I

Test Suite Overview

Test Suite Structure

Suite Name : Rel4_RGSM_v420
Standards Ref : EN 300 911, TS 100 930, TS 24.008, GSM 04.18
PICS Ref : TS 100.607−2
PIXIT Ref : EN 300 607−3
Test Method(s) : Distributed Single Layer Test Method
Comments : Release 4 RGSM ATS v 4.2.0 after GERAN#5 (GERAN4#4)

Test Group Reference	Selection Ref	Test Group Objective	Page Nr
CellSelection/	SelExpr_1000R		392
Signalling/	SelExpr_1000R		435

Detailed Comments :

Test Case Index

Test Group Reference	Test Case Id	Selection Ref	Description	Page Nr
CellSelection/	TC_20_21_1R			392
CellSelection/	TC_20_21_2R			396
CellSelection/	TC_20_21_4R			398
CellSelection/	TC_20_21_5R			402
CellSelection/	TC_20_21_6R			405
CellSelection/	TC_20_21_7R			410
CellSelection/	TC_20_21_8R			414
CellSelection/	TC_20_21_9R			416
CellSelection/	TC_20_21_10R			418
CellSelection/	TC_20_21_12R			420
CellSelection/	TC_20_21_13R			422
CellSelection/	TC_20_21_15R			425
CellSelection/	TC_20_21_16R			430
CellSelection/	TC_20_21_17R			433
Signalling/	TC_26_10_2_1R	SelExpr_1001	R−GSM signalling/ RR / measurement	435
Signalling/	TC_26_10_2_2R	SelExpr_1000R	R−GSM signalling/ RR / immediate assignment	440
Signalling/	TC_26_10_2_3R	SelExpr_1000R	R−GSM signalling/ RR / channel assignment procedure	442
Signalling/	TC_26_10_2_4_1R	SelExpr_1001	R−GSM signalling/ RR / Handover / successful case	448
Signalling/	TC_26_10_2_4_2R	SelExpr_1001	R−GSM signalling/ RR / Handover / layer 1 failure	454
Signalling/	TC_26_10_2_5R		R−GSM signalling/ RR / frequency redifinition	457
Signalling/	TC_26_10_3_1R	SelExpr_1002	R−GSM signalling/ structured procedure / mobile originating call	460
Signalling/	TC_26_10_3_2R	SelExpr_1003	R−GSM signalling/ structured procedure / emergency call	463

Detailed Comments :

Test Step Index			
Test Step Group Reference	**Test Step Id**	**Description**	**Page Nr**
management/ChConfig/	CCConfigTCH_R		466
management/ChConfig/	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.	467
management/ChConfig/	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.	468
management/ChConfig/	Config_BCCH_CCCH_A_ho_1		469
management/ChConfig/	Config_BCCH_CCCH_A_2	To set a physical channel and map the second BCCH, CCCH onto the physical channel which represents cell A.	470
management/ChConfig/	Config_BCCH_CCCH_A_3	To set a physical channel and map the third BCCH, CCCH onto the physical channel which represents cell A.	471
management/ChConfig/	Config_BCCH_CCCH_A_4	To set a physical channel and map the fourth BCCH, CCCH onto the physical channel which represents cell A.	472
management/ChConfig/	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.	473
management/ChConfig/	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.	474
management/ChConfig/	Config_BCCH_CCCH_C_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell C.	475
management/ChConfig/	Config_BCCH_CCCH_D_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell D.	476
management/ChConfig/	Config_BCCH_CCCH_E_1	To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell E.	477
management/ChConfig/	Config_FACCHF_A_1	To set one physical channel used as TCHF_ACCH's of cell A.	478
management/ChConfig/	Config_FACCHF_A_2	To set one physical channel used as TCHF_ACCH's for instance 2 of cell A.	479
management/ChConfig/	Config_FACCHF_B_1	To set one physical channel used as TCHF_ACCH's of cell B.	480
management/ChConfig/	Config_FACCHH_A_1	To set one physical channel used as TCHH_ACCH's of cell A.	481
management/ChConfig/	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.	482

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
management/ChConfig/	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.	483
management/ChConfig/	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	484
management/ChConfig/	Config_SDCCH4_D_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell D	485
management/ChConfig/	Config_SDCCH4_E_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell E	486
management/ChConfig/	Config_SDCCH4_F_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell F	487
management/ChConfig/	Config_SDCCH4_G_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell G	488
management/ChConfig/	Config_SDCCH4_H_1	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell H	489
management/ChConfig/	Config_SDCCH8_A_1	To set one physical channel used as SDCCH8 channel for instance 1 of cell A.	490
management/ChConfig/	Config_SDCCH8_A_2	To set one physical channel used as SDCCH8 channel for instance 2 of cell A.	491
management/ChConfig/	Config_SDCCH8_B_1	To set one physical channel used as SDCCH8 channel for instance 1 of cell B.	492
management/ChConfig/	Config_SDCCH8_C_1	To set one physical channel used as SDCCH8 channel for instance 1 of cell C.	493
management/ChConfig/	CombinedBCCH_A_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.	494
management/ChConfig/	CombinedBCCH_A_CBMS_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell A.	494
management/ChConfig/	CombinedBCCH_B_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell B.	494
management/ChConfig/	CombinedBCCH_C_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell C.	495

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
management/ChConfig/	CombinedBCCH_D_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell D.	495
management/ChConfig/	CombinedBCCH_E_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell E.	495
management/ChConfig/	CombinedBCCH_F_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell F.	496
management/ChConfig/	CombinedBCCH_G_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell G.	496
management/ChConfig/	CombinedBCCH_H_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell H.	496
management/ChConfig/	NonCombinedBCCH_A_R	To set one physical channel used as FCHH_SCH_BCCH_CCCH for cell A.	497
management/ChConfig/	NonCombinedBCCH_A_2_R	To set a physical channel and map the second BCCH, CCCH onto the physical channel which represents cell A.	497
management/ChConfig/	NonCombinedBCCH_A_3_R	To set a physical channel and map the third BCCH, CCCH onto the physical channel which represents cell A.	497
management/ChConfig/	NonCombinedBCCH_A_4_R	To set a physical channel and map the fourth BCCH, CCCH onto the physical channel which represents cell A.	498
management/ChConfig/	NonCombinedBCCH_B_R	To set a physical channel and map FCCH, SCH, BCCH, CCCH onto the physical channel which represents cell B for RR testing.	498
management/ChConfig/	FullRateCh_A_1_R	To set one physical channel used as TCHF_ACCH's for instance 1 of cell A.	499
management/ChConfig/	FullRateCh_A_1_nociph_R	To set one physical channel used as TCHF_ACCH's for instance 1 of cell A.	499
management/ChConfig/	FullRateCh_A_2_nociph_R	To set one physical channel used as TCHF_ACCH's for instance 2 of cell A for TC_26_6_13_1.	500
management/ChConfig/	FullRateCh_B_1_R		500
management/ChConfig/	FullRateCh_B_1_nociph_R	To set one physical channel used as TCHF_ACCH's for instance 1 of cell B.	500
management/ChConfig/	HalfRateCh_A_1_R	To set one physical channel used as TCHH_ACCH's for instance 1 of cell A.	501

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
management/ChConfig/	HalfRateCh_A_1_nociph_R	To set one physical channel used as TCHH_ACCH's for instance 1 of cell A.	501
management/ChConfig/	SDCCH8_A_1_nociph_R	To set one physical channel used as SDCCH8 channel for instance 1 of cell A.	502
management/ChConfig/	SDCCH8_A_2_nociph	To set one physical channel used as hopping SDCCH8 channel for instance 2 of cell A, for TC_26_6_13_1.	502
management/ChConfig/	SDCCH8_B_1_R	To set one physical channel used as hopping SDCCH8 channel for instance 1 of cell B .	503
management/ChConfig/	SDCCH8_B_1_nociph_R	To set one physical channel used as hopping SDCCH8 channel for instance 1 of cell B For TC_26_6_13_5.	503
management/ChConfig/	SDCCH8_C_1_nociph_R	To set one physical channel used as hopping SDCCH8 channel for instance 1 of cell C For TC_26_6_13_5.	504
Miscellaneous/	Adjust_gsmanddcs_powerlvl_R		504
Miscellaneous/	AssCh_complete		505
Miscellaneous/	AssCmdGen		506
Miscellaneous/	Authentication	To execute the authentication procedure.	506
Miscellaneous/	CallProcGen	To generate a CALL PROCEEDING message .	507
Miscellaneous/	CCCH_group_Paging_group		509
Miscellaneous/	CCstatuschk_02	To check whether the MS under test is in the CC state 'st'.	510
Miscellaneous/	Check_HoaccPara_R	To check the power level and the timing advance of handover access bursts.	511
Miscellaneous/	Check_Result	The test step checks the TCV_Res value.	512
Miscellaneous/	Check_Time	The test step calls FnArith to calculate the MS return time for HANDOVER COMPLETE or AOC acknowledge stored in TCV_Time.	513
Miscellaneous/	Ciphering_on	To switch on the ciphering.	513
Miscellaneous/	FnArith	To calcurate the difference of two frame numbers 'fn1' and 'fn'. The test step uses TCV_Cnt1, TCV_Cnt2, TCV_K and TCV_Time.	514
Miscellaneous/	ImsiDetach	If possible, SIM_Remove or PowerDown or SwitchOff, and then to execute the IMSI Detach procedure. This test step does not allowed to execute the switch−off even if the MS supports both the SIM remove detach and the soft switch−off detach.	515
Miscellaneous/	MM_LupInit2_R	To execute the initialisation of theLocation Update Procedure.	516

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Miscellaneous/	MM_LupRej_R	To execute the Location Update Procedure, which shall be rejected.	516
Miscellaneous/	MM_PwrOrSimOff	If possible, emove SIM card or to switch off the MS or to remove power source.	517
Miscellaneous/	MM_PwrOrSimOn	If possible to insert the SIM card or to switch on the MS or to restore the power source.	517
Miscellaneous/	NoReaction	To check whether MS doesn't initiate any RR connections.	518
Miscellaneous/	PageOnCarriers		519
Miscellaneous/	Pause		520
Miscellaneous/	RR_hocomp1	To finish the HO−procedure in non−synchronized HO cases.	521
Miscellaneous/	RRmtcallprepare	To establish a full rate call with non hopping in cell A for GSM900 and DCS1800. IUT should be in idle updated state.	521
Miscellaneous/	RcvSetupOrEsetup		522
Miscellaneous/	SelectPagingCh		523
Miscellaneous/	SendSeqNo_chk	check sending sequence number.	524
Miscellaneous/	Switchcell_R		524
Miscellaneous/	SvcSupportedChk	To check wether the Basic Service svc is supported against PICS answers.	525
Miscellaneous/	Varinit_fix_R		526
Miscellaneous/	Wait	To check whether MS doesn't initiate any RR connections.	527
Miscellaneous/SysInfo/	SysInfoSending_cbms_R	To send system information messages for the CBMS tests.	527
Miscellaneous/SysInfo/	SysInfoSending_e	Sending of Systeminformation for cell A in EGSM test case TC_26_10_2_2.	528
Miscellaneous/SysInfo/	SysInfoSending_fh_R	To send system information messages for the L3 tests. The following parameters specified by input parameters.	529
Miscellaneous/SysInfo/	SysInfoSending_fh_pl_R		530
Miscellaneous/SysInfo/	SysInfoSending_fh_pl_R2		531
Miscellaneous/SysInfo/	SysInfoSending_nfh_R	To send system information messages with default parameters defined for L3 tests for which no other special parameters indicated.	533
Miscellaneous/SysInfo/	SysInfoSending_r	Sending of Systeminformation for cell A in RGSM test case TC_26_10_2_2R.	534
Miscellaneous/SysInfo/	SysInfo_SacchSending	To send SYSTEM INFORMATION 5 and 6 messages defined by parameters 'sysinfo5_pdu' and 'sysinfo6_pdu' in the parametrised 'ch' channel.	535

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Miscellaneous/SysInfo/	SysInfo_5bisSending	To send SYSTEM INFORMATION 5bis message defined by parameters 'sysinfo5bis_pdu' in the parametrised 'ch' channel.	535
OperatorOP/	InitCall	To initiate a call of the basic service 'srv'.	535
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.	536
Postambles/	PostMainLinkRel	To release the main signalling link 'ch', and bring the MS back to Idle state.	536
Postambles/	ChanRel	To release the RR connection on the channel TCV_ch and bring the MS back to Idle state.	536
Preambles/	BasicServiceMO	1. This test step generates an appropriate MO SETUP message according to the IXIT parameters of Bearer services/Teleservices for an MO call. 2. The detailed algorithms for Bcap derivation for each Bearer service/Teleservice are described in test steps BS2xMO, BS3xMO, BS4xMO, BS5xMO, BS61or81MO, TS61MO and TS62MO.	537
Preambles/	BasicServiceMT	1. This test step generates an appropriate MT SETUP message according to the IXIT parameters of Bearer services/Teleservices for an MT call. If possible the test step generates a 2nd MT SETUP message for TC_11_1_1. 2. The Bearer Capability in the 2nd SETUP message is chosen in such a way that as many parameters as possible are different from the Bcap in the first MT SETUP message. The detailed algorithms for Bcap derivation for each Bearer service/Teleservice are described in test steps BS2xMT, BS3xMT, BS61or81MT, TS61MT and TS62MT.	540
Preambles/	BS2xMO	MO SETUP message with right Bcap for BS2x service.	543
Preambles/	BS2xMT	MT SETUP message with right Bcap for BS2x service.	547
Preambles/	BS3xMO	MO SETUP message with right Bcap for BS3x service.	551
Preambles/	BS3xMT	MT SETUP message with right Bcap for BS3x service.	555
Preambles/	BS4xMO	MO SETUP message with right Bcap for BS4x service.	561

Continued on next page

Continued from previous page

Test Step Index

Test Step Group Reference	Test Step Id	Description	Page Nr
Preambles/	BS5xMO	MO SETUP message with right Bcap for BS5x service.	563
Preambles/	BS61or81MO	To get a MO SETUP message with right BC IE for BS61 or BS81 service.	564
Preambles/	BS61or81MT	To get a MT SETUP message with right BC IE for BS61 or BS81 service.	569
Preambles/	Chmod	To assign correct values to the test case variables TCV_ChMod and TCV_ChModb.	574
Preambles/	Est_MO_Call_init_R	To initiate a mobile originating call for the supported bearer capability. Only in HO cases.	575
Preambles/	IdleUpdated_R	This test step is called at the preamble of each test case.	577
Preambles/	IdleUpdated_R_CS	This test step is called at the preamble of each test case.	588
Preambles/	PreEnterCCstateU10_r01_R	To establish a mobile terminating call for the supported bearer capability and put the MS in the CC state U10 with late assignment.	594
Preambles/	PreEstRRConn	To establish a RR connection on TSPX_SDCCH4SubDef	595
Preambles/	StartCellB_R	To broadcast SYSTEM INFORMATION messages 2, 3, 4, 5 and 6 with default parameters in cell B.	596
Preambles/	StartCellB_rgsm	To broadcast SYSTEM INFORMATION messages 2, 3, 4, 5 and 6 with default parameters in cell B.	597
Preambles/	StartMultiCells_02_R	To broadcast SYSTEM INFORMATION messages 1, 2, 3, 4, 5 and 6 in multiple cells for measurement testing. Neighbour cells description for cell S1 is a formal parameter.	598
Preambles/	StartMultiCells_02r_R	To broadcast SYSTEM INFORMATION messages 1, 2, 3, 4, 5 and 6 in multiple cells for measurement testing. Neighbour cells description for cell S1 is a formal parameter.	600
Preambles/	TS61MO	MO SETUP message with right Bcap for TS61 service.	602
Preambles/	TS62MO	MO SETUP message with right Bcap for TS62 service.	604
Preambles/	TS61MT	MT SETUP message with right Bcap for TS61 service.	606
Preambles/	TS62MT	MT SETUP message with right Bcap for TS62 service.	609

Detailed Comments :

Default Index

Default Group Reference	Default Id	Description	Page Nr
	OtherEvents	To match unexpected events and asign a final verdict for preambles.	611
	OtherEventsFail	To match unexpected events and fail the test case.	611
	OtherEventsFail_01	To match irrelevant CHANNEL REQUEST msg and return or match other unexpected events and fail the test case.	612
	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.	612
	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.	613
	RcvHdOvAcc	To match any HANDOVER ACCESS then return to calling tree.	613

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
B_27	BITSTRING [27]		Generic type of 27 bits
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
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
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
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
IEI_4	BITSTRING [4]		information element identifier, type 1
IEI_8	BITSTRING [8]		information element identifier, type 2−4

Continued on next page

Continued from previous page

Simple Type Definitions			
Type Name	**Type Definition**	**Type Encoding**	**Comments**
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		
MAIO	BITSTRING[6]		MAIO,GSM 04.08 10.5.2.5
MT	BITSTRING[8]		message type, structure of it see MT0
NCC	BITSTRING [3]		PLMN colour code
NCCP	OCTETSTRING [1]		ncc permitted
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
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
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

Continued on next page

Continued from previous page

Simple Type Definitions			
Type Name	**Type Definition**	**Type Encoding**	**Comments**
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
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
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 : ALPT
Encoding Variation :
Comments : Alerting pattern
GSM 04.08, 10.5.4.26

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'00100100'B
iel	LENGTH		
sp4b	B_4		4 spare bits
alptv	B_4		alerting pattern value

Detailed Comments :

Structured Type Definition

Type Name : BARANGE
Encoding Variation :
Comments : BA range, GSM 04.08, 10.5.2.1a

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		'73'O
iel	LENGTH		
nor	B_8		number of ranges
ranges	OCTETSTRING [3..248]		ranges

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 : 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, 10.5.2.1b

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, 10.5.2.5a

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01100100’B, ’1C’O, ’1D’O
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		’63’O, ’66’O, ’11’O, ’12’O − ’18’O
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 : 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 : DCR
Encoding Variation :
Comments : Descriptive group or broadcast call reference
GSM 04.08, 10.5.1.9

Element Name	Type Definition	Field Encoding	Comments
gbcref	B_27		
sf	B_1		SF service flag
af	B_1		AF acknowledgment flag
callpr	B_3		call priority
ciphinfo	B_4		ciphering information

Detailed Comments : &COMMON_U06

Structured Type Definition

Type Name : Dm_TBF
Encoding Variation :
Comments : Dedicate mode or TBF
GSM 04.08, 10.5.2.25b

Element Name	Type Definition	Field Encoding	Comments
spb	B_1		spare bit
tma	B_1		two msg assignment
d	B_1		Downlink
td	B_1		TBF/Dedicate mode

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	Components		Components of asn.1 type, for receiving uplink components

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, ’1E’O
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, 10.5.2.13, 10.5.2.14

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 : GCHD
Encoding Variation :
Comments : Group channel description, GSM 04.08, 10.5.2.14b

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’74’O
iel	LENGTH		3−11
chd	CHD		without CHD iei
ma	MA		without MA iei, iel

Detailed Comments :

Structured Type Definition

Type Name : GCI
Encoding Variation :
Comments : Group call information
GSM 04 08, 9.1.21a

Element Name	Type Definition	Field Encoding	Comments
gbcr	DCR		descriptive group or broadcast call reference M OCTETS [4.5]
gchdi	B_1		group channel description indicator M BITSTRING[1]
gchd	GCHD		channel description O BITSTRING[24]
hci	B_1		hopping case indicator M BITSTRING[1]
mai	B_1		mobile allocation indicator M BITSTRING[1]
ma	MA		mobile allocation O
fsli	B_1		frequency short list indicator M BITSTRING[1]
fsl	BITSTRING[64]		frequency short list C lh=0 lh1=1 lh2=1 lh3=1 BITSTRING[64]

Detailed Comments :

Structured Type Definition

Type Name : GCKN
Encoding Variation :
Comments : Group cipher key number, GSM 04.08, 9.1.7.2

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_4		'8x'O
gckn	B_4		group cipher key number

Detailed Comments :

Structured Type Definition

Type Name : GPRSR
Encoding Variation :
Comments : GPRS Resumption
GSM 04.08, 10.5.2.14c

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_4		'01'O
spr	SP3B		spare bits
ack	B_1		

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 : HOACC_PARA
Encoding Variation :
Comments : Handover access parameters

Element Name	Type Definition	Field Encoding	Comments
num_bursts	OCTETSTRING[1]		number of bursts
min_mspwr	OCTETSTRING[1]		minimum MS power level
max_mspwr	OCTETSTRING[1]		minimum MS power level
min_msta	BITSTRING[8]		minimum timing advance
max_msta	BITSTRING[8]		maximum timing advance

Detailed Comments :

Structured Type Definition

Type Name : IARESTOCT
Encoding Variation :
Comments : GSM 04 08 10.5.2.16 IA Rest Octets

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 : 04.08, 10.5.2.16 Frequency parameter before time

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 : 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, ’21’O
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 : MSA
Encoding Variation :
Comments : Multislot allocation
GSM 04.08, 10.5.2.21b

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’10’
iel	LENGTH		
extb3	EXTB		extension bit, octet 3
dan	BITSTRING [7]		downlink ass. Tn = (n+TNm)mod8
extb3a	EXTB		extension bit, octet 3a
uan	BITSTRING [7]		uplink ass. Tn = (n+TNm)mod8
chs_1n	BITSTRING [8]		tn7 − tn0 of channel set 1
chs_2n	BITSTRING [8]		tn7 − tn0 of channel set 2
chs_3n	BITSTRING [8]		tn7 − tn0 of channel set 3
chs_4n	BITSTRING [8]		tn7 − tn0 of channel set 4
chs_5n	BITSTRING [8]		tn7 − tn0 of channel set 5
chs_6n	BITSTRING [8]		tn7 − tn0 of channel set 6
chs_7n	BITSTRING [8]		tn7 − tn0 of channel set 7
chs_8n	BITSTRING [8]		tn7 − tn0 of channel set 8

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 (esind)	BITSTRING [1]		Controlled early classmark sending indicator
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 (esind)	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 : 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, 10.5.2.22a

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 : PCHD

Encoding Variation :

Comments : Packet channel description and packet response
GSM 04.08, 10.5.2.25a

Element Name	Type Definition	Field Encoding	Comments
ct	BITSTRING [5]		channel type, M, (’0001’B all other values are reserved)
tn	BITSTRING [3]		time slot number, M, (0−7)
tsc	BITSTRING[3]		training sequence code, M, (0−7)
lh	BITSTRING[1]		flag, O
sp	BITSTRING[2]		’00’B, C
arfcn	BITSTRING[10]		absolute RF channel number, C, (0−1023)
man	BITSTRING[4]		mobile allocation and associated hopping sequence number and training sequence code, C, (0−15)
maio	BITSTRING[6]		mobile allocation index offset, C, (0−63)
macm	BITSTRING[2]		identification of the consistent set of SI14 messages for the decoding of MA_NUMBER field, C

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 : P1RO
Encoding Variation :
Comments : GSM 04.08, 10.5.2.23

Element Name	Type Definition	Field Encoding	Comments
nlni	B_1		nln indicator (PCH)
nln	B_2		notification list number (PCH), optional
nlns	B_1		nln status, optional
pi1	B_1		priority indicator1
priority1	B_3		optional
pi2	B_1		priority indicator2
priority2	B_3		optional
gcii	B_1		group call information indicator
gci	GCI		group call information optional
ppi1	B_1		packet page indication 1
ppi2	B_1		packet page indication 2
spb	BITSTRING [1..7]		spare bits, optional
spd	OCTETSTRING [1..16]		’2B’O padding, optional

Detailed Comments : 1. Spare bits spb are constituted of spare L (GSM 04.07, B.1.2.2). The padding sequence is ’0010 1011’B. The bit value is corresponding to the padding spare bit for that position.
2. Spare padding octets are a sequence of ’2B’O. The total spare padding consists of the fields spb and spd (GSM 04.07, B.1.3).

Structured Type Definition

Type Name : PRT
Encoding Variation :
Comments : Priority
GSM 04.08, 10.5.1.11

Element Name	Type Definition	Field Encoding	Comments
pliei	B_4		priority iei
sp	B_1		spare
cp	B_3		call priority

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 : 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 : SI1RO
Encoding Variation :
Comments : GSM 04.08, 10.5.2.32

Element Name	Type Definition	Field Encoding	Comments
nchi	B_1		NCH indicator, ASCI supported M
nchp	B_5		NCH position, optional
spb	BITSTRING [1..7]		spare bits, optional

Detailed Comments : 1. Spare bits spb are constituted of spare L (GSM 04.07, B.1.2.2). The padding sequence is '0010 1011'B. The bit value is corresponding to the padding spare bit for that position.

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, M
cbq	BITSTRING [1]		Cell bar qualify, O
cro	BITSTRING [6]		Cell reselect offset, O
to	BITSTRING [3]		Temporary offset, O
pt	BITSTRING [5]		Penalty time, O
poi	BITSTRING [1]		Power offset indicator, M
po	BITSTRING [2]		Power offset, O
si2ti	BITSTRING [1]		SI 2ter indicator, M
ecsc	BITSTRING [1]		Early classmark setting control, M
si9i	BITSTRING [1]		SI9 indicator, M
where	BITSTRING [3]		scheduling si9 on BCCH, O
gprsi	BITSTRING [1]		GPRS supported M
racolour	BITSTRING [3]		routing area colour code O
si13p	BITSTRING [1]		SI13 position O
spb	BITSTRING [1..7]		spare bits, O
spd	OCTETSTRING [1..4]		’2B..2B’O padding, O

Detailed Comments : The info element has a fixed length of 4 octets.
1. Spare bits spb are constituted of spare L (GSM 04.07, B.1.2.2). The padding sequence is ’0010 1011’B. The bit value is corresponding to the padding spare bit for that position.
2. Spare padding octets are a sequence of ’2B’O. The total spare padding consists of the fields spb and spd (GSM 04.07, B.1.3).

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 M
cbq	BITSTRING [1]		Cell bar qualify O
cro	BITSTRING [6]		Cell reselect offset O
to	BITSTRING [3]		Temporary offset O
pt	BITSTRING [5]		Penalty time O
poi	BITSTRING [1]		Power offset indicator M
po	BITSTRING [2]		Power offset O
gprsi	BITSTRING [1]		GPRS supported M
racolour	BITSTRING [3]		routing area colour code O
si13p	BITSTRING[1]		SI13 position O
spb	BITSTRING [1..7]		spare bits, O
spd	OCTETSTRING [1..9]		'2B..2B'O padding, O

Detailed Comments : The info element has a maximun length of 10 octets.
1. Spare bits spb are constituted of spare L (GSM 04.07, B.1.2.2). The padding sequence is '0010 1011'B. The bit value is corresponding to the padding spare bit for that position.
2. Spare padding octets are a sequence of '2B'O. The total spare padding consists of the fields spb and spd (GSM 04.07, B.1.3).

Structured Type Definition

Type Name : SI6RO

Encoding Variation :

Comments : GSM 04.08, 10.5.2.35a

Element Name	Type Definition	Field Encoding	Comments
pchnchi	B_1		PCH/NCH info indicator M
pchrestr	B_1		pch restructuring O
nln_saccch	B_2		notification list number (SACCH), O
cpi	B_1		call priority indicator
call_priority	B_3		optional
nlns	B_1		optional
vbs_vgcsi	B_1		VBS/VGCS indicator
inband_notification	B_1		notification on FACCH M
inband_paging	B_1		paging thro notification/facch M
spb	BITSTRING [1..7]		spare bits, optional
spd	OCTETSTRING [1..6]		'2B..2B'O padding, optional

Detailed Comments : 1. Spare bits spb are constituted of spare L (GSM 04.07, B.1.2.2). The padding sequence is '0010 1011'B. The bit value is corresponding to the padding spare bit for that position.
2. Spare padding octets are a sequence of '2B'O. The total spare padding consists of the fields spb and spd (GSM 04.07, B.1.3).

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
gprsi	BITSTRING [1]		GPRS supported
racolour	BITSTRING [3]		routing area colour code
si13p	BITSTRING[1]		SI13 position O
spb	BITSTRING [1..7]		spare bits, O
spd	OCTETSTRING [1..19]		'2B..2B'O padding, O

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 : 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 : 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 : 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.

Structured Type Definition

Type Name : VGCSTMI
Encoding Variation :
Comments : VGCS target mode indication
GSM 04.08, 10.5.2.42c

Element Name	Type Definition	Field Encoding	Comments
iei	IEI_8		’01’O
tmod	B_2		target mode
gckn	B_4		group ciphering key number
spr	SP2B		spare bits

Detailed Comments :

ASN.1 Type Definition

Type Name : Component
Encoding Variation :
Comments : ASN1_Encoding: BER

Continued on next page

Continued from previous page

ASN.1 Type Definition

Type Definition

```
CHOICE {
  invoke     [1]  IMPLICIT      Invoke_TCAP,
  rResult    [2]   IMPLICIT     RResult_TCAP,
  rError     [3]    IMPLICIT    RError_TCAP,
  reject     [4]   IMPLICIT     RejectComponent
}
Invoke_TCAP ::=  SEQUENCE
{                       invokeID                          InvokeIDType,
                        linkedID                          [0] IMPLICIT InvokeIDType  OPTIONAL,
                        operation                         Operation,
                        arg_registerSS                     InvArg_RegisterSS  OPTIONAL,
                        arg_eraseSS                        SS_ForBS_Code  OPTIONAL,
                        arg_activateSS                    SS_ForBS_Code  OPTIONAL,
                        arg_deactivateSS                   SS_ForBS_Code  OPTIONAL,
                        arg_interrogateSS                  SS_ForBS_Code  OPTIONAL,
                        arg_notifySS                       InvArg_NotifySS  OPTIONAL,
                        arg_registerPassword                SS_Code  OPTIONAL,
                        arg_getPassword                    InvArg_GetPassword  OPTIONAL,
                        arg_processUnstructuredSSData       SS_UserData  OPTIONAL,
                        arg_processUnstructuredSSRequest   USSD_Arg  OPTIONAL,
                        arg_unstructuredSSRequest          USSD_Arg  OPTIONAL,
                        arg_unstructuredSSNotify           USSD_Arg  OPTIONAL,
                        arg_forwardCUGInfo                 InvArg_ForwardCUGInfo  OPTIONAL,
                        arg_forwardChargeAdvice            InvArg_ForwardChargeAdvice  OPTIONAL
}
RResult_TCAP  ::=  SEQUENCE
{     invokeID            InvokeIDType,
      valueAndResult   SEQUENCE {
                        operation                    Operation,
                        registerSS_rr                      SS_Info        OPTIONAL,
                        eraseSS_rr                         SS_Info        OPTIONAL,
                        activateSS_rr                      SS_Info        OPTIONAL,
                        deactivateSS_rr                    SS_Info        OPTIONAL,
                        interrogateSS_rr                   RRArg_InterrogateSS   OPTIONAL,
                        registerPassword_rr                Password               OPTIONAL,
                        getPassword_rr                     Password               OPTIONAL,
                        processUnstructuredSSData_rr       SS_UserData            OPTIONAL,
                        processUnstructuredSSRequest_rr   USSD_Res               OPTIONAL,
                        unstructuredSSRequest_rr           USSD_Res               OPTIONAL
                       } OPTIONAL
}
RError_TCAP   ::=  SEQUENCE
{     invokeID InvokeIDType,
      error_value               ErrorLocalValues (
                                  absentSubscriber
                                  | bearerServiceNotProvisioned
                                  | dataMissing
                                  | facilityNotSupported
                                  | illegalEquipment
                                  | illegalSS_Operation
                                  | illegalSubscriber
                                  | longTermDenial
                                  | maxNumberOfMPTY_ParticipantsExceeded
                                  | negativePW_Check
                                  | numberOfPW_AttemptsViolation
                                  | resourcesNotAvailable
                                  | shortTermDenial
                                  | ss_NotAvailable
                                  | teleserviceNotProvisioned
                                  | unexpectedDataValue
                                  | unknownAlphabet
                                  | ussd_Busy
)     OPTIONAL,
      callBarredErr                    ErrorLocalValues ( callBarred )    OPTIONAL,
      par_callBarredErr                CallBarringCause    OPTIONAL,
      ss_IncompatibilityErr            ErrorLocalValues ( ss_Incompatibility )    OPTIONAL,
      par_ss_IncompatibilityErr        SS_IncompatibilityCause    OPTIONAL,
      systemFailureErr                 ErrorLocalValues ( systemFailure )    OPTIONAL,
      par_systemFailureErr             NetworkResource    OPTIONAL,
      ss_ErrorStatusErr                ErrorLocalValues ( ss_ErrorStatus )  OPTIONAL,
      par_ErrorStatusErr               SS_Status     OPTIONAL,
```

Continued from previous page

ASN.1 Type Definition

Detailed Comments :

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 : InvArg_ForwardChargeAdvice

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
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 }
}
```

Detailed Comments : Timer T_aoc = 1 − 40 s

ASN.1 Type Definition

Type Name : InvArg_ForwardCUGInfo

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
SEQUENCE{
                        cug_Index           [0]  IMPLICIT CUG_Index OPTIONAL,
                        suppressPrefCUG     [1]  IMPLICIT NULL      OPTIONAL,
                        suppressOA          [2]  IMPLICIT NULL      OPTIONAL
 }
```

Detailed Comments :

ASN.1 Type Definition

Type Name : InvArg_GetPassword

Encoding Variation :

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

Type Definition

```
ENUMERATED {
  enterPW  (0),
  enterNewPW  (1),
  enterNewPW_Again  (2)
}
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : InvArg_NotifySS

Encoding Variation :

Comments : GSM 04.80, 4.2

Type Definition

```
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 ,
                ect_Indicator           [19]   IMPLICIT    ECT_Indicator        OPTIONAL ,
                nameIndicator           [20]   IMPLICIT    NameIndicator        OPTIONAL ,
                ccbs_feature            [21]   IMPLICIT    CCBS_Feature         OPTIONAL ,
                alertingPattern         [22]   IMPLICIT    AlertingPattern      OPTIONAL
}
```

Detailed Comments :

ASN.1 Type Definition

Type Name : InvArg_RegisterSS

Encoding Variation :

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

Type Definition

```
SEQUENCE{
        ss_Code                 SS_Code,
        basicService            BasicServiceCode        OPTIONAL,
        forwardedToNumber       [4] IMPLICIT AddressString      OPTIONAL,
        forwardedToSubaddress   [6] IMPLICIT ISDN_SubaddressString      OPTIONAL,
        noReplyConditionTime    [5] IMPLICIT NoReplyConditionTime       OPTIONAL ,
        defaultPriority         [7]  IMPLICIT DefaultPriority  OPTIONAL
  }
```

Detailed Comments : Timer m

ASN.1 Type Definition

Type Name : RRArg_InterrogateSS

Encoding Variation :

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

Type Definition

```
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 }
}
```

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 : 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 : AlertingPattern
Encoding Variation :
Comments : GSM 04 80

Type Definition

OCTET STRING (SIZE(1))

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 : CCBS_Feature

Encoding Variation :

Comments : GSM 04 80 annex A

Type Definition

```
SEQUENCE {
ccbs_index  [0] IMPLICIT INTEGER (1..5) OPTIONAL,
b_subscriberNumber  [1]  IMPLICIT OCTET STRING (SIZE (1..20)) (SIZE (1..9)) OPTIONAL,
b_subscriberSubaddress [2] IMPLICIT OCTET STRING (SIZE (1..21)) OPTIONAL,
basicsericeGroup [3] BasicServiceCode OPTIONAL
}
```

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 : DefaultPriority
Encoding Variation :
Comments : GSM 04 80 , annex A

Type Definition

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

Detailed Comments :

ASN.1 Type Definition

Type Name : ECT_Indicator
Encoding Variation :
Comments : GSM 04 80, 4.4.2

Type Definition

```
SEQUENCE {
    ect_CallState    [0]  ENUMERATED  { alerting(0),   active (1)},
    rdn              [1]  RDN  OPTIONAL
}
```

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),
  shortTermDenial                        (29),
  longTermDenial                         (30)
 }
```

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 : Name
Encoding Variation :
Comments : GSM 04 80 4.4.2

Type Definition

```
CHOICE {
namePresentationAllowed     [0]  NameSet,
presentationRestricted      [1]  NULL,
nameUnavailable             [2]  NULL,
namePresentationRestricted  [3]  NameSet
}
```

Detailed Comments :

ASN.1 Type Definition

Type Name : NameSet

Encoding Variation :

Comments : GSM 04 80 4.4.2

Type Definition

```
SEQUENCE {
dataCodingScheme   [0]  USSD_DataCodingScheme,
lengthInCharacters   [1]   INTEGER,
nameString   [2]   USSD_String
}
```

Detailed Comments :

ASN.1 Type Definition

Type Name : NameIndicator

Encoding Variation :

Comments : GSM 04 80 4.4.2

Type Definition

```
SEQUENCE {
callingName    [0] Name OPTIONAL
}
```

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 : RDN

Encoding Variation :

Comments : GSM 04 80 4.4.2

Type Definition

```
CHOICE {
presentationAlowedAddress               [0]  RemotePartyNumber,
presentationRestricted                  [1]  NULL,
numberNotAvailableDueToInterworking     [2]  NULL,
presentationRestrictedAddress           [3]  RemotePartyNumber
}
```

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 : RemotePartyNumber

Encoding Variation :

Comments : GSM 04 80 4.4.2

Type Definition

```
SEQUENCE {
    partyNumber             [0]  ISDN_AddressString,
    partyNumberSubaddress   [1]  ISDN_SubaddressString
}
```

Detailed Comments :

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,
     defaultPriority           DefaultPriority  OPTIONAL,
     ccbs_Feature              [5]   IMPLICIT CCBS_Feature
}
```

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,
      alertingPattern         AlertingPattern      OPTIONAL
 }
```

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_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_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_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_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_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_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_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_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	**:** 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	: 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_ChangeRFOfCell(cellid1 : CellID; bspwr1 : INTEGER)

Result Type : BOOLEAN

Comments :

Description

This TSO is to change the RF level of the cell 'cellid1' to the RF level 'bspwr1', then the TSO returns TRUE.
The major purpose of the TSO is to force the MS to reselect a new cell according to the GSM 05.08, section 6.6.2. The following example shows the use of the TSO.
The default cell selection parameters in the ATS are set as:
Cell reselect hysteresis: 12db
MX TXPWR MX CCH: minimu level (0)
ACS: no additional cell selection parameter (0)
RXLEV ACCESS MIN: minimum (0).

Therefore the path loss criterion parameter C1 equals to the reselection criterion parameter C2. In the whole ATS the C1 and C2 keeps greater than 10 (e.g. TC_26_7_4_2_3) for a cell not suitable. The MS received level average from a not suitable cell keeps also greater than 10 (−100 dBm) corresponding to the reference sensitivity level of DCS 1800 MS.

For GSM arfcn = 20, F = 941 MHz , the fields strengths (E) on tester integral antenna:
E (dB microV/m) >= (−110 + 10) (dBm) + 20log(941) + 77.2 = 36.7,
For DCS arfcn = 590, F = (−110 + 10) (dBm) + 20log(1820.8) + 77.2 = 42.4 (the formula see GSM 05.05, section 5 and GSM 05.08, section 8.1.4).

In order to keep C1, C2 greater 10, a reasonable field strength making a cell not suitable
is set to C_E_notsuitable = 45 in comparison with the default field strength C_E_default = 63. Considering that the cell recelect hysteresis is set to 12 db, the field strength of a cell being reselected need to be set to greater than 57. In the ATS the default value C_E_default (63) and C_E_notsuitable (45) are used for this TSO to force MS making cell reselection.

C1 needs to be less than zero to force the MS reselect a better cell. For GSM, E <= 26.7, for DCS, E<=32.4. The value C_E_notsuitable_sameLAI = 24 applies to the TSO to force the MS making cell reselection.

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 re−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_CorruptPgBlks (pch : LOGICCH; pgg : PGG; blks : INTEGER)
Result Type : BOOLEAN
Comments :
Description
This TSO corrupts the next 'blks' number of paging blocks of the paging channel 'pch' for paging group 'pgg'. The major purpose of the TSO is to force the MS to reselect a new cell by forcing the downlink signalling failure counter DSC to be < 0. The initial DSC value for the MS shall be initialised to a value equal to the nearest integer to 90/N where N is the BS_PA_MFRMS parameter for the cell the MS is camped on (see GSM 05.02). The DSC counter is decreased by 4 whenever it cannot decode a message in its paging subchannel.
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_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 acknowledge the ‘i‘th occurrence 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 channel ’ch’,
− or a data link SABM message is received on the channel ’ch’.

Detailed Comments :

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_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_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_SetTSCEqualToBCC(cell: CellID; bcc: BCC)

Result Type : BOOLEAN

Comments :

Description

To set the Training Sequence Code value equal to the passed BCC value for the passed cell. The time taken to complete this operation should be such that broadcasting on the control channel for the cell is unaffected.

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 and also in IdleUpdated.

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 : OM_StopVaryRFOfCell(ch : LOGICCH)

Result Type : BOOLEAN

Comments :

Description

Detailed Comments :

Test Suite Operation Definition

Operation Name : OM_VaryRFOfCell(ch : LOGICCH; delay1, bspwr1, delay2, bspwr2 : INTEGER)

Result Type : BOOLEAN

Comments :

Description

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_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_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_SetCellTestMode
Result Type : BOOLEAN
Comments :

Description

Instructs operator to switch MS to 'Cell Test Mode', used in test case 20.7.

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_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_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_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. 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.

If the call is a MT speech call: after sending the CONNECT message the through connecting of the TCH in both directions is checked by verifying that after transmission of the first L2 frame containing the complete CONNECT message, the MS is sending appropriate speech or data frames whenever it does not have to transmit or acknowledge an I frame on layer 2 of the FACCH.

If the call is a MT data call: after receipt of a subsequent CONNECT ACKNOWLEDGE message the through connecting of the TCH in both directions is checked by verifying that there is a point in time after transmission of the first L2 frame containing the complete CONNECT ACKNOWLEDGE message, where the MS is sending appropriate speech or data frames whenever it does not have to transmit or acknowledge an I frame on layer 2 of the FACCH.

If the call is a MO call it 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 does not have to transmit or acknowledge an I frame on layer 2 of the FACCH.

