reception, octet 5b1
fctx	*		flow control on transmission, octet 5b1
fcrx	*		flow control on reception, octet 5b1
spb5b1	*		spare bit, '0'B, octet 5b1
extb5b2	*		extension bit, octet 5b2
hdrb	*		Hdr/no Hdr−bit, octet 5b2
mfs	*		Multi frame support, octet 5b2
mode	*		Mode, octet 5b2
llineg	*		LLI negotiation, octet 5b2
ass	*		Assignor/Assignee, octet 5b2
ibob	*		Inband/outband negotiation, octet 5b2
spb5b2	*		spare bit, '0'B, octet 5b2

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
extb5c	*		extension bit, octet 5c
nsb	*		number of stop bits, octet 5c
ndb	*		number of data bits, octet 5c
pi	*		parity information, octet 5c
extb5d	*		extension bit, ’1’B, octet 5d
dplxm	*		duplex mode, octet 5d
modemt	*		modem type, octet 5d
extb6	*		extension bit, octet 6
l2id	*		L2 identity, ’10’B, octet 6
uil2	*		not checked
extb6a1	*		not checked
modl2	*		not checked
spb6a1	*		not checked
q933	*		not checked
extb6a2	*		not checked
usl2pi	*		not checked
extb6b	*		not checked
ws	*		not checked
extb7	*		not checked
l3id	*		not checked
uil3	*		not checked
extb7a1	*		not checked
modl3	*		not checked
spb7a1	*		not checked
extb7a2	*		not checked
usl3pi	*		not checked
extb7b	*		not checked
spb7b	*		not checked
dps	*		not checked
extb7c	*		not checked
pws	*		not checked
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : Llcmp_NotApplicable
Structured Type : LLCMP
Derivation Path :
Encoding Variation :
Comments : High Layer Compatibility IE which is not applicable for a specific service (empty IE).

Element Name	Element Value	Element Encoding	Comments
iei	'01111100'B		'01111100'B
iel	'00'O		empty IE −> length is 0
extb3	−		extension bit, octet 3
cs	−		coding standard, octet 3
itc	−		information transfer capability, octet 3
extb3a	−		extension bit, '1'B, octet 3a
negind	−		negotiation indicator
spb3a	−		spare bits, '000000'B
extb4	−		extension bit, octet 4
tm	−		transfer mode
itr	−		information transfer rate, octet 4
extb4_1	−		extension bit, octet 4.1
rateml	−		not checked
extb5	−		extension bit, octet 5
l1id	−		L1 identity, '01'B, octet 5
uil1	−		user information L1 protocol, octet 5
extb5a	−		extension bit, octet 5a
sb	−		synchronous bit, octet 5a
neg	−		negotiation bit, octet 5a
ur	−		user rate, octet 5a
extb5b1	−		extension bit, octet 5b1
ir	−		intermediate rate, octet 5b1
nictx	−		network independent clock on transmission, octet 5b1
nicrx	−		network independent clock on reception, octet 5b1
fctx	−		flow control on transmission, octet 5b1
fcrx	−		flow control on reception, octet 5b1
spb5b1	−		spare bit, '0'B, octet 5b1
extb5b2	−		extension bit, octet 5b2
hdrb	−		Hdr/no Hdr−bit, octet 5b2
mfs	−		Multi frame support, octet 5b2
mode	−		Mode, octet 5b2
llineg	−		LLI negotiation, octet 5b2
ass	−		Assignor/Assignee, octet 5b2
ibob	−		Inband/outband negotiation, octet 5b2
spb5b2	−		spare bit, '0'B, octet 5b2

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
extb5c	–		extension bit, octet 5c
nsb	–		number of stop bits, octet 5c
ndb	–		number of data bits, octet 5c
pi	–		parity information, octet 5c
extb5d	–		extension bit, ’1’B, octet 5d
dplxm	–		duplex mode, octet 5d
modemt	–		modem type, octet 5d
extb6	–		extension bit, octet 6
l2id	–		L2 identity, ’10’B, octet 6
uil2	–		not checked
extb6a1	–		not checked
modl2	–		not checked
spb6a1	–		not checked
q933	–		not checked
extb6a2	–		not checked
usl2pi	–		not checked
extb6b	–		not checked
ws	–		not checked
extb7	–		not checked
l3id	–		not checked
uil3	–		not checked
extb7a1	–		not checked
modl3	–		not checked
spb7a1	–		not checked
extb7a2	–		not checked
usl3pi	–		not checked
extb7b	–		not checked
spb7b	–		not checked
dps	–		not checked
extb7c	–		not checked
pws	–		not checked
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : Llcmp_BS2xOrBS4x_UDI
Structured Type : LLCMP
Derivation Path :
Encoding Variation :
Comments : Lower Layer Compatibility IE for a setup PDU, B.2.2.1 of GSM 07.01.

Element Name	Element Value	Element Encoding	Comments
iei	’01111100’B		’01111100’B
iel	(’06’O, ’07’O, ’08’O, ’09’O, ’0A’O, ’0B’O, ’0C’O, ’0D’O, ’0E’O)		at least 6 OCTETs
extb3	’?’B		extension bit , octet 3
cs	’00’B		CCITT standardized coding
itc	’01000’B		unrestricted digital information
extb3a	’1’B IF_PRESENT		extension bit, ’1’B, octet 3a
negind	*		negotiation indicator
spb3a	’000000’B IF_PRESENT		spare bits, ’000000’B
extb4	’1’B		extension bit, octet 4
tm	’00’B		circuit mode
itr	’10000’B		64 kBit/s
extb4_1	−		not presented
rateml	−		not presented
extb5	’0’B		extension bit, octet 5
l1id	’01’B		L1 identity, ’01’B, octet 5
uil1	’00001’B		V.110/X.30 rate adaption
extb5a	’0’B		extension bit, octet 5a
sb	’1’B		asynchronous
neg	’0’B		negotiation not possible
ur	(’00010’B, ’00011’B, ’00101’B, ’01000’B, ’10111’B, ’11110’B)		1.2, 2.4, 4.8, 9.6, 0.075/1.2, 0.3 kbit/s
extb5b1	’0’B		extension bit, octet 5b1
ir	(’01’B, ’10’B)		8, 16 kbit/s
nictx	’?’B		not relevant
nicrx	’?’B		not relevant
fctx	’0’B		not required
fcrx	’0’B		cannot accept data with flow control
spb5b1	’0’B		spare bit
extb5b2	−		extension bit, octet 5b2
hdrb	−		Hdr/no Hdr−bit, octet 5b2
mfs	−		Multi frame support, octet 5b2
mode	−		Mode, octet 5b2
llineg	−		LLI negotiation, octet 5b2
ass	−		Assignor/Assignee, octet 5b2
ibob	−		Inband/outband negotiation, octet 5b2
spb5b2	−		spare bit, ’0’B, octet 5b2
extb5c	’?’B		extension bit, octet 5c

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
nsb	(’01’B, ’11’B)		1, 2 bits
ndb	(’10’B, ’11’B)		7, 8 bits
pi	(’000’B,’010’B, ’011’B, ’100’B, ’101’B)		odd, even, none, forced to 0, forced to 1
extb5d	’1’B IF_PRESENT		extension bit, octet 5d
dplxm	*		half or full duplex
modemt	*		not relevant
extb6	*		extension bit, octet 6
l2id	*		L2 identity, ’10’B, octet 6
uil2	*		not checked
extb6a1	*		not checked
modl2	*		not checked
spb6a1	*		not checked
q933	*		not checked
extb6a2	*		not checked
usl2pi	*		not checked
extb6b	*		not checked
ws	*		not checked
extb7	*		not checked
l3id	*		not checked
uil3	*		not checked
extb7a1	*		not checked
modl3	*		not checked
spb7a1	*		not checked
extb7a2	*		not checked
usl3pi	*		not checked
extb7b	*		not checked
spb7b	*		not checked
dps	*		not checked
extb7c	*		not checked
pws	*		not checked
Detailed Comments : Corresponding to the Bearer Services 21 ... 26, 41 ... 46 Unrestrict digital information transfer capability.			

Structured Type Constraint Declaration

Constraint Name : Llcmp_BS2xOrBS61aOrBS81a_3100
Structured Type : LLCMP
Derivation Path : Llcmp_BS2xOrBS4x_UDI.
Encoding Variation :
Comments : Lower Layer Compatibility IE for a setup PDU, B.2.2.2 of GSM 07.01.

Element Name	Element Value	Element Encoding	Comments
iel	('07'O,'08'O, '09'O, '0A'O, '0B'O, '0C'O, '0D'O, '0E'O)		at least 7 OCTETs
extb3	'?'B		extension bit , octet 3
cs	'00'B		CCITT standardized coding
itc	'10000'B		3.1kHz audio
extb4	'1'B		extension bit, octet 4
tm	'00'B		circuit mode
itr	'10000'B		64 kBit/s
extb5	'0'B		extension bit, octet 5
l1id	'01'B		L1 identity, '01'B, octet 5
uil1	'00011'B		G.711 A−law
extb5a	'0'B		can not be omitted in order to have 5d
sb	?		not checked
neg	?		not checked
ur	?		not checked
extb5b1	'0'B		can not be omitted in order to have 5d
ir	?		not checked
nictx	?		not checked
nicrx	?		not checked
fctx	?		not checked
fcrx	?		not checked
spb5b1	'0'B		not checked
extb5c	'0'B		can not be omitted in order to have 5d
nsb	?		not checked
ndb	?		not checked
pi	?		not checked
extb5d	'1'B		extension bit, octet 5d
dplxm	?		half or full duplex
modemt	('010001'B, '010010'B, '010011'B, '010100'B, '010111'B, '011100'B)		V.21, V.22, V.22bis, V.23, V.26ter, V.32

Detailed Comments : Corresponding to the Bearer Services 21 ... 26, 61 Asynchronous or 81 Asynchronous 3.1 kHz audio ex−PLMN information transfer capability.

Structured Type Constraint Declaration

Constraint Name : Llcmp_BS3x_UDI
Structured Type : LLCMP
Derivation Path : Llcmp_BS2xOrBS4x_UDI.
Encoding Variation :
Comments : Lower Layer Compatibility IE for a setup PDU, B.2.3.1 of GSM 07.01.

Element Name	Element Value	Element Encoding	Comments
iel	('05'O, '06'O, '07'O, '08'O, '09'O, '0A'O, '0B'O, '0C'O, '0D'O, '0E'O)		at least 5 OCTETs
extb3	'?'B		extension bit, octet 3
cs	'00'B		CCITT standardized coding
itc	'01000'B		unrestricted digital information
extb4	'1'B		extension bit, octet 4
tm	'00'B		circuit mode
itr	'10000'B		64 kBit/s
extb5	'0'B		extension bit, octet 5
l1id	'01'B		L1 identity, '01'B, octet 5
uil1	('00001'B, '01001'B)		V.100/X.30 rate adaption or X.31 flag stuffing
extb5a	'0'B		extension bit, octet 5a
sb	'0'B		synchronous
neg	'0'B		negotiation not possible
ur	('00010'B, '00011'B, '00101'B, '01000'B)		1.2, 2.4, 4.8, 9.6 kbit/s
extb5b1	'?'B		extension bit, octet 5b1
ir	('01'B, '10'B)		8, 16 kbit/s
nictx	'?'B		not checked
nicrx	'?'B		not checked
fctx	'?'B		not relevant
fcrx	'?'B		not relevant
spb5b1	'0'B		spare bit
extb5c	'0'B IF_PRESENT		extension bit, octet 5c
nsb	*		any or omit
ndb	*		any or omit
pi	*		any or omit
extb5d	'1'B IF_PRESENT		extension bit, octet 5d
dplxm	*		duplex mode
modemt	*		not relevant
extb6	*		extension bit, octet 6
l2id	'10'B IF_PRESENT		L2 identity, '10'B, octet 6
uil2	('00110'B, '00111'B) IF_PRESENT		X.25 link layer or X.25 Multilink
extb7	*		extension bit, octet 7
l3id	'11'B IF_PRESENT		L3 identity, '11'B, octet 7
uil3	('00110'B, '00111'B) IF_PRESENT		X.25 packet layer, or X.25 packet level protocol for data terminal

Detailed Comments : Corresponding to the Bearer Services 31 ... 34 unrestricted digital information transfer capability.

Structured Type Constraint Declaration

Constraint Name : Llcmp_BS3xOrBS61sOrBS81s_3100
Structured Type : LLCMP
Derivation Path : Llcmp_BS2xOrBS4x_UDI.
Encoding Variation :
Comments : Lower Layer Compatibility IE for a setup PDU, B.2.3.2 of GSM 07.01.

Element Name	Element Value	Element Encoding	Comments
iel	('03'O,'04'O, '08'O, '09'O, '0A'O, '0B'O, '0C'O, '0D'O, '0E'O)		at least 3 OCTETs
extb3	'?'B		extension bit , octet 3
cs	'00'B		CCITT standardized coding
itc	'10000'B		3.1kHz audio
extb4	'1'B		extension bit, octet 4
tm	'00'B		circuit mode
itr	'10000'B		64 kBit/s
extb5	'?'B		extension bit, octet 5
l1id	'01'B		L1 identity, '01'B, octet 5
uil1	'00011'B		G.711 A−law
extb5a	'0'B IF_PRESENT		extension bit, octet 5a
sb	*		any or omit
neg	*		any or omit
ur	*		any or omit
extb5b1	*		any or omit
ir	*		any or omit
nictx	*		any or omit
nicrx	*		any or omit
fctx	*		any or omit
fcrx	*		any or omit
spb5b1	*		any or omit
extb5c	'0'B IF_PRESENT		extension bit, octet 5c
nsb	*		any or omit
ndb	*		any or omit
pi	*		any or omit
extb5d	'1'B IF_PRESENT		extension bit, octet 5d
dplxm	*		duplex mode
modemt	('010010'B, '010011'B, '010111'B, '011100'B) IF_PRESENT		V.22, V.22bis, V.26ter, V.32
extb6	*		extension bit, octet 6
l2id	'10'B IF_PRESENT		L2 identity, '10'B, octet 6
uil2	('00110'B, '00111'B) IF_PRESENT		X.25 link layer or X.25 Multilink
extb7	*		extension bit, octet 7
l3id	'11'B IF_PRESENT		L3 identity, '11'B, octet 7
uil3	('00110'B, '00111'B) IF_PRESENT		X.25 packet layer, or X.25 packet level protocol for data terminal

Detailed Comments : Corresponding to the Bearer Services 31 ... 34, 61 Synchronous or 81 Synchronous 3.1kHz audio information transfer capability.

Structured Type Constraint Declaration

Constraint Name : Llcmp_BS5x
Structured Type : LLCMP
Derivation Path : Llcmp_BS2xOrBS4x_UDI.
Encoding Variation :
Comments : Lower Layer Compatibility IE for a setup PDU, B.2.5.1 of GSM 07.01.

Element Name	Element Value	Element Encoding	Comments
iel	('06'O,'07'O, '08'O, '09'O, '0A'O, '0B'O, '0C'O, '0D'O, '0E'O)		at least 6 OCTETs
extb3	'?'B		extension bit, octet 3
cs	'00'B		CCITT standardized coding
itc	'01000'B		unrestricted digital information
extb4	'1'B		extension bit, octet 4
tm	'00'B		circuit mode
itr	'10000'B		64 kBit/s
extb5	'0'B		extension bit, octet 5
l1id	'01'B		L1 identity, '01'B, octet 5
uil1	'01001'B		X.31 flag stuffing
extb5a	'?'B		extension bit, octet 5a
sb	'0'B		synchronous
neg	'0'B		negotiation not possible
ur	('00011'B, '00101'B, '01000'B)		2.4, 4.8, 9.6 kbit/s
extb5b1	*		extension bit, octet 5b1
ir	*		not checked
nictx	*		not checked
nicrx	*		not checked
fctx	*		not checked
fcrx	*		not checked
spb5b1	*		not checked
extb5c	*		not checked
nsb	*		not checked
ndb	*		not checked
pi	*		not checked
extb5d	*		not checked
dplxm	*		not checked
modemt	*		not checked
extb6	?		extension bit, octet 6
l2id	'10'B		L2 identity, '10'B, octet 6
uil2	('00110'B, '00111'B)		X.25 link layer or X.25 Multilink
extb7	?		extension bit, octet 7
l3id	'11'B		L3 identity, '11'B, octet 7
uil3	('00110'B, '00111'B)		X.25 packet layer, or X.25 packet level protocol for data terminal

Detailed Comments : Corresponding to the Bearer Services 51 ... 53 unrestricted digital information transfer capability.

Structured Type Constraint Declaration

Constraint Name : LocAreaId(mcc, mnc, lac : OCTETSTRING)
Structured Type : LAI
Derivation Path :
Encoding Variation :
Comments : Cell A default value for L3 test

Element Name	Element Value	Element Encoding	Comments
iei	–		not used in SI3
mcc	mcc		mobile country code
mnc	mnc		mobile network code
lac	lac		location area code, Cell A

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : LocUpType(locup : B_2)
Structured Type : LUT
Derivation Path :
Encoding Variation :
Comments : location updating type

Element Name	Element Value	Element Encoding	Comments
foreq	'?'B		no follow–on request pending
sprb	'0'B		spare bit '0'B
lut	locup		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MiImsi_01iei
Structured Type : MI
Derivation Path :
Encoding Variation :
Comments : the TSPX_IMSI is the IMSI of the MS under test

Element Name	Element Value	Element Encoding	Comments
iei	'00010111'B		
iel	OC_IntToOct(((LENGTH_OF(TSPX_IMSI))/2 + 1),1)		
idigit_1	OC_FirstDigi(TSPX_IMSI)		
oei	OC_OeBit(TSPX_IMSI)		
toi	'001'B		IMSI is used
idigits_other	OC_OtherDigi(TSPX_IMSI)		

Detailed Comments :

Structured Type Constraint Declaration			
Constraint Name : MiImsi_01			
Structured Type : MI			
Derivation Path :			
Encoding Variation :			
Comments : the TSPX_IMSI is the IMSI of the MS under test			
Element Name	**Element Value**	**Element Encoding**	**Comments**
iei	–		
iel	OC_IntToOct(((LENGTH_OF(TSPX_IMSI))/2 + 1),1)		
idigit_1	OC_FirstDigi(TSPX_IMSI)		
oei	OC_OeBit(TSPX_IMSI)		
toi	’001’B		IMSI is used
idigits_other	OC_OtherDigi(TSPX_IMSI)		
Detailed Comments :			

Structured Type Constraint Declaration			
Constraint Name : MiImsi_02			
Structured Type : MI			
Derivation Path :			
Encoding Variation :			
Comments : This constraint is used in paging filling message. No mobile is paged (Type of identity is set to ’000’B, i.e. no identity).			
Element Name	**Element Value**	**Element Encoding**	**Comments**
iei	–		
iel	’01’O		
idigit_1	’1111’B		
oei	’0’B		
toi	’000’B		no identity
idigits_other	–		
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : MiTmsi_01
Structured Type : MI
Derivation Path :
Encoding Variation :
Comments : default TMSI for testing.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'05'O		
idigit_1	'1111'B		
oei	'0'B		
toi	'100'B		TMSI is used
idigits_other	TSPX_TMSI		4 octets long

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MiTmsi_01iei
Structured Type : MI
Derivation Path :
Encoding Variation :
Comments : default TMSI for testing.

Element Name	Element Value	Element Encoding	Comments
iei	'00010111'B		
iel	'05'O		
idigit_1	'1111'B		
oei	'0'B		
toi	'100'B		TMSI is used
idigits_other	TSPX_TMSI		4 octets long

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MiTmsi_02
Structured Type : MI
Derivation Path :
Encoding Variation :
Comments : A new TMSI

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'05'O		
idigit_1	'1111'B		
oei	'0'B		
toi	'100'B		TMSI is used
idigits_other	TSPX_TMSI1		new TMSI

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MiTmsi_r02iei
Structured Type : MI
Derivation Path :
Encoding Variation :
Comments : default TMSI for testing with toi setting to no id.

Element Name	Element Value	Element Encoding	Comments
iei	'00010111'B		
iel	'05'O		
idigit_1	'1111'B		
oei	'0'B		
toi	'000'B		no identity
idigits_other	TSPX_TMSI		4 octets long

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_omit
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	–		
mac_8n	–		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_01
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : non hopping

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'00'O		non hopping
mac_8n	–		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_01iei
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : non hopping

Element Name	Element Value	Element Encoding	Comments
iei	'01110010'B		
iel	'00'O		non hopping
mac_8n	–		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_ind5A
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : Indicates 5 frequences (17, 34, 46, 52, 73)

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'03'O		
mac_8n	'00000000'B		
mac_7n	'00001001'B		
mac_6n	'10010010'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_ind5Ad
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : Indicates 5 frequences (734, 741, 762, 791, 832)

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'03'O		
mac_8n	'00000000'B		
mac_7n	'10010000'B		
mac_6n	'00100011'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_281
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with
indicates to the ARFCN_list={73,74,75} of cell allocation A.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'03'O		
mac_8n	'00000000'B		MoblAllc_281d
mac_7n	'00111000'B		
mac_6n	'00000000'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_281d
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with
indicates to the ARFCN_list={773,775,779} of cell allocation A.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'03'O		
mac_8n	'00000000'B		
mac_7n	'00000111'B		
mac_6n	'00000000'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_ad281d
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with
indicates to the ARFCN_list={734,762,791} of cell allocation A.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'03'O		
mac_8n	'00000000'B		
mac_7n	'00010000'B		
mac_6n	'00100001'B		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_02
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE matching any value.

Element Name	Element Value	Element Encoding	Comments
ba_used	?		
dtx_used	'0'B		dtx was not used
rxlev_fsc	?		
spr1	'0'B		spare bit
meas_valid	?		measurement results valid
rxlev_ssc	?		
spr2	'0'B		spare bit
rxqual_fsc	?		
rxqual_ssc	?		
no_nc	?		no result
rxlev_nc1	*		received signal strength on nc1
bcchfrq_nc1	*		bcch frequency position of nc1
bsic_nc1	*		base station identity code of nc1
rxlev_nc2	*		received signal strength on nc2
bcchfrq_nc2	*		bcch frequency position of nc2
bsic_nc2	*		base station identity code of nc2
rxlev_nc3	*		received signal strength on nc3
bcchfrq_nc3	*		bcch frequency position of nc3
bsic_nc3	*		base station identity code of nc3
rxlev_nc4	*		received signal strength on nc4
bcchfrq_nc4	*		bcch frequency position of nc4
bsic_nc4	*		base station identity code of nc4
rxlev_nc5	*		received signal strength on nc5
bcchfrq_nc5	*		bcch frequency position of nc5
bsic_nc5	*		base station identity code nc5
rxlev_nc6	*		received signal strength on nc6
bcchfrq_nc6	*		bcch frequency position of nc6
bsic_nc6	*		base station identity code of nc6

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_03a
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE containing 6 measurement results.

Element Name	Element Value	Element Encoding	Comments
ba_used	'0'B		
dtx_used	'0'B		dtx was not used
rxlev_fsc	?		
spr1	'0'B		spare bit
meas_valid	'0'B		measurement results valid
rxlev_ssc	?		
spr2	'0'B		spare bit
rxqual_fsc	?		
rxqual_ssc	?		
no_nc	?		
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	?		bcch frequency position of nc1
bsic_nc1	?		base station identity code of nc1
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	?		bcch frequency position of nc2
bsic_nc2	?		base station identity code of nc2
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	?		bcch frequency position of nc3
bsic_nc3	?		base station identity code of nc3
rxlev_nc4	?		received signal strength on nc4
bcchfrq_nc4	?		bcch frequency position of nc4
bsic_nc4	?		base station identity code of nc4
rxlev_nc5	?		received signal strength on nc5
bcchfrq_nc5	?		bcch frequency position of nc5
bsic_nc5	?		base station identity code nc5
rxlev_nc6	?		received signal strength on nc6
bcchfrq_nc6	?		bcch frequency position of nc6
bsic_nc6	?		base station identity code of nc6

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : NT_BothT_BothNTorT_BothT_BothNT
Structured Type : CE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
TorBothTorBothNT	('00'B, '10'B, '11'B)		
NTorBothTorBothNT	('01'B, '10'B, '11'B)		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Pcmd_19(powerlevel:BITSTRING)
Structured Type : PCMD
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
sprb	'000'B		spare bits
pl	powerlevel		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Pcmd_20(powerlevel : B_5)
Structured Type : PCMD
Derivation Path :
Encoding Variation :
Comments : PowerCmd used in synchronized and non synchronized HO cases.

Element Name	Element Value	Element Encoding	Comments
sprb	'011'B		spare bits
pl	powerlevel		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Pm(pgm : B_2)
Structured Type : PM
Derivation Path :
Encoding Variation :
Comments : normal paging mode

Element Name	Element Value	Element Encoding	Comments
sprb	’00’B		spare bits
pgm	pgm		page mode

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ProgInd_omit
Structured Type : PI
Derivation Path :
Encoding Variation :
Comments : Omitted Progress Indicator.

Element Name	Element Value	Element Encoding	Comments
iei	–		’00011110’B
iel	–		
extb3	–		extension bit (’0’B)
cs	–		coding standard
spb	–		arbitrary spare bit
loc	–		location
extb4	–		extension bit (’0’B)
prd	–		progress description

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : RachCntrlPara(maxtx : B_2; txint : B_4; re : B_1)
Structured Type : RACHCP
Derivation Path :
Encoding Variation :
Comments : Default value for L 3 testing.

Element Name	Element Value	Element Encoding	Comments
maxrtx	maxtx		
txint	txint		
cba	’0’B		not barred
re	re		
acc_2	’00000’B		access not barred
ec	’0’B		emergency call allowed
acc_1	’0000000000’B		access not barred

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : RachCntrlPara_r01
Structured Type : RACHCP
Derivation Path :
Encoding Variation :
Comments : Call reestablishment is not allowed in the cell.

Element Name	Element Value	Element Encoding	Comments
maxrtx	'00'B		Max−retrans = 1
txint	'0010'B		TX−integer=5
cba	'0'B		not barred
re	'1'B		no reestablishment
acc_2	'00000'B		access not barred
ec	'0'B		emergency call allowed
acc_1	'0000000000'B		access not barred

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : RelTmdDif_omit
Structured Type : TDIF
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	−		
iel	−		
value	−		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Rqr2(Rr: BITSTRING; Fn: FN)
Structured Type : RQR
Derivation Path :
Encoding Variation :
Comments : To address the MS under test

Element Name	Element Value	Element Encoding	Comments
ra	Rr		
fn	Fn		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI3RestOctetDef
Structured Type : SI3RO
Derivation Path :
Encoding Variation :
Comments : Default value for L 3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	’0’B		
cbq	’0’B		
cro	’101011’B		
to	’001’B		
pt	’01011’B		
poi	’0’B		
po	’01’B		
ti	’0’B		
ecsc	’1’B		
spr1	’011’B		
spr2	’00101011’B		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI4RestOctetDef
Structured Type : SI4RO
Derivation Path :
Encoding Variation :
Comments : Default value for L 3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	’0’B		
cbq	’0’B		
cro	’101011’B		
to	’001’B		
pt	’01011’B		
poi	’0’B		
po	’01’B		
spr1	’01011’B		
spr2	’00101011’B		
spr3	’00101011’B		
spr4	’00101011’B		
spr5	’00101011’B		
spr6	’00101011’B		
spr7	’00101011’B		
spr8	’00101011’B		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI4RestOctet_CBMS
Structured Type : SI4RO
Derivation Path :
Encoding Variation :
Comments : Rest octets for SMSCB SysInf4 message.

Element Name	Element Value	Element Encoding	Comments
pi	'0'B		
cbq	'0'B		
cro	'101011'B		
to	'001'B		
pt	'01011'B		
poi	'0'B		
po	'01'B		
spr1	'01011'B		
spr2	'00101011'B		
spr3	–		
spr4	–		
spr5	–		
spr6	–		
spr7	–		
spr8	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Signal_01
Structured Type : SIGNAL
Derivation Path :
Encoding Variation :
Comments : signal value is arbitrarily selected.

Element Name	Element Value	Element Encoding	Comments
iei	'00110100'B		
sigv	'00000000'B		dial tone on

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : StartingTm_omit
Structured Type : STRT
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
fn	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Synchi(rot : B_1; si : B_2)
Structured Type : SYNCHI
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	’1101’B		
nci	’0’B		ignore out of range TA
rot	rot		report observed time difference
si	si		synchronisation type

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Synchi_omit
Structured Type : SYNCHI
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
nci	–		
rot	–		
si	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : TimingAdv(bits : INTEGER)
Structured Type : TA
Derivation Path :
Encoding Variation :
Comments : Time Advance value is pameterized.

Element Name	Element Value	Element Encoding	Comments
iei	–		
sprb	'00'B		spare bits
value	INT_TO_BIT(bits, 6)		one unit = 48/13 microseconds (in 1 bit periods)

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : TimingAdv_omit
Structured Type : TA
Derivation Path :
Encoding Variation :
Comments : Time Advance value is pameterized.

Element Name	Element Value	Element Encoding	Comments
iei	–		
sprb	–		
value	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : TI_01
Structured Type : TI
Derivation Path :
Encoding Variation :
Comments : used by the MS in the transaction initiated by the test system.

Element Name	Element Value	Element Encoding	Comments
ti_f	'1'B		
ti_v	'000'B		TI value

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : TI_02
Structured Type : TI
Derivation Path :
Encoding Variation :
Comments : used in the messages sent to the MS in the transaction initiated by test system.

Element Name	Element Value	Element Encoding	Comments
ti_f	'0'B		from network to MS
ti_v	'000'B		TI value

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : TI_09
Structured Type : TI
Derivation Path :
Encoding Variation :
Comments : used by the MS in the transaction initiated by the MS.

Element Name	Element Value	Element Encoding	Comments
ti_f	'0'B		
ti_v	?		TI value is selected by the MS when it starts the transaction

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : UuInfo_omit
Structured Type : UU
Derivation Path :
Encoding Variation :
Comments : Omitted UUInfo.

Element Name	Element Value	Element Encoding	Comments
iei	−		'01111110'B
iel	−		
uupd	−		user−user protocol discriminator
uui	−		user user information

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : V22bisOrV26ter
Structured Type : MODEMTYPE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
V22bisOrV26ter	('00011'B, '00101'B)		

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ActivateSS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CF all synchronous services

Constraint Value

```
activateSSComponents    activateSS_InvokeComp
                                    {
                                      invokeID              ?,
                                      localValue             12,
                                      ss_ForBS
                                                  {
                                                    ss_Code             '20'H,
                                                    basicService    bearerService    '68'H
                                                  }
                                    }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ActivateSS_02
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFU all basic services

Constraint Value

```
activateSSComponents    activateSS_InvokeComp
                                    {
                                      invokeID              ?,
                                      localValue             12,
                                      ss_ForBS
                                                  {
                                                    ss_Code             '21'H
                                                  }
                                    }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ActivateSS_03
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BAOC all synchronous services

Constraint Value

```
activateSSComponents    activateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             12,
                                         ss_ForBS
                                                        {
                                                          ss_Code                '92'H,
                                                          basicService      bearerService        '68'H
                                                         }
                                         }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ActivateSS_04
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BIC−Roam

Constraint Value

```
activateSSComponents    activateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             12,
                                         ss_ForBS
                                                        {
                                                          ss_Code                '9B'H  −− BIC−Roam
                                                         }
                                         }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ActivateSS_05
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BOIC

Constraint Value

```
activateSSComponents    activateSS_InvokeComp
                                        {
                                          invokeID              ?,
                                          localValue             12,
                                          ss_ForBS
                                                        {
                                            ss_Code                 '93'H
                                                        }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ActivateSS_06
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BAIC

Constraint Value

```
activateSSComponents    activateSS_InvokeComp
                                        {
                                          invokeID              ?,
                                          localValue             12,
                                          ss_ForBS
                                                        {
                                            ss_Code                 '9A'H
                                                        }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : BldMptySS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : build multiparty request

Constraint Value

```
buildMPTYComponents buildMPTY_InvokeComp
                                        {
                                          invokeID              ?,
                                          localValue             124
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : Comps_01(comp : Component)
ASN1 Type : Components
Derivation Path :
Encoding Variation :
Comments : A received FIE may contain several components, but at least contains "comp".

Constraint Value

SUPERSET({ comp})

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : DeactivateSS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFC for speech

Constraint Value

```
deactivateSSComponents    deactivateSS_InvokeComp
                                        {
                                         invokeID                ?,
                                         localValue               13,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '28'H,
                                                   basicService   teleservice   '10'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : DeactivateSS_02
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFNRc for all facsimile

Constraint Value

```
deactivateSSComponents    deactivateSS_InvokeComp
                                        {
                                         invokeID                ?,
                                         localValue               13,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '2B'H,
                                                   basicService   teleservice   '60'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : DeactivateSS_03
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : deactivation for barring

Constraint Value

```
deactivateSSComponents    deactivateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue            13,
                                         ss_ForBS
                                                  {
                                                   ss_Code      '90'H,
                                                   basicService   teleservice   '10'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : DeactivateSS_04
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : deactivation for barring of outging calls

Constraint Value

```
deactivateSSComponents    deactivateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue            13,
                                         ss_ForBS
                                                  {
                                                   ss_Code      '91'H,
                                                   basicService   teleservice   '60'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : DeactivateSS_05
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : deactivation for barring of incoming calls

Constraint Value

```
deactivateSSComponents    deactivateSS_InvokeComp
                                        {
                                         invokeID                ?,
                                         localValue               13,
                                         ss_ForBS
                                                  {
                                                   ss_Code      '99'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : DeactivateSS_06
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : deactivation for BOICExHC

Constraint Value

```
deactivateSSComponents    deactivateSS_InvokeComp
                                       {
                                         invokeID                ?,
                                         localValue               13,
                                         ss_ForBS
                                                 {
                                                  ss_Code      '94'H
                                                  }
                                       }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : EraseSS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFC for all facsimile

Constraint Value

```
eraseSSComponents    eraseSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue            11,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '28'H,
                                                   basicService   teleservice   '60'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : EraseSS_02
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFNRc for all basic services

Constraint Value

```
eraseSSComponents    eraseSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue            11,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '2B'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : EraseSS_03
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFU for speech

Constraint Value

```
eraseSSComponents    eraseSS_InvokeComp
                                        {
                                         invokeID                ?,
                                         localValue               11,
                                         ss_ForBS
                                                    {
                                                     ss_Code     '21'H,
                                                     basicService   teleservice   '10'H
                                                     }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : EraseSS_04
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFNRy for all facsimile

Constraint Value

```
eraseSSComponents    eraseSS_InvokeComp
                                        {
                                         invokeID                ?,
                                         localValue               11,
                                         ss_ForBS
                                                    {
                                                     ss_Code     '2A'H,
                                                     basicService   teleservice   '60'H
                                                     }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : FwdChAdvRslt_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
forwardChargeAdviceComponents    forwardChargeAdvice_ReturnResultComp
        {
                                                    invokeID                0,
                                                    result                  *


          }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : GetPasswdRslt_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
getPasswordComponents    getPassword_ReturnResultComp
                                                   {
                                                    invokeID                0,
                                                    result
                                                          {
                                                           localValue        18,
                                                           currentPassword   "1234"
                                                          }
                                                    }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : GetPasswdRslt_02
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
getPasswordComponents    getPassword_ReturnResultComp
                                                {
                                                 invokeID                0,
                                                 result
                                                        {
                                                         localValue        18,
                                                         currentPassword   "9876"
                                                        }
                                                 }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : GetPasswdRslt_03
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
getPasswordComponents    getPassword_ReturnResultComp
                                                {
                                                 invokeID                0,
                                                 result
                                                        {
                                                         localValue        18,
                                                         currentPassword   "9877"
                                                        }
                                                 }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFB for all basic services

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp      {
   invokeID                ?,
   localValue              14,
   ss_ForBS    {
                                                          ss_Code      '29'H
          }
                                   }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_02
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFNRy for Speech

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID                ?,
                                         localValue              14,
                                         ss_ForBS
                                                  {
                                                   ss_Code      '2A'H, −−CNFRy
                                                   basicService   teleservice   '10'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_03
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFNRc for all basic services

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             14,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '2B'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_04
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFB for all facsimile

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             14,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '29'H,
                                                   basicService   teleservice   '60'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_05
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BICRoam

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             14,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '9B'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_06
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BOIC

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             14,
                                         ss_ForBS
                                                  {
                                                   ss_Code     '93'H
                                                   }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_07
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BAIC

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             14,
                                         ss_ForBS
                                                 {
                                                  ss_Code     '9A'H
                                                  }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : InterrogateSS_08
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : BOICExHC

Constraint Value

```
interrogateSSComponents    interrogateSS_InvokeComp
                                        {
                                         invokeID              ?,
                                         localValue             14,
                                         ss_ForBS
                                                 {
                                                  ss_Code     '94'H
                                                  }
                                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : NotificationSS_07(Invkid: OCTETSTRING)
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
unstructuredSSNotifyComponents   unstructuredSSNotify_ReturnResultComp
                                 {
                                   invokeID              OC_OctToInvokeIDType(Invkid),
                                   result
                                         {
                                           localValue          61
                                         }
                                  }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : NotificationSS_09(Invkid: OCTETSTRING)
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
unstructuredSSNotifyComponents   unstructuredSSNotify_ReturnErrorComp

                                 errorCodes
                                        {
                                          invokeID              OC_OctToInvokeIDType(Invkid),
                                          errorCode             ussd_Busy
                                         }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : RegisterSS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFNRy Speech

Constraint Value

```
registerSSComponents    registerSS_InvokeComp   {
    invokeID                ?,
    localValue              10,                                     registerSS_Arg  {
                                                                     ss_Code                    '2A'H,
                                                                     basicService    teleservice     '10'H,
                                                                     forwardedToNumber          '8100342143'H,
                            -- Unknown Number/ISDN telephony + '00431234'
                                                                     noReplyConditionTime       5
                                                                        }
                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : RegisterSS_02
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFU

Constraint Value

```
registerSSComponents    registerSS_InvokeComp    {
   invokeID                 ?,
   localValue               10,
   registerSS_Arg  {
                                                        ss_Code                     '21'H,
                                                        basicService    teleservice     '60'H,
                                                        forwardedToNumber           '8100342143'H
                            -- Unknown Number/ISDN telephony + '00431234'

            }
                                    }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : RegisterSS_03
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CFB

Constraint Value

```
registerSSComponents    registerSS_InvokeComp    {
    invokeID                ?,
    localValue              10,
    registerSS_Arg    {
                                                    ss_Code                 '29'H,
                                                    basicService    bearerService    '60'H,
                                                    forwardedToNumber       '8100342143'H
                    -- Unknown Number/ISDN telephony + '00431234'
            }
                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : RegisterSS_04
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : CF for all facsimile

Constraint Value

```
registerSSComponents    registerSS_InvokeComp    {
    invokeID                ?,
    localValue              10,
    registerSS_Arg    {
                                                    ss_Code                 '20'H,
                                                    basicService    teleservice    '60'H,
                                                    forwardedToNumber       '8100342143'H
                    -- Unknown Number/ISDN telephony + '00431234'
            }
                        }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : RegPasswdSS_01
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : All call restriction services.

Constraint Value

```
registerPasswordComponents    registerPassword_InvokeComp
                                {
                                  invokeID              ?,
                                  localValue             17,
                                  ss_Code               '90'H
                                 }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ProcessUSSData_01(ussdString: IA5String)
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
processUnstructuredSSDataComponents    processUnstructuredSSData_InvokeComp
                                {
                                  invokeID              ?,
                                  localValue             19,
                                  ss_UserData           ussdString
                                }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : ProcessUSSDReq_01(ussdString: IA5String)
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments :

Constraint Value

```
processUnstructuredSSRequestComponents    processUnstructuredSSRequest_InvokeComp
                                {
                                  invokeID              ?,
                                  localValue             59,
                                  ussd_Arg
                                          {
                                           ussd_DataCodingScheme     'F0'O,
                                           ussd_String          OC_CodingOfUssdString(ussdString)
                                          }
                                 }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : USSDReq_04(Invkid : OCTETSTRING; ussdString: IA5String)
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : ReturnResult for Unstructured SS request

Constraint Value

```
unstructuredSSRequestComponents    unstructuredSSRequest_ReturnResultComp
                                   {
                                     invokeID                OC_OctToInvokeIDType(Invkid),
                                     result
                                           {
                                             localValue              60,
                                             ussd_Res
                                                   {
                                                    ussd_DataCodingScheme     'F0'O,
                                                    ussd_String OC_CodingOfUssdString(ussdString)
                                                    }
                                           }
                                     }
```

Detailed Comments :

ASN.1 Type Constraint Declaration

Constraint Name : USSDReq_05(Invkid : OCTETSTRING)
ASN1 Type : Component
Derivation Path :
Encoding Variation :
Comments : Return Error for UnstructuredSS−Request with the error code USSD Busy

Constraint Value

```
unstructuredSSRequestComponents    unstructuredSSRequest_ReturnErrorComp
                                   errorCodes
                                           {
                                             invokeID                OC_OctToInvokeIDType(Invkid),
                                             errorCode               ussd_Busy
                                           }
```

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AlertRcv(pdu: ALERT_PDU)
ASP Type : DL_DatInAlert
Derivation Path :
Comments : To receive an ALERTING message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AlertSnd(ch : LOGICCH; pdu: ALERT_PDU)
ASP Type : DL_DatRqAlert
Derivation Path :
Comments : To send an ALERTING message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AssCmd(ch : LOGICCH; pdu: ASS_CMD_PDU)
ASP Type : DL_DatRqAssCmd
Derivation Path :
Comments : To send an ASSIGNMENT COMMAND message which is assigned in send statement.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AssCmp(ch : LOGICCH; pdu : ASS_COM_PDU)
ASP Type : DL_DatInAssCom
Derivation Path :
Comments : To match any received ASSIGNMENT COMPLETE message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AssFl_any_cau(ch: LOGICCH)
ASP Type : DL_DatInAssfl
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	AssgnFl_01	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AuthReq(ch: LOGICCH; pdu: AUTH_RQ_PDU)
ASP Type : DL_DatRqAuthRq
Derivation Path :
Comments : To send an AUTHENTICATION REQUEST message with default values.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : AuthRes(pdu: AUTH_RES_PDU)
ASP Type : DL_DatInAuthRes
Derivation Path :
Comments : To match any received AUTHENTICATION RESPONSE message which contains any SRES.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CallCfm(pdu: CALL_CO_PDU)
ASP Type : DL_DatInCallCo
Derivation Path :
Comments : To receive a CC CALL CONFIRMED message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CallProc(ch: LOGICCH; pdu: CALL_PROC_PDU)
ASP Type : DL_DatRqCallProc
Derivation Path :
Comments : To send a CALL PROCEEDING message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ChmmoAckRcv(ch : LOGICCH; msg : CHMMO_ACK_PDU)
ASP Type : DL_DatInChmmoAck
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	msg	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ChmmoReqSnd(ch : LOGICCH; msg : CHMMO_PDU)
ASP Type : DL_DatRqChmmo
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	msg	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ChRel(ch: LOGICCH; pdu: CH_REL_PDU)
ASP Type : DL_DatRqChRel
Derivation Path :
Comments : To send a CHANNEL RELEASE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ChReq(pdu: CH_RQ_PDU)
ASP Type : DL_RacInChRq
Derivation Path :
Comments : To receive a primitive containing a CHANNEL REQUEST message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
fn	?	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CMSerReq(pdu: CMS_RQ_PDU)
ASP Type : DL_EstInCmsRq
Derivation Path :
Comments : To receive a CM SERVICE REQUEST message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	?	
establish_mode	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ConnAckRcv(pdu: CONN_ACK_PDU)
ASP Type : DL_DatInConnAck
Derivation Path :
Comments : To receive a CONNECT ACKNOWLEDGE message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ConnAckSnd(ch : LOGICCH; pdu: CONN_ACK_PDU)
ASP Type : DL_DatRqConnAck
Derivation Path :
Comments : To send a CONNECT ACKNOWLEDGE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ConnRcv(pdu: CONN_PDU)
ASP Type : DL_DatInConn
Derivation Path :
Comments : To receive a CC CONNECT message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ConnSnd(ch: LOGICCH; pdu: CONN_PDU)
ASP Type : DL_DatRqConn
Derivation Path :
Comments : To send a CC CONNECT message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CphCmd(ch: LOGICCH; pdu: CPHM_CMD_PDU)
ASP Type : DL_DatRqCphmCmd
Derivation Path :
Comments : To send a CIPHERING MODE COMMAND message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CphCom(pdu: CPHM_COM_PDU)
ASP Type : DL_DatInCphmCom
Derivation Path :
Comments : To receive a CIPHERING MODE COMPLETE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DiscRcv(ch : LOGICCH; pdu: DISC_PDU)
ASP Type : DL_DatInDisc
Derivation Path :
Comments : To match a received DISCONNECT message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	any cause, no option IEs
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DiscSnd(ch : LOGICCH; pdu: DISC_PDU)
ASP Type : DL_DatRqDisc
Derivation Path :
Comments : To send a DISCONNECT message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DLEstInd(ch : LOGICCH)
ASP Type : DL_EstIn
Derivation Path :
Comments : Indication of a layer 2 connection establishment

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	
establish_mode	?	Norm
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DLRelInd_01
ASP Type : DL_RelIn
Derivation Path :
Comments : Layer 2 indication of the layer 2 connection has been released.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
release_mode	?	
outstanding_indicator	?	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DLRelInd_02
ASP Type : DL_RelIn
Derivation Path :
Comments : Layer 2 indication of the layer 2 connection has been released.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
release_mode	C_LocEndRel	
outstanding_indicator	?	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ESetupRcv(pdu: ESETUP_PDU)
ASP Type : DL_DatInESetup
Derivation Path :
Comments : To receive an emergency call SETUP message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : HndOvCmpRcv(ch : LOGICCH; msg : HO_COM_PDU)
ASP Type : DL_DatInHoCom
Derivation Path :
Comments : To receive any HANDOVER COMPLETE message on channel ch.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	msg	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : HndOvSnd(ch : LOGICCH; msg : HO_CMD_PDU)
ASP Type : DL_DatRqHoCmd
Derivation Path :
Comments : To send a HANDOVER COMMAND indicating a finely synchronised intra cell handover.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	msg	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : HndOvAcc_01
ASP Type : DL_RacInHoacc
Derivation Path :
Comments : To received any HANDOVER ACCESS message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0,
logic_ch	?	
msg	HandOverAcc_01	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : HndOvAccRcv(ch: LOGICCH; msg : HOACC_PDU)
ASP Type : DL_RacInHoacc
Derivation Path :
Comments : To received any HANDOVER ACCESS message on channel ‘ch‘.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0,
logic_ch	ch	
msg	msg	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : HndOvFlRcv(ch : LOGICCH; msg :HOFL_PDU)
ASP Type : DL_DatInHofl
Derivation Path :
Comments : To match a received HANDOVER FAILURE message containing any RR cause.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	msg	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ImmAss(ch: LOGICCH; pdu: IMMASS_PDU)
ASP Type : DL_UdatRqImmass
Derivation Path :
Comments : To send an IMMEDIATE ASSIGNMENT message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ImsiDet(pdu : IMSID_IN_PDU)
ASP Type : DL_EstInImsidIn
Derivation Path :
Comments : To match a received IMSI DETACH INDICATION message matching any MS classmark1 value and any mobile identity value

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	check not required
establish_mode	?	check not required
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : LocAcp(ch : LOGICCH; pdu : LUP_ACP_PDU)
ASP Type : DL_DatRqLupAcp
Derivation Path :
Comments : To send a LOCATION UPDATING ACCEPT message with a new mobile identity TMSI and location area.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : LocUp(ch: LOGICCH; pdu : LUP_RQ_PDU)
ASP Type : DL_EstInLupRq
Derivation Path :
Comments : To match a received LOCATION UPDATING REQUEST message containing location updating type = IMSI attach.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
establish_mode	?	no check required
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : LocRej(ch: LOGICCH; pdu : LUP_REJ_PDU)
ASP Type : DL_DatRqLupRej
Derivation Path :
Comments : To send a LOCATION UPDATING REJECT message containing the reject cause IMSI unknown in HLR.
Used var's:
ch

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : MDLRelReq(ch : LOGICCH)
ASP Type : MDL_RelRq
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
release_mode	C_LocEndRel	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : MsrRept(pdu : MSR_RPT_PDU)
ASP Type : DL_UdatInMsrRpt
Derivation Path :
Comments : To receive a measurement report without measurement results.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	?	SACCH
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : PgReq1(ch : LOGICCH; pgg : PGG; pg1_req_pdu: PG1_RQ_PDU)
ASP Type : DL_UdatRqPg1Rq
Derivation Path :
Comments : To send a PAGING REQUEST TYPE 1 message to the paging group indicated by the parameter pgg which is derived from system parameters. The PAGING REQUEST TYPE1 message requests normal paging mode, any channel and for the MS identity TSPX_TMSI.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
pgg	pgg	Paging group of the mobile to be paged.
msg	pg1_req_pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : PagingRes(pdu: PG_RES_PDU)
ASP Type : DL_EstInPgRes
Derivation Path :
Comments : To receive a PAGING RESPONSE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
establish_mode	?	
msg	pdu	PAGING RESPONSE
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : PhyInfo_01(ch: LOGICCH; ta : TA)
ASP Type : DL_DatRqPhyinfo
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	
msg	Phyinfor_01(ta)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ReleaseRcv(pdu: REL_PDU)
ASP Type : DL_DatInRel
Derivation Path :
Comments : To receive a CC RELEASE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : ReleaseSnd(ch: LOGICCH; pdu: REL_PDU)
ASP Type : DL_DatRqRel
Derivation Path :
Comments : To send a RELEASE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : RelComRcv(pdu: REL_COM_PDU)
ASP Type : DL_DatInRelCmp
Derivation Path :
Comments : To receive a RELEASE COMPLETE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : RelComSnd(ch : LOGICCH; pdu: REL_COM_PDU)
ASP Type : DL_DatRqRelCmp
Derivation Path :
Comments : To send a RELEASE COMPLETE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SetupRcv(pdu: SETUP_MO_PDU)
ASP Type : DL_DatInSetup
Derivation Path :
Comments : To receive a SETUP message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	?	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SetupSnd(ch: LOGICCH; setup : SETUP_MT_PDU)
ASP Type : DL_DatRqSetup
Derivation Path :
Comments : To send a SETUP message which is the input parameter of this ASP.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	setup	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SyncInfo(ch : LOGICCH; bcc : BCC; ncc : NCC)
ASP Type : DL_UdatRqSchinfo
Derivation Path :
Comments : To send SYNCHRONIZATION INFORMATION message with default parameters.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	SCH
msg	SyncInfor(bcc, ncc)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo1(ch : LOGICCH; cchd : CCHD; maxtx : B_2; txint : B_4; re : B_1)
ASP Type : DL_UdatRqSysinfo1
Derivation Path :
Comments : To send SYSTEM INFORMATION TYPE 1 in cell A for RR testing of GSM900.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf1(cchd, maxtx, txint, re)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo1_nh(ch : LOGICCH; ci : CI; mcc, mnc, lac : OCTETSTRING; ccd : CCD; co : CO; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1)
ASP Type : DL_UdatRqSysinfo1_nh
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf3(ci, mcc, mnc, lac, ccd, co, crh, mtmc, neci, maxtx, txint, re)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo2(ch: LOGICCH; bcchfl: NCD; maxtx : B_2; txint : B_4; re : B_1; nccp : NCCP)
ASP Type : DL_UdatRqSysinfo2
Derivation Path :
Comments : To send the default SYSTEM INFORMATION TYPE 2 message containing default neighbour cells description in cell B for GSM900.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf2(bcchfl, maxtx, txint, re, nccp)	Default system Information message 2

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo2bis(ch: LOGICCH; bcchflegsm: NCD)
ASP Type : DL_UdatRqSysinfo2bis
Derivation Path :
Comments : SYSTEM INFORMATION 2bis in cell A under EGSM with the ARFCN list = {988, 990, 1003}.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	BCCH
msg	SysInf2bis(bcchflegsm)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo3(ch : LOGICCH; ci : CI; mcc, mnc, lac : OCTETSTRING; ccd : CCD; co : CO; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1)
ASP Type : DL_UdatRqSysinfo3
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf3(ci, mcc, mnc, lac, ccd, co, crh, mtmc, neci, maxtx, txint, re)	Default System Information 3 with CCCH combined with SDCCH

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo4(ch : LOGICCH; mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1)
ASP Type : DL_UdatRqSysinfo4
Derivation Path :
Comments : To send a default SYSTEM INFORMATION TYPE 4 message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf4(mcc, mnc, lac, crh, mtmc, neci, maxtx, txint, re)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo4_CBMS(ch : LOGICCH; mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1; cbchd : CHD)

ASP Type : DL_UdatRqSysinfo4

Derivation Path :

Comments : To send a SYSTEM INFORMATION TYPE 4 message for SMSCB with the values or GSM 11.10, 34.3

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf4_CBMS(mcc, mnc, lac, crh, mtmc, neci, maxtx,txint, re, cbchd)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo5(sacch: LOGICCH; sysinfo5_pdu : SYSINFO5_PDU)

ASP Type : DL_UdatRqSysinfo5

Derivation Path :

Comments : To send a SYSTEM INFORMATION TYPE 5 message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	sacch	
msg	sysinfo5_pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo5bis(sacch: LOGICCH; sysinfo5bis_pdu : SYSINFO5bis_PDU)

ASP Type : DL_UdatRqSysinfo5bis

Derivation Path :

Comments : To send a SYSTEM INFORMATION TYPE 5bis message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	sacch	
msg	sysinfo5bis_pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo6(sacch: LOGICCH; sysinfo6_pdu : SYSINFO6_PDU)
ASP Type : DL_UdatRqSysinfo6
Derivation Path :
Comments : To send a SYSTEM INFORMATION TYPE 6 message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	sacch	
msg	sysinfo6_pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : TmsiReallocSnd(par : MI; ch : LOGICCH; mcc, mnc, lac : OCTETSTRING)
ASP Type : DL_DatRqTmsireCmd
Derivation Path :
Comments : To send a TMSI REALLOCATION COMMAND message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	TmsiReallocCmd(par, mcc, mnc, lac)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : TmsiReallocCmp(ch : LOGICCH)
ASP Type : DL_DatInTmsireCom
Derivation Path :
Comments : To receive a TMSI REALLOCATION COMPLETE message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	TmsiReallocComp	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DLUdatInd(ch:LOGICCH)
ASP Type : DL_UdatIn
Derivation Path :
Comments : Indication of filler frame

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : DLUdatInd_01
ASP Type : DL_UdatIn
Derivation Path :
Comments : Detection of any filler frame

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	?	
fn	?	

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Alerting_01(Ti :TI)
PDU Type : ALERT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An ALERTING message containing mandatory IE's only to be sent to the MS.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		transaction identifier
ccpd	'0011'B		CC protocol discriminator
mt	'00000001'B		message type
fie	−		facility
pi	−		progress indicator
uu	−		user−user
ssvi	−		SS version indicator

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AlertingInd_01(ti : TI)
PDU Type : ALERT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match an received ALERTING message

Field Name	Field Value	Field Encoding	Comments
ti	ti		'1000'B
ccpd	'0011'B		CC protocol discriminator
mt	'0?000001'B		message type
fie	*		facility
pi	−		progress indicator
uu	*		user−user
ssvi	*		SS version indicator

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AsgnCmd_fh(slot : SN; tsc : TSC; chtype : CH_TDMA; par_pwlvl : INTEGER; maio : MAIO; hsn : HSN; frql : FRQL; Cchd : CCHD; chmod : CHMOD; ma : MA; Cphms : CPHMS)

PDU Type : ASS_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An ASSIGNMENT COMMAND message for hopping without starting time.

Field Name	Field Value	Field Encoding	Comments
ski	’0000’B		
rrpd	’0110’B		
mt	’00101110’B		
ch1d_at	ChDescrp_fh(chtype, slot, tsc, maio, hsn)		TCH/F + ACCHs hopping
pcmd	Pcmd_19(INT_TO_BIT(par_pwlvl, 5))		
frql_at	frql		
cchd	Cchd		
ch1mod	chmod		
ch2d_at	–		
ch2mod	–		
ma_at	ma		
strt	–		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	Cphms		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AsgnCmd_nfh(type : CH_TDMA; slot : SN; tsc : TSC; par_pwlvl, arfcn : INTEGER; cchd : CCHD; chmod : CHMOD; strt : STRT; cphms : CPHMS)

PDU Type : ASS_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An ASSIGNMENT COMMAND message. The channel mode and type are specified as parameters.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101110'B		
ch1d_at	ChDescrp_nfh(type, slot, tsc, arfcn)		
pcmd	Pcmd_19(INT_TO_BIT(par_pwlvl, 5))		
frql_at	–		
cchd	cchd		
ch1mod	chmod		
ch2d_at	–		
ch2mod	–		
ma_at	–		
strt	strt		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	cphms		

Detailed Comments : used iin CC testing

PDU Constraint Declaration

Constraint Name : AsgnCmp_02

PDU Type : ASS_COM_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An ASSIGNMENT COMPLETE message containing any cause.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00101001'B		message type
rrcau	?		any cause value

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AssgnFl_01
PDU Type : ASSFL_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments :

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00101111'B		message type
rrcau	?		RR cause

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AuthRequest(cksn: B_3; rand : RAND)
PDU Type : AUTH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An AUTHENTICATION REQUEST message containing default ciphering key sequence number and default challenge RAND from PIXIT.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'00010010'B		
shoct	'0000'B		
cphksn	CphKeySN_07(cksn)		
rand	rand		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AuthResponse
PDU Type : AUTH_RES_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An AUTHENTICATION RESPONSE message matching any SRES.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'0?010100'B		
sres	?		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CallConfirm_01(ti : TI)
PDU Type : CALL_CO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : a CALL CONFIRMED message with TI parameterized

Field Name	Field Value	Field Encoding	Comments
ti	ti		
ccpd	'0011'B		CC protocol discriminator
mt	'0?001000'B		message type
bcri	*		BC repeat indicator
bcap1	*		bearer capab.
bcap2	*		bearer capab.
cau	–		
cccap	*		cc capabilities O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CallConfirm_04(ti : TI; Bcap1 : BCAP)
PDU Type : CALL_CO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : a CALL CONFIRMED message with TI and bcap1 parameterized

Field Name	Field Value	Field Encoding	Comments
ti	ti		
ccpd	'0011'B		CC protocol discriminator
mt	'0?001000'B		message type
bcri	–		BC repeat indicator
bcap1	Bcap1		bearer capab.
bcap2	–		bearer capab.
cau	–		
cccap	*		cc capabilities O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CallConfirm_20(ti : TI)
PDU Type : CALL_CO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : a CALL CONFIRMED message without bearer capability information elements

Field Name	Field Value	Field Encoding	Comments
ti	ti		
ccpd	'0011'B		CC protocol discriminator
mt	'0?001000'B		message type
bcri	−		BC repeat indicator
bcap1	−		bearer capab.
bcap2	−		bearer capab.
cau	−		cause
cccap	*		cc capabilities O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CallConfirm_Efr_20(ti : TI)
PDU Type : CALL_CO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : a CALL CONFIRMED message for EFR Mobiles

Field Name	Field Value	Field Encoding	Comments
ti	ti		
ccpd	'0011'B		CC protocol discriminator
mt	'0?001000'B		message type
bcri	−		BC repeat indicator
bcap1	*		bearer capab.
bcap2	−		bearer capab.
cau	−		cause
cccap	*		cc capabilities O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CallProced_03
PDU Type : CALL_PROC_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An CALL PROCEEDING message templet used for CallProcGen

Field Name	Field Value	Field Encoding	Comments
ti	?		transaction identifier
ccpd	'0011'B		CC protocol discriminator
mt	'00000010'B		message type
bcri	–		BC repeat indicator
bcap1	–		bearer capab.
bcap2	–		bearer capab.
fie	–		facility
pi	–		progress indicator

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChmomoAck_01(chmd : CHMOD; chd: CHD)
PDU Type : CHMMO_ACK_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments :

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00010111'B		message type
chd	chd		TCH/F
chmod	chmd		channel mode

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChmomoReq_01(chmd : CHMOD; chd : CHD)
PDU Type : CHMMO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : the channel being modified is default full rate traffic channel.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00010000'B		message type
chd	chd		TCH/F
chmod	chmd		channel mode

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRelease_01
PDU Type : CH_REL_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CHANNEL RELEASE message with RR cause = normal event

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00001101'B		
rrcau	'00000000'B		normal event

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRequest_02
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received CHANNEL REQUEST message

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	'????????'B		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRequest_04
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match the received CHANNEL REQUEST message which originates a call (establishment cause = '111'B), "Originating call and TCH/F is needed, or IMSI detach, SMS or SS procedures that can be completed with an SDCCH and NECI set to 0".

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	'111?????'B		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRequest_15
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match the received CHANNEL REQUEST message with establishment cause = '111'B or '0100'B or '0101'B −−− initiate outgoing call or SDCCH other procedures.

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	('111?????'B, '0100????'B, '0101????'B, '101?????'B, '0001????'B)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRequest_16
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match the received CHANNEL REQUEST message with establishment cause = '101'B for emergency call.

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	'101?????'B		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRequest_17
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CHANNEL REQUEST message containing establishment cause = 100, 0010, 0011, 0001

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	('100?????'B, '0010????'B, '0011????'B, '0001????'B)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CMServiceReq_01
PDU Type : CMS_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received CM SERVICE REQUEST message

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'0?100100'B		
cphksn	?		
svtype	?		
msclm	?		
mi	?		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CMServiceReq_04
PDU Type : CMS_RQ_PDU
Derivation Path : CMServiceReq_01.
Encoding Rule Name :
Encoding Variation :
Comments : To match a received CM SERVICE REQUEST message containing CM service type = "Mobile originating call establishment or packet mode connection establishment" or " emergency call establishment".

Field Name	Field Value	Field Encoding	Comments
svtype	('0001'B, '0010'B)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CMServiceReq_06(cksn : B_3)
PDU Type : CMS_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received CM SERVICE REQUEST message for emergency call with TMSI and correct CKSN.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'0?100100'B		
cphksn	CphKeySN_01(cksn)		default CKSN
svtype	C_CMServiceTypeE		emergency call.
msclm	?		
mi	MiTmsi_01		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Connect_01(ti : TI)
PDU Type : CONN_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : CC CONNECT message matching any received value

Field Name	Field Value	Field Encoding	Comments
ti	ti		'1000'B
ccpd	'0011'B		CC protocol discriminator
mt	'0?000111'B		
fie	*		
pi	–		
cnn	–		connected number
cns	*		connected subaddress
uu	*		
ssvi	*		SS version

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Connect_02(Ti : TI)
PDU Type : CONN_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : CC CONNECT message containing mandatory IE's only.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		transaction identifier
ccpd	'0011'B		CC protocol discriminator
mt	'00000111'B		
fie	–		
pi	–		
cnn	–		connected number
cns	–		connected subaddress
uu	–		
ssvi	–		SS version

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ConnectAck_01(ti : TI)
PDU Type : CONN_ACK_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CONNECT ACKNOWLEDGE message n –> ms

Field Name	Field Value	Field Encoding	Comments
ti	ti		'0000'B
ccpd	'0011'B		CC protocol discriminator
mt	'00001111'B		message type

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ConnectAck_02(Ti : TI)
PDU Type : CONN_ACK_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CONNECT ACKNOWLEDGE message

Field Name	Field Value	Field Encoding	Comments
ti	Ti		transaction identifier
ccpd	'0011'B		CC protocol discriminator
mt	'0?001111'B		message type

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CphModeCmd_01
PDU Type : CPHM_CMD_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : CIPHERING MODE COMMAND message, the ciphering algorithm is specified by PIXIT.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00110101'B		
cph_res	CiphRes_01		IMEISV not included
cphms	CphMod_01		ciphering "on"

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CphModeCmp_01
PDU Type : CPHM_COM_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received CIPHERING MODE COMPLETE message

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00110010'B		message type
mei	*		mobile equipment identity

Detailed Comments :

PDU Constraint Declaration

Constraint Name : DisconnS(Ti : TI; cau : CAU; progind : PI; uuinf : UU)
PDU Type : DISC_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A DISCONNECT message that contains cause = normal clearing and progress indicator = #8.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
ccpd	’0011’B		
mt	’00100101’B		
cau	cau		
fie	–		
pi	progind		
uu	uuinf		
ssvi	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : DisconnR(Ti : TI; cau : CAU)
PDU Type : DISC_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A default DISCONNECT message matching parametrised value ms −> n.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
ccpd	’0011’B		
mt	’0?100101’B		
cau	cau		
fie	*		
pi	*		
uu	*		
ssvi	*		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : HandOverAcc_01
PDU Type : HOACC_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received HANDOVER ACCESS message

Field Name	Field Value	Field Encoding	Comments
horf	?		handover reference

Detailed Comments :

PDU Constraint Declaration

Constraint Name : HandOverAcc_02(horef : HORF)
PDU Type : HOACC_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match a received HANDOVER ACCESS message with handover reference ‘ horef‘.

Field Name	Field Value	Field Encoding	Comments
horf	horef		handover reference

Detailed Comments :

PDU Constraint Declaration

Constraint Name : HandOverCmd_fh(type : CH_TDMA; slot : SN; tsc : TSC; ncc, bcc : B_3; sdccharfcn, mod, hsn : INTEGER; horf : HORF; pwlvl : B_5; synchi : SYNCHI; frql1, frql2 : FRQL; cchd : CCHD; chmod : CHMOD; frqchs : FRQCHS; ma : MA; rtdif : TDIF; ta : TA; cphms : CPHMS)

PDU Type : HO_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : A derived HANDOVER COMMAND containing TCH/F_FH in non synchronized new CELL A.

Field Name	Field Value	Field Encoding	Comments
ski	’0000’B		
rrpd	’0110’B		
mt	’00101011’B		
cd	CellDescrp(ncc, bcc, sdccharfcn)		
ch1d_at	ChDescrp_fh(type, slot, tsc, INT_TO_BIT((TSPX_MAIO MOD mod), 6), INT_TO_BIT(hsn, 6))		
horf	horf		Hand over reference
pcmd	Pcmd_20(pwlvl)		power level
synchi	synchi		
frqsl_at	frql1		
frql_at	frql2		
cchd	cchd		
ch1mod	chmod		
ch2d_at	–		
ch2mod	–		
frqchs_at	frqchs		
ma_at	ma		
strt	–		
rtdif	rtdif		
ta	ta		
frqsl_bt	–		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	cphms		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : HandOverCmd_nfh(type : CH_TDMA; slot : SN; tsc : TSC; ncc, bcc : B_3; sdccharfcn, tcharfcn : INTEGER; horf : HORF; pwlvl : B_5; synchi : SYNCHI; chmod : CHMOD; rtdif : TDIF; ta : TA; cphms : CPHMS)

PDU Type : HO_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An HANDOVER COMMAND indicating finely synchronised intra cell handover.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101011'B		
cd	CellDescrp(ncc, bcc, sdccharfcn)		
ch1d_at	ChDescrp_nfh(type, slot, tsc, tcharfcn)		
horf	horf		from PIXIT
pcmd	Pcmd_20(pwlvl)		
synchi	synchi		
frqsl_at	–		
frql_at	–		
cchd	–		
ch1mod	chmod		
ch2d_at	–		
ch2mod	–		
frqchs_at	–		
ma_at	–		
strt	–		
rtdif	rtdif		
ta	ta		
frqsl_bt	–		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	cphms		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : HandOverCmp_20
PDU Type : HO_COM_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A basic received constraint for HANDOVER COMPLETE message.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00101100'B		message type
rrcau	'00000000'B		RR cause: normal event
motdif	*		mobile observed time difference

Detailed Comments :

PDU Constraint Declaration

Constraint Name : HandOvFail_01
PDU Type : HOFL_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A HANDOVER FAILURE message matching any RR cause.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101000'B		
rrcau	?		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ImmAsgn_01Def(Rr: BITSTRING; Fn: FN; sub : B_2; slot : SN; tsc : TSC; par_arfcn : INTEGER; ta : TA)

PDU Type : IMMASS_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An IMMEDIATE ASSIGNMENT message to assign SDCCH/4 channel.

Field Name	Field Value	Field Encoding	Comments
l2_pl	’2D’O		L2 pseudo length = 11 octets for non hopping, no starting time
ski	’0000’B		
rrpd	’0110’B		
mt	’00111111’B		
shoct	’0000’B		
pm	Pm(C_normal_paging)		normal mode
chd	ChDescrp_nfh(INT_TO_BIT((4 + BIT_TO_INT(sub)), 5), slot, tsc, par_arfcn)		SDCCH/4 subchannel
rqr	Rqr2(Rr, Fn)		
ta	ta		0 time advance IE
ma	MoblAllc_01		non hopping
strt	–		
iaroct	IaRestOct_01		11 octets of ’2B’O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ImmAsgn_221(Rr : BITSTRING; Fn : FN; ts_ccch : SN; tsc : TSC; par_chtype : CH_TDMA; maio : MAIO; hsn : HSN; ta : TA; ma : MA; iel : LENGTH)

PDU Type : IMMASS_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An IMMEDIATE ASSIGNMENT message.

Field Name	Field Value	Field Encoding	Comments
l2_pl	OC_IntToOct(((11 + OC_OctToInt(iel)) * 4) + 1, 1)		L2 pseudo length = 14 octets
ski	’0000’B		
rrpd	’0110’B		
mt	’00111111’B		
shoct	’0000’B		
pm	Pm(C_normal_paging)		normal mode
chd	ChDescrp_fh(par_chtype, ts_ccch, tsc, maio, hsn)		
rqr	Rqr2(Rr, Fn)		
ta	ta		
ma	ma		
strt	–		
iaroct	IaRestOct(iel)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ImmAsgn_nfh(Rr: BITSTRING; Fn: FN; type: CH_TDMA; slot : SN; tsc : TSC; ta : TA; arfcn: INTEGER; pgm : B_2)

PDU Type : IMMASS_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An IMMEDIATE ASSIGNMENT message to assign TCH/F or TCH/H channel.