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 Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPC_R96	BOOLEAN	PICS	release 96 supported
TSPC_R97	BOOLEAN	PICS	release 97 supported
TSPC_R98	BOOLEAN	PICS	release 98 supported
TSPC_R99	BOOLEAN	PICS	release 99 supported
TSPC_R00	BOOLEAN	PICS	release 2000 supported
TSPC_SvcOnTCH	BOOLEAN	PICS Table A.25 Item 22	at least one service on traffic channel supported
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_DualRate	BOOLEAN	PICS Table A.25 Item 23	dual rate channel types supported
TSPC_HalfRateData	BOOLEAN	PICS Table A.25 Item 6	half rate data supported
TSPC_HalfRateSpeech	BOOLEAN	PICS Table A.25 Item 3	half rate speech mode supported
TSPC_NonCallSS	BOOLEAN	PICS Table A.25 Item 30	non call related supplementary service supported
TSPC_MOsvc	BOOLEAN	PICS Table A.25 Item 20	at least one MO circuit switched basic service 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_TS22	BOOLEAN	PICS Table A.3 Item 4	SMS MO/PP 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

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
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
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_RGSM	BOOLEAN		R−GSM band supported
TSPX_AltNb	BOOLEAN	PIXIT	use alternative neighbour cells description. default : FALSE
TSPX_TE_stopbit	B_1	PIXIT	TE conf. number stop bits. default value : ’0’B, 1 bit
TSPX_TE_databit	B_1	PIXIT	TE conf. number data bits. default value : ’1’B, 8 bits
TSPX_TE_FLCT	FLWCNTL	PIXIT	TE flow control. default : 0, outband flow control
TSPX_TE_parity	B_3	PIXIT	TE conf. parity. default : ’001’B, no parity
TSPX_BS_21_itc1	B_3	PIXIT	itc value supported for BS21. default value : ’001’B, UDI
TSPX_BS_21_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS21. default value : FALSE, only one itc
TSPX_BS_21_T_NT	BOOLEAN	PIXIT	both T and NT supported by BS21. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_21_ce	B_2	PIXIT	ce 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	sacp 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_21_itc2	B_3	PIXIT	other itc value supported by BS21, it shall be differenet from the previous one if the answer to TSPX_BS_21_more_itc is TRUE. default value :'001'B, UDI
TSPX_BS_22_itc1	B_3	PIXIT	itc value supported by BS22. default value : '010'B, 3.1kHz audio, exPLMN
TSPX_BS_22_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS22. default value : FALSE, only one itc
TSPX_BS_22_T_NT	BOOLEAN	PIXIT	both T and NT supported by BS22. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_22_ce	B_2	PIXIT	ce 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	sacp value used in MO call. default value : '001'B, I440/450
TSPX_BS_22_itc2	B_3	PIXIT	other itc value supported by BS22, it shall be differenet 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	itc value supported by BS23. default value : '010'B, 3.1kHz audio, exPLMN
TSPX_BS_23_T_NT	BOOLEAN	PIXIT	both T and NT supported for BS23. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_23_ce	B_2	PIXIT	ce value used in MO call. default value : '00'B, transparent
TSPX_BS_23_sacp	B_3	PIXIT	sacp value used in MO call. default value : '001'B, I440/450
TSPX_BS_24_itc1	B_3	PIXIT	itc value supported by BS24. default value : '010'B, 3.1kHz audio, exPLMN
TSPX_BS_24_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS24. 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_24_T_NT	BOOLEAN	PIXIT	both T and NT supported by BS24. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_24_ce	B_2	PIXIT	ce 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	sacp value used in MO call. default value : '001'B, I440/450
TSPX_BS_24_itc2	B_3	PIXIT	other itc value supported by BS24, it shall be differenet 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	itc value supported by BS25. default value : '001'B, UDI
TSPX_BS_25_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS25. default value : FALSE, only one itc
TSPX_BS_25_T_NT	BOOLEAN	PIXIT	both T and NT supported by BS25. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_25_ce	B_2	PIXIT	ce 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	sacp value used in MO call. default value : '001'B, I440/450
TSPX_BS_25_itc2	B_3	PIXIT	other itc value supported by BS25, it shall be differenet from the previous one if the answer to TSPX_BS_25_more_itc is TRUE. default value : '001'B, UDI
TSPX_BS_26_itc1	B_3	PIXIT	itc value supported by BS26. default value : '010'B, 3.1kHz audio, exPLMN
TSPX_BS_26_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS26. 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_26_T_NT	BOOLEAN	PIXIT	both T and NT supported by BS26. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_26_ce	B_2	PIXIT	ce 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
TSPX_BS_26_sacp	B_3	PIXIT	sacp value used in MO call. default value : ’001’B, I440/450
TSPX_BS_26_itc2	B_3	PIXIT	other itc value supported by BS26, it shall be differenet 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	more than one itc value supported by BS31. default value : FALSE, only one itc
TSPX_BS_31_more_sacp	BOOLEAN	PIXIT	more than one sacp value supported by BS31. default value : FALSE, only one sacp
TSPX_BS_31_itc1	B_3	PIXIT	itc value supported by BS31. default value : ’001’B, UDI
TSPX_BS_31_sacp1	B_3	PIXIT	sacp value supported by BS31. default value : ’001’B, I440/450
TSPX_BS_31_itc2	B_3	PIXIT	other itc value supported by BS31, it shall be differenet from the previous one if the answer to TSPX_BS_31_more_itc is TRUE. default value : ’001’B, UDI
TSPX_BS_31_sacp2	B_3	PIXIT	other sacp value supported by BS31, it shall be differenet from the previous one if the answer to TSPX_BS_31_more_sacp is TRUE. default value : ’001’B, I440/450
TSPX_BS_32_itc1	B_3	PIXIT	itc value supported by BS32. default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_32_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS32. 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_32_more_sacp	BOOLEAN	PIXIT	more than one sacp value supported by BS32. default value : FALSE, only one sacp
TSPX_BS_32_sacp1	B_3	PIXIT	sacp value supported by BS32. default value : ’110’B, X.32
TSPX_BS_32_X32_T_NT	BOOLEAN	PIXIT	BS 32 both T and NT supported for X32 3.1kHz. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_32_X32_ce	B_2	PIXIT	if the answer to TSPX_BS_32_X32_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’01’B, non transparent
TSPX_BS_32_itc2	B_3	PIXIT	other itc value supported by BS32, it shall be differenet 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	other sacp value supported by BS32, it shall be differenet from the previous one if the answer to TSPX_BS_32_more_sacp is TRUE. default value : ’110’B, X.32
TSPX_BS_33_itc1	B_3	PIXIT	itc value supported by BS33. default value : ’001’B, UDI
TSPX_BS_33_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS33. default value : FALSE, only one itc
TSPX_BS_33_more_sacp	BOOLEAN	PIXIT	more than one sacp value supported by BS33. default value : FALSE, only one sacp
TSPX_BS_33_sacp1	B_3	PIXIT	sacp value supported by BS33. default value : ’010’B, X.21
TSPX_BS_33_X32_T_NT	BOOLEAN	PIXIT	BS 33 both T and NT supported for X32 3.1kHz. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_33_X32_ce	B_2	PIXIT	if the answer to TSPX_BS_33_X32_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_33_itc2	B_3	PIXIT	other itc value supported by BS32, it shall be differenet from the previous one if the answer to TSPX_BS_33_more_itc is TRUE. default value : ’001’B, UDI
TSPX_BS_33_sacp2	B_3	PIXIT	other sacp value supported by BS33, it shall be differenet from the previous one if the answer to TSPX_BS_33_more_sacp is TRUE. default value : ’010’B, X.21
TSPX_BS_34_itc1	B_3	PIXIT	itc value supported by BS34. default value : ’010’B, 3.1kHz audio, exPLMN
TSPX_BS_34_more_itc	BOOLEAN	PIXIT	more than one itc value supported by BS34. default value : FALSE, only one itc
TSPX_BS_34_more_sacp	BOOLEAN	PIXIT	more than one sacp value supported by BS34. default value : FALSE, only one sacp
TSPX_BS_34_sacp1	B_3	PIXIT	sacp value supported by BS34. default value : ’001’B, I440/450
TSPX_BS_34_X32_T_NT	BOOLEAN	PIXIT	BS 34 both T and NT supported for X32 3.1kHz. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_34_X32_ce	B_2	PIXIT	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	other itc value supported by BS34, it shall be differenet 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	other sacp value supported by BS34, it shall be differenet from the previous one if the answer to TSPX_BS_34_more_sacp is TRUE. default value : ’001’B, I440/450
TSPX_BS_41_T_NT	BOOLEAN	PIXIT	BS41 both T and NT supported . default value : FALSE, not support both

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_41_ce	B_2	PIXIT	ce value used in BS41 MO call. default value : ’00’B, transparent
TSPX_BS_42_T_NT	BOOLEAN	PIXIT	BS42 both T and NT supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_42_ce	B_2	PIXIT	ce value used in BS42 MO call. default value : ’010’B, non transparent
TSPX_BS_43_T_NT	BOOLEAN	PIXIT	BS43 both T and NT supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_43_ce	B_2	PIXIT	ce value used in BS43 MO call. default value : ’00’B, transparent
TSPX_BS_44_T_NT	BOOLEAN	PIXIT	BS44 both T and NT supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_44_ce	B_2	PIXIT	ce value used in BS44 MO call. default value : ’00’B, transparent
TSPX_BS_45_T_NT	BOOLEAN	PIXIT	BS45 both T and NT supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_45_ce	B_2	PIXIT	ce value used in BS45 MO call. default value : ’01’B, non transparent
TSPX_BS_46_T_NT	BOOLEAN	PIXIT	BS46 both T and NT supported . default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_46_ce	B_2	PIXIT	ce value used in BS46 MO call. default value : ’00’B, transparent
TSPX_BS_61_S	BOOLEAN	PIXIT	synchronous data supported by BS 61 . default value : TRUE, synchronous mode
TSPX_BS_61_S_more_ur	BOOLEAN	PIXIT	more than one user rate for synchronous data service supported by BS61. default value : FALSE, only ine user rate
TSPX_BS_61_S_ur1	B_4	PIXIT	usrer rate value supported for synchronous data service of BS61. default value : ’0100’B, 4.8 kbit/s

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_BS_61_S_ur2	B_4	PIXIT	another usrer rate value supported for synchronous data service of BS61. 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	asynchronous data service supported by BS61 . default value : TRUE, asynchronous
TSPX_BS_61_A_ur1	B_4	PIXIT	usrer rate value supported for asynchronous data service of BS61. default value : ’0101’B, 9.6 kbit/s
TSPX_BS_61_A_ur1_T_NT	BOOLEAN	PIXIT	both T and NT supported for the rate TSPX_BS_61_A_ur1 of BS61. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_61_A_ur1_ce	B_2	PIXIT	if the answer to TSPX_BS_61_A_ur1_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : ’00’B, transparent
TSPX_BS_61_A_more_ur	BOOLEAN	PIXIT	more than one user rate for asynchronous data service supported by BS61. default value : FALSE, only one user rate
TSPX_BS_61_A_ur2	B_4	PIXIT	another usrer rate value supported for asynchronous data service of BS61. 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	both T and NT supported for the rate TSPX_BS_61_A_ur2 of BS61. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_81_S	BOOLEAN	PIXIT	synchronous data supported by BS 81. default value : TRUE, synchronous mode
TSPX_BS_81_S_more_ur	BOOLEAN	PIXIT	more than one user rate for synchronous data service supported by BS81. 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_BS_81_S_ur1	B_4	PIXIT	usrer rate value supported for synchronous data service of BS81. default value : '0101'B, 9.6 kbit/s
TSPX_BS_81_S_ur2	B_4	PIXIT	another usrer rate value supported for synchronous data service of BS81. 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
TSPX_BS_81_A	BOOLEAN	PIXIT	asynchronous data service supported by BS81. default value : TRUE, asynchronous mode
TSPX_BS_81_A_ur1	B_4	PIXIT	usrer rate value supported for asynchronous data service of BS81. default value : '0100'B, 4.8 kbit/s
TSPX_BS_81_A_ur1_T_NT	BOOLEAN	PIXIT	both T and NT supported for the rate TSPX_BS_81_A_ur1 of BS81. default value : FALSE, not support both transparent and non transparent modes
TSPX_BS_81_A_ur1_ce	B_2	PIXIT	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 : '01'B, non transparent
TSPX_BS_81_A_more_ur	BOOLEAN	PIXIT	more than one user rate for asynchronous data service supported by BS81. default value : FALSE, only one user rate
TSPX_BS_81_A_ur2	B_4	PIXIT	another usrer rate value supported for asynchronous data service of BS81. 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	both T and NT supported for the rate TSPX_BS_81_A_ur2 of BS81. 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	both T and NT supported for TS61(fax). default value : FALSE, not support both transparent and non transparent modes
TSPX_TS_61_ce	B_2	PIXIT	if the answer to TSPX_TS_61_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : '01'B, non transparent
TSPX_TS_61_ur1	B_4	PIXIT	user rate for TS61(fax). default value : '0101'B, 9.6 kbit/s
TSPX_TS_61_more_ur	BOOLEAN	PIXIT	more than one user rate supported for TS61. default value : FALSE, only one user rate
TSPX_TS_61_ur2	B_4	PIXIT	another user rate for TS61, 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	both T and NT supported for TS62. default value : FALSE, not support both transparent and non transparent modes
TSPX_TS_62_ce	B_2	PIXIT	if the answer to TSPX_TS_62_T_NT is FALSE, the manufacturer must precise which mode is supported by the MS. default value : '00'B, transparent
TSPX_TS_62_ur1	B_4	PIXIT	user rate for TS62. default value : '0100'B, 4.8 kbit/s
TSPX_TS_62_more_ur	BOOLEAN	PIXIT	more than one user rate supported by TS62. default value : FALSE, only one user rate
TSPX_TS_62_ur2	B_4	PIXIT	another user rate for TS62, 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	call control capabilities: value '1'B means MS supports DTMF
TSPX_CKSNDef	CKSN	PIXIT	default cipher key sequence number
TSPX_CphAlgDef	CPHALG	PIXIT	default ciphering algorithm
TSPX_NoOfHoAccA	INTEGER	PIXIT	number of handover access bursts, value range 10 − 20

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_NoOfHoAccI	INTEGER	PIXIT	number of handover access bursts, value range 2 − 5
TSPX_HoRefA	HORF	PIXIT	Hand over reference, GSM 04.08, 10.5.2.15
TSPX_HSN	INTEGER	PIXIT	Hopping sequence number value range: 0 − 63
TSPX_IMSI	HEXSTRING	PIXIT	IMSI of the MS
TSPX_Ki	BITSTRING	PIXIT	default authentication key used in testing
TSPX_MAIO	INTEGER	PIXIT	mobile allocation index offset, value range: 0 − 63
TSPX_PwrlvlA	INTEGER	PIXIT	power level, value between 2 to 15
TSPX_RANDDef	RAND	PIXIT	default challenge RAND
TSPX_SDCCH4SubA	B_2	PIXIT	TDMA offset of SDCCH/4 subchannel
TSPX_SDCCH4SubDef	B_2	PIXIT	TDMA offset of default SDCCH/4 subchannel
TSPX_SDCCH8SubA	B_3	PIXIT	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubB	B_3	PIXIT	TDMA offset of SDCCH/8 subchannel
TSPX_SDCCH8SubDef	B_3	PIXIT	TDMA offset of default SDCCH/8 subchannel
TSPX_TmSltA	SN	PIXIT	time slot
TSPX_TmSltC	SN	PIXIT	time slot
TSPX_TmSltG	SN	PIXIT	time slot
TSPX_TmSltDef	SN	PIXIT	default time slot
TSPX_TmSltNotZero	SN	PIXIT	time slot, arbitrarily value, but not zero.
TSPX_TmSltNotZero1	SN	PIXIT	time slot, arbitrarily value, but not zero and not TSPX_TmSltNotZero
TSPX_TscDef	TSC	PIXIT	default TSC
TSPX_TCHcarrierA_ho	INTEGER	PIXIT	the value can be choosen arbitrarily from cell allocation of cell A (GSM), but not BCCH carrier .possible value is : 10, 17, 26, 34, 42, 45, 46, 52, 59, 66, 73, 74, 75, 76, 108, 114
TSPX_TCHHSubDef	B_1	PIXIT	TDMA offset of default half rate subchannel
TSPX_TMSI	TMSI_V	PIXIT	TMSI of the MS used in test
TSPX_ChMod1	CHMOD_VAL	PIXIT	channel mode for TC_26_6_13_1, any value for the channel type of TSPX_Chtp1
TSPX_MOBscSvcD	SERVICES	PIXIT	any supported MO basic service
TSPX_MTBscSvcA	SERVICES	PIXIT	any supported MT basic service

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
TSPX_MTBscSvcD	SERVICES	PIXIT	any supported MT basic service
TSPX_MOChRateD	RATE	PIXIT	channel rate for TSPX_MOBscSvcD, default value : C_Full
TSPX_MTChRateA	RATE	PIXIT	channel rate for TSPX_MTBscSvcA, default value : C_Full
TSPX_MTChRateD	RATE	PIXIT	channel rate for TSPX_MTBscSvcD, default value : C_Full
TSPX_EmgCallRate	RATE	PIXIT	Rate for the basic service supported for MO emergency calls
TSPX_aTav	INTEGER	PIXIT	averaging period a*Tav in milliseconds
TSPX_bTav	INTEGER	PIXIT	averaging period b*Tav in milliseconds
Detailed Comments :			

Test Case Selection Expression Definitions		
Expression Name	**Selection Expression**	**Comments**
SelExpr_96	TSPC_R96 OR TSPC_R97 OR TSPC_R98 OR TSPC_R99 OR TSPC_R00	Relase 96 supported
SelExpr_0104	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 OR TSPC_Serv_TS61 OR TSPC_Serv_TS62	at least one data service supported
SelExpr_0301	TSPC_Serv_TS11 OR TSPC_Serv_TS12 OR SelExpr_0104	CC protocol (TSPC_CC) supported if supporting one of the data or speech services
SelExpr_1000R	TSPC_RGSM AND SelExpr_96	R− band supported
SelExpr_1001	SelExpr_0301	CC protocol for at least one BC supported
SelExpr_1002	TSPC_Serv_TS11 OR TSPC_Serv_TS61 OR TSPC_Serv_TS62	telephony or alternate speech/data or Teleservice automatic G3 fax supported
SelExpr_1003	TSPC_Serv_TS12	emergency call supported
Detailed Comments :		

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_arfcnA	INTEGER	20	ARFCN 20
C_arfcn_2	INTEGER	2	ARFCN 2
C_arfcn_965	INTEGER	965	ARFCN 965
C_arfcn_972	INTEGER	972	ARFCN 972
C_arfcn_990	INTEGER	990	ARFCN 990
C_arfcn_34	INTEGER	34	ARFCN 34
C_arfcnC	INTEGER	30	ARFCN 30
C_arfcn_tchA	INTEGER	30	ARFCN 30
C_arfcn_tchB	INTEGER	114	ARFCN 114
C_arfcnEgsm	INTEGER	990	ARFCN 990
C_arfcnEgsm_iacmd	INTEGER	40	ARFCN 40
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_1	NCC	’001’B	Network color code value ’001’B
C_NCCP_2	NCCP	’02’O	Network color code permitted value ’02’O
C_BCC	BCC	’101’B	Base Station color code
C_BCC_1	BCC	’001’B	Base Station color code value 1
C_BCC_3	BCC	’011’B	Base Station color code value 3
C_BCC_7	BCC	’111’B	Base Station color code value 7
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_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_cchd_r_407	OCTETSTRING	’8DC007F43F7C42FFFF10000000000000’O	f−list for cellchdescr using of 128 format. The coded set is { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972 }. Length = 16
C_cchd_r_408	OCTETSTRING	’8A0A1CFD3EF4610E2FFFFA0000000000’O	f−list for cellchdescr using of 256 format. The coded set is {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114,115} Length = 16

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_cchd_r_409	OCTETSTRING	’89E001FF40FF7F0453FFFFD010000000’O	f−list for cellchdescr using of 512 format. The coded set is { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 978, 990 }. Length = 16
C_cchd_r_410	OCTETSTRING	’841E97893F798C03B961C40000000000’O	f−list for cellchdescr using of 1024 format. (0,30, 40, 66, 80, 520, 955, 964, 969, 972, 990, 1020) Length = 16
C_cchd_r_411	OCTETSTRING	’8FE07FF8000000000000000000000000’O	f−list for cellchdescr using of variable bit format. The coded set is { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972 }. Length = 16
C_cchd_r_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} Length = 16
C_cchd_r_414	OCTETSTRING	’000000000000002000020000020080000’O	f−list for cellchdescr using of bit map 0 format. Length = 16 f_list: 20, 30, 50 ,70
C_cchd_r_415	OCTETSTRING	’87FCE8117F798483BDF6000000000000’O	f−list for cellchdescr using range 1024 format. The coded set is { 0, 30, 40, 66, 80, 964, 969, 972, 990, 1020 }. Length = 16
C_cchd_r_416	OCTETSTRING	’89E001FF60FF7F040BFFFFFC10000000’O	f−list for cellchdescr using range 512 format. The coded set is { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972 }. Length = 16
C_ChMod_s	CHMOD_VAL	’00000000’B	Mode : signalling only.
C_ChMod_r	CHMOD_VAL	’00000001’B	Mode : speech full or half rate, version 1.
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_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

Continued on next page

Continued from previous page

Test Suite Constant Declarations

Constant Name	Type	Value	Comments
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_E_default	INTEGER	63	Default field strength
C_E_neighbourdefault	INTEGER	53	Default field strength of neighbouriong cells.
C_E_notsuitable	INTEGER	−10	Field strength of a not suitable cell for reselection
C_flist_r_401	OCTETSTRING	’8DE208F400’O	f’ list { 964, 969, 972} range 128 format, length = 5
C_fshlist_r_401	OCTETSTRING	’8DE208F40000000000’O	f’short list {964, 969, 972}, range 128 format, length = 9
C_flist_r_402	OCTETSTRING	’8A24867B0BE0’O	f’ list { 73, 76, 80, 85, 90 }, range 256 format, length = 6
C_flist_r_403	OCTETSTRING	’89DE053C0100’O	f’list { 956, 960, 976, 980 }, range 512 format, length = 6
C_flist_r_404	OCTETSTRING	’83FCE80AFEFB0900’O	f’ list { 30, 40, 969, 972, 990, 1020 }, range 1024 format, length = 6
C_fshlist_r_404	OCTETSTRING	’83FCE80AFEFB090000’O	f’ short list { 30, 40, 969, 972, 990, 1020 }, range 1024 format, length =9
C_flist_r_405	OCTETSTRING	’8FDE092480’O	f’ list { 956, 960, 963, 966, 969, 972 }, length = 5, v−bitmap.
C_flist_r_406	OCTETSTRING	’00000000000000002000008000080000’O	f’ list {20, 40, 66}, bitmap 0 format, length = 16
C_flist_r_407	OCTETSTRING	’8DC007F43F7C42FFFF10000000000000’O	f’ list { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972 }, range 128 format, length = 16
C_flist_r_408	OCTETSTRING	’8A0A1CFD3EF4610E2FFFFA0000’O	f’ list {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114, 115}, range 256 format, length = 13

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_flist_r_409	OCTETSTRING	’89E001FF40FF7F0453FFFFD010000000’O	f’ list { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 978, 990 }, range 512 format, length = 16
C_flist_r_410	OCTETSTRING	’841E97893F798C03B961C4’O	1024 format. (0,30, 40, 66, 80, 520, 955, 964, 969, 972, 990, 1020) Length = 11
C_flist_r_411	OCTETSTRING	’8FE07FF8’O	f’ list { 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972 }, variable bitmap format, length = 4
C_flist_r_412	OCTETSTRING	’00020800000007F020000008000080000’O	f’list {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114, 115}, bitmap 0 format, length = 16
C_flist_r_415	OCTETSTRING	’87FCE8117F798483BDF6’O	f’ list {0, 30, 40, 66, 80, 1005, 1010, 1015}, range 1024 format, length = 10
C_flist_r_416	OCTETSTRING	’00020800000007F020000008000080000’O	bit map 0 format, {20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114}, length = 16
C_fshlist_r_417	OCTETSTRING	’89DE023F0180’O	f’ shortlist { 956, 960, 964, 970 }, range 512 format, length=6
C_flist_r_417	OCTETSTRING	’89DE023F018000000000000000000000’O	f’ shortlist { 956, 960, 964, 970 }, range 512 format,
C_flist_r_418	OCTETSTRING	’8FDE08926000000000000000000000’O	f’ shortlist { 956, 960, 964, 967, 970, 973 }, variable bitmap format
C_fshlist_r_418	OCTETSTRING	’8FDE089260’O	f’ shortlist { 956, 960, 964, 967, 970, 973 }, variable bitmap format, length = 5
C_flist_r_419	OCTETSTRING	’8A2486F917E800000000000000000000’O	f’ shortlist { 73, 76, 79, 86, 97 }, range 256 format,
C_fshlist_r_419	OCTETSTRING	’8A2486F917E8’O	f’ shortlist { 73, 76, 79, 86, 97 }, range 256 format, length = 6
C_fshlist_r_420	OCTETSTRING	’8DE01EB000’O	f’ list { 960, 970, 990 } range 128 format, length = 5
C_BABR_0	B_3	’000’B	BABR value 0
C_BABR_1	B_3	’001’B	BABR value 1
C_BPM_2	B_3	’010’B	BPM value 2, 4 multiframes
C_BPM_3	B_3	’011’B	BPM value 3, 5 multiframes

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
C_T3212_0	OCTETSTRING	'00'O	T3212 infinite
C_T3212_1	OCTETSTRING	'01'O	T3212 6 mn / 1 deci hours
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_1111	OCTETSTRING	'1111'O	lac '1111'H
C_LAC_2222	OCTETSTRING	'2222'O	lac '2222'H
C_MCC_1	OCTETSTRING	'00F1'O	MCC 1
C_MCC_2	OCTETSTRING	'00F2'O	MCC 2
C_Max_1	INTEGER	1	Max Retrans value 1
C_Max_2	INTEGER	2	Max Retrans value 2
C_NECI_0	INTEGER	0	NECI value 0, not supported
C_PLMN_1	OCTETSTRING	'10'O	PLMN 1 (mnc=01)
C_TxInt_5	INTEGER	5	Tx Integer value 5
C_NotCombined	BOOLEAN	FALSE	CCCH not combined with SDCCH
C_Combined	BOOLEAN	TRUE	CCCH combined with SDCCH
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_FCCH_SCH_BCCH_CCCH	LOGCH	"C_FCCH_SCH_BCCH_CCCH"	FCCH_SCH_BCCH_CCCH type channel
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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_S0	SN	'000'B	time slot 0
C_U10	CCSTATE	10	CC state U10
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_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_Rcv	BITSTRING	’101’B	activation for receiving only
C_LocEndRel	RelMode	’01’O	local end release
C_MaxPwrLvlG	INTEGER	19	Max power level for MS GSM
C_rc_LAnotallowed	REJCAU	’0C’O	reject cause: LocationArea is not allowed
C_rc_networkfailure	REJCAU	’11’O	reject cause: network failure
C_rc_plmn_not	REJCAU	’0B’O	reject cause: PLMN not allowed
C_Restablishment	B_1	’0’B	Restablishment allowed
C_noRestablishment	B_1	’1’B	No restablishment allowed
C_NoSI2terSI7	B_1	’0’B	No SI2ter or no SI7&8
C_SI2ter	B_1	’1’B	SI2ter exists
C_SI7_8	B_1	’1’B	SI7 & 8 exist
C_SlotNU	SN	’111’B	Slot Number not used
C_CellReselectHys0	INTEGER	0	Cell Reselection Hysteresis value 0db
C_CellReselectHys10	INTEGER	5	Cell Reselection Hysteresis value 10db
C_CellReselectHys12	INTEGER	6	Cell Reselection Hysteresis value 12db
C_CellReselectHys14	INTEGER	7	Cell Reselection Hysteresis value 14db
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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_rgsm	INTEGER	3	Test case for rgsm
C_Test_cbms	INTEGER	4	Test case for CBMS
C_imsi_attach	B_2	'10'B	location updating lMSI 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_reorg_paging	B_2	'10'B	paging reorganisation
C_T_Wait1stChReq	INTEGER	35000	max time for full BCCH data decoding
C_T_Wait	INTEGER	5000	5s. T3126max
C_T_Wait1	INTEGER	1000	1s
C_release_time	INTEGER	11	release timer value
C_Full	IA5String	"F"	full rate channel
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_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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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
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_modemt_abt1	B_5	'01000'B	modem autobaud type1
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
maxAddressLength	Asn1Integer	20	

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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	
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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
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_2"	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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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
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_B_1	LOGICCH	"C_FACCHF_B_1"	FACCH associated with 1st TCH/F of cell B
C_FACCHH_A_1	LOGICCH	"C_FACCHH_A_1"	FACCH associated with 1st TCH/H of cell A
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_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_SACCHC8_C_1	LOGICCH	"C_SACCHC8_C_1"	1st SACCH/8 channel of cell C
C_SACCHH_A_1	LOGICCH	"C_SACCHH_A_1"	SACCH associated with 1st TCH/H of cell A
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_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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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_B_1	LOGICCH	"C_SACCHC8_B_1"	1st SACCH/8 channel of cell B
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
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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_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_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
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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
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_B_1	LOGICCH	"C_SDCCH8_B_1"	1st SDCCH/8 channel of cell B
C_SDCCH8_C_1	LOGICCH	"C_SDCCH8_C_1"	1st SDCCH/8 channel of cell C
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

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
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_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_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
Detailed Comments :			

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_AssCmd	ASS_CMD_PDU		to hold ASSIGNMENT COMMAND PDU
TCV_arfcn_PE	INTEGER		ARFCN of actual P− or E−GSM cell
TCV_BcapMT1	BCAP		to hold BC for MT call
TCV_BcapMO1	BCAP		to hold BC for MO call
TCV_BcapMT2	BCAP		to hold BC for MT call
TCV_bcchfl_P	NCD		bcch freqlst for P−GSM
TCV_bcchfl_DE	NCD		bcch freqlst for E−GSM or DCS
TCV_bs_ag_blks_res	INTEGER		
TCV_bs_cc_chans	INTEGER		
TCV_bs_pa_mfrms	INTEGER		
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_CE	CE		
TCV_Ccchg	INTEGER		CCCH_GROUP
TCV_Ccd0A	CCD		a copy of control channel description in use for cell A
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 RGSM
TCV_CCst	CST		to hold call status
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_sacch8	LOGICCH	"dummy"	to hold logic channel
TCV_sacchTch	LOGICCH	"dummy"	to hold logic channel
TCV_sacchTch1	LOGICCH	"dummy"	to hold logic channel
TCV_chTch	LOGICCH	"dummy"	to hold logic channel
TCV_chTch1	LOGICCH	"dummy"	to hold logic channel

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_agch	LOGICCH	"dummy"	downlink access grant channel
TCV_cellid	CellID		to hold the cell identifier
TCV_chd1	CHD		to hold channel description IE
TCV_ChMod	CHMOD		to hold channel mode IE
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_Cntref	INTEGER		Reference for general loop counter
TCV_CphKey	BITSTRING		ciphering key
TCV_ChRate	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 RGSM for IE "frequency list" or "frequency short list"
TCV_flistl	LENGTH		Length of Frequency List, used in RGSM 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_HoaccPara	HOACC_PARA		Variable for HO access parameters
TCV_Horf	HORF		Variable for HO reference in HO−Messages of HO cases.
TCV_hotime	INTEGER		to store the maximum handover time periods in ms
TCV_iel	LENGTH		Length of IE
TCV_kcnt	INTEGER		loop counter
TCV_K	INTEGER		

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
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_maio	MAIO		to hold MAIO
TCV_mar1	BITSTRING		Mobile Allocation in RGSM. It includes the 1.octet of MA
TCV_mar2	BITSTRING		Mobile Allocation in RGSM. It includes the 2.octet of MA
TCV_Max	BITSTRING		to hold Max−retrans bitstring
TCV_MsrRes	MSRR		to hold measurement results
TCV_Mt	BITSTRING		to hold message type
TCV_Mt1	BITSTRING		to hold message type
TCV_n	INTEGER		
TCV_Neci	BITSTRING		
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_PO	B_2		power offset
TCV_Pwrlvl_ho	BITSTRING		power level for HO_CMD, used in HO cases.
TCV_RaCh	LOGICCH	"dummy"	logical channel f RACH
TCV_Rchr	B_2		to hold Radio channel requirement
TCV_Res	BOOLEAN		to hold the result of a test suite operation
TCV_Rr	BITSTRING		to hold the random request reference
TCV_Service	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_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_Sres	OCTETSTRING		to hold the SRES returned
TCV_sysinf5	SYSINFO5_PDU		to hold the SysInfo5 PDU

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_sysinf5bis	SYSINFO5bis_PDU		to hold the SysInfo5bis PDU for cell A
TCV_sysinf5ter	SYSINFO5ter_PDU		to hold the SysInfo5ter PDU
TCV_sysinfo5	SYSINFO5_PDU		to hold the SysInfo5 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_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_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.
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_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_Strt	STRT		to hold starting time
TCV_SMTypeM	INTEGER		Replace short message type

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
TCV_SMTypeN	INTEGER		Replace short message type
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_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_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
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 :			

Timer Declarations			
Timer Name	**Duration**	**Unit**	**Comments**
T_dly		ms	general purpose delay timer
T_dly1		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
T_pause		ms	pause timer
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_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_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_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_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_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_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_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_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_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_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_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_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_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_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_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_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_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_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_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_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 : MDL_RelRq
(MDL_RELEASE_REQUEST)

PCO Type : SAP0_3

Comments : The ASP is used to request the local end termination 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		Assignment command 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], ’05’O
cchd	CCHD		cell channel description O, ’62’O OCTETSTRING [17]
dmsc	MSA		description of the multislot configuration, C, ’10’O, OCTETSTRING [3..12]
ch1mod	CHMOD		mode of the 1st channel (set 1), O, ’63’O, OCTETSTRING [2]
modchs2	CHMOD		mode of channel set 2, O, ’11’O, OCTETSTRING [2]
modchs3	CHMOD		mode of channel set 3, O, ’13’O, OCTETSTRING [2]
modchs4	CHMOD		mode of channel set 4, O, ’14’O, OCTETSTRING [2]
modchs5	CHMOD		mode of channel set 5, O, ’15’O, OCTETSTRING [2]
modchs6	CHMOD		mode of channel set 6, O, ’16’O, OCTETSTRING [2]
modchs7	CHMOD		mode of channel set 7, O, ’17’O, OCTETSTRING [2]
modchs8	CHMOD		mode of channel set 8, O, ’18’O, OCTETSTRING [2]
ch2d_at	CHD		description of the 2nd channel, after time, ’64’O O OCTETSTRING [4]
ch2mod	CHMOD		mode of the 2nd channel O ’66’O, OCTETSTRING [2]
ma_at	MA		mobile allocation after time C, ’72’O, OCTETSTRING [3..10]

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	**Field Type**	**Field Encoding**	**Comments**
strt	STRT		starting time O, '7C'O, OCTETSTRING [3]
frql_bt	FRQL		frequency list, before time C OCTETSTRING [4..132], '19'O
ch1d_bt	CHD		description of the 1st channel, before time O, '1C'O OCTETSTRING [4]
ch2d_bt	CHD		description of the 2nd channel, before time O, '1D'O, OCTETSTRING [4]
frqchs_bt	FRQCHS		frequency channel sequence before time C, '1E'O OCTETSTRING [10]
ma_bt	MA		mobile allocation before time C, '21'O OCTETSTRING [3..10]
cphms	CPHMS		cipher mode setting O, '9−'O, OCTETSTRING [1]
vgcstmi	VGCSTMI		VGCS target mode indication O, '01'O, OCTETSTRING [2]
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 cause 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 cause 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 cause, M BITSTRING [8]
barng	BARANGE		BA range, O, '73'O, OCTETSTRING [6..251]
gchd	GCHD		group channel description, O '74'O, OCTETSTRING [4..13]
gckn	GCKN		group cipherkey number, O '8x'O, BITSTRING [8]
gprsr	GPRSR		GPRS resumption, O, 'Cx', BITSTRING[8]

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_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 '17'O, OCTETSTRING [10], 15 digits, BCD

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, '62'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 and Access type M OCTETSTRING [1]
synchi	SYNCHI		synchronization indication, O, 'D−', OCTETSTRING [1]
frqsl_at	FRQL		frequency short list, after time C, '02'O, OCTETSTRING [10]
frql_at	FRQL		frequency list, after time C '05'O, OCTETSTRING [4..132]
cchd	CCHD		cell channel description C, '62'O, OCTETSTRING [17]
dmsc	MSA		description of the multislot configuration, O, phase 2+, '10'O, OCTETSTRING [3..12]
ch1mod	CHMOD		mode of the 1st channel (set 1), O, '63'O, OCTETSTRING [2]
modchs2	CHMOD		mode of channel set 2, O, phase 2+, '11'O, OCTETSTRING [2]
modchs3	CHMOD		mode of channel set 3, O, phase 2+, '13'O, OCTETSTRING [2]
modchs4	CHMOD		mode of channel set 4, O, phase 2+, '14'O, OCTETSTRING [2]
modchs5	CHMOD		mode of channel set 5, O, phase 2+, '15'O, OCTETSTRING [2]
modchs6	CHMOD		mode of channel set 6, O, phase 2+, '16'O, OCTETSTRING [2]

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	**Field Type**	**Field Encoding**	**Comments**
modchs7	CHMOD		mode of channel set 7, O, phase 2+, ’17’O, OCTETSTRING [2]
modchs8	CHMOD		mode of channel set 8, O, phase 2+, ’18’O, OCTETSTRING [2]
ch2d_at	CHD		description of the 2nd channel, after time O ’64’O, OCTETSTRING [4]
ch2mod	CHMOD		mode of the 2nd channel O ’66’O, OCTETSTRING [2]
frqchs_at	FRQCHS		frequency channel sequence, after time, C, ’69’O, OCTETSTRING [10]
ma_at	MA		mobile allocation, after time, C, ’72’O, OCTETSTRING [3..10]
strt	STRT		starting time O ’7C’O, OCTETSTRING [3]
rtdif	TDIF		real time difference C ’7B’O, OCTETSTRING [3]
ta	TA		timing advance C ’7D’O, OCTETSTRING [2]
frqsl_bt	FRQL		frequency short list, before time C ’12’O, OCTETSTRING [10]
frql_bt	FRQL		frequency list, before time C ’19’O, OCTETSTRING [4..132]
ch1d_bt	CHD		description of the 1st channel, before time O ’1C’O, OCTETSTRING [4]
ch2d_bt	CHD		description of the 2nd channel, before time O ’1D’O, OCTETSTRING [4]
frqchs_bt	FRQCHS		frequency channel sequence, before time C ’1E’O, OCTETSTRING [10]
ma_bt	MA		mobile allocation, before time C ’21’O, OCTETSTRING [3..10]
cphms	CPHMS		cipher mode setting O ’9−’O, OCTETSTRING [1]
vgcstmi	VGCSTMI		VGCS target mode indication O, ’01’O, phase 2+, OCTETSTRING [2]
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 cause M OCTETSTRING [1]
motdif	MTDIF		mobile observed time difference O '77'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 cause 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]
dm_tbf	Dm_TBF		TBF or dedicate mode M, BITSTRING [4]
pm	PM		page mode M BITSTRING [4]
chd	CHD		channel description C OCTETSTRING [3]
pchd	PCHD		packet channel description C 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 ’7C’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 : 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 '17'O OCTETSTRING [3..10]
p1roct	OCTETSTRING [0..17]		P1 rest octets '2B2B2B...'O
p1roct2	P1RO		P1 rest octets, phase 2+

Detailed Comments : The message has a fixed 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 : 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 : 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	SI1RO		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 n −> ms
GSM 04.08, 9.1.33

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]		SI2bis rest octets M 1

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.