Field Name	Field Value	Field Encoding	Comments
l2_pl	’2D’O		L2 pseudo length = 11 octets for non hopping, no starting time
ski	’0000’B		
rrpd	’0110’B		
mt	’00111111’B		
shoct	’0000’B		
pm	Pm(pgm)		paging mode
chd	ChDescrp_nfh(type, slot, tsc, arfcn)		Traffic channel, non hopping
rqr	Rqr2(Rr, Fn)		
ta	ta		
ma	MoblAllc_01		non hopping
strt	–		
iaroct	IaRestOct_01		11 octets of ’2B’O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ImmAsgn_r13(Rr: BITSTRING; Fn: FN; chd : CHD; ta : TA)
PDU Type : IMMASS_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An IMMEDIATE ASSIGNMENT message to assign TCH/F channel with a time advance of 30 bit periods.

Field Name	Field Value	Field Encoding	Comments
l2_pl	’2D’O		L2 pseudo length = 11 octets for non hopping, no starting time
ski	’0000’B		
rrpd	’0110’B		
mt	’00111111’B		
shoct	’0000’B		
pm	Pm(C_normal_paging)		normal mode
chd	chd		TCH/F + ACCHs
rqr	Rqr2(Rr, Fn)		
ta	ta		30 bit periods of time advance
ma	MoblAllc_01		non hopping
strt	–		
iaroct	IaRestOct_01		11 octets of ’2B’O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ImsiDetach_30(par:MI)
PDU Type : IMSID_IN_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : IMSI DETACH INDICATION message matching any MS classmark1 value and given IMSI.

Field Name	Field Value	Field Encoding	Comments
ski	’0000’B		
mmpd	’0101’B		
mt	’0?000001’B		
msclm	?		any mobile station classmark1
mi	par		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : LocUpdtAcp_01(newmi: MI; mcc, mnc, lac : OCTETSTRING)
PDU Type : LUP_ACP_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To assign a new TMSI with or without mobile identity.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'00000010'B		
lai	LocAreaId(mcc, mnc, lac)		location area identification
mi	newmi		
fop	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : LocUpdtReq_01(locup : B_2)
PDU Type : LUP_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A LOCATION UPDATING REQUEST message containing location updating type.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'0?001000'B		
cphksn	?		no check required
lutype	LocUpType(locup)		
lai	?		no check required
msclm	?		no check required
mi	?		no check required

Detailed Comments :

PDU Constraint Declaration

Constraint Name : LocUpdtRej_01(par:REJCAU)
PDU Type : LUP_REJ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A LOCATION UPDATING REJECT message.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'00000100'B		
rejcau	par		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : MsrReport_02
PDU Type : MSR_RPT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A measurement report match any received MSR_RPT_PDU.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		RR protocol discriminator
mt	'00010101'B		measurement message type
msrr	MsrResult_02		containing any value

Detailed Comments :

PDU Constraint Declaration

Constraint Name : MsrReport_03e(par_measres: MSRR)
PDU Type : MSR_RPT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A measurement report containing results for 6 strongest carriers.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		RR protocol discriminator
mt	'00010101'B		measurement message type
msrr	par_measres		6 strongest carriers

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PagingRes_01
PDU Type : PG_RES_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received PAGING RESPONSE message.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00100111'B		
shoct	'0000'B		
cphksn	?		no check requirement
msclm	?		no check requirement
mi	?		no check requirement

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PagingRes_03(cksn : B_3)
PDU Type : PG_RES_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match a received PAGING RESPONSE message with default TMSI, CKSN and classmark2.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00100111'B		
shoct	'0000'B		
cphksn	CphKeySN_01(cksn)		TSPX_CKSNDef
msclm	ClassMark2		
mi	MiTmsi_01		TSPX_TMSI

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PagingRes_r02
PDU Type : PG_RES_PDU
Derivation Path : PagingRes_01.
Encoding Rule Name :
Encoding Variation :
Comments : To match the received PAGING RESPONSE message containing IMSI of the IUT for RR tests.

Field Name	Field Value	Field Encoding	Comments
mi	MiImsi_01		same as in the PgRq

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PgReqTp1_01
PDU Type : PG1_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An PAGING REQUEST TYPE1 message requesting any channel with normal paging mode, the mobile identity is TMSI.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'25'O		L2 pseudo length = 9
ski	'0000'B		
rrpd	'0110'B		
mt	'00100001'B		
chn_m1_2	Chneed_01		any channel
pm	Pm(C_normal_paging)		normal paging
mi1	MiTmsi_01		TSPX_TMSI
mi2	–		
p1roct	'2B2B2B2B2B2B2B2B2B2B2B2B2B2B'O		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PgReqTp1_05
PDU Type : PG1_RQ_PDU
Derivation Path : PgReqTp1_01.
Encoding Rule Name :
Encoding Variation :
Comments : PAGING REQUEST TYPE1 message with mobile identity 1 being IMSI of the MS.

Field Name	Field Value	Field Encoding	Comments
l2_pl	OC_IntToOct(((3 + OC_LengthOf(MiImsi_01)) * 4) + 1, 1)		
mi1	MiImsi_01		
p1roct	OC_SubOctet('2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B'O, 19 − OC_LengthOf(MiImsi_01))		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PgReqTp1Norm
PDU Type : PG1_RQ_PDU
Derivation Path : PgReqTp1_01.
Encoding Rule Name :
Encoding Variation :
Comments : PAGING REQUEST TYPE1 message containing normal paging mode, used as paging filling message.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'15'O		L2 pseudo length = 5
mi1	MiImsi_02		no identity
p1roct	'2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B'O		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : PgReqTp1Reorg
PDU Type : PG1_RQ_PDU
Derivation Path : PgReqTp1_01.
Encoding Rule Name :
Encoding Variation :
Comments : PAGING REQUEST TYPE1 message containing paging reorganisation mode, used as paging filling message.

Field Name	Field Value	Field Encoding	Comments
l2_pl	OC_IntToOct(((3 + OC_LengthOf(MiImsi_02)) * 4) + 1, 1)		
pm	Pm(C_reorg_paging)		paging reorganisation
mi1	MiImsi_02		no identity
p1roct	OC_SubOctet('2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B'O, 19 − OC_LengthOf(MiImsi_02))		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Phyinfor_01(ta : TA)
PDU Type : PHYINFO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments :

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00101101'B		message type
ta	ta		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Release_08(Ti : TI)
PDU Type : REL_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CC RELEASE message with mandatory IE's only used in structured procedures test .

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
ccpd	'0011'B		CC protocol discriminator
mt	'00101101'B		message type
cau	−		cause
cau2	−		second cause
fie	−		facility
uu	−		user−use
ssvi	−		SS version

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Release_10(Ti : TI)
PDU Type : REL_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CC RELEASE message used to match any received CC RELEASE message with a controllable TI.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
ccpd	'0011'B		CC protocol discriminator
mt	'0?101101'B		message type
cau	*		cause
cau2	*		second cause
fie	*		facility
uu	*		user−use
ssvi	*		SS version

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ReleaseCmp_03(Ti : TI)
PDU Type : REL_COM_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received RELEASE message.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
cc_sspd	'0011'B		CC protocol discriminator
mt	'0?101010'B		
cau	*		
fie	*		facility
uu	*		user−user information
ssvi	*		SS version

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ReleaseCmp_08(Ti :TI)
PDU Type : REL_COM_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A RELEASE COMPLETE message containing mandatory IEs only. n −> ms

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
cc_sspd	’0011’B		CC protocol discriminator
mt	’00101010’B		
cau	−		
fie	−		facility
uu	−		user−user information
ssvi	−		SS version

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_01(ti : TI)
PDU Type : SETUP_MT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message containing speech bearer capability. n −> ms.

Field Name	Field Value	Field Encoding	Comments
ti	ti		’0000’B
ccpd	’0011’B		
mt	’00000101’B		message type
bcri	−		
bcap1	EFR_Bcap_Speech_MT		speech
bcap2	−		
fie	−		
pi	−		
sig	−		
cgpn	−		
cgps	−		
cdpn	−		
cdps	−		
llcri	−		
llcmp1	−		
llcmp2	−		
hlcri	−		
hlcmp1	−		
hlcmp2	−		
uu	−		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_02(bcap: BCAP)
PDU Type : SETUP_MO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message containing speech bearer capability. ms −> n

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	'0011'B		
mt	'0?000101'B		message type
bcri	−		
bcap1	bcap		speech full rate only or dual rate
bcap2	−		
fie	*		
cgps	*		
cdpn	?		
cdps	*		
llcri	*		
llcmp1	*		
llcmp2	*		
hlcri	*		
hlcmp1	*		
hlcmp2	*		
uu	*		
ssvi	*		SS version ms −> n
clirsup	*		CLIR suppression ms −> n
clirinv	*		CLIR invocation ms −> n
cccap	*		call control capabilities ms−>n

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_03(ti : TI)
PDU Type : SETUP_MT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message with mandatory IE's only. This is default for BIBO testing.

Field Name	Field Value	Field Encoding	Comments
ti	ti		'0000'B
ccpd	'0011'B		
mt	'00000101'B		message type
bcri	–		
bcap1	–		
bcap2	–		
fie	–		
pi	–		
sig	–		
cgpn	–		
cgps	–		
cdpn	–		
cdps	–		
llcri	–		
llcmp1	–		
llcmp2	–		
hlcri	–		
hlcmp1	–		
hlcmp2	–		
uu	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ESetupBcap(bcap: BCAP)
PDU Type : ESETUP_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A Emergency call SETUP message with speech bearer capability. ms –> n.

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	'0011'B		
mt	'0?001110'B		message type
bcap	bcap		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ESetupNoBcap
PDU Type : ESETUP_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A Emergency call SETUP message without speech bearer capability. ms −> n.

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	’0011’B		
mt	’0?001110’B		message type
bcap	−		Full Rate Speech by default

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_10_UDI(bcap: BCAP; llc1 : LLCMP; hlc1 : HLCMP)
PDU Type : SETUP_MO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message containing bearer capability supported by the mobile station, used for Unrestricted digital information transfer capability. ms −> n.

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	’0011’B		
mt	’0?000101’B		message type
bcri	−		
bcap1	bcap		
bcap2	−		
fie	*		
cgps	*		calling party subaddr. O OCTETSTRING [2..23]
cdpn	?		called party BCD number ms −> n M OCTETSTRING[2..23]
cdps	*		called party subaddr. O OCTETSTRING [2..23]
llcri	−		LLC repeat indicator O OCTETSTRING [1]
llcmp1	llc1		low layer compatib.1 O OCTETSTRING [2..15]
llcmp2	−		low layer compatib.2 n −> ms C; ms −> n O
hlcri	−		HLC repeat indicator O OCTETSTRING [1]
hlcmp1	hlc1 IF_PRESENT		high layer compat.1 O OCTETSTRING [2..5]
hlcmp2	−		high layer compat. 2 n −> ms C; ms −> n O
uu	*		user−user information O OCTETSTRING [3..35]
ssvi	*		SS version ms −> n O OCTETSTRING [2..3]
clirsup	*		CLIR suppression ms −> n C
clirinv	*		CLIR invocation ms −> n O
cccap	CallCntrlCap IF_PRESENT		call control capabilities ms−>n O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_10_3100(bcap: BCAP; llc1 : LLCMP; hlc1 : HLCMP)
PDU Type : SETUP_MO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message containing bearer capability supported by the mobile station. used for 3.1 kHz audio information transfer capability. ms −> n.

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	’0011’B		
mt	’0?000101’B		message type
bcri	−		
bcap1	bcap		
bcap2	−		
fie	*		
cgps	*		calling party subaddr. O OCTETSTRING [2..23]
cdpn	?		called party BCD number ms −> n M OCTETSTRING[2..23]
cdps	*		called party subaddr. O OCTETSTRING [2..23]
llcri	−		LLC repeat indicator O OCTETSTRING [1]
llcmp1	llc1 IF_PRESENT		low layer compatib.1 O OCTETSTRING [2..15]
llcmp2	−		low layer compatib.2 n −> ms C; ms −> n O
hlcri	−		HLC repeat indicator O OCTETSTRING [1]
hlcmp1	hlc1 IF_PRESENT		high layer compat.1 O OCTETSTRING [2..5]
hlcmp2	−		high layer compat. 2 n −> ms C; ms −> n O
uu	*		user−user information O OCTETSTRING [3..35]
ssvi	*		SS version ms −> n O OCTETSTRING [2..3]
clirsup	*		CLIR suppression ms −> n C
clirinv	*		CLIR invocation ms −> n O
cccap	CallCntrlCap IF_PRESENT		call control capabilities ms−>n O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_11_3100(bcri1: RPI; bcap, bcap2nd: BCAP; llc1, llc2 : LLCMP; llcri1, hlcri1 : RPI ; hlc1, hlc2 : HLCMP)

PDU Type : SETUP_MO_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : A SETUP message containing two bearer capabilities supported by the mobile station, used for 3.1 kHz audio information transfer capability. ms −> n.

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	'0011'B		
mt	'0?000101'B		message type
bcri	bcri1		
bcap1	bcap		
bcap2	bcap2nd		
fie	*		
cgps	*		calling party subaddr. O OCTETSTRING [2..23]
cdpn	?		called party BCD number ms −> n M OCTETSTRING[2..23]
cdps	*		called party subaddr. O OCTETSTRING [2..23]
llcri	llcri1 IF_PRESENT		LLC repeat indicator O OCTETSTRING [1]
llcmp1	llc1 IF_PRESENT		low layer compatib.1 O OCTETSTRING [2..15]
llcmp2	llc2 IF_PRESENT		low layer compatib.2 n −> ms C; ms −> n O
hlcri	hlcri1 IF_PRESENT		HLC repeat indicator O OCTETSTRING [1]
hlcmp1	hlc1 IF_PRESENT		high layer compat.1 O OCTETSTRING [2..5]
hlcmp2	hlc2 IF_PRESENT		high layer compat. 2 n −> ms C; ms −> n O
uu	*		user−user information O OCTETSTRING [3..35]
ssvi	*		SS version ms −> n O OCTETSTRING [2..3]
clirsup	*		CLIR suppression ms −> n C
clirinv	*		CLIR invocation ms −> n O
cccap	CallCntrlCap IF_PRESENT		call control capabilities ms−>n O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_20(par: BCAP; ti : TI)
PDU Type : SETUP_MT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message containing bearer capability supported by the mobile station.

Field Name	Field Value	Field Encoding	Comments
ti	ti		’0000’B
ccpd	’0011’B		
mt	’00000101’B		message type
bcri	–		
bcap1	par		
bcap2	–		
fie	–		
pi	–		
sig	–		
cgpn	–		
cgps	–		
cdpn	–		
cdps	–		
llcri	–		
llcmp1	–		
llcmp2	–		
hlcri	–		
hlcmp1	–		
hlcmp2	–		
uu	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : Setup_21(par1 : RPI; par2, par3 : BCAP; ti : TI)
PDU Type : SETUP_MT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SETUP message containing two bearer capabilities supported by the mobile station.

Field Name	Field Value	Field Encoding	Comments
ti	ti		'0000'B
ccpd	'0011'B		
mt	'00000101'B		message type
bcri	par1		
bcap1	par2		
bcap2	par3		
fie	–		
pi	–		
sig	–		
cgpn	–		
cgps	–		
cdpn	–		
cdps	–		
llcri	–		
llcmp1	–		
llcmp2	–		
hlcri	–		
hlcmp1	–		
hlcmp2	–		
uu	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SetupInd_01
PDU Type : SETUP_MO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match any received SETUP message

Field Name	Field Value	Field Encoding	Comments
ti	TI_09		
ccpd	'0011'B		
mt	'0?000101'B		
bcri	*		BC repeat indicator
bcap1	?		bearer capab.
bcap2	*		bearer capab.
fie	*		facility
cgps	*		calling party subaddr.
cdpn	?		called party BCD number
cdps	*		called party subaddr.
llcri	*		LLC repeat indicator
llcmp1	*		low layer compatib.1
llcmp2	*		low layer compatib.2
hlcri	*		HLC repeat indicator
hlcmp1	*		high layer compat.1
hlcmp2	*		high layer compat.
uu	*		user−user information
ssvi	*		SS version
clirsup	*		CLIR suppression
clirinv	*		CLIR invocation
cccap	CallCntrlCap IF_PRESENT		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SyncInfor(bcc : BCC; ncc : NCC)
PDU Type : SCHINFO_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments :

Field Name	Field Value	Field Encoding	Comments
ncc	ncc		PLMN colour code
bcc	bcc		BS colour code
t1	?		
t2	?		
t3_	?		

Detailed Comments : 1. The values of t1, t2 and t3_ shall be correctly set by the L2 simulation module when this message is sent to air on the synchronization channel. It is assumed that there is a set of timebase counters in the L2 module, these counters run continuously and keep the correct timeslot number SN and TDMA frame number FN as long as the test system is poweron , the T1, T2, T3' (values for t1, t2 and t3_) can be derived from these counters.

PDU Constraint Declaration

Constraint Name : SysInf1(cchd : CCHD; maxtx : B_2; txint : B_4; re : B_1)
PDU Type : SYSINFO1_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : Default parameters for cell A in RR testing of GSM 900

Field Name	Field Value	Field Encoding	Comments
l2_pl	'55'O		L2 pseudo length = 21
ski	'0000'B		
rrpd	'0110'B		
mt	'00011001'B		
cchd	cchd		ARFCNs: 20, 30, 50, 70
rachcp	RachCntrlPara(maxtx, txint, re)		rach control parameters
si1roct	'2B'O		SI1 rest octets

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf2(bcchfl: NCD; maxtx : B_2; txint : B_4; re : B_1; nccp : NCCP)
PDU Type : SYSINFO2_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : SYSTEM INFORMATION TYPE2 containing default neighbour cells description

Field Name	Field Value	Field Encoding	Comments
l2_pl	'59'O		L2 pseudo length = 22
ski	'0000'B		
rrpd	'0110'B		
mt	'00011010'B		
bcchfl	bcchfl		
nccp	nccp		NCC to be monitored
rachcp	RachCntrlPara(maxtx, txint, re)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf2bis(bcchflegsm: NCD)
PDU Type : SYSINFO2bis_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : SYSTEM INFORMATION 2bis in cell A under EGSM with the ARFCN list = {988, 990, 1003}.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'55'O		L2 pseudo length
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00000010'B		message type
xbcchfl	bcchflegsm		ARFCN: 988, 990, 1003
rachcp	RachCntrlPara_r01		rach control parameters
si2bisroct	'00'O		SI2bis rest octet

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf3(ci : CI; mcc, mnc, lac : OCTETSTRING; ccd : CCD; co : CO; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1)

PDU Type : SYSINFO3_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : CCCH combined or not with SDCCH

Field Name	Field Value	Field Encoding	Comments
l2_pl	'49'O		L2 pseudo length = 18
ski	'0000'B		
rrpd	'0110'B		
mt	'00011011'B		
ci	ci		
lai	LocAreaId(mcc, mnc, lac)		MCC= 001, MNC= 01, LAC= 0001
ccd	ccd		
co	co		DTX not allowed
csp	CellSelPara(crh, mtmc, neci)		
rachcp	RachCntrlPara(maxtx, txint, re)		
si3roct	SI3RestOctetDef		PI=0, C2 parameters not present

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf4(mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1)

PDU Type : SYSINFO4_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : A SYSTEM INFORMATION TYPE 4 message containing default values

Field Name	Field Value	Field Encoding	Comments
l2_pl	'31'O		L2 pseudo length = 12
ski	'0000'B		
rrpd	'0110'B		
mt	'00011100'B		
lai	LocAreaId(mcc, mnc, lac)		
csp	CellSelPara(crh, mtmc, neci)		
rachcp	RachCntrlPara(maxtx, txint, re)		
cbchd	–		
cbchma	–		
si4roct	SI4RestOctetDef		PI=0, C2 parameters not present

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf4_CBMS(mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci : B_1; maxtx : B_2; txint : B_4; re : B_1; cbchd : CHD)

PDU Type : SYSINFO4_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : To send a SYSTEM INFORMATION TYPE 4 message for SMSCB with the values or GSM 11.10, 34.3

Field Name	Field Value	Field Encoding	Comments
l2_pl	'41'O		L2 pseudo length = 18
ski	'0000'B		
rrpd	'0110'B		
mt	'00011100'B		
lai	LocAreaId(mcc, mnc, lac)		MCC=001'H, MNC=01'H, LAC=0001'H
csp	CellSelPara(crh, mtmc, neci)		
rachcp	RachCntrlPara(maxtx, txint, re)		
cbchd	cbchd		
cbchma	–		
si4roct	SI4RestOctet_CBMS		PI=0, C2 parameters not present

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf5(bcchfl: NCD)

PDU Type : SYSINFO5_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : SYSTEM INFORMATION TYPE 5 containing default neighbour cells description for GSM900

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00011101'B		
bcchfl	bcchfl		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf5bis(bcchfl: NCD)
PDU Type : SYSINFO5bis_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : SYSTEM INFORMATION TYPE 5bis containing partial neighbour cells description with 1 frequency for DCS1800.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00000101'B		message type
xbcchfl	bcchfl		ARFCN 500

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf6(ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; nccp : NCCP)
PDU Type : SYSINFO6_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SYSTEM INFORMATION TYPE 6 message containing default parameters for RR testing.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00011110'B		message type
ci	ci		cell identity
lai	LocAreaId(mcc, mnc, lac)		
co	co		DTX not allowed, RadioTimeout = 8
nccp	nccp		ncc permitted

Detailed Comments :

PDU Constraint Declaration

Constraint Name : TmsiReallocCmd(par:MI; mcc, mnc, lac: OCTETSTRING)
PDU Type : TMSIRE_CMD_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : TMSI REALLOCATION COMMAND message.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'00011010'B		
lai	LocAreaId(mcc, mnc, lac)		
mi	par		new TMSI

Detailed Comments :

PDU Constraint Declaration

Constraint Name : TmsiReallocComp
PDU Type : TMSIRE_COM_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : TMSI REALLOCATION COMPLETE message.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'0?011011'B		

Detailed Comments :

IV

Dynamic Part

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_1

Group : EFR/

Purpose : To verify that the MS (in an RR connected state) with a TCH/F allocated acknowledging a CHANNEL MODE MODIFY message by sending a CHANNEL MODE MODIFY ACKNOWLEDGEMENT message specifying and switching to the correct mode.
- − the new mode if that mode is supported
- − the old mode if the new mode is not supported.

This shall be verified for the channel modes
- − speech full rate versions 1/2 and 3
- − data 9.6 Kb/s
- − data 4.8 Kb/s full rate
- − data 2.4 Kb/s full rate.

Configuration :

Default : OtherEventsFail

Comments : apply only to the MS supporting TCH/F (EFR).

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_02, CellChDes_03, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			1.
3		+gsmOrDcs			
4		+FullRateCh_A_1_nociph(C_Ass, TSPX_TmSltDef, TSPX_TscDef, ChMod_speech, FreqTCH(C_arfcn_tchA), FreqTCH(C_arfcn_tchAd), TimingAdv(30), C_BABR_0, C_cch_1Comb, C_BPM_3)			
5		L!DL_UdatRqPg1Rq	PgReq1(TCV_PgCh, TCV_Pgg, PgReqTp1_05)		
6		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_17)		
7		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
8		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_r13(TCV_Rr, TCV_Fn, TCV_chd1, TimingAdv(30)))		2.
9		L?DL_EstInPgRes	PagingRes(PagingRes_r02)		
10		ACTIVATE(OtherEventsFail)			Restore Normal default
11		+test_channel_modes			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
		test_channel_modes			
12		+set_FullSpeech_v2			
13		+test_24DataF			
14		+set_FullSpeech_v2			
15		+test_48DataF			
16		+set_FullSpeech_v2			
17		+test_96DataF			
18		+set_FullSpeech_v2			
19		+test_FullSpeech_v1			
20		+set_FullSpeech_v2			
21		+test_FullSpeech_v3			
22		+PostMainLinkRel(TCV_chTch)			
		set_FullSpeech_v2			
23		L!DL_DatRqChmmo	ChmmoReqSnd(TCV_chTch, ChmomoReq_01(ChMod_EFR2, TCV_chd1))		
24	E_0001	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_EFR2, TCV_chd1))	(P)	
		test_FullSpeech_v3			
25		L!DL_DatRqChmmo	ChmmoReqSnd(TCV_chTch, ChmomoReq_01(ChMod_EFR3, TCV_chd1))		
26		[TSPC_EFR_Speech_v3]			
27	E_0002	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_EFR3, TCV_chd1))	(P)	
28		[NOT TSPC_EFR_Speech_v3]			
29	E_0003	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_EFR2, TCV_chd1))	(P)	
		test_FullSpeech_v1			
30		L!DL_DatRqChmmo	ChmmoReqSnd(TCV_chTch, ChmomoReq_01(ChMod_speech, TCV_chd1))		
31	E_0004	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_speech, TCV_chd1))	(P)	
		test_24DataF			
32		L!DL_DatRqChmmo	ChmmoReqSnd(TCV_chTch, ChmomoReq_01(ChMod_3k, TCV_chd1))		
33		[TSPC_24DataF]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
34	E_0005	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_3k, TCV_chd1))	(P)	
35		[NOT TSPC_24DataF]			
36	E_0006	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_EFR2, TCV_chd1))	(P)	
		test_48DataF			
37		L!DL_DatRqChmmo	ChmmoReqSnd(TCV_chTch, ChmomoReq_01(ChMod_6k, TCV_chd1))		
38		[TSPC_48DataF]			
39	E_0007	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_6k, TCV_chd1))	(P)	
40		[NOT TSPC_48DataF]			
41	E_0008	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_EFR2, TCV_chd1))	(P)	
		test_96DataF			
42		L!DL_DatRqChmmo	ChmmoReqSnd(TCV_chTch, ChmomoReq_01(ChMod_12k, TCV_chd1))		
43		[TSPC_96Data]			
44	E_0009	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_12k, TCV_chd1))	(P)	
45		[NOT TSPC_96Data]			
46	E_0010	L?DL_DatInChmmoAck	ChmmoAckRcv(TCV_chTch, ChmomoAck_01(ChMod_EFR2, TCV_chd1))	(P)	
		gsmOrDcs			
47		[TSPC_PGSM OR TSPC_EGSM]			
48		(TCV_chd1 := ChDescrp_nfh('00001'B, TSPX_TmSltDef, TSPX_TscDef, C_arfcn_tchA))			
49		[TSPC_DCS]			
50		(TCV_chd1 := ChDescrp_nfh('00001'B, TSPX_TmSltDef, TSPX_TscDef, C_arfcn_tchAd))			

Detailed Comments : 1. Default system infirmations for RR testing.
2. TCH/F channel.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_1

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv2, without frequency hopping towards a TCH/F,sv2, without frequency hopping and,

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+FullRateCh_A_1(C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellA

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full_v2, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish a MT call with non hopping channel.
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 500, TCV_Cntref := TSPX_NoOfHoAccA, TCV_T := 20, TCV_Horf:= TSPX_HoRefA, TCV_ts := TSPX_TmSltB, TCV_tsc := TSPX_TscA, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, nonFH on cellB
11		+ltree_arfcnB			
12		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_ch := TCV_chTch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_chdescr_arfcn := C_BCCHcarrierB_ho, TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
21		[TSPC_DCS]			
22		(TCV_chdescr_arfcn := C_BCCHcarrierB_hod, TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments	**:** IUT enters state U10 with TCH/F_nonFH in cell A and then HandOver from TCH/F_nonFH sv2 of cell A to TCH/F_nonFH sv2 in cell B Notes: 1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_2

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F sv2, without frequency hopping on cell B towards a TCH/F, sv2, with frequency hopping on cell A

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellB
7		+ltree_arfcnB			
8		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full_v2, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish a MTcall with non hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
10		(TCV_n := 500, TCV_Cntref := TSPX_NoOfHoAccB, TCV_T := TSPX_TimadvA, TCV_ts := TSPX_TmSltNotZero, TCV_Horf:= TSPX_HoRefB, TCV_tsc := TSPX_TscB, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
11		+FullRateCh_A_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, FH on cellA
12		(TCV_ch := TCV_chTch)			
13		+ltree_BuildHoCMD			
14		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHF_B_1)			
15		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
16		(TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
17		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
18		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
19		+RR_hocomp1(TCV_n, TCV_chTch1)			
20		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
21		[TSPC_PGSM OR TSPC_EGSM]			
22		(TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
23		[TSPC_DCS]			
24		(TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			
		ltree_BuildHoCMD			
25		[TSPC_PGSM OR TSPC_EGSM]			
26		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnA, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_omit, Freqchseq_22, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
27		[TSPC_DCS]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
28		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnAd_4, 9, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frqls_20_Ad, Frql_omit, CellChDes_Omit, ChMod_omit, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HO from TCH/F_nonFH sv2 of cell A to TCH/F_nonFH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_3

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv2, with frequency hopping towards a TCH/F, sv2, without frequency hopping and,

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+FullRateCh_A_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, FH on cellA
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Full_v2, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 500, TCV_Cntref := TSPX_NoOfHoAccC, TCV_T := 20, TCV_ts := TSPX_TmSltC, TCV_Horf:= TSPX_HoRefC, TCV_tsc := TSPX_TscC, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellB
11		+ltree_arfcnB			
12		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_report_otd, C_non_synchronised), ChMod_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_ch := TCV_chTch)			
14		(TCV_Fn := OM_ComingFn(TCV_chTch1), TCV_Null:= OM_SendNextOn(TCV_chTch1, TCV_Fn), TCV_Strt := OC_StartTime(TCV_Fn, (C_StartingTimeHO + 100), 1), TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			starting time is set to 1.1 s plus 100 frames
15		L!DL_DatRqHoCmd(DL_DatRqHoCmd.msg.strt:= TCV_Strt)	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ltree_FirstBurstOnTimeCHK			
19		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
20		[TSPC_PGSM OR TSPC_EGSM]			
21		(TCV_chdescr_arfcn := C_BCCHcarrierB_ho, TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
22		[TSPC_DCS]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
23		(TCV_chdescr_arfcn := C_BCCHcarrierB_hod, TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			
		ltree_FirstBurstOnTimeCHK			
24		+FnArith(TCV_FnBurst, TCV_Strt.fn)			
25	E_01 26	[(TCV_Time > 5) OR (TCV_Time < 0)]		(F)	
26	E_01 27	[(TCV_Time >= 0) AND (TCV_Time < 5)]		(P)	

Detailed Comments : HO from TCH/F_FH sv2 of cell A to TCH/F_nonFH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_4

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv2, without frequency hopping towards a TCH/F, sv1, without frequency hopping and,

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellB
7		+ltree_arfcnB			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
8		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full_v2, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish a MTcall with non hopping channel.
9	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
10		(TCV_n := 500, TCV_Cntref := TSPX_NoOfHoAccA, TCV_T := 20, TCV_ts := TSPX_TmSltD, TCV_Horf:= TSPX_HoRefD, TCV_tsc := TSPX_TscD, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
11		+FullRateCh_A_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_speech, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv1, NonFH on cellA
12		+ltree_arfcnA			
13		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_speech_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
14		(TCV_ch := TCV_chTch)			
15		(TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
16		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
17		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
18		+RR_hocomp1(TCV_n, TCV_chTch1)			
19		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			
20		[TSPC_PGSM OR TSPC_EGSM]			
21		(TCV_chdescr_arfcn := C_arfcnA, TCV_tch_arfcn := TSPX_TCHcarrierA_ho)			
22		[TSPC_DCS]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
23		(TCV_chdescr_arfcn := C_arfcnAd_4, TCV_tch_arfcn := TSPX_TCHcarrierA_hod)			
		ltree_arfcnB			
24		[TSPC_PGSM OR TSPC_EGSM]			
25		(TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
26		[TSPC_DCS]			
27		(TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Detailed Comments : HO from TCH/F_nonFH sv2 of cell B to TCH/F_nonFH sv1 in cell A

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_5

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv1, without frequency hopping towards a TCH/F,sv2, without frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+FullRateCh_A_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv1, NonFH on cellA
7		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish a MT call with non hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 500, TCV_Cntref := TSPX_NoOfHoAccB, TCV_T := TSPX_TimadvB, TCV_ts := TSPX_TmSltE, TCV_Horf:= TSPX_HoRefE, TCV_tsc := TSPX_TscE, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod) , TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, nonFH on cellB
11		+ltree_arfcnB			
12		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_ch := TCV_chTch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_ ch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_chdescr_arfcn := C_BCCHcarrierB_ho, TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
21		[TSPC_DCS]			
22		(TCV_chdescr_arfcn := C_BCCHcarrierB_hod, TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Detailed Comments : HandOver from TCH/F_nonFH sv1 of cell A to TCH/F_nonFH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_6

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F sv2, without frequency hopping on cell B towards a TCH/F, sv1, with frequency hopping on cell A

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvB, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellB
7		+ltree_arfcnB			
8		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full_v2, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish MTcall with NonFH channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
10		(TCV_n := 500, TCV_Cntref := TSPX_NoOfHoAccC, TCV_T := TSPX_TimadvC, TCV_ts := TSPX_TmSltNotZero, TCV_Horf:= TSPX_HoRefF, TCV_tsc := TSPX_TscF, TCV_chtype := '00001'B)			
11		+FullRateCh_A_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			
12		(TCV_ch := TCV_chTch)			
13		+ltree_BuildHoCMD			
14		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHF_B_1)			
15		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
16		(TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
17		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
18		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
19		+RR_hocomp1(TCV_n, TCV_chTch1)			
20		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
21		[TSPC_PGSM OR TSPC_EGSM]			
22		(TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
23		[TSPC_DCS]			
24		(TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			
		ltree_BuildHoCMD			
25		[TSPC_PGSM OR TSPC_EGSM]			
26		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnA, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_22, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
27		[TSPC_DCS]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
28		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnAd_4, 9, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frqls_20_Ad, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HO from TCH/F_nonFH sv2 of cell B to TCH/F_FH sv1 in cell A

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_7

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F sv1, with frequency hopping on cell A towards a TCH/F, sv2, with frequency hopping on cell B

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+FullRateCh_A_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv1, FH on cellA
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Full, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccA, TCV_T := 20, TCV_ts := TSPX_TmSltNotZero1, TCV_Horf:= TSPX_HoRefG, TCV_tsc := TSPX_TscG, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof1, FreqTCHb_hof1d, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, FH on cellB
11		(TCV_ch := TCV_chTch)			
12		+ltree_BuildHoCMD			
13		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHF_A_1)			
14		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
15		(TCV_Fn := OM_ComingFn(TCV_chTch1), TCV_Null:= OM_SendNextOn(TCV_chTch1, TCV_Fn), TCV_Strt := OC_StartTime(TCV_Fn, (C_StartingTimeHO + 100), 1))			starting time is set to 1.1 s plus 100 frames
16		(TCV_Null := OM_SendNextOnHO(TC V_ch, TCV_Cntref))			
17		L!DL_DatRqHoCmd(DL_DatRqHoCmd.msg .strt:= TCV_Strt)	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
18		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
19		+RR_hocomp1(TCV_n, TCV_chTch1)			
20		+ltree_FirstBurst OnTimeCHK			
21		+ChanRel_end(TCV_ch)			
		ltree_BuildHoCMD			
22		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
23		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_BCCHcarrierB_ho, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_20_B3, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
24		[TSPC_DCS]			
25		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_BCCHcarrierB_hod, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_20_B3d, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
		ltree_FirstBurstOnTimeCHK			
26		+FnArith(TCV_FnBurst, TCV_Strt.fn)			
27	E_0136	[(TCV_Time > 5) OR (TCV_Time < 0)]		(F)	
28	E_0137	[(TCV_Time >= 0) AND (TCV_Time < 5)]		(P)	

Detailed Comments : HO from TCH/F_FH sv1 of cell A to TCH/F_FH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_8

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv2, without frequency hopping towards a TCH/F, sv1, with frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof1, FreqTCHb_hof1d, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, FH on cellB
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Full_v2, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
9		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccB, TCV_T := TSPX_TimadvA, TCV_ts := TSPX_TmSltNotZero, TCV_Horf:= TSPX_HoRefH, TCV_tsc := TSPX_TscA, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_A_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv1, NonFH on cellA
11		(TCV_ch := TCV_chTch)			
12		+ltree_BuildHoCMD			
13		(TCV_Null := OM_SendNextOnHO(TCV_ch , TCV_Cntref))			
14		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
15		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
16		+RR_hocomp1(TCV_n, TCV_chTch1)			
17		+ChanRel_end(TCV_ch)			
		ltree_BuildHoCMD			
18		[TSPC_PGSM OR TSPC_EGSM]			
19		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnA, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_22, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
20		[TSPC_DCS]			
21		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnAd_4, 9, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frqls_20_Ad, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/F_FH sv2 of cell B to TCH/F_FH sv1 in cell A

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_9

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv1, with frequency hopping towards a TCH/F, sv2, without frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+FullRateCh_A_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv1, FH on cellA
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Full, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccC, TCV_T := TSPX_TimadvB, TCV_ts := TSPX_TmSltF, TCV_Horf:= TSPX_HoRefI, TCV_tsc := TSPX_TscB, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod) , TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, nonFH on cellB
11		+ltree_arfcnB			
12		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_ch := TCV_chTch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_ ch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_chdescr_arfcn := C_BCCHcarrierB_ho, TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
21		[TSPC_DCS]			
22		(TCV_chdescr_arfcn := C_BCCHcarrierB_hod, TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Detailed Comments : HandOver from TCH/F_nonFH sv1 of cell A to TCH/F_nonFH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_10

Group : EFR/

Purpose : o test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv2, without frequency hopping towards a TCH/H, sv1, with frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvB, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellB
7		+ltree_arfcnB			
8		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full_v2, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish MTcall with NonFH channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
10		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccD, TCV_T := TSPX_TimadvC, TCV_ts := TSPX_TmSltNotZero1, TCV_Horf:= TSPX_HoRefJ, TCV_tsc := TSPX_TscC, TCV_chtype := INT_TO_BIT((2+BIT_TO_INT(TSPX_TCHHSubA)),5))			
11		+HalfRateCh_A_1(TSPX_TCHHSubA, C_Asynho, TCV_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			
12		(TCV_ch := TCV_chTch)			
13		+ltree_BuildHoCMD			
14		(TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
21		[TSPC_DCS]			
22		(TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			
		ltree_BuildHoCMD			
23		[TSPC_PGSM OR TSPC_EGSM]			
24		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnA, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_22, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
25		[TSPC_DCS]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
26		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnAd_4, 9, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frqls_20_Ad, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/F_non sv2 of cell B to TCH/H_FH sv1 in cell A

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_11

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/H, sv1, with frequency hopping towards a TCH/F, sv2, with frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+HalfRateCh_A_1(TSPX_TCHHSubA, C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/H, sv1, FH on cellA
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Half, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccA, TCV_T := 20, TCV_ts := TSPX_TmSltNotZero, TCV_Horf:= TSPX_HoRefA, TCV_tsc := TSPX_TscD, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof1, FreqTCHb_hof1d, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, FH on cellB
11		(TCV_ch := TCV_chTch)			
12		+ltree_BuildHoCMD			
13		(TCV_Fn := OM_ComingFn(TCV_chTch1), TCV_Null:= OM_SendNextOn(TCV_chTch1, TCV_Fn), TCV_Strt := OC_StartTime(TCV_Fn, (C_StartingTimeHO + 100), 1))			starting time is set to 1.1 s plus 100 frames
14		(TCV_Null := OM_SendNextOnHO(TCV_ ch, TCV_Cntref))			
15		L!DL_DatRqHoCmd(DL_DatRqHoCmd.msg.str t:= TCV_Strt)	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ChanRel_end(TCV_ch)			
		ltree_BuildHoCMD			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_BCCHcarrierB_ho, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_20_B3, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
21		[TSPC_DCS]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
22		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_BCCHcarrierB_hod, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_20_B3d, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit)) ltree_FirstBurstOnTimeCHK			
23		+FnArith(TCV_FnBurst, TCV_Strt.fn)			
24	E_0146	[(TCV_Time > 5) OR (TCV_Time < 0)]		(F)	
25	E_0147	[(TCV_Time >= 0) AND (TCV_Time < 5)]		(P)	

Detailed Comments : HandOver from TCH/H_FH sv1 of cell A to TCH/F_FH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_12

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F, sv2, with frequency hopping towards a TCH/H, sv1, with frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof1, FreqTCHb_hof1d, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, FH on cellB
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Full_v2, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
9		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccE, TCV_T := TSPX_TimadvA, TCV_ts := TSPX_TmSltNotZero1, TCV_Horf:= TSPX_HoRefB, TCV_tsc := TSPX_TscD, TCV_chtype := INT_TO_BIT((2+BIT_TO_INT(TSPX_TCHHSubDef)),5))			Parameters initialization for Handover procedure.
10		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Asynho, TCV_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/H, sv1, FH on cellA
11		(TCV_ch := TCV_chTch)			
12		+ltree_BuildHoCMD			
13		(TCV_Null := OM_SendNextOnHO(TCV_ch , TCV_Cntref))			
14		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
15		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
16		+RR_hocomp1(TCV_n, TCV_chTch1)			
17		+ChanRel_end(TCV_ch)			
		ltree_BuildHoCMD			
18		[TSPC_PGSM OR TSPC_EGSM]			
19		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnA, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_22, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
20		[TSPC_DCS]			
21		(TCV_HoCMD := HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, C_arfcnAd_4, 9, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frqls_20_Ad, Frql_omit, CellChDes_Omit, ChMod_speech_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/F_FH sv2 of cell B to TCH/H_FH sv1 in cell A

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_13

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/H, sv1, with frequency hopping towards a TCH/F,sv2, without frequency hopping

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := TSPX_TimadvA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCHa_hof1, FreqTCHa_hof1d, C_Noarfcn, C_Noarfcn, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/H, sv1, FH on cellA
7		+Est_MT_Call_FH(TimingAdv(TCV_T), C_Half, INT_TO_BIT(TSPX_MAIO, 6), TSPX_HSN, TSPX_RANDDef, TCV_asscmd_ts, TSPX_TscDef, TSPX_TCHHSubDef)			To establish a MTcall with hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccB, TCV_T := 20, TCV_ts := TSPX_TmSltG, TCV_Horf:= TSPX_HoRefC, TCV_tsc := TSPX_TscD, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod) , TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, nonFH on cellB
11		+ltree_arfcnB			
12		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_ch := TCV_chTch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_ ch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(TCV_n, TCV_chTch1)			
18		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_chdescr_arfcn := C_BCCHcarrierB_ho, TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
21		[TSPC_DCS]			
22		(TCV_chdescr_arfcn := C_BCCHcarrierB_hod, TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Detailed Comments : HandOver from TCH/H_FH sv1 of cell A to TCH/F_nonFH sv2 in cell B

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_14

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/F sv2, without frequency hopping on cell B towards a TCH/H, sv1, with no frequency hopping on cell A

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellB, C_SCH_B, C_BCCH_B_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellB, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Bman, CellChDes_201_Bd, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnB, C_arfcnBd, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellA_ho(C_E_neighbourdefault, C_BCCHcarrierA_ho, C_BCCHcarrierA_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Aman, CellChDes_201_Ad, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell A is 100 in advance of frame number in cell B.
6		+FullRateCh_B_1(C_Asynho, TCV_asscmd_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, NonFH on cellB
7		+ltree_arfcnB			
8		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Full_v2, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish a MTcall with non hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
10		(TCV_n := 750, TCV_Cntref := TSPX_NoOfHoAccF, TCV_T := 20, TCV_ts := TSPX_TmSltA, TCV_Horf:= TSPX_HoRefD, TCV_tsc := TSPX_TscE, TCV_chtype := INT_TO_BIT((2+BIT_TO_INT(TSPX_TCHHSubA)),5))			Parameters initialization for Handover procedure.
11		+HalfRateCh_A_1(TSPX_TCHHSubA, C_Asynho, TCV_ts, TCV_tsc, ChMod_speech, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/H, sv1,nonFH on cellA
12		+ltree_arfcnA			
13		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_speech_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
14		(TCV_ch := TCV_chTch)			
15		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHF_B_1)			
16		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
17		(TCV_Null := OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
18		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
19		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
20		+RR_hocomp1(TCV_n, TCV_chTch1)			
21		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			
22		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
23		(TCV_chdescr_arfcn := C_arfcnA, TCV_tch_arfcn := TSPX_TCHcarrierA_ho)			
24		[TSPC_DCS]			
25		(TCV_chdescr_arfcn := C_arfcnAd_4, TCV_tch_arfcn := TSPX_TCHcarrierA_hod)			
		ltree_arfcnB			
26		[TSPC_PGSM OR TSPC_EGSM]			
27		(TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
28		[TSPC_DCS]			
29		(TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Detailed Comments : HO from TCH/H_nonFH sv2 of cell B to TCH/F_nonFH sv1 in cell A

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_1_15

Group : EFR/

Purpose : To test that the MS shall correctly apply the handover procedure in the non−synchronized case when:
− a call is in progress and,
− handover is performed from a TCH/H sv1, with no frequency hopping on cell A towards a TCH/F, sv2, with no frequency hopping on cell B

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts := TSPX_TmSltA, TCV_T := 20, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5))			
3		+EFR_BasicServiceMT(TSPX_MTBscSvcI, TSPX_MTChRateI)			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_201_Ad, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_201_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC, TSPX_k, 100)			frame number in cell B is 100 in advance of frame number in cell A.
6		+HalfRateCh_A_1(TSPX_TCHHSubA, C_Asynho, TSPX_TmSltA, TCV_tsc, ChMod_speech, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/H, sv1, NonFH on cellA
7		+Est_MT_CallNonFH(TimingAdv(TCV_T), C_Half, TSPX_TCHHSubDef, TSPX_RANDDef)			To establish a MT call with non hopping channel.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
8	body	(TCV_ch := TCV_chTch, TCV_chTch1:=TCV_ch)			
9		(TCV_n :=750, TCV_Cntref := TSPX_NoOfHoAccC, TCV_T := 20, TCV_ts := TSPX_TmSltB, TCV_Horf:= TSPX_HoRefE, TCV_tsc := TSPX_TscF, TCV_chtype := '00001'B)			Parameters initialization for Handover procedure.
10		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod) , TimingAdv(TCV_T), C_BABR_0, C_cch_1Comb, C_BPM_3)			Set TCH/F, sv2, nonFH on cellB
11		+ltree_arfcnB			
12		(TCV_HoCMD := HandOverCmd_nfh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC, C_BCC, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_ch := TCV_chTch)			
14		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHH_A_1)			
15		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
16		(TCV_Null := OM_SendNextOnHO(TC V_ch, TCV_Cntref))			
17		L!DL_DatRqHoCmd	HndOvSnd(TCV_chTch1, TCV_HoCMD)		HO command
18		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
19		+RR_hocomp1(TCV_n, TCV_chTch1)			
20		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
21		[TSPC_PGSM OR TSPC_EGSM]			
22		(TCV_chdescr_arfcn := C_BCCHcarrierB_ho, TCV_tch_arfcn := TSPX_TCHcarrierB_ho)			
23		[TSPC_DCS]			
24		(TCV_chdescr_arfcn := C_BCCHcarrierB_hod, TCV_tch_arfcn := TSPX_TCHcarrierB_hod)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments :	HO from TCH/H_nonFH sv1 of cell A to TCH/F_nonFH sv2 in cell B Notes: 1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_1

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from SDCCH/4 with no frequency hopping to TCH/F using full rate speech version 2 with frequency hopping.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ia_ts:=TSPX_TmSltNotZero, TCV_ts:='000'B, TCV_tsc:=TSPX_TscA, TCV_Cntref:=TSPX_NoOfHoAccA, TCV_Horf:=TSPX_HoRefA, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlA,5), TCV_cchannel:=C_cch_1Comb, TCV_T := 20)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
6		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof3, FreqTCHb_hof3d, TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F FH on cellB
7		+ltree_arfcnA			Set the ARFCN for NonFH channel on cellA
8		+Est_MO_Call_init(C_CHSDCCH4_NonFH, MoblAllc_omit, MoblAllc_omit, TimingAdv(30), C_BCC, C_one, C_one)			Initiate MO call on cellA
9	body	+ltree_BuildHoCMD			
10		(TCV_chTch1 := TCV_ch)			
11		(TCV_nc :='010'B)			
12		+Check_Cells_Present (TCV_nc, TCV_sacch)			
13		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
14		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
15		(TCV_ch:=TCV_chTch)			
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
18		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
19		+SendSeqNo_chk			Is N(SD) correct?
20		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			
21		[TSPC_PGSM OR TSPC_EGSM]			
22		(TCV_tch_arfcn:=C_arfcnA)			
23		[TSPC_DCS]			
24		(TCV_tch_arfcn:=C_arfcnAd_4)			
		ltree_BuildHoCMD			
25		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
26		(TCV_HoCMD:= HandOverCmd_fh(’00001’B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, C_BCCHcarrierB_ho, 15, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_06, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
27		[TSPC_DCS]			
28		(TCV_HoCMD:= HandOverCmd_fh(’00001’B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, C_BCCHcarrierB_hod, 11, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_20_B5d, Frql_omit, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HO from SDCCH/4, nonFH to TCH/F, sv2, FH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_2

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from SDCCH/8 with frequency hopping to TCH/F using full rate speech version 2 without frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:='000'B, TCV_tsc:=TSPX_TscB, TCV_Cntref:=TSPX_NoOfHoAccB, TCV_Horf:=TSPX_HoRefB, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlB,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero1)			
6		+SDCCH8_A_1_nociph(TSPX_SDCCH8SubDef, C_Immass, TCV_ia_ts, TCV_tsc, ChMod_sign, FreqSDCCHa_hof1, FreqSDCCHa_hof1d, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F nonFH on cellB
10		+Est_MO_Call_init(C_CHSDCCH8_FH, MoblAllc_281, MoblAllc_281d, TimingAdv(30), TCV_tsc, 3, 3)			Initiate MO call on cellA
11		+ltree_arfcnB			
12	body	(TCV_HoCMD:= HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
13		(TCV_chTch1 := TCV_ch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
16		(TCV_ch:=TCV_chTch)			
17		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
18		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
19		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
20		+SendSeqNo_chk			Is N(SD) correct?

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
21		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
22		[TSPC_PGSM OR TSPC_EGSM]			
23		(TCV_tch_arfcn:=TSPX_TCHcarrierB_ho, TCV_chdescr_arfcn:=C_BCCHcarrierB_ho)			
24		[TSPC_DCS]			
25		(TCV_tch_arfcn:=TSPX_TCHcarrierB_hod, TCV_chdescr_arfcn:=C_BCCHcarrierB_hod)			

Detailed Comments : HandOver from SDCCH/8, FH to TCH/F, sv2, nonFH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_3

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from SDCCH/8 without frequency hopping to TCH/F using full rate speech version 2 with frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		+ltree_data_init			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+SDCCH8_A_1_nociph(TSPX_SDCCH8SubDef, C_Immass, TCV_ia_ts, TCV_tsc, ChMod_sign, FreqTCH(TSPX_TCHcarrierA_ho), FreqTCH(TSPX_TCHcarrierA_hod), TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set SDCCH /8 nonFH on cell A
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof6, FreqTCHb_hof6d, TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FFH on cellB
10		+ltree_arfcnA			Set the ARFCN for NonFH channel on cellA
11		+Est_MO_Call_init(C_CHSDCCH8_NonFH, MoblAllc_omit, MoblAllc_omit, TimingAdv(30), TCV_tsc, C_one, C_one)			Initiate MO call on cellA
12	body	+ltree_BuildHoCMD			
13		(TCV_chTch1 := TCV_ch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
16		(TCV_ch:=TCV_chTch)			
17		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
18		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
19		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
20		+SendSeqNo_chk			Is N(SD) correct?
21		+ChanRel_end(TCV_ch)			
		ltree_data_init			
22		[TSPC_PGSM OR TSPC_EGSM]			
23		(TCV_ts:='000'B, TCV_tsc:=TSPX_TscC, TCV_Cntref:=TSPX_NoOfHoAccC, TCV_Horf:=TSPX_HoRefC, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlC,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			
24		[TSPC_DCS]			
25		(TCV_ts:=TSPX_TmSltNotZero, TCV_tsc:=TSPX_TscC, TCV_Cntref:=TSPX_NoOfHoAccC, TCV_Horf:=TSPX_HoRefC, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlC,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
		ltree_arfcnA			
26		[TSPC_PGSM OR TSPC_EGSM]			
27		(TCV_tch_arfcn:=C_arfcnA)			
28		[TSPC_DCS]			
29		(TCV_tch_arfcn:=C_arfcnAd_4)			
		ltree_BuildHoCMD			
30		[TSPC_PGSM OR TSPC_EGSM]			
31		(TCV_HoCMD:= HandOverCmd_fh('00001'B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, C_BCCHcarrierB_ho, 2, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_03, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
32		[TSPC_DCS]			
33		(TCV_HoCMD:= HandOverCmd_fh('00001'B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, C_BCCHcarrierB_hod, 3, C_HSN_0, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_20_B10d, Frql_omit, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from SDCCH/8, nonFH to TCH/F, sv2, FH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_4

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/F without frequency hopping to TCH/F using full rate speech version 2 with frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:=TSPX_TmSltNotZero1, TCV_tsc:=TSPX_TscA, TCV_Cntref:=TSPX_NoOfHoAccA, TCV_Horf:=TSPX_HoRefA, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlA,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20, TCV_ChMod.mode := C_ChMod_r)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+FullRateCh_A_1_nociph(C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCH(TSPX_TCHcarrierA_ho), FreqTCH(TSPX_TCHcarrierA_hod), TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F nonFH on cell A

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof1, FreqTCHb_hof1d, TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FH on cellB
10		+ltree_arfcnA			Set the ARFCN for NonFH channel on cellA
11		+Est_MO_Call_init(C_CHTCHF_NonFH, MoblAllc_omit, MoblAllc_omit, TimingAdv(30), TCV_tsc, C_one, C_one)			Initiate MO call on cellA
12	body	+ltree_BuildHoCMD			
13		(TCV_chTch1 := TCV_ch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
16		(TCV_ch:=TCV_chTch)			
17		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
18		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
19		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
20		+SendSeqNo_chk			Is N(SD) correct?
21		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			
22		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
23		(TCV_tch_arfcn:=66)			
24		[TSPC_DCS]			
25		(TCV_tch_arfcn:=754)			
		ltree_BuildHoCMD			
26		[TSPC_PGSM OR TSPC_EGSM]			
27		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, C_BCCHcarrierB_ho, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_20_B3, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
28		[TSPC_DCS]			
29		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, C_BCCHcarrierB_hod, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_20_B3d, Frql_omit, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/F, nonFH to TCH/F, sv2, FH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_5

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/F without frequency hopping to TCH/F using full rate speech version 2 without frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:=TSPX_TmSltNotZero1, TCV_tsc:=TSPX_TscB, TCV_Cntref:=TSPX_NoOfHoAccB, TCV_Horf:=TSPX_HoRefB, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlB,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20, TCV_ChMod.mode := C_ChMod_r)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+FullRateCh_A_1_nociph(C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCH(TSPX_TCHcarrierA_ho), FreqTCH(TSPX_TCHcarrierA_hod), TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F nonFH on cell A

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F nonFH on cellB
10		+ltree_arfcnA			Set the ARFCN for NonFH channel on cellA
11		+Est_MO_Call_init(C_CHTCHF_NonFH, MoblAllc_omit, MoblAllc_omit, TimingAdv(30), TCV_tsc, C_one, C_one)			Initiate MO call on cellA
12		+ltree_arfcnB			
13	body	+ltree_BuildHoCMD			
14		(TCV_chTch1 := TCV_ch)			
15		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
16		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
17		(TCV_ch:=TCV_chTch)			
18		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
19		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
20		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
21		+SendSeqNo_chk			Is N(SD) correct?
22		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
23		[TSPC_PGSM OR TSPC_EGSM]			
24		(TCV_tch_arfcn:=66)			
25		[TSPC_DCS]			
26		(TCV_tch_arfcn:=791)			
		ltree_arfcnB			
27		[TSPC_PGSM OR TSPC_EGSM]			
28		(TCV_tch_arfcn:=31, TCV_chdescr_arfcn:=C_BCCHcarrierB_ho)			
29		[TSPC_DCS]			
30		(TCV_tch_arfcn:=749, TCV_chdescr_arfcn:=C_BCCHcarrierB_hod)			
		ltree_BuildHoCMD			
31		[TSPC_PGSM OR TSPC_EGSM]			
32		(TCV_HoCMD:=HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, TCV_chdescr_arfcn, 31, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
33		[TSPC_DCS]			
34		(TCV_HoCMD:=HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, TCV_chdescr_arfcn, 749, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/F, nonFH to TCH/F, sv2, nonFH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_6

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/F with frequency hopping to TCH/F using full rate speech version 2 with frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		+ltree_data_init			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+FullRateCh_A_1_nociph(C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCHa_hof_ind5, FreqTCHa_hof_ind5d, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F FH on cell A
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch:=TCV_chTch)			Store TCH on cell A for call setup

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof_ind8, FreqTCHb_hof_ind8d, TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FH on cellB
10		+Est_MO_Call_init(C_CHTCHF_FH, MoblAllc_ind5A, MoblAllc_ind5Ad, TimingAdv(30), TCV_tsc, 5, 5)			Initiate MO call on cellA
11	body	+ltree_BuildHoCMD			
12		(TCV_chTch1 := TCV_ch)			
13		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHF_A_1)			
14		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
15		(TCV_Fn := OM_ComingFn(TCV_ch), TCV_n := 0−442, TCV_Fn := OC_FnInc(TCV_Fn, TCV_n), TCV_Null:= OM_SendNextOn(TCV_ch, TCV_Fn), TCV_Strt := OC_StartTime(TCV_Fn, (C_StartingTimeHO + 100), 1))			Starting time is set to 1.1 s pluse 100 frames
16		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
17		L!DL_DatRqHoCmd(DL_DatRqHoCmd.msg .strt:= TCV_Strt)	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
18		(TCV_ch:=TCV_chTch)			
19		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
20		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
21		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
22		+SendSeqNo_chk			Is N(SD) correct?
23		+ChanRel_end(TCV_ch)			
		ltree_data_init			
24		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
25		(TCV_ts:='000'B, TCV_tsc:=TSPX_TscC, TCV_Cntref:=TSPX_NoOfHoAccC, TCV_Horf:=TSPX_HoRefC, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlC,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20, TCV_ChMod.mode := C_ChMod_r)			Parameters initialization for Immediate Assignment and Handover commands.
26		[TSPC_DCS]			
27		(TCV_ts:=TSPX_TmSltNotZero1, TCV_tsc:=TSPX_TscC, TCV_Cntref:=TSPX_NoOfHoAccC, TCV_Horf:=TSPX_HoRefC, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlC,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20, TCV_ChMod.mode := C_ChMod_r) ltree_BuildHoCMD			
28		[TSPC_PGSM OR TSPC_EGSM]			
29		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TSPX_TscB, C_NCC_3, C_BCC_0, C_BCCHcarrierB_ho, 8, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_ind8B, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
30		[TSPC_DCS]			
31		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TSPX_TscB, C_NCC_3, C_BCC_0, C_BCCHcarrierB_hod, 8, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_ind8Bd, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HO from TCH/F, FH to TCH/F, sv2, FH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_7

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/F with frequency hopping to TCH/F using full rate speech version 2 without frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:='000'B, TCV_tsc:=TSPX_TscA, TCV_Cntref:=TSPX_NoOfHoAccA, TCV_Horf:=TSPX_HoRefA, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlA,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20, TCV_ChMod.mode := C_ChMod_r)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+FullRateCh_A_1_nociph(C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCHa_hof_ind3, FreqTCHa_hof_ind3d, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FFH on cell A

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F nonFH on cellB
10		+Est_MO_Call_init(C_CHTCHF_FH, MoblAllc_281, MoblAllc_ad281d, TimingAdv(30), TCV_tsc, 3, 3)			Initiate MO call on cellA
11		+ltree_arfcnB			
12		(TCV_chTch1 := TCV_ch)			
13		(TCV_HoCMD:= HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
14		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
16		(TCV_ch:=TCV_chTch)			
17		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
18		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
19		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
20		+SendSeqNo_chk			Is N(SD) correct?

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
22		[TSPC_PGSM OR TSPC_EGSM]			
23		(TCV_tch_arfcn:=TSPX_TCHcarrierB_ho, TCV_chdescr_arfcn:=C_BCCHcarrierB_ho)			
24		[TSPC_DCS]			
25		(TCV_tch_arfcn:=TSPX_TCHcarrierB_hod, TCV_chdescr_arfcn:=C_BCCHcarrierB_hod)			

Detailed Comments : HandOver from TCH/F, FH to TCH/F, sv2, nonFH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_8

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/H without frequency hopping to TCH/F using full rate speech version 2 with frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:=TSPX_TmSltNotZero1, TCV_tsc:=TSPX_TscA, TCV_Cntref:=TSPX_NoOfHoAccA, TCV_Horf:=TSPX_HoRefA, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlA,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/H nonFH on cell A

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof1, FreqTCHb_hof1d, TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FH on cellB
10		+ltree_arfcnA			Set the ARFCN for NonFH channel on cellA
11		+Est_MO_Call_init(C_CHTCHH_NonFH, MoblAllc_omit, MoblAllc_omit, TimingAdv(30), TCV_tsc, C_one, C_one)			Initiate MO call on cellA
12	body	+ltree_BuildHoCMD			
13		(TCV_chTch1 := TCV_ch)			
14		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
15		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
16		(TCV_ch:=TCV_chTch)			
17		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
18		+RR_hocomp1(500, TCV_chTch1)			Complete the HO procedure
19		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
20		+SendSeqNo_chk			Is N(SD) correct?
21		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			
22		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
23		(TCV_tch_arfcn:=66)			
24		[TSPC_DCS]			
25		(TCV_tch_arfcn:=754)			
		ltree_BuildHoCMD			
26		[TSPC_PGSM OR TSPC_EGSM]			
27		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, C_BCCHcarrierB_ho, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_20_B3, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
28		[TSPC_DCS]			
29		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, C_BCCHcarrierB_hod, 12, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_20_B3d, Frql_omit, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/H, nonFH to TCH/F, sv2, FH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_9

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/H without frequency hopping to TCH/F using full rate speech version 2 without frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:=TSPX_TmSltNotZero1, TCV_tsc:=TSPX_TscB, TCV_Cntref:=TSPX_NoOfHoAccB, TCV_Horf:=TSPX_HoRefB, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlB,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, TSPX_TCHcarrierA_ho, TSPX_TCHcarrierA_hod, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/H nonFH on cell A

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/F nonFH on cellB
10		+ltree_arfcnA			Set the ARFCN for NonFH channel on cellA
11		+Est_MO_Call_init(C_CHTCHH_NonFH, MoblAllc_omit, MoblAllc_omit, TimingAdv(30), TCV_tsc, C_one, C_one)			
12		+ltree_arfcnB			
13	body	+ltree_BuildHoCMD			
14		(TCV_chTch1 := TCV_ch)			
15		(TCV_Fn := OM_ComingFn(TCV_ch), TCV_n := 0−442, TCV_Fn := OC_FnInc(TCV_Fn, TCV_n), TCV_Null:= OM_SendNextOn(TCV_ch, TCV_Fn), TCV_Strt := OC_StartTime(TCV_Fn, (C_StartingTimeHO + 100), 1))			starting time is set to 1.1 s pluse 100 frames
16		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
17		L!DL_DatRqHoCmd(DL_DatRqHoCmd.msg .strt:= TCV_Strt)	HndOvSnd(TCV_ch, TCV_HoCMD)		
18		(TCV_ch:=TCV_chTch)			
19		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
20		+RR_hocomp1(500, TCV_chTch1)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
21		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
22		+SendSeqNo_chk			
23		+ChanRel_end(TCV_ch)			
		ltree_arfcnA			
24		[TSPC_PGSM OR TSPC_EGSM]			
25		(TCV_tch_arfcn:=66)			
26		[TSPC_DCS]			
27		(TCV_tch_arfcn:=791)			
		ltree_arfcnB			
28		[TSPC_PGSM OR TSPC_EGSM]			
29		(TCV_tch_arfcn:=31, TCV_chdescr_arfcn:=C_BCCHcarrierB_ho)			
30		[TSPC_DCS]			
31		(TCV_tch_arfcn:=749, TCV_chdescr_arfcn:=C_BCCHcarrierB_hod)			
		ltree_BuildHoCMD			
32		[TSPC_PGSM OR TSPC_EGSM]			
33		(TCV_HoCMD:=HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, TCV_chdescr_arfcn, 31, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
34		[TSPC_DCS]			
35		(TCV_HoCMD:=HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_5, TCV_chdescr_arfcn, 749, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/H, nonFH to TCH/F, sv2, nonFH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_10

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/H with frequency hopping to TCH/F using full rate speech version 2 with frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		+ltree_data_init			
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+HalfRateCh_A_1_nociph(TSPX_TCHHSubDef, C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCHa_hof_ind5, FreqTCHa_hof_ind5d, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/H FH on cell A
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCHb_hof_ind8, FreqTCHb_hof_ind8d, TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FH on cellB
10		+Est_MO_Call_init(C_CHTCHH_FH, MoblAllc_ind5A, MoblAllc_ind5Ad, TimingAdv(30), TCV_tsc, 5, 5)			
11	body	+ltree_BuildHoCMD			
12		(TCV_chTch1 := TCV_ch)			
13		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
14		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
15		(TCV_ch:=TCV_chTch)			
16		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.
17		+RR_hocomp1(500, TCV_chTch1)			
18		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
19		+SendSeqNo_chk			
20		+ChanRel_end(TCV_ch)			
		ltree_data_init			
21		[TSPC_PGSM OR TSPC_EGSM]			
22		(TCV_ts:='000'B, TCV_tsc:=TSPX_TscC, TCV_Cntref:=TSPX_NoOfHoAccC, TCV_Horf:=TSPX_HoRefC, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlC,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			Parameters initialization for Immediate Assignment and Handover commands.
23		[TSPC_DCS]			
24		(TCV_ts:=TSPX_TmSltNotZero1, TCV_tsc:=TSPX_TscC, TCV_Cntref:=TSPX_NoOfHoAccC, TCV_Horf:=TSPX_HoRefC, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlC,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			
		ltree_BuildHoCMD			
25		[TSPC_PGSM OR TSPC_EGSM]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
26		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TSPX_TscB, C_NCC_3, C_BCC_0, C_BCCHcarrierB_ho, 8, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_ind8B, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
27		[TSPC_DCS]			
28		(TCV_HoCMD:=HandOverCmd_fh('00001'B, TCV_ts, TSPX_TscB, C_NCC_3, C_BCC_0, C_BCCHcarrierB_hod, 8, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi(C_not_report_otd, C_non_synchronised), Frql_omit, Frql_ind8Bd, CellChDes_Omit, ChMod_EFR2_iei, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			

Detailed Comments : HandOver from TCH/H, FH to TCH/F, sv2, FH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_2_2_11

Group : EFR/

Purpose : To verify that the MS correctly applies the handover procedure from TCH/H with frequency hopping to TCH/F using full rate speech version 2 with no frequency hopping in the nonsynchronized case during call establishment.
To verify that if during call establishment a layer 3 MM or CC message just sent by the MS is not Layer 2 acknowledge before the channel change caused by the HANDOVER COMMAND message, the MS shall send the Layer 3 message to the new cell, using the same value in the N(SD) field, after the handover procedure.

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+EFR_BasicServiceMO(TSPX_MOBscSvcA, TSPX_MOChRateA)			
3		(TCV_ts:='000'B, TCV_tsc:=TSPX_TscA, TCV_Cntref:=TSPX_NoOfHoAccA, TCV_Horf:=TSPX_HoRefA, TCV_Pwrlvl_ho:=INT_TO_BIT(TSPX_PwrlvlA,5), TCV_cchannel:=C_cch_1nonComb, TCV_T := 20)			Parameters initialization for Immediate Assignment and Handover commands.
4		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, CellChDes_202_Adman, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_48, BcchFreqLst_Omit, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, C_arfcnAd_4, TSPX_IMSI, C_Test_fh, C_NCCP_10)			
5		(TCV_ia_ts:=TSPX_TmSltNotZero)			
6		+HalfRateCh_A_1_nociph(TSPX_TCHHSubDef, C_Immass, TCV_ia_ts, TCV_tsc, TCV_ChMod, FreqTCHa_hof_ind3, FreqTCHa_hof_ind3d, TimingAdv(30), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/HFH on cell A

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
7		+StartCellB_ho(C_E_neighbourdefault, C_BCCHcarrierB_ho, C_BCCHcarrierB_hod, C_Immass, TCV_slot, TCV_tsc, TimingAdv(TCV_T), C_TxInt_5, C_Max_7, C_NECI_0, C_ATT_0, C_BABR_0, TCV_cchannel, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2, CellChDes_20_Bman, CellChDes_202_Bd, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC_0, C_NCC_3, TSPX_k, 0)			The timeslots of Cells A and B are not coincident at the antenna connector.
8		(TCV_ch :=TCV_chTch)			
9		+FullRateCh_B_1(C_Asynho, TCV_ts, TCV_tsc, ChMod_EFR2, FreqTCH(TSPX_TCHcarrierB_ho), FreqTCH(TSPX_TCHcarrierB_hod), TimingAdv(TCV_T), C_BABR_0, TCV_cchannel, C_BPM_3)			Set TCH/FnonFH on cellB
10		+Est_MO_Call_init(C_CHTCHH_FH, MoblAllc_281, MoblAllc_ad281d, TimingAdv(30), TCV_tsc, 3, 3)			Initiate MO call on cellA
11	body	+ltree_arfcnB			
12		(TCV_chTch1 := TCV_ch)			
13		(TCV_nc :='010'B, TCV_sacchTch1 := C_SACCHH_A_1)			
14		+Check_Cells_Present (TCV_nc, TCV_sacchTch1)			
15		(TCV_HoCMD:= HandOverCmd_nfh('00001'B, TCV_ts, TCV_tsc, C_NCC_3, C_BCC_0, TCV_chdescr_arfcn, TCV_tch_arfcn, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, ChMod_EFR2_iei, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))			
16		(TCV_Null := OM_SendNextOnHO(TCV_chTch, TCV_Cntref))			
17		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, TCV_HoCMD)		HO command
18		(TCV_ch:=TCV_chTch)			
19		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, TimingAdv(TCV_T))		See Note 1.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
20		+RR_hocomp1(500,TCV_chTch1)			Complete the HO procedure
21		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
22		+SendSeqNo_chk			Is N(SD) correct?
23		+ChanRel_end(TCV_ch)			
		ltree_arfcnB			
24		[TSPC_PGSM OR TSPC_EGSM]			
25		(TCV_tch_arfcn:=TSPX_TCHcarrierB_ho, TCV_chdescr_arfcn:=C_BCCHcarrierB_ho)			
26		[TSPC_DCS]			
27		(TCV_tch_arfcn:=TSPX_TCHcarrierB_hod, TCV_chdescr_arfcn:=C_BCCHcarrierB_hod)			

Detailed Comments : HO from TCH/H, FH to TCH/F, sv2, nonFH

Notes:
1. PhyInfo is dequeued from the sending buffer, but held through OM_SendNextOnHO at the lower layers of the tester for sending till having received TCV_Cntref access bursts.