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, 9.1.35

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]
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, 9.1.36

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 '64'O O OCTETSTRING [4]
cbchma	MA		mobile allocation '72'O C OCTETSTRING [3..6]
si4roct	SI4RO		SI4 rest octets M OCTETSTRING [1..10]

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, 9.1.37

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, 9.1.40

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 , OCTETSTRING[1]
nccp	NCCP		ncc permitted M OCTETSTRING [1]
si6roct	SI6RO		SI6 rest octets M OCTETSTRING [7]

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 : 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 : 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]
pr	PRT		priority '8−'O O OCTETSTRING[1]

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 '17'O O OCTETSTRING [2..9]
fop	IEI_8		follow−on proceed 'A1' O BITSTRING [8]

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 cause 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 M OCTETSTRING [5]
msclm	MSCLM1		mobile station classmark 1 M OCTETSTRING [4]
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 '1C' O
pi	PI		progress indicator n −> ms '1E' O OCTETSTRING [4]
uu	UU		user−user '7E'O O OCTETSTRING [3..35]
ssvi	SSVI		SS version indicator ms −> n '7F'O 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 BITSTRING [4]
mt	MT		message type M BITSTRING [8]
bcri	RPI		BC repeat indicator 'D−'O C BITSTRING [8]
bcap1	BCAP		bearer capab. '04'O O OCTETSTRING [3..10]
bcap2	BCAP		bearer capab. '04'O O
cau	CAU		cause '08'O O OCTETSTRING [4..32]
cccap	CCCAP		cc capabilities '15'O O OCTETSTRING [3]

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 'D–'O C BITSTRING [8]
bcap1	BCAP		bearer capab. O '04'O OCTETSTRING [3..10]
bcap2	BCAP		bearer capab. O '04'O OCTETSTRING [3..10]
fie	FIE		facility '1C'O O
pi	PI		progress indicator '1E'O O OCTETSTRING [4]
pr	PRT		priority granted '8–'O O OCTETSTRING[1]

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 '1C' O
pi	PI		progress indicator n −> ms '1E'O O OCTETSTRING [4]
cnn	CNN		connected number n −> ms '4C'O O OCTETSTRING [3..14]
cns	CNS		connected subaddress '4D'O O OCTETSTRING [2..23]
uu	UU		user−user information '7E'O O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n '7F'O O OCTETSTRING [2..3]

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 '1C'O O
pi	PI		progress indicator n −> ms '1E'O O OCTETSTRING [4]
uu	UU		user−user information '1F'O O OCTETSTRING [3..35]
ssvi	SSVI		SS version indicator '7F'O O OCTETSTRING [3]

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. '04'O O OCTETSTRING [3..10]

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 '08'O O OCTETSTRING [4..32]
cau2	CAU		second cause '08'O O OCTETSTRING [4..32]
fie	FIE		facility '1C'O O
uu	UU		user−user information '7E'O O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n '7F'O 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 '08'O O OCTETSTRING [4..32]
fie	FIE		facility '1C'O O
uu	UU		user−user information '7E'O O OCTETSTRING [3..35]
ssvi	SSVI		SS version ms −> n '7F'O 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 'D−'O C BITSTRING [8]
bcap1	BCAP		bearer capab. '04'O M
bcap2	BCAP		bearer capab. '04'O O OCTETSTRING [3..10]
fie	FIE		facility '1C'O O
cgps	CGPS		calling party subaddr. '5D'O O OCTETSTRING [2..23]
cdpn	CDPN		called party BCD number '5E'O M OCTETSTRING[2..23]
cdps	CDPS		called party subaddr. '6D'O O OCTETSTRING [2..23]
llcri	RPI		LLC repeat indicator 'D−'O O OCTETSTRING [1]
llcmp1	LLCMP		low layer compatib.1 '7C'O O OCTETSTRING [2..15]
llcmp2	LLCMP		low layer compatib.2 '7C'O O
hlcri	RPI		HLC repeat indicator 'D−' O OCTETSTRING [1]
hlcmp1	HLCMP		high layer compat.1 '7D'O O OCTETSTRING [2..5]
hlcmp2	HLCMP		high layer compat. 2 '7D'O O
uu	UU		user−user information '7E'O O OCTETSTRING [3..35]
ssvi	SSVI		SS version '7F'O O OCTETSTRING [2..3]
clirsup	CLRSUP		CLIR suppression 'A1'O C OCTETTRING[1]
clirinv	CLRINV		CLIR invocation 'A2'O C OCTETSTRING[1]

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	**Field Type**	**Field Encoding**	**Comments**
cccap	CCCAP		call control capabilities '15'O O OCTETSTRING[3]
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 'D−'O C BITSTRING [8]
bcap1	BCAP		bearer capab. '08'O O
bcap2	BCAP		bearer capab. '08'O O OCTETSTRING [3..10]
fie	FIE		facility '1C'O O
pi	PI		progress indicator '1E'O O OCTETSTRING [4]
sig	SIGNAL		signal '34'o O OCTETSTRING[2..34]
cgpn	CGPN		calling party BCD number '5C'o O OCTETSTRING [2..24]
cgps	CGPS		calling party subaddr. '5D'O O OCTETSTRING [2..23]
cdpn	CDPN		called party BCD number '5E'O O OCTETSTRING[2..23]
cdps	CDPS		called party subaddr. '6D' O OCTETSTRING [2..23]
llcri	RPI		LLC repeat indicator 'D−'O O OCTETSTRING [1]
llcmp1	LLCMP		low layer compatib.1 '7C'O O OCTETSTRING [2..15]
llcmp2	LLCMP		low layer compatib.2 C '7C'O OCTETSTRING [2..15]
hlcri	RPI		HLC repeat indicator 'D−'O O OCTETSTRING [1]
hlcmp1	HLCMP		high layer compat.1 '7D'O O OCTETSTRING [2..5]
hlcmp2	HLCMP		high layer compat. 2 C '7D'O OCTETSTRING [2..5]
uu	UU		user−user information '7E'O O OCTETSTRING [3..35]
pr	PRT		priority '8−'O O OCTETSTRING[1]

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	**Field Type**	**Field Encoding**	**Comments**
alert	ALPT		alerting pattern '19'O O OCTETSTRING[3]
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 '24'O 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 :			

III

Constraints Part

Structured Type Constraint Declaration

Constraint Name : Bcap_MO(rchr1 : B_2; itc1 : B_3; strc1 : B_2; nirr : B_1; ra1 :B_2; sacp : B_3; sb, nsb1, ndb1 : B_1; ur1 : B_4; ir1 : B_2; parity : B_3; ce1 : B_2; modemt1 : B_5)

Structured Type : BCAP

Derivation Path :

Encoding Variation :

Comments : BS bearer capability without OCTET 7 for direction ms −> n

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	(’07’O, ’08’O, ’09’O, ’0A’O, ’0B’O, ’0C’O, ’0D’O)		length of the IE contents
extb3	?		
rchr	rchr1		radio channel requirement
cs	’0’B		coding standard : GSM
tm	’0’B		circuit
itc	itc1		information transfer capability
oct_3a_etc_1	*		octet 3a etc no.1
oct_3a_etc_2	*		octet 3a etc no.2
oct_3a_etc_3	*		octet 3a etc no.3
oct_3a_etc_4	*		octet 3a etc no.4
oct_3a_etc_5	*		octet 3a etc no.5
oct_3a_etc_6	*		octet 3a etc no.6
extb4	’1’B		Extension
spb	’0’B		Spare
strc	strc1		structure
dplxm	’1’B		Full duplex
config	’0’B		Point to point
nirr	nirr		
est	’0’B		Demand
extb5	’1’B		Extension
accid	’00’B		Access id
ra	ra1		rate adaption
sacp	sacp		Signalling access protocol
extb6	’0’B		Extension
l1id	’01’B		L1 id: default
uil1	’0000’B		User information L1 protocol
sb	sb		Synchronous
extb6a	’0’B		Extension
nsb	nsb1		number of stop bits, not checked
nb	’0’B		In band negotiation not possible
ndb	ndb1		number of data bits, not checked
ur	ur1		user rate
extb6b	’0’B		Extension
ir	ir1		intermediate rate
nictx	’0’B		NIC on Tx: not required
nicrx	’0’B		NIC on Rx: not supported

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
pi	parity		Parity, not checked
extb6c	’1’B		Extension
ce	ce1		connection element
modemt	modemt1		modem type
extb7	–		Extension
l2id	–		layer 2 id
uil2	–		user information layer 2 protocol
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : Bcap_MT(itc1 : B_3; strc1 : B_2; nirr : B_1; ra1 :B_2; sacp : B_3; sb : B_1; nbsb , nbdb : B_1; ur1 : B_4; ir1 : B_2; parity : B_3; ce1 : B_2; modemt1 : B_5)

Structured Type : BCAP

Derivation Path :

Encoding Variation :

Comments : BS bearer capability for direction n−>ms

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	’07’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
itc	itc1		information transfer capability
oct_3a_etc_1	−		octet 3a etc no.1
oct_3a_etc_2	−		octet 3a etc no.2
oct_3a_etc_3	−		octet 3a etc no.3
oct_3a_etc_4	−		octet 3a etc no.4
oct_3a_etc_5	−		octet 3a etc no.5
oct_3a_etc_6	−		octet 3a etc no.6
extb4	’1’B		Extension
spb	’0’B		Spare
strc	strc1		structure
dplxm	’1’B		Full duplex
config	’0’B		Point to point
nirr	nirr		No meaning
est	’0’B		Demand
extb5	’1’B		Extension
accid	’00’B		Access id
ra	ra1		rate adaption
sacp	sacp		Signalling access protocol
extb6	’0’B		Extension
l1id	’01’B		L1 id: default
uil1	’0000’B		User information L1 protocol
sb	sb		Synchronous
extb6a	’0’B		Extension
nsb	nbsb		number of stop bits
nb	’0’B		In band negotiation not possible
ndb	nbdb		number of data bits
ur	ur1		user rate
extb6b	’0’B		Extension
ir	ir1		intermediate rate
nictx	’0’B		NIC on Tx: not required
nicrx	’0’B		NIC on Rx: not supported
pi	parity		Parity
extb6c	’1’B		Extension

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
ce	ce1		connection element
modemt	modemt1		modem type
extb7	–		Extension
l2id	–		layer 2 id
uil2	–		user information layer 2 protocol
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : BcapX_MO(rchr1 : B_2; itc1 : B_3; strc1 : B_2; nirr : B_1; ra1 :B_2; sacp : B_3; sb, nsb1, ndb1 : B_1; ur1 : B_4; ir1 : B_2; parity : B_3; ce1 : B_2; modemt1 : B_5; fctl : B_5)

Structured Type : BCAP

Derivation Path :

Encoding Variation :

Comments : BS bearer capability with OCTET 7 for direction ms −> n

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	(’07’O, ’08’O, ’09’O, ’0A’O, ’0B’O, ’0C’O, ’0D’O, ’0E’O)		length of the IE contents
extb3	?		
rchr	rchr1		radio channel requirement
cs	’0’B		coding standard : GSM
tm	’0’B		circuit
itc	itc1		information transfer capability
oct_3a_etc_1	*		octet 3a etc no.1
oct_3a_etc_2	*		octet 3a etc no.2
oct_3a_etc_3	*		octet 3a etc no.3
oct_3a_etc_4	*		octet 3a etc no.4
oct_3a_etc_5	*		octet 3a etc no.5
oct_3a_etc_6	*		octet 3a etc no.6
extb4	’1’B		Extension
spb	’0’B		Spare
strc	strc1		structure
dplxm	’1’B		Full duplex
config	’0’B		Point to point
nirr	nirr		
est	’0’B		Demand
extb5	’1’B		Extension
accid	’00’B		Access id
ra	ra1		rate adaption
sacp	sacp		Signalling access protocol
extb6	’0’B		Extension
l1id	’01’B		L1 id: default
uil1	’0000’B		User information L1 protocol
sb	sb		Synchronous
extb6a	’0’B		Extension
nsb	nsb1		number of stop bits
nb	’0’B		In band negotiation not possible
ndb	ndb1		number of data bits, not checked
ur	ur1		user rate
extb6b	’0’B		Extension
ir	ir1		intermediate rate
nictx	’0’B		NIC on Tx: not required
nicrx	’0’B		NIC on Rx: not supported
pi	parity		Parity

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
extb6c	’1’B		Extension
ce	ce1		connection element
modemt	modemt1		modem type
extb7	’1’B IF_PRESENT		Extension
l2id	’10’B IF_PRESENT		layer 2 id
uil2	fctl IF_PRESENT		user information layer 2 protocol
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : BcapX_MT(itc1 : B_3; strc1 : B_2; nirr : B_1; ra1 :B_2; sacp : B_3; sb : B_1; nbsb , nbdb : B_1; ur1 : B_4; ir1 : B_2; parity : B_3; ce1 : B_2; modemt1 : B_5; flct : B_5)

Structured Type : BCAP

Derivation Path :

Encoding Variation :

Comments : BS bearer capability for direction n−>ms BS 21, .. 26, except BS23

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	’08’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	itc1		information transfer capability
oct_3a_etc_1	−		octet 3a etc no.1
oct_3a_etc_2	−		octet 3a etc no.2
oct_3a_etc_3	−		octet 3a etc no.3
oct_3a_etc_4	−		octet 3a etc no.4
oct_3a_etc_5	−		octet 3a etc no.5
oct_3a_etc_6	−		octet 3a etc no.6
extb4	’1’B		
spb	’0’B		
strc	strc1		structure
dplxm	’1’B		
config	’0’B		
nirr	nirr		
est	’0’B		
extb5	’1’B		
accid	’00’B		
ra	ra1		rate adaption
sacp	sacp		
extb6	’0’B		
l1id	’01’B		
uil1	’0000’B		
sb	sb		
extb6a	’0’B		
nsb	nbsb		number of stop bits, 1
nb	’0’B		negotiation
ndb	nbdb		number of data bits, 8
ur	ur1		user rate
extb6b	’0’B		
ir	ir1		intermediate rate, 16 or 8 kbits/s
nictx	’0’B		
nicrx	’0’B		
pi	parity		Parity none
extb6c	’1’B		
ce	ce1		connection element

Continued on next page

Continued from previous page

Structured Type Constraint Declaration			
Element Name	**Element Value**	**Element Encoding**	**Comments**
modemt	modemt1		modem type
extb7	’1’B		
l2id	’10’B		layer 2 id
uil2	flct		user information layer 2 protocol
Detailed Comments :			

Structured Type Constraint Declaration

Constraint Name : Bcap_Speech_MO(rcr : B_2)
Structured Type : BCAP
Derivation Path :
Encoding Variation :
Comments : speech bearer capability for direction ms−>n.

Element Name	Element Value	Element Encoding	Comments
iei	’00000100’B		IEI
iel	(’01’O, ’02’O,’03’O, ’04’O, ’05’O, ’06’O, ’07’O)		length of the IE contents
extb3	?		
rchr	rcr		radio channel reqirement
cs	’0’B		coding standard : GSM
tm	’0’B		circuit mode
itc	’000’B		speech
oct_3a_etc_1	*		octet 3a etc no.1
oct_3a_etc_2	*		octet 3a etc no.2
oct_3a_etc_3	*		octet 3a etc no.3
oct_3a_etc_4	*		octet 3a etc no.4
oct_3a_etc_5	*		octet 3a etc no.5
oct_3a_etc_6	*		octet 3a etc no.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 : 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	−		octet 3a etc no.1
oct_3a_etc_2	−		octet 3a etc no.2
oct_3a_etc_3	−		octet 3a etc no.3
oct_3a_etc_4	−		octet 3a etc no.4
oct_3a_etc_5	−		octet 3a etc no.5
oct_3a_etc_6	−		octet 3a etc no.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 : 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 : BcchFreqLst2_Omit			
Structured Type : NCD2			
Derivation Path :			
Encoding Variation :			
Comments : Omit Bcch Frequency list 2 (Neighbour cells description)			
Element Name	**Element Value**	**Element Encoding**	**Comments**
b128	–		
mbrpt	–		
baind	–		
b122_124	–		
rfl1	–		
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		
extind	'0'B		
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_21
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Neighbour cells description for SYSTEM INFORMATION 5 for measurement testing. BA list = {2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24, 26, 28, 29, 30, 32, 34, 35, 36, 38, 40, 44}, format is bitmap 0, IE carries complete BA.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0 format
extind	'0'B		complete BA
baind	'1'B		BCCH allocation sequence number
rfl4	'0000'B		
rfl	'0000000000000000008AEBAFFBFFA'O		ARFCNs: 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24, 26, 28, 29, 30, 32, 34, 35, 36, 38, 40, 44

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_27
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Neighbour cells description for SYSTEM INFORMATION 5 for DCS1800. BA list = {514, 530, 549, 602, 665, 686, 762, 810}, format is 1024 range, IE carries complete BA.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		range 1024 format
extind	'0'B		complete BA
baind	'1'B		BCCH allocation sequence number
rfl4	'0010'B		
rfl	'99C6187B6D0D4C3800000000000000'O		ARFCN 514, 530, 549, 602, 665, 686, 762, 810

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_33
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Neighbour cells description for SYSTEM INFORMATION 5 for GSM900. BA list = {2, 14, 20, 38, 44}, format is bitmap 0, IE carries only a part of the BA.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0
extind	'1'B		partial BA
baind	'1'B		
rfl4	'0000'B		
rfl	'00000000000000000082000082002'O		ARFCN 2, 14, 20, 38, 44

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r46
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis and 5bis in cell A under RGSMwith the ARFCN list = { 962, 965, 980, 990}.

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	'1011'B		
rfl	'E10971140000000000000000000000'O		ARFCN: 962, 965, 980, 990

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r47
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis and 5bis in cell B under RGSMwith the ARFCN list = { 969, 970, 972, 1005, 1010}.

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	'1011'B		
rfl	'E1097415E800000000000000000000'O		ARFCN: 969, 970, 972, 1005, 1010

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_50c
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {3, 9, 18, 25, 41, 43, 49, 50, 54, 58, 62, 66, 70, 80, 92, 124}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0 format
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1000'B		
rfl	'000000080080222223050 001020104'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_52
Structured Type : NCD
Derivation Path : BcchFreqLst_33.
Encoding Variation :
Comments : Neighbour cells description for SYSTEM INFORMATION 5 of cell B for RGSM. BA list = {10, 36, 40, 114, 118}, format is bitmap 0, IE carries only a part of the BA.

Element Name	Element Value	Element Encoding	Comments
baind	'0'B		
rfl	'220000000000000000008 800000200'O		ARFCN 10, 36, 40, 114, 118

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_52c_R
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis under RGSM with the ARFCN list = {956, 960, 969, 985, 989, 995, 1010, 1014}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		512 format
extind	'1'B		not complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'1001'B		
rfl	'DE087B057BF120100000 0000000000'O		ARFCN: 956, 960, 969, 985, 989, 995, 1010, 1014

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_52e_R
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis under RGSM with the ARFCN list = {960, 969, 989, 995, 1010, 1014}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		512 format
extind	'1'B		not complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'1001'B		
rfl	'E008FE84F67C000000000000000000'O		ARFCN: 960, 969, 989, 995, 1010, 1014

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_53
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {3, 9, 18, 25}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0 format
extind	'1'B		complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'0000'B		
rfl	'00000000000000000000001020104'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_53e
Structured Type : NCD
Derivation Path : BcchFreqLst_53.
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {3, 9, 18}.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000000020104'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_55_R
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis under RGSM with the ARFCN list = {956}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		512 format
extind	'1'B		not complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'1001'B		
rfl	'DE00000000000000000000000000000'O		ARFCN: 956

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_56_R
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2bis under RGSM with the ARFCN list = {9, 43, 70, 500, 550, 958, 963, 990, 995}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		1024 format
extind	'1'B		not complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'0011'B		
rfl	'DE0B1334378C86FDDB000000000000'O		ARFCN: 9, 43, 70, 500, 550, 958, 963, 990, 995

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_57e
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {3, 18, 25, 41, 43, 49, 50, 54, 58, 62, 66, 70, 80, 92, 124}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0 format
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1000'B		
rfl	'000000080080222223050 001020004'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_60
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {3, 18, 25, 41}.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		bit map 0 format
extind	'1'B		complete BA
baind	'0'B		BCCH allocation sequence number
rfl4	'0000'B		
rfl	'000000000000000000010 001020004'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_63
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description for SYSTEM INFORMATION 2 and 5 under GSM900 with the ARFCN list = {3, 18, 25, 41, 43, 49}.

Element Name	Element Value	Element Encoding	Comments
rfl2	’00’B		bit map 0 format
extind	’1’B		
baind	’0’B		BCCH allocation sequence number
rfl4	’0000’B		
rfl	’00000000000000000010501001020004’O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst2_65_R
Structured Type : NCD2
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description type 2 for SYSTEM INFORMATION 2 and 5 for DCS1800 with the ARFCN list = {45, 76, 891, 905, 956}.

Element Name	Element Value	Element Encoding	Comments
b128	’1’B		Range 512, format−ID
mbrpt	’00’B		Multiband reporting − see 05.08
baind	’0’B		BA(BCCH) measurement reporting only
b122_124	’100’B		Bit 124 − 122, Range 512 format ID
rfl1	’1’B		1 reference frequency
rfl	’BDA3E3C7E680000000000000000000’O		ARFCN: 45, 76, 891, 905, 956

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r201
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		128 format
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1100'B		
rfl	'038C0D0C000000000000 0000000000'O		ARFCN: 26 38 Format: 128

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r202
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		bit map 0 format
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1011'B		
rfl	'E00F6C000000000000000 000000000'O		ARFCN: 960, 970, 990 Format: 256

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r203
Structured Type : NCD
Derivation Path : BcchFreqLst_01.
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'0'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1001'B		
rfl	'04797F83F980000000000000000000'O		ARFCN: 520,990,1003,1005,1020 Format: 512

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r204
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'0111'B		
rfl	'DEF611FDBD00000000000000000000'O		ARFCN: 0, 26, 38, 960, 970, 990 Format: 1024

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r205
Structured Type : NCD
Derivation Path : BcchFreqLst_01.
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'0'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1110'B		
rfl	'00000020020000000000 000000000'O		ARFCN: 0,26,38 Format: variable bit map

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r206
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'00'B		
extind	'0'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'0000'B		
rfl	'00000000000000000000 000000200'O		ARFCN: 26 Format: Bitmap 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r207
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		512 format
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1001'B		
rfl	'0475BB07BB000000000000000000000'O		ARFCN: 520, 960, 970, 990, 1020 Format: 512

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r208
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1100'B		
rfl	'058C0026D000000000000000000000'O		ARFCN: 0,26,38 Format: 128

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r209
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'1'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1010'B		
rfl	'058A00137400000000000000000000'O		ARFCN: 0,26,38 Format: 256

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r210
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'0'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'0011'B		
rfl	'E810050000000000000000000000000'O		ARFCN: 520, 1000, 1020 Format: 1024

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r211
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'0'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1111'B		
rfl	'BA0000000000000000000802000020'O		ARFCN: 884, 960, 970, 990 884, 990, 1003, 1005 Format: Variable bit map

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : BcchFreqLst_r212
Structured Type : NCD
Derivation Path :
Encoding Variation :
Comments : Default neighbour cells description any cell for SYSTEM INFORMATION 2 and 5 under RGSM.

Element Name	Element Value	Element Encoding	Comments
rfl2	'10'B		
extind	'0'B		
baind	'0'B		BCCH allocation sequence number
rfl4	'1001'B		
rfl	'0475B8000000000000000000000000'O		ARFCN: 520, 960, 990 Format: 512

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_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 : 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_10
Structured Type : CAU
Derivation Path :
Encoding Variation :
Comments : cause information element with cause value = #1, location = user.

Element Name	Element Value	Element Encoding	Comments
iei	'00001000'B		
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	'000'B		normal event
cau_va	'0001'B		unallocated number
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_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	'00000000000000000000000000000200'O		ARFCNs : 10

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_05
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N1 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 00000000080'O		ARFCNs : 8

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_06
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N2 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 00000002000'O		ARFCNs : 14

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_07
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N3 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 00000080000'O		ARFCNs : 20

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_08
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N4 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 00002000000'O		ARFCNs : 26

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_09
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N5 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 00080000000'O		ARFCNs : 32

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_10
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N6 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 02000000000'O		ARFCNs : 38

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_11
Structured Type : CCHD
Derivation Path : CellChDes_04.
Encoding Variation :
Comments : for cell N7 in measurement testing of GSM 900.

Element Name	Element Value	Element Encoding	Comments
rfl	'00000000000000000000 80000000000'O		ARFCNs : 44

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_r19
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : Frequency list for RGSM test case TC_26_10_2_2R: 0, 30, 40, 66, 80, 958, 1010, 972.

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'841ED7893CB989838000 000000000000'O		

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_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	'0002080000000F02080000A060A22000'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_20_Be(par_rfl : OCTETSTRING)
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell B in RR testing of RGSM HO cases with the complete Cell Allocation ARFCN_list={par_rfl}.

Element Name	Element Value	Element Encoding	Comments
iei	'01100010'B		
rfl	par_rfl		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_22
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell A in RGSM testing: 20, 30, 50, 70

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'00000000000002000020000020080000'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_23A
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell A in RGSM testing, TC_26_10_2_4_1 : 20, 40, 66, 73, 74, 75, 76, 77, 78, 79, 108, 114

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'0002080000007F0200000 08000080000'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_23B
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell B in RGSM testing, TC_26_10_2_4_1 : 10, 12, 40, 60, 62, 63

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'000000000000000068000 08000000A00'O		Format:Bit map 0

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : CellChDes_24
Structured Type : CCHD
Derivation Path :
Encoding Variation :
Comments : for cell selection testing : 3, 9, 18, 25, 41, 43, 49, 50, 54, 58, 62, 66, 70, 80, 92, 124

Element Name	Element Value	Element Encoding	Comments
iei	–		
rfl	'080000000800802222230 50001020104'O		Format:Bit map 0

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 : CellSelPara_pl(crh, mtmc : INTEGER; neci, acs : B_1; ram : B_6)
Structured Type : CSP
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
crh	INT_TO_BIT(crh, 3)		
mtmc	INT_TO_BIT(mtmc, 5)		
acs	acs		SI7, SI8 present
neci	neci		new establishment causes
ram	ram		

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 : ChDescrp_nfhiei(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, with iei.

Element Name	Element Value	Element Encoding	Comments
iei	'01100100'B		
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 : ChDescrp_sdcch8_fh(slot : SN; tsc : TSC; sch : B_3; maio : MAIO; hsn : HSN)
Structured Type : CHD
Derivation Path :
Encoding Variation :
Comments : Channel description for SDCCH8.

Element Name	Element Value	Element Encoding	Comments
iei	–		'01100100'B
cht_schn	INT_TO_BIT((8 + BIT_TO_INT(sch)), 5)		
tn	slot		
tsc	tsc		
hch	'1'B		hopping
maio	maio		
hsn	hsn		
spr	–		
arfcn	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : ChDescrp_sdcch8_nfh(slot : SN; tsc : TSC; subch : B_3; arfcn : INTEGER)
Structured Type : CHD
Derivation Path :
Encoding Variation :
Comments : Channel description for SDCCH8 in cell A for RR test.

Element Name	Element Value	Element Encoding	Comments
iei	–		'01100100'B
cht_schn	INT_TO_BIT((8 + BIT_TO_INT(subch)), 5)		
tn	slot		
tsc	tsc		
hch	'0'B		non hopping
maio	–		
hsn	–		
spr	'00'B		
arfcn	INT_TO_BIT(arfcn, 10)		

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_r01
Structured Type : CHMOD
Derivation Path :
Encoding Variation :
Comments : for TC_26_6_13_1 after time

Element Name	Element Value	Element Encoding	Comments
iei	'01100011'B		
mode	TSPX_ChMod1		

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_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 : 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 : 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 : 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 : Dm_TBF_def
Structured Type : Dm_TBF
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
spb	’0’B		spare
tma	’0’B		no meaning
d	’0’B		no meaning
td	’0’B		dedicate mode

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 : 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 : FreqTCH_rf1(par_ma: BITSTRING; par_freqlist: OCTETSTRING; par_flistl: LENGTH; n :INTEGER)
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : Broadcast channel of cell A for RGSM.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD n), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'01'O		
mac_8n	par_ma		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		mac
flst	Frql_20_rgsm(par_freqlist, par_flistl)		frequency list, after time for cell A

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqTCH_rf2(par_ma1: BITSTRING;par_ma2: BITSTRING; par_freqlist: OCTETSTRING; par_flistl: LENGTH; n :INTEGER)

Structured Type : FRQPARA

Derivation Path :

Encoding Variation :

Comments : Broadcast channel of cell A for RGSM.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD n), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	–		
arfcn	–		
maclength	'02'O		
mac_8n	par_ma2		
mac_7n	par_ma1		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		
flst	Frql_20_rgsm(par_freqlist, par_flistl)		frequency list, after time for cell A

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : FreqSDCCH8_r1
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : SDCCH8 channel of TC_26_6_10_2_2R for RGSM.

Element Name	Element Value	Element Encoding	Comments
hch	'0'B		non hopping
maio	−		mobile allocation index offset
hsn	−		hopping sequence number
spr	'00'B		'00'B − non hopping, if hch = 0, otherwise −
arfcn	INT_TO_BIT(972, 10)		
maclength	'00'O		
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 : FreqSDCCH8_r2
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : hopping with
indicates to the ARFCN_list={0, 80, 1005, 1010} of cell allocation A.

Element Name	Element Value	Element Encoding	Comments
hch	’1’B		hopping
maio	INT_TO_BIT((TSPX_MAIO MOD 4), 6)		
hsn	INT_TO_BIT(TSPX_HSN, 6)		
spr	−		
arfcn	−		
maclength	’01’O		
mac_8n	’10111000’B		0, 80, 958, 1010
mac_7n	−		
mac_6n	−		
mac_5n	−		
mac_4n	−		
mac_3n	−		
mac_2n	−		
mac_1n	−		
flst	Frql_20_A0Er		R−GSM

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Freq_rg8
Structured Type : FRQPARA
Derivation Path :
Encoding Variation :
Comments : hopping channel of TC_26_4_2_2 for GSM assignment and TC_26_10_2_5.

Element Name	Element Value	Element Encoding	Comments
hch	'1'B		hopping
maio	'000000'B		arbitrarily selected
hsn	'000100'B		arbitrarily selected
spr	−		
arfcn	−		
maclength	'01'O		
mac_8n	'00001110'B		mac64 − mac57
mac_7n	−		
mac_6n	−		
mac_5n	−		
mac_4n	−		
mac_3n	−		
mac_2n	−		
mac_1n	−		
flst	Frql_07		{20,30,50,70}

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql(par_flist, par_flistl : LENGTH)
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : parameterized frequncy list information element.

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	par_flistl		length of the contents
fl	par_flist		

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_short(flist : OCTETSTRING)
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : parameterized frequency short list information element.

Element Name	Element Value	Element Encoding	Comments
iei	'00000010'B		
iel	–		no length element for short list
fl	flist		9 octets

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_07
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : ARFCN_list = {20, 30, 50, 70} used in TC_26_6_4_2_2 and TC_26_10_5_2.

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'0000000000000020000020 00020080000'O		bit map 0 format, ARFCNs 20, 30, 50, 70

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_rgsm(par_flist: OCTETSTRING; par_flistl: LENGTH)
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Complete Cell Allocation in RGSM cases.

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	par_flistl		
fl	par_flist		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Frql_20_A0Er
Structured Type : FRQL
Derivation Path :
Encoding Variation :
Comments : Complete Cell Allocation of cell A in RGSM cases using of 1024 format.
ARFCN_list={0, 30, 40, 66, 958, 1010, 1015}.

Element Name	Element Value	Element Encoding	Comments
iei	'00000101'B		
iel	'10'O		length = 16
fl	'841EEA893339814000000000000000000'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 : Hlcmp_TS61
Structured Type : HLCMP
Derivation Path :
Encoding Variation :
Comments : High Layer Compatibility IE for teleservice 61.

Element Name	Element Value	Element Encoding	Comments
iei	'01111101'B		'01111101'B
iel	'02'O		
extb3	'1'B		extension bit ('1'B)
cs	'00'B		CCITT
in	'100'B		first HL characteristics identification to be used in the call.
pmpp	'01'B		High layer protocol profile
extb4	'?'B		extension bit, octet 4
hlci	'0000100'B		facsimile group 2/3
extb4a	*		extension bit, '1'B, octet 4a
ehlci	*		extended high layer characteristics identification

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : Hlcmp_TS62
Structured Type : HLCMP
Derivation Path :
Encoding Variation :
Comments : High Layer Compatibility IE for teleservice 62.

Element Name	Element Value	Element Encoding	Comments
iei	'01111101'B		'01111101'B
iel	'02'O		
extb3	'1'B		extension bit ('1'B)
cs	'00'B		CCITT
in	'100'B		first HL characteristics identification to be used in the call.
pmpp	'01'B		High layer protocol profile
extb4	'?'B		extension bit, octet 4
hlci	'0000100'B		facsimile group 2/3
extb4a	*		extension bit, '1'B, octet 4a
ehlci	*		extended high layer characteristics identification

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 2B2B2B2B2B2B'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 : IaRestOct_02
Structured Type : IARESTOCT
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
iei	–		
iaroct1	'2B2B2B2B2B2B2B2B2B2B 'O		p=00 format, 10 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	?		
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_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_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_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_04man
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping used in TC_26_6_4_2_2.

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'01'O		hopping
mac_8n	'00001110'B		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_281r
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping sequence with RGSM frequencies. The frequencies which are used in the mobile hopping sequence are {0, 80, 958, 1010} .

Element Name	Element Value	Element Encoding	Comments
iei	−		
iel	'01'O		
mac_8n	'10111000'B		
mac_7n	−		
mac_6n	−		
mac_5n	−		
mac_4n	−		
mac_3n	−		
mac_2n	−		
mac_1n	−		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_281e2
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with
indicates to the ARFCN_list={20,66,78} of E−GSM cell allocation with ARFCN_List={20,40,66,73,74,75,76,77,78,79,108,114}

Element Name	Element Value	Element Encoding	Comments
iei	−		
iel	'02'O		
mac_8n	'00000001'B		
mac_7n	'00000101'B		
mac_6n	−		
mac_5n	−		
mac_4n	−		
mac_3n	−		
mac_2n	−		
mac_1n	−		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_20_Be1(par_ma: BITSTRING)
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with par_ma

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'01'O		
mac_8n	par_ma		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_20_Be1iei(par_ma: BITSTRING)
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with par_ma

Element Name	Element Value	Element Encoding	Comments
iei	'01110010'B		
iel	'01'O		
mac_8n	par_ma		
mac_7n	–		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_20_Be2(par_ma1: BITSTRING; par_ma2: BITSTRING)
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with par_ma1 and par_ma2

Element Name	Element Value	Element Encoding	Comments
iei	–		
iel	'02'O		
mac_8n	par_ma1		
mac_7n	par_ma2		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MoblAllc_20_Be2iei(par_ma1: BITSTRING; par_ma2: BITSTRING)
Structured Type : MA
Derivation Path :
Encoding Variation :
Comments : hopping with par_ma1 and par_ma2

Element Name	Element Value	Element Encoding	Comments
iei	'01110010'B		
iel	'02'O		
mac_8n	par_ma1		
mac_7n	par_ma2		
mac_6n	–		
mac_5n	–		
mac_4n	–		
mac_3n	–		
mac_2n	–		
mac_1n	–		

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 : 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_03
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	'1'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	'110'B		6 results
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_03r1
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE containing 6 measurement results for RGSM.

Element Name	Element Value	Element Encoding	Comments
ba_used	'1'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	'110'B		6 results
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	'00001'B		arfcn=26
bsic_nc1	'001011'B		ncc=1, bcc=3
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	'00010'B		arfcn= 38
bsic_nc2	'001111'B		ncc=1, bcc= 7
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	'00100'B		arfcn= 960
bsic_nc3	'001001'B		ncc=1, bcc= 1
rxlev_nc4	?		received signal strength on nc4
bcchfrq_nc4	'00101'B		arfcn= 970
bsic_nc4	'001111'B		ncc=1, bcc= 7
rxlev_nc5	?		received signal strength on nc5
bcchfrq_nc5	'00110'B		arfcn= 990
bsic_nc5	'001101'B		ncc=1, bcc= 5
rxlev_nc6	?		received signal strength on nc6
bcchfrq_nc6	'00111'B		arfcn= 1020
bsic_nc6	'001101'B		ncc=1, bcc= 5

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_03r2
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE containing 6 measurement results for RGSM.

Element Name	Element Value	Element Encoding	Comments
ba_used	'1'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	'110'B		6 results
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	'00010'B		arfcn= 26
bsic_nc1	'001011'B		ncc=1, bcc= 3
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	'00011'B		arfcn= 38
bsic_nc2	'001111'B		ncc=1, bcc= 7
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	'00100'B		arfcn= 960
bsic_nc3	'001001'B		ncc=1, bcc= 1
rxlev_nc4	?		received signal strength on nc4
bcchfrq_nc4	'00101'B		arfcn= 970
bsic_nc4	'001111'B		ncc=1, bcc= 7
rxlev_nc5	?		received signal strength on nc5
bcchfrq_nc5	'00110'B		arfcn= 990
bsic_nc5	'001101'B		ncc=1, bcc= 5
rxlev_nc6	?		received signal strength on nc6
bcchfrq_nc6	'00001'B		arfcn= 0
bsic_nc6	'001001'B		ncc=1, bcc= 1

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_03r3
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE containing 6 measurement results for RGSM.

Element Name	Element Value	Element Encoding	Comments
ba_used	'1'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	'110'B		6 results
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	'00010'B		arfcn= 26
bsic_nc1	'001011'B		ncc=1, bcc= 3
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	'00011'B		arfcn= 38
bsic_nc2	'001111'B		ncc=1, bcc= 7
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	'00110'B		arfcn= 960
bsic_nc3	'001001'B		ncc=1, bcc= 1
rxlev_nc4	?		received signal strength on nc4
bcchfrq_nc4	'01000'B		arfcn= 970
bsic_nc4	'001111'B		ncc=1, bcc= 7
rxlev_nc5	?		received signal strength on nc5
bcchfrq_nc5	'01000'B		arfcn= 1020
bsic_nc5	'001101'B		ncc=1, bcc= 5
rxlev_nc6	?		received signal strength on nc6
bcchfrq_nc6	'00001'B		arfcn= 0
bsic_nc6	'001001'B		ncc=1, bcc= 1

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_03r4
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE containing 6 measurement results for RGSM.

Element Name	Element Value	Element Encoding	Comments
ba_used	'1'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	'110'B		6 results
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	'00010'B		arfcn= 26
bsic_nc1	'001011'B		ncc=1, bcc= 3
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	'00011'B		arfcn= 38
bsic_nc2	'001111'B		ncc=1, bcc= 7
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	'00110'B		arfcn= 960
bsic_nc3	'001001'B		ncc=1, bcc= 1
rxlev_nc4	?		received signal strength on nc4
bcchfrq_nc4	'00111'B		arfcn= 970
bsic_nc4	'001111'B		ncc=1, bcc= 1
rxlev_nc5	?		received signal strength on nc5
bcchfrq_nc5	'01000'B		arfcn= 990
bsic_nc5	'001111'B		ncc=1, bcc= 7
rxlev_nc6	?		received signal strength on nc6
bcchfrq_nc6	'00001'B		arfcn= 0
bsic_nc6	'001001'B		ncc=1, bcc= 1

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_03r5
Structured Type : MSRR
Derivation Path :
Encoding Variation :
Comments : A measurement results IE containing 6 measurement results for RGSM.

Element Name	Element Value	Element Encoding	Comments
ba_used	'1'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	'110'B		6 results
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	'00010'B		arfcn= 26
bsic_nc1	'001011'B		ncc=1, bcc= 3
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	'00011'B		arfcn= 38
bsic_nc2	'001111'B		ncc=1, bcc= 7
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	'00101'B		arfcn= 960
bsic_nc3	'001001'B		ncc=1, bcc= 1
rxlev_nc4	?		received signal strength on nc4
bcchfrq_nc4	'00110'B		arfcn= 970
bsic_nc4	'001111'B		ncc=1, bcc= 7
rxlev_nc5	?		received signal strength on nc5
bcchfrq_nc5	'00111'B		arfcn= 990
bsic_nc5	'001101'B		ncc=1, bcc= 5
rxlev_nc6	?		received signal strength on nc6
bcchfrq_nc6	'00001'B		arfcn= 0
bsic_nc6	'001001'B		ncc=1, bcc= 1

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : MsrResult_04r
Structured Type : MSRR
Derivation Path : MsrResult_03.
Encoding Variation :
Comments : A measurement results IE containing 3 measurement results for RGSM

Element Name	Element Value	Element Encoding	Comments
no_nc	’100’B		3 results
rxlev_nc1	?		received signal strength on nc1
bcchfrq_nc1	’00001’B		arfcn= 26
bsic_nc1	’001011’B		ncc=1, bcc= 3
rxlev_nc2	?		received signal strength on nc2
bcchfrq_nc2	’00011’B		arfcn= 960
bsic_nc2	’001001’B		ncc=1, bcc= 1
rxlev_nc3	?		received signal strength on nc3
bcchfrq_nc3	’00110’B		arfcn= 990
bsic_nc3	’001101’B		ncc=1, bcc= 5

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_pl(re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5)
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	cba		
re	re		
acc_2	acc		
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 : SI1RestOctetDef
Structured Type : SI1RO
Derivation Path :
Encoding Variation :
Comments : Default value for L3 testing.

Element Name	Element Value	Element Encoding	Comments
nchi	'0'B		
nchp	−		
spb	'0101011'B		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI3RestOctetDef
Structured Type : SI3RO
Derivation Path :
Encoding Variation :
Comments : Default value for L3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	'0'B		L
cbq	–		
cro	–		
to	–		
pt	–		
poi	'0'B		L
po	–		
si2ti	'1'B		L
ecsc	'0'B		L
si9i	'1'B		L
where	–		
gprsi	–		
racolour	–		
si13p	–		
spb	'011'B		
spd	'2B2B2B'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI3RestOctetPara_pl(cbq, si2ti, ecsc : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
Structured Type : SI3RO
Derivation Path :
Encoding Variation :
Comments : Default value for L 3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	'1'B		
cbq	cbq		
cro	cro		
to	to		
pt	pt		
poi	'1'B		
po	po		
si2ti	si2ti		
ecsc	ecsc		High = 0
si9i	'0'B		L
where	–		
gprsi	–		GPRS supported
racolour	–		routing area colour code
si13p	–		
spb	'11'B		
spd	'2B'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI4RestOctetDef
Structured Type : SI4RO
Derivation Path :
Encoding Variation :
Comments : Default value for L3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	'0'B		L
cbq	–		
cro	–		
to	–		
pt	–		
poi	'0'B		L
po	–		
gprsi	'1'B		L
racolour	–		
si13p	–		
spb	'01011'B		
spd	'2B2B2B2B2B2B2B2B'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI4RestOctetPara(cbq : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
Structured Type : SI4RO
Derivation Path :
Encoding Variation :
Comments : Default value for L 3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	'1'B		H
cbq	cbq		
cro	cro		
to	to		
pt	pt		
poi	'1'B		H
po	po		
gprsi	'0'B		L
racolour	–		
si13p	–		
spb	'1011'B		
spd	'2B2B2B2B2B2B2B'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI4RestOctet_CBMS
Structured Type : SI4RO
Derivation Path : SI4RestOctetDef.
Encoding Variation :
Comments : Rest octets for SMSCB SysInf4 message.

Element Name	Element Value	Element Encoding	Comments
spd	'2B2B2B2B2B'O		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : SI6RestOctetDef
Structured Type : SI6RO
Derivation Path :
Encoding Variation :
Comments : Default value for L3 testing.

Element Name	Element Value	Element Encoding	Comments
pchnchi	'0'B		L
pchrestr	–		
nln_saccch	–		
cpi	–		
call_priority	–		
nlns	–		
vbs_vgcsi	'1'B		H
inband_notification	'0'B		no inband notification
inband_paging	'0'B		no inband paging
spb	'1101'B		
spd	'2B2B2B2B2B2B'O		

Detailed Comments : 7 octets

Structured Type Constraint Declaration

Constraint Name : SI7RestOctetPara(cbq : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
Structured Type : SI7RO
Derivation Path :
Encoding Variation :
Comments : Default value for L3 testing.

Element Name	Element Value	Element Encoding	Comments
pi	'1'B		H
cbq	cbq		
cro	cro		
to	to		
pt	pt		
poi	'1'B		H
po	po		
gprsi	'0'B		L
racolour	–		
si13p	–		
spb	'1011'B		
spd	'2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B2B'O		

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 : 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_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 : V21orAbt1
Structured Type : MODEMTYPE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
V22bisOrV26ter	('00001'B, '01000'B)		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : V22orAbt1
Structured Type : MODEMTYPE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
V22bisOrV26ter	('00010'B, '01000'B)		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : V23orAbt1
Structured Type : MODEMTYPE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
V22bisOrV26ter	('00100'B, '01000'B)		

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 :

Structured Type Constraint Declaration

Constraint Name : V22bisOrV26terOrAbt1
Structured Type : MODEMTYPE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
V22bisOrV26ter	('00011'B, '00101'B, '01000'B)		

Detailed Comments :

Structured Type Constraint Declaration

Constraint Name : V32orAbt1
Structured Type : MODEMTYPE
Derivation Path :
Encoding Variation :
Comments :

Element Name	Element Value	Element Encoding	Comments
V22bisOrV26ter	('00110'B, '01000'B)		

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 : CCStRcv(ch: LOGICCH; pdu : CCST_PDU)
ASP Type : DL_DatInCcst
Derivation Path :
Comments : To match any received STATUS message with TI = Ti.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	
fn	?	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CCStESnd(ch: LOGICCH; pdu: CCST_ENQ_PDU)
ASP Type : DL_DatRqCcstEnq
Derivation Path :
Comments : To send a STATUS ENQUIRY message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

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 : ChReq_01(ch : LOGICCH; pdu : CH_RQ_PDU)
ASP Type : DL_RacInChRq
Derivation Path :
Comments : To receive a CHANNEL REQUEST message in cell B

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
fn	?	
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CMSerAcp(ch : LOGICCH; pdu: CMS_ACP_PDU)
ASP Type : DL_DatRqCmsAcp
Derivation Path :
Comments : To send a CM service accept message.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	DCCH
msg	pdu	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : CMSerRej(ch : LOGICCH; pdu: CMS_REJ_PDU)
ASP Type : DL_DatRqCmsRej
Derivation Path :
Comments : reject cause = "service or option not available, unspecified"

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	
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 : 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 :

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 : FrqRedfSnd(ch : LOGICCH; pdu : FRQRE_PDU)
ASP Type : DL_DatRqFrqre
Derivation Path :
Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	pdu	

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 : 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 : 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 : PgRes(ch : LOGICCH; pdu : PG_RES_PDU)
ASP Type : DL_EstInPgRes
Derivation Path :
Comments : To match any received PAGING RESPONSE message on the channel ‘ch‘.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	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 : SysInfo1_pl(ch : LOGICCH; cchd : CCHD; re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5)
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_pl(cchd, maxtx, txint, re, cba, acc)	

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 : SysInfo2_pl(ch: LOGICCH; bcchfl: NCD; nccp : NCCP; re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5)

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_pl(bcchfl, nccp, re, cba, maxtx, txint, acc)	

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 : SysInfo2bis_egsm(ch: LOGICCH; freqlst : NCD)
ASP Type : DL_UdatRqSysinfo2bis
Derivation Path :
Comments : SYSTEM INFORMATION 2bis under EGSM with the ARFCN list = freqlst.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	BCCH
msg	SysInf2bis_egsm(freqlst)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo2ter(ch: LOGICCH; fl: NCD2)
ASP Type : DL_UdatRqSysinfo2ter
Derivation Path :
Comments : SYSTEM INFORMATION 2ter with ther ARFCN list = fl.