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_3

Group : EFR/

Purpose : 1) To verify that the MS indicates and includes in the mobile originating SETUP for speech call all the speech versions that it supports.

2) To verify that subsequently after receipt of an IMMEDIATE ASSIGNMENT message allocating an SDCCH, after completion of establishment of the main signalling link, after having sent a CM SERVICE REQUEST message, after having successfully performed authentication and cipher mode setting
procedures, after having sent a SETUP message, after having received a CALL PROCEEDING message followed by an ALERTING message and an ASSIGNMENT COMMAND message allocating either speech full rate version 1 TCH or speech full rate version 2 TCH or speech half rate version 1 TCH (for an MS also supporting half rate), the MS sends an ASSIGNMENT COMPLETE message.

3) To verify that subsequently, after the suite of actions specified in test purposes 1 and 2, the MS after receiving a CONNECT message returns a CONNECT ACKNOWLEDGE message.

4) To verify that after the suite of actions specified in test purposes 1 and 2, the MS after receiving a CONNECT message attaches the user connection to the radio path. (This is checked by verifying that there is a point in time after transmission of the first L2 frame containing the (complete) CONNECT message, where the MS is sending appropriate speech or data frames whenever it doesn't have to transmit or acknowledge an I frame on layer 2 of the FACCH.)

These test purposes are tested for all rates supported by the MS (full rate versions 1 and 2, plus optional half rate version 1 support).

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		[TSPC_Serv_TS11]			Telephony supported
3		+EFR_BasicServiceMO(C_Telephony, C_Full_v2)			Init Full Rate v2.
4		+local_testfullrate			M = 1.
5		(TCV_Null := OM_CphMd (TCV_ch, CphMod_02, TCV_CphKey))			
6		+EFR_BasicServiceMO(C_Telephony, C_Full)			Init Full Rate v1.
7		+local_testfullrate			M = 2.
8		[TSPC_DualRate]			Half rate supported.
9		(TCV_Null := OM_CphMd (TCV_ch, CphMod_02, TCV_CphKey))			
10		+EFR_BasicServiceMO(C_Telephony, C_Half)			Init Half Rate v1.
11		+local_testhalfrate			M = 3.
12		[NOT TSPC_DualRate]			
13	E_0023	[C_Yes] local_testfullrate		I	IXIT error

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
14		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_Omit, CellChDes_Omit, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_45, BcchFreqLst_02, BcchFreqLst_03, BcchFreqLst_04, C_Restablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNA, TSPX_RANDA, C_arfcnA, C_arfcnAd, TSPX_IMSI, C_Test_nfh, C_NCCP_2)			
15		+FullRateCh_A_1(C_Ass, TSPX_TmSltDef, TSPX_TscDef, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, C_arfcn_tchA, C_arfcn_tchAd, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
16		+local_initcall			
		local_testhalfrate			
17		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Ass, TSPX_TmSltDef, TSPX_TscDef, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, C_arfcn_tchA, C_arfcn_tchAd, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			7.
18		+local_initcall			
		local_initcall			
19		(TCV_Null := OO_DialCalledNum())			4.
20		[TSPC_CalledNumDisp]			
21		(TCV_Res := OO_CalledNumCHK())			5.
22		+Check_Result			
23		+InitCall(TCV_Service)			6.
24		+localtree			
25		[NOT TSPC_CalledNumDisp]			
26		+InitCall(TCV_Service)			
27		+localtree			
		localtree			
28	E_0026	L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_04)	(P)	
29		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
30		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, TSPX_SDCCH4SubDef, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, TimingAdv(0)))		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
31		L?DL_EstInCmsRq	CMSerReq(CMServiceReq_04)		
32		ACTIVATE(OtherEventsFail)			Restore Normal default
33		L!DL_DatRqAuthRq	AuthReq(TCV_ch, AuthRequest(TSPX_CKSNA, TSPX_RANDA))		
34		L?DL_DatInAuthRes (TCV_Sres := DL_DatInAuthRes.msg.sres)	AuthRes(AuthResponse)		
35		(TCV_Res := OC_ChkSRES(TCV_Sres, TSPX_Ki, TSPX_RANDA))			
36	E_0027	[NOT TCV_Res]		(F)	
37		+PostMainLinkRel(TCV_ch)			
38		[TCV_Res]			
39		+localtree1			
		localtree1			
40		+Ciphering_on(TCV_ch)			
41		+EFR_RcvSetupOrEsetup(TCV_Setup_mo, TCV_Esetup, TCV_Ecall)			
42	E_0028	L!DL_DatRqCallProc	CallProc(TCV_ch, TCV_CallProc)	(P)	
43		L!DL_DatRqAlert	AlertSnd(TCV_ch, Alerting_01(TCV_TI))		
44		[TSPC_AlertInd]			
45		(TCV_Res := OO_AltIndCHK())			
46		+Check_Result			
47		+localtree2			
48		[NOT TSPC_AlertInd]			
49		+localtree2			
		localtree2			
50		+AssCmdGen(TCV_cellid, TCV_ChRate, TSPX_TmSltDef, TSPX_TscDef, TSPX_TCHHSubDef)			
51		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
52		+localtree3			
		localtree3			
53		L!DL_DatRqConn	ConnSnd(TCV_chTch, Connect_02(TCV_TI))		
54	E_0030	L?DL_DatInConnAck	ConnAckRcv(ConnectAck_02(TCV_TI0))	(P)	
55		(TCV_Res := OO_TCHThroConnCHK())			
56		+Check_Result			
57		L!DL_DatRqDisc	DiscSnd(TCV_chTch, DisconnS(TCV_TI, Cause_01, ProgInd_omit, UuInfo_omit))		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
58	E_0033	L?DL_DatInRel	ReleaseRcv(Release_10(TCV_TI0))	(P)	
59		L!DL_DatRqRelCmp	RelComSnd(TCV_chTch, ReleaseCmp_08(TCV_TI))		
60		+PostMainLinkRel(TCV_chTch)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_4

Group : EFR/

Purpose : 1) In acceptance to a SETUP message indicating speech, the MS shall indicate and include in the CALL CONFIRMED message all the speech versions that it supports.

2) To verify that upon receipt of the ASSIGNMENT COMMAND message specifying either speech full rate version 1 or speech full rate version 2 or speech half rate verson 1 (for an MS also supporting half rate), the mobile station continues a mobile terminating call establishment with early assignment of traffic channel

a) by replying to the ASSIGNMENT COMMAND with an ASSIGNMENT COMPLETE message and,

b) by continuing the call establishment by through connecting TCH in both directions if it supports immediate connect, or by sending an ALERTING message otherwise.

3) To verfiy that for speech calls, themobile station shall attach the user connection at latest when sending the connect message, except if there is no compatible radio resource available at this time. In this case the attachment shall be delayed until such a resource becomes available.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2	body	[TSPC_Serv_TS11]			
3		+EFR_BasicServiceMT(C_Telephony, C_Full_v2)			
4		+local_testfullrate			M = 1.
5		(TCV_Null := OM_CphMd (TCV_ch, CphMod_02, TCV_CphKey))			
6		+EFR_BasicServiceMT(C_Telephony, C_Full)			
7		+local_testfullrate			M = 2.
8		[TSPC_DualRate]			
9		(TCV_Null := OM_CphMd (TCV_ch, CphMod_02, TCV_CphKey))			
10		+EFR_BasicServiceMT(C_Telephony, C_Half)			
11		+local_testhalfrate			M = 3.
12		[NOT TSPC_DualRate]			
13	E_0034	[C_Yes] local_testfullrate		I	
14		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_Omit, CellChDes_Omit, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_45, BcchFreqLst_02, BcchFreqLst_03, BcchFreqLst_04, C_Restablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNA, TSPX_RANDA, C_arfcnA, C_arfcnAd, TSPX_IMSI, C_Test_nfh, C_NCCP_2)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
15		+FullRateCh_A_1(C_Ass, TSPX_TmSltDef, TSPX_TscDef, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, C_arfcn_tchA, C_arfcn_tchAd, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
16		+localtree(C_Full)			
		local_testhalfrate			
17		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Ass, TSPX_TmSltDef, TSPX_TscDef, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, C_arfcn_tchA, C_arfcn_tchAd, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			4.
18		+localtree(C_Half)			
		localtree(rate : IA5String)			
19		L!DL_UdatRqPg1Rq	PgReq1(TCV_PgCh, TCV_Pgg, PgReqTp1_01)		
20		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_17)		
21		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
22		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, TSPX_SDCCH4SubDef, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, TimingAdv(0)))		
23		L?DL_EstInPgRes	PagingRes(PagingRes_03(TSPX_CKSNDef))		
24		ACTIVATE(OtherEventsFail)			Restore Normal default
25		L!DL_DatRqAuthRq	AuthReq(TCV_ch, AuthRequest(TSPX_CKSNA, TSPX_RANDA))		
26		L?DL_DatInAuthRes(TCV_Sres := DL_DatInAuthRes.msg.sres)	AuthRes(AuthResponse)		
27		(TCV_Res := OC_ChkSRES(TCV_Sres, TSPX_Ki, TSPX_RANDA))			
28	E_0035	[NOT TCV_Res]		(F)	
29		+PostMainLinkRel(TCV_ch)			
30	E_0036	[TCV_Res]		(P)	5.
31		+localtree1(rate)			
		localtree1(rate : IA5String)			
32		+Ciphering_on(TCV_ch)			
33		L!DL_DatRqSetup	SetupSnd(TCV_ch, TCV_Setup_mt)		6.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
34		+ltree_RcvCallConfirmed			
35		+EFR_ChkRcvSpchPrefs			check all spchprefs are present in bcap
36		[TSPX_Immconn]			
37		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
38		+asstrafficch(rate)			7.
39		+localtree3			
40		[NOT TSPX_Immconn]			
41		+asstrafficch(rate)			7.
42		L?DL_DatInAlert	AlertRcv(AlertingInd_01(TI_01))		
43		(TCV_Res := OO_AltIndCHK())			
44		+Check_Result			
45		(TCV_Null := OO_HookOff())			9.
46		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
47		+localtree3			
		localtree3			
48		[TCV_Setup_mt.bcap1.itc = '000'B]			Speech Call
49		(TCV_Res := OO_TCHThroConnCHK())			
50		+Check_Result			
51		L!DL_DatRqConnAck	ConnAckSnd(TCV_chTch, ConnectAck_01(TI_02))		
52		+localtree5			
53		[TCV_Setup_mt.bcap1.itc <> '000'B]			Data Call
54		L!DL_DatRqConnAck	ConnAckSnd(TCV_chTch, ConnectAck_01(TI_02))		
55		(TCV_Res := OO_TCHThroConnCHK())			
56		+Check_Result			
57		+localtree5			
		localtree5			
58		(TCV_Null := OO_TermCall())			
59		L?DL_DatInDisc	DiscRcv(TCV_chTch, DisconnR(TI_01, Cause_Def))		
60		L!DL_DatRqRel	ReleaseSnd(TCV_chTch, Release_08(TI_02))		
61		L?DL_DatInRelCmp	RelComRcv(ReleaseCmp_03(TI_01))		
62		+PostMainLinkRel(TCV_chTch)			
		ltree_RcvCallConfirmed			
63		[TSPX_EFR_SpchPrefsNo = 2]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
64		L?DL_DatInCallCo(TCV_CallCfm := DL_DatInCallCo.msg, TCV_EFR_svi1 := DL_DatInCallCo.msg.bcap1.oct_3a_etc_1.svi, TCV_EFR_svi2 := DL_DatInCallCo.msg.bcap1.oct_3a_etc_2.svi)	CallCfm(CallConfirm_04(TI_01, TCV_BcapMO1))		
65		[TSPX_EFR_SpchPrefsNo = 3]			
66		L?DL_DatInCallCo(TCV_CallCfm := DL_DatInCallCo.msg, TCV_EFR_svi1 := DL_DatInCallCo.msg.bcap1.oct_3a_etc_1.svi, TCV_EFR_svi2 := DL_DatInCallCo.msg.bcap1.oct_3a_etc_2.svi, TCV_EFR_svi3 := DL_DatInCallCo.msg.bcap1.oct_3a_etc_3.svi)	CallCfm(CallConfirm_04(TI_01, TCV_BcapMO1))		
67	E_0043	[C_Yes] asstrafficch(rate : IA5String)		I	IXIT ERROR
68		(TCV_Res := FALSE)			
69		+AssCmdGen(TCV_cellid, rate, TSPX_TmSltDef, TSPX_TscDef, TSPX_TCHHSubDef)			
70		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_26_12_5

Group : EFR/

Purpose : 1) To verify that an MS supporting speech in the MM state "idle, updated", when made to call the number 112, sends a CHANNEL REQUEST message with establishment cause "emergency call".

2) To verify that after assignment of a dedicated channel the first layer message sent by the MS on the assigned dedicated channel is a CM SERVICE REQUEST message specifying the correct CKSN and TMSI, with CM Service Type "emergency call establishment" .

3) To verify that authentication and cipher mode setting are performed successfully.

4) To verify that after cipher mode setting acceptance by the SS, the MS sends an EMERGENCY SETUP message.

5) The EFR mobile station shall accept channel assignment to a TCH full rate speech version 1 or 2 and if it supports half rate, in addition it shall accept channel assignment to a TCH half rate speech verson 1.

6) After receipt of a CONNECT ACKNOWLEDGE message during correct establishment of the emergency call, the TCH shall be through connected in both directions if an appropriate TCH is available.

7) To verify that the call is cleared correctly.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(900)			
2		+IdleUpdated(C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_Omit, CellChDes_Omit, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_45, BcchFreqLst_02, BcchFreqLst_03, BcchFreqLst_04, C_Restablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNA, TSPX_RANDA, C_arfcnA, C_arfcnAd, TSPX_IMSI, C_Test_nfh, C_NCCP_2)			1.
3		[TSPC_Serv_TS11]			Telephony supported
4		+EFR_BasicServiceMO(C_EmgCall, C_Full)			Init Full Rate v1.
5		+local_make_ecall			M = 1.
6		[TSPC_EFR_EmgCallBcap]			BCAP present
7		(TCV_Null := OM_CphMd (TCV_ch, CphMod_02, TCV_CphKey))			
8		+EFR_BasicServiceMO(C_EmgCall, C_Full_v2)			Init Full Rate v2.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9		+local_make_ecall			M = 2.
10		[TSPC_DualRate]			Half rate
11		(TCV_Null := OM_CphMd (TCV_ch, CphMod_02, TCV_CphKey))			
12		+EFR_BasicServiceMO(C_EmgCall, C_Half)			Init Half Rate v1.
13		+local_make_ecall			M = 3.
14		[NOT TSPC_DualRate]			
15		[NOT TSPC_EFR_EmgCallBcap]			
16	E_0044	[C_Yes]		I	IXIT ERROR
		local_make_ecall			
17	body	+InitCall(TCV_Service)			
18	E_0045	L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_16)	(P)	2.
19		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
20		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, TSPX_SDCCH4SubDef, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, TimingAdv(0)))		
21	E_0046	L?DL_EstInCmsRq	CMSerReq(CMServiceReq_06(TSPX_CKSNA))	(P)	3.
22		ACTIVATE(OtherEventsFail)			Restore Normal default
23		L!DL_DatRqAuthRq	AuthReq(TCV_ch, AuthRequest(TSPX_CKSNA, TSPX_RANDA))		
24		L?DL_DatInAuthRes(TCV_Sres := DL_DatInAuthRes.msg.sres)	AuthRes(AuthResponse)		
25		+local_continue			
		local_continue			
26		(TCV_Res := OC_ChkSRES(TCV_Sres, TSPX_Ki, TSPX_RANDA))			
27	E_0047	[NOT TCV_Res]		(F)	4.
28		+PostMainLinkRel(TCV_ch)			
29	E_0048	[TCV_Res]		(P)	5.
30		+Ciphering_on(TCV_ch)			
31		+EFR_RcvSetupOrEsetup(TCV_Setup_mo, TCV_Esetup, TCV_Ecall)			6.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
32		+local_preferred_rate			
33		L!DL_DatRqCallProc	CallProc(TCV_ch, TCV_CallProc)		
34		L!DL_DatRqAlert	AlertSnd(TCV_ch, Alerting_01(TCV_TI))		
35		+AssCmdGen(TCV_cellid, TCV_ChRate, TSPX_TmSltDef, TSPX_TscDef, TSPX_TCHHSubDef)			
36		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
37		+local_continue1			
		local_continue1			
38		L!DL_DatRqConn	ConnSnd(TCV_chTch, Connect_02(TCV_TI))		
39	E_0049	L?DL_DatInConnAck	ConnAckRcv(ConnectAck_02(TCV_TI0))	(P)	
40		(TCV_Res := OO_TCHThroConnCHK())			
41		+Check_Result			
42		L!DL_DatRqDisc	DiscSnd(TCV_chTch, DisconnS(TCV_TI, Cause_01, ProgInd_omit, UuInfo_omit))		
43	E_0052	L?DL_DatInRel	ReleaseRcv(Release_10(TCV_TI0))	(P)	
44		L!DL_DatRqRelCmp	RelComSnd(TCV_chTch, ReleaseCmp_08(TCV_TI))		
45		+PostMainLinkRel(TCV_chTch)			
		local_preferred_rate			
46		(TCV_Rchr := TCV_Setup_mo1.bcap1.rchr)			
47		[TCV_ChRate = C_Full]			9.
48		+FullRateCh_A_1(C_Ass, TSPX_TmSltDef, TSPX_TscDef, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, C_arfcn_tchA, C_arfcn_tchAd,TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
49		[(TCV_ChRate = C_Half) AND ((TCV_Rchr = '10'B) OR (TCV_Rchr = '11'B))]			10.
50		+HalfRateCh_A_1(TSPX_TCHHSubDef, C_Ass, TSPX_TmSltDef, TSPX_TscDef, TCV_ChMod, FreqTCH_omit, FreqTCH_omit, C_arfcn_tchA, C_arfcn_tchAd, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			

Detailed Comments :

1. To setup a physical channels as BCCH CCCH and SDCCH4 combined channel.
2. CHANNEL REQUEST with "emergency call establishment" cause received.
3. CM SERVICE REQUEST with "emergency call establishment" service type correct TMSI and CKSN received.
4. Authentication failed.
5. Authentication passed.
6. EMERGENCY SETUP with BC or without BC received.
7. The TCH channel is not through connected, fail.
8. The TCH channel is through connected.
9. dual rate / full rate preferred" case. setup a physical channel as full rate traffic channel and the preferred full rate channel is to be assigned.
10. dual rate / half rate preferred" case. setup a physical channel as half rate traffic channel and the preferred half rate channel is to be assigned.

Test Step Dynamic Behaviour

Test Step Name : Config_CBMS_A_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH, SDCCH4 and CBCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, C_FCCH_A, C_SCH_A, C_BCCH_A_1, C_PCH_A_1, C_AGCH_A_1, C_RACH_A_1, C_SDCCH4_A, C_SACCHC4_A, C_CBCH_A))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_A, C_SDCCH40_A, C_SDCCH41_A, C_SDCCH42_A, C_SDCCH43_A, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_A, C_SACCHC40_A, C_SACCHC41_A, C_SACCHC42_A, C_SACCHC43_A, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_FACCHF_A_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_FACCHF_A_1, C_SACCHF_A_1, "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_FACCHF_A_2(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 2 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_FACCHF_A_2, C_SACCHF_A_2, "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_FACCHF_B_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's of cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellB, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_FACCHF_B_1, C_SACCHF_B_1, "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_FACCHH_A_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_FACCHH_A_1, C_SACCHH_A_1, "dummy"))			
2		(TCV_Null := OM_Assoc(C_FACCHH_A_1, C_FACCHH0_A_1, C_FACCHH1_A_1, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_SACCHH_A_1, C_SACCHH0_A_1, C_SACCHH1_A_1, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_FACCHH_A_2(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_FACCHH_A_2, C_SACCHH_A_2, "dummy"))			
2		(TCV_Null := OM_Assoc(C_FACCHH_A_2, C_FACCHH0_A_2, C_FACCHH1_A_2, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_SACCHH_A_2, C_SACCHH0_A_2, C_SACCHH1_A_2, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_FACCHH_B_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's of cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellB, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_FACCHH_B_1, C_SACCHH_B_1, "dummy"))			
2		(TCV_Null := OM_Assoc(C_FACCHH_B_1, C_FACCHH0_B_1, C_FACCHH1_B_1, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_SACCHH_B_1, C_SACCHH0_B_1, C_SACCHH1_B_1, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_A_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, C_FCCH_A, C_SCH_A, C_BCCH_A_1, C_PCH_A_1, C_AGCH_A_1, C_RACH_A_1, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_A_ho_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell A with controlable td and fn.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, td, fn, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, C_FCCH_A, C_SCH_A, C_BCCH_A_1, C_PCH_A_1, C_AGCH_A_1, C_RACH_A_1, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_A_2(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map the second BCCH, CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", C_BCCH_A_2, C_PCH_A_2, C_AGCH_A_2, C_RACH_A_2, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_A_3(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map the third BCCH, CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", C_BCCH_A_3, C_PCH_A_3, C_AGCH_A_3, C_RACH_A_3, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_A_4(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map the fourth BCCH, CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", C_BCCH_A_4, C_PCH_A_4, C_AGCH_A_4, C_RACH_A_4, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_B_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellB, C_FCCH_B, C_SCH_B, C_BCCH_B_1, C_PCH_B_1, C_AGCH_B_1, C_RACH_B_1, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_B_ho_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell B with controlable td and fn.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, td, fn, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellB, C_FCCH_B, C_SCH_B, C_BCCH_B_1, C_PCH_B_1, C_AGCH_B_1, C_RACH_B_1, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_A_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, C_FCCH_A, C_SCH_A, C_BCCH_A_1, C_PCH_A_1, C_AGCH_A_1, C_RACH_A_1, C_SDCCH4_A, C_SACCHC4_A, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_A, C_SDCCH40_A, C_SDCCH41_A, C_SDCCH42_A, C_SDCCH43_A, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_A, C_SACCHC40_A, C_SACCHC41_A, C_SACCHC42_A, C_SACCHC43_A, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_A_ho_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A with controlable ta and fn.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, td, fn, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, C_FCCH_A, C_SCH_A, C_BCCH_A_1, C_PCH_A_1, C_AGCH_A_1, C_RACH_A_1, C_SDCCH4_A, C_SACCHC4_A, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_A, C_SDCCH40_A, C_SDCCH41_A, C_SDCCH42_A, C_SDCCH43_A, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_A, C_SACCHC40_A, C_SACCHC41_A, C_SACCHC42_A, C_SACCHC43_A, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_B_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellB, C_FCCH_B, C_SCH_B, C_BCCH_B_1, C_PCH_B_1, C_AGCH_B_1, C_RACH_B_1, C_SDCCH4_B, C_SACCHC4_B, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_B, C_SDCCH40_B, C_SDCCH41_B, C_SDCCH42_B, C_SDCCH43_B, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_B, C_SACCHC40_B, C_SACCHC41_B, C_SACCHC42_B, C_SACCHC43_B, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_B_ho_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B with controlable ta and fn.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, td, fn, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellB, C_FCCH_B, C_SCH_B, C_BCCH_B_1, C_PCH_B_1, C_AGCH_B_1, C_RACH_B_1, C_SDCCH4_B, C_SACCHC4_B, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_B, C_SDCCH40_B, C_SDCCH41_B, C_SDCCH42_B, C_SDCCH43_B, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_B, C_SACCHC40_B, C_SACCHC41_B, C_SACCHC42_B, C_SACCHC43_B, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_C_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell C.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellC, C_FCCH_C, C_SCH_C, C_BCCH_C_1, C_PCH_C_1, C_AGCH_C_1, C_RACH_C_1, C_SDCCH4_C, C_SACCHC4_C, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_C, C_SDCCH40_C, C_SDCCH41_C, C_SDCCH42_C, C_SDCCH43_C, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_C, C_SACCHC40_C, C_SACCHC41_C, C_SACCHC42_C, C_SACCHC43_C, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH8_A_1(par_bspwr , par_mspwr : INTEGER; chmod : CHMOD; acttype : BITSTRING; slot : SN; tsc : TSC; rf : FRQPARA; chcmbn : LOGCH; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as SDCCH8 channel for instance 1 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_ChConf(par_bspwr, par_mspwr, acttype, chmod, ta, slot, tsc, rf, chcmbn, 0, 0, babr, cch_con, bpm, PgReqTp1Norm, '0'B, '0'B, C_CellA, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_SDCCH8_A_1, C_SACCHC8_A_1, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH8_A_1, C_SDCCH80_A_1, C_SDCCH81_A_1, C_SDCCH82_A_1, C_SDCCH83_A_1, C_SDCCH84_A_1, C_SDCCH85_A_1, C_SDCCH86_A_1, C_SDCCH87_A_1, C_SACCHC8_A_1, C_SACCHC80_A_1, C_SACCHC81_A_1, C_SACCHC82_A_1, C_SACCHC83_A_1, C_SACCHC84_A_1, C_SACCHC85_A_1, C_SACCHC86_A_1, C_SACCHC87_A_1))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_A_CBMS(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.

Default : OtherEvents

Comments : time slot = 0, ARFCN = 20 (GSM900) or ARFCN = 590 (DCS1800) cell A for RR testing.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_CBMS_A_1(par_bspwr,mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_CBCH_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_CBMS_A_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_CBCH_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_A(par_bspwr , mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_SDCCH4_A_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_SDCCH4_A_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_A_ho(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_SDCCH4_A_ho_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm, td, fn)			
3		[TSPC_DCS]			2.
4		+Config_SDCCH4_A_ho_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm, td, fn)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_B(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_SDCCH4_B_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_SDCCH4_B_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_B_ho(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_SDCCH4_B_ho_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm, td, fn)			
3		[TSPC_DCS]			2.
4		+Config_SDCCH4_B_ho_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm, td, fn)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_C(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell C.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_SDCCH4_C_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_SDCCH4_C_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as FCHH_SCH_BCCH_CCCH for cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_A_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_A_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A_2(acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map the second BCCH, CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_A_2(63, 19, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnA), C_BCCH_CCCH_2, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_A_2(63, 15, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnAd), C_BCCH_CCCH_2, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A_3(acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map the third BCCH, CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments : time slot = 4, ARFCN = TSPX_BCCHcarrierA

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_A_3(63, 19, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnA), C_BCCH_CCCH_3, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_A_3(63, 15, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnAd), C_BCCH_CCCH_3, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A_4(acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map the fourth BCCH, CCCH onto the physical channel which represents cell A.

Default : OtherEvents

Comments : time slot = 6, ARFCN = TSPX_BCCHcarrierA

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_A_4(63, 19, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnA), C_BCCH_CCCH_4, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_A_4(63, 15, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnAd), C_BCCH_CCCH_4, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A_ho(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell A for RR testing.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_A_ho_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm, td, fn)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_A_ho_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm, td, fn)			

Detailed Comments : 1. For P−GSM900.
1.1 Frequence and cell_id for cell A in HO cases
2. For DCS1800.
2.1 Frequence and cell_id for cell A in HO cases

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_B(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell B for RR testing.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_B_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_B_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P−GSM900.
1.1 Frequenz and cell_id for cell B in HO cases
2. For DCS1800.
2.1 Frequenz and cell_id for cell B in HO cases

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_B_ho(par_bspwr, mtmcgsm, mtmcdcs, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3; td, fn : INTEGER)

Group : management/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell B for RR testing.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			1.
2		+Config_BCCH_CCCH_B_ho_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm, td, fn)			
3		[TSPC_DCS]			2.
4		+Config_BCCH_CCCH_B_ho_1(par_bspwr, mtmcdcs, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn2), C_FCCH_SCH_BCCH_CCCH, ta, babr, cch_con, bpm, td, fn)			

Detailed Comments : 1. For P−GSM900.
1.1 Frequenz and cell_id for cell B in HO cases
2. For DCS1800.
2.1 Frequenz and cell_id for cell B in HO cases

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_1(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg, Freqd : FRQPARA; arfcng , arfcnd : INTEGER; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 1 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[arfcng <> C_Noarfcn]			Non hopping
2		+FullRateCh_A_1_nociph(acttypeT, slotT, tscT, chmod, FreqTCH(arfcng) , FreqTCH(arfcnd), ta, babr, cch_con, bpm)			
3		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			
4		[TSPC_PGSM OR TSPC_EGSM]			
5		(TCV_tch_arfcn:= arfcng)			
6		[TSPC_DCS]			
7		(TCV_tch_arfcn:= arfcnd)			
8		[arfcng = C_Noarfcn]			Hopping
9		+FullRateCh_A_1_nociph(acttypeT, slotT, tscT, chmod, Freqg , Freqd, ta, babr, cch_con, bpm)			
10		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_1_nociph(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg, Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 1 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch := C_FACCHF_A_1, TCV_sacchTch := C_SACCHF_A_1, TCV_chtype := '00001'B)			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_FACCHF_A_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHF_ACCHF_1, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			
5		[TSPC_DCS]			
6		+Config_FACCHF_A_1(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_TCHF_ACCHF_1, ta, babr, cch_con, bpm)			
7		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_2(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 2 of cell A for TC_26_6_13_1.

Default : OtherEvents

Comments : used as after time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+FullRateCh_A_2_nociph(acttypeT, slotT, tscT, chmod, Freqg ,Freqd, ta, babr, cch_con, bpm)			
2		(TCV_Null := OM_CphMd(TCV_chTch1, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_2_nociph(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 2 of cell A for TC_26_6_13_1.

Default : OtherEvents

Comments : used as after time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch1 := C_FACCHF_A_2, TCV_sacchTch1 := C_SACCHF_A_2)			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_FACCHF_A_2(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHF_ACCHF_2, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacchTch1, TCV_sysinfo5, TCV_sysinfo6)			
5		[TSPC_DCS]			
6		+Config_FACCHF_A_2(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_TCHF_ACCHF_2, ta, babr, cch_con, bpm)			
7		+SysInfo_SacchSending(TCV_sacchTch1, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_B_1(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 1 of cell B.

Default : OtherEvents

Comments : FullRateCh_B_1.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+FullRateCh_B_1_nociph(acttypeT, slotT, tscT, chmod, Freqg, Freqd, ta, babr, cch_con, bpm)			
2		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_B_1_nociph(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHF_ACCH's for instance 1 of cell B.

Default : OtherEvents

Comments : Used in TC_26_6_13_5 for after time channel

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch := C_FACCHF_B_1, TCV_sacchTch := C_SACCHF_B_1)			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_FACCHF_B_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHF_ACCHF_1, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5_B, TCV_sysinfo6_B)			
5		[TSPC_DCS]			
6		+Config_FACCHF_B_1(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_TCHF_ACCHF_1, ta, babr, cch_con, bpm)			
7		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5_B, TCV_sysinfo6_B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_A_1(sub : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg, Freqd : FRQPARA; arfcng , arfcnd : INTEGER; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's for instance 1 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[arfcng <> C_Noarfcn]			not hopping
2		+HalfRateCh_A_1_nociph(sub, acttypeT, slotT, tscT, chmod, FreqTCH(arfcng) , FreqTCH(arfcnd), ta, babr, cch_con, bpm)			
3		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			
4		[TSPC_PGSM OR TSPC_EGSM]			
5		(TCV_tch_arfcn:= arfcng)			
6		[TSPC_DCS]			
7		(TCV_tch_arfcn:= arfcnd)			
8		[arfcng = C_Noarfcn]			hopping
9		+HalfRateCh_A_1_nociph(sub, acttypeT, slotT, tscT, chmod, Freqg , Freqd, ta, babr, cch_con, bpm)			
10		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_A_1_nociph(sub : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg, Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's for instance 1 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch := OC_SubchOfFacchh(sub, C_CellA, 1), TCV_sacchTch := OC_SubchOfSacchh(sub, C_CellA, 1))			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_FACCHH_A_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHH_ACCHH_1, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			
5		+localtree			
6		[TSPC_DCS]			
7		+Config_FACCHH_A_1(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_TCHH_ACCHH_1, ta, babr, cch_con, bpm)			
8		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			
9		+localtree			
		localtree			
10		[sub = '0'B]			
11		(TCV_chtype := '00010'B)			
12		[sub = '1'B]			
13		(TCV_chtype := '00011'B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_A_2(bstring : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's for instance 2 of cell A for TC_26_6_13_1.

Default : OtherEvents

Comments : used for after time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+HalfRateCh_A_2_nociph(bstring, acttypeT, slotT, tscT, chmod, Freqg, Freqd, ta, babr, cch_con, bpm)			
2		(TCV_Null := OM_CphMd(TCV_chTch1, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_A_2_nociph(bstring : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's for instance 2 of cell A for TC_26_6_13_1.

Default : OtherEvents

Comments : used for after time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch1 := OC_SubchOfFacchh(bstring, C_CellA, 2), TCV_sacchTch1 := OC_SubchOfSacchh(bstring, C_CellA, 2))			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_FACCHH_A_2(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHH_ACCHH_2, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacchTch1, TCV_sysinfo5, TCV_sysinfo6)			
5		[TSPC_DCS]			
6		+Config_FACCHH_A_2(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_TCHH_ACCHH_2, ta, babr, cch_con, bpm)			
7		+SysInfo_SacchSending(TCV_sacchTch1, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_B_1(bstring : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's for instance 1 of cell B

Default : OtherEvents

Comments : used as after time channel assigned by handover command

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+HalfRateCh_B_1_nociph(bstring, acttypeT, slotT, tscT, chmod, Freqg, Freqd, ta, babr, cch_con, bpm)			
2		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_B_1_nociph(bstring : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as TCHH_ACCH's for instance 1 of cell B

Default : OtherEvents

Comments : used as after time channel assigned by handover command

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch := OC_SubchOfFacchh(bstring, C_CellB, 1), TCV_sacchTch := OC_SubchOfSacchh(bstring, C_CellB, 1))			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_FACCHH_B_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHH_ACCHH_1, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5_B, TCV_sysinfo6_B)			
5		[TSPC_DCS]			
6		+Config_FACCHH_B_1(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_TCHH_ACCHH_1, ta, babr, cch_con, bpm)			
7		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5_B, TCV_sysinfo6_B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SDCCH8_A_1_nociph(bstring : B_3; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg ,Freqd : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/

Objective : To set one physical channel used as SDCCH8 channel for instance 1 of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_ch := OC_SubchOfSdcch8(bstring, C_CellA, 1), TCV_sacch8 := OC_SubchOfSacch8(bstring, C_CellA, 1))			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		+Config_SDCCH8_A_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_SDCCH8_SACCHC8_1, ta, babr, cch_con, bpm)			
4		+SysInfo_SacchSending(TCV_sacch8, TCV_sysinfo5, TCV_sysinfo6)			
5		[TSPC_DCS]			
6		+Config_SDCCH8_A_1(63, 15, chmod, acttypeT, slotT, tscT, Freqd, C_SDCCH8_SACCHC8_1, ta, babr, cch_con, bpm)			
7		+SysInfo_SacchSending(TCV_sacch8, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Adjust_gsmanddcs_powerlvl(powerlevel1,powerlevel2 : INTEGER; pdu_ass: ASS_CMD_PDU)
Group : Miscellaneous/
Objective :
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TSPC_PGSM OR TSPC_EGSM]			
2		[powerlevel1 <> 0]			
3		(pdu_ass.pcmd := Pcmd_19(INT_TO_BIT(powerlevel1,5)))			
4		[powerlevel1 = 0]			
5		[TSPC_DCS]			
6		(pdu_ass.pcmd := Pcmd_19(INT_TO_BIT(powerlevel2,5)))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : AssCh_complete(oldch,newch : LOGICCH; pdu_ass: ASS_CMD_PDU)
Group : Miscellaneous/
Objective : To provide a generic test step to Assign a traffic channel. Assign complete expected and verdict is PASS.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqAssCmd START T_dlyAss	AssCmd(oldch,pdu_ass)		
2	E_0053	?TIMEOUT T_dlyAss		(F)	
3		+PostLinkRelEnd(oldch)			
4		L?DL_EstIn (TCV_FnAss := DL_EstIn.fn) CANCEL T_dlyAss	DLEstInd(newch)		
5	E_0054	L?DL_DatInAssCom	AssCmp(newch, AsgnCmp_02)	(P)	
6		L!MDL_RelRq	MDLRelReq(oldch)		
7		L?DL_EstIn CANCEL T_dlyAss	DLEstInd(oldch)		
8	E_0055	L?DL_DatInAssfl	AssFl_any_cau(oldch)	(F)	
9		+PostLinkRelEnd(oldch)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : AssCmdGen(cell : CellID; rate : IA5String; slot : SN; tsc : TSC; sch : B_1)
Group : Miscellaneous/
Objective : To generate the ASSIGNMENT COMMAND message.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[rate <> C_Half]			Full rate
2		(TCV_Tchtype := '00001'B)			
3		+ltree_Asgn			
4		[rate = C_Half]			Half rate
5		[sch = '0'B]			
6		(TCV_Tchtype := '00010'B)			
7		+ltree_Asgn			
8		[sch = '1'B]			
9		(TCV_Tchtype := '00011'B)			
10		+ltree_Asgn			
		ltree_Asgn			
11		[cell = C_CellA]			
12		+ltree_AsgnTchA			
13		[cell = C_CellB]			
14		+ltree_AsgnTchB			
		ltree_AsgnTchA			
15		[TSPC_PGSM OR TSPC_EGSM]			
16		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, slot, tsc, 7, C_arfcn_tchA, CellChDes_Omit, TCV_ChMod, StartingTm_omit, CphMod_omit))			
17		[TSPC_DCS]			
18		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, slot, tsc, 3, C_arfcn_tchAd, CellChDes_Omit, TCV_ChMod, StartingTm_omit, CphMod_omit))			
		ltree_AsgnTchB			
19		[TSPC_PGSM OR TSPC_EGSM]			
20		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, slot, tsc, 7, C_arfcn_tchB, CellChDes_Omit, TCV_ChMod, StartingTm_omit, CphMod_omit))			
21		[TSPC_DCS]			
22		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, slot, tsc, 3, C_arfcn_tchBd, CellChDes_Omit, TCV_ChMod, StartingTm_omit, CphMod_omit))			

Detailed Comments : The test step makes the assumption that the channel rate is known when the test step is called.

Test Step Dynamic Behaviour

Test Step Name : AssCmdGen_fh(rate : IA5String; slot : SN; tsc : TSC; sch : B_1; par_pwlvlg, par_pwlvld : INTEGER; maio : MAIO; hsn : HSN; frql_g, frql_d : FRQL; cchd_g, cchd_d : CCHD; chmod : CHMOD; ma_g, ma_d : MA; Cphms : CPHMS)

Group : Miscellaneous/

Objective : To generate a ASSIGNMENT COMMAND message in case of hopping without starting time.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[rate <> C_Half]			Full rate
2		(TCV_Tchtype := '00001'B)			
3		+ltree_Asgn			
4		[rate = C_Half]			Half rate
5		[sch = '0'B]			
6		(TCV_Tchtype := '00010'B)			
7		+ltree_Asgn			
8		[sch = '1'B]			
9		(TCV_Tchtype := '00011'B)			
10		+ltree_Asgn			
		ltree_Asgn			
11		[TSPC_PGSM OR TSPC_EGSM]			
12		(TCV_AssCmd := AsgnCmd_fh(slot, tsc, TCV_Tchtype, par_pwlvlg, maio, hsn, frql_g, cchd_g, chmod, ma_g, Cphms))			
13		[TSPC_DCS]			
14		(TCV_AssCmd := AsgnCmd_fh(slot, tsc, TCV_Tchtype, par_pwlvld, maio, hsn, frql_d, cchd_d, chmod, ma_d, Cphms))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Authentication(ch: LOGICCH; cksn: BITSTRING; rand : RAND)

Group : Miscellaneous/

Objective :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqAuthRq	AuthReq(ch, AuthRequest(cksn, rand))		
2		L?DL_DatInAuthRes (TCV_Sres := DL_DatInAuthRes.msg.sres)	AuthRes(AuthResponse)		
3		(TCV_Res := OC_ChkSRES(TCV_Sres, TSPX_Ki, rand))			
4	E_0056	[NOT TCV_Res]		(F)	1)
5	E_0057	[TCV_Res]		(P)	

Detailed Comments : Authentication fails.

Test Step Dynamic Behaviour

Test Step Name : CallProcGen(setup : SETUP_MO_PDU; srv : SERVICES)

Group : Miscellaneous/

Objective : To generate a CALL PROCEEDING message which is compatible with input value 'setup'..

Default : OtherEvents

Comments : The generated CALL PROCEEDING is in TCV_CallProc. This test step can handle the following dual mode services: the first BCAP is speech or the second BCAP is speech

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_CallProc := CallProced_03, TCV_CallProc.ti := TCV_TI)			
2		[(srv = C_AltSpchData) OR (srv = C_SpchData) OR (srv = C_AltSpchFax)]			dual service
3		[setup.bcap1.itc = C_Speech]			1st BCAP is speech
4		(TCV_ce3 := setup.bcap2.ce)			
5		[(TCV_ce3 <> C_BothT) OR (TCV_ce3 <> C_BothNT)]			1.
6		[(TCV_ce3 = C_BothT) OR (TCV_ce3 = C_BothNT)]			2.
7		(TCV_CallProc.bcap1 := setup.bcap1, TCV_CallProc.bcap1.rchr := '00'B, TCV_CallProc.bcri := setup.bcri, TCV_CallProc.bcap2 := setup.bcap2, TCV_CallProc.bcap2.ce := INT_TO_BIT((BIT_TO_INT(setup.bcap2.ce) −2), 2), TCV_CallProc.bcap2.nirr := C_nirr_nomeaning)			
8		[NOT (setup.bcap1.itc = C_Speech)]			
9		(TCV_ce1 := setup.bcap1.ce)			
10		[(TCV_ce1 <> C_BothT) AND (TCV_ce1 <> C_BothNT)]			3.
11		[(TCV_ce1 = C_BothT) OR (TCV_ce1 = C_BothNT)]			4.
12		(TCV_CallProc.bcap1 := setup.bcap1, TCV_CallProc.bcap1.ce := INT_TO_BIT((BIT_TO_INT(setup.bcap1.ce) −2), 2), TCV_CallProc.bcap1.nirr := C_nirr_nomeaning, TCV_CallProc.bcri := setup.bcri, TCV_CallProc.bcap2 := setup.bcap2, TCV_CallProc.bcap2.rchr := '11'B)			
13		[(srv <> C_AltSpchData) AND (srv <> C_SpchData) AND (srv <> C_AltSpchFax)]			
14		[srv = C_EmgCall]			8.
15		[srv <> C_EmgCall]			
16		[setup.bcap1.itc = C_Speech]			5.
17		[setup.bcap1.itc <> C_Speech]			
18		(TCV_ce1 := setup.bcap1.ce)			
19		[(TCV_ce1 = C_BothT) OR (TCV_ce1 = C_BothNT)]			6.
20		(TCV_CallProc.bcap1 := setup.bcap1, TCV_CallProc.bcap1.ce := INT_TO_BIT((BIT_TO_INT(setup.bcap1.ce) −2), 2), TCV_CallProc.bcap1.nirr := C_nirr_nomeaning)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
21		[(TCV_ce1 <> C_BothT) OR (TCV_ce1 <> C_BothNT)]			7.

Detailed Comments :
1. BCAP1 is speech and BCAP2 does not reqire negotiation, the CallProc will contain no BCAP's
2. BCAP1 is speech and BCAP2 requires negotiation
3. BCAP2 is speech and BCAP1 does not reqire negotiation, the CallProc will contain no BCAP's
4. BCAP2 is speech and BCAP1 requires negotiation
5. BCAP2 does not exist and BCAP1 is speech, the CallProc will contain no BCAP's
6. BCAP2 does not exist and BCAP1 requires negotiation
7. BCAP2 does not exist and BCAP1 does not require negotiation
8. Emergency call, the CallProc will contain no BCAP

Test Step Dynamic Behaviour

Test Step Name : CCCH_group_Paging_group(ccd: CCD; imsi: HEXSTRING)

Group : Miscellaneous/

Objective : To calculate the CCCH group and the Paging group from the IMSI and the Control Channel Descriptor according to GSM 05.02, subclause 6.5.2.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[ccd.ccch_con = '000'B]			
2		(TCV_bs_cc_chans := 1)			
3		+localtree1			
4		[ccd.ccch_con = '001'B]			
5		(TCV_bs_cc_chans := 1)			
6		+localtree1			
7		[ccd.ccch_con = '010'B]			
8		(TCV_bs_cc_chans := 2)			
9		+localtree1			
10		[ccd.ccch_con = '100'B]			
11		(TCV_bs_cc_chans := 3)			
12		+localtree1			
13		[ccd.ccch_con = '110'B]			
14		(TCV_bs_cc_chans := 4)			
15		+localtree1			
		localtree1			
16		(TCV_bs_ag_blks_res := BIT_TO_INT(ccd.babr), TCV_bs_pa_mfrms := (2 + BIT_TO_INT(ccd.bpm)))			
17		[ccd.ccch_con = '001'B]			i.e. combined
18	E_0058	[(TCV_bs_ag_blks_res < 0) OR (TCV_bs_ag_blks_res > 2)]		I	Stop! A tester error.
19		[(TCV_bs_ag_blks_res >= 0) AND (TCV_bs_ag_blks_res <= 2)]			
20		(TCV_tmp := 3 − TCV_bs_ag_blks_res)			
21		+localtree2			
22		[ccd.ccch_con <> '001'B]			i.e. not combined
23	E_0059	[(TCV_bs_ag_blks_res < 0) OR (TCV_bs_ag_blks_res > 7)]		I	Stop! A tester error.
24		[(TCV_bs_ag_blks_res >= 0) AND (TCV_bs_ag_blks_res <= 7)]			
25		(TCV_tmp := 9 − TCV_bs_ag_blks_res)			
26		+localtree2			ini. TCV Pgg
		localtree2			
27		(TCV_tmp := TCV_tmp * TCV_bs_pa_mfrms)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
28		(TCV_Pgg := INT_TO_BIT((((OC_BCDtoInt(imsi, 3)) MOD (TCV_bs_cc_chans * TCV_tmp)) MOD TCV_tmp), 8), TCV_Ccchg := (((OC_BCDtoInt(imsi, 3)) MOD (TCV_bs_cc_chans * TCV_tmp)) / TCV_tmp))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CC_Est_MT_Call
Group : Miscellaneous/
Objective : To perform the CC message exchange to establish a mobile station terminating call (late assignment).
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Setup_mt.sig := Signal_01)			
2		L!DL_DatRqSetup	SetupSnd(TCV_ch, TCV_Setup_mt)		
3		L?DL_DatInCallCo	CallCfm(CallConfirm_01(TI_01))		1)
4		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
5		L?DL_DatInAlert	AlertRcv(AlertingInd_01(TI_01))		
6		(TCV_Null := OO_HookOff())			
7		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		

Detailed Comments : 1. If the MS supports the bearer capabilities, which are give in Setup message, it has to accept them. However, it may still send bearer capabilites in the Call Confirm message for either EFR or Half Rate

Test Step Dynamic Behaviour

Test Step Name : CC_EstMsTermCall(rate : IA5String; sub : B_1)

Group : Miscellaneous/

Objective : To perform the CC message exchange to establish a mobile station terminating call with non hopping channel (speech or data call). (Similar to EstMsTermFullRateCallNonFH but without initial RRmtcallprepare)

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+CC_Est_MT_Call			
2		+ltree_Asgn			
3		+Adjust_gsmanddcs_powerlvl(0, 15, TCV_AssCmd)			
4		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
5		L!DL_DatRqConnAck	ConnAckSnd(TCV_chTch, ConnectAck_01(TI_02))		
		ltree_Asgn			
6		[rate = C_Full]			Full rate
7		(TCV_Tchtype := '00001'B)			
8		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, TCV_asscmd_ts, TSPX_TscDef, 9, TCV_tch_arfcn, CellChDes_Omit, ChMod_speech_iei, StartingTm_omit, CphMod_omit))			
9		[rate = C_Full_v2]			Full rate v2
10		(TCV_Tchtype := '00001'B)			
11		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, TCV_asscmd_ts, TSPX_TscDef, 9, TCV_tch_arfcn, CellChDes_Omit, ChMod_EFR2_iei, StartingTm_omit, CphMod_omit))			
12		[rate = C_Half]			Half rate
13		(TCV_Tchtype := INT_TO_BIT((2 + BIT_TO_INT(sub)), 5))			
14		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, TCV_asscmd_ts, TSPX_TscDef, 9, TCV_tch_arfcn, CellChDes_Omit, ChMod_omit, StartingTm_omit, CphMod_omit))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Check_Cells_Present (no_of_cells:B_3; sacch:LOGICCH)

Group : Miscellaneous/

Objective : To receive a specified number of measurement reports (TSPX_Msr_Cell_Present) describing a specified number of cells present.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_StartMsrReport(sacch))			
2		(TCV_Measure_Count :=0, TCV_CellPresent :=0)			
3		REPEAT get_measurements (no_of_cells, sacch) UNTIL [TCV_CellPresent = 1]			
		get_measurements (no_of_cells:B_3; sacch:LOGICCH)			
4		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr,TCV_No_of_cells:= DL_UdatInMsrRpt.msg.msrr.no_nc)	MsrRept(MsrReport_03e(MsrResult_03a))		
5		(TCV_CellPresent:=0)			
6		[TCV_No_of_cells = no_of_cells]			
7		(TCV_Measure_Count := TCV_Measure_Count + 1)			
8		[TCV_Measure_Count = TSPX_MsrCellPresent]			
9		(TCV_CellPresent :=1)			
10		(TCV_Measure_Count :=0)			
11		(TCV_Null := OM_StopMsrReport(sacch))			
12		[TCV_Measure_Count<> TSPX_MsrCellPresent]			
13		(TCV_CellPresent:=0)			
14		[TCV_No_of_cells <> no_of_cells]			
15		(TCV_CellPresent:=0)			

Detailed Comments : 1. Convert powers to integers (TCV_mindbm, TCV_maxdbm).
2. Check power is within tolerance.
3. Convert expected power to dBm (TCV_expdbm) and work out tolerance (TCV_n). This is different for DCS vs GSM (see GSM 05.05 section 4.1.1).

Test Step Dynamic Behaviour

Test Step Name : Check_Result

Group : Miscellaneous/

Objective :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	E_00 31	[NOT TCV_Res]		(F)	
2	E_00 32	[TCV_Res]		(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Ciphering_on(ch: LOGICCH)
Group : Miscellaneous/
Objective : Send cipher command and wait for successful completion
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OM_CphMdChg(ch, CphMod_01, TCV_CphKey))			
2		L!DL_DatRqCphmCmd	CphCmd(ch, CphModeCmd_01)		
3	E_0060	L?DL_DatInCphmCom	CphCom(CphModeCmp_01)	(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : EFR_ChkRcvSpchPrefs
Group : Miscellaneous/
Objective : OtherEvents
Default :
Comments : Check speech versions received in EFR BCAP against IXIT file values. All supported speech versons must be present (in any order) in the received EFR BCAP.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+EFR_SpchPrefInBCAP(TSPX_EFR_SpchPref1)			
2		+EFR_SpchPrefInBCAP(TSPX_EFR_SpchPref2)			
3		[TSPX_EFR_SpchPrefsNo > 2]			
4		+EFR_SpchPrefInBCAP(TSPX_EFR_SpchPref3)			
5	E_0062	[TSPX_EFR_SpchPrefsNo = 2]		(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : EFR_SpchPrefInBCAP(svi : B_4)
Group : Miscellaneous/
Objective : OtherEvents
Default :
Comments : Check for an individual speech preferences being present in received EFR BCAP.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	E_00 63	[TCV_EFR_svi1 = svi]		(P)	
2	E_00 64	[TCV_EFR_svi2 = svi]		(P)	
3		[TSPX_EFR_SpchPrefsNo = 3]			
4	E_00 65	[TCV_EFR_svi3 = svi]		(P)	
5	E_00 66	[C_Yes]		I	IXIT ERROR

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : EFR_RcvSetupOrEsetup(Setup: SETUP_MO_PDU; Esetup : ESETUP_PDU; Ecall : BOOLEAN)

Group : Miscellaneous/

Objective : To check that Mobile Originated SETUP or ESETUP message contains a bearer capability IE (optional for ESETUP) with all the speech versions supported by the EFR mobile. If matching is successful then a CALL PROCEEDING message response will be sent to the EFR mobile.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[NOT Ecall]			
2		[TSPX_EFR_SpchPrefsNo = 2]			
3		L?DL_DatInSetup (TCV_Setup_mo1 := DL_DatInSetup.msg, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B, TCV_EFR_svi1 := TCV_Setup_mo1.bcap1.oct_3a_etc_1.svi, TCV_EFR_svi2 := TCV_Setup_mo1.bcap1.oct_3a_etc_2.svi)	SetupRcv(Setup)		
4		+ltree_chkspeech			
5		[TSPX_EFR_SpchPrefsNo = 3]			
6		L?DL_DatInSetup (TCV_Setup_mo1 := DL_DatInSetup.msg, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B, TCV_EFR_svi1 := TCV_Setup_mo1.bcap1.oct_3a_etc_1.svi, TCV_EFR_svi2 := TCV_Setup_mo1.bcap1.oct_3a_etc_2.svi, TCV_EFR_svi3 := TCV_Setup_mo1.bcap1.oct_3a_etc_3.svi)	SetupRcv(Setup)		
7		+ltree_chkspeech			
8	E_00 67	[C_Yes]		I	IXIT ERROR
9		[Ecall]			
10		[TSPC_EFR_EmgCallBcap]			
11		[TSPX_EFR_SpchPrefsNo = 2]			
12		L?DL_DatInESetup (TCV_Esetup1 := DL_DatInESetup.msg, TCV_Setup_mo1.ti := TCV_Esetup1.ti, TCV_Setup_mo1.bcap1 := TCV_Esetup1.bcap, TCV_Setup_mo1.ccpd := TCV_Esetup1.ccpd, TCV_Setup_mo1.mt := TCV_Esetup1.mt, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B, TCV_EFR_svi1 := TCV_Setup_mo1.bcap1.oct_3a_etc_1.svi, TCV_EFR_svi2 := TCV_Setup_mo1.bcap1.oct_3a_etc_2.svi)	ESetupRcv(Esetup)		
13		+ltree_chkspeech			
14		[TSPX_EFR_SpchPrefsNo = 3]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
15		L?DL_DatInESetup (TCV_Esetup1 := DL_DatInESetup.msg, TCV_Setup_mo1.ti := TCV_Esetup1.ti, TCV_Setup_mo1.bcap1 := TCV_Esetup1.bcap, TCV_Setup_mo1.ccpd := TCV_Esetup1.ccpd, TCV_Setup_mo1.mt := TCV_Esetup1.mt, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B, TCV_EFR_svi1 := TCV_Setup_mo1.bcap1.oct_3a_etc_1.svi, TCV_EFR_svi2 := TCV_Setup_mo1.bcap1.oct_3a_etc_2.svi, TCV_EFR_svi3 := TCV_Setup_mo1.bcap1.oct_3a_etc_3.svi)	ESetupRcv(Esetup)		
16		+ltree_chkspeech			
17	E_0068	[C_Yes]		I	IXIT ERROR
18		[NOT TSPC_EFR_EmgCallBcap]			
19		L?DL_DatInESetup (TCV_Esetup1 := DL_DatInESetup.msg, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B)	ESetupRcv(Esetup)		
20		+CallProcGen(TCV_Setup_mo1, TCV_Service)			
		ltree_chkspeech			
21		+EFR_ChkRcvSpchPrefs			check all spchprefs are present in bcap
22		+CallProcGen(TCV_Setup_mo1, TCV_Service)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FnArith(fn, fn1 : FN)

Group : Miscellaneous/

Objective :

Default : OtherEventsFail

Comments : To calculate the difference of two frame numbers 'fn1' and 'fn'. The test step uses TCV_Cnt1, TCV_Cnt2, TCV_K and TCV_Time

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Cnt1 := (51 * ((BIT_TO_INT(fn.t3) − BIT_TO_INT(fn.t2) + 26) MOD 26)) + BIT_TO_INT(fn.t3) + (1326 * BIT_TO_INT(fn.t1_)), TCV_Cnt2 := (51 * ((BIT_TO_INT(fn1.t3) − BIT_TO_INT(fn1.t2) + 26) MOD 26)) + BIT_TO_INT(fn1.t3) + (1326 * BIT_TO_INT(fn1.t1_)), TCV_K := ((TCV_Cnt2−TCV_Cnt1+42432) MOD 42432), TCV_Time := TCV_K * 120 / 26)			GSM 04.08 section 10.5.2.30. TCV_K stores the frame number difference. TCV_Time stores the time difference in ms.

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ImsiDetach (par_mi:MI; ta : TA; sim_rmvd : BOOLEAN)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[(sim_rmvd) AND (TSPC_SIMRmv) AND (TSPC_DetachOnSIMRmv)]			SIM needs to be removed .
2		(TCV_Null :=OO_SIMRmv())			
3		+ltree_imsidetach			
4		[TSPC_SwitchOnOff]			
5		(TCV_Null := OO_SwitchOff())			
6		+ltree_imsidetach			
7		[TSPC_DetachOnPwrDn]			
8		(TCV_Null := OO_PowerDown())			Remove power source
9		+ltree_imsidetach			
10		[(((NOT sim_rmvd) OR (NOT TSPC_SIMRmv) OR (NOT TSPC_DetachOnSIMRmv)) AND (NOT TSPC_SwitchOnOff) AND (NOT TSPC_DetachOnPwrDn))]			
11		(TCV_Null := OO_PowerDown()) ltree_imsidetach			Remove power source
12		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_02)		
13		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
14		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, TSPX_SDCCH4SubDef, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, ta))		
15	E_0069	L?DL_EstInImsidIn	ImsiDet(ImsiDetach_30(par_mi))	(P)	
16		ACTIVATE(OtherEventsFail)			Restore normal default
17		+ChanRel(TCV_ch)			

Detailed Comments : If IMSI detach procedure has been executed, the channel will be relesed. If ATT=0 no detach has been executed, the channel is still connected.

Test Step Dynamic Behaviour

Test Step Name : MM_PwrOrSimOff (sim_rmvd : BOOLEAN)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[(sim_rmvd) AND (TSPC_SIMRmv)]			SIM needs to be removed .
2		(TCV_Null :=OO_SIMRmv())			
3		[TSPC_SwitchOnOff]			
4		(TCV_Null := OO_SwitchOff())			
5		[(((NOT sim_rmvd) OR (NOT TSPC_SIMRmv)) AND (NOT TSPC_SwitchOnOff))]			
6		(TCV_Null := OO_PowerDown())			Remove power source

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : MM_PwrOrSimOn (sim_rmvd : BOOLEAN)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[(sim_rmvd) AND (TSPC_SIMRmv)]			SIM has been removed
2		(TCV_Null := OO_SIMIns())			
3		[TSPC_SwitchOnOff]			
4		(TCV_Null := OO_SwitchOn())			
5		[(((NOT sim_rmvd) OR (NOT TSPC_SIMRmv)) AND (NOT TSPC_SwitchOnOff))]			
6		(TCV_Null := OO_PowerUp())			Restore power source

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : RcvSetupOrEsetup(Setup: SETUP_MO_PDU; Esetup : ESETUP_PDU; Ecall : BOOLEAN)

Group : Miscellaneous/

Objective : To manage Setup Mobile Originated.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[NOT Ecall]			
2		L?DL_DatInSetup (TCV_Setup_mo1 := DL_DatInSetup.msg, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B)	SetupRcv(Setup)		
3		+CallProcGen(TCV_Setup_mo1, TCV_Service)			
4		[Ecall]			
5		L?DL_DatInESetup (TCV_Esetup1 := DL_DatInESetup.msg, TCV_Setup_mo1.ti := TCV_Esetup1.ti, TCV_Setup_mo1.bcap1 := TCV_Esetup1.bcap, TCV_Setup_mo1.ccpd := TCV_Esetup1.ccpd, TCV_Setup_mo1.mt := TCV_Esetup1.mt, TCV_TI := TCV_Setup_mo1.ti, TCV_TI0 := TCV_TI, TCV_TI.ti_f := '1'B)	ESetupRcv(Esetup)		
6		+CallProcGen(TCV_Setup_mo1, TCV_Service)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : RR_hocomp1(time_fn_fn1:INTEGER; oldch:LOGICCH)
Group : Miscellaneous/
Objective : To finish the HO−procedure. Timing advance = 20 bits period
Default : RcvHdOvAcc, OtherEvents
Comments : used var's: TCV_ch, TCV_Horf, TCV_Cntref

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RacInHoacc (TCV_FnBurst := DL_RacInHoacc.fn)	HndOvAccRcv(TCV_ch, HandOverAcc_02(TCV_Horf))		1.
2		L?DL_EstIn	DLEstInd(TCV_ch)		
3		(TCV_HoaccPara := OM_GetHoaccPara(TCV_ch))			
4	E_0148	[(OC_OctToInt(TCV_HoaccPara.num_bursts) >= TCV_Cntref) AND (OC_OctToInt(TCV_HoaccPara.num_bursts) <= (TCV_Cntref+20))]		(P)	2.
5		+localtree_continue			
6	E_0149	[C_Yes]		(F)	
7		+localtree_continue			
		localtree_continue			
8		(TCV_Fn1:= OM_ReturnFn(TCV_ch))			
9		L?DL_DatInHoCom(TCV_Fn := DL_DatInHoCom.fn)	HndOvCmpRcv(TCV_ch, HandOverCmp_20)		
10		L!MDL_RelRq	MDLRelReq(oldch)		Release the original channel.
11		+FnArith(TCV_Fn1, TCV_Fn)			3.
12	E_0070	[TCV_Time > time_fn_fn1]		(F)	
13	E_0071	[TCV_Time <= time_fn_fn1]		(P)	

Detailed Comments : 1. Match the 1st access burst. The access bursts except the 1st one are matched by the default test step RcvHdOvAcc.
2. Check the number of access bursts received with tolerance +20 (10 for FACCH interleave & 10 for MS processing).
3. Check HO−time.

Test Step Dynamic Behaviour

Test Step Name : RRmtcallprepare(ta : TA; rand : RAND)
Group : Miscellaneous/
Objective : To prepare a mobile terminating call establishment.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+PreEstRRConn(TCV_slot, TCV_tsc, ta)			
2		+Authentication(TCV_ch, TCV_cksn, rand)			
3		+Ciphering_on(TCV_ch)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SelectPagingCh(cell : CellID)

Group : Miscellaneous/

Objective : To assign a paging channel and an access grant channel to the variable TCV_PgCh and TCV_agch respectively, depending on the parameter 'cell' (cell ID)

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[cell = C_CellA]			
2		[TCV_Ccchg=0]			
3		(TCV_PgCh := C_PCH_A_1, TCV_agch := C_AGCH_A_1, TCV_RaCh := C_RACH_A_1)			
4		[TCV_Ccchg=1]			
5		(TCV_PgCh := C_PCH_A_2, TCV_agch := C_AGCH_A_2, TCV_RaCh := C_RACH_A_2)			
6		[TCV_Ccchg=2]			
7		(TCV_PgCh := C_PCH_A_3, TCV_agch := C_AGCH_A_3, TCV_RaCh := C_RACH_A_3)			
8		[TCV_Ccchg=3]			
9		(TCV_PgCh := C_PCH_A_4, TCV_agch := C_AGCH_A_4, TCV_RaCh := C_RACH_A_4)			
10		[cell = C_CellB]			
11		[TCV_Ccchg=0]			
12		(TCV_PgCh := C_PCH_B_1, TCV_agch := C_AGCH_B_1, TCV_RaCh := C_RACH_B_1)			
13		[TCV_Ccchg=1]			
14		(TCV_PgCh := C_PCH_B_2, TCV_agch := C_AGCH_B_2, TCV_RaCh := C_RACH_B_2)			
15		[TCV_Ccchg=2]			
16		(TCV_PgCh := C_PCH_B_3, TCV_agch := C_AGCH_B_3, TCV_RaCh := C_RACH_B_3)			
17		[TCV_Ccchg=3]			
18		(TCV_PgCh := C_PCH_B_4, TCV_agch := C_AGCH_B_4, TCV_RaCh := C_RACH_B_4)			
19		[cell = C_CellC]			
20		[TCV_Ccchg=0]			
21		(TCV_PgCh := C_PCH_C_1, TCV_agch := C_AGCH_C_1, TCV_RaCh := C_RACH_C_1)			
22		[TCV_Ccchg=1]			
23		(TCV_PgCh := C_PCH_C_2, TCV_agch := C_AGCH_C_2, TCV_RaCh := C_RACH_C_2)			
24		[TCV_Ccchg=2]			
25		(TCV_PgCh := C_PCH_C_3, TCV_agch := C_AGCH_C_3, TCV_RaCh := C_RACH_C_3)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
26		[TCV_Ccchg=3]			
27		(TCV_PgCh := C_PCH_C_4, TCV_agch := C_AGCH_C_4, TCV_RaCh := C_RACH_C_4)			
Detailed Comments :					

Test Step Dynamic Behaviour					
Test Step Name : SendSeqNo_chk					
Group : Miscellaneous/					
Objective : To check whether the sending sequence number is the same					
Default : OtherEvents					
Comments :					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
1	E_0072	[TCV_Mt1 = TCV_Mt]		(P)	same N(SD)
2	E_0073	[TCV_Mt1 <> TCV_Mt]		(F)	
Detailed Comments :					

Test Step Dynamic Behaviour

Test Step Name : SvcSupportedChk(svc : IA5String)
Group : Miscellaneous/
Objective : To check wether the Basic Service svc is supported against PICS answers.
Default : OtherEvents
Comments : In PICS, if the svc is declared supported, the test case variable TCV_supported is set to TRUE, otherwisw FALSE.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_supported := C_No)			
2		[(svc = C_Telephony) AND TSPC_Serv_TS11]			TS11
3		(TCV_supported := C_Yes)			
4		[(svc = C_EmgCall) AND TSPC_Serv_TS12]			TS12
5		(TCV_supported := C_Yes)			
6		[(svc = C_AltSpchFax) AND TSPC_Serv_TS61]			TS61
7		(TCV_supported := C_Yes)			
8		[(svc = C_AutoFax) AND TSPC_Serv_TS62]			TS62
9		(TCV_supported := C_Yes)			
10		[(svc = C_Async300) AND TSPC_Serv_BS21]			BS21
11		(TCV_supported := C_Yes)			
12		[(svc = C_Async1200) AND TSPC_Serv_BS22]			BS22
13		(TCV_supported := C_Yes)			
14		[(svc = C_Async120075) AND TSPC_Serv_BS23]			BS23
15		(TCV_supported := C_Yes)			
16		[(svc = C_Async2400) AND TSPC_Serv_BS24]			BS24
17		(TCV_supported := C_Yes)			
18		[(svc = C_Async4800) AND TSPC_Serv_BS25]			BS25
19		(TCV_supported := C_Yes)			
20		[(svc = C_Async9600) AND TSPC_Serv_BS26]			BS26
21		(TCV_supported := C_Yes)			
22		[(svc = C_Sync1200) AND TSPC_Serv_BS31]			BS31
23		(TCV_supported := C_Yes)			
24		[(svc = C_Sync2400) AND TSPC_Serv_BS32]			BS32
25		(TCV_supported := C_Yes)			
26		[(svc = C_Sync4800) AND TSPC_Serv_BS33]			BS33
27		(TCV_supported := C_Yes)			
28		[(svc = C_Sync9600) AND TSPC_Serv_BS34]			BS34
29		(TCV_supported := C_Yes)			
30		[(svc = C_PAD300) AND TSPC_Serv_BS41]			BS41
31		(TCV_supported := C_Yes)			
32		[(svc = C_PAD1200) AND TSPC_Serv_BS42]			BS42
33		(TCV_supported := C_Yes)			
34		[(svc = C_PAD120075) AND TSPC_Serv_BS43]			BS43
35		(TCV_supported := C_Yes)			
36		[(svc = C_PAD2400) AND TSPC_Serv_BS44]			BS44
37		(TCV_supported := C_Yes)			
38		[(svc = C_PAD4800) AND TSPC_Serv_BS45]			BS45
39		(TCV_supported := C_Yes)			
40		[(svc = C_PAD9600) AND TSPC_Serv_BS46]			BS46
41		(TCV_supported := C_Yes)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
42		[(svc = C_Packet2400) AND TSPC_Serv_BS51]			BS51
43		(TCV_supported := C_Yes)			
44		[(svc = C_Packet4800) AND TSPC_Serv_BS52]			BS52
45		(TCV_supported := C_Yes)			
46		[(svc = C_Packet9600) AND TSPC_Serv_BS53]			BS53
47		(TCV_supported := C_Yes)			
48		[(svc = C_AltSpchData) AND TSPC_Serv_BS61]			BS61
49		(TCV_supported := C_Yes)			
50		[(svc = C_SpchData) AND TSPC_Serv_BS81]			BS81
51		(TCV_supported := C_Yes)			
52		[C_Yes]			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Varinit_fix(activ_cell : CellID; lac : OCTETSTRING; subch : B_2; cksn : B_3; rand : RAND; arfcngsm, arfcndcs : INTEGER; ccd: CCD; imsi : HEXSTRING)