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	0
logic_ch	ch	BCCH
msg	SysInf2ter(fl)	

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 : SysInfo3_pl(ch : LOGICCH; ci : CI; mcc, mnc, lac : OCTETSTRING; ccd : CCD; co : CO; crh, mtmc : INTEGER; neci, acs, re, cba, cbq, si2ti, ecsc : B_1; maxtx, po : B_2; to : B_3; txint : B_4; acc, pt : B_5; ram, cro : B_6)

ASP Type : DL_UdatRqSysinfo3

Derivation Path :

Comments :

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf3_pl(ci, mcc, mnc, lac, ccd, co, crh, mtmc, neci, acs, re, cba, cbq, si2ti, ecsc, maxtx, po, to, txint, acc, pt, ram, cro)	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 : SysInfo4_pl(ch : LOGICCH; mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci, acs, re, cba, cbq : B_1; maxtx, po : B_2; to : B_3; txint : B_4; acc, pt : B_5; ram, cro : B_6)

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_pl(mcc, mnc, lac, crh, mtmc, neci, acs, re, cba, cbq, maxtx, po, to, txint, acc, pt, ram, cro)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo4_pl_R(ch : LOGICCH; mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci, acs, re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5; ram : B_6)

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_pl_R(mcc, mnc, lac, crh, mtmc, neci, acs, re, cba, maxtx, txint, acc, ram)	

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 : SysInfo7(ch : LOGICCH; cbq : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
ASP Type : DL_UdatRqSysinfo7
Derivation Path :
Comments : To send a default SYSTEM INFORMATION TYPE 7 message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf7(cbq, po, to, pt, cro)	

Detailed Comments :

ASP Constraint Declaration

Constraint Name : SysInfo8(ch : LOGICCH; cbq : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
ASP Type : DL_UdatRqSysinfo8
Derivation Path :
Comments : To send a default SYSTEM INFORMATION TYPE 8 message

Parameter Name	Parameter Value	Comments
sapi	C_Sap0	
logic_ch	ch	
msg	SysInf8(cbq, po, to, pt, cro)	

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 :

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 : AlertingInd_02(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		transaction identifier
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_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		
dmsc	–		Phase 2+
ch1mod	chmod		
modchs2	–		Phase 2+
modchs3	–		Phase 2+
modchs4	–		Phase 2+
modchs5	–		Phase 2+
modchs6	–		Phase 2+
modchs7	–		Phase 2+
modchs8	–		Phase 2+
ch2d_at	–		
ch2mod	–		
ma_at	–		
strt	strt		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	cphms		
vgcstmi	–		Phase 2+

Detailed Comments : used iin CC testing

PDU Constraint Declaration

Constraint Name : AsgnCmd_22_Ar1 (ts_ccch: BITSTRING; par_chtype : CH_TDMA; par_flist:OCTETSTRING; par_flistl: LENGTH; n : INTEGER)

PDU Type : ASS_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An ASSIGNMENT COMMAND message assigning TCH/F non FH channel in cell A for RGSM.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101110'B		
ch1d_at	ChDescrp_fh(par_chtype, ts_ccch, TSPX_TscDef, INT_TO_BIT((TSPX_MAIO MOD n),6), INT_TO_BIT(TSPX_HSN, 6))		assign TCH/F + ACCHs
pcmd	Pcmd_19('01001'B)		power level = 19, for GSM
frql_at	Frql_20_rgsm (par_flist, par_flistl)		
cchd	–		
dmsc	–		Phase 2+
ch1mod	–		
modchs2	–		Phase 2+
modchs3	–		Phase 2+
modchs4	–		Phase 2+
modchs5	–		Phase 2+
modchs6	–		Phase 2+
modchs7	–		Phase 2+
modchs8	–		Phase 2+
ch2d_at	–		
ch2mod	–		
ma_at	–		
strt	–		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	–		
vgcstmi	–		Phase 2+

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AsgnCmd_22_Ae2(ts_ccch: BITSTRING; par_chtype : CH_TDMA; par_cchd: OCTETSTRING; par_ma:BITSTRING; n : INTEGER)

PDU Type : ASS_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An ASSIGNMENT COMMAND message assigning TCH/F non FH channel in cell A for EGSM.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101110'B		
ch1d_at	ChDescrp_fh(par_chtype, ts_ccch, TSPX_TscDef, INT_TO_BIT((TSPX_MAIO MOD n),6), INT_TO_BIT(TSPX_HSN, 6))		assign TCH/F + ACCHs or SDCCH8
pcmd	Pcmd_19('01001'B)		power level = 19, for GSM
frql_at	–		
cchd	CellChDes_20_Be(par_cchd)		
dmsc	–		Phase 2+
ch1mod	–		
modchs2	–		Phase 2+
modchs3	–		Phase 2+
modchs4	–		Phase 2+
modchs5	–		Phase 2+
modchs6	–		Phase 2+
modchs7	–		Phase 2+
modchs8	–		Phase 2+
ch2d_at	–		
ch2mod	–		
ma_at	MoblAllc_20_Be1iei(par_ma)		
strt	–		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	–		
vgcstmi	–		Phase 2+

Detailed Comments :

PDU Constraint Declaration

Constraint Name : AsgnCmd_22_Ar3(ts_ccch: BITSTRING; par_chtype : CH_TDMA; par_cchd: OCTETSTRING; par_ma1:BITSTRING; par_ma2:BITSTRING; n : INTEGER)

PDU Type : ASS_CMD_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : An ASSIGNMENT COMMAND message assigning TCH/F non FH channel in cell A for RGSM.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101110'B		
ch1d_at	ChDescrp_fh(par_chtype, ts_ccch, TSPX_TscDef, INT_TO_BIT((TSPX_MAIO MOD n),6), INT_TO_BIT(TSPX_HSN, 6))		assign TCH/F + ACCHs or SDCCH8
pcmd	Pcmd_19('01001'B)		power level = 19, for GSM
frql_at	–		
cchd	CellChDes_20_Be (par_cchd)		
dmsc	–		Phase 2+
ch1mod	–		
modchs2	–		Phase 2+
modchs3	–		Phase 2+
modchs4	–		Phase 2+
modchs5	–		Phase 2+
modchs6	–		Phase 2+
modchs7	–		Phase 2+
modchs8	–		Phase 2+
ch2d_at	–		
ch2mod	–		
ma_at	MoblAllc_20_Be2iei (par_ma1, par_ma2)		
strt	–		
frql_bt	–		
ch1d_bt	–		
ch2d_bt	–		
frqchs_bt	–		
ma_bt	–		
cphms	–		
vgcstmi	–		Phase 2+

Detailed Comments :

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 : 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
pr	−		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CCStatus_01(Ti : TI)
PDU Type : CCST_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CC STATUS message to match any received CC STATUS.

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
ccpd	'0011'B		CC protocol discriminator
mt	'0?111101'B		message type
cau	?		cause
cst	?		call state
acst	*		auxiliary status

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CCStatusEq_01(Ti : TI)
PDU Type : CCST_ENQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A STATUS ENQUIRY message

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
ccpd	’0011’B		
mt	’00110100’B		

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
barng	–		phase 2+
gchd	–		phase 2+
gckn	–		phase 2+
gprsr	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ChRequest_01
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A CHANNEL REQUEST message containing establishment cause = answer to paging.

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	’100?????’B		

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_09
PDU Type : CH_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : To match the received CHANNEL REQUEST message with establishment cause = '000'B.

Field Name	Field Value	Field Encoding	Comments
ecau_rrf	'000?????'B		location updating

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 : CMServiceAcp_01
PDU Type : CMS_ACP_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : a CM SERVICE ACCEPT message

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		mm
mt	'00100001'B		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : CMServiceRej_30(par: REJCAU)
PDU Type : CMS_REJ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : reject cause = 'service or option not available, unsepcified"

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
mmpd	'0101'B		MM protocol discriminator
mt	'00100010'B		message type
mmcau	par		

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	?		
pr	*		

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		
pr	*		

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 : FreqRedef_21(ts_ccch: BITSTRING; par_chtype : CH_TDMA; par_ma:BITSTRING; par_stime:STRT; n : INTEGER)

PDU Type : FRQRE_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : Frequency Redefinition

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00010100'B		message type
chd	ChDescrp_fh(par_chtype, ts_ccch, TSPX_TscDef, INT_TO_BIT((TSPX_MAIO MOD n), 6), INT_TO_BIT(TSPX_HSN, 6))		
ma	MoblAllc_20_Be1(par_ma)		
strt	par_stime		
cchd	–		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : FreqRedef_22(ts_ccch: BITSTRING; par_chtype : CH_TDMA; par_cchd: OCTETSTRING; par_ma1:BITSTRING; par_ma2:BITSTRING; par_stime:STRT; n : INTEGER)

PDU Type : FRQRE_PDU

Derivation Path :

Encoding Rule Name :

Encoding Variation :

Comments : Frequency Redefinition

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00010100'B		message type
chd	ChDescrp_fh(par_chtype, ts_ccch, TSPX_TscDef, INT_TO_BIT((TSPX_MAIO MOD n), 6), INT_TO_BIT(TSPX_HSN, 6))		
ma	MoblAllc_20_Be2(par_ma1, par_ma2)		
strt	par_stime		
cchd	CellChDes_20_Be(par_cchd)		

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		
dmsc	–		Phase 2+
ch1mod	chmod		
modchs2	–		Phase 2+
modchs3	–		Phase 2+
modchs4	–		Phase 2+
modchs5	–		Phase 2+
modchs6	–		Phase 2+
modchs7	–		Phase 2+
modchs8	–		Phase 2+
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		
vgcstmi	–		Phase 2+

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_02
PDU Type : HOFL_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A HANDOVER FAILURE message matching any abnormal release RR cause.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
rrpd	'0110'B		
mt	'00101000'B		
rrcau	('00000001'B, '00000010'B, '00000011'B, '00000100'B, '01101111'B)		abnormal release causes

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		
dm_tbf	Dm_TBF_def		
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
pchd	–		
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_01Def_R(Rr: BITSTRING; Fn: FN; sub : B_3; 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		
dm_tbf	Dm_TBF_def		
pm	Pm(C_normal_paging)		normal mode
chd	ChDescrp_nfh(INT_TO_BIT((8 + BIT_TO_INT(sub)), 5), slot, tsc, par_arfcn)		SDCCH/4 subchannel
pchd	–		
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_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		
dm_tbf	Dm_TBF_def		
pm	Pm(pgm)		paging mode
chd	ChDescrp_nfh(type, slot, tsc, arfcn)		Traffic channel, non hopping
pchd	–		
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_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		
dm_tbf	Dm_TBF_def		
pm	Pm(C_normal_paging)		normal mode
chd	ChDescrp_fh(par_chtype, ts_ccch, tsc, maio, hsn)		
pchd	–		
rqr	Rqr2(Rr, Fn)		
ta	ta		
ma	ma		
strt	–		
iaroct	IaRestOct(iel)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : ImmAsgn_27(Rr: BITSTRING; Fn: FN;ts_ccch: BITSTRING; par_arfcn: INTEGER; ta : TA)
PDU Type : IMMASS_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An IMMEDIATE ASSIGNMENT message to assign SDCCH/8 NonFH channel in any cell.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'2D'O		L2 pseudo length = 11 octets
ski	'0000'B		
rrpd	'0110'B		
mt	'00111111'B		
dm_tbf	Dm_TBF_def		
pm	Pm(C_normal_paging)		normal mode
chd	ChDescrp_sdcch8_nfh(ts_ccch, TSPX_TscDef, TSPX_SDCCH8SubDef, par_arfcn)		SDCCH/8, non hopping,
pchd	–		
rqr	Rqr2(Rr, Fn)		
ta	ta		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_E_01(Rr: BITSTRING; Fn: FN;ts_ccch: BITSTRING; 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/8 NonFH channel in any cell.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'2D'O		L2 pseudo length = 11 octets
ski	'0000'B		
rrpd	'0110'B		
mt	'00111111'B		
dm_tbf	Dm_TBF_def		
pm	Pm(C_normal_paging)		normal mode
chd	ChDescrp_sdcch8_nfh(ts_ccch, tsc, TSPX_SDCCH8SubA, par_arfcn)		SDCCH/8, non hopping,
pchd	–		
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_R_02(Rr: BITSTRING; Fn: FN;ts_ccch: BITSTRING; tsc : TSC; ta : TA)
PDU Type : IMMASS_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : An IMMEDIATE ASSIGNMENT message to assign SDCCH/8 FH channel in RGSM cases.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'31'O		L2 pseudo length = 12 octets
ski	'0000'B		
rrpd	'0110'B		
mt	'00111111'B		
dm_tbf	Dm_TBF_def		
pm	Pm(C_normal_paging)		
chd	ChDescrp_sdcch8_fh(ts_ccch, tsc, TSPX_SDCCH8SubB, INT_TO_BIT((TSPX_MAIO MOD 4), 6), INT_TO_BIT(TSPX_HSN, 6))		
pchd	–		
rqr	Rqr2(Rr, Fn)		
ta	ta		
ma	MoblAllc_281r		
strt	–		
iaroct	IaRestOct_02		10 Bytes

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 : LocUpdtReq_05
PDU Type : LUP_RQ_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : to match any LOCATION UPDATING REQUEST message.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		
mmpd	'0101'B		
mt	'0?001000'B		
cphksn	?		no check required
lutype	?		
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_03r(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 : MsrReport_04r
PDU Type : MSR_RPT_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A measurement report containing 4 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	MsrResult_04r		

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 : 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		
p1roct2	–		phase 2+

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_04(Ti : TI)
PDU Type : REL_COM_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A RELEASE COMPLETE message containing cause value #1. n −> ms

Field Name	Field Value	Field Encoding	Comments
ti	Ti		
cc_sspd	’0011’B		CC protocol discriminator
mt	’00101010’B		
cau	Cause_10		cause value #1
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	Bcap_Speech_MT		speech
bcap2	−		
fie	−		
pi	−		
sig	−		
cgpn	−		
cgps	−		
cdpn	−		
cdps	−		
llcri	−		
llcmp1	−		
llcmp2	−		
hlcri	−		
hlcmp1	−		
hlcmp2	−		
uu	−		
pr	−		
alert	−		

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_04(bcap: BCAP)
PDU Type : ESETUP_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A Emergency call SETUP message with/without speech bearer capability. ms −> n.

Field Name	Field Value	Field Encoding	Comments
ti	?		
ccpd	’0011’B		
mt	’0?001110’B		message type
bcap	bcap IF_PRESENT		full rate emergency call if not present

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	–		
pr	–		
alert	–		

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	–		
pr	–		
alert	–		

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	SI1RestOctetDef		'2B'O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf1_pl(cchd : CCHD; re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5)
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		
rachcp	RachCntrlPara_pl(re, cba, maxtx, txint, acc)		rach control parameters
si1roct	SI1RestOctetDef		'2B'O

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 for EGSM testing

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		
rachcp	RachCntrlPara_r01		rach control parameters
si2bisroct	'2B'O		SI2bis rest octet

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf2bis_egsm(freqlst : NCD)
PDU Type : SYSINFO2bis_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : SYSTEM INFORMATION 2bis under EGSM with the ARFCN list = {985, 989, 995, 1010, 1014}.

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	freqlst		EGSM freq list
rachcp	RachCntrlPara_r01		rach control parameters
si2bisroct	'2B'O		SI2bis rest octet

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf2_pl(bcchfl: NCD; nccp : NCCP; re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5)
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_pl(re, cba, maxtx, txint, acc)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf2ter(fl : NCD2)
PDU Type : SYSINFO2ter_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : SYSTEM INFORMATION 2ter with the ARFCN list = fl.

Field Name	Field Value	Field Encoding	Comments
l2_pl	'49'O		L2 pseudo length = 18
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00000011'B		message type
xbcchfl	fl		
si2terroct	'2B2B2B2B'O		SI2bis rest octets

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		'2B2B2B2B'O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf3_pl(ci : CI; mcc, mnc, lac : OCTETSTRING; ccd : CCD; co : CO; crh, mtmc : INTEGER; neci, acs, re, cba, cbq, si2ti, ecsc : B_1; maxtx, po : B_2; to : B_3; txint : B_4; acc, pt : B_5; ram, cro : B_6)

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)		
ccd	ccd		
co	co		
csp	CellSelPara_pl(crh, mtmc, neci, acs, ram)		
rachcp	RachCntrlPara_pl(re, cba, maxtx, txint, acc)		
si3roct	SI3RestOctetPara_pl(pt, cro, to, po, cbq, ecsc, si2ti)		

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		'2B2B2B2B2B2B2B2B2B2B'O

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 = 16
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		'2B2B2B2B2B2B'O

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf4_pl(mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci, acs, re, cba, cbq : B_1; maxtx, po : B_2; to : B_3; txint : B_4; acc, pt : B_5; ram, cro : B_6)

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_pl(crh, mtmc, neci, acs, ram)		
rachcp	RachCntrlPara_pl(re, cba, maxtx, txint, acc)		
cbchd	–		
cbchma	–		
si4roct	SI4RestOctetPara(cbq, po, to, pt, cro)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf4_pl_R(mcc, mnc, lac : OCTETSTRING; crh, mtmc : INTEGER; neci, acs, re, cba : B_1; maxtx : B_2; txint : B_4; acc : B_5; ram : B_6)

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_pl(crh, mtmc, neci, acs, ram)		
rachcp	RachCntrlPara_pl(re, cba, maxtx, txint, acc)		
cbchd	–		
cbchma	–		
si4roct	SI4RestOctetDef		

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

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		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf5ter(bcchfl: NCD2)
PDU Type : SYSINFO5ter_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : SYSTEM INFORMATION TYPE 5ter containing partial neighbour cells description.

Field Name	Field Value	Field Encoding	Comments
ski	'0000'B		skip identifier
rrpd	'0110'B		RR protocol discriminator
mt	'00000110'B		message type
xbcchfl	bcchfl		

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
si6roct	SI6RestOctetDef		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf7(cbq : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
PDU Type : SYSINFO7_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SYSTEM INFORMATION TYPE 7 message containing default values

Field Name	Field Value	Field Encoding	Comments
l2_pl	'09'O		value = 2
ski	'0000'B		
rrpd	'0110'B		
mt	'00011111'B		
si7roct	SI7RestOctetPara(cbq, po, to, pt , cro)		

Detailed Comments :

PDU Constraint Declaration

Constraint Name : SysInf8(cbq : B_1; po : B_2; to : B_3; pt : B_5; cro : B_6)
PDU Type : SYSINFO8_PDU
Derivation Path :
Encoding Rule Name :
Encoding Variation :
Comments : A SYSTEM INFORMATION TYPE 8 message containing default values

Field Name	Field Value	Field Encoding	Comments
l2_pl	'09'O		value = 2
ski	'0000'B		
rrpd	'0110'B		
mt	'00011000'B		
si8roct	SI7RestOctetPara(cbq, po, to, pt , cro)		

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_20_21_1R

Group : CellSelection/

Purpose : 1. To verify that:
The MS shall be able to select the correct (forth strongest) cell and be able to respond to paging on that cell within 30 seconds of switch on, when the three strongest cells are not suitable. This assumes that a valid SIM, with PIN disabled and ideal radio conditions; GSM 05.08, 6.1.

2. To verify that:
2.1 The MS does not select a cell of a PLMN which is not the selected PLMN.
2.2 The MS does not select a cell which is "barred".
2.3 The MS does not select a cell with C1<0.

3. To verify that the MS selects suitable cells in descending order of received signal strength.

4. To verify that the MS does not select a cell with C1<0.

Configuration :

Default : OtherEventsFail

Comments : This test case achieves the test purpose in one phase rather than the two described in ETS 300−607−2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 985)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers(C_CellA, C_CellB, C_CellC, C_CellD, "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(33000)			
9	CS_0 001	?TIMEOUT T_dly		F	
10	CS_0 002	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
11	CS_0 003	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
12	CS_0 004	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
13		L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)		
14		+MM_PwrOrSimOff(C_SIMIn)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
15		(TCV_Null := OM_StopCell(C_CellB))			
16		+ltree_StartCarrier5			
17		+PageOnCarriers("dummy", "dummy", "dummy", "dummy", C_CellE, "dummy")			
18		+MM_PwrOrSimOn(C_SIMIn)			
19		START T_dly(33000)			
20	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)		
21		->Label1			
22		L?DL_RacInChRq	ChReq_01(C_RACH_E_1, ChRequest_02)		
23		->Label1			
24	CS_0005	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
25	CS_0006	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
26	CS_0007	?TIMEOUT T_dly		P	
		ltree_StartCarriers			
27		+ltree_StartCarrier1			
28		+ltree_StartCarrier2			
29		+ltree_StartCarrier3			
30		+ltree_StartCarrier4			
		ltree_StartCarrier1			
31		+Config_BCCH_CCCH_A_1(48, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(985), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
32		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '1'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
		ltree_StartCarrier2			
33		+Config_BCCH_CCCH_B_1(38, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
34		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '101011'B, '000000'B)			
		ltree_StartCarrier3			
35		+Config_BCCH_CCCH_C_1(43, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(18), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
36		+SysInfoSending_fh_pl_R(C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellC, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
		ltree_StartCarrier4			
37		+Config_BCCH_CCCH_D_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(960), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
38		+SysInfoSending_fh_pl_R(C_SCH_D, C_BCCH_D_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellD, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010110'B, '000000'B)			
		ltree_StartCarrier5			
39		+Config_BCCH_CCCH_E_1(28, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(41), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
40		+SysInfoSending_fh_pl_R(C_SCH_E, C_BCCH_E_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellE, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '001100'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_2R

Group : CellSelection/

Purpose : 1. The MS shall:

1.1 The MS shall search all RF channels in the system, take readings of received RF signal strength on each RF channel, and calculate the received level average for each.

1.2 The averaging is based on at least five measurement samples per RF carrier spread over Tav (3 to 5 s).

1.3 The measurement samples from the different RF carriers being spread evenly during this period.

2. These quantities are termed the "receive level averages", shall be unweighted averages of the received signal strengths measured in dBm.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 956)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers(C_CellA, C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		(TCV_Null := OM_VaryRFOfCell(C_BCCH_B_1, TSPX_bTav, 28, TSPX_aTav, 58))			
8		+MM_PwrOrSimOn(C_SIMIn)			
9		START T_dly(33000)			
10	CS_0008	?TIMEOUT T_dly		F	
11	CS_0009	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
12		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
13	CS_0010	(TCV_Null := OM_StopVaryRFOfCell(C_BCCH_B_1))		P	
		ltree_StartCarriers			
14		+ltree_StartCarrier1			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
15		+ltree_StartCarrier2 ltree_StartCarrier1			
16		+Config_BCCH_CCCH_A_1(23, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
17		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '001010'B, '000000'B) ltree_StartCarrier2			
18		+Config_BCCH_CCCH_B_1(58, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(9), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
19		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '001010'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_4R

Group : CellSelection/

Purpose : 1. To verify that the MS shall be able to calculate the path loss criterion parameter C2 used for cell reselection.

2. To verify that the MS can correctly calculate the C2 parameter correctly when the CELL_RESELECT_OFFSET, TEMPORARY_OFFSET and PENALTY_TIME parameters are used.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_53e, TCV_bcchfl_DE := BcchFreqLst_55_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers1_and_2			
6		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(50000)			
9	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
10		−>Label1			
11	CS_0011	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
12		?TIMEOUT T_dly			
13		(TCV_Null := OM_ChangeRFOfCell(C_CellB, 54))			
14		START T_dly(20000)			
15	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_01)		
16		−>Label2			
17	CS_0012	?TIMEOUT T_dly		F	
18		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
19		CANCEL T_dly			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
20		(TCV_Null := OM_PgFill(C_CellB, PgReqTp1Norm))			
21		+Pause(20000)			
22		+ltree_StartCarriers3_and_4			
23		+PageOnCarriers("dummy", "dummy", C_CellC, C_CellD, "dummy", "dummy")			
24		START T_dly(38000)			
25	Label3	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_01)		
26		->Label3			
27		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_01)		
28		->Label3			
29	CS_0013	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
30	CS_0014	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)	F	
31		?TIMEOUT T_dly			
32		START T_dly(52000)			
33	Label4	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_01)		
34		->Label4			
35		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_01)		
36		->Label4			
37	CS_0015	?TIMEOUT T_dly		F	
38	CS_0016	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)	F	
39	CS_0017	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	P	
		ltree_StartCarriers1_and_2			
40		(TCV_PO := '00'B)			
41		+ltree_StartCarrier1			
42		(TCV_PO := '00'B)			
43		+ltree_StartCarrier2			
		ltree_StartCarriers3_and_4			
44		(TCV_PO := '00'B)			
45		+ltree_StartCarrier3			
46		+ltree_StartCarrier4			
		ltree_StartCarrier1			
47		+Config_BCCH_CCCH_A_1(53, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
48		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_53e, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, TCV_PO, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '011110'B, '000000'B)			
		ltree_StartCarrier2			
49		+Config_BCCH_CCCH_B_1(43, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(9), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
50		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_53e, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, TCV_PO, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '11111'B, '001010'B, '001000'B)			
		ltree_StartCarrier3			
51		+Config_BCCH_CCCH_C_1(48, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(18), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
52		+SysInfoSending_fh_pl_R(C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellC, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_53e, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, TCV_PO, C_BABR_0, C_cch_1nonComb, C_BPM_3, '010'B, '00000'B, '00001'B, '011001'B, '001010'B)			
		ltree_StartCarrier4			
53		+Config_BCCH_CCCH_D_1(48, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
54		+SysInfoSending_fh_pl_R(C_SCH_D, C_BCCH_D_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellD, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_53e, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, TCV_PO, C_BABR_0, C_cch_1nonComb, C_BPM_3, '010'B, '00000'B, '00010'B, '011001'B, '001010'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_5R

Group : CellSelection/

Purpose : 1. To verify that the MS shall be able to calculate the path loss criterion parameter C2 used for cell reselection.

2. Whilst in idle mode, an MS shall continue to monitor all BCCH carriers as indicated by the BCCH allocation.

3. Mobile stations shall treat all ARFCNs in the set (0..1023) as valid ARFCN values even if the mobile station is unable to transmit or receive on that ARFCN.

4. The MS shall correctly decode parameters transmitted in the system information type 2 ter message.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_57e, TCV_bcchfl_DE := BcchFreqLst_56_R , TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_1, C_BPM_2, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers("dummy", C_CellB, C_CellC, "dummy", "dummy", "dummy")			
7		+ltree_continue			
8		+ltree_RGSMOnly			
		ltree_continue			
9		+MM_PwrOrSimOn(C_SIMIn)			
10		START T_dly(50000)			
11	CS_0018	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
12	CS_0019	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
13		?TIMEOUT T_dly			
14		(TCV_Null := OM_ChangeRFOfCell(C_CellB, 42))			
15		START T_dly(20000)			
16		L?DL_RacInChRq CANCEL T_dly	ChReq_01(C_RACH_B_1, ChRequest_02)		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
17		(TCV_Null := OM_PgFill(C_CellB, PgReqTp1Norm))			
18	CS_0020	?TIMEOUT T_dly		(F)	
19		(TCV_Null := OM_PgFill(C_CellB, PgReqTp1Norm))			
		ltree_RGSMOnly			
20		START T_dly(30000)			
21	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
22		−>Label1			
23		?TIMEOUT T_dly			
24		(TCV_Null := OM_ChangeRFOfCell(C_CellC, 60))			
25		START T_dly(20000)			
26	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
27		−>Label2			
28	CS_0021	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	(P)	
29	CS_0022	?TIMEOUT T_dly		(F)	
		ltree_StartCarriers			
30		+ltree_StartCarrier1			
31		+ltree_StartCarrier2			
32		+ltree_StartCarrier3			
		ltree_StartCarrier1			
33		+Config_BCCH_CCCH_A_ho_1(53, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_1, C_cch_1nonComb, C_BPM_2, 0, 0)			
34		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_57e, BcchFreqLst_56_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’0’B, ’1’B, ’00’B, C_BABR_1, C_cch_1nonComb, C_BPM_2, ’000’B, ’00000’B, ’00000’B, ’010100’B, ’000000’B)			
		ltree_StartCarrier2			
35		+Config_BCCH_CCCH_B_ho_1(32, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(9), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_1, C_cch_1nonComb, C_BPM_2, 0, 27)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
36		+SysInfoSending_fh_pl_R2(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52e_R, BcchFreqLst2_65_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, C_SI2ter, '1'B, C_SI7_8, '00'B, C_BABR_1, C_cch_1nonComb, C_BPM_2, '000'B, '00000'B, '00000'B, '010100'B, '001000'B)			
		ltree_StartCarrier3			
37		+Config_BCCH_CCCH_C_1(40, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_1, C_cch_1nonComb, C_BPM_2)			
38		+SysInfoSending_fh_pl_R(C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellC, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_1, C_cch_1nonComb, C_BPM_2, '000'B, '00000'B, '00000'B, '011011'B, '000101'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_6R

Group : CellSelection/

Purpose : 1. At least every 5 s the MS shall calculate the value of C1 and C2 for the serving cell and recalculate C1 and C2 values for non serving cells (if necessary). The MS shall then check whether:

1.1 The calculated value of C2 for a non serving suitable cell exceeds the value of C2 for the serving cell for a period of 5 seconds.

1.2 Cell reselection shall not take place if there was a cell reselection within the previous 15 seconds.

1.3 Cell reselection for any other reason shall take place immediately, but the cell that the MS was camped on shall not be returned to within 5 seconds if another suitable cell can be found.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_60, TCV_bcchfl_DE := BcchFreqLst_55_R , TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, 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_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers1_and_2			
6		+MM_PwrOrSimOn(C_SIMIn)			
7		+NoReaction(50000)			
8		+PageOnCarriers(C_CellA, C_CellB, "dummy", "dummy", "dummy", "dummy")			
9		START T_dly(20000)			
10		+ltree_ChRq1_and_2			
11		+ltree_continue			
12		+ltree_continue1			
		ltree_continue			
13		(TCV_Null := OM_PgFill(C_CellA, PgReqTp1Norm), TCV_Null := OM_PgFill(C_CellB, PgReqTp1Norm))			
14		START T_dly(20000)			
15	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
16		−>Label1			
17		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
18		−>Label1			
19		?TIMEOUT T_dly			
20		(TCV_Null := OM_PgFill(C_CellB, PgReqTp1_01), TCV_Null := OM_ChangeRFOfCell(C_CellB, 61))			
21		START T_dly(4000)			
22	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
23		−>Label2			
24	CS_0023	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
25		?TIMEOUT T_dly			
26		(TCV_Null := OM_ChangeRFOfCell(C_CellB, 41))			
27		START T_dly(30000)			
28	Label3	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
29		−>Label3			
30	CS_0024	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
31		?TIMEOUT T_dly			
32		(TCV_Null := OM_ChangeRFOfCell(C_CellB, 61))			
33		START T_dly(100000)			
34	Label4	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
35		−>Label4			
36	CS_0025	?TIMEOUT T_dly		F	
37		L?DL_RacInChRq READTIMER T_dly(TCV_Time)	ChReq_01(C_RACH_B_1, ChRequest_02)		
38		CANCEL T_dly			
		ltree_continue1			
39		(TCV_Null := OM_PgFill(C_CellB, PgReqTp1Norm), TCV_Null := OM_ChangeRFOfCell(C_CellB, 41))			
40		START T_dly(20000)			
41	Label5	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
42		−>Label5			
43		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
44		−>Label5			
45		?TIMEOUT T_dly			
46		(TCV_Time := TCV_Time + 2000, TCV_Null := OM_ChangeRFOfCell(C_CellB, 61))			
47		START T_dly(TCV_Time)			
48	Label6	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
49		−>Label6			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
50		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
51		−>Label6			
52		?TIMEOUT T_dly			
53		(TCV_Null := OM_PgFill(C_CellA, PgReqTp1_01), TCV_Null := OM_ChangeRFOfCell(C_CellB, 41))			
54		START T_dly(11000)			
55	Label7	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
56		−>Label7			
57	CS_0026	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
58	CS_0027	?TIMEOUT T_dly		P	
		ltree_ChRq1_and_2			
59		(TCV_Cnt := 0)			
60	Label8	(TCV_Cnt := TCV_Cnt)			dummy line to allow label
61	CS_0028	?TIMEOUT T_dly		F	
62	CS_0029	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_01)	F	
63		L?DL_RacInChRq (TCV_Cnt := TCV_Cnt + 1)	ChReq_01(C_RACH_A_1, ChRequest_02)		
64		[TCV_Cnt = 1]			Rxed 2 RACHs on cell A?
65		−>Label8			
66		[TCV_Cnt <> 1]			
67		START T_dly1(4500)			
68		(TCV_Cnt := 0)			
69	Label9	(TCV_Cnt := TCV_Cnt)			dummy line to allow label
70	CS_0030	?TIMEOUT T_dly		F	
71		?TIMEOUT T_dly1			
72		−>Label9			
73		L?DL_RacInChRq READTIMER T_dly1(TCV_Time)	ChReq_01(C_RACH_A_1, ChRequest_02)		
74	CS_0031	[TCV_Time <> 0]		F	
75		[TCV_Time = 0]			
76		−>Label9			
77		L?DL_RacInChRq (TCV_Cnt := TCV_Cnt + 1)	ChReq_01(C_RACH_B_1, ChRequest_02)		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
78		[TCV_Cnt = 1]			Rxed 2 RACHs on cell B?
79		−>Label9			
80		[TCV_Cnt <> 1]			
81	Label10	L?DL_RacInChRq READTIMER T_dly1(TCV_Time)	ChReq_01(C_RACH_A_1, ChRequest_01)		
82	CS_0032	[TCV_Time <> 0]		F	
83		[TCV_Time = 0]			
84		−>Label10			
85		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_01)		
86		−>Label10			
87		?TIMEOUT T_dly1			
88		−>Label10			
89		?TIMEOUT T_dly			
90		CANCEL T_dly1			
		ltree_StartCarriers1_and_2			
91		+ltree_StartCarrier1			
92		+ltree_StartCarrier2			
		ltree_StartCarrier1			
93		+Config_BCCH_CCCH_A_1(51, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
94		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_60, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010101'B, '000000'B)			
		ltree_StartCarrier2			
95		+Config_BCCH_CCCH_B_1(41, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
96		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_60, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '011001'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_7R

Group : CellSelection/

Purpose : 1. To verify that the MS does not select a cell of low priority when a suitable cell of normal priority exists with a lower received signal strength.

2. To verify that the MS takes into account CELL_BAR_ACCESS and CELL_BAR_QUALIFY when performing cell selection and reselection.

3. To verify that if the following conditions are met:

- the cell belongs to the MS HPLMN
- the MS is in cell test operation mode
- the CELL_BAR_ACCESS is set to 1
- the CELL_BAR_QUALIFY is set to 0
- the Access Control class 15 is barred

then the "Cell Selection priority" and the "Status for cell reselection" shall be set to normal.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers(C_CellA, C_CellB, C_CellC, C_CellD, "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(33000), START T_dly1(50000)			
9	CS_0033	?TIMEOUT T_dly		F	
10	CS_0034	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
11	CS_0035	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
12	CS_0036	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
13		L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)		
14		CANCEL T_dly			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
15	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)		
16		−>Label1			
17	CS_0037	?TIMEOUT T_dly1		F	
18	CS_0038	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
19	CS_0039	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
20		L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
21	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
22		−>Label2			
23	CS_0040	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
24	CS_0041	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)	F	
25		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
26		+ltree_continue			
27		?TIMEOUT T_dly			
28		+ltree_continue			
		ltree_continue			
29		CANCEL T_dly, CANCEL T_dly1			
30		+MM_PwrOrSimOff(C_SIMIn)			
31		(TCV_Null := OM_StopAllBCCH(C_CellA, C_CellB, C_CellC, C_CellD, "dummy", "dummy", "dummy", "dummy"))			
32		(TCV_Null := OO_SetCellTestMode())			Switch MS to cell test mode
33		+ltree_StartCarriers			
34		+PageOnCarriers(C_CellA, C_CellB, C_CellC, C_CellD, "dummy", "dummy")			
35		+MM_PwrOrSimOn(C_SIMIn)			
36		START T_dly(33000), START T_dly1(50000)			
37	CS_0042	?TIMEOUT T_dly		F	
38	CS_0043	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
39	CS_0044	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
40	CS_0045	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)	F	
41		L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)		
42		CANCEL T_dly			
43	CS_0046	?TIMEOUT T_dly1		F	

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
44	CS_0047	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
45	CS_0048	L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)	F	
46	CS_0049	L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)	F	
47		L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
48	Label3	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
49		−>Label3			
50		L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)		
51		−>Label3			
52		L?DL_RacInChRq	ChReq_01(C_RACH_D_1, ChRequest_02)		
53		−>Label3			
54	CS_0050	?TIMEOUT T_dly1		P	
55	CS_0051	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	P	
		ltree_StartCarriers			
56		+ltree_StartCarrier1			
57		+ltree_StartCarrier2			
58		+ltree_StartCarrier3			
59		+ltree_StartCarrier4			
		ltree_StartCarrier1			
60		+Config_BCCH_CCCH_A_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
61		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’1’B, ’1’B, ’00’B, C_BABR_0, C_cch_1nonComb, C_BPM_3, ’000’B, ’10000’B, ’00000’B, ’000000’B, ’000000’B)			
		ltree_StartCarrier2			
62		+Config_BCCH_CCCH_B_1(43, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(9), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
63		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '1'B, '1'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '10000'B, '00000'B, '010100'B, '000000'B)			
		ltree_StartCarrier3			
64		+Config_BCCH_CCCH_C_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(18), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
65		+SysInfoSending_fh_pl_R(C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellC, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '1'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '10000'B, '00000'B, '001010'B, '000000'B)			
		ltree_StartCarrier4			
66		+Config_BCCH_CCCH_D_1(23, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
67		+SysInfoSending_fh_pl_R(C_SCH_D, C_BCCH_D_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellD, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '10000'B, '00000'B, '001010'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_8R

Group : CellSelection/

Purpose : 1. To verify that at least every 5 s the MS shall calculate the value of C1 and C2 for the serving cell and recalculate the C1 and C2 for non serving cells (if necessary). The MS shall then check whether the path loss criterion (C1) for current serving cell falls below zero for a period of 5 seconds. This indicates that the path loss has become too high.

2. To verify that whilst camped on a cell of the selected PLMN, the MS may need to select a different cell. The path loss criterion (C1) for the current serving cell falling below zero triggers cell reselection.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(50000)			
9	CS_0052	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
10	CS_0053	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
11		?TIMEOUT T_dly			
12		(TCV_Null := OM_ChangeRFOfCell(C_CellA, 33))			
13		+Pause(4000)			
14		(TCV_Null := OM_ChangeRFOfCell(C_CellA, 63))			
15		START T_dly(30000)			
16	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
17		->Label1			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
18	CS_0054	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
19		?TIMEOUT T_dly			
20		(TCV_Null := OM_ChangeRFOfCell(C_CellA, 33))			
21		START T_dly(20000)			
22	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
23		->Label2			
24	CS_0055	?TIMEOUT T_dly		F	
25	CS_0056	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	P	
		ltree_StartCarriers			
26		+ltree_StartCarrier1			
27		+ltree_StartCarrier2			
		ltree_StartCarrier1			
28		+Config_BCCH_CCCH_A_1(63, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
29		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '101000'B, '001111'B)			
		ltree_StartCarrier2			
30		+Config_BCCH_CCCH_B_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(969), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
31		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_9R

Group : CellSelection/

Purpose : 1. To verify that if the MS calculates a received level average (over 5 seconds) for a non−serving suitable cell which results in the value of C2 exceeding the value of C2 for the serving cell, then cell reselection takes place to the non−serving cell.

2. To verify that by using suitable varying levels of signal strength for non serving cells, the MS samples on non serving cell BCCH carriers are as far as possible distributed uniformly over each evaluated period.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_63, TCV_bcchfl_DE := BcchFreqLst_55_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_2, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(50000)			
9	CS_0057	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
10	CS_0058	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
11		?TIMEOUT T_dly			
12		(TCV_Null := OM_VaryRFOfCell(C_BCCH_B_1, 2700, 56, 2700, 33))			
13		START T_dly(25000)			
14	CS_0059	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
15	CS_0060	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
16		?TIMEOUT T_dly			
17		(TCV_Null := OM_ChangeRFOfCell(C_CellA, 37))			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
18		START T_dly(20000)			
19	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
20		−>Label1			
21	CS_0061	?TIMEOUT T_dly		F	
22		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
23	CS_0062	(TCV_Null := OM_StopVaryRFOfCell(C_BCCH_B_1))		P	
		ltree_StartCarriers			
24		+ltree_StartCarrier1			
25		+ltree_StartCarrier2			
		ltree_StartCarrier1			
26		+Config_BCCH_CCCH_A_1(53, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_2)			
27		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_63, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_2, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
		ltree_StartCarrier2			
28		+Config_BCCH_CCCH_B_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_2)			
29		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_63, BcchFreqLst_55_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_2, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_10R

Group : CellSelection/

Purpose : 1. To verify that by using suitable varying levels of signal strength for the serving cell, the MS performs a running average over 5 consecutive paging blocks.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(50000)			
9	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
10		−>Label1			
11		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
12		−>Label1			
13		?TIMEOUT T_dly			
14		(TCV_Null := OM_VaryRFOfCell(C_BCCH_A_1, 3000, 33, 3000, 63))			
15		START T_dly(25000)			
16	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
17		−>Label2			
18	CS_0063	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
19		?TIMEOUT T_dly			
20		(TCV_Null := OM_ChangeRFOfCell(C_CellB, 57))			
21		START T_dly(30000)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
22	Label3	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
23		−>Label3			
24	CS_0064	?TIMEOUT T_dly		F	
25		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
26	CS_0065	(TCV_Null := OM_StopVaryRFOfCell(C_BCCH_A_1))		P	
		ltree_StartCarriers			
27		+ltree_StartCarrier1			
28		+ltree_StartCarrier2			
		ltree_StartCarrier1			
29		+Config_BCCH_CCCH_A_1(63, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
30		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’0’B, ’1’B, ’00’B, C_BABR_0, C_cch_1nonComb, C_BPM_3, ’000’B, ’00000’B, ’00000’B, ’010100’B, ’000000’B)			
		ltree_StartCarrier2			
31		+Config_BCCH_CCCH_B_1(39, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
32		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’0’B, ’0’B, ’00’B, C_BABR_0, C_cch_1nonComb, C_BPM_3, ’000’B, ’00000’B, ’00000’B, ’010100’B, ’000000’B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_12R

Group : CellSelection/

Purpose : 1. To verify that the list of the strongest 6 non−serving carriers shall be updated at least as often as the duration of the running average defined for measurements on the BCCH allocation and may be updated more frequently.

2. To verify that when the MS recognizes that a new BCCH carrier has become one of the 6 strongest, and the BCCH data shall be decoded for the new carrier within 30 seconds.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(50000)			
9	CS_0066	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
10	CS_0067	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
11		?TIMEOUT T_dly			
12		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '001010'B, '000000'B)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
13		START T_dly(345000)			
14	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
15		−>Label1			
16	CS_0068	?TIMEOUT T_dly		F	
17	CS_0069	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	P	
		ltree_StartCarriers			
18		+ltree_StartCarrier1			
19		+ltree_StartCarrier2			
		ltree_StartCarrier1			
20		+Config_BCCH_CCCH_A_1(38, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
21		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’0’B, ’1’B, ’00’B, C_BABR_0, C_cch_1nonComb, C_BPM_3, ’000’B, ’00000’B, ’00000’B, ’010100’B, ’000000’B)			
		ltree_StartCarrier2			
22		+Config_BCCH_CCCH_B_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
23		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’0’B, ’0’B, ’00’B, C_BABR_0, C_cch_1nonComb, C_BPM_3, ’000’B, ’00000’B, ’00000’B, ’010100’B, ’000000’B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_13R

Group : CellSelection/

Purpose : 1. To verify that the MS will check the BSIC of the non−serving cell, which is in the list of 6 strongest neighbour cells, by changing the BSIC and the BCCH data of the non−serving cell such that the value of C2 for that cell exceeds the value of C2 of the serving cell, and observing that the MS performs cell reselection within the time allowed to check the BSIC, redetermine the BCCH data and perform cell reselection.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(50000)			
9	CS_0070	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)	F	
10	CS_0071	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
11		?TIMEOUT T_dly			
12		(TCV_Null := OM_SetTSCEqualToBCC(C_CellB, C_BCC_3))			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
13		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC_3, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '001010'B, '000000'B)			
14		START T_dly(85000)			
15	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
16		−>Label1			
17	CS_0072	?TIMEOUT T_dly		F	
18	CS_0073	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	P	
		ltree_StartCarriers			
19		+ltree_StartCarrier1			
20		+ltree_StartCarrier2			
		ltree_StartCarrier1			
21		+Config_BCCH_CCCH_A_1(38, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
22		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
		ltree_StartCarrier2			
23		+Config_SDCCH4_B_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
24		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_15R

Group : CellSelection/

Purpose : 1. To verify that if an LU is rejected with cause "LA not allowed" that the LAI of that cell is written into a forbidden list which prevents the MS from performing LU onto another cell in that LA. This is verified indirectly in test purpose 2, 3 and 4.

2. To verify that the MS will not reject a cell for camping on because the cell is part of a LA in the list of "forbidden LAIs for regional provision of service". This is verified indirectly by making the MS attempt an emergency call and checking that the channel request message is transmitted on the correct cell.

3. To verify that the MS when receiving an LU reject with cause value "LA not allowed", the MS continues to perform cell−reselection. Cell reselection is triggered if there is a better cell in the same LA, or a much better cell in another LA of the selected PLMN (using the CRH parameter).