Group : Miscellaneous/

Objective :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_cellid := activ_cell, TCV_ch := OC_SubchOfSdcch4(subch, activ_cell), TCV_sacch := OC_SubchOfSacch4(subch, activ_cell), TCV_ia_ts := '000'B, TCV_lac := lac, TCV_cksn :=cksn, TCV_CphKey := OC_CphKeyGen(TSPX_Ki, rand))			
2		+CCCH_group_Paging_group(ccd, imsi)			
3		+SelectPagingCh(activ_cell)			
4		[TSPC_PGSM OR TSPC_EGSM]			
5		(TCV_chdescr_arfcn:= arfcngsm)			
6		[TSPC_DCS]			
7		(TCV_chdescr_arfcn:= arfcndcs)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ChgLAC_A(t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; mcc, mnc, lac : OCTETSTRING)

Group : Miscellaneous/

Objective : To change the LAC of cell A and of System Information Messages according to used testcase.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+SysInfoSending_fh(C_SCH_A, C_BCCH_A_1, t, retr, neci, att, babr, cch_con, bpm, t3212, C_ci_cellA, mcc, mnc, lac, CellOpt_01, CellChDes_04, CellChDes_03, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC_1, C_NCCP_2)			
2		(TCV_Ccd0A := TCV_Ccd0H, TCV_sysinfo5 := TCV_sysinf5, TCV_sysinfo6 := TCV_sysinf6)			
3		+SysInfo_SacchSending(TCV_sacch, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ChgLAC_B(t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; mcc, mnc, lac : OCTETSTRING)

Group : Miscellaneous/

Objective : To change the LAC of cell B and of System Information Messages according to used testcase.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+SysInfoSending_fh(C_SCH_B, C_BCCH_B_1, t, retr, neci, att, babr, cch_con, bpm, t3212, C_ci_cellB, mcc, mnc, lac, CellOpt_01, CellChDes_04, CellChDes_03, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, BcchFreqLst_01, BcchFreqLst_48, C_noRestablishment, C_BCC, C_NCC_1, C_NCCP_2)			
2		(TCV_Ccd0B := TCV_Ccd0H, TCV_sysinfo5_B := TCV_sysinf5, TCV_sysinfo6_B := TCV_sysinf6)			
3		+SysInfo_SacchSending(TCV_sacch_B, TCV_sysinfo5_B, TCV_sysinfo6_B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ChgLAI_C(att : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, par_mnc, lac : OCTETSTRING)

Group : Miscellaneous/

Objective : To change the LAI of cell C to HPLMN.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := '0010'B, TCV_Max := '00'B, TCV_Neci := '0'B, TCV_Ccd0C := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		L!DL_UdatRqSysinfo4	SysInfo4(C_BCCH_C_1, mcc, par_mnc, lac, C_CellReselectHys12, C_MaxPwrLvlG, TCV_Neci , TCV_Max, TCV_Tx, C_noRestablishment)		
4		L!DL_UdatRqSysinfo6	SysInfo6(TCV_sacch_C, SysInf6(C_ci_cellC, mcc, par_mnc, lac, CellOpt_01, C_NCCP_2))		
5		L!DL_UdatRqSysinfo3	SysInfo3(C_BCCH_C_1, C_ci_cellC, mcc, par_mnc, lac, TCV_Ccd0C, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlG, TCV_Neci, TCV_Max, TCV_Tx, C_noRestablishment)		
6		[TSPC_DCS]			
7		L!DL_UdatRqSysinfo4	SysInfo4(C_BCCH_C_1, mcc, par_mnc, lac, C_CellReselectHys12, C_MaxPwrLvlD, TCV_Neci , TCV_Max, TCV_Tx, C_noRestablishment)		
8		L!DL_UdatRqSysinfo6	SysInfo6(TCV_sacch_C, SysInf6(C_ci_cellC, mcc, par_mnc, lac, CellOpt_01, C_NCCP_2))		
9		L!DL_UdatRqSysinfo3	SysInfo3(C_BCCH_C_1, C_ci_cellC, mcc, par_mnc, lac, TCV_Ccd0C, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlD, TCV_Neci, TCV_Max, TCV_Tx, C_noRestablishment)		

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_cbms(sch, bcch : LOGICCH; t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; crh, mtmcgsm, mtmcdcs : INTEGER; bcchflgsm, bcchfldcs : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP; Sub : B_2)

Group : Miscellaneous/

Objective : To send system information messages for the CBMS tests.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := OC_CnvtTx(t), TCV_Max := OC_CnvtMax(retr), TCV_Neci := INT_TO_BIT(neci,1), TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212), TCV_Tchtype := INT_TO_BIT((4 + BIT_TO_INT(Sub)), 5))			
2		(TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp))			
3		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		
4		+ltree_sysinfo			
		ltree_sysinfo			
5		[TSPC_PGSM OR TSPC_EGSM]			
6		L!DL_UdatRqSysinfo4	SysInfo4_CBMS(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re, ChDescrp_nfh(TCV_Tchtype, C_S0, C_BCC, C_arfcnA))		
7		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
8		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
9		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
10		(TCV_sysinf5 := SysInf5(bcchflgsm))			
11		[TSPC_DCS]			
12		L!DL_UdatRqSysinfo4	SysInfo4_CBMS(bcch, mcc, mnc, lac, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re, ChDescrp_nfh(TCV_Tchtype, C_S0, C_BCC, C_arfcnAd))		
13		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchfldcs, TCV_Max, TCV_Tx, Re, nccp)		
14		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
15		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		
16		(TCV_sysinf5 := SysInf5(bcchfldcs))			
Detailed Comments :					

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_fh(sch, bcch : LOGICCH; t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; cchdgsm, cchddcs : CCHD; crh, mtmcgsm, mtmcdcs : INTEGER; bcchflgsm, bcchfldcs : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/

Objective : To send system information messages for the L3 tests. The following parameters specified by input parameters:
- bcch: BCCH
- sch: SCH
- bcchfl: BCCH frequency list
- ci: Cell identity
- lai: Location area identification
- ccd: Control channel description
- csp: Cell selection parameters
- cchd: Cell channel description
- rachcpar: RACH controll paramenters
- nccp: NCC permitted

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := OC_CnvtTx(t), TCV_Max := OC_CnvtMax(retr), TCV_Neci := INT_TO_BIT(neci,1), TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		(TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp))			
3		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		
4		+ltree_sysinfo			
		ltree_sysinfo			
5		[TSPC_PGSM OR TSPC_EGSM]			
6		L!DL_UdatRqSysinfo1	SysInfo1(bcch, cchdgsm, TCV_Max, TCV_Tx, Re)		
7		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
8		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
9		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
10		(TCV_sysinf5 := SysInf5(bcchflgsm))			
11		[TSPC_DCS]			
12		L!DL_UdatRqSysinfo1	SysInfo1(bcch, cchddcs, TCV_Max, TCV_Tx, Re)		
13		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchfldcs, TCV_Max, TCV_Tx, Re, nccp)		
14		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
15		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		
16		(TCV_sysinf5 := SysInf5(bcchfldcs), TCV_sysinf6.co.pwrc := '1'B)			
Detailed Comments :					

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_nfh(sch, bcch : LOGICCH; t, retr, neci , att: INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; crh, mtmcgsm, mtmcdcs : INTEGER; bcchflgsm, bcflaltgsm, bcchfldcs, bcflaltdcs : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/

Objective : To send system information messages with default parameters defined for the L3 tests for which no special parameters indicated.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := OC_CnvtTx(t), TCV_Max := OC_CnvtMax(retr), TCV_Neci := INT_TO_BIT(neci,1), TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		(TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp))			
3		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		
4		+gsmOrDcs			
		gsmOrDcs			
5		[TSPC_PGSM OR TSPC_EGSM]			1.
6		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci , TCV_Max, TCV_Tx, Re)		
7		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
8		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
9		[NOT TSPX_AltNb]			3.
10		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
11		(TCV_sysinf5 := SysInf5(bcchflgsm))			
12		[TSPX_AltNb]			4.
13		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcflaltgsm, TCV_Max, TCV_Tx, Re, nccp)		
14		(TCV_sysinf5 := SysInf5(bcflaltgsm))			
15		[TSPC_DCS]			2.
16		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcdcs, TCV_Neci , TCV_Max, TCV_Tx, Re)		
17		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		
18		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		
19		[NOT TSPX_AltNb]			3.

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
20		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchfldcs, TCV_Max, TCV_Tx, Re, nccp)		
21		(TCV_sysinf5 := SysInf5(bcchfldcs))			
22		[TSPX_AltNb]			4.
23		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcflaltdcs, TCV_Max, TCV_Tx, Re, nccp)		
24		(TCV_sysinf5 := SysInf5(bcflaltdcs))			
Detailed Comments : 1. For GSM900 mobile station testing. 2. For DCS1800 mobile station testing. 3. To use alternative neighbour cells description. 4. To use default neighbour cells description.					

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_nfhCb(sch, bcch : LOGICCH; t, retr, neci , att: INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; crh, mtmcgsm, mtmcdcs : INTEGER; bcchflgsm, bcflaltgsm, bcchfldcs, bcflaltdcs : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/

Objective : To send system information messages with default parameters defined for the L3 tests for which no special parameters indicated.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := OC_CnvtTx(t), TCV_Max := OC_CnvtMax(retr), TCV_Neci := INT_TO_BIT(neci,1), TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		(TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp))			
3		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		
4		+gsmOrDcs			
		gsmOrDcs			
5		[TSPC_PGSM OR TSPC_EGSM]			1.
6		L!DL_UdatRqSysinfo4 (DL_UdatRqSysinfo4.msg.rachcp.cba := '1'B)	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci , TCV_Max, TCV_Tx, Re)		
7		L!DL_UdatRqSysinfo1_nh (DL_UdatRqSysinfo1_nh.msg.rachcp.cba := '1'B)	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
8		L!DL_UdatRqSysinfo3 (DL_UdatRqSysinfo3.msg.rachcp.cba := '1'B)	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
9		[NOT TSPX_AltNb]			3.
10		L!DL_UdatRqSysinfo2 (DL_UdatRqSysinfo2.msg.rachcp.cba := '1'B)	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
11		(TCV_sysinf5 := SysInf5(bcchflgsm))			
12		[TSPX_AltNb]			4.
13		L!DL_UdatRqSysinfo2 (DL_UdatRqSysinfo2.msg.rachcp.cba := '1'B)	SysInfo2(bcch, bcflaltgsm, TCV_Max, TCV_Tx, Re, nccp)		
14		(TCV_sysinf5 := SysInf5(bcflaltgsm))			
15		[TSPC_DCS]			2.
16		L!DL_UdatRqSysinfo4 (DL_UdatRqSysinfo4.msg.rachcp.cba := '1'B)	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcdcs, TCV_Neci , TCV_Max, TCV_Tx, Re)		
17		L!DL_UdatRqSysinfo1_nh (DL_UdatRqSysinfo1_nh.msg.rachcp.cba := '1'B)	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		
18		L!DL_UdatRqSysinfo3 (DL_UdatRqSysinfo3.msg.rachcp.cba := '1'B)	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcdcs, TCV_Neci, TCV_Max, TCV_Tx, Re)		
19		[NOT TSPX_AltNb]			3.

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
20		L!DL_UdatRqSysinfo2 (DL_UdatRqSysinfo2.msg.rachcp.cba := '1'B)	SysInfo2(bcch, bcchfldcs, TCV_Max, TCV_Tx, Re, nccp)		
21		(TCV_sysinf5 := SysInf5(bcchfldcs))			
22		[TSPX_AltNb]			4.
23		L!DL_UdatRqSysinfo2 (DL_UdatRqSysinfo2.msg.rachcp.cba := '1'B)	SysInfo2(bcch, bcflaltdcs, TCV_Max, TCV_Tx, Re, nccp)		
24		(TCV_sysinf5 := SysInf5(bcflaltdcs))			

Detailed Comments :
1. For GSM900 mobile station testing.
2. For DCS1800 mobile station testing.
3. To use alternative neighbour cells description.
4. To use default neighbour cells description.

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_e(sch, bcch : LOGICCH; t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; cchdgsm : CCHD; crh, mtmcgsm : INTEGER; bcchflgsm, bcchflegsm : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/

Objective : To send SYNCHRONIZATION CHANNEL INFORMATION and SYSTEM INFORMATION messages with default parameters except CCCH_CONF, Tx−integer, Max−Retrans and ATT which are specified by input parameters.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := OC_CnvtTx(t), TCV_Max := OC_CnvtMax(retr), TCV_Neci := INT_TO_BIT(neci,1), TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		1.
3		(TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp))			
4		L!DL_UdatRqSysinfo1	SysInfo1(bcch, cchdgsm, TCV_Max, TCV_Tx, Re)		2.
5		L!DL_UdatRqSysinfo2 (DL_UdatRqSysinfo2.msg.bcchfl.extind := '1'B)	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
6		L!DL_UdatRqSysinfo2bis	SysInfo2bis(bcch, bcchflegsm)		
7		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci , TCV_Max, TCV_Tx, Re)		
8		(TCV_sysinf5 := SysInf5(bcchflgsm), TCV_sysinfo5.bcchfl.extind := '1'B)			
9		(TCV_sysinfo5bis := SysInf5bis(bcchflegsm))			
10		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		

Detailed Comments : 1. To send SYNCHRONIZATION INFORMATION message with default parameter for cell A.
2. System Information Type 1 with Cell Channel Description for HO−testin in cell A (GSM900).

Test Step Dynamic Behaviour

Test Step Name : SysInfo_SacchSending(ch : LOGICCH; sysinfo5_pdu : SYSINFO5_PDU; sysinfo6_pdu : SYSINFO6_PDU)

Group : Miscellaneous/

Objective : To send SYSTEM INFORMATION 5 and 6 messages defined by parameters 'sysinfo5_pdu' and 'sysinfo6_pdu' in the parametrised 'ch' channel.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[ch <> "dummy"]			
2		L!DL_UdatRqSysinfo6	SysInfo6(ch , sysinfo6_pdu)		
3		L!DL_UdatRqSysinfo5	SysInfo5(ch , sysinfo5_pdu)		
4	E_0074	[ch = "dummy"]		I	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SysInfo_5bisSending(ch : LOGICCH; sysinfo5bis_pdu : SYSINFO5bis_PDU)

Group : Miscellaneous/

Objective : To send SYSTEM INFORMATION 5bis message defined by parameters 'sysinfo5bis_pdu' in the parametrised 'ch' channel.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[ch <> "dummy"]			
2		L!DL_UdatRqSysinfo5bis	SysInfo5bis(ch , sysinfo5bis_pdu)		
3	E_0075	[ch = "dummy"]		I	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SetNECI(att : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, mnc, lac : OCTETSTRING)
Group : Miscellaneous/
Objective : To set the NECI =1.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := '0010'B, TCV_Max := '11'B, TCV_Neci := '1'B, TCV_Ccd0A := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		[TSPC_PGSM OR TSPC_EGSM]			
3		L!DL_UdatRqSysinfo4	SysInfo4(C_BCCH_A_1, mcc, mnc, lac, C_CellReselectHys12, C_MaxPwrLvlG, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
4		L!DL_UdatRqSysinfo3	SysInfo3(C_BCCH_A_1, C_ci_cellA, mcc, mnc, lac, TCV_Ccd0A, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlG, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
5		[TSPC_DCS]			
6		L!DL_UdatRqSysinfo4	SysInfo4(C_BCCH_A_1, mcc, mnc, lac, C_CellReselectHys12, C_MaxPwrLvlD, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
7		L!DL_UdatRqSysinfo3	SysInfo3(C_BCCH_A_1, C_ci_cellA, mcc, mnc, lac, TCV_Ccd0A, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlD, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SetATT(t, retr, att, neci : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, mnc, lac : OCTETSTRING)
Group : Miscellaneous/
Objective : To set the ATT flag to "MS's in the cell should apply IMSI attach and detach procedure"
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Tx := OC_CnvtTx(t), TCV_Max := OC_CnvtMax(retr), TCV_Neci := INT_TO_BIT(neci,1), TCV_Ccd0A := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
2		[TSPC_PGSM OR TSPC_EGSM]			1.
3		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(C_BCCH_A_1, C_ci_cellA, mcc, mnc, lac, TCV_Ccd0A, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlG, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
4		L!DL_UdatRqSysinfo3	SysInfo3(C_BCCH_A_1, C_ci_cellA, mcc, mnc, lac, TCV_Ccd0A, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlG, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
5		START T_dly(5000)			
6		?TIMEOUT T_dly			3.
7		[TSPC_DCS]			2.
8		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(C_BCCH_A_1, C_ci_cellA, mcc, mnc, lac, TCV_Ccd0A, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlD, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
9		L!DL_UdatRqSysinfo3	SysInfo3(C_BCCH_A_1, C_ci_cellA, mcc, mnc, lac, TCV_Ccd0A, CellOpt_01, C_CellReselectHys12, C_MaxPwrLvlD, TCV_Neci , TCV_Max, TCV_Tx, C_Restablishment)		
10		START T_dly(5000)			3.
11		?TIMEOUT T_dly			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.
3. Wait for 5 seconds to allow the MS read BCCH information.

Test Step Dynamic Behaviour

Test Step Name : InitCall(srv: SERVICES)
Group : OperatorOP/
Objective : To initiate a call of the basic service ‘srv‘.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OO_InitCall(srv))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : InitNonCallSupp
Group : OperatorOP/
Objective : To attempt a non call related supplementary service at the MS under test.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OO_InitNonCallSS())			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : AtmpShortMsg
Group : OperatorOP/
Objective : To attempt a short message service transaction at the MS under test.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Null := OO_SendMOShortMessage())			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CheckUssdStringDisplayed(strg: IA5String)
Group : OperatorOP/
Objective : To check whether the correct USSD String 'strg' is displayed on the MS
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Res := OO_CheckUssdStringDisplayed(strg))			
2	E_0076	[TCV_Res]		(P)	
3	E_0077	[NOT TCV_Res]		(F)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : PLMNsCHK
Group : OperatorOP/
Objective : To check whether the MS presents a list of available PLMNs.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Res := OO_PLMNsCHK())			
2	E_0078	[TCV_Res]		P	
3	E_0079	[NOT TCV_Res]		F	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : RFtransCHK
Group : OperatorOP/
Objective : To check whether the MS transmits any radio signal.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Res := OO_RFoutputCHK())			
2	E_0080	[TCV_Res]		F	
3	E_0081	[NOT TCV_Res]		(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : PostLinkRelEnd(ch: LOGICCH)
Group : Postambles/
Objective : To release the RR connection and bring the MS back to Idle state.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqChRel START T_release	ChRel(ch, ChRelease_01)		
2		L?DL_RelIn	DLRelInd_01		
3	E_0082	CANCEL		R	1.
4		?TIMEOUT T_release			
5	E_0083	CANCEL		R	1.

Detailed Comments : 1. Cancel of all running timers and final verdict.

Test Step Dynamic Behaviour

Test Step Name : PostMainLinkRel(chnl : LOGICCH)
Group : Postambles/
Objective : To release the main signalling link ‘ch‘, and bring the MS back to Idle state.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqChRel START T_release	ChRel(chnl, ChRelease_01)		
2		L?DL_RelIn CANCEL T_release	DLRelInd_01		
3		?TIMEOUT T_release			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ChanRel(ch: LOGICCH)
Group : Postambles/
Objective : To release the RR connection on the channel TCV_ch and bring the MS back to Idle state.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqChRel START T_release	ChRel(ch, ChRelease_01)		
2		L?DL_RelIn CANCEL T_release	DLRelInd_01		
3	E_0084	?TIMEOUT T_release		(F)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ChanRel_P(ch: LOGICCH)
Group : Postambles/
Objective : To release the RR connection on the channel TCV_chmaindcch and bring the MS back to Idle state.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqChRel START T_release	ChRel(ch, ChRelease_01)		
2	E_0085	L?DL_RelIn	DLRelInd_01	(P)	
3		?TIMEOUT T_release			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : ChanRel_end(ch: LOGICCH)
Group : Postambles/
Objective : To release the RR connection and bring the MS back to Idle state.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqChRel START T_release	ChRel(ch, ChRelease_01)		
2		L?DL_RelIn	DLRelInd_01		
3	E_0086	CANCEL		(P)	1.
4		?TIMEOUT T_release			
5	E_0087	CANCEL		(F)	1.

Detailed Comments : 1. Cancel of all running timers.

Test Step Dynamic Behaviour

Test Step Name : Chmod(rate : RATE; ce : B_2; ur : B_4; svc : SERVICES)

Group : Preambles/

Objective : To assign correct values to the test case variables TCV_ChMod and TCV_ChModb.

Default : OtherEvents

Comments : The rules for assignment are in Table 10−1 of GSM 11.10 section 10.3. If the value for ce is "BothT", the rule for transparent is used. If the value for ce is "BothNT", the rule for non transparent is used. That means the preferred value will be used in test configuration.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_tree			
2		[(svc = C_AltSpchData) OR (svc = C_SpchData) OR (svc = C_AltSpchFax)]			dual mode call
3		(TCV_ChMod.mode := C_ChMod_r)			
4		[(svc<> C_AltSpchData) AND (svc <> C_SpchData) AND (svc <> C_AltSpchFax)]			
		local_tree			
5		[((ce = C_notransparent) OR (ce = C_BothNT)) AND (rate = C_Full)]			
6		(TCV_ChMod.mode := C_ChMod_12k, TCV_ChModb.mode := C_ChMod_12k)			
7		[((ce = C_notransparent) OR (ce = C_BothNT)) AND (rate = C_Half)]			
8		[ur <> C_9600bs]			
9		(TCV_ChMod.mode := C_ChMod_6k, TCV_ChModb.mode := C_ChMod_6k)			
10	E_0088	[ur = C_9600bs]		I	not applicable
11		[(ce = C_transparent) OR (ce = C_BothT)]			
12		[(ur = C_9600bs) AND (rate = C_Full)]			
13		(TCV_ChMod.mode := C_ChMod_12k, TCV_ChModb.mode := C_ChMod_12k)			
14		[ur = C_4800bs]			
15		(TCV_ChMod.mode := C_ChMod_6k, TCV_ChModb.mode := C_ChMod_6k)			
16		[(ur <> C_4800bs) AND (ur <> C_9600bs)]			
17		(TCV_ChMod.mode := C_ChMod_3k, TCV_ChModb.mode := C_ChMod_3k)			
18	E_0089	[(ur = C_9600bs) AND (rate = C_Half)]		I	not applicable

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : EFR_BasicServiceMO(svc : SERVICES; rate : RATE)

Group : Preambles/

Objective : To generate a EFR Bcap and a MO SETUP/ESETUP message (to match later against received MO SETUP/ESETUP message), and an appropriate channel mode based on the service and rate, as well as on several IXIT parameters, itc, ce, ur, synchronous bit for an MO call.

Bcap shall match the correct NUMBER of speech versions that the EFR mobile supports.

Default : OtherEvents

Comments : Used for CC tests

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_init			Default values
2		[TCV_Service = C_EmgCall]			
3		(TCV_Ecall := C_Yes)			Emergency Call
4		[(TCV_ChRate = C_Full) OR (TCV_ChRate = C_Half)]			
5		(TCV_ChMod.mode := C_ChMod_r)			Speech v1
6		[NOT TSPC_EFR_EmgCallBcap]			
7		(TCV_Esetup := ESetupNoBcap)			Speech v1 − no bcap
8		[TSPX_EFR_SpchPrefsNo = 2]			
9		(TCV_Esetup := ESetupBcap(EFR_Bcap_Speech_MO2(TCV_Rchr)))			Speech v1 − 2 speech prefs
10		[TSPX_EFR_SpchPrefsNo = 3]			
11		(TCV_Esetup := ESetupBcap(EFR_Bcap_Speech_MO3(TCV_Rchr)))			Speech v1 − 3 speech prefs
12	E_0090	[C_Yes]		I	IXIT error
13		[TCV_ChRate = C_Full_v2]			
14		(TCV_ChMod.mode := C_ChMod_r2,TCV_ChRate := C_Full)			Speech v2
15		[NOT TSPC_EFR_EmgCallBcap]			
16		(TCV_Esetup := ESetupNoBcap)			Speech v2 − no bcap
17		[TSPX_EFR_SpchPrefsNo = 2]			
18		(TCV_Esetup := ESetupBcap(EFR_Bcap_Speech_MO2(TCV_Rchr)))			Speech v2 − 2 speech prefs
19		[TSPX_EFR_SpchPrefsNo = 3]			
20		(TCV_Esetup := ESetupBcap(EFR_Bcap_Speech_MO3(TCV_Rchr)))			Speech v2 − 3 speech prefs
21	E_0091	[C_Yes]		I	IXIT error
22		[TCV_Service = C_Telephony]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
23		[(TCV_ChRate = C_Full) OR (TCV_ChRate = C_Half)]			Telephony
24		(TCV_ChMod.mode := C_ChMod_r)			Speech v1
25		[TSPX_EFR_SpchPrefsNo = 2]			
26		(TCV_Setup_mo := Setup_02(EFR_Bcap_Speech_MO2(TCV_Rchr)))			Speech v1 − 2 speech prefs
27		[TSPX_EFR_SpchPrefsNo = 3]			
28		(TCV_Setup_mo := Setup_02(EFR_Bcap_Speech_MO3(TCV_Rchr)))			Speech v1 − 3 speech prefs
29	E_0092	[C_Yes]		I	IXIT error
30		[TCV_ChRate = C_Full_v2]			
31		(TCV_ChMod.mode := C_ChMod_r2, TCV_ChRate := C_Full)			Speech v2
32		[TSPX_EFR_SpchPrefsNo = 2]			
33		(TCV_Setup_mo := Setup_02(EFR_Bcap_Speech_MO2(TCV_Rchr)))			Speech v2 − 2 speech prefs
34		[TSPX_EFR_SpchPrefsNo = 3]			
35		(TCV_Setup_mo := Setup_02(EFR_Bcap_Speech_MO3(TCV_Rchr)))			Speech v2 − 3 speech prefs
36	E_0093	[C_Yes]		I	IXIT error
37	E_0094	[C_Yes]		I	IXIT error
		local_init			
38		(TCV_Service := svc, TCV_Rchr := C_Rchr_Full, TCV_Ecall := C_No, TCV_ChRate :=rate, TCV_ChMod.iei := '01100011'B, TCV_ChMod.mode := C_ChMod_12k, TCV_ChModb.iei := '01100011'B, TCV_ChModb.mode := C_ChMod_12k, TCV_MODEM := V22bisOrV26ter)			
39		[TSPC_DualRate]			
40		(TCV_Rchr := C_Rchr_DualForDualH)			
41		[NOT TSPC_DualRate]			

Detailed Comments : This test step is called in the preamble of test cases where an MO call is established in the test body.

Test Step Dynamic Behaviour

Test Step Name : EFR_BasicServiceMT(svc : SERVICES; rate : RATE)

Group : Preambles/

Objective : To generate a EFR Bcap (to match against received CALL CONNECTED Bcap), an MT SETUP or ESETUP message (to be sent to EFR mobile) and an appropriate channel mode based on the service and rate, as well as on several IXIT parameters, itc, ce, ur, synchronous bit for an MT call.

EFR Bcap shall match the correct NUMBER of speech versions that the EFR mobile supports.

Default : OtherEvents

Comments : Used for CC tests

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_Service := svc, TCV_ChRate :=rate, TCV_ChMod.iei := '01100011'B, TCV_ChMod.mode := C_ChMod_12k, TCV_ChModb.iei := '01100011'B, TCV_ChModb.mode := C_ChMod_12k, TCV_Rchr := C_Rchr_Full, TCV_2ndtest := C_No)			Default values
2		[TSPC_DualRate]			
3		(TCV_Rchr := C_Rchr_DualForDualH)			
4		+SvcSupportedChk(svc)			
5		+local_main			
6		[NOT TSPC_DualRate]			
7		+SvcSupportedChk(svc)			
8		+local_main			
		local_main			
9		[TCV_Service = C_Telephony]			
10		[(TCV_ChRate = C_Full) OR (TCV_ChRate = C_Half)]			Speech v1
11		(TCV_ChMod.mode := C_ChMod_r, TCV_Setup_mt := Setup_01(TI_02))			
12		[TSPX_EFR_SpchPrefsNo = 2]			
13		(TCV_BcapMO1 := EFR_Bcap_Speech_MO2(TCV_Rchr))			EFR BCAP − 2 speech prefs
14		+local_IFsetup_Telephony			Config speech
15		[TSPX_EFR_SpchPrefsNo = 3]			
16		(TCV_BcapMO1 := EFR_Bcap_Speech_MO3(TCV_Rchr))			EFR BCAP − 3 speech prefs
17		+local_IFsetup_Telephony			Config speech
18	E_0095	[C_Yes]		I	IXIT error
19		[TCV_ChRate = C_Full_v2]			Speech v2
20		(TCV_ChRate := C_Full,TCV_ChMod.mode := C_ChMod_r2,TCV_Setup_mt := Setup_01(TI_02))			
21		[TSPX_EFR_SpchPrefsNo = 2]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
22		(TCV_BcapMO1 := EFR_Bcap_Speech_MO2(TCV_Rchr))			EFR BCAP − 2 speech prefs
23		+local_IFsetup_Telephony			Config speech
24		[TSPX_EFR_SpchPrefsNo = 3]			
25		(TCV_BcapMO1 := EFR_Bcap_Speech_MO3(TCV_Rchr))			EFR BCAP − 3 speech prefs
26		+local_IFsetup_Telephony			Config speech
27	E_0096	[C_Yes]		I	IXIT error
		local_IFsetup_Telephony			
28		[TCV_supported]			
29		(TCV_Null := OO_IFsetup_Telephony())			
30		[NOT TCV_supported]			
31		[TCV_SpecialCase]			
32		START T_dly(C_T_Wait)			
33		?TIMEOUT T_dly			
34	E_0097	[NOT TCV_SpecialCase]		I	parameter error
Detailed Comments : This test step is called in the preamble of test cases where an MT call is established in the test body.					

Test Step Dynamic Behaviour

Test Step Name : Est_MO_Call_init(par1: OCTETSTRING; mag, mad : MA; ta : TA; tsc1 : B_3; nfrqg, nfrqd : INTEGER)

Group : Preambles/

Objective : To initiate a mobile originating call for the supported bearer capability. The channel in use is frequency hopping channel.

Default : OtherEvents

Comments : The test case variable TCV_Mt holds the message type of the received SETUP msg.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_cellid:=C_CellA)			
2		+InitCall(TCV_Service)			
3		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_15)		
4		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
5		+localtree_ia(par1)			
6		L?DL_EstInCmsRq	CMSerReq(CMServiceReq_01)		
7		ACTIVATE(OtherEvents)			Restore Normal default
8		+ltree_continue			
		ltree_continue			
9		(TCV_Null := OM_NoL2Ack(2,TCV_ch))			Don't acknowledge setup message
10		+Ciphering_on(TCV_ch)			
11		L?DL_DatInSetup (TCV_Mt := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
		localtree_ia(par_chn: OCTETSTRING)			
12		[par_chn = C_CHTCHF_NonFH]			
13		(TCV_Tchtype := '00001'B)			
14		+ltree_immass_nfh(TCV_tch_arfcn, tsc1)			
15		[par_chn = C_CHSDCCH4_NonFH]			
16		(TCV_Tchtype := INT_TO_BIT((4 + BIT_TO_INT(TSPX_SDCCH4SubDef)), 5))			
17		+ltree_immass_nfh(TCV_chdescr_arfcn, tsc1)			
18		[par_chn = C_CHSDCCH8_NonFH]			
19		(TCV_Tchtype := INT_TO_BIT((8 + BIT_TO_INT(TSPX_SDCCH8SubDef)), 5))			
20		+ltree_immass_nfh(TCV_tch_arfcn, tsc1)			
21		[par_chn = C_CHTCHF_FH]			
22		(TCV_Tchtype := '00001'B)			
23		+ltree_immass_fh(nfrqg, nfrqd)			
24		[par_chn = C_CHTCHH_FH]			
25		(TCV_Tchtype := INT_TO_BIT((2 + BIT_TO_INT(TSPX_TCHHSubDef)), 5))			
26		+ltree_immass_fh(nfrqg, nfrqd)			
27		[par_chn = C_CHTCHH_NonFH]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
28		(TCV_Tchtype := INT_TO_BIT((2 + BIT_TO_INT(TSPX_TCHHSubDef)), 5))			
29		+ltree_immass_nfh(TCV_tch_arfcn, tsc1)			
30		[par_chn = C_CHSDCCH8_FH]			
31		(TCV_Tchtype := INT_TO_BIT((8 + BIT_TO_INT(TSPX_SDCCH8SubDef)), 5))			
32		+ltree_immass_fh(nfrqg, nfrqd)			
		ltree_immass_fh(frqnog, frqnod : INTEGER)			
33		[TSPC_PGSM OR TSPC_EGSM]			
34		(TCV_maio := INT_TO_BIT((TSPX_MAIO MOD frqnog), 6), TCV_iel := mag.iel)			1.
35		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_221(TCV_Rr, TCV_Fn, TCV_ia_ts, tsc1, TCV_Tchtype, TCV_maio, INT_TO_BIT(TSPX_HSN, 6), ta, mag, TCV_iel))		
36		[TSPC_DCS]			
37		(TCV_maio := INT_TO_BIT((TSPX_MAIO MOD frqnod), 6), TCV_iel := mad.iel)			1.
38		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_221(TCV_Rr, TCV_Fn, TCV_ia_ts, tsc1, TCV_Tchtype, TCV_maio, INT_TO_BIT(TSPX_HSN, 6), ta, mad, TCV_iel))		
		ltree_immass_nfh(arfcn : INTEGER; tsc : B_3)			
39		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_nfh(TCV_Rr, TCV_Fn, TCV_Tchtype, TCV_ia_ts, tsc, ta, arfcn, C_normal_paging))		

Detailed Comments : 1. To make the MAIO is within 0 ... N−1, where N is the number of frequencies of MA. "frqnog" and
"frqnod" are number of of frequencies of MA for GSM and DCS respectively.