4. To verify that a new LU attempt will be performed when a new LA (or new PLMN) is entered.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys14, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+SDCCH8_A_1_nociph_R(TSPX_SDCCH8SubDef, C_Immass, TCV_slot, TCV_tsc, ChMod_sign, FreqBCCH(TCV_chdescr_arfcn), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
7		+MM_PwrOrSimOn(C_SIMIn)			
8		START T_dly(33000)			
9	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
10		−>Label1			
11		L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)		
12		−>Label1			
13	CS_0074	?TIMEOUT T_dly		F	

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
14		L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
15		+MM_LupRej_R(C_rc_LAnotallowed, C_imsi_attach, TimingAdv(0))			
16		START T_dly(30000)			
17	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
18		->Label2			
19		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
20		->Label2			
21		L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)		
22		->Label2			
23		?TIMEOUT T_dly			
24		(TCV_chdescr_arfcn := 62)			
25		(TCV_Null := OM_FreeResource(TCV_ch))			
26		+SDCCH8_C_1_nociph_R(TSPX_SDCCH8SubDef, C_Immass, TCV_slot, TCV_tsc, ChMod_sign, FreqBCCH(TCV_chdescr_arfcn), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
27		+BasicServiceMO(C_EmgCall, TSPX_EmgCallRate)			
28		(TCV_Null := OO_InitCall(C_EmgCall))			
29		START T_dly(15000)			
30	Label3	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
31		->Label3			
32		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
33		->Label3			
34	CS_0075	?TIMEOUT T_dly		F	
35		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq.msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq_01(C_RACH_C_1, ChRequest_02)		
36		+ltree_continue			
		ltree_continue			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
37		L!DL_UdatRqImmass	ImmAss(C_AGCH_C_1, ImmAsgn_01Def_R(TCV_Rr, TCV_Fn, TSPX_SDCCH8SubDef, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, TimingAdv(0)))		
38		L?DL_EstInCmsRq	CMSerReq(CMServiceReq_01)		
39		L!DL_DatRqCmsRej	CMSerRej(TCV_ch, CMServiceRej_30(C_rc_networkfailure))		
40		+ChanRel(TCV_ch)			
41		START T_dly(10000)			
42	Label4	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
43		−>Label4			
44		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
45		−>Label4			
46		L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)		
47		−>Label4			
48		?TIMEOUT T_dly			
49		(TCV_chdescr_arfcn := 70, TCV_Null := OM_FreeResource(TCV_ch))			
50		+SDCCH8_B_1_nociph_R(TSPX_SDCCH8SubDef, C_Immass, TCV_slot, TCV_tsc, ChMod_sign, FreqBCCH(TCV_chdescr_arfcn), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
51		(TCV_Null := OM_ChangeRFOfCell(C_CellB, 65))			
52		START T_dly(30000)			
53	Label5	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
54		−>Label5			
55		L?DL_RacInChRq	ChReq_01(C_RACH_C_1, ChRequest_02)		
56		−>Label5			
57	CS_0076	?TIMEOUT T_dly		F	
58		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq.msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq_01(C_RACH_B_1, ChRequest_02)		
59		L!DL_UdatRqImmass	ImmAss(C_AGCH_B_1, ImmAsgn_01Def_R(TCV_Rr, TCV_Fn, TSPX_SDCCH8SubDef, 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**
60	CS_0077	L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_05)	P	
		ltree_StartCarriers			
61		+ltree_StartCarrier1			
62		+ltree_StartCarrier2			
63		+ltree_StartCarrier3			
64		(TCV_chdescr_arfcn := 41)			
		ltree_StartCarrier1			
65		+Config_BCCH_CCCH_A_1(63, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
66		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_1, C_CellReselectHys14, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '110010'B, '000000'B)			
		ltree_StartCarrier2			
67		+Config_BCCH_CCCH_B_1(54, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(9), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
68		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_1, C_CellReselectHys0, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_2222, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '100000'B, '000000'B)			
		ltree_StartCarrier3			
69		+Config_BCCH_CCCH_C_1(44, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
70		+SysInfoSending_fh_pl_R(C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_1, C_CellReselectHys10, C_MaxPwrLvlG, C_ci_cellC, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '011010'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_16R

Group : CellSelection/

Purpose : 1. To verify that the MS initializes the DSC counter in accordance with the conformance requirement. This is verified indirectly.

2. To verify that whenever the MS successfully decodes a message on paging subchannel , the DSC is increased by 1, (however never beyond the nearest integer to 90/N). This is verified indirectly.

3. To verify that whenever the MS can not successfully decode a message on paging subchannel, the DSC is decreased by 4. This is verified indirectly.

4. To verify that when the DSC reaches 0, a downlink signalling failure shall be declared and the MS will perform cell reselection.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarriers			
6		+MM_PwrOrSimOn(C_SIMIn)			
7		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
8		START T_dly(40000)			
9	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
10		−>Label1			
11	CS_0078	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
12		?TIMEOUT T_dly			
13		+CCCH_group_Paging_group(TCV_Ccd0A, TSPX_IMSI)			
14		+SelectPagingCh(C_CellA)			
15		(TCV_Null := OM_CorruptPgBlks(TCV_PgCh , TCV_Pgg, 4))			
16		START T_dly(30000)			
17	Label2	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
18		−>Label2			
19	CS_0079	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	F	
20		?TIMEOUT T_dly			
21		+CCCH_group_Paging_group(TCV_Ccd0A, TSPX_IMSI)			
22		+SelectPagingCh(C_Cell A)			
23		(TCV_Null := OM_CorruptPgBlks(TCV_PgCh, TCV_Pgg, 5))			
24		+ltree_continue			
		ltree_continue			
25		START T_dly(15000)			
26	Label3	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
27		−>Label3			
28	CS_0080	?TIMEOUT T_dly		F	
29		L?DL_RacInChRq READTIMER T_dly(TCV_Time), CANCEL T_dly	ChReq_01(C_RACH_B_1, ChRequest_02)		
30		START T_dly1(30000 − TCV_Time)			
31	Label4	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
32		−>Label4			
33		L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)		
34		−>Label4			
35	CS_0081	?TIMEOUT T_dly1		P	
		ltree_StartCarriers			
36		+ltree_StartCarrier1			
37		+ltree_StartCarrier2			
		ltree_StartCarrier1			
38		+Config_BCCH_CCCH_A_1(43, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
39		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, ’0’B, ’0’B, ’1’B, ’00’B, C_BABR_0, C_cch_1nonComb, C_BPM_3, ’000’B, ’00000’B, ’00000’B, ’010100’B, ’000000’B)			
40		(TCV_Ccd0A := TCV_Ccd0H)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
		ltree_StartCarrier2			
41		+Config_BCCH_CCCH_B_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(956), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
42		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
43		(TCV_Ccd0B := TCV_Ccd0H)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_20_21_17R

Group : CellSelection/

Purpose : If no suitable cell is found within cell reselection during a 10 second period, then the cell selection algorithm of GSM 03.22 shall be performed, GSM 05.08; 6.6.2.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(600)			
2		(TCV_n := C_Test_rgsm, TCV_bcchfl_P := BcchFreqLst_50c, TCV_bcchfl_DE := BcchFreqLst_52c_R, TCV_arfcn_PE := 3)			
3		+IdleUpdated_R_CS(C_E_default, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, C_BCC, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, TimingAdv(0), C_BABR_0, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, C_CellReselectHys4, C_MaxPwrLvlG, TCV_bcchfl_P, BcchFreqLst_Omit, TCV_bcchfl_DE, C_Restablishment, C_BCC, C_NCC, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, TCV_arfcn_PE, TSPX_IMSI, TCV_n, C_NCCP_2)			
4	body	+MM_PwrOrSimOff(C_SIMIn)			
5		+ltree_StartCarrier1			
6		+MM_PwrOrSimOn(C_SIMIn)			
7		(TCV_Null := OO_PressKeyWhenInService())			
8		(TCV_Null := OM_ChangeRFOfCell(C_CellA, 13))			
9		+ltree_StartCarrier2			
10		+PageOnCarriers("dummy", C_CellB, "dummy", "dummy", "dummy", "dummy")			
11		START T_dly(60000)			
12	Label1	L?DL_RacInChRq	ChReq_01(C_RACH_A_1, ChRequest_02)		
13		−>Label1			
14	CS_0082	?TIMEOUT T_dly		F	
15	CS_0083	L?DL_RacInChRq	ChReq_01(C_RACH_B_1, ChRequest_02)	P	
		ltree_StartCarrier1			
16		+Config_BCCH_CCCH_A_1(43, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(3), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
17		+SysInfoSending_fh_pl_R(C_SCH_A, C_BCCH_A_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellA, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '1'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B) ltree_StartCarrier2			
18		+Config_BCCH_CCCH_B_1(33, C_MaxPwrLvlG, ChMod_sign, C_Immass, TCV_slot, TCV_tsc, FreqBCCH(965), C_FCCH_SCH_BCCH_CCCH, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			uses AFRCN 965 (not in BcchFreqLst_52c_R − NCD)
19		+SysInfoSending_fh_pl_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_2, C_NECI_0, C_ATT_0, C_CellReselectHys4, C_MaxPwrLvlG, C_ci_cellB, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1111, CellOpt_01, CellChDes_24, BcchFreqLst_50c, BcchFreqLst_52c_R, C_BCC, C_NCC, C_NCCP_2, C_Restablishment, '0'B, '0'B, '0'B, '00'B, C_BABR_0, C_cch_1nonComb, C_BPM_3, '000'B, '00000'B, '00000'B, '010100'B, '000000'B)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_2_1R

Group : Signalling/

Purpose : To verify that the MS reports appropriate results when the test system gives information about neighbouring cells

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(480)			
2		+ BasicServiceMT (TSPX_MTBscSvcA, TSPX_MTChRateA)			
3		+ IdleUpdated_R (C_E_neighbourdefault, 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, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_21, BcchFreqLst_Omit, BcchFreqLst_27, C_noRestablishment, C_BCC_3, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubA, TSPX_CKSNDef, TSPX_RANDDef, C_arfcn_2, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
4		+StartMultiCells_02_R (BcchFreqLst_21, C_Immass, TCV_slot, TCV_tsc, TimingAdv(30), C_ATT_0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_1, C_LAC_1, C_LAC_1, C_LAC_1, C_LAC_1, C_LAC_1)			
5		(TCV_Cnt:=0)			
6		REPEAT ltree_loopForC UNTIL [TCV_Cnt =6]			
		ltree_loopForC			
7		+ltree_StartMultiCells			
8		+CCConfigTCH_R (TSPX_TCHHSubDef, C_Ass, TSPX_TmSltDef, TSPX_TscDef, TimingAdv(30), C_BABR_0, C_cch_1Comb, C_BPM_3, TCV_ChMod.mode, FreqTCH_omit, C_arfcn_tchA)			
9	body	+PreEnterCCstateU10_r01_R (TimingAdv(30), TSPX_SDCCH4SubA, 7, TSPX_CKSNDef, TSPX_RANDDef, TSPX_TmSltDef, TSPX_TscDef, TSPX_TCHHSubDef, TCV_ChRate)			
10		(TCV_Null := OM_StartMsrReport(TCV_sacchTch))			
11		+ltree_receiveMsrRept			
12		+local_MsrReptCHK(1, 2)			
13		START T_dly1(960)			960 ms
14		+ltree_receiveMsrRept2			
15		+local_MsrReptCHK(1, 2)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
16		(TCV_Null := OM_StopMsrReport(TCV_sacchTch))			
17		+PostMainLinkRel(TCV_chTch)			
18		+execution2			
		execution2			
19		+ltree_sysinfo5and5bis			1.
20		+PreEnterCCstateU10_r01_R (TimingAdv(30), TSPX_SDCCH4SubA, 7, TSPX_CKSNDef, TSPX_RANDDef, TSPX_TmSltDef, TSPX_TscDef, TSPX_TCHHSubDef, TCV_ChRate)			
21		(TCV_Null := OM_StartMsrReport(TCV_sacchTch))			
22		+ltree_receiveMsrRept			
23		+local_MsrReptCHK(1, 3)			
24		START T_dly1(960)			960 ms
25		+ltree_receiveMsrRept2			
26		+local_MsrReptCHK(1, 3)			
27		(TCV_Null := OM_StopMsrReport(TCV_sacchTch))			
28		+PostMainLinkRel(TCV_chTch)			
29		(TCV_Cnt:=(TCV_Cnt +1))			
		ltree_sysinfo5and5bis			
30		+Wait(C_T_Wait1)			
31		[TCV_Cnt = 1]			
32		L!DL_UdatRqSysinfo5	SysInfo5(TCV_sacch, SysInf5(BcchFreqLst_r201))		
33		L!DL_UdatRqSysinfo5bis	SysInfo5bis(TCV_sacch,SysInf5bis(BcchFreqLst_r207))		
34		[TCV_Cnt = 2]			
35		L!DL_UdatRqSysinfo5	SysInfo5(TCV_sacch, SysInf5(BcchFreqLst_r202))		
36		L!DL_UdatRqSysinfo5bis	SysInfo5bis(TCV_sacch, SysInf5bis(BcchFreqLst_r208))		
37		[TCV_Cnt = 3]			
38		L!DL_UdatRqSysinfo5	SysInfo5(TCV_sacch, SysInf5(BcchFreqLst_r207))		
39		L!DL_UdatRqSysinfo5bis	SysInfo5bis(TCV_sacch, SysInf5bis(BcchFreqLst_r209))		
40		[TCV_Cnt = 4]			
41		L!DL_UdatRqSysinfo5	SysInfo5(TCV_sacch, SysInf5(BcchFreqLst_r204))		
42		L!DL_UdatRqSysinfo5bis	SysInfo5bis(TCV_sacch, SysInf5bis(BcchFreqLst_r210))		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
43		[TCV_Cnt = 5]			
44		L!DL_UdatRqSysinfo5	SysInfo5(TCV_sacch, SysInf5(BcchFreqLst_r205))		
45		L!DL_UdatRqSysinfo5bis	SysInfo5bis(TCV_sacch, SysInf5bis(BcchFreqLst_r211))		
46		[TCV_Cnt = 6]			
47		L!DL_UdatRqSysinfo5	SysInfo5(TCV_sacch, SysInf5(BcchFreqLst_r206))		
48		L!DL_UdatRqSysinfo5bis	SysInfo5bis(TCV_sacch, SysInf5bis(BcchFreqLst_r212))		
		local_MsrReptCHK(in1, in2 : INTEGER)			
49		(TCV_Res := OC_MsrReptChk(TCV_MsrRes, in1))			
50	L_0555	[NOT TCV_Res]		(F)	
51	L_0556	[TCV_Res]		(P)	
		ltree_receiveMsrRept			
52		[TCV_Cnt = 1]			
53		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr)	MsrRept(MsrReport_03r(MsrResult_03r1))		
54		[TCV_Cnt = 2]			
55		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr)	MsrRept(MsrReport_03r(MsrResult_03r2))		
56		[TCV_Cnt = 3]			
57		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr)	MsrRept(MsrReport_03r(MsrResult_03r3))		
58		[TCV_Cnt = 4]			
59		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr)	MsrRept(MsrReport_03r(MsrResult_03r4))		
60		[TCV_Cnt = 5]			
61		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr)	MsrRept(MsrReport_03r(MsrResult_03r5))		
62		[TCV_Cnt = 6]			
63		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr)	MsrRept (MsrReport_04r)		
		ltree_receiveMsrRept2			
64		[TCV_Cnt = 1]			
65	L_0557	?TIMEOUT T_dly1		(F)	1.
66		+PostMainLinkRel(TCV_chTch)			
67		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr) CANCEL T_dly1	MsrRept(MsrReport_03r(MsrResult_03r1))		
68		[TCV_Cnt = 2]			
69	L_0558	?TIMEOUT T_dly1		(F)	1.
70		+PostMainLinkRel(TCV_chTch)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
71		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr) CANCEL T_dly1	MsrRept(MsrReport_03r(MsrResult_03r2))		
72		[TCV_Cnt = 3]			
73	L_0559	?TIMEOUT T_dly1		(F)	1.
74		+PostMainLinkRel(TCV_chTch)			
75		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr) CANCEL T_dly1	MsrRept(MsrReport_03r(MsrResult_03r3))		
76		[TCV_Cnt = 4]			
77	L_0560	?TIMEOUT T_dly1		(F)	1.
78		+PostMainLinkRel(TCV_chTch)			
79		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr) CANCEL T_dly1	MsrRept(MsrReport_03r(MsrResult_03r4))		
80		[TCV_Cnt = 5]			
81	L_0561	?TIMEOUT T_dly1		(F)	1.
82		+PostMainLinkRel(TCV_chTch)			
83		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr) CANCEL T_dly1	MsrRept(MsrReport_03r(MsrResult_03r5))		
84		[TCV_Cnt = 6]			
85	L_0562	?TIMEOUT T_dly1		(F)	1.
86		+PostMainLinkRel(TCV_chTch)			
87		L?DL_UdatInMsrRpt (TCV_MsrRes := DL_UdatInMsrRpt.msg.msrr) CANCEL T_dly1	MsrRept(MsrReport_04r)		
		ltree_StartMultiCells			
88		[TCV_Cnt = 0]			
89		[TCV_Cnt = 1]			
90		+StartMultiCells_02r_R (BcchFreqLst_r201, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), 0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_2, C_LAC_3, C_LAC_4, C_LAC_5, C_LAC_6, C_LAC_7, C_LAC_8)			
91		[TCV_Cnt = 2]			
92		+StartMultiCells_02r_R (BcchFreqLst_r202, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), 0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_2, C_LAC_3, C_LAC_4, C_LAC_5, C_LAC_6, C_LAC_7, C_LAC_8)			
93		[TCV_Cnt = 3]			
94		+StartMultiCells_02r_R (BcchFreqLst_r203, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), 0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_2, C_LAC_3, C_LAC_4, C_LAC_5, C_LAC_6, C_LAC_7, C_LAC_8)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
95		[TCV_Cnt = 4]			
96		+StartMultiCells_02r_R (BcchFreqLst_r204, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), 0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_2, C_LAC_3, C_LAC_4, C_LAC_5, C_LAC_6, C_LAC_7, C_LAC_8)			
97		[TCV_Cnt = 5]			
98		+StartMultiCells_02r_R (BcchFreqLst_r205, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), 0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_2, C_LAC_3, C_LAC_4, C_LAC_5, C_LAC_6, C_LAC_7, C_LAC_8)			
99		[TCV_Cnt = 6]			
100		+StartMultiCells_02r_R (BcchFreqLst_r206, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), 0, C_BABR_0, C_cch_1Comb, C_BPM_3, C_T3212_0, C_MCC_1, C_PLMN_1, C_LAC_1, C_LAC_2, C_LAC_3, C_LAC_4, C_LAC_5, C_LAC_6, C_LAC_7, C_LAC_8)			
Detailed Comments : 1. The interval between 2 successive layer 2 frames containing MEASUREMENT REPORT exceeds one layer 2 frame, fail.					

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_2_2R
Group : Signalling/
Purpose : To verify that the MS can correctly set up a dedicated control channel when R−GSM frequencies are used.
Configuration :
Default : OtherEventsFail_01
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	pre	START T_guard(300)			
2		+IdleUpdated_R (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_1nonComb, C_BPM_3, C_T3212_1, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_r19, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcn_972, TSPX_IMSI, C_Test_rgsm, C_NCCP_2)			
3		(TCV_ia_ts:= TSPX_TmSltNotZero, TCV_Cnt:= 0)			
4	body	REPEAT localtree_body UNTIL [TCV_Cnt =2]			
		localtree_body			
5		+local_config			
6		+Wait(C_T_Wait1)			
7		L!DL_UdatRqPg1Rq	PgReq1(TCV_PgCh, TCV_Pgg, PgReqTp1_01)		
8		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_17)		
9		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
10		+ltree_send_immass			
11		L?DL_EstInPgRes	PgRes(TCV_ch, PagingRes_01)	(P)	
12		ACTIVATE(OtherEventsFail)			Restore Normal default
13		+ChanRel(TCV_ch)			
14		(TCV_Null := OM_FreeResource(TCV_ch), TCV_Cnt := TCV_Cnt + 1)			
		local_config			
15		[TCV_Cnt = 0]			
16		+SDCCH8_A_1_nociph_R (TSPX_SDCCH8SubA, C_Ass, TCV_ia_ts, TSPX_TscDef, ChMod_sign, FreqSDCCH8_r1, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			for 1)

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
17		+SysInfo_5bisSending(TCV_sacch8, TCV_sysinf5bis)			
18		[TCV_Cnt = 1]			
19		+SDCCH8_A_2_nociph(TSPX_SDCCH8SubB, C_Ass, TCV_ia_ts, TSPX_TscDef, ChMod_r01, FreqSDCCH8_r2, FreqSDCCH8_r2, TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			for 2)
20		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
21		(TCV_ch := TCV_chTch)			
		ltree_send_immass			
22		[TCV_Cnt = 0]			
23		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_E_01 (TCV_Rr, TCV_Fn, TCV_ia_ts, TSPX_TscDef, TCV_chdescr_arfcn, TimingAdv(0)))		1)
24		[TCV_Cnt = 1]			
25		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_R_02 (TCV_Rr, TCV_Fn, TCV_ia_ts, TSPX_TscDef, TimingAdv(0)))		2)
Detailed Comments : 1) Immediate Assignment with single RF on ARFCN=972 2) Immediate Assignment with frequency hopping					

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_2_3R

Group : Signalling/

Purpose : 1. To verify that upon receipt of an ASSIGNMENT COMMAND, the MS switches to the channel defined in the ASSIGNMENT COMMAND, establishes the link and sends an ASSIGNMENT COMPLETE message.

2. To verify that an MS, having received an ASSIGNMENT COMMAND, is able in case of frequency hopping to decode the mobile allocation and frequency list correctly and applies the specified frequencies.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	pre	START T_guard (300)			
2		+IdleUpdated_R (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_22, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TSPX_IMSI, C_Test_rgsm, C_NCCP_2)			
3		(TCV_ia_ts := TSPX_TmSltA, TCV_Cnt := 1, TCV_chdescr_arfcn := C_arfcnC)			
4	body	REPEAT localtree_body UNTIL [TCV_Cnt =3]			
		localtree_body			
5		+SDCCH8_A_2_nociph (TSPX_SDCCH8SubDef, C_Ass, TCV_ia_ts, TSPX_TscDef, ChMod_sign, FreqBCCH(TCV_chdescr_arfcn), FreqBCCH(TCV_chdescr_arfcn), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
6		+SysInfo_5bisSending (TCV_sacchTch, TCV_sysinf5bis)			
7		(TCV_ch := TCV_chTch)			
8		+Wait (C_T_Wait1)			
9		L!DL_UdatRqPg1Rq	PgReq1 (TCV_PgCh, TCV_Pgg, PgReqTp1_01)		
10		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_17)		
11		ACTIVATE (OtherEventsFail_02)			To match ChReq retrans.
12		L!DL_UdatRqImmass	ImmAss (TCV_agch, ImmAsgn_27(TCV_Rr, TCV_Fn, TCV_ia_ts, 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**
13		L?DL_EstInPgRes	PgRes(TCV_ch, PagingRes_01)		
14		ACTIVATE(OtherEventsFail)			Restore Normal default
15		+ltree_chtype			
16		+ltree_send_ch_assign			
17		+ChanRel(TCV_chTch)			
18		(TCV_Null := OM_FreeResource(TCV_chTch), TCV_Cnt := (TCV_Cnt + 1))			Loop for K
19		+Wait(C_T_Wait1)			
		ltree_chtype			
20		[TSPC_SvcOnTCH]			
21		(TCV_chtype := '00001'B)			for TCH
22		[NOT TSPC_SvcOnTCH]			
23		(TCV_chtype := INT_TO_BIT((8 + BIT_TO_INT(TSPX_SDCCH8SubA)), 5))			for SDCCH 8
		ltree_send_ch_assign			
24		[TCV_Cnt = 1]			
25		(TCV_asscmd_ts := INT_TO_BIT(((BIT_TO_INT(TSPX_TmSltA) + 1) MOD 8), 3), TCV_Cnt1 := 1)			
26		REPEAT ltree_assign1 UNTIL [TCV_Cnt1 = 7]			
27		[TCV_Cnt = 2]			
28		(TCV_Cnt1 := 1)			
29		REPEAT ltree_assign2 UNTIL [TCV_Cnt1 = 7]			
		ltree_assign1			
30		+ltree_assign1_setup			
31		+ltree_chnassign1			
32		(TCV_AssCmd := AsgnCmd_22_Ar1(TCV_asscmd_ts, TCV_chtype, TCV_flist, TCV_flistl, TCV_n))			
33		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			1)
34		START T_dly(500)			6)
35		?TIMEOUT T_dly			
36		(TCV_Cnt1 := (TCV_Cnt1 + 1), TCV_Null := OM_FreeResource(TCV_ch), TCV_ch := TCV_chTch)			Loop for C
		ltree_assign2			
37		+ltree_assign2_setup			
38		+ltree_asscmdsending			
39		START T_dly(500)			6)
40		?TIMEOUT T_dly			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
41		(TCV_Cnt1 := (TCV_Cnt1 + 1), TCV_Null := OM_FreeResource(TCV_ch), TCV_ch := TCV_chTch)			
		ltree_asscmdsending			
42		[TCV_Cnt1 = 4]			
43		+ltree_chnassign1			3)
44		(TCV_AssCmd := AsgnCmd_22_Ae2(TCV_asscmd_ts, TCV_chtype, TCV_cchdescr, TCV_mar1, TCV_n))			
45		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			2)
46		[TCV_Cnt1 <> 4]			
47		+ltree_chnassign2			4)
48		(TCV_AssCmd := AsgnCmd_22_Ar3 (TCV_asscmd_ts, TCV_chtype, TCV_cchdescr, TCV_mar1, TCV_mar2, TCV_n))			
49		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			2)
		ltree_chnassign1			
50		[TSPC_SvcOnTCH]			1)
51		[(TCV_Cnt1 MOD 2) > 0]			
52		+FullRateCh_A_1_nociph_R (C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCH_rf1 (TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
53		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
54		[(TCV_Cnt1 MOD 2) = 0]			
55		+FullRateCh_A_2_nociph_R (C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCH_rf1(TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
56		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
57		(TCV_chTch := TCV_chTch1)			
58		[(NOT TSPC_SvcOnTCH)]			2)
59		[(TCV_Cnt1 MOD 2) > 0]			
60		(TCV_chTch := TCV_ch)			
61		+SDCCH8_A_1_nociph_R(TSPX_SDCCH8SubA, C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_sign, FreqTCH_rf1(TCV_mar1,TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
62		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
63		(TCV_chTch1 := TCV_chTch, TCV_chTch := TCV_ch, TCV_ch := TCV_chTch1)			5)
64		[(TCV_Cnt1 MOD 2) = 0]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
65		+SDCCH8_A_2_nociph(TSPX_SDCCH8SubA, C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_sign, FreqTCH_rf1(TCV_mar1,TCV_flist, TCV_flistl, TCV_n), FreqTCH_rf1(TCV_mar1,TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
66		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
		ltree_chnassign2			
67		[TSPC_SvcOnTCH]			1)
68		[(TCV_Cnt1 MOD 2) > 0]			
69		+FullRateCh_A_1_nociph_R (C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCH_rf2 (TCV_mar2, TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
70		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
71		[(TCV_Cnt1 MOD 2) = 0]			
72		+FullRateCh_A_2_nociph_R (C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCH_rf2(TCV_mar2, TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
73		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
74		(TCV_chTch := TCV_chTch1)			
75		[(NOT TSPC_SvcOnTCH)]			2)
76		[(TCV_Cnt1 MOD 2) > 0]			
77		(TCV_chTch := TCV_ch)			
78		+SDCCH8_A_1_nociph_R (TSPX_SDCCH8SubA, C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_sign, FreqTCH_rf2 (TCV_mar2, TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
79		+SysInfo_5bisSending (TCV_sacchTch, TCV_sysinf5bis)			
80		(TCV_chTch1 := TCV_chTch, TCV_chTch := TCV_ch, TCV_ch := TCV_chTch1)			
81		[(TCV_Cnt1 MOD 2) = 0]			
82		+SDCCH8_A_2_nociph (TSPX_SDCCH8SubA, C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_sign, FreqTCH_rf2 (TCV_mar2, TCV_mar1, TCV_flist, TCV_flistl, TCV_n), FreqTCH_rf2 (TCV_mar2, TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
83		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
		ltree_assign1_setup			
84		[TCV_Cnt1 = 1]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
85		(TCV_flist := C_flist_r_401, TCV_flistl:='05'O, TCV_mar1 :='00000111'B, TCV_n := 3)			
86		[TCV_Cnt1 = 2]			
87		(TCV_flist := C_flist_r_402, TCV_flistl:='06'O, TCV_mar1:='00011111'B, TCV_n := 5)			
88		[TCV_Cnt1 = 3]			
89		(TCV_flist := C_flist_r_403, TCV_flistl:='06'O, TCV_mar1:= '00001111'B, TCV_n := 4)			
90		[TCV_Cnt1 = 4]			
91		(TCV_flist := C_flist_r_404, TCV_flistl:='06'O, TCV_mar1:='00111111'B, TCV_n := 6)			
92		[TCV_Cnt1 = 5]			
93		(TCV_flist := C_flist_r_405, TCV_flistl:='07'O, TCV_mar1:='00111111'B, TCV_n := 6)			
94		[TCV_Cnt1 = 6]			
95		(TCV_flist := C_flist_r_406, TCV_flistl:='10'O, TCV_mar1:='00000111'B, TCV_n := 3)			
		ltree_assign2_setup			
96		[TCV_Cnt1 = 1]			
97		(TCV_cchdescr := C_cchd_r_407, TCV_mar1:='00010010'B, TCV_mar2:='00010000'B, TCV_flist:= C_flist_r_407, TCV_flistl:='10'O, TCV_n := 3)			
98		[TCV_Cnt1 = 2]			
99		(TCV_cchdescr := C_cchd_r_408, TCV_mar1:='00000110'B, TCV_mar2:='01001100'B, TCV_flist:= C_flist_r_408, TCV_flistl:='0D'O, TCV_n := 5)			
100		[TCV_Cnt1 = 3]			
101		(TCV_cchdescr := C_cchd_r_409, TCV_mar1:='00011000'B, TCV_mar2:='00010001'B, TCV_flist:= C_flist_r_409, TCV_flistl:='10'O, TCV_n := 4)			
102		[TCV_Cnt1 = 4]			
103		(TCV_cchdescr := C_cchd_r_415, TCV_mar1:='00000011'B, TCV_mar2:='11000111'B, TCV_flist:= C_flist_r_415, TCV_flistl:='0A'O, TCV_n := 7)			
104		[TCV_Cnt1 = 5]			
105		(TCV_cchdescr := C_cchd_r_411, TCV_mar1:='00010001'B, TCV_mar2:='00100100'B, TCV_flist:= C_flist_r_411, TCV_flistl:='04'O, TCV_n := 4)			
106		[TCV_Cnt1 = 6]			
107		(TCV_cchdescr := C_cchd_r_412, TCV_mar1:='00000000'B, TCV_mar2:='00000111'B, TCV_flist:= C_flist_r_412, TCV_flistl:='10'O, TCV_n := 3)			mae

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments	**:** 1) Assignment Command without Cell Channel Description IE 2) Assignment Command with Cell Channel Description IE 3) Length of mobile alocation contents is 1 octet 4) PDU contains Mobile Allocation with 2 octets 5) To ensure the correct channel id is in the correct test case variable 6) Ensure all communication is complete before freeing the resource

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_2_4_1R

Group : Signalling/

Purpose : To check that the MS correctly performs a non−synchronized handover, from a non hopping primary band SDCCH to a hopping TCH or SDCCH using R−GSM frequencies, whatever the coding used for the hopping sequence description and that it activates the new channel correctly.
This is tested in the following cases:
R−GSM signalling/ Handover / successful / call under establishment / non−synchronized /
− from SDCCH/8 to TCH/F if the MS supports a TCH
− from SDCCH/8 to SDCCH/8 if not

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(1500)			
2		+IdleUpdated_R (C_E_default, C_CellA, C_SCH_A, C_BCCH_A_1, C_Immass, C_S0, C_SlotNU, C_SlotNU, C_SlotNU, TSPX_TscDef, C_TxInt_5, C_Max_1, C_NECI_0, C_ATT_0, TimingAdv(30), C_BABR_0, C_cch_1nonComb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_23A, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubA, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TSPX_IMSI, C_Test_rgsm, C_NCCP_2)			
3		(TCV_Horf:= TSPX_HoRefA, TCV_Pwrlvl_ho:= INT_TO_BIT(TSPX_PwrlvlA, 5), TCV_flist := C_flist_r_416, TCV_flistl:= ’10’O, TCV_mar1 := ’00000101’B, TCV_mar2 := ’00000001’B, TCV_ia_ts:= TSPX_TmSltNotZero, TCV_sysinfo5bis_A := TCV_sysinf5bis, TCV_n := 3, TCV_ts:= TSPX_TmSltNotZero)			1.
4		+local_continue			
		local_continue			
5		+StartCellB_rgsm (C_E_neighbourdefault, C_arfcn_965, C_Immass, C_S0, TCV_tsc, TimingAdv(20), C_TxInt_5, C_Max_1, 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_1, CellChDes_23B, BcchFreqLst_52, BcchFreqLst_r47, C_noRestablishment, C_BCC_3, C_NCC)			
6		+SDCCH8_A_1_nociph_R (TSPX_SDCCH8SubDef, C_Immass, TCV_ia_ts, TCV_tsc, ChMod_sign, FreqTCH_rf2(TCV_mar1, TCV_mar2, TCV_flist,TCV_flistl, TCV_n), TimingAdv(30), C_BABR_0, C_cch_1nonComb, C_BPM_3)			TCV_ch
7		+SysInfo_5bisSending(TCV_sacch8, TCV_sysinfo5bis_A)			
8	body	(TCV_counter_k :=1)			
9		+ltree_chtype			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
10		REPEAT local_body UNTIL [TCV_counter_k =4]			Loop of k = 1 to 3
		local_body			
11		(TCV_counter_c :=1)			
12		REPEAT ltree_execloopC UNTIL [TCV_counter_c=7]			Loop of c = 1 to 6
13		(TCV_counter_k := (TCV_counter_k +1))			
		ltree_execloopC			
14		[(TCV_counter_k <>1) OR (TCV_counter_c<>6)]			
15		+ltree_chconfig			
16		+Est_MO_Call_init_R (C_CHSDCCH8_NonFH, MoblAllc_281e2, TimingAdv(30), TSPX_TscDef, TCV_n, C_arfcn_34)			
17		(TCV_ch1 := TCV_ch, TCV_ch := TCV_chTch, TCV_Null:= OM_SendNextOnHO(TCV_ch, TCV_Cntref))			
18		+ltree_hocmd			
19		+RR_hocomp1(TCV_hotime, TimingAdv(20), TCV_ch1)			
20		L?DL_DatInSetup (TCV_Mt1 := DL_DatInSetup.msg.mt, TCV_TI := DL_DatInSetup.msg.ti)	SetupRcv(SetupInd_01)		
21		+SendSeqNo_chk			
22		L!DL_DatRqRelCmp	RelComSnd(TCV_ch, ReleaseCmp_04(TCV_TI))		
23		+ChanRel(TCV_ch)			
24		+Switchcell_R (C_CellB, C_CellA, C_LAC_1, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TCV_Ccd0A, TSPX_IMSI)			4)
25		(TCV_counter_c := TCV_counter_c+1, TCV_ch := TCV_ch1, TCV_Null := OM_CphMd(TCV_ch, CphMod_02, TCV_CphKey), TCV_ia_ts:= TSPX_TmSltNotZero)			Back to SDCCH 8_A_1, non cipherin g mode
26		+Wait(C_T_Wait1stChReq)			
27		[(TCV_counter_k = 1) AND (TCV_counter_c = 6)]			no test for k=1 and c=6
28		(TCV_counter_c := (TCV_counter_c+1), TCV_ch := TCV_ch1)			
		ltree_chtype			
29		[TSPC_MOsvc]			
30		+BasicServiceMO(TSPX_MOBscSvcD, TSPX_MOChRateD)			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
31		(TCV_hotime := 500, TCV_Cntref := TSPX_NoOfHoAccA)			TCH
32		[TSPC_NonCallSS OR TSPC_Serv_TS22]			
33		(TCV_ChMod := ChMod_sign_iei, TCV_hotime := 750, TCV_Cntref:= TSPX_NoOfHoAccI)			SDCCH 8
34		[C_Yes]		I	
		ltree_hocmd			
35		[TCV_counter_k = 1]			for k = 1
36		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch1, HandOverCmd_fh(TCV_Tchtype, TCV_ts, TSPX_TscDef, C_NCC_1, C_BCC_3, C_arfcn_965, TCV_n, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_short(TCV_flist), Frql_omit, CellChDes_Omit, TCV_ChMod, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))		
37		[TCV_counter_k = 2]			for k = 2
38		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch1, HandOverCmd_fh(TCV_Tchtype, TCV_ts, TSPX_TscDef, C_NCC_1, C_BCC_3, C_arfcn_965, TCV_n, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql(TCV_flist, TCV_flistl), CellChDes_Omit, TCV_ChMod, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))		
39		[TCV_counter_k = 3]			for k = 3

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
40		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch1, HandOverCmd_fh(TCV_Tchtype, TCV_ts, TSPX_TscDef, C_NCC_1, C_BCC_3, C_arfcn_965, TCV_n, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_omit, Frql_omit, CellChDes_20_Be(TCV_cchdescr), TCV_ChMod, Freqchseq_omit, MoblAllc_20_Be2iei(TCV_mar2, TCV_mar1), RelTmdDif_omit, TimingAdv_omit, CphMod_omit))		
		ltree_chconfig			
41		+local_setHoppingPara			
42		+local_configChOfCellB			
		local_setHoppingPara			
43		[TCV_counter_k = 1]			
44		[TCV_counter_c = 1]			
45		(TCV_flist := C_fshlist_r_401, TCV_flistl:='05'O, TCV_mar1:='00000111'B, TCV_n := 3)			k =1 and c =1
46		[TCV_counter_c = 2]			
47		(TCV_flist := C_fshlist_r_419, TCV_flistl:='06'O, TCV_mar1:='00011111'B, TCV_n := 5)			k =1 and c =2
48		[TCV_counter_c = 3]			
49		(TCV_flist := C_fshlist_r_417, TCV_flistl:='06'O, TCV_mar1:= '00001111'B, TCV_n := 4)			k =1 and c =3
50		[TCV_counter_c = 4]			
51		(TCV_flist := C_fshlist_r_404, TCV_flistl:='09'O, TCV_mar1:='00111111'B, TCV_n := 6)			k =1 and c =4
52		[TCV_counter_c = 5]			
53		(TCV_flist := C_fshlist_r_418, TCV_flistl:='09'O, TCV_mar1:='00111111'B, TCV_n := 6)			k =1 and c =5
54		[TCV_counter_k = 2]			
55		[TCV_counter_c = 1]			
56		(TCV_flist := C_flist_r_401, TCV_flistl:='05'O, TCV_mar1:='00000111'B, TCV_n := 3)			k =2 and c = 1
57		[TCV_counter_c = 2]			
58		(TCV_flist := C_flist_r_419, TCV_flistl:='10'O, TCV_mar1:='00011111'B, TCV_n := 5)			k =2 and c = 2
59		[TCV_counter_c = 3]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
60		(TCV_flist := C_flist_r_417, TCV_flistl:='10'O, TCV_mar1:= '00001111'B, TCV_n := 4)			k =2 and c = 3
61		[TCV_counter_c = 4]			
62		(TCV_flist := C_flist_r_404, TCV_flistl:='06'O, TCV_mar1:='00001111'B, TCV_n := 4)			k =2 and c = 4
63		[TCV_counter_c = 5]			
64		(TCV_flist := C_flist_r_405, TCV_flistl:='07'O, TCV_mar1:='00111111'B, TCV_n := 6)			k =2 and c = 5
65		[TCV_counter_c = 6]			
66		(TCV_flist := C_flist_r_418, TCV_flistl:='10'O, TCV_mar1:='00000111'B, TCV_n := 3)			k =2 and c = 6
67		[TCV_counter_k = 3]			
68		[TCV_counter_c = 1]			
69		(TCV_cchdescr := C_cchd_r_416, TCV_mar1:='00010010'B, TCV_mar2:='00010000'B, TCV_flist:= C_flist_r_416, TCV_flistl:='10'O, TCV_n := 3)			k =3 and c = 1
70		[TCV_counter_c = 2]			
71		(TCV_cchdescr := C_cchd_r_408, TCV_mar1:='000101100'B, TCV_mar2:='01001000'B, TCV_flist:= C_flist_r_408, TCV_flistl:='0D'O, TCV_n := 5)			k =3 and c = 2
72		[TCV_counter_c = 3]			
73		(TCV_cchdescr := C_cchd_r_411, TCV_mar1:='00010010'B, TCV_mar2:='00010001'B, TCV_flist:= C_flist_r_411, TCV_flistl:='10'O, TCV_n := 4)			k =3 and c = 3
74		[TCV_counter_c = 4]			
75		(TCV_cchdescr := C_cchd_r_410, TCV_mar1:='00001111'B, TCV_mar2:='00000110'B, TCV_flist:= C_flist_r_410, TCV_flistl:='0B'O, TCV_n := 6)			k =3 and c = 4
76		[TCV_counter_c = 5]			
77		(TCV_cchdescr := C_cchd_r_411, TCV_mar1:='00010010'B, TCV_mar2:='01001001'B, TCV_flist:= C_flist_r_411, TCV_flistl:='07'O, TCV_n := 5)			k =3 and c = 5
78		[TCV_counter_c = 6]			
79		(TCV_cchdescr := C_cchd_r_412, TCV_mar1:='00000000'B, TCV_mar2:='00000111'B, TCV_flist:= C_flist_r_412, TCV_flistl:='10'O, TCV_n := 3)			k =3 and c = 6
		local_configChOfCellB			
80		(TCV_cellid:=C_CellB, TCV_Null := OM_ChangeRFOf2Cells(C_CellA, C_E_default, C_CellB, C_E_neighbourdefault))			
81		[TSPC_MOsvc]			2.
82		[TCV_counter_k <> 3]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
83		+FullRateCh_B_1_R (C_Asynho, TCV_ts, TSPX_TscDef, TCV_ChMod, FreqTCH_rf1(TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(20), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
84		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinfo5bis_B)			
85		[TCV_counter_k = 3]			
86		+FullRateCh_B_1_R (C_Asynho, TCV_ts, TSPX_TscDef, TCV_ChMod, FreqTCH_rf2(TCV_mar1, TCV_mar2, TCV_flist, TCV_flistl, TCV_n), TimingAdv(20), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
87		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinfo5bis_B)			
88		[TSPC_NonCallSS OR TSPC_Serv_TS22]			3.
89		[TCV_counter_k <> 3]			
90		+SDCCH8_B_1_R (TSPX_SDCCH8SubDef, C_Asynho, TCV_ts, TSPX_TscDef, TCV_ChMod, FreqTCH_rf1(TCV_mar1, TCV_flist, TCV_flistl, TCV_n), TimingAdv(20), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
91		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinfo5bis_B)			
92		[TCV_counter_k = 3]			
93		+SDCCH8_B_1_R (TSPX_SDCCH8SubDef, C_Asynho, TCV_ts, TSPX_TscDef, TCV_ChMod, FreqTCH_rf2(TCV_mar1, TCV_mar2, TCV_flist, TCV_flistl, TCV_n), TimingAdv(20), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
94		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinfo5bis_B)			

Detailed Comments : 1. Values TCV_flist , TCV_mae1 and TCV_mae2 are initialized for hopping frequencies : 20, 66, 78 in cell A.
2. for HO from SDCCH8 no FH in cell A to TCH/F FH in cell B
3. for HO from SDCCH8 no FH in cell A to SDCCH8 FH in cell B

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_2_4_2R

Group : Signalling/

Purpose : To check that the MS correctly returns to the old channel in the case of an handover failure caused by a layer 1 failure on the target cell, even if the origin is in the P−GSM band and the target in the R−GSM band.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		(TCV_asscmd_ts:= TSPX_TmSltNotZero, TCV_ts:= TSPX_TmSltNotZero1, TCV_Horf:= TSPX_HoRefA, TCV_Pwrlvl_ho:= '01100'B)			
3		+BasicServiceMT(TSPX_MTBscSvcD, TSPX_MTChRateD)			
4		+IdleUpdated_R(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_20_Aman, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubA, TSPX_CKSNDef, TSPX_RANDDef, C_arfcn_990, TSPX_IMSI, C_Test_rgsm, C_NCCP_2)			
5		+StartCellB_R (C_E_neighbourdefault, C_arfcn_965, C_arfcn_965, C_Immass, TCV_slot, TCV_tsc, TimingAdv(0), C_TxInt_5, C_Max_1, 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, BcchFreqLst_01, C_noRestablishment, C_BCC_3, C_NCC_1)			
6		+ltree_configTchA			
7		+ltree_u10			
8	body	+ltree_body			
		ltree_body			
9		(TCV_flist := C_fshlist_r_420, TCV_flistl := '05'O, TCV_mar1:= '00000111'B, TCV_n := 3, TCV_ch := TCV_chTch, TCV_sacch := TCV_sacchTch)			
10		+ltree_configTchB			
11		(TCV_L1Head0 := OM_GetL1Hd(TCV_sacch), TCV_Fn1 := OM_ComingFn(TCV_ch), TCV_Null := OM_SendNextOn(TCV_ch, TCV_Fn1))			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
12		L!DL_DatRqHoCmd	HndOvSnd(TCV_ch, HandOverCmd_fh(TCV_chtype, TCV_ts, TCV_tsc, C_NCC_1, C_BCC_3, C_arfcn_965, TCV_n, TSPX_HSN, TCV_Horf, TCV_Pwrlvl_ho, Synchi_omit, Frql_short(TCV_flist), Frql_omit, CellChDes_Omit, TCV_ChMod, Freqchseq_omit, MoblAllc_omit, RelTmdDif_omit, TimingAdv_omit, CphMod_omit))		
13		ACTIVATE(RcvHdOvAcc, OtherEventsFail)			
14		+l_hofl			
		l_hofl			
15		L?DL_RacInHoacc	HndOvAccRcv(TCV_chTch, HandOverAcc_02(TCV_Horf))		
16		(TCV_Null := OM_Deactivate("dummy", TCV_sacchTch))			Lower layer failure
17		L?DL_EstIn	DLEstInd(TCV_ch)		
18		(TCV_HoaccPara := OM_GetHoaccPara(TCV_chTch), TCV_L1Head := OM_GetL1Hd(TCV_sacch))			
19		L?DL_DatInHofl (TCV_Fn := DL_DatInHofl.fn)	HndOvFlRcv(TCV_ch, HandOvFail_02)		
20		+PostMainLinkRel(TCV_ch)			
21		+Check_Time(3000, TCV_Fn1, TCV_Fn)			
22		+Check_HoaccPara_R (0, TCV_Pwrlvl_ho, TCV_HoaccPara)			
23	L_0564	[TCV_L1Head.mspwrlvl = TCV_L1Head0.mspwrlvl]		(P)	
24	L_0565	[TCV_L1Head.mspwrlvl <> TCV_L1Head0.mspwrlvl]		(F)	
		ltree_u10			
25		+RRmtcallprepare(TimingAdv(0), TSPX_RANDDef)			
26		(TCV_Setup_mt.sig := Signal_01)			
27		L!DL_DatRqSetup	SetupSnd(TCV_ch, TCV_Setup_mt)		
28		L?DL_DatInCallCo	CallCfm(CallConfirm_01(TI_01))		1.
29		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
30		+localtree			
31		L?DL_DatInAlert	AlertRcv(AlertingInd_01(TI_01))		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
32		(TCV_Null := OO_HookOff())			
33		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
34		+localtree			
		localtree			
35		(TCV_AssCmd := AsgnCmd_nfh(TCV_chtype, TCV_asscmd_ts, TCV_tsc, 10, TSPX_TCHcarrierA_ho, CellChDes_Omit, ChMod_omit, StartingTm_omit, CphMod_omit))			
36		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			2.
37		L!DL_DatRqConnAck	ConnAckSnd(TCV_chTch, ConnectAck_01(TI_02))		
		ltree_configTchA			
38		(TCV_chTch := C_FACCHF_A_1, TCV_sacchTch := C_SACCHF_A_1, TCV_chtype := '00001'B)			
39		+ Config_FACCHF_A_1 (63 , 10, TCV_ChMod, C_Ass, TCV_asscmd_ts, TCV_tsc, FreqTCH(TSPX_TCHcarrierA_ho), C_TCHF_ACCHF_1, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
40		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			
41		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			
42		(TCV_tch_arfcn := TSPX_TCHcarrierA_ho)			
		ltree_configTchB			
43		(TCV_chTch := C_FACCHF_B_1, TCV_sacchTch := C_SACCHF_B_1)			
44		+Config_FACCHF_B_1(63, 12, TCV_ChMod, C_Rcv, TCV_ts, TCV_tsc, FreqTCH_rf1(TCV_mar1, TCV_flist, TCV_flistl, TCV_n), C_TCHF_ACCHF_1, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
45		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			
46		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments : 1. If the MS supports the bearer capabilities, which are give in Setup message, it has to accept them. Therefor, they are no bearer capabilites expected in Call Confirm message.
2. TCH/F or SDCCH/4 with non hopping in selected cell, power level = 10.