Test Step Dynamic Behaviour

Test Step Name : Est_MT_Call_FH(ta : TA; rate : IA5String; maio : MAIO; hsn : INTEGER; rand : RAND; slot : SN; tsc : TSC; tch : B_1)

Group : Preambles/

Objective : To establish a mobile station terminating call with hopping channel (speech or data call).

Default : OtherEvents

Comments : used var's:
TCV_cellid, TCV_ch

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+RRmtcallprepare(ta, rand)			
2		+CC_Est_MT_Call			
3		+localtree			
4		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
5		L!DL_DatRqConnAck	ConnAckSnd(TCV_chTch, ConnectAck_01(TI_02))		
		localtree			
6		[rate = C_Full]			Full rate
7		[TCV_cellid = C_CellA]			
8		+AssCmdGen_fh(C_Full, slot, tsc, tch, 9, 15, maio, INT_TO_BIT(hsn, 6), Frql_20_A, Frql_20_Ad, CellChDes_Omit, CellChDes_Omit, ChMod_speech_iei, MoblAllc_omit, MoblAllc_omit, CphMod_omit)			
9		[TCV_cellid = C_CellB]			
10		+AssCmdGen_fh(C_Full, slot, tsc, tch, 9, 15, maio, INT_TO_BIT(hsn, 6), Frql_20_B3, Frql_20_B3d, CellChDes_Omit, CellChDes_Omit, ChMod_speech_iei, MoblAllc_omit, MoblAllc_omit, CphMod_omit)			
11		[rate = C_Full_v2]			Full rate
12		[TCV_cellid = C_CellA]			
13		+AssCmdGen_fh(C_Full_v2, slot, tsc, tch, 9, 15, maio, INT_TO_BIT(hsn, 6), Frql_20_A, Frql_20_Ad, CellChDes_Omit, CellChDes_Omit, ChMod_EFR2_iei, MoblAllc_omit, MoblAllc_omit, CphMod_omit)			
14		[TCV_cellid = C_CellB]			
15		+AssCmdGen_fh(C_Full_v2, slot, tsc, tch, 9, 15, maio, INT_TO_BIT(hsn, 6), Frql_20_B3, Frql_20_B3d, CellChDes_Omit, CellChDes_Omit, ChMod_EFR2_iei, MoblAllc_omit, MoblAllc_omit, CphMod_omit)			
16		[rate = C_Half]			Half rate
17		[TCV_cellid = C_CellA]			
18		+AssCmdGen_fh(C_Half, slot, tsc, tch, 9, 15, maio, INT_TO_BIT(hsn, 6), Frql_20_A0, Frql_20_A0d, CellChDes_Omit, CellChDes_Omit, ChMod_omit, MoblAllc_omit, MoblAllc_omit, CphMod_omit)			
19		[TCV_cellid = C_CellB]			
20		+AssCmdGen_fh(C_Half, slot, tsc, tch, 9, 15, maio, INT_TO_BIT(hsn, 6), Frql_20_B3, Frql_20_B3d, CellChDes_Omit, CellChDes_Omit, ChMod_omit, MoblAllc_omit, MoblAllc_omit, CphMod_omit)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Est_MT_CallNonFH(ta : TA; rate : IA5String; sub : B_1; rand : RAND)

Group : Preambles/

Objective : To establish a mobile station terminating call with non hopping channel (speech or data call).

Default : OtherEvents

Comments : used var's:
TCV_cellid, TCV_chdescr_arfcn, TCV_Bcap1, TCV_chTch, TCV_asscmd_ts

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+RRmtcallprepare(ta, rand)			
2		+CC_EstMsTermCall(rate, sub)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : IdleUpdated(par_bspwr : INTEGER; activ_cell : CellID; sch, bcch : LOGICCH; acttype : BITSTRING; slot, slot2, slot3, slot4 : SN; tsc : TSC; t, retr, neci, att : INTEGER; ta : TA; babr, cch_con, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; cchdgsm, cchddcs : CCHD; crh, mtmcgsm, mtmcdcs : INTEGER; bcchflgsm, bcflaltgsm, bcchfldcs, bcflaltdcs : NCD; Re : B_1; bcc : BCC; ncc : NCC; StartType : INTEGER; subch : B_2; cksn : B_3; rand : RAND; arfcngsm, arfcndcs : INTEGER; imsi : HEXSTRING; sysinfonb : INTEGER; nccp : NCCP)

Group : Preambles/

Objective : To ensure that the SIM is either updated to the initial conditions and the MS with CKSN valid, TMSI valid and idle updated in cell A, B or C; or idle updated without TMSI; or totally not updated and all LAI information deleted.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_slot := slot, TCV_tsc := tsc, TCV_Ccd0H := CntrlChDscrp(att, babr, C_cch_1Comb, bpm, t3212))			
2		+MM_PwrOrSimOff (C_SIMneedRmv)			
3		+Varinit_fix(activ_cell, lac, subch, cksn, rand, arfcngsm, arfcndcs, TCV_Ccd0H, imsi)			
4		+ltree_chconf(C_cch_1Comb)			Config LT
5		+localtree1			
		localtree1			
6		+ltree_sysinfo(1, C_cch_1Comb)			ATT= '1'B
7		+MM_PwrOrSimOn (C_SIMneedRmv)			
8		+localtree_Lup_Auth			
		localtree_Lup_Auth			
9		START T_dly(C_T_Wait1stChReq)			
10		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn) CANCEL T_dly	ChReq(ChRequest_02)		Any channel request PDU
11		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
12		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, subch, slot, tsc, TCV_chdescr_arfcn, ta))		
13		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(C_norm_period_attach))		Any location update type
14		ACTIVATE(OtherEvents)			Restore Normal default
15		[StartType = C_Start_Rej]			
16		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(C_rc_LAnotallowed))		Delete LAI, tmsi, cksn
17		+PostMainLinkRel(TCV_ch)			Release Ch

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
18		+MM_PwrOrSimOff (C_SIMneedRmv)			
19		+localtree2			
20		[StartType = C_Start_Imsi]			
21		+Authentication(TCV_ch, cksn, rand)			
22		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiImsi_01iei, mcc, mnc, TCV_lac))		Location Accept with IMSI
23		+PostMainLinkRel(TCV_ch)			Release Ch
24		+localtree2			
25		[(StartType = C_Start_Tmsi) OR (StartType = C_Start_TmsiOff)]			
26		+Authentication(TCV_ch, cksn, rand)			
27		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiTmsi_01iei, mcc, mnc, TCV_lac))		Location Accept with TMSI
28		L?DL_DatInTmsireCom	TmsiReallocCmp(TCV_ch)		
29		+PostMainLinkRel(TCV_ch)			Release Ch
30		+localtree2			
31		[StartType = C_Start_PLMNnot]			
32		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(C_rc_plmn_not))		Delete LAI, tmsi, cksn
33		+PostMainLinkRel(TCV_ch)			Release Ch
34		+localtree2			
35	E_0098	[(StartType <> C_Start_Rej) AND (StartType <> C_Start_Tmsi) AND (StartType <> C_Start_TmsiOff) AND (StartType <> C_Start_Imsi) AND (StartType <> C_Start_PLMNnot)]		I	
36		?TIMEOUT T_dly			
37		+ltree_rmvPLMNfbdnlist			
		localtree2			
38		[(att = 1) AND (StartType <> C_Start_TmsiOff)]			ATT no change
39		[cch_con <> C_cch_1Comb]			If non combined
40		+localtree3			
41		+ltree_PgReorg			
42		[cch_con = C_cch_1Comb]			If combined
43		[(att = 1) AND (StartType = C_Start_TmsiOff)]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
44		+ImsiDetach (MiTmsi_01, ta, C_SIMneedRmv)			Switch off, detach
45		[cch_con <> C_cch_1Comb]			If non combined
46		+localtree3			
47		[cch_con = C_cch_1Comb]			If combined
48		[att = 0]			
49		[cch_con = C_cch_1Comb]			If combined
50		+ltree_sysinfo(0, C_cch_1Comb)			ATT=0
51		+ltree_PgReorg			
52		[cch_con <> C_cch_1Comb]			If non combined
53		+localtree3			
54		+ltree_PgReorg			
		localtree3			
55		(TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
56		+Varinit_fix(activ_cell, lac, subch, cksn, rand, arfcngsm, arfcndcs, TCV_Ccd0H, imsi)			
57		+ltree_chconf (cch_con)			Config LT
58		+ltree_sysinfo(att, cch_con)			
		ltree_PgReorg			
59		(TCV_Null:=OM_PgFill(activ_cell, PgReqTp1Reorg))			
60		START T_dly(5000)			
61		?TIMEOUT T_dly			
62		(TCV_Null:=OM_PgFill(activ_cell, PgReqTp1Norm))			
		ltree_sysinfo(attach : INTEGER; con_ch : B_3)			
63		[activ_cell = C_CellA]			
64		+SysInfoSending(attach, con_ch)			SI1−4, cellA
65		(TCV_Ccd0A := TCV_Ccd0H, TCV_sysinfo5 := TCV_sysinf5, TCV_sysinfo6 := TCV_sysinf6)			
66		[con_ch = C_cch_1Comb]			If combined
67		+SysInfo_SacchSending(TCV_sacch, TCV_sysinfo5, TCV_sysinfo6)			SI5, 6 on SACCH/C4, cell A
68		[con_ch <> C_cch_1Comb]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
69		+ltree_sysinfo_other(attach)			More SIs, cellA
70		[activ_cell = C_CellB]			
71		+SysInfoSending(attach, con_ch)			SI1−4, cellB
72		(TCV_Ccd0B := TCV_Ccd0H, TCV_sacch_B := TCV_sacch, TCV_sysinfo5_B := TCV_sysinf5, TCV_sysinfo6_B := TCV_sysinf6)			SI5, 6 on SACCH/ C4, cell B
73		[con_ch = C_cch_1Comb]			If combined
74		+SysInfo_SacchSending(TCV_sacch_B, TCV_sysinfo5_B, TCV_sysinfo6_B)			SI5, 6 on SACCH/ C4, cell A
75		[con_ch <> C_cch_1Comb]			No more SIs, cell B
76		[activ_cell = C_CellC]			
77		+SysInfoSending(attach, con_ch)			SI1−4, cellC
78		(TCV_Ccd0C := TCV_Ccd0H, TCV_sacch_C := TCV_sacch, TCV_sysinfo5_C := TCV_sysinf5, TCV_sysinfo6_C := TCV_sysinf6)			
79		[con_ch = C_cch_1Comb]			If combined
80		+SysInfo_SacchSending(TCV_sacch_C, TCV_sysinfo5_C, TCV_sysinfo6_C)			SI5, 6 on SACCH/ C4, cell C
81		[con_ch <> C_cch_1Comb]			No more SIs, cell C
		SysInfoSending(attachb : INTEGER; con_chs : B_3)			
82		[sysinfonb = C_Test_fh]			Hopping
83		+SysInfoSending_fh(sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, cchdgsm, cchddcs, crh, mtmcgsm, mtmcdcs, bcchflgsm, bcchfldcs, Re, bcc, ncc, nccp)			
84		[sysinfonb = C_Test_nfh]			Non hopping
85		+SysInfoSending_nfh(sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, crh, mtmcgsm, mtmcdcs, bcchflgsm, bcflaltgsm, bcchfldcs, bcflaltdcs, Re, bcc, ncc, nccp)			
86		[sysinfonb = C_Test_egsm]			EGSM

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
87		+SysInfoSending_e(sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, cchdgsm, crh, mtmcgsm, bcchflgsm, bcchfldcs, Re, bcc, ncc, nccp)			
88		[sysinfonb = C_Test_cbms]			CBMS
89		+SysInfoSending_cbms(sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, crh, mtmcgsm, mtmcdcs, bcchflgsm, bcchfldcs, Re, bcc, ncc, nccp, subch)			
		ltree_chconf(conf : B_3)			
90		[activ_cell = C_CellA]			
91		+ltree_chconf_A(conf)			Config LT
92		[activ_cell = C_CellB]			
93		+ltree_chconf_B(conf)			Config LT
94		[activ_cell = C_CellC]			
95		+ltree_chconf_C(conf)			Config LT
		ltree_chconf_A(conf_a : B_3)			
96		[conf_a = C_cch_1nonComb]			
97		+NonCombinedBCCH_A(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_a, bpm)			1 non combined CCCH, cell A
98		(TCV_Upd := C_NotCombined)			
99		[conf_a = C_cch_1Comb]			
100		[sysinfonb <> C_Test_cbms]			
101		+CombinedBCCH_A(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_a, bpm)			1 combined CCCH, cell A
102		(TCV_Upd := C_Combined)			
103		[sysinfonb = C_Test_cbms]			
104		+CombinedBCCH_A_CBMS(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_a, bpm)			1 combined CCCH + CBMS, cell A
105		(TCV_Upd := C_Combined)			
106		[conf_a = C_cch_2nonComb]			2 CCCHs
107		+NonCombinedBCCH_A(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_a, bpm)			1st non combined CCCH, cell A

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
108		+NonCombinedBCCH_A_2(acttype, slot2, tsc, ta, babr, conf_a, bpm)			2nd non combined CCCH, cell A
109		(TCV_Upd := C_NotCombined)			
110		[conf_a = C_cch_3nonComb]			
111		+NonCombinedBCCH_A(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_a, bpm)			1st non combined CCCH, cell A
112		+NonCombinedBCCH_A_2(acttype, slot2, tsc, ta, babr, conf_a, bpm)			2nd non combined CCCH, cell A
113		+NonCombinedBCCH_A_3(acttype, slot3, tsc, ta, babr, conf_a, bpm)			3rd non combined CCCH, cell A
114		(TCV_Upd := C_NotCombined)			
115		[conf_a = C_cch_4nonComb]			
116		+NonCombinedBCCH_A(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_a, bpm)			1st non combined CCCH, cell A
117		+NonCombinedBCCH_A_2(acttype ,slot2, tsc, ta, babr, conf_a, bpm)			2nd non combined CCCH, cell A
118		+NonCombinedBCCH_A_3(acttype, slot3, tsc, ta, babr, conf_a, bpm)			3rd non combined CCCH, cell A
119		+NonCombinedBCCH_A_4(acttype, slot4, tsc, ta, babr, conf_a, bpm)			4th non combined CCCH, cell A
120		(TCV_Upd := C_NotCombined)			
121	E_0099	[(conf_a <> C_cch_1nonComb) AND (conf_a <> C_cch_2nonComb) AND (conf_a <> C_cch_3nonComb) AND (conf_a <> C_cch_4nonComb) AND (conf_a <> C_cch_1Comb)] ltree_chconf_B(conf_b : B_3)		I	A wrong CCCH−CONF value, cell A
122		[conf_b = C_cch_1nonComb]			
123		+NonCombinedBCCH_B(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype,slot,tsc, ta, babr, conf_b, bpm)			1 non combined CCCH, cell B

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
124		(TCV_Upd := C_NotCombined)			
125		[conf_b = C_cch_1Comb]			
126		+CombinedBCCH_B(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_b, bpm)			1 combined CCCH, cell B
127		(TCV_Upd := C_Combined)			
128	E_0100	[(conf_b <> C_cch_1nonComb) AND (conf_b <> C_cch_1Comb)]		I	A wrong CCCH−CONF value,cell B
		ltree_chconf_C(conf_c : B_3)			
129		[conf_c = C_cch_1Comb]			
130		+CombinedBCCH_C(par_bspwr, mtmcgsm, mtmcdcs, arfcngsm, arfcndcs, acttype, slot, tsc, ta, babr, conf_c, bpm)			1 combined CCCH, cell C
131		(TCV_Upd := C_Combined)			
132	E_0101	[conf_c <> C_cch_1Comb]		I	
		ltree_sysinfo_other(attach : INTEGER)			
133		[(cch_con = C_cch_1nonComb) OR (cch_con = C_cch_1Comb)]			
134		[cch_con = C_cch_2nonComb]			
135		+SysInfoSending_fh(sch, C_BCCH_A_2, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, CellChDes_03, crh, mtmcgsm, mtmcdcs, BcchFreqLst_01, BcchFreqLst_48, Re, bcc, ncc, nccp)			SIs on 2nd CCCH
136		[cch_con = C_cch_3nonComb]			
137		+SysInfoSending_fh(sch, C_BCCH_A_2, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, CellChDes_03, crh, mtmcgsm, mtmcdcs, BcchFreqLst_01, BcchFreqLst_48, Re, bcc, ncc, nccp)			SIs on 2nd CCCH
138		+SysInfoSending_fh(sch, C_BCCH_A_3, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, CellChDes_03, crh, mtmcgsm, mtmcdcs, BcchFreqLst_01, BcchFreqLst_48, Re, bcc, ncc, nccp)			SIs on 3rd CCCH
139		[cch_con = C_cch_4nonComb]			
140		+SysInfoSending_fh(sch, C_BCCH_A_2, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, CellChDes_03, crh, mtmcgsm, mtmcdcs, BcchFreqLst_01, BcchFreqLst_48, Re, bcc, ncc, nccp)			SIs on 2nd CCCH

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
141		+SysInfoSending_fh(sch, C_BCCH_A_3, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, CellChDes_03, crh, mtmcgsm, mtmcdcs, BcchFreqLst_01, BcchFreqLst_48, Re, bcc, ncc, nccp)			SIs on 3rd CCCH
142		+SysInfoSending_fh(sch, C_BCCH_A_4, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, CellChDes_03, crh, mtmcgsm, mtmcdcs, BcchFreqLst_01, BcchFreqLst_48, Re, bcc, ncc, nccp)			SIs on 4th CCCH
		ltree_rmvPLMNfbdnlist			
143		(TCV_Null := OO_PLMNselModeMan())			
144		(TCV_Null := OO_SelPLMN(mnc))			
145		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn) CANCEL T_dly	ChReq(ChRequest_02)		Any channel request PDU
146		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
147		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, subch, slot, tsc, TCV_chdescr_arfcn, ta))		
148		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(C_norm_period_attach))		Any location update type
149		ACTIVATE(OtherEvents)			Restore Normal default
150		[StartType = C_Start_PLMNnot]			The PLMN is already in forbidden list on short SIM
151		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(C_rc_plmn_not))		Delete LAI, tmsi, cksn, PLMN still in list
152		+PostMainLinkRel(TCV_ch)			Release Ch
153		(TCV_Null := OO_PLMNselModeAuto())			
154		[StartType <> C_Start_PLMNnot]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
155		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiTmsi_r02iei, mcc, mnc, TCV_lac))		Location Accept with no TMSI
156		+PostMainLinkRel(TCV_ch)			Release Ch
157	E_01 02	(TCV_Null := OO_PLMNselModeAuto())		I	No ChReq received: Inconclusive

Detailed Comments : Formal Parameter Type Comments & Explanations Possible values

par_bspwr INTEGER Base station power level C_E_default, C_E_neighbourdefault

activ_cell CellID Activated cell identity C_CellA, C_CellB, C_CellC

sch LOGICCH Sync. channel identity
If formal para activ_cell = C_CellA, C_SCH_A
If formal para activ_cell = C_CellB, C_SCH_B
If formal para activ_cell = C_CellC, C_SCH_C

bcch LOGICCH 1st broadcast channel id
If formal para activ_cell = C_CellA, C_BCCH_A_1
If formal para activ_cell = C_CellB, C_BCCH_B_1
If formal para activ_cell = C_CellC, C_BCCH_C_1

acttype BITSTRING Type for activation of a channel C_Immass

slot SN Slot number for the 1st slot C_S0

slot2 SN Slot number for the 2nd slot
In case of 2 or 3 or 4 non combined CCCH C_S2
Otherwise (1 combined or 1 non combined) C_SlotNU

slot3 SN Slot number for the 3rd slot
In case of 3 or 4 non combined CCCH C_S4
Otherwise (1 combined or 1, 2 non combined) C_SlotNU

slot4 SN Slot number for the 4th slot
In case of 4 non combined CCCH channels C_S6
Otherwise (1 combined,1, 2, 3 non combined) C_SlotNU

tsc TSC Training Sequence code C_BCC

t INTEGER Rach Control Parameter field, Tx−Integer
for SysInfo1, SysInfo2, SysInfo3, SysInfo4 C_TxInt_5,
C_TxInt_7, C_TxInt_10

retr INTEGER Rach Control Parameter field, Max−Retrans
for SysInfo1, SysInfo2, SysInfo3, SysInfo4 C_Max_1,
C_Max_2, C_Max_7

neci INTEGER Requested value f. NECI bit in test case C_NECI_0,
C_NECI_1

att INTEGER Requested value f. ATTACH bit in test case C_ATT_0,
C_ATT_1

ta TA Requested Timing Advance value

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

babr B_3 Number of blocks reserved for acces grant
If arbitrary TSPX_AGBLKSx
If fixed C_BABR_0, C_BABR_1, C_BABR_2

cch_con B_3 Type of CCCH channel.
If arbitrary TSPX_CcchConfx
1 conbined CCCH channel C_cch_1Comb
1 non conbined CCCH channel C_cch_1nonComb
2 non conbined CCCH channels C_cch_2nonComb
3 non conbined CCCH channels C_cch_3nonComb
4 non conbined CCCH channels C_cch_4nonComb

bpm B_3 Number of multiframes for transmission
of PAGING REQUEST messages to the same
paging subgroup
If arbitrary TSPX_PAMFRMSx
If fixed (2, 4, 5, 9 multiframes) C_BPM_0, C_BPM_2,
C_BPM_3, C_BPM_7

t3212 OCTETSTR. Requested value for T3212 in test case (Periodic Location Update Value)
infinite C_T3212_0
6 mn / 1 deci hours C_T3212_1
12 mn / 2 deci hours C_T3212_2
30 mn / 5 deci hours C_T3212_5

ci CI Cell Identity
If formal para activ_cell = C_CellA C_ci_CellA
If formal para activ_cell = C_CellB C_ci_CellB
If formal para activ_cell = C_CellC C_ci_CellC

mcc OCTETSTR. Mobile Contry Code for the LAI C_MCC, C_MCC_x

mnc OCTETSTR. Mobile Network Code for the LAI C_PLMN_Home, C_PLMN_x

lac OCTETSTR. Local Area Code for the LAI C_LAC_x

co CO Cell Options CellOpt_xx

cchdgsm CCHD Cell Channels Description (GSM)
If formal para sysinfonb=C_Test_nfh CellChDes_omit
If formal para sysinfonb=C_Test_cbms CellChDes_omit
If formal para sysinfonb=C_Test_fh GSM CellChDes_xx

cchddcs CCHD Cell Channels Description (DCS)
If formal para sysinfonb=C_Test_nfh CellChDes_omit
If formal para sysinfonb=C_Test_cbms CellChDes_omit
If formal para sysinfonb=C_Test_fh DCS CellChDes_xx

crh INTEGER Cell Reselection Hysteresis C_CellReselectHys12,
C_CellReselectHys4

mtmcgsm INTEGER Requested maximum power level
for GSM MS C_MaxPwrLvlG

mtmcdcs INTEGER Requested maximum power level
for DCS MS C_MaxPwrLvlD

bcchflgsm NCD Neighbour Cells Description for GSM BcchFreqLst_xx

bcflaltgsm NCD Alternative Neighbour Cells Description for GSM.

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

If sysinfonb = C_Test_nfh Alternat.GSM BcchFreqLst_xx
Otherwise: BcchFreqLst_Omit

bcchfldcs NCD Neighbour Cells Description for DCS BcchFreqLst_xx

bcflaltdcs NCD Alternative Neighbour Cells Description for DCS
If sysinfonb = C_Test_nfh Alternat.DCS BcchFreqLst_xx
Otherwise: BcchFreqLst_Omit

Re B_1 Rach Control Parameter field,
Re−establishment bit for SysInfo1,
SysInfo2, SysInfo3, SysInfo4 C_Restablishment,
C_noRestablishmen
bcc BCC Base station Color Code C_BCC

ncc NCC Network Color Code C_NCC

StartType INTEGER Initial conditions for MS
Starting without TMSI C_Start_Imsi
Starting with Tmsi_01 C_Start_Tmsi
Starting with Tmsi_01, MS deactivated C_Start_TmsiOff
Starting with LAI deleted, MS deactivated C_Start_Rej
Starting with PLMN not allowed C_Start_PLMNnot

subch BITSTRING Selected sub−channel
If Formal para cch_con = C_cch_1Comb, TSPX_SDCCH4SubDef
TSPX_SDCCH4SubA, SubB, SubC
cksn BITSTRING Ciphering Key Sequence Number used for
authentication TSPX_CKSNDef, TSPX_CKSNA, B, C

rand BITSTRING Random Number for authentication TSPX_RANDDef,
TSPX_RANDA, B , C

arfcngsm INTEGER BCCH channel ARFCN (GSM)
If sysinfonb = C_Test_fh C_Noarfcn
Otherwise: If activ_cell = C_CellA
C_arfcnA_x If activ_cell = C_CellB C_arfcnB_x If
activ_cell = C_CellC C_arfcnC_x

arfcndcs INTEGER BCCH channel ARFCN (DCS)
If sysinfonb = C_Test_fh C_Noarfcn
Otherwise: If activ_cell = C_CellA
C_arfcnAd_x If activ_cell = C_CellB C_arfcnBd_x If
activ_cell = C_CellC C_arfcnCd_x

imsi HEXSTRING The expected IMSI TSPX_IMSI
In TC_26_7_3_2 C_shortIMSI

sysinfonb INTEGER Type of sysinfo sending
Frequency Hopping C_Test_fh
Non Frequency hopping C_Test_nfh
EGSM sequence C_Test_egsm
CBMS sequence C_Test_cbms

nccp NCCP Network Country Code Permitted
in SysInfo2 C_NCCP_2, C_NCCP_5

Test Step Dynamic Behaviour

Test Step Name : PreEstRRConn(slot : SN; tsc : TSC; ta : TA)

Group : Preambles/

Objective : To establish a RR connection on SDCCH subchannel defined by TSPX_SDCCH4SubDef in cell A

Default : OtherEvents

Comments : The calling tree shall prepare variable for the step: TCV_ch for the SDCCH4 subchannel TSPX_SDCCH4SubDef. This test step is used for non RR testing.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_UdatRqPg1Rq	PgReq1(TCV_PgCh, TCV_Pgg, PgReqTp1_01)		
2		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_17)		
3		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
4		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, TSPX_SDCCH4SubDef, slot, tsc, TCV_chdescr_arfcn, ta))		
5		L?DL_EstInPgRes	PagingRes(PagingRes_01)		
6		ACTIVATE (OtherEvents)			Restore Normal default

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : StartCellA_ho(par_bspwr, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, mnc, lac : OCTETSTRING; cchdgsm, cchddcs : CCHD; bcchflgsm, bcchfldcs : NCD; Re : BITSTRING; bcc : BCC; ncc : NCC; td, fn : INTEGER)

Group : Preambles/

Objective : To setup a physical channel as combined BCCH, CCCH and SDCCH4 then broadcast SYSTEM INFORMATION messages 2, 3, 4, 5 and 6 with default parameters of cell A.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_sacch := OC_SubchOfSacch4(TSPX_SDCCH4SubDef, C_CellA))			
2		+ltree_chconf			
3		+SysInfoSending_fh(C_SCH_A, C_BCCH_A_1, t, retr, neci, att, babr, cch_con, bpm, t3212, C_ci_cellA, mcc, mnc, lac, CellOpt_01, cchdgsm, cchddcs, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, bcchflgsm, bcchfldcs, Re, bcc, ncc, C_NCCP_2)			
4		(TCV_Ccd0A := TCV_Ccd0H, TCV_sysinfo5 := TCV_sysinf5, TCV_sysinfo6 := TCV_sysinf6)			
5		[cch_con = C_cch_1Comb]			
6		+SysInfo_SacchSending(TCV_sacch, TCV_sysinfo5, TCV_sysinfo6)			
7		[cch_con = C_cch_1nonComb]			
8	E_01 03	[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	
		ltree_chconf			
9		[cch_con = C_cch_1Comb]			
10		+CombinedBCCH_A_ho(par_bspwr, C_MaxPwrLvlG, C_MaxPwrLvlD, arfcn1, arfcn2, acttype, slot, tsc, ta, babr, cch_con, bpm, td, fn)			
11		[cch_con = C_cch_1nonComb]			
12		+NonCombinedBCCH_A_ho(par_bspwr, C_MaxPwrLvlG, C_MaxPwrLvlD, arfcn1, arfcn2, acttype, slot, tsc, ta, babr, cch_con, bpm, td, fn)			
13	E_01 04	[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : StartCellB_ho(par_bspwr, arfcn1, arfcn2 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; t, retr, neci, att : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, mnc, lac : OCTETSTRING; cchdgsm, cchddcs : CCHD; bcchflgsm, bcchfldcs : NCD; Re : BITSTRING; bcc : BCC; ncc : NCC; td, fn : INTEGER)

Group : Preambles/

Objective : To setup a physical channel as combined BCCH, CCCH and SDCCH4 then broadcast SYSTEM INFORMATION messages 2, 3, 4, 5 and 6 with default parameters of cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_sacch_B := OC_SubchOfSacch4(TSPX_SDCCH4SubDef, C_CellB))			
2		+ltree_chconf			
3		+SysInfoSending_fh(C_SCH_B, C_BCCH_B_1, t, retr, neci, att, babr, cch_con, bpm, t3212, C_ci_cellB, mcc, mnc, lac, CellOpt_01, cchdgsm, cchddcs, C_CellReselectHys12, C_MaxPwrLvlG, C_MaxPwrLvlD, bcchflgsm, bcchfldcs, Re, bcc, ncc, C_NCCP_2)			
4		(TCV_Ccd0B := TCV_Ccd0H, TCV_sysinfo5_B := TCV_sysinf5, TCV_sysinfo6_B := TCV_sysinf6)			
5		[cch_con = C_cch_1Comb]			
6		+SysInfo_SacchSending(TCV_sacch_B, TCV_sysinfo5_B, TCV_sysinfo6_B)			
7		[cch_con = C_cch_1nonComb]			
8	E_0105	[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	
		ltree_chconf			
9		[cch_con = C_cch_1Comb]			
10		+CombinedBCCH_B_ho(par_bspwr, C_MaxPwrLvlG, C_MaxPwrLvlD, arfcn1, arfcn2, acttype, slot, tsc, ta, babr, cch_con, bpm, td, fn)			
11		[cch_con = C_cch_1nonComb]			
12		+NonCombinedBCCH_B_ho(par_bspwr, C_MaxPwrLvlG, C_MaxPwrLvlD, arfcn1, arfcn2, acttype, slot, tsc, ta, babr, cch_con, bpm, td, fn)			
13	E_0106	[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	

Detailed Comments :

Default Dynamic Behaviour

Default Name : OtherEvents
Group :
Objective : To match unexpected events and sign final verdict for preambles.
Comments : used in preambles.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RelIn	DLRelInd_02		4.
2		RETURN			
3		L?DL_UdatIn	DLUdatInd_01		
4		RETURN			
5		?TIMEOUT T_guard			1.
6		(TCV_Null := OO_TguardTimeOut())			
7	E_01 07	CANCEL		I	3.
8		L?OTHERWISE			2.
9	E_01 09	CANCEL		I	3.

Detailed Comments : 1. The guard timer times out, inconclusive.
2. Unexpected events, inconclusive.
3. Cancel of all running timers.
4. Local end termination request allowed.

Default Dynamic Behaviour

Default Name : OtherEventsFail
Group :
Objective : To match unexpected events and fail the test case.
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RelIn	DLRelInd_02		
2		RETURN			
3		L?DL_UdatIn	DLUdatInd_01		
4		RETURN			
5		?TIMEOUT T_guard			1.
6		(TCV_Null := OO_TguardTimeOut())			
7	E_01 10	CANCEL		I	3.
8		L?OTHERWISE			2.
9	E_01 12	CANCEL		F	3.

Detailed Comments : 1. The guard timer times out, inconclusive.
2. Unexpected events, fail.
3. Cancel of all running timers.

Default Dynamic Behaviour

Default Name : OtherEventsFail_01

Group :

Objective : To match irrelevant CHANNEL REQUEST msg and MEASUREMENT REPORT msg and return or match other unexpected events and fail the test case.

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RelIn	DLRelInd_01		6.
2		RETURN			
3		L?DL_UdatIn	DLUdatInd_01		
4		RETURN			
5		?TIMEOUT T_guard			1.
6		(TCV_Null := OO_TguardTimeOut())			
7	E_01 13	CANCEL		I	5.
8		L?DL_RacInChRq	ChReq(ChRequest_02)		2.
9		RETURN			
10		L?DL_UdatInMsrRpt	MsrRept(MsrReport_02)		3.
11		RETURN			
12		L?OTHERWISE			4.
13	E_01 15	CANCEL		F	5.

Detailed Comments :
1. The guard timer times out, inconclusive.
2. To throw away any irrelevant channel request.
3. To throw away any measurement report.
4. Other unexpected events, fail.
5. Cancel of all running timers.
6. Cover also DLRelInd_02 for local end termination.

Default Dynamic Behaviour

Default Name : OtherEventsFail_02

Group :

Objective : To match unexpected events and fail the test case but ignore channel request messages that are sent before the lower tester has sent (and the MS received) the Immediate Assigment message.

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RelIn	DLRelInd_02		
2		RETURN			
3		L?DL_UdatIn	DLUdatInd_01		
4		RETURN			
5		L?DL_RacInChRq	ChReq(ChRequest_02)		1.
6		RETURN			
7		?TIMEOUT T_guard			2.
8		(TCV_Null := OO_TguardTimeOut())			
9	E_0116	CANCEL		I	4.
10		L?OTHERWISE			3.
11	E_0118	CANCEL		F	4.

Detailed Comments : 1. To throw away any irrelevant channel request.
2. The guard timer times out, inconclusive.
3. Unexpected events, fail.
4. Cancel of all running timers.

Default Dynamic Behaviour

Default Name : OtherEvents_01

Group :

Objective : To match irrelevant messages and return

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?OTHERWISE			1.
2		RETURN			

Detailed Comments : 1. Cover also DLRelInd_02 for local end termination.

Default Dynamic Behaviour

Default Name : OtherEvents_02
Group :
Objective : To match unexpected events and sign final verdict for preambles but ignore channel request messages that are sent before the lower tester has sent (and the MS received) the Immediate Assigment message.
Comments : used in preambles.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RelIn	DLRelInd_02		
2		RETURN			
3		L?DL_UdatIn	DLUdatInd_01		
4		RETURN			
5		L?DL_RacInChRq	ChReq(ChRequest_02)		1.
6		RETURN			
7		?TIMEOUT T_guard			2.
8		(TCV_Null := OO_TguardTimeOut())			
9	E_0119	CANCEL		I	4.
10		L?OTHERWISE			3.
11	E_0121	CANCEL		I	4.

Detailed Comments : 1. To throw away any irrelevant channel request.
2. The guard timer times out, inconclusive.
3. Unexpected events, inconclusive.
4. Cancel of all running timers.

Default Dynamic Behaviour

Default Name : RcvHdOvAcc
Group :
Objective : To match any HANDOVER ACCESS then return to calling tree.
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RacInHoacc	HndOvAcc_01		
2		RETURN			

Detailed Comments :