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_2_5R

Group : Signalling/

Purpose : 1) To verify that the MS, after receiving a FREQUENCY REDEFINITION message, starts using the new frequencies and hopping sequence when some R−GSM freuencies are used.

2) To check that the last received Cell Channel Description information element is used to decode the Mobile Allocation IE received in the FREQUENCY REDEFINITION message.

Configuration :

Default : OtherEventsFail

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	pre	START T_guard(300)			
2		+IdleUpdated_R (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_1nonComb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_22, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TSPX_IMSI, C_Test_rgsm, C_NCCP_2)			
3		(TCV_ia_ts:= TSPX_TmSltNotZero)			
4		(TCV_Cnt1 := 1)			
5	body	REPEAT localtree_body UNTIL [TCV_Cnt1 = 7]			
		localtree_body			
6		+local_ChConfig			
7		+Wait(C_T_Wait1)			
8		L!DL_UdatRqPg1Rq	PgReq1(TCV_PgCh, TCV_Pgg, PgReqTp1_01)		
9		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_17)		
10		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
11		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_221(TCV_Rr, TCV_Fn, TCV_ia_ts, TSPX_TscDef, TCV_chtype, '000000'B, '000100'B, TimingAdv(30), MoblAllc_04man, '01'O))		
12		L?DL_EstInPgRes	PgRes(TCV_ch, PagingRes_01)		
13		ACTIVATE(OtherEventsFail)			Restore Normal default
14		+ltree_setFreqRedefPara			
15		+ltree_LowerTesterManage			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
16		+ltree_sendFreqRedef			
17		+ltree_checkchnafterfreqred			
18		+ChanRel(TCV_ch)			
19		(TCV_Cnt1 := (TCV_Cnt1 +1), TCV_Null := OM_FreeResource(TCV_ch))			
		ltree_checkchnafterfreqred			
20		[(TCV_Cnt1 = 4) OR (TCV_Cnt1 = 6)]			Ma1)
21		(TCV_Res := OM_FHCHK(TCV_ch, CellChDes_20_Be(TCV_cchdescr), MoblAllc_20_Be1 (TCV_mar1), TCV_chd1, 100, TCV_Fn))			
22		[NOT TCV_Res]		(F)	
23		[TCV_Res]		(P)	
24		[(TCV_Cnt1 <> 4) AND (TCV_Cnt1 <> 6)]			Ma2)
25		(TCV_Res := OM_FHCHK(TCV_ch, CellChDes_20_Be(TCV_cchdescr), MoblAllc_20_Be2(TCV_mar1, TCV_mar2), TCV_chd1, 100, TCV_Fn))			
26		[NOT TCV_Res]		(F)	
27		[TCV_Res]		(P)	
		ltree_LowerTesterManage			
28		(TCV_Fn := OM_ComingFn(TCV_ch), TCV_chd1 := ChDescrp_fh(TCV_chtype, TSPX_TmSltNotZero1, TSPX_TscDef, INT_TO_BIT((TSPX_MAIO MOD TCV_n), 6), INT_TO_BIT(TSPX_HSN, 6)), TCV_Strt := OC_StartTime(TCV_Fn, 100, 0), TCV_Null := OM_SendNextOn(TCV_ch, TCV_Fn))			
29		[(TCV_Cnt1 = 4) OR (TCV_Cnt1=6)]			Ma1)
30		(TCV_Null := OM_FreqDef(TCV_Strt, MoblAllc_20_Be1(TCV_mar1), TCV_ch, TCV_chd1, CellChDes_20_Be(TCV_cchdescr), TCV_Fn))			
31		[(TCV_Cnt1 <>4) AND (TCV_Cnt1<>6)]			Ma2)
32		(TCV_Null := OM_FreqDef(TCV_Strt, MoblAllc_20_Be2(TCV_mar1, TCV_mar2), TCV_ch, TCV_chd1, CellChDes_20_Be(TCV_cchdescr), TCV_Fn))			
		ltree_sendFreqRedef			
33		[TCV_Cnt1 <> 6]			
34		L!DL_DatRqFrqre	FrqRedfSnd(TCV_ch, FreqRedef_22(TCV_ia_ts, TCV_chtype, TCV_cchdescr, TCV_mar1, TCV_mar2, TCV_Strt, TCV_n))		
35		[TCV_Cnt1 = 6]			Ma1)
36		L!DL_DatRqFrqre	FrqRedfSnd(TCV_ch, FreqRedef_21(TCV_ia_ts, TCV_chtype, TCV_mar1, TCV_Strt, TCV_n))		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
		ltree_setFreqRedefPara			
37		[TCV_Cnt1 = 1]			
38		(TCV_cchdescr := C_cchd_r_407, TCV_mar1:='00010010'B, TCV_mar2:='00010000'B, TCV_n := 3)			
39		[TCV_Cnt1 = 2]			
40		(TCV_cchdescr := C_cchd_r_408, TCV_mar1:='00001110'B, TCV_mar2:='01001000'B, TCV_n := 5)			
41		[TCV_Cnt1 = 3]			
42		(TCV_cchdescr := C_cchd_r_416, TCV_mar1:='00010001'B, TCV_mar2:='00010001'B, TCV_n := 4)			
43		[TCV_Cnt1 = 4]			
44		(TCV_cchdescr := C_cchd_r_415, TCV_mar1:='00000011'B, TCV_mar2:='11000110'B, TCV_n := 6)			
45		[TCV_Cnt1 = 5]			
46		(TCV_cchdescr := C_cchd_r_411, TCV_mar1:='00010100'B, TCV_mar2:='10010001'B, TCV_n := 5)			
47		[TCV_Cnt1 = 6]			
48		(TCV_cchdescr := C_cchd_r_414, TCV_mar1:='00001110'B, TCV_n := 3)			
		local_ChConfig			
49		[TSPC_SvcOnTCH]			
50		+FullRateCh_A_1_nociph_R (C_Immass, TCV_ia_ts, TSPX_TscDef, ChMod_sign, Freq_rg8, TimingAdv(30), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
51		(TCV_ch := TCV_chTch)			
52		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
53		[C_Yes]			
54		(TCV_chtype := INT_TO_BIT((8 + BIT_TO_INT(TSPX_SDCCH8SubDef)),5))			
55		+SDCCH8_A_1_nociph_R (TSPX_SDCCH8SubDef, C_Immass, TCV_ia_ts, TSPX_TscDef, ChMod_sign, Freq_rg8, TimingAdv(30), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
56		+SysInfo_5bisSending(TCV_sacch8, TCV_sysinf5bis)			
Detailed Comments : Ma1) Mobile Allocation with 1 octets Ma2) Mobile Allocation with 2 octets					

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_3_1R

Group : Signalling/

Purpose : 1) To verify that the MS in MM state "idle, updated" and in RR idle mode with a TMSI assigned, when made to initiate a call for a selected teleservice that is supported ba the MS as declared in a PICS/PIXIT statement, starts to initiate an immediate assignment procedure bz sending the CHANNEL REQUEST message.

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 authenticiation 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 an appropriate TCH, 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.)

Configuration :

Default : OtherEventsFail_01

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_guard(300)			
2		+ ltree_SelectTeleservice			
3		+ IdleUpdated_R (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_1nonComb, C_BPM_3, C_T3212_0, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_22, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TSPX_IMSI, C_Test_rgsm, C_NCCP_2)			
4		(TCV_chdescr_arfcn:= C_arfcnEgsm_iacmd, TCV_ia_ts := TSPX_TmSltG, TCV_asscmd_ts := TSPX_TmSltC)			
5		+ SDCCH8_A_1_nociph_R (TSPX_SDCCH8SubDef, C_Immass, TCV_ia_ts, TCV_tsc, ChMod_sign, FreqTCH(TCV_chdescr_arfcn), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			
6		+ SysInfo_5bisSending (TCV_sacch8, TCV_sysinf5bis)			
7		+ InitCall (TCV_Service)			
8		+ local_continue			
		local_continue			
9		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq (ChRequest_04)		

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
10		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
11		L!DL_UdatRqImmass	ImmAss (TCV_agch, ImmAsgn_27(TCV_Rr, TCV_Fn, TCV_ia_ts, TCV_chdescr_arfcn, TimingAdv(0)))		
12		L?DL_EstInCmsRq	CMSerReq(CMServiceReq_01)		
13		ACTIVATE(OtherEventsFail)			Restore Normal default
14		+Authentication(TCV_ch, TCV_cksn, TSPX_RANDDef)			
15		+Ciphering_on(TCV_ch)			
16		+ltree_ccest			
		ltree_SelectTeleservice			
17		[TSPC_Serv_TS11]			
18		+BasicServiceMO(C_Telephony, C_Full)			
19		[TSPC_Serv_TS61]			
20		+BasicServiceMO(C_AltSpchFax, C_Full)			
21		[TSPC_Serv_TS62]			
22		+BasicServiceMO(C_AutoFax, C_Full)			
		ltree_ccest			
23		(TCV_Null := OM_FreeResource(TCV_ch))			
24		+FullRateCh_A_1_R (C_Ass, TCV_asscmd_ts, TCV_tsc, ChMod_speech, FreqTCH_omit, C_arfcn_965, TimingAdv(0), C_BABR_0, C_cch_1Comb, C_BPM_3)			
25		+SysInfo_5bisSending(TCV_sacchTch, TCV_sysinf5bis)			
26		+RcvSetupOrEsetup(TCV_Setup_mo, TCV_Esetup, TCV_Ecall)			
27		L!DL_DatRqCallProc	CallProc(TCV_ch, TCV_CallProc)		
28		L!DL_DatRqAlert	AlertSnd(TCV_ch, Alerting_01(TCV_TI))		
29		(TCV_AssCmd := AsgnCmd_nfh('00001'B, TCV_asscmd_ts, TSPX_TscDef, 9, C_arfcn_965, CellChDes_Omit, TCV_ChMod, StartingTm_omit, CphMod_omit))			
30		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
31		L!DL_DatRqConn	ConnSnd(TCV_chTch, Connect_02(TCV_TI))		
32		L?DL_DatInConnAck	ConnAckRcv(ConnectAck_02(TCV_TI0))		
33		(TCV_Res := OO_TCHThroConnCHK())			
34		+Check_Result			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
35		(TCV_Null := OO_TermCall())			
36		L?DL_DatInDisc	DiscRcv(TCV_chTch, DisconnR(TCV_TI0, Cause_Def))		
37		L!DL_DatRqRel	ReleaseSnd(TCV_chTch, Release_08(TCV_TI))		
38		L?DL_DatInRelCmp	RelComRcv(ReleaseCmp_03(TCV_TI0))		
39		+PostMainLinkRel(TCV_chTch)			
Detailed Comments :					

Test Case Dynamic Behaviour

Test Case Name : TC_26_10_3_2R

Group : Signalling/

Purpose : 1) To verify that the MS in the "idle, no IMSI" state (no SIM inserted) 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 specifiying the correct CKSN and TMSI, with CM Service Type "emergency call establishment".

3) To verify that after receipt of a CM SERVICE ACCEPT message from the SS, the MS sends an EMERGENCY SETUP message.

4) To verify that subsequently, the SS having sent a CALL PROCEEDING message and then an ALERT message and having initiated the assignment procedure, the MS performs correctly that assignment procedure.

5) To verify subsequent correct performance of a connect procedure.

6) To verify that subsequently the MS has throught connected the TCH in both directions.

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(300)			
2		+BasicServiceMO(C_EmgCall, TSPX_EmgCallRate)			
3		+IdleUpdated_R (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_1nonComb, C_BPM_3, C_T3212_1, C_ci_cellA, C_MCC_1, C_PLMN_1, C_LAC_1, CellOpt_01, CellChDes_20_Aman, C_CellReselectHys12, C_MaxPwrLvlG, BcchFreqLst_01, BcchFreqLst_Omit, BcchFreqLst_r46, C_noRestablishment, C_BCC, C_NCC, C_Start_Tmsi, TSPX_SDCCH4SubDef, TSPX_CKSNDef, TSPX_RANDDef, C_arfcnA, TSPX_IMSI, C_Test_fh, C_NCCP_2)			
4		(TCV_chdescr_arfcn:= C_arfcnEgsm_iacmd, TCV_ia_ts:= TSPX_TmSltG)			
5		+SDCCH8_A_1_nociph_R (TSPX_SDCCH8SubDef, C_Immass, TCV_ia_ts, TCV_tsc, ChMod_sign, FreqTCH(TCV_chdescr_arfcn), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			1.
6		+local_TrafficChConf			
7	body	+InitCall(TCV_Service)			
8		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_16)	(P)	2.

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
9		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
10		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_27(TCV_Rr, TCV_Fn, TCV_ia_ts, TCV_chdescr_arfcn, TimingAdv(0)))		
11		L?DL_EstInCmsRq	CMSerReq(CMServiceReq_06(TSPX_CKSNDef))	(P)	3.
12		ACTIVATE(OtherEventsFail)			Restore Normal default
13		+local_continue			
		local_continue			
14		L!DL_DatRqCmsAcp	CMSerAcp(TCV_ch, CMServiceAcp_01)		
15		+RcvSetupOrEsetup(TCV_Setup_mo, TCV_Esetup, TCV_Ecall)			
16		L!DL_DatRqCallProc	CallProc(TCV_ch, TCV_CallProc)		
17		L!DL_DatRqAlert	AlertSnd(TCV_ch, Alerting_01(TCV_TI))		
18		+AssCmdGen(TCV_cellid, TCV_ChRate, TSPX_TmSltDef, TSPX_TscDef, TSPX_TCHHSubDef)			
19		(TCV_AssCmd.ch1d_at.arfcn:= INT_TO_BIT(C_arfcn_965,10))			
20		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
21		L!DL_DatRqConn	ConnSnd(TCV_chTch, Connect_02(TCV_TI))		
22		L?DL_DatInConnAck	ConnAckRcv(ConnectAck_02(TCV_TI0))	(P)	
23		(TCV_Res := OO_TCHThroConnCHK())			
24		+Check_Result			
25		L!DL_DatRqDisc	DiscSnd(TCV_chTch, DisconnS(TCV_TI, Cause_01, ProgInd_omit, UuInfo_omit))		
26		L?DL_DatInRel	ReleaseRcv(Release_10(TCV_TI0))	(P)	
27		L!DL_DatRqRelCmp	RelComSnd(TCV_chTch, ReleaseCmp_08(TCV_TI))		
28		+PostMainLinkRel(TCV_chTch)			
		local_TrafficChConf			
29		[TCV_ChRate = C_Full]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
30		+FullRateCh_A_1_nociph_R (C_Ass, TSPX_TmSltDef, TSPX_TscDef, ChMod_speech, FreqTCH(C_arfcn_965), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			4.
31		[TCV_ChRate = C_Half]			
32		+HalfRateCh_A_1_nociph_R (TSPX_TCHHSubDef, C_Ass, TSPX_TmSltDef, TSPX_TscDef, ChMod_speech, FreqTCH(C_arfcn_965), TimingAdv(0), C_BABR_0, C_cch_1nonComb, C_BPM_3)			5.

Detailed Comments :
1. To setup a physical 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. To setup aphysical channel as full rate traffic channel.
5. To setup aphysical channel as half rate traffic channel.

Test Step Dynamic Behaviour

Test Step Name : CCConfigTCH_R(sub : B_1; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3; mod : B_8; Freqg : FRQPARA; arfcng : INTEGER)

Group : management/ChConfig/

Objective : Configure Tester for TCH/H or TCH/F depending on TCV_ChRate.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[TCV_ChRate = C_Full]			1.
2		+FullRateCh_A_1_R(acttype, slot, tsc, ChMod(mod), Freqg, arfcng , ta, babr, cch_con, bpm)			
3		[TCV_ChRate = C_Half]			2.
4		+HalfRateCh_A_1_R(sub, acttype, slot, tsc, ChMod(mod), Freqg, arfcng , ta, babr, cch_con, bpm)			

Detailed Comments : 1. Full rate channel needed, to setup a physical channel as full rate traffic channel.
2. Half rate channel needed, to setup the physical channel as half rate traffic channel.

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/ChConfig/

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_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/ChConfig/

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/ChConfig/

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/ChConfig/

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/ChConfig/

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/ChConfig/

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/ChConfig/

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/ChConfig/

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_BCCH_CCCH_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH 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, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_D_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell D.

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_CellD, C_FCCH_D, C_SCH_D, C_BCCH_D_1, C_PCH_D_1, C_AGCH_D_1, C_RACH_D_1, "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_BCCH_CCCH_E_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH and CCCH onto the physical channel which represents cell E.

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_CellE, C_FCCH_E, C_SCH_E, C_BCCH_E_1, C_PCH_E_1, C_AGCH_E_1, C_RACH_E_1, "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/ChConfig/

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/ChConfig/

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/ChConfig/

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/ChConfig/

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_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/ChConfig/

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_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/ChConfig/

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_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/ChConfig/

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_SDCCH4_D_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell D

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_CellD, C_FCCH_D, C_SCH_D, C_BCCH_D_1, C_PCH_D_1, C_AGCH_D_1, C_RACH_D_1, C_SDCCH4_D, C_SACCHC4_D, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_D, C_SDCCH40_D, C_SDCCH41_D, C_SDCCH42_D, C_SDCCH43_D, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_D, C_SACCHC40_D, C_SACCHC41_D, C_SACCHC42_D, C_SACCHC43_D, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_E_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell E

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_CellE, C_FCCH_E, C_SCH_E, C_BCCH_E_1, C_PCH_E_1, C_AGCH_E_1, C_RACH_E_1, C_SDCCH4_E, C_SACCHC4_E, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_E, C_SDCCH40_E, C_SDCCH41_E, C_SDCCH42_E, C_SDCCH43_E, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_E, C_SACCHC40_E, C_SACCHC41_E, C_SACCHC42_E, C_SACCHC43_E, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_F_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell F

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_CellF, C_FCCH_F, C_SCH_F, C_BCCH_F_1, C_PCH_F_1, C_AGCH_F_1, C_RACH_F_1, C_SDCCH4_F, C_SACCHC4_F, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_F, C_SDCCH40_F, C_SDCCH41_F, C_SDCCH42_F, C_SDCCH43_F, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_F, C_SACCHC40_F, C_SACCHC41_F, C_SACCHC42_F, C_SACCHC43_F, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_G_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell G

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_CellG, C_FCCH_G, C_SCH_G, C_BCCH_G_1, C_PCH_G_1, C_AGCH_G_1, C_RACH_G_1, C_SDCCH4_G, C_SACCHC4_G, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_G, C_SDCCH40_G, C_SDCCH41_G, C_SDCCH42_G, C_SDCCH43_G, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_G, C_SACCHC40_G, C_SACCHC41_G, C_SACCHC42_G, C_SACCHC43_G, "dummy", "dummy", "dummy", "dummy"))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH4_H_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/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell H

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_CellH, C_FCCH_H, C_SCH_H, C_BCCH_H_1, C_PCH_H_1, C_AGCH_H_1, C_RACH_H_1, C_SDCCH4_H, C_SACCHC4_H, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH4_H, C_SDCCH40_H, C_SDCCH41_H, C_SDCCH42_H, C_SDCCH43_H, "dummy", "dummy", "dummy", "dummy", C_SACCHC4_H, C_SACCHC40_H, C_SACCHC41_H, C_SACCHC42_H, C_SACCHC43_H, "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/ChConfig/

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 : Config_SDCCH8_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/ChConfig/

Objective : To set one physical channel used as SDCCH8 channel 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_SDCCH8_A_2, C_SACCHC8_A_2, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH8_A_2, C_SDCCH80_A_2, C_SDCCH81_A_2, C_SDCCH82_A_2, C_SDCCH83_A_2, C_SDCCH84_A_2, C_SDCCH85_A_2, C_SDCCH86_A_2, C_SDCCH87_A_2, C_SACCHC8_A_2, C_SACCHC80_A_2, C_SACCHC81_A_2, C_SACCHC82_A_2, C_SACCHC83_A_2, C_SACCHC84_A_2, C_SACCHC85_A_2, C_SACCHC86_A_2, C_SACCHC87_A_2))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH8_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/ChConfig/

Objective : To set one physical channel used as SDCCH8 channel for instance 1 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_SDCCH8_B_1, C_SACCHC8_B_1, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH8_B_1, C_SDCCH80_B_1, C_SDCCH81_B_1, C_SDCCH82_B_1, C_SDCCH83_B_1, C_SDCCH84_B_1, C_SDCCH85_B_1, C_SDCCH86_B_1, C_SDCCH87_B_1, C_SACCHC8_B_1, C_SACCHC80_B_1, C_SACCHC81_B_1, C_SACCHC82_B_1, C_SACCHC83_B_1, C_SACCHC84_B_1, C_SACCHC85_B_1, C_SACCHC86_B_1, C_SACCHC87_B_1))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Config_SDCCH8_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/ChConfig/

Objective : To set one physical channel used as SDCCH8 channel for instance 1 of 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, "dummy", "dummy", "dummy", "dummy", "dummy", "dummy", C_SDCCH8_C_1, C_SACCHC8_C_1, "dummy"))			
2		(TCV_Null := OM_Assoc(C_SDCCH8_C_1, C_SDCCH80_C_1, C_SDCCH81_C_1, C_SDCCH82_C_1, C_SDCCH83_C_1, C_SDCCH84_C_1, C_SDCCH85_C_1, C_SDCCH86_C_1, C_SDCCH87_C_1, C_SACCHC8_C_1, C_SACCHC80_C_1, C_SACCHC81_C_1, C_SACCHC82_C_1, C_SACCHC83_C_1, C_SACCHC84_C_1, C_SACCHC85_C_1, C_SACCHC86_C_1, C_SACCHC87_C_1))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_A_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_SDCCH4_A_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACC HC4, ta, babr, cch_con, bpm)			

Detailed Comments : 1. For P, E−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_A_CBMS_R(par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+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)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_B_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_SDCCH4_B_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACC HC4, ta, babr, cch_con, bpm)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_C_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+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)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_D_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell D.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+Config_SDCCH4_D_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_E_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell E.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+Config_SDCCH4_E_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_F_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell F.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+Config_SDCCH4_F_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), 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_G_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell G.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+Config_SDCCH4_G_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : CombinedBCCH_H_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set a physical channel and map FCCH, SCH, BCCH, CCCH and SDCCH4 onto the physical channel which represents cell H.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+Config_SDCCH4_H_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcn1), C_FCCH_SCH_BCCH_CCCH_SDCCH4_SACCHC4, ta, babr, cch_con, bpm)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A_R(par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+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)			

Detailed Comments : 1. For P−GSM900.
2. For DCS1800.

Test Step Dynamic Behaviour

Test Step Name : NonCombinedBCCH_A_2_R (acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_BCCH_CCCH_A_2(63, 19, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnA), 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_R(acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_BCCH_CCCH_A_3(63, 19, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnA), 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_R(acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_BCCH_CCCH_A_4(63, 19, ChMod_sign, acttype, slot, tsc, FreqBCCH(C_arfcnA), 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_B_R (par_bspwr, mtmcgsm, arfcn1 : INTEGER; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+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)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_1_R (acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; arfcng : INTEGER; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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_R(acttypeT, slotT, tscT, chmod, FreqTCH(arfcng) , ta, babr, cch_con, bpm)			
3		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			
4		(TCV_tch_arfcn:= arfcng)			
5		[arfcng = C_Noarfcn]			Hopping
6		+FullRateCh_A_1_nociph_R (acttypeT, slotT, tscT, chmod, Freqg , ta, babr, cch_con, bpm)			
7		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_1_nociph_R (acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_FACCHF_A_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHF_ACCHF_1, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_A_2_nociph_R (acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_FACCHF_A_2(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHF_ACCHF_2, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacchTch1, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FullRateCh_B_1_R(acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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_R (acttypeT, slotT, tscT, chmod, Freqg, 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_R (acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_FACCHF_B_1 (63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHF_ACCHF_1, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5_B, TCV_sysinfo6_B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_A_1_R(sub : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; arfcng : INTEGER; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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_R(sub, acttypeT, slotT, tscT, chmod, FreqTCH(arfcng) , ta, babr, cch_con, bpm)			
3		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			
4		(TCV_tch_arfcn:= arfcng)			
5		[arfcng = C_Noarfcn]			hopping
6		+HalfRateCh_A_1_nociph_R(sub, acttypeT, slotT, tscT, chmod, Freqg , ta, babr, cch_con, bpm)			
7		(TCV_Null := OM_CphMd(TCV_chTch, CphMod_01, TCV_CphKey))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : HalfRateCh_A_1_nociph_R (sub : B_1; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_FACCHH_A_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_TCHH_ACCHH_1, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			
4		+localtree			
		localtree			
5		[sub = '0'B]			
6		(TCV_chtype := '00010'B)			
7		[sub = '1'B]			
8		(TCV_chtype := '00011'B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SDCCH8_A_1_nociph_R(bstring : B_3; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

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		+Config_SDCCH8_A_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_SDCCH8_SACCHC8_1, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacch8, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SDCCH8_A_2_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/ChConfig/

Objective : To set one physical channel used as hopping SDCCH8 channel for instance 2 of cell A, for TC_26_6_13_1.

Default : OtherEvents

Comments : hopping parameters are from PIXIT, used as before time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch := OC_SubchOfSdcch8(bstring, C_CellA, 2), TCV_sacchTch := OC_SubchOfSacch8(bstring, C_CellA, 2))			
2		+Config_SDCCH8_A_2 (63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_SDCCH8_SACCHC8_2, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SDCCH8_B_1_R (bstring : B_3; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set one physical channel used as hopping SDCCH8 channel for instance 1 of cell B.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+SDCCH8_B_1_nociph_R(bstring, acttypeT, slotT, tscT, chmod, Freqg, 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 : SDCCH8_B_1_nociph_R(bstring : B_3; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set one physical channel used as hopping SDCCH8 channel for instance 1 of cell B For TC_26_6_13_5.

Default : OtherEvents

Comments : hopping parameters are from PIXIT, used by handover command after time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_chTch := OC_SubchOfSdcch8(bstring, C_CellB, 1), TCV_sacchTch := OC_SubchOfSacch8(bstring, C_CellB, 1))			
2		+Config_SDCCH8_B_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_SDCCH8_SACCHC8_1, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacchTch, TCV_sysinfo5_B, TCV_sysinfo6_B)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SDCCH8_C_1_nociph_R(bstring : B_3; acttypeT : BITSTRING; slotT : SN; tscT : TSC; chmod : CHMOD; Freqg : FRQPARA; ta : TA; babr, cch_con, bpm : B_3)

Group : management/ChConfig/

Objective : To set one physical channel used as hopping SDCCH8 channel for instance 1 of cell C For TC_26_6_13_5.

Default : OtherEvents

Comments : hopping parameters are from PIXIT, used by handover command after time channel.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_ch := OC_SubchOfSdcch8(bstring, C_CellC, 1), TCV_sacch := OC_SubchOfSacch8(bstring, C_CellC, 1))			
2		+Config_SDCCH8_C_1(63, 19, chmod, acttypeT, slotT, tscT, Freqg, C_SDCCH8_SACCHC8_1, ta, babr, cch_con, bpm)			
3		+SysInfo_SacchSending(TCV_sacch, TCV_sysinfo5, TCV_sysinfo6)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Adjust_gsmanddcs_powerlvl_R(powerlevel1 : INTEGER)

Group : Miscellaneous/

Objective :

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_AssCmd.pcmd := Pcmd_19(INT_TO_BIT(powerlevel1,5)))			

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		?TIMEOUT T_dlyAss		(F)	
3		+PostLinkRelEnd(oldch)			
4		L?DL_EstIn (TCV_FnAss := DL_EstIn.fn) CANCEL T_dlyAss	DLEstInd(newch)		
5		L?DL_DatInAssCom	AssCmp(newch, AsgnCmp_02)	(P)	
6		L!MDL_RelRq	MDLRelReq(oldch)		
7		L?DL_EstIn CANCEL T_dlyAss	DLEstInd(oldch)		
8		L?DL_DatInAssfl	AssFl_any_cau(oldch)	(F)	
9		+PostLinkRelEnd(oldch)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : AssCmdGen(cell : CellID; rate : RATE; 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			
		ltree_AsgnTchB			
15		(TCV_AssCmd := AsgnCmd_nfh(TCV_Tchtype, slot, tsc, 7, C_arfcn_tchB, 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 : 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		[NOT TCV_Res]		(F)	1)
5		[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) AND (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) AND (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			
16		[ccd.ccch_con = '??1'B]		I	
		localtree1			
17		(TCV_bs_ag_blks_res := BIT_TO_INT(ccd.babr), TCV_bs_pa_mfrms := (2 + BIT_TO_INT(ccd.bpm)))			
18		[ccd.ccch_con = '001'B]			i.e. combined
19		[(TCV_bs_ag_blks_res < 0) OR (TCV_bs_ag_blks_res > 2)]		I	Stop! A tester error.
20		[(TCV_bs_ag_blks_res >= 0) AND (TCV_bs_ag_blks_res <= 2)]			
21		(TCV_tmp := 3 − TCV_bs_ag_blks_res)			
22		+localtree2			
23		[ccd.ccch_con <> '001'B]			i.e. not combined
24		[(TCV_bs_ag_blks_res < 0) OR (TCV_bs_ag_blks_res > 7)]		I	Stop! A tester error.
25		[(TCV_bs_ag_blks_res >= 0) AND (TCV_bs_ag_blks_res <= 7)]			
26		(TCV_tmp := 9 − TCV_bs_ag_blks_res)			
27		+localtree2			ini. TCV Pgg
		localtree2			
28		(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
29		(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 : CCstatuschk_02(ch: LOGICCH; st : CCSTATE; ti_orig, ti_dest: TI)
Group : Miscellaneous/
Objective : To check whether the MS under test is in the CC state ‘st‘ and cause = #30.
Default : OtherEvents
Comments : if the cause value is #30 and the CC state value is ‘st‘, the preliminary verdict is pass.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqCcstEnq	CCStESnd(ch, CCStatusEq_01(ti_orig))		
2		L?DL_DatInCcst (TCV_Cau := DL_DatInCcst.msg.cau, TCV_CCst := DL_DatInCcst.msg.cst)	CCStRcv(ch, CCStatus_01(ti_dest))		
3		[(TCV_Cau.cau_class = ’001’B) AND (TCV_Cau.cau_va = ’1110’B)]			
4		[TCV_CCst.csv = INT_TO_BIT(st, 6)]			1.
5		[TCV_CCst.csv <> INT_TO_BIT(st, 6)]		(I)	2.
6		+PostLinkRelEnd(ch)			
7		[(TCV_Cau.cau_class <> ’001’B) OR (TCV_Cau.cau_va<> ’1110’B)]		(I)	3.
8		+PostLinkRelEnd(ch)			

Detailed Comments : 1. Now in CC state ’st’ and cause = #30.
2. Not in CC state ’st’.
3. Cause is not #30.

Test Step Dynamic Behaviour

Test Step Name : Check_HoaccPara_R(tadv: INTEGER; pwrlvl : B_5; hoapara : HOACC_PARA)
Group : Miscellaneous/
Objective : To check whether the access power level and the timing advance in 'hoapara' are within the tolerance.
Default : OtherEvents
Comments : GSM 05.05, 4.1.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+l_pwlchk			1.
2		+l_tachk			2.
		l_pwlchk			
3		+l_dbm			3.
4		+l_mapping			4.
5		+l_preset			5.
6		(TCV_counter_k := TCV_counter_k − TCV_kcnt * 2)			Output power in dBm
7		[(TCV_SMTypeM <= TCV_counter_k + TCV_counter_c) AND (TCV_SMTypeN >= TCV_counter_k − TCV_counter_c)]		(P)	
8		[C_Yes]		(F)	
		l_dbm			
9		(TCV_SMTypeM := OC_OctToInt(hoapara.max_mspwr), TCV_SMTypeN := OC_OctToInt(hoapara.min_mspwr))			Store input values, max positive 63
10		[TCV_SMTypeN > TCV_SMTypeM]		(I)	
11		[TCV_SMTypeM > 63]			
12		(TCV_SMTypeM := TCV_SMTypeM − 256, TCV_SMTypeN := TCV_SMTypeN − 256)			max_ms pwr <0
13		[TCV_SMTypeN > 63]			
14		(TCV_SMTypeN := TCV_SMTypeN − 256)			min_msp wr <0
15		[C_Yes]			
		l_mapping			
16		(TCV_kcnt := BIT_TO_INT(pwrlvl))			
17		[TCV_kcnt < 2]			
18		(TCV_kcnt := 2)			
19		[TCV_kcnt > 19]			
20		(TCV_kcnt := 19)			
21		[C_Yes]			
		l_preset			
22		(TCV_counter_k := 43)			Offset
23		[TCV_kcnt=2]			
24		(TCV_counter_c := 2)			−2 to +2 dBm
25		[(TCV_kcnt>2) AND (TCV_kcnt<16)]			
26		(TCV_counter_c := 3)			−3 to +3 dBm
27		[TCV_kcnt > 15]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
28		(TCV_counter_c := 5) l_tachk			−5 to +5 dBm
29		+l_tcvta			6.
30		[(TCV_Fk <= tadv + 1) AND (TCV_S >= tadv − 1)]		(P)	Within −1 to +1 bit
31		[(TCV_Fk > tadv + 1) OR (TCV_S < tadv − 1)] l_tcvta		(F)	
32		(TCV_Fk := BIT_TO_INT(hoapara.max_msta), TCV_S := BIT_TO_INT(hoapara.min_msta))			
33		[TCV_S > TCV_Fk]		(I)	
34		[TCV_Fk > 127]			
35		(TCV_Fk := TCV_Fk − 256, TCV_S := TCV_S − 256)			max_msta <0
36		[TCV_S > 127]			
37		(TCV_S := TCV_S − 256)			min_msta <0
38		[C_Yes]			

Detailed Comments : 1. Power level check.
2. Timing advance check.
3. Convert an octet in modulo 256 into integer in dBm.
4. Mapping the power control levels of 0 − 31 into the range between 2 to 19 for GSM and −3 to 15 for DCS.
5. Preset the tolerance under normal condition.
6. Convert an octet in modulo 256 into integer.

Test Step Dynamic Behaviour

Test Step Name : Check_Result
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L_0526	[NOT TCV_Res]		(F)	
2	L_0527	[TCV_Res]		(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Check_Time (maxtime: INTEGER; fn, fn1 : FN)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+FnArith(fn, fn1)			
2	L_0856	[TCV_Time <= maxtime]		(P)	
3	L_0857	[TCV_Time > maxtime]		(F)	

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		L?DL_DatInCphmCom	CphCom(CphModeCmp_01)	(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : FnArith(fn, fn1 : FN)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

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		[C_Yes]			
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		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_LupInit2_R(locup : B_2; ta : TA)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?DL_RacInChRq (TCV_Rr := DL_RacInChRq. msg.ecau_rrf, TCV_Fn := DL_RacInChRq.fn)	ChReq(ChRequest_09)		
2		ACTIVATE(OtherEventsFail_02)			To match ChReq retrans.
3		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def_R(TCV_Rr, TCV_Fn, TSPX_SDCCH8SubDef, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, ta))		
4		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(locup))		
5		ACTIVATE(OtherEventsFail)			Restore Normal default

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : MM_LupRej_R(par_rej: REJCAU; locup : B_2; ta : TA)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+MM_LupInit2_R(locup, ta)			
2		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(par_rej))	(P)	
3		+ChanRel(TCV_ch)			

Detailed Comments :

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 : NoReaction(par_int:INTEGER)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_dly(par_int)			
2		?TIMEOUT T_dly		(P)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : PageOnCarriers(CellA_Id, CellB_Id, CellC_Id, CellD_Id, CellE_Id, CellF_Id : CellID)
Group : Miscellaneous/
Objective :
Default :
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ltree_A			
2		+ltree_B			
3		+ltree_C			
4		+ltree_D			
5		+ltree_E			
6		+ltree_F			
		ltree_A			
7		[CellA_Id = C_CellA]			
8		(TCV_Null := OM_PgFill(C_CellA, PgReqTp1_01))			Start continuous paging on Cell A
9		[CellA_Id <> C_CellA]			
		ltree_B			
10		[CellB_Id = C_CellB]			
11		(TCV_Null := OM_PgFill(C_CellB, PgReqTp1_01))			Start continuous paging on Cell B
12		[CellB_Id <> C_CellB]			
		ltree_C			
13		[CellC_Id = C_CellC]			
14		(TCV_Null := OM_PgFill(C_CellC, PgReqTp1_01))			Start continuous paging on Cell C
15		[CellC_Id <> C_CellC]			
		ltree_D			
16		[CellD_Id = C_CellD]			
17		(TCV_Null := OM_PgFill(C_CellD, PgReqTp1_01))			Start continuous paging on Cell D
18		[CellD_Id <> C_CellD]			
		ltree_E			
19		[CellE_Id = C_CellE]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
20		(TCV_Null := OM_PgFill(C_CellE, PgReqTp1_01))			Start continuous paging on Cell E
21		[CellE_Id <> C_CellE]			
		ltree_F			
22		[CellF_Id = C_CellF]			
23		(TCV_Null := OM_PgFill(C_CellF, PgReqTp1_01))			Start continuous paging on Cell F
24		[CellF_Id <> C_CellF]			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Pause(Millisecs : INTEGER)
Group : Miscellaneous/
Objective :
Default :
Comments : Will wait for specified delay period. Any messages received during this period will be queued and processed later.

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_pause(Millisecs)			
2		?TIMEOUT T_pause			
3		?TIMEOUT T_guard			
4	CS_0093	(TCV_Null := OO_TguardTimeOut())		R	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : RR_hocomp1(time_fn_fn1:INTEGER; ta : TA; oldch : LOGICCH)
Group : Miscellaneous/
Objective : To finish the HO−procedure
Default : RcvHdOvAcc, OtherEvents
Comments : used var's: TCV_ch, TCV_Horf, TCV_Cntref

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!DL_DatRqPhyinfo	PhyInfo_01(TCV_ch, ta)		1.
2		L?DL_RacInHoacc(TCV_FnBurst := DL_RacInHoacc.fn)	HndOvAccRcv(TCV_ch, HandOverAcc_02(TCV_Horf))		2.
3		L?DL_EstIn	DLEstInd(TCV_ch)		
4		(TCV_Fn1:= OM_ReturnFn(TCV_ch), TCV_HoaccPara := OM_GetHoaccPara(TCV_ch))			
5		L?DL_DatInHoCom(TCV_Fn := DL_DatInHoCom.fn)	HndOvCmpRcv(TCV_ch, HandOverCmp_20)		
6		L!MDL_RelRq	MDLRelReq(oldch)		
7		+Check_Time(time_fn_fn1, TCV_Fn1, TCV_Fn)			3.
8	L_1043	[(OC_OctToInt(TCV_HoaccPara.num_bursts) >= TCV_Cntref) AND (OC_OctToInt(TCV_HoaccPara.num_bursts) <= (TCV_Cntref+20))]		(P)	4.
9	L_1044	[C_Yes]		(F)	4.

Detailed Comments : 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.
2. Match the 1st access burst. The access bursts except the 1st one are matched by the default test step RcvHdOvAcc.
3. Check HO−time.
4. Check the number of access bursts received with tolerance +3.

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 : 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_TI := TCV_Esetup1.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 : 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	L_0915	[TCV_Mt1 = TCV_Mt]		(P)	same N(SD)
2	L_0916	[TCV_Mt1 <> TCV_Mt]		(F)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Switchcell_R(old_cell, activ_cell : CellID; lac : OCTETSTRING; subch, cksn, rand : BITSTRING; arfcngsm : INTEGER; ccd: CCD; imsi : HEXSTRING)
Group : Miscellaneous/
Objective : To force the MS to reselect cells
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+Varinit_fix_R (activ_cell, lac, subch, cksn, rand, arfcngsm, ccd, imsi)			
2		(TCV_Null := OM_ChangeRFOf2Cells(activ_cell, C_E_default, old_cell, C_E_notsuitable))			

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]		I	1.

Detailed Comments : 1. svc mismatches any defined Basic Services, inclusive.

Test Step Dynamic Behaviour

Test Step Name : Varinit_fix_R (activ_cell : CellID; lac : OCTETSTRING; subch : B_2; cksn : B_3; rand : RAND; arfcngsm : 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		(TCV_chdescr_arfcn:= arfcngsm)			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Wait(par_int:INTEGER)
Group : Miscellaneous/
Objective :
Default : OtherEventsFail
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		START T_dly(par_int)			
2		?TIMEOUT T_dly			

Detailed Comments : This test step can be used , e.g. for 1s. delay of a PAGING REQUEST message following immediately a CHANNEL RELEASE.

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_cbms_R (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 : INTEGER; bcchflgsm : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP; Sub : B_2)
Group : Miscellaneous/SysInfo/
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		L!DL_UdatRqSysinfo4	SysInfo4_CBMS(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re, ChDescrp_nfhiei(TCV_Tchtype, C_S0, C_BCC, C_arfcnA))		
6		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
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		(TCV_sysinf5 := SysInf5(bcchflgsm))			

Detailed Comments :

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/SysInfo/

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_sysinf5bis := 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 : SysInfoSending_fh_R(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 : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/SysInfo/

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		L!DL_UdatRqSysinfo1	SysInfo1(bcch, cchdgsm, TCV_Max, TCV_Tx, Re)		
6		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
7		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
8		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
9		(TCV_sysinf5 := SysInf5(bcchflgsm))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_fh_pl_R(sch, bcch : LOGICCH; t, retr, neci, att, crh, mtmcgsm : INTEGER; ci : CI; t3212, mcc, mnc, lac : OCTETSTRING; co : CO; cchdgsm : CCHD; bcchflegsm1, bcchflegsm2 : NCD; bcc : BCC; ncc : NCC; nccp : NCCP; re, cba, cbq, ecsc : B_1; po : B_2; babr, cch_con, bpm, to : B_3; acc, pt : B_5; ram, cro : B_6)

Group : Miscellaneous/SysInfo/

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 for P−GSM SI2 E−GSM SI2 & SI2bis, P−GSM SI2ter
- 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), TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp), TCV_sysinf5bis := SysInf5bis(bcchflegsm2))			
2		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		
3		+ltree_sendingSI2_2bis			
4		L!DL_UdatRqSysinfo1	SysInfo1_pl(bcch, cchdgsm, re, cba, TCV_Max, TCV_Tx, acc)		
5		L!DL_UdatRqSysinfo3	SysInfo3_pl(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, '0'B, re, cba, cbq, '0'B, ecsc, TCV_Max, po, to, TCV_Tx, acc, pt, ram, cro)		
6		L!DL_UdatRqSysinfo4	SysInfo4_pl(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci, '0'B, re, cba, cbq, TCV_Max, po, to, TCV_Tx, acc, pt, ram, cro)		
		ltree_sendingSI2_2bis			
7		L!DL_UdatRqSysinfo2bis	SysInfo2bis_egsm(bcch, bcchflegsm2)		
8		L!DL_UdatRqSysinfo2	SysInfo2_pl(bcch, bcchflegsm1, nccp, re, cba, TCV_Max, TCV_Tx, acc)		
9		(TCV_sysinf5 := SysInf5(bcchflegsm1))			

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_fh_pl_R2(sch, bcch : LOGICCH; t, retr, neci, att, crh, mtmcgsm : INTEGER; ci : CI; t3212, mcc, mnc, lac : OCTETSTRING; co : CO; cchdgsm : CCHD; bcchflegsm1, bcchflegsm2 : NCD; bcchfl_2ter : NCD2; bcc : BCC; ncc : NCC; nccp : NCCP; re, cba, cbq, si2tigsm, ecsc, acs : B_1; po : B_2; babr, cch_con, bpm, to : B_3; acc, pt : B_5; ram, cro : B_6)

Group : Miscellaneous/SysInfo/

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 for P−GSM SI2 E−GSM SI2 & SI2bis, P−GSM SI2ter
- si2tigsm: SI2ter indication in SI3 for GSM
- 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), TCV_sysinf6 := SysInf6(ci, mcc, mnc, lac, co, nccp), TCV_sysinf5bis := SysInf5bis(bcchflegsm2), TCV_sysinf5ter := SysInf5ter(bcchfl_2ter))			
2		L!DL_UdatRqSchinfo	SyncInfo(sch, bcc, ncc)		
3		+ltree_sendingSI2_2bis			
4		+ltree_sendingSI2ter			
5		+ltree_sendingSI7_8			
6		L!DL_UdatRqSysinfo1	SysInfo1_pl(bcch, cchdgsm, re, cba, TCV_Max, TCV_Tx, acc)		
7		L!DL_UdatRqSysinfo3	SysInfo3_pl(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, '0'B, re, cba, cbq, '1'B, ecsc, TCV_Max, po, to, TCV_Tx, acc, pt, ram, cro)		
8		L!DL_UdatRqSysinfo4	SysInfo4_pl_R(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci, '1'B, re, cba, TCV_Max, TCV_Tx, acc, ram)		
		ltree_sendingSI2_2bis			
9		L!DL_UdatRqSysinfo2bis	SysInfo2bis_egsm(bcch, bcchflegsm2)		
10		L!DL_UdatRqSysinfo2	SysInfo2_pl(bcch, bcchflegsm1, nccp, re, cba, TCV_Max, TCV_Tx, acc)		
11		(TCV_sysinf5 := SysInf5(bcchflegsm1))			
		ltree_sendingSI2ter			
12		L!DL_UdatRqSysinfo2ter	SysInfo2ter(bcch, bcchfl_2ter)		

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
		ltree_sendingSI7_8			
13		L!DL_UdatRqSysinfo7	SysInfo7(bcch, cbq, po, to, pt , cro)		
14		L!DL_UdatRqSysinfo8	SysInfo8(bcch, cbq, po, to, pt , cro)		
Detailed Comments :					

Test Step Dynamic Behaviour

Test Step Name : SysInfoSending_nfh_R (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 : INTEGER; bcchflgsm, bcflaltgsm : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/SysInfo/

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		L!DL_UdatRqSysinfo4	SysInfo4(bcch, mcc, mnc, lac, crh, mtmcgsm, TCV_Neci , TCV_Max, TCV_Tx, Re)		
6		L!DL_UdatRqSysinfo1_nh	SysInfo1_nh(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
7		L!DL_UdatRqSysinfo3	SysInfo3(bcch, ci, mcc, mnc, lac, TCV_Ccd0H, co, crh, mtmcgsm, TCV_Neci, TCV_Max, TCV_Tx, Re)		
8		[NOT TSPX_AltNb]			3.
9		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcchflgsm, TCV_Max, TCV_Tx, Re, nccp)		
10		(TCV_sysinf5 := SysInf5(bcchflgsm))			
11		[TSPX_AltNb]			4.
12		L!DL_UdatRqSysinfo2	SysInfo2(bcch, bcflaltgsm, TCV_Max, TCV_Tx, Re, nccp)		
13		(TCV_sysinf5 := SysInf5(bcflaltgsm))			

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_r (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, bcchflrgsm : NCD; Re : B_1; bcc : BCC; ncc : NCC; nccp : NCCP)

Group : Miscellaneous/SysInfo/

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, bcchflrgsm)		
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_sysinf5bis := SysInf5bis(bcchflrgsm))			
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/SysInfo/

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		[ch = "dummy"]		I	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : SysInfo_5bisSending(ch : LOGICCH; sysinfo5bis_pdu : SYSINFO5bis_PDU)

Group : Miscellaneous/SysInfo/

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		[ch = "dummy"]		I	

Detailed Comments :

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 : 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		CANCEL		R	1.
4		?TIMEOUT T_release			
5		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		?TIMEOUT T_release		(F)	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : BasicServiceMO(svc : SERVICES; rate : RATE)

Group : Preambles/

Objective : To generate a Bcap, an MO SETUP 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.

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_ChMod.mode := C_ChMod_r, TCV_Esetup := Setup_04(Bcap_Speech_MO(TCV_Rchr)), TCV_Ecall := C_Yes)			Emergency Call
4		[TCV_Service = C_Telephony]			
5		(TCV_ChMod.mode := C_ChMod_r, TCV_Setup_mo := Setup_02(Bcap_Speech_MO(TCV_Rchr)))			Speech
6		[TCV_Service = C_AltSpchFax]			
7		[rate = C_Full]			
8		+TS61MO(C_AltSpchFax)			Speech/fax
9		[rate <> C_Full]		I	IXIT error
10		[TCV_Service = C_AutoFax]			
11		[rate = C_Full]			
12		+TS62MO(C_AutoFax)			Auto fax
13		[rate <> C_Full]		I	IXIT error
14		[TCV_Service = C_Async300]			
15		(TCV_MODEM := V21orAbt1)			
16		+BS2xMO(C_Async300, TSPX_BS_21_sacp, TSPX_BS_21_T_NT, TSPX_BS_21_itc1, C_300bs, TCV_MODEM.V22bisOrV26ter, TSPX_BS_21_ce)			BS21
17		[TCV_Service = C_Async1200]			
18		(TCV_MODEM := V22orAbt1)			
19		+BS2xMO(C_Async1200, TSPX_BS_22_sacp, TSPX_BS_22_T_NT, TSPX_BS_22_itc1, C_1200bs, TCV_MODEM.V22bisOrV26ter, TSPX_BS_22_ce)			BS22
20		[TCV_Service = C_Async120075]			
21		(TCV_MODEM := V23orAbt1)			
22		+BS2xMO(C_Async120075, TSPX_BS_23_sacp, TSPX_BS_23_T_NT, TSPX_BS_23_itc, C_120075bs, TCV_MODEM.V22bisOrV26ter, TSPX_BS_23_ce)			BS23
23		[TCV_Service = C_Async2400]			
24		(TCV_MODEM := V22bisOrV26terOrAbt1)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
25		+BS2xMO(C_Async2400, TSPX_BS_24_sacp, TSPX_BS_24_T_NT, TSPX_BS_24_itc1, C_2400bs, TCV_MODEM.V22bisOrV26ter, TSPX_BS_24_ce)			BS24
26		[TCV_Service = C_Async4800]			
27		(TCV_MODEM := V32orAbt1)			
28		+BS2xMO(C_Async4800, TSPX_BS_25_sacp, TSPX_BS_25_T_NT, TSPX_BS_25_itc1, C_4800bs, TCV_MODEM.V22bisOrV26ter, TSPX_BS_25_ce)			BS25
29		[TCV_Service = C_Async9600]			
30		(TCV_MODEM := V32orAbt1)			
31		+BS2xMO(C_Async9600, TSPX_BS_26_sacp, TSPX_BS_26_T_NT, TSPX_BS_26_itc1, C_9600bs, TCV_MODEM.V22bisOrV26ter, TSPX_BS_26_ce)			BS26
32		[TCV_Service = C_Sync1200]			
33		+BS3xMO(C_Sync1200, FALSE, TSPX_BS_31_itc1, TSPX_BS_31_sacp1, C_1200bs, C_transparent, C_modemt_V22)			BS31
34		[TCV_Service = C_Sync2400]			
35		+BS3xMO(C_Sync2400, TSPX_BS_32_X32_T_NT, TSPX_BS_32_itc1, TSPX_BS_32_sacp1, C_2400bs, TSPX_BS_32_X32_ce, TCV_MODEM.V22bisOrV26ter)			BS32
36		[TCV_Service = C_Sync4800]			
37		+BS3xMO(C_Sync4800, TSPX_BS_33_X32_T_NT, TSPX_BS_33_itc1, TSPX_BS_33_sacp1, C_4800bs, TSPX_BS_33_X32_ce, C_modemt_V32)			BS33
38		[TCV_Service = C_Sync9600]			
39		+BS3xMO(C_Sync9600, TSPX_BS_34_X32_T_NT, TSPX_BS_34_itc1, TSPX_BS_34_sacp1, C_9600bs, TSPX_BS_34_X32_ce, C_modemt_V32)			BS34
40		[TCV_Service = C_PAD300]			
41		+BS4xMO(C_PAD300, C_300bs, TSPX_BS_41_ce, TSPX_BS_41_T_NT)			BS41
42		[TCV_Service = C_PAD1200]			
43		+BS4xMO(C_PAD1200, C_1200bs, TSPX_BS_42_ce, TSPX_BS_42_T_NT)			BS42
44		[TCV_Service = C_PAD120075]			
45		+BS4xMO(C_PAD120075, C_120075bs, TSPX_BS_43_ce, TSPX_BS_43_T_NT)			BS43
46		[TCV_Service = C_PAD2400]			
47		+BS4xMO(C_PAD2400, C_2400bs, TSPX_BS_44_ce, TSPX_BS_44_T_NT)			BS44
48		[TCV_Service = C_PAD4800]			
49		+BS4xMO(C_PAD4800, C_4800bs, TSPX_BS_45_ce, TSPX_BS_45_T_NT)			BS45

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
50		[TCV_Service = C_PAD9600]			
51		+BS4xMO(C_PAD9600, C_9600bs, TSPX_BS_46_ce, TSPX_BS_46_T_NT)			BS46
52		[TCV_Service = C_Packet2400]			
53		+BS5xMO(C_Packet2400, C_2400bs)			BS51
54		[TCV_Service = C_Packet4800]			
55		+BS5xMO(C_Packet4800, C_4800bs)			BS52
56		[TCV_Service = C_Packet9600]			
57		+BS5xMO(C_Packet9600, C_9600bs)			BS53
58		[TCV_Service = C_AltSpchData]			
59		+BS61or81MO(C_AltSpchData, TSPX_BS_61_A, TSPX_BS_61_A_ur1_T_NT, TSPX_BS_61_A_ur1, TSPX_BS_61_A_ur1_ce, TSPX_BS_61_S, TSPX_BS_61_S_ur1, C_RI_alternate)			BS61
60		[TCV_Service = C_SpchData]			
61		+BS61or81MO(C_SpchData, TSPX_BS_81_A, TSPX_BS_81_A_ur1_T_NT, TSPX_BS_81_A_ur1, TSPX_BS_81_A_ur1_ce, TSPX_BS_81_S, TSPX_BS_81_S_ur1, C_RI_follow)			BS81
62		[C_Yes] local_init		I	IXIT error
63		(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 := V22bisOrV26terOrAbt1)			
64		[TSPC_DualRate AND (TSPC_HalfRateData AND (TCV_Service <> C_EmgCall) AND (TCV_Service <> C_Telephony) OR TSPC_HalfRateSpeech AND ((TCV_Service = C_EmgCall) OR (TCV_Service = C_Telephony)))]			
65		(TCV_Rchr := C_Rchr_DualForDualH)			
66		[C_Yes]			

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 : BasicServiceMT(svc : SERVICES; rate : RATE)

Group : Preambles/

Objective : To generate a Bcap, an MT SETUP 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 MT call.
To generate a second SETUP message for TC_11_1_1.

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_2ndtest := C_No)			Default values
2		+SvcSupportedChk(svc)			
3		+local_main			
		local_main			
4		[TCV_Service = C_Telephony]			
5		(TCV_ChMod.mode := C_ChMod_r, TCV_Setup_mt := Setup_01(TI_02))			Speech
6		+local_IFsetup_Telephony			Config speech
7		[TCV_Service = C_AltSpchFax]			
8		+TS61MT(C_AltSpchFax)			Speech/fax
9		[TCV_Service = C_AutoFax]			
10		+TS62MT(C_AutoFax)			Auto fax
11		[TCV_Service = C_Async300]			
12		+BS2xMT(C_Async300, TSPX_BS_21_more_itc, TSPX_BS_21_T_NT, TSPX_BS_21_itc1, TSPX_BS_21_itc2, C_300bs, C_modemt_V21, TSPX_BS_21_ce)			BS21
13		[TCV_Service = C_Async1200]			
14		+BS2xMT(C_Async1200, TSPX_BS_22_more_itc, TSPX_BS_22_T_NT, TSPX_BS_22_itc1, TSPX_BS_22_itc2, C_1200bs, C_modemt_V22, TSPX_BS_22_ce)			BS22
15		[TCV_Service = C_Async2400]			
16		+BS2xMT(C_Async2400, TSPX_BS_24_more_itc, TSPX_BS_24_T_NT, TSPX_BS_24_itc1, TSPX_BS_24_itc2, C_2400bs, C_modemt_V22bis, TSPX_BS_24_ce)			BS24
17		[TCV_Service = C_Async4800]			
18		+BS2xMT(C_Async4800, TSPX_BS_25_more_itc, TSPX_BS_25_T_NT, TSPX_BS_25_itc1, TSPX_BS_25_itc2, C_4800bs, C_modemt_V32, TSPX_BS_25_ce)			BS25
19		[TCV_Service = C_Async9600]			
20		+BS2xMT(C_Async9600, TSPX_BS_26_more_itc, TSPX_BS_26_T_NT, TSPX_BS_26_itc1, TSPX_BS_26_itc2, C_9600bs, C_modemt_V32, TSPX_BS_26_ce)			BS26
21		[TCV_Service = C_Sync1200]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
22		+BS3xMT(C_Sync1200, TSPX_BS_31_more_itc, C_No, TSPX_BS_31_more_sacp, TSPX_BS_31_itc1, TSPX_BS_31_itc2, TSPX_BS_31_sacp1, TSPX_BS_31_sacp2, C_1200bs, C_transparent)			BS31
23		[TCV_Service = C_Sync2400]			
24		+BS3xMT(C_Sync2400, TSPX_BS_32_more_itc, TSPX_BS_32_X32_T_NT, TSPX_BS_32_more_sacp, TSPX_BS_32_itc1, TSPX_BS_32_itc2, TSPX_BS_32_sacp1, TSPX_BS_32_sacp2, C_2400bs, TSPX_BS_32_X32_ce)			BS32
25		[TCV_Service = C_Sync4800]			
26		+BS3xMT(C_Sync4800, TSPX_BS_33_more_itc, TSPX_BS_33_X32_T_NT, TSPX_BS_33_more_sacp, TSPX_BS_33_itc1, TSPX_BS_33_itc2, TSPX_BS_33_sacp1, TSPX_BS_33_sacp2, C_4800bs, TSPX_BS_33_X32_ce)			BS33
27		[TCV_Service = C_Sync9600]			
28		+BS3xMT(C_Sync9600, TSPX_BS_34_more_itc, TSPX_BS_34_X32_T_NT, TSPX_BS_34_more_sacp, TSPX_BS_34_itc1, TSPX_BS_34_itc2, TSPX_BS_34_sacp1, TSPX_BS_34_sacp2, C_9600bs, TSPX_BS_34_X32_ce)			BS34
29		[TCV_Service = C_AltSpchData]			
30		+BS61or81MT(C_AltSpchData, TSPX_BS_61_A, TSPX_BS_61_A_ur1_T_NT, TSPX_BS_61_A_ur2_T_NT, TSPX_BS_61_A_more_ur, TSPX_BS_61_S, TSPX_BS_61_S_more_ur, TSPX_BS_61_A_ur1, TSPX_BS_61_A_ur2, TSPX_BS_61_S_ur1, TSPX_BS_61_S_ur2, C_RI_alternate, TSPX_BS_61_A_ur1_ce)			BS61
31		[TCV_Service = C_SpchData]			
32		+BS61or81MT(C_SpchData, TSPX_BS_81_A, TSPX_BS_81_A_ur1_T_NT, TSPX_BS_81_A_ur2_T_NT, TSPX_BS_81_A_more_ur, TSPX_BS_81_S, TSPX_BS_81_S_more_ur, TSPX_BS_81_A_ur1, TSPX_BS_81_A_ur2, TSPX_BS_81_S_ur1, TSPX_BS_81_S_ur2, C_RI_follow, TSPX_BS_81_A_ur1_ce)			BS81

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
33		[(TCV_Service <> C_Telephony) AND (TCV_Service <> C_AltSpchFax) AND (TCV_Service <> C_AutoFax) AND (TCV_Service <> C_Async300) AND (TCV_Service <> C_Async1200) AND (TCV_Service <> C_Async2400) AND (TCV_Service <> C_Async4800) AND (TCV_Service <> C_Async9600) AND (TCV_Service <> C_Sync1200) AND (TCV_Service <> C_Sync2400) AND (TCV_Service <> C_Sync4800) AND (TCV_Service <> C_Sync9600) AND (TCV_Service <> C_AltSpchData) AND (TCV_Service <> C_SpchData)]		I	
		local_IFsetup_Telephony			
34		[TCV_supported]			
35		(TCV_Null := OO_IFsetup_Telephony())			
36		[NOT TCV_supported]			
37		[TCV_SpecialCase]			
38		START T_dly(C_T_Wait)			
39		?TIMEOUT T_dly			
40		[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 : BS2xMO(srv : SERVICES; sacp: B_3; both : BOOLEAN; itc1 : B_3; ur : B_4; modem : B_5; ce : B_2)

Group : Preambles/

Objective : To generate an MO SETUP message with appropriate Bcap and LLC for BS2x service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_ur := ur, TCV_CE := NT_BothT_BothNTorT_BothT_BothNT, TCV_ir := C_ir_16kbs)			
2		+local_pixitvaluechk			validate IXIT values
3		+local_selectCe			select a connection element
4		+local_bcap			generate BC
5		+local_setupmsg			generate SETUP
6		+Chmod(TCV_ChRate, TCV_ce, ur, srv)			initiate channel mode variables
7		(TCV_Null := OO_IFsetup_BS2xorBS3x_MO(TCV_srv, itc1, sacp, TCV_ce, TCV_ChRate))			ask test operator to configure the MS
		local_bcap			
8		[NOT both]			
9		[TCV_ce = C_notransparent]			
10		[itc1 = C_UDI]			
11		+local_BS2x_UDI_NT(C_notransparent, sacp)			
12		[itc1 = C_3100Hz]			
13		+local_BS2x_3100_NT(C_notransparent, sacp)			
14		[TCV_ce = C_transparent]			CE:T only MS
15		[itc1 = C_UDI]			
16		+local_BS2x_UDI_T(sacp)			
17		[itc1 = C_3100Hz]			
18		+local_BS2x_3100_T(sacp)			
19		[both]			
20		[TCV_ce = C_transparent]			
21		[itc1 = C_UDI]			
22		+local_BS2x_UDI_Tboth(TCV_CE.TorBothTorBothNT,sacp)			
23		[itc1 = C_3100Hz]			3.1kHz
24		+local_BS2x_3100_Tboth(TCV_CE.TorBothTorBothNT, sacp)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
25		[(TCV_ce = C_notransparent)]			
26		[itc1 = C_UDI]			
27		+local_BS2x_UDI_NT(TCV_CE.NTorBothTor BothNT,sacp)			
28		[itc1 = C_3100Hz]			3.1kHz
29		+local_BS2x_3100_NT(TCV_CE.NTorBothTo rBothNT, sacp)			
		local_BS2x_3100_NT(cel: B_2; sacp : B_3)			
30		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, C_SDUintegrity, C_nirr_dontcare, C_no_rate_adaption, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, cel, modem, C_uil2p_dontcare))			11.8.2.1 .1.2
		local_BS2x_UDI_NT(cel: B_2; sacp : B_3)			
31		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_SDUintegrity, C_nirr_dontcare, C_rate_adaption_V110, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, cel, C_modemt_none, C_uil2p_dontcare))			11.8.2.1 .1.4
		local_BS2x_3100_T(sacp : B_3)			
32		[ur = C_9600bs]			
33		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, C_transparent, modem))			11.8.2.1 .1.1
34		[(ur = C_300bs) OR (ur = C_1200bs) OR (ur = C_120075bs) OR (ur = C_2400bs) OR (ur = C_4800bs)]			
35		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, modem), TCV_ir := C_ir_8kbs)			11.8.2.1 .1.1
		local_BS2x_UDI_T(sacp : B_3)			
36		[ur = C_9600bs]			
37		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, C_transparent, C_modemt_none))			11.8.2.1 .1.3
38		[(ur = C_300bs) OR (ur = C_1200bs) OR (ur = C_120075bs) OR (ur = C_2400bs) OR (ur = C_4800bs)]			
39		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, C_modemt_none), TCV_ir := C_ir_8kbs)			11.8.2.1 .1.3

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
		local_BS2x_3100_Tboth(cel: B_2; sacp : B_3)			
40		[ur <> C_9600bs]			
41		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, C_struc_dontchk, C_nirr_dontcare, C_no_rate_adaption, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8or16kbs, C_parity_dontcare, cel, modem, C_uil2p_dontcare))			11.8.2.1 .1.1 conbinrd with 11.8.2.1 .1.2 , CE and structure without exact match
42		[ur = C_9600bs]			
43		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, C_struc_dontchk, C_nirr_nomeaning, C_no_rate_adaption, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8or16kbs, C_parity_dontcare, cel, modem, C_uil2p_dontcare))			11.8.2.1 .1.1 conbinrd with 11.8.2.1 .1.2 , CE and structure without exact match
		local_BS2x_UDI_Tboth(cel: B_2; sacp : B_3)			
44		[ur = C_9600bs]			
45		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_struc_dontchk, C_nirr_nomeaning, C_rate_adaption_V110, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8or16kbs, C_parity_dontcare, cel, C_modemt_none, C_uil2p_dontcare))			11.8.2.1 .1.3 conbine d with 11.8.2.1 .1.4, CE and structure without exact match
46		[ur <> C_9600bs]			
47		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_struc_dontchk, C_nirr_dontcare, C_rate_adaption_V110, sacp, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8or16kbs, C_parity_dontcare, cel, C_modemt_none, C_uil2p_dontcare))			11.8.2.1 .1.3 conbine d with 11.8.2.1 .1.4, CE and structure without exact match
		local_setupmsg			
48		[itc1 = C_UDI]			
49		(TCV_Setup_mo := Setup_10_UDI(TCV_BcapMO1, Llcmp_BS2xOrBS4x_UDI, Hlcmp_AnyOrOmit))			
50		[itc1 = C_3100Hz]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
51		(TCV_Setup_mo := Setup_10_3100(TCV_BcapMO1, Llcmp_BS2xOrBS61aOrBS81a_3100, Hlcmp_AnyOrOmit))			
		local_selectCe			
52		[both]			
53		[(ur = C_300bs) OR (ur = C_120075bs) OR (ur = C_4800bs)]			
54		(TCV_ce := C_notransparent)			
55		[(ur = C_1200bs) OR (ur = C_2400bs) OR (ur = C_9600bs)]			
56		(TCV_ce := C_transparent)			
57		[NOT both]			
58		(TCV_ce := ce)			
		local_pixitvaluechk			
59		[(ce <> C_transparent) AND (ce <> C_notransparent) AND (NOT both)]		I	IXIT error
60		[(itc1 <> C_UDI) AND (itc1 <> C_3100Hz)]		I	IXIT error
61		[(sacp <> C_I440_450) AND (sacp <> C_X28_nond)]		I	IXIT error
62		[C_Yes]			IXIT values OK

Detailed Comments : The BS2x Bearer Capabilities and SETUP messages for MO is constructed according to section 11.8.2.1.1.1, section 11.8.2.1.1.2, section 11.8.2.1.1.3 and section 11.8.2.1.1.4 of GSM 11.10 and section B.1.2.1, section B.1.2.2, section B.1.2.1 and section B.2.2.2 of GSM 07.01.

The algorithm for selecting parameters:
IF (TSPX_BS_2x_T_NT)
    TCV_ce := C_notransparent
ELSE
    TCV_ce := TSPX_BS_2x_ce

use (TCV_ce, TSPX_BS_2x_itc1, TSPX_BS_2x_sacp1, svc) for OO_IFsetup_BS2xorBS3x_MO

Test Step Dynamic Behaviour

Test Step Name : BS2xMT(srv : SERVICES; more, both : BOOLEAN; itc1, itc2 : B_3; ur : B_4; modem : B_5; ce : B_2)

Group : Preambles/

Objective : To generate an MT SETUP message with appropriate Bcap for BS2x service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_main			
2		+local_IFsetup			
		local_main			
3		(TCV_srv := srv, TCV_itc := itc1)			
4		+local_bcap1			
5		(TCV_Setup_mt := Setup_20(TCV_BcapMT1, TI_02))			1st SETUP
6		+Chmod(TCV_ChRate, TCV_ce, ur, srv)			Ini ch mode
7		+local_bcap2			
8		(TCV_Setup_mt1:= Setup_20(TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes)			2nd SETUP
		local_bcap1			
9		[itc1 = C_3100Hz]			3.1kHz
10		[both OR (ce = C_notransparent)]			
11		+local_BS2x_3100_NT(C_notransparent)			
12		[NOT both AND (ce = C_transparent)]			
13		+local_BS2x_3100_T			
14		[itc1 = C_UDI]			UDI
15		[both OR (ce = C_notransparent)]			
16		+local_BS2x_UDI_NT(C_notransparent)			
17		[NOT both AND (ce = C_transparent)]			
18		+local_BS2x_UDI_T			
19		[(itc1 <> C_3100Hz) AND (itc1 <> C_UDI)]		I	
		local_bcap2			
20		+local_itc			
21		+local_bcap			
22		(TCV_BcapMT2 := TCV_BcapMT1)			
		local_itc			
23		(TCV_itc2 := itc1)			
24		[more]			3.1 kHz
25		(TCV_itc2 := itc2)			
26		[NOT more]			
		local_bcap			
27		[both AND (TCV_itc2 = C_3100Hz)]			
28		+local_BS2x_3100_T			
29		[both AND (TCV_itc2 = C_UDI)]			UDI
30		+local_BS2x_UDI_T			
31		[NOT both AND (TCV_itc2 = C_3100Hz)]			
32		+local_BS2x_3100_NT(C_BothT)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
33		[NOT both AND (TCV_itc2 = C_UDI)]			UDI
34		+local_BS2x_UDI_NT(C_BothT)			
35	L_0943	[(TCV_itc2 <> C_3100Hz) AND (TCV_itc2 <> C_UDI)]		I	
		local_BS2x_3100_NT(cel: B_2)			
36		[TSPX_TE_FLCT = C_Inband]			
37		(TCV_BcapMT1 := BcapX_MT(C_3100Hz, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, C_modemt_abt1, C_ISO6429), TCV_ce := cel)			11.8.1.1.1.2
38		[TSPX_TE_FLCT = C_Nocontrol]			
39		(TCV_BcapMT1 := BcapX_MT(C_3100Hz, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, C_modemt_abt1, C_COPnoFLCT), TCV_ce := cel)			11.8.1.1.1.2
40	L_0944	[(TSPX_TE_FLCT = C_Outband) OR ((TSPX_TE_FLCT <> C_Inband) AND (TSPX_TE_FLCT <> C_Nocontrol))]		I	
		local_BS2x_UDI_NT(cel: B_2)			
41		[TSPX_TE_FLCT = C_Outband]			
42		(TCV_BcapMT1 := Bcap_MT(C_UDI, C_SDUintegrity, C_nirr_nomeaning, C_rate_adaption_V110, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, C_modemt_none), TCV_ce := cel)			11.8.1.1.1.4
43		[TSPX_TE_FLCT = C_Inband]			
44		(TCV_BcapMT1 := BcapX_MT(C_UDI, C_SDUintegrity, C_nirr_nomeaning, C_rate_adaption_V110, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, C_modemt_none, C_ISO6429), TCV_ce := cel)			11.8.1.1.1.4
45		[TSPX_TE_FLCT = C_Nocontrol]			
46		(TCV_BcapMT1 := BcapX_MT(C_UDI, C_SDUintegrity, C_nirr_nomeaning, C_rate_adaption_V110, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, C_modemt_none, C_COPnoFLCT), TCV_ce := cel)			11.8.1.1.1.4
47		[(TSPX_TE_FLCT <> C_Outband) AND (TSPX_TE_FLCT <> C_Inband) AND (TSPX_TE_FLCT <> C_Nocontrol)]		I	
		local_BS2x_3100_T			
48		[ur = C_9600bs]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
49		(TCV_BcapMT1 := Bcap_MT(C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, C_transparent, modem), TCV_itc2 := C_3100Hz, TCV_ce := C_transparent)			11.8.1.1 .1.1
50		[ur <> C_9600bs]			
51		(TCV_BcapMT1 := Bcap_MT(C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_8kbs, TSPX_TE_parity, C_transparent, modem), TCV_itc2 := C_3100Hz, TCV_ce := C_transparent)			11.8.1.1 .1.1
		local_BS2x_UDI_T			
52		[ur = C_9600bs]			
53		(TCV_BcapMT1 := Bcap_MT(C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, C_transparent, C_modemt_none), TCV_itc2 := C_UDI, TCV_ce := C_transparent)			11.8.1.1 .1.3
54		[ur <> C_9600bs]			
55		(TCV_BcapMT1 := Bcap_MT(C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_8kbs, TSPX_TE_parity, C_transparent, C_modemt_none), TCV_itc2 := C_UDI, TCV_ce := C_transparent)			11.8.1.1 .1.3
		local_IFsetup			
56		[TCV_supported]			
57		(TCV_Null := OO_IFsetup_BS2x_MT(TCV_srv, TCV_itc))			
58		[NOT TCV_supported]			
59		[TCV_SpecialCase]			
60		START T_dly(C_T_Wait)			
61		?TIMEOUT T_dly			
62		[NOT TCV_SpecialCase]		I	Paramet er error

Detailed Comments : The algorithm for derivation of BS2x Bearer Capabilities :

The Bearer Capability for 1st SETUP (used for an MT call):
itc= TSPX_BS_2x_itc1
IF itc=3.1kHz
  IF TSPX_BS_2x_T_MT OR (NOT TSPX_BS_2x_T_MT AND (TSPX_BS_2x_ce = NT))
    use coding of sec. 11.8.1.1.1.2 with (ce=NT)
  ELSE IF NOT TSPX_BS_2x_T_MT AND (TSPX_BS_2x_ce = T)
        use coding of sec. 11.8.1.1.1.1 with (ce=T)
ELSE IF itc=udi

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

IF TSPX_BS_2x_T_MT OR (NOT TSPX_BS_2x_T_MT AND (TSPX_BS_2x_ce = NT))
use coding of sec. 11.8.1.1.1.4 with (ce=NT)
ELSE IF NOT TSPX_BS_2x_T_MT AND (TSPX_BS_2x_ce = T)
use coding of sec. 11.8.1.1.1.3 with (ce=T)

The derived 2nd Bearer Capability for 2nd SETUP: (used for TC_11_1_1)
IF TSPX_BS_2x_more_itc
itc=TSPX_BS_2x_itc2
ELSE
itc= TSPX_BS_2x_itc1
IF TSPX_BS_2x_T_NT
IF itc=3.1kHz use coding of sec. 11.8.1.1.1.1 with (ce=T)
IF itc=udi use coding of sec. 11.8.1.1.1.3 with (ce=T)
ELSE
IF itc=3.1Khz use coding of sec. 11.8.1.1.1.2 with (ce=both/T)
IF itc=udi use coding of sec. 11.8.1.1.1.4 with (ce=both/T)

Test Step Dynamic Behaviour

Test Step Name : BS3xMO(srv : SERVICES; both : BOOLEAN; itc1, sacp1 : B_3; ur : B_4; ce : B_2; modem : B_5)
Group : Preambles/
Objective : To generate an MO SETUP message with appropriate Bcap and LLC for BS3x service.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_ur := ur, TCV_CE := NT_BothT_BothNTorT_BothT_BothNT, TCV_ir := C_ir_16kbs, TCV_uil1p := C_no_rate_adaption)			
2		+local_pixitvaluechk			2.
3		+local_selectCe			select a CE
4		+local_bcap			generate BC
5		+local_setupmsg			3.
6		+Chmod(TCV_ChRate, TCV_ce, ur, srv)			initiate channel mode variables
7		(TCV_Null := OO_IFsetup_BS2xorBS3x_MO(TCV_srv, itc1, sacp1, TCV_ce, TCV_ChRate))			ask test operator to configure the MS
		local_bcap			
8		[NOT both]			
9		[TCV_ce = C_notransparent]			
10		[itc1 = C_UDI]			
11		[sacp1 = C_X32]			
12		+local_BS3x_UDI_X32_NT			
13		[sacp1 <> C_X32]		I	5.
14		[itc1 = C_3100Hz]			
15		[sacp1 <> C_X32]		I	6.
16		[sacp1 = C_X32]			
17		+local_BS3x_3100_X32_NT(C_notransparent)			
18		[TCV_ce = C_transparent]			CE:T only MS
19		[itc1 = C_UDI]			
20		[sacp1 = C_X32]		I	7.
21		[sacp1<> C_X32]			
22		+local_BS3x_UDI_T(sacp1)			
23		[itc1 = C_3100Hz]			
24		[sacp1 = C_X21]		I	8.
25		[sacp1 = C_I440_450]			
26		+local_BS3x_3100_T			
27		[sacp1 = C_X32]			
28		+local_BS3x_3100_X32_T			
29		[both]			
30		[TCV_ce = C_transparent]			
31		[itc1 = C_UDI]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
32		[sacp1 = C_X32]		I	7.
33		[sacp1<> C_X32]			
34		+local_BS3x_UDI_T(sacp1)			
35		[itc1 = C_3100Hz]			3.1kHz
36		[sacp1 = C_X21]		I	8.
37		[sacp1 = C_I440_450]			
38		+local_BS3x_3100_T			
39		[sacp1 = C_X32]			
40		+local_BS3x_3100_X32_NT(TCV_CE.TorBothTorBothNT)			
41		[(TCV_ce = C_notransparent)]			
42		[itc1 = C_UDI]			
43		[sacp1 = C_X32]			
44		+local_BS3x_UDI_X32_NT			
45		[sacp1 <> C_X32]		I	5.
46		[itc1 = C_3100Hz]			3.1kHz
47		[sacp1 <> C_X32]		I	6.
48		[sacp1 = C_X32]			
49		+local_BS3x_3100_X32_Tboth(TCV_CE.NTorBothTorBothNT)			
		local_setupmsg			
50		[itc1 = C_UDI]			
51		(TCV_Setup_mo := Setup_10_UDI(TCV_BcapMO1, Llcmp_BS3x_UDI, Hlcmp_AnyOrOmit))			
52		[itc1 = C_3100Hz]			
53		(TCV_Setup_mo := Setup_10_3100(TCV_BcapMO1, Llcmp_BS3xOrBS61sOrBS81s_3100, Hlcmp_AnyOrOmit))			
		local_BS3x_3100_T			
54		[ur = C_9600bs]			
55		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, C_transparent, modem))			11.8.2.2 .1.1
56		[ur <> C_9600bs]			
57		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_8kbs, C_parity_NA, C_transparent, modem))			11.8.2.2 .1.1
		local_BS3x_UDI_T(sacp : B_3)			
58		[ur = C_9600bs]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
59		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, sacp, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, C_transparent, C_modemt_none), TCV_uil1p := C_rate_adaption_V110)			11.8.2.2 .1.2
60		[NOT (ur = C_9600bs)]			
61		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, sacp, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_8kbs, C_parity_NA, C_transparent, C_modemt_none), TCV_ir := C_ir_8kbs, TCV_uil1p := C_rate_adaption_V110)			11.8.2.2 .1.2
		local_BS3x_3100_X32_T			
62		[ur = C_9600bs]			
63		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, C_transparent, modem))			11.8.2.2 .1.3
64		[ur <> C_9600bs]			
65		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_8kbs, C_parity_NA, C_transparent, modem))			11.8.2.2 .1.3
		local_BS3x_3100_X32_NT(ce2 : B_2)			
66		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, C_SDUintegrity, C_nirr_dontcare, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, ce2, modem, C_X25_flct))			11.8.2.2 .1.4
		local_BS3x_UDI_X32_NT			
67		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_SDUintegrity, C_nirr_dontcare, C_rate_adaption_X31, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, C_notransparent, C_modemt_none, C_X25_flct), TCV_uil1p := C_rate_adaption_X31)			11.8.2.2 .1.5
		local_BS3x_3100_X32_Tboth(ce2 : B_2)			
68		[ur = C_9600bs]			
69		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, C_struc_dontchk, C_nirr_nomeaning, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, ce2, modem, C_X25_flct))			11.8.2.2 .1.3 combined with 11.8.2.2 .1.4, CE and structure without exact match
70		[ur <> C_9600bs]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
71		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, C_struc_dontchk, C_nirr_dontcare, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_8or16kbs, C_parity_NA, ce2, modem, C_X25_flct))			11.8.2.2.1.3 combined with 11.8.2.2.1.4, CE, IR and structure without exact match
		local_selectCe			
72		[both]			
73		[(ur = C_300bs) OR (ur = C_120075bs) OR (ur = C_4800bs)]			
74		(TCV_ce := C_transparent)			
75		[(ur = C_1200bs) OR (ur = C_2400bs) OR (ur = C_9600bs)]			
76		(TCV_ce := C_notransparent)			
77		[NOT both]			
78		(TCV_ce := ce)			
		local_pixitvaluechk			
79		[(ce <> C_transparent) AND (ce <> C_notransparent) AND (NOT both)]		I	IXIT error
80		[(itc1 <> C_UDI) AND (itc1 <> C_3100Hz)]		I	IXIT error
81		[(sacp1 <> C_I440_450) AND (sacp1 <> C_X21) AND (sacp1 <>C_X32)]		I	IXIT error
82		[(ur = C_1200bs) AND (sacp1 = C_X32)]		I	4.
83		[C_Yes]			IXIT values OK, but combination not checked yet

Detailed Comments :

1. The BS3x Bearer Capabilities and SETUP messages for MO is constructed according to section 11.8.2.2.1.1, section 11.8.2.2.1.2, section 11.8.2.2.1.3, section 11.8.2.2.1.4 and section 11.8.2.2.1.5 of GSM 11.10 and section B.1.3.1.1, section B.1.3.1.2, section B.1.3.2.1, section B.1.3.2.2, section B.2.3.1 and section B.2.3.2 of GSM 07.01.
2. validate IXIT values.
3. generate SETUP.
4. IXIT error, BS31 does not support X32.
5. IXIT error, UDI non−X32 case has no non transparent mode.
6. IXIT error, 3.1kHz non−X32 case has no non transparent mode.
7. IXIT error, UDI X32 case has no transparent mode.
8. IXIT error, 3.1kHz case has no X21 signalling access protocol.

The algorithm for selecting parameters:

```
IF (TSPX_BS_3x_T_NT)
        TCV_ce := C_notransparent
ELSE
        TCV_ce := TSPX_BS_3x_ce
```

use (TCV_ce, TSPX_BS_3x_itc1, TSPX_BS_3x_sacp1, svc) for OO_IFsetup_BS2xorBS3x_MO

Test Step Dynamic Behaviour

Test Step Name : BS3xMT(srv : SERVICES; more_itc, both, more_sacp : BOOLEAN; itc1, itc2, sacp1, sacp2 : B_3; ur : B_4; ce : B_2)

Group : Preambles/

Objective : To generate an MT SETUP message with appropriate Bcap for BS3x service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_main			
2		+local_IFsetup local_main			
3		(TCV_srv := srv, TCV_ur := ur, TCV_ur2 := ur, TCV_ce := C_transparent)			
4		+local_bcap1			
5		(TCV_BcapMT1 := TCV_BcapMT2)			
6		[more_itc OR more_sacp OR (itc1 = C_3100Hz) AND (sacp1 = C_X32)]			
7		[more_itc]			
8		+local_bcap2			
9		(TCV_Setup_mt1 := Setup_20(TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes)			2nd SETUP
10		[NOT more_itc]			
11		[itc1 = C_3100Hz]			
12		(TCV_itc2 := C_3100Hz)			
13		[more_sacp]			
14		+local_3100_1(sacp2)			
15		(TCV_Setup_mt1 := Setup_20(TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes, TCV_sacp2 := sacp2)			2nd SETUP
16		[NOT more_sacp]			
17		[sacp1 = C_I440_450]			no second BC
18		[sacp1 = C_X32]			
19		+local_3100_1(sacp1)			
20		(TCV_Setup_mt1 := Setup_20(TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes, TCV_sacp2 := sacp1)			2nd SETUP
21		[(sacp1 <> C_I440_450) AND (sacp1 <> C_X32)]		I	
22		[itc1 = C_UDI]			
23		[more_sacp]			
24		+local_UDI(sacp2)			
25		(TCV_Setup_mt1 := Setup_20(TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes, TCV_sacp2 := sacp2, TCV_itc2 := C_UDI)			2nd SETUP
26		[NOT more_sacp]			no second BC

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
27		[(NOT more_itc) AND (NOT more_sacp) AND ((NOT(itc1=C_3100Hz)) OR (NOT(sacp1=C_X32)))]			
		local_bcap1			
28		[itc1 = C_3100Hz]			
29		+local_3100(sacp1)			
30		(TCV_itc := C_3100Hz, TCV_sacp := sacp1)			
31		(TCV_Setup_mt := Setup_20(TCV_BcapMT2, TI_02), TCV_sacp := sacp1)			1st SETUP
32		+Chmod(TCV_ChRate, TCV_ce, ur, srv)			
33		[itc1 = C_UDI]			
34		+local_UDI(sacp1)			
35		(TCV_Setup_mt := Setup_20(TCV_BcapMT2, TI_02), TCV_sacp := sacp1, TCV_itc := C_UDI)			
36		[sacp1 = C_X32]			
37		+Chmod(TCV_ChRate, C_notransparent, ur, srv)			
38		[NOT (sacp1 = C_X32)]			
39		+Chmod(TCV_ChRate, TCV_ce, ur, srv)			
40		[(itc1 <> C_3100Hz) AND (itc1 <> C_UDI)]		I	
		local_bcap2			
41		[itc2 = C_3100Hz]			
42		[more_sacp]			
43		+local_3100_1(sacp2)			
44		(TCV_itc2 := C_3100Hz, TCV_sacp2 := sacp2)			
45		[NOT more_sacp AND (sacp1 <> C_X21)]			
46		+local_3100_1(sacp1)			
47		(TCV_itc2 := C_3100Hz, TCV_sacp2 := sacp1)			
48		[NOT more_sacp AND (sacp1 = C_X21)]			no second BC
49		[itc2 = C_UDI]			
50		[more_sacp]			
51		+local_UDI(sacp2)			
52		(TCV_itc2 := C_UDI, TCV_sacp2 := sacp2)			
53		[NOT more_sacp]			
54		+local_UDI(sacp1)			
55		(TCV_itc2 := C_UDI, TCV_sacp2 := sacp1)			
56		[(itc2 <> C_3100Hz) AND (itc2 <> C_UDI)]		I	
		local_3100(sacp : B_3)			
57		[sacp = C_I440_450]			
58		+local_BS3x_3100			
59		[sacp = C_X32]			
60		[NOT (ur= C_1200bs)]			
61		[both OR (ce = C_notransparent)]			
62		+local_BS3x_3100_X32_NT(C_notransparent)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
63		(TCV_ce := C_notransparent)			
64		[NOT both AND (ce = C_transparent)]			
65		+local_BS3x_3100_X32_T			
66		[ur= C_1200bs]		I	BS31 not support X32
67		[(sacp <> C_I440_450) AND (sacp <> C_X32)]		I	
		local_3100_1(sacp : B_3)			
68		[sacp = C_I440_450]			
69		+local_BS3x_3100			
70		[sacp = C_X32]			
71		+local_BS3x_3100_X32_NT(C_BothT)			
72		[(sacp <> C_I440_450) AND (sacp <> C_X32)]		I	
		local_UDI(sacp : B_3)			
73		[sacp = C_I440_450]			
74		[ur = C_9600bs]			
75		+ltree_BS3x_UDI_I440_T(C_ir_16kbs)			
76		[ur <> C_9600bs]			
77		+ltree_BS3x_UDI_I440_T(C_ir_8kbs)			
78		[sacp = C_X21]			
79		[ur = C_9600bs]			
80		+ltree_BS3x_UDI_X21_T(C_ir_16kbs)			
81		[ur <> C_9600bs]			
82		+ltree_BS3x_UDI_X21_T(C_ir_8kbs)			
83		[sacp = C_X32]			
84		[ur <> C_1200bs]			
85		+ltree_BS3x_UDI_X32_NT			
86		[ur = C_1200bs]		I	
87		[(sacp <> C_I440_450) AND (sacp <> C_X21) AND (sacp <> C_X32)]		I	
		ltree_BS3x_3100_T(ir : B_2; modemt : B_5)			
88		(TCV_BcapMT2 := Bcap_MT(C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, modemt))			11.8.1.2 .1.1
		ltree_BS3x_UDI_I440_T(ir : B_2)			
89		(TCV_BcapMT2 := Bcap_MT(C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, C_I440_450, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, C_modemt_none))			11.8.1.2 .1.2
		ltree_BS3x_UDI_X21_T(ir : B_2)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
90		(TCV_BcapMT2 := Bcap_MT(C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, C_X21, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, C_modemt_none))			11.8.1.2 .1.2
		ltree_BS3x_3100_X32_T(ir : B_2; modemt : B_5)			
91		(TCV_BcapMT2 := Bcap_MT(C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, modemt))			11.8.1.2 .1.3
		ltree_BS3x_3100_X32_NT(modemt : B_5; ce1 : B_2)			
92		(TCV_BcapMT2 := BcapX_MT(C_3100Hz, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, ce1, modemt, C_X25_flct))			11.8.1.2 .1.4
		ltree_BS3x_UDI_X32_NT			
93		(TCV_BcapMT2 := BcapX_MT(C_UDI, C_SDUintegrity, C_nirr_nomeaning, C_rate_adaption_X31, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, C_notransparent, C_modemt_none, C_X25_flct))			11.8.1.2 .1.5
		local_BS3x_3100			
94		[ur = C_1200bs]			
95		+ltree_BS3x_3100_T(C_ir_8kbs, C_modemt_V22)			
96		[ur = C_2400bs]			
97		+ltree_BS3x_3100_T(C_ir_8kbs, C_modemt_V22bis)			
98		[ur = C_4800bs]			
99		+ltree_BS3x_3100_T(C_ir_8kbs, C_modemt_V32)			
100		[ur = C_9600bs]			
101		+ltree_BS3x_3100_T(C_ir_16kbs, C_modemt_V32)			
102		[(ur <> C_1200bs) AND (ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
		local_BS3x_3100_X32_NT(ce2 : B_2)			
103		[ur = C_2400bs]			
104		+ltree_BS3x_3100_X32_NT(C_modemt_V22bis, ce2)			
105		[(ur = C_4800bs) OR (ur = C_9600bs)]			
106		+ltree_BS3x_3100_X32_NT(C_modemt_V32, ce2)			
107		[(ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
		local_BS3x_3100_X32_T			
108		[ur = C_2400bs]			
109		+ltree_BS3x_3100_X32_T(C_ir_8kbs, C_modemt_V22bis)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
110		[ur = C_4800bs]			
111		+ltree_BS3x_3100_X32_T(C_ir_8kbs, C_modemt_V32)			
112		[ur = C_9600bs]			
113		+ltree_BS3x_3100_X32_T(C_ir_16kbs, C_modemt_V32)			
114		[(ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
		local_IFsetup			
115		[TCV_supported]			
116		(TCV_Null := OO_IFsetup_BS3x_MT(TCV_srv, TCV_itc, TCV_sacp))			
117		[NOT TCV_supported]			
118		[TCV_SpecialCase]			
119		START T_dly(C_T_Wait)			
120		?TIMEOUT T_dly			
121		[NOT TCV_SpecialCase]		I	Paramet er error

Detailed Comments : The algorithm for derivation of BS3x Bearer Capabilities :

The Bearer Capability for 1st SETUP (used for an MT call):
itc= TSPX_BS_3x_itc1
sacp= TSPX_BS_3x_sacp1
IF itc= 3.1Khz
IF sacp= I440 use coding of sec. 11.8.1.2.1.1 with (ce=T)
IF sacp= X32
IF TSPX_BS_3x_T_NT OR (TSPX_BS_3x_ce = NT)
use coding of sec. 11.8.1.2.1.4 with (ce= NT)
IF NOT TSPX_BS_3x_T_NT AND (TSPX_BS_3x_ce = T)
use coding of sec. 11.8.1.2.1.3 with (ce= T)
IF itc=udi
IF sacp= I440 use coding of sec. 11.8.1.2.1.2 with (ce=T)
IF sacp= X21 use coding of sec. 11.8.1.2.1.2 with (ce=T)
IF sacp= X32 use coding of sec. 11.8.1.2.1.5 with (ce=NT)

The derived 2nd Bearer Capability for 2nd SETUP (used for TC_11_1_ only):
Note: the 2nd BC is derived only IF
TSPX_BS_3x_more_itc OR TSPX_BS_3x_more_sacp OR
((TSPX_BS_3x_itc1 = 3.1kHZ) AND (TSPX_BS_3x_scap1 = X32))
IF TSPX_BS_3x_more_itc
itc= TSPX_BS_3x_itc2
IF itc= 3.1kHz
IF TSP_BS_3x_more_sacp
sacp= TSPX_BS_3x_sacp2
ELSE
sacp = TSPX_BS_3x_sacp1
IF sacp= I440 use coding of sec. 11.8.1.2.1.1 with (ce=T)
IF sacp= X32 use coding of sec. 11.8.1.2.1.4 with (ce=both/T)
IF itc= udi
IF TSP_BS_3x_more_sacp
sacp= TSPX_BS_3x_sacp2
IF sacp= I440 use coding of sec. 11.8.1.2.1.2 with (ce=T)
IF sacp= X21 use coding of sec. 11.8.1.2.1.2 with (ce=T)
IF sacp= X32 use coding of sec. 11.8.1.2.1.5 with (ce=NT)
ELSE
sacp= TSPX_BS_3x_sacp1
IF sacp=I440 use coding of sec. 11.8.1.2.1.2 with (ce=T)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

IF sacp= X32 use coding of sec. 11.8.1.2.1.5 with (ce=NT)
ELSE
itc= TSPX_BS_3x_itc1
IF itc= 3.1kHz
IF TSPX_BS_3x_more_sacp
sacp= TSPX_BS_3x_sacp2
IF sacp= I440
use coding of sec. 11.8.1.2.1.1 with (ce=T)
IF sacp= X32
use coding of sec. 11.8.1.2.1.4 with (ce=both/T)
ELSE
sacp= TSPX_BS_3x_sacp1
IF sacp= I440 no second Bearer Capability
IF sacp= X32 use coding of sec. 11.8.1.2.1.4 with (ce=both/T)
IF itc= udi
IF TSP_BS_3x_more_sacp
sacp= TSPX_BS_3x_sacp2
IF sacp= I440 use coding of sec. 11.8.1.2.1.2 with (ce=T)
IF sacp= X21 use coding of sec. 11.8.1.2.1.2 with (ce=T)
IF sacp= X32 use coding of sec. 11.8.1.2.1.5 with (ce=NT)
ELSE
sacp= TSPX_BS_3x_sacp1
IF sacp=I440 no second Bearer Capability
IF sacp=X21 no second Bearer Capability
IF sacp= X32 no second Bearer Capability

Test Step Dynamic Behaviour

Test Step Name : BS4xMO(srv : SERVICES; ur : B_4; ce : B_2; both : BOOLEAN)

Group : Preambles/

Objective : To generate an MO SETUP message with appropriate Bcap and LLC for BS4x service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_ur := ur, TCV_CE := NT_BothT_BothNTorT_BothT_BothNT, TCV_ir := C_ir_16kbs)			
2		+local_pixitvaluechk			validate IXIT values
3		+local_selectCe			select a CE
4		+local_bcap			generate BC
5		(TCV_Setup_mo := Setup_10_UDI(TCV_BcapMO1, Llcmp_BS2xOrBS4x_UDI, Hlcmp_AnyOrOmit))			generate SETUP
6		+Chmod(TCV_ChRate, TCV_ce, ur, srv)			initiate channel mode variables
7		(TCV_Null := OO_IFsetup_BS4x_MO(TCV_srv, TCV_ce, TCV_ChRate))			ask test operator to set up the MS
		local_bcap			
8		[NOT both]			
9		[TCV_ce = C_notransparent]			
10		+local_BS4x_UDI_NT(C_notransparent)			
11		[TCV_ce = C_transparent]			CE:T only MS
12		[ur = C_9600bs]			
13		+local_BS4x_UDI_T(C_ir_16kbs)			
14		[(ur = C_300bs) OR (ur = C_1200bs) OR (ur = C_120075bs) OR (ur = C_2400bs) OR (ur = C_4800bs)]			
15		+local_BS4x_UDI_T(C_ir_8kbs)			
16		[both]			
17		[TCV_ce = C_transparent]			
18		+local_BS4x_UDI_Tboth(TCV_CE.TorBothTor BothNT)			
19		[(TCV_ce = C_notransparent)]			
20		+local_BS4x_UDI_NT(TCV_CE.NTorBothTorB othNT)			
		local_BS4x_UDI_T(ir : B_2)			
21		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_UDI, C_Unstructured, C_nirr_nomeaning, C_rate_adaption_V110, C_X28_unui, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, ir, C_parity_dontcare, C_transparent, C_modemt_none), TCV_ir := ir)			11.8.2.3 .1.1
		local_BS4x_UDI_NT(ce1 : B_2)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
22		[(ur = C_300bs) OR (ur = C_1200bs) OR (ur = C_120075bs) OR (ur = C_2400bs) OR (ur = C_4800bs)]			
23		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_SDUintegrity, C_nirr_dontcare, C_rate_adaption_V110, C_X28_unui, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, ce1, C_modemt_none, C_uil2p_dontcare))			11.8.2.3 .1.2
24		[ur = C_9600bs]			
25		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_SDUintegrity, C_nirr_nomeaning, C_rate_adaption_V110, C_X28_unui, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, ce1, C_modemt_none, C_uil2p_dontcare))			11.8.2.3 .1.2
		local_BS4x_UDI_Tboth(ce1 : B_2)			
26		[(ur = C_300bs) OR (ur = C_1200bs) OR (ur = C_120075bs) OR (ur = C_2400bs) OR (ur = C_4800bs)]			
27		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_struc_dontchk, C_nirr_dontcare, C_rate_adaption_V110, C_X28_unui, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8or16kbs, C_parity_dontcare, ce1, C_modemt_none, C_uil2p_dontcare))			11.8.2.3 .1.2 conbine d with 11.8.2.3 .1.1, CE, IR and structure without exact match
28		[ur = C_9600bs]			
29		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_struc_dontchk, C_nirr_nomeaning, C_rate_adaption_V110, C_X28_unui, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8or16kbs, C_parity_dontcare, ce1, C_modemt_none, C_uil2p_dontcare))			11.8.2.3 .1.2 conbine d with 11.8.2.3 .1.1, CE, IR and structure without exact match
		local_pixitvaluechk			
30		[(ce <> C_transparent) AND (ce <> C_notransparent) AND (NOT both)]		I	IXIT error
31		[C_Yes]			IXIT values OK
		local_selectCe			
32		[both]			
33		[(ur = C_300bs) OR (ur = C_120075bs) OR (ur = C_4800bs)]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
34		(TCV_ce := C_notransparent)			
35		[(ur = C_1200bs) OR (ur = C_2400bs) OR (ur = C_9600bs)]			
36		(TCV_ce := C_transparent)			
37		[NOT both]			
38		(TCV_ce := ce)			

Detailed Comments : The BS4x Bearer Capabilities and SETUP messages for MO is constructed according to section 11.8.2.3.1.1 and section 11.8.2.3.1.2 of GSM 11.10 and section B.1.4 and section B.2.2.1 of GSM 07.01.

The algorithm for selecting parameters:

```
IF (TSPX_BS_4x_T_NT)
        IF (user rate is one of the C_300bs, C_120075bs, C_4800bs)
                TCV_ce := C_notransparent
        ELSE   /* user rate is one of C_1200bs, C_2400bs, C_9600bs */
                TCV_ce := C_transparent
ELSE
        TCV_ce := TSPX_BS_4x_ce

use (TCV_ce, TSPX_BS_4x_ur, svc) for OO_IFsetup_BS4x_MO
```

Test Step Dynamic Behaviour

Test Step Name : BS5xMO(srv : SERVICES; ur : B_4)

Group : Preambles/

Objective : To generate an MO SETUP message with appropriate Bcap for BS5x service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_ur := ur)			
2		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_UDI, C_SDUintegrity, C_nirr_nomeaning, C_rate_adaption_X31, C_X32, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, C_notransparent, C_modemt_none, C_X25_flct))			11.8.2.4 .1
3		(TCV_Setup_mo := Setup_10_UDI(TCV_BcapMO1, Llcmp_BS5x, Hlcmp_AnyOrOmit))			
4		+Chmod(TCV_ChRate, C_notransparent, ur, srv)			channel mode
5		(TCV_Null := OO_IFsetup_BS5x_MO(TCV_srv, TCV_ur, TCV_ChRate))			

Detailed Comments : 1. Bearer Caoability is coded according to section 11.8.2.4.1 of GSM 11.10 ver 4.16
2. low layer compatibility is coded according to section B.2.5 of GSM 07.01 ver 4.90

Test Step Dynamic Behaviour

Test Step Name : BS61or81MO(srv : SERVICES; async, both: BOOLEAN; a_ur1: B_4; ce : B_2; sync : BOOLEAN; s_ur1: B_4; ri : B_8)

Group : Preambles/

Objective : To get a MO SETUP message with right BC IE and LLC for BS61 or BS81 service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_CE := NT_BothT_BothNTorT_BothT_BothNT, TCV_ir := C_ir_8kbs, TCV_MODEM := V22bisOrV26ter)			
2		+local_pixitvaluechk			1.
3		+local_selectSaAndCe			2.
4		+local_bcap			generate BC
5		+local_setupmsg			generate SETUP
6		+Chmod(TCV_ChRate, TCV_ce, TCV_ur, srv)			
7		(TCV_Null := OO_IFsetup_BS61orBS81_MO(TCV_srv, TCV_ur, TCV_sa, TCV_ce, TCV_ChRate))			
		local_bcap			
8		[TCV_sa = C_Synchronous]			
9		[TCV_ur = C_1200bs]			
10		+local_BS61or81_S_T(C_ir_8kbs, C_1200bs, C_modemt_V22)			
11		[TCV_ur = C_2400bs]			
12		+local_BS61or81_S_T(C_ir_8kbs, C_2400bs, TCV_MODEM.V22bisOrV26ter)			
13		[TCV_ur = C_4800bs]			
14		+local_BS61or81_S_T(C_ir_8kbs, C_4800bs, C_modemt_V32)			
15		[TCV_ur = C_9600bs]			
16		+local_BS61or81_S_T(C_ir_16kbs, C_9600bs, C_modemt_V32)			
17		(TCV_ir := C_ir_16kbs)			
18		[TCV_sa = C_Asynchronous]			
19		[(TCV_ce = C_transparent) AND (NOT both)]			CE:T only MS
20		+local_BS61or81_A_T(TCV_ur)			
21		[(TCV_ce = C_notransparent) AND (NOT both)]			CE:NT only MS
22		+local_BS61or81_A_NT(TCV_ur, C_notransparent, C_SDUintegrity, C_nirr_dontcare, C_ir_16kbs)			
23		[(TCV_ce = C_notransparent) AND both]			
24		+local_BS61or81_A_NT(TCV_ur, TCV_CE.NTorBothTorBothNT, C_SDUintegrity, C_nirr_dontcare, C_ir_16kbs)			
25		[(TCV_ce = C_transparent) AND both]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
26		+local_BS61or81_A_NT(TCV_ur, TCV_CE.TorBothTorBothNT, C_struc_dontchk, C_nirr_dontcare, C_ir_8or16kbs) local_BS61or81_A_T(ur : B_4)			
27		[ur = C_300bs]			
28		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, C_modemt_V21))			11.8.2.5 .1
29		[ur = C_1200bs]			
30		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, C_modemt_V22))			11.8.2.5 .1
31		[ur = C_120075bs]			
32		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, C_modemt_V23))			11.8.2.5 .1
33		[ur = C_2400bs]			
34		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, TCV_MODEM.V22bisOrV26ter))			11.8.2.5 .1
35		[ur = C_4800bs]			
36		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_8kbs, C_parity_dontcare, C_transparent, C_modemt_V32))			11.8.2.5 .1
37		[ur = C_9600bs]			
38		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, C_transparent, C_modemt_V32))			11.8.2.5 .1
39		(TCV_ir := C_ir_16kbs) local_BS61or81_A_NT(ur : B_4; ce : B_2; struc : B_2 ; nirr :B_1; ir : B_2)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
40		(TCV_ir := C_ir_16kbs)			
41		[ur = C_300bs]			
42		(TCV_MODEM := V21orAbt1, TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, struc, nirr, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, ir, C_parity_dontcare, ce, TCV_MODEM.V22bisOrV26ter, C_uil2p_dontcare))			11.8.2.5 .2
43		[ur = C_1200bs]			
44		(TCV_MODEM := V22orAbt1, TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, struc, nirr, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, ir, C_parity_dontcare, ce, TCV_MODEM.V22bisOrV26ter, C_uil2p_dontcare))			11.8.2.5 .2
45		[ur = C_120075bs]			
46		(TCV_MODEM := V23orAbt1, TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, struc, nirr, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, ir, C_parity_dontcare, ce, C_modemt_V23, C_uil2p_dontcare))			11.8.2.5 .2
47		[ur = C_2400bs]			
48		(TCV_MODEM := V22bisOrV26terOrAbt1, TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, struc, nirr, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, ir, C_parity_dontcare, ce, TCV_MODEM.V22bisOrV26ter, C_uil2p_dontcare))			11.8.2.5 .2
49		[ur = C_4800bs]			
50		(TCV_MODEM := V32orAbt1, TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, struc, nirr, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, ir, C_parity_dontcare, ce, TCV_MODEM.V22bisOrV26ter, C_uil2p_dontcare))			11.8.2.5 .2
51		[ur = C_9600bs]			
52		(TCV_MODEM := V32orAbt1, TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_3100Hz, struc, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, C_stopbit_dontcare, C_databit_dontcare, ur, C_ir_16kbs, C_parity_dontcare, ce, TCV_MODEM.V22bisOrV26ter, C_uil2p_dontcare)) local_BS61or81_S_T(ir : B_2; ur: B_4; modemt : B_5)			11.8.2.5 .2
53		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, modemt))			11.8.2.5 .3

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
		local_pixitvaluechk			
54		[(ce <> C_transparent) AND (ce <> C_notransparent) AND (NOT both) AND async]		I	IXIT error
55		[(a_ur1 <> C_300bs) AND (a_ur1 <> C_1200bs) AND (a_ur1 <> C_2400bs) AND (a_ur1 <> C_4800bs) AND (a_ur1 <> C_9600bs) AND (a_ur1 <> C_120075bs) AND async]		I	IXIT error
56		[(s_ur1 <> C_1200bs) AND (s_ur1 <> C_2400bs) AND (s_ur1 <> C_4800bs) AND (s_ur1 <> C_9600bs) AND sync]		I	IXIT error
57		[C_Yes]			IXIT values OK
		local_selectSaAndCe			
58		[async]			3.
59		(TCV_sa := C_Asynchronous, TCV_ur := a_ur1)			
60		[both]			
61		[(a_ur1 = C_300bs) OR (a_ur1 = C_120075bs) OR (a_ur1 = C_4800bs)]			
62		(TCV_ce := C_notransparent)			
63		[(a_ur1 = C_1200bs) OR (a_ur1 = C_2400bs) OR (a_ur1 = C_9600bs)]			
64		(TCV_ce := C_transparent)			
65		[NOT both]			
66		(TCV_ce := ce)			
67		[sync]			
68		(TCV_sa := C_Synchronous, TCV_ce := C_transparent, TCV_ur := s_ur1)			
		local_setupmsg			
69		[TCV_sa = C_Asynchronous]			
70		(TCV_Setup_mo := Setup_11_3100(ri, Bcap_Speech_MO(C_Rchr_dontcare), TCV_BcapMO1, Llcmp_NotApplicable, Llcmp_BS2xOrBS61aOrBS81a_3100, ri, ri, Hlcmp_NotApplicable, Hlcmp_AnyOrOmit))			
71		[TCV_sa = C_Synchronous]			
72		(TCV_Setup_mo := Setup_11_3100(ri, Bcap_Speech_MO(C_Rchr_dontcare), TCV_BcapMO1, Llcmp_NotApplicable, Llcmp_BS3xOrBS61sOrBS81s_3100, ri, ri, Hlcmp_NotApplicable, Hlcmp_AnyOrOmit))			

Detailed Comments : 1. validate IXIT values.
2. select sync or async mode and corresponding CE.
3. If the MS supports both synchronous and asynchronous mode, selete the asynchronous mode.
4. The BS61 or BS81 Bearer Capabilities and SETUP messages for MO is constructed according to section 11.8.2.5.1, section 11.8.2.5.2 and section 11.8.2.5.3 of GSM 11.10 and section B.1.6.1, section B.1.6.2.1, section B.1.6.2.2, section B.2.2.2 and section B.2.3.2 of GSM 07.01.

The algorithm for selecting parameters for BS61 basice service:
IF (TSPX_BS_61_A)
TCV_sa := C_Asynchronous
IF (TSPX_BS_61_A_ur1_T_NT)
TCV_ce := C_notransparent
ELSE

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour	
Detailed Comments	**:** ... TCV_ce := TSPX_BS_61_A_ur1_ce use (TCV_sa, TCV_ce, TSPX_BS_61_A_ur1, svc) for OO_IFsetup_BS61orBS81_MO ELSE TCV_sa := C_Synchronous use (TCV_sa, C_transparent, TSPX_BS_61_S_ur1, svc) for OO_IFsetup_BS61orBS81_MO The algorithm for selecting parameters for BS81 basice service is the same as above except using corresponding IXIT parameters.

Test Step Dynamic Behaviour

Test Step Name : BS61or81MT(srv : SERVICES; async, t_nt_ur1, t_nt_ur2, a_more_ur, sync, s_more_ur : BOOLEAN; a_ur1, a_ur2, s_ur1, s_ur2: B_4; ri : B_8; ce : B_2)

Group : Preambles/

Objective : To get a MT SETUP message with right BC IE for BS61 or BS81 service.

Default : OtherEvents

Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_main			
2		+local_IFsetup			
		local_main			
3		(TCV_srv := srv)			
4		(TCV_BcapMT1 := Bcap_Speech_MT)			
5		[async]			If asynchro
6		+local_bcapa			
7		(TCV_Setup_mt := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_ur := a_ur1, TCV_sa := C_Asynchronous)			1st SETUP
8		+Chmod(TCV_ChRate, TCV_ce, a_ur1, srv)			2nd ch mode
9		(TCV_ChMod.mode := C_ChMod_r)			Speech mode
10		[sync]			
11		+local_bcap2S(s_ur1)			
12		(TCV_Setup_mt1 := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_2ndtest :=C_Yes, TCV_ur2 := s_ur1, TCV_sa2 := C_Synchronous)			2nd SETUP
13		[a_more_ur]			
14		[t_nt_ur2]			
15		+local_bcap2A(a_ur2, C_transparent)			
16		(TCV_Setup_mt1 := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_2ndtest :=C_Yes, TCV_ur2 := a_ur2, TCV_sa2 := C_Asynchronous)			2nd SETUP
17		[NOT t_nt_ur2]			
18		+local_bcap2A(a_ur2, C_BothT)			
19		(TCV_Setup_mt1 := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_2ndtest :=C_Yes, TCV_ur2 := a_ur2, TCV_sa2 := C_Asynchronous)			2nd SETUP
20		[NOT a_more_ur]			
21		[t_nt_ur1]			
22		+local_bcap2A(a_ur1, C_transparent)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
23		(TCV_Setup_mt1 := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes, TCV_ur2 := a_ur1, TCV_sa2 := C_Asynchronous)			2nd SETUP
24		[NOT t_nt_ur1]			
25		+local_bcap2A(a_ur1, C_BothT)			
26		(TCV_Setup_mt1 := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_2ndtest := C_Yes, TCV_ur2 := a_ur1, TCV_sa2 := C_Asynchronous)			2nd SETUP
27		[sync]			If synchro
28		+local_bcap2S(s_ur1)			
29		(TCV_Setup_mt := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_ur := s_ur1, TCV_sa := C_Synchronous)			1st SETUP
30		+Chmod(TCV_ChRate, C_transparent, s_ur1, srv)			2nd ch mode
31		(TCV_ChMod.mode := C_ChMod_r)			Speech mode
32		[s_more_ur]			
33		+local_bcap2S(s_ur2)			
34		(TCV_Setup_mt1 := Setup_21(ri, TCV_BcapMT1, TCV_BcapMT2, TI_02), TCV_2ndtest :=C_Yes, TCV_ur2 := s_ur1, TCV_sa2 := C_Synchronous)			2nd SETUP
35		[NOT s_more_ur]			no second BC
36		[C_Yes]		I	If asynchro
		local_bcap2A(ur : B_4; ce2: B_2)			
37		[ce2 = C_transparent]			
38		[ur = C_300bs]			
39		+local_BS61or81_A_T(C_ir_8kbs, ur, C_modemt_V21)			
40		[ur = C_1200bs]			
41		+local_BS61or81_A_T(C_ir_8kbs, ur, C_modemt_V22)			
42		[ur = C_2400bs]			
43		+local_BS61or81_A_T(C_ir_8kbs, ur, C_modemt_V22bis)			
44		[ur = C_4800bs]			
45		+local_BS61or81_A_T(C_ir_8kbs, ur, C_modemt_V32)			
46		[ur = C_9600bs]			
47		+local_BS61or81_A_T(C_ir_16kbs, ur, C_modemt_V32)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
48		[(ur <> C_300bs) AND (ur <> C_1200bs) AND (ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
49		[ce2 = C_BothT]			
50		[ur = C_300bs]			
51		+local_BS61or81_A_NT(ur, C_modemt_V21, C_BothT)			
52		[ur = C_1200bs]			
53		+local_BS61or81_A_NT(ur, C_modemt_V22, C_BothT)			
54		[ur = C_2400bs]			
55		+local_BS61or81_A_NT(ur, C_modemt_V22bis, C_BothT)			
56		[ur = C_4800bs]			
57		+local_BS61or81_A_NT(ur, C_modemt_V32, C_BothT)			
58		[ur = C_9600bs]			
59		+local_BS61or81_A_NT(ur, C_modemt_V32, C_BothT)			
60		[(ur <> C_300bs) AND (ur <> C_1200bs) AND (ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
61		[ce2 = C_notransparent]			
62		+local_BS61or81_A_NT(ur, C_modemt_abt1, C_notransparent)			
63		[ce2 = C_BothNT]		I	
		local_bcap2S(ur : B_4)			
64		[ur = C_1200bs]			
65		+local_BS61or81_S_T(C_ir_8kbs, ur, C_modemt_V22)			
66		[ur = C_2400bs]			
67		+local_BS61or81_S_T(C_ir_8kbs, ur, C_modemt_V22bis)			
68		[ur = C_4800bs]			
69		+local_BS61or81_S_T(C_ir_8kbs, ur, C_modemt_V32)			
70		[ur = C_9600bs]			
71		+local_BS61or81_S_T(C_ir_16kbs, ur, C_modemt_V32)			
72		[(ur <> C_1200bs) AND (ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
		local_BS61or81_A_T(ir : B_2; ur : B_4; modemt : B_5)			
73		(TCV_BcapMT2 := Bcap_MT(C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, ir, TSPX_TE_parity, C_transparent, modemt))			11.8.1.3 .1
		local_BS61or81_A_NT(ur : B_4; modemt : B_5; cel : B_2)			
74		[TSPX_TE_FLCT = C_Outband]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
75		(TCV_BcapMT2 := Bcap_MT(C_3100Hz, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, modemt))			11.8.1.3 .2
76		[TSPX_TE_FLCT = C_Inband]			
77		(TCV_BcapMT2 := BcapX_MT(C_3100Hz, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, modemt, C_ISO6429))			11.8.1.3 .2
78		[TSPX_TE_FLCT = C_Nocontrol]			
79		(TCV_BcapMT2 := BcapX_MT(C_3100Hz, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Asynchronous, TSPX_TE_stopbit, TSPX_TE_databit, ur, C_ir_16kbs, TSPX_TE_parity, cel, modemt, C_COPnoFLCT))			11.8.1.3 .2
80		[(TSPX_TE_FLCT <> C_Outband) AND (TSPX_TE_FLCT <> C_Inband) AND (TSPX_TE_FLCT <> C_Nocontrol)]		I	
		local_BS61or81_S_T(ir : B_2; ur: B_4; modemt : B_5)			
81		(TCV_BcapMT2 := Bcap_MT(C_3100Hz, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_I440_450, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, modemt))			11.8.1.3 .3
		local_bcapa			
82		[t_nt_ur1 OR (ce = C_notransparent)]			
83		+local_bcap2A(a_ur1, C_notransparent)			
84		(TCV_ce := C_notransparent)			
85		[NOT t_nt_ur1 AND (ce = C_transparent)]			
86		+local_bcap2A(a_ur1, C_transparent)			
87		(TCV_ce := C_transparent)			
		local_IFsetup			
88		[TCV_supported]			
89		(TCV_Null := OO_IFsetup_BS61orBS81_MT(TCV_srv, TCV_ur, TCV_sa))			
90		[NOT TCV_supported]			
91		[TCV_SpecialCase]			
92		START T_dly(C_T_Wait)			
93		?TIMEOUT T_dly			
94		[NOT TCV_SpecialCase]		I	Parameter error

Detailed Comments : The algorithm for derivation of BS61 Bearer Capabilities:

The data Bearer Capability for 1st SETUP (used for an MT call):
IF TSPX_BS_61_A
ur= TSPX_BS_61_A_ur1
IF TSPX_BS_61_A_ur1_T_NT OR (TSPX_BS_61_A_ur1_ce = NT)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

```
            use coding of sec. 11.8.1.3.2 with (ur & ce=NT)
        ELSE
            use coding of sec. 11.8.1.3.1 with (ur & ce=T)
    ELSE
       IF TSPX_BS_61_S
        ur= TSPX_BS_61_S_ur1 use coding of sec. 11.8.1.3.3 with (ur & ce=T)


    The 2nd Bearer Capability for second SETUP ( used for TC_11_1_1):
        Note: the 2nd BC is derived IF
         (TSPX_BS_61_A OR (TSPX_BS_61_S AND TSPX_BS_61_S_more_ur))

    IF TSPX_BS_61_A AND TSPX_BS_61_S
        ur= TSPX_BS_61_S_ur1 use coding of sec. 11.8.1.3.3 with (ur & ce= T)
    ELSE
        IF TSPX_BS_61_A_more_ur  AND TSPX_BS_61_A
              ur= TSPX_BS_61_A_ur2
            IF TSPX_BS_61_A_ur2_T_NT
                use coding of sec. 11.8.1.3.1 with (ur & ce=T)
            ELSE
                use coding of sec. 11.8.1.3.2 with (ur & ce=both/T)
        ELSE
            IF TSPX_BS_61_S AND TSPX_BS_61_S_more_ur
                ur=TSPX_BS_61_S_ur2
                use coding of sec. 11.8.1.3.3 with (ur & ce= T)
            ELSE
                IF TSPX_BS_61_A
                     ur= TSPX_BS_61_A_ ur1
                   IF TSPX_BS_61_A_ur1_T_NT
                     use coding of sec. 11.8.1.3.1 with (ur & ce=T)
                   ELSE
                     use coding of sec. 11.8.1.3.2 with (ur & ce=both/T)

    The algorithm for derivation of BS81 Bearer Capabilities is same as above except all parameters are
    changed to corresponding BS81's parameters.
```

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		[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		[(ur = C_9600bs) AND (rate = C_Half)]		I	not applicable

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : Est_MO_Call_init_R (par1: OCTETSTRING; mag : MA; ta : TA; tsc1 : B_3; nfrqg, arfcnA : 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		(TCV_Null := OM_NoL2Ack(2, TCV_ch))			
9		+Ciphering_on(TCV_ch)			
10		L?DL_DatInSetup (TCV_Mt := DL_DatInSetup.msg.mt)	SetupRcv(SetupInd_01)		
		localtree_ia(par_chn: OCTETSTRING)			
11		[par_chn = C_CHTCHF_NonFH]			
12		(TCV_Tchtype := '00001'B)			
13		+ltree_immass_nfh(TCV_tch_arfcn, tsc1)			
14		[par_chn = C_CHSDCCH4_NonFH]			
15		(TCV_Tchtype := INT_TO_BIT((4 + BIT_TO_INT(TSPX_SDCCH4SubDef)), 5))			
16		+ltree_immass_nfh(TCV_chdescr_arfcn, tsc1)			
17		[par_chn = C_CHSDCCH8_NonFH]			
18		(TCV_Tchtype := INT_TO_BIT((8 + BIT_TO_INT(TSPX_SDCCH8SubDef)), 5))			
19		+ltree_immass_nfh(TCV_tch_arfcn, tsc1)			
20		[par_chn = C_CHTCHF_FH]			
21		(TCV_Tchtype := '00001'B)			
22		+ltree_immass_fh(nfrqg)			
23		[par_chn = C_CHTCHH_FH]			
24		(TCV_Tchtype := INT_TO_BIT((2 + BIT_TO_INT(TSPX_TCHHSubDef)), 5))			
25		+ltree_immass_fh(nfrqg)			
26		[par_chn = C_CHTCHH_NonFH]			
27		(TCV_Tchtype := INT_TO_BIT((2 + BIT_TO_INT(TSPX_TCHHSubDef)), 5))			
28		+ltree_immass_nfh(arfcnA, tsc1)			
29		[par_chn = C_CHSDCCH8_FH]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
30		(TCV_Tchtype := INT_TO_BIT((8 + BIT_TO_INT(TSPX_SDCCH8SubDef)), 5))			
31		+ltree_immass_fh(nfrqg)			
32		[C_Yes]			
		ltree_immass_fh(frqnog : INTEGER)			
33		(TCV_maio := INT_TO_BIT((TSPX_MAIO MOD frqnog), 6), TCV_iel := mag.iel)			1.
34		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))		
		ltree_immass_nfh(arfcn : INTEGER; tsc : B_3)			
35		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 : IdleUpdated_R (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 : CCHD; crh, mtmcgsm : INTEGER; bcchflgsm, bcflaltgsm, bcchfle : NCD; Re : B_1; bcc : BCC; ncc : NCC; StartType : INTEGER; subch : B_2; cksn : B_3; rand : RAND; arfcngsm : 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_Cnt2 := 1)			
2		REPEAT l_idleUpdated UNTIL [TCV_Cnt2 = 0]			
		l_idleUpdated			
3		(TCV_Cnt2 := 0)			Only once
4		(TCV_slot := slot, TCV_tsc := tsc, TCV_Ccd0H := CntrlChDscrp(att, babr, C_cch_1Comb, bpm, t3212))			
5		+MM_PwrOrSimOff (C_SIMneedRmv)			
6		+Varinit_fix_R (activ_cell, lac, subch, cksn, rand, arfcngsm, TCV_Ccd0H, imsi)			
7		+ltree_chconf(C_cch_1Comb)			Config LT
8		+localtree1			
		localtree1			
9		+ltree_sysinfo(1, C_cch_1Comb)			ATT= '1'B
10		+MM_PwrOrSimOn (C_SIMneedRmv)			
11		+localtree_Lup_Auth			
		localtree_Lup_Auth			
12		START T_dly(C_T_Wait1stChReq)			
13		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
14		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
15		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, subch, slot, tsc, TCV_chdescr_arfcn, ta))		
16		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(C_norm_period_attach))		Any location update type
17		ACTIVATE(OtherEvents)			Restore Normal default
18		[StartType = C_Start_Rej]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
19		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(C_rc_LAnotallowed))		Delete LAI, tmsi, cksn
20		+PostMainLinkRel(TCV_ch)			Release Ch
21		+MM_PwrOrSimOff (C_SIMneedRmv)			
22		+localtree2			
23		[StartType = C_Start_Imsi]			
24		+Authentication(TCV_ch, cksn, rand)			
25		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiImsi_01iei, mcc, mnc, TCV_lac))		Location Accept with IMSI
26		+PostMainLinkRel(TCV_ch)			Release Ch
27		+localtree2			
28		[(StartType = C_Start_Tmsi) OR (StartType = C_Start_TmsiOff)]			
29		+Authentication(TCV_ch, cksn, rand)			
30		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiTmsi_01iei, mcc, mnc, TCV_lac))		Location Accept with TMSI
31		L?DL_DatInTmsireCom	TmsiReallocCmp(TCV_ch)		
32		+PostMainLinkRel(TCV_ch)			Release Ch
33		+localtree2			
34		[StartType = C_Start_PLMNnot]			
35		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(C_rc_plmn_not))		Delete LAI, tmsi, cksn
36		+PostMainLinkRel(TCV_ch)			Release Ch
37		+localtree2			
38		[(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	
39		?TIMEOUT T_dly			
40		+ltree_rmvPLMNfbdnlist			
		localtree2			
41		[StartType <> C_Start_TmsiOff]			
42		[cch_con <> C_cch_1Comb]			If non combined
43		+localtree3			
44		+ltree_PgReorg			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
45		[(cch_con = C_cch_1Comb) AND (att = 0)]			If combined, ATT change
46		+ltree_sysinfo(0, C_cch_1Comb)			ATT=0
47		+ltree_PgReorg			
48		[(cch_con = C_cch_1Comb) AND (att = 1)]			If combined, ATT nochange
49		[StartType = C_Start_TmsiOff]			
50		+ImsiDetach (MiTmsi_01, ta, C_SIMneedRmv)			Switch off, detach
51		[cch_con <> C_cch_1Comb]			If non combined
52		+localtree3			
53		[(cch_con = C_cch_1Comb) AND (att = 0)]			If combined
54		+ltree_sysinfo(0, C_cch_1Comb)			ATT=0
55		[(cch_con = C_cch_1Comb) AND (att = 1)]			If combined, ATT nochange
		localtree3			
56		(TCV_Ccd0H := CntrlChDscrp(att, babr, cch_con, bpm, t3212))			
57		+Varinit_fix_R(activ_cell, lac, subch, cksn, rand, arfcngsm, TCV_Ccd0H, imsi)			
58		+ltree_chconf (cch_con)			Config LT
59		+ltree_sysinfo(att, cch_con)			
		ltree_PgReorg			
60		(TCV_Null:=OM_PgFill(activ_cell, PgReqTp1Reorg))			
61		START T_dly(5000)			
62		?TIMEOUT T_dly			
63		(TCV_Null:=OM_PgFill(activ_cell, PgReqTp1Norm))			
		ltree_sysinfo(attach : INTEGER; con_ch : B_3)			
64		[activ_cell = C_CellA]			
65		+SysInfoSending(attach, con_ch)			SI1−4, cellA
66		(TCV_Ccd0A := TCV_Ccd0H, TCV_sysinfo5 := TCV_sysinf5, TCV_sysinfo6 := TCV_sysinf6)			
67		[con_ch = C_cch_1Comb]			If combined

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
68		+SysInfo_SacchSending(TCV_sacch, TCV_sysinfo5, TCV_sysinfo6)			SI5, 6 on SACCH/ C4, cell A
69		[con_ch <> C_cch_1Comb]			
70		+ltree_sysinfo_other(attach)			More SIs, cellA
71		[activ_cell = C_CellB]			
72		+SysInfoSending(attach, con_ch)			SI1−4, cellB
73		(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
74		[con_ch = C_cch_1Comb]			If combined
75		+SysInfo_SacchSending(TCV_sacch_B, TCV_sysinfo5_B, TCV_sysinfo6_B)			SI5, 6 on SACCH/ C4, cell A
76		[con_ch <> C_cch_1Comb]			No more SIs, cell B
77		[activ_cell = C_CellC]			
78		+SysInfoSending(attach, con_ch)			SI1−4, cellC
79		(TCV_Ccd0C := TCV_Ccd0H, TCV_sacch_C := TCV_sacch, TCV_sysinfo5_C := TCV_sysinf5, TCV_sysinfo6_C := TCV_sysinf6)			
80		[con_ch = C_cch_1Comb]			If combined
81		+SysInfo_SacchSending(TCV_sacch_C, TCV_sysinfo5_C, TCV_sysinfo6_C)			SI5, 6 on SACCH/ C4, cell C
82		[con_ch <> C_cch_1Comb]			No more SIs, cell C
		SysInfoSending(attachb : INTEGER; con_chs : B_3)			
83		[sysinfonb = C_Test_fh]			Hopping
84		+ SysInfoSending_fh_R (sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, cchdgsm, crh, mtmcgsm, bcchflgsm, Re, bcc, ncc, nccp)			
85		[sysinfonb = C_Test_nfh]			Non hopping

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
86		+SysInfoSending_nfh_R(sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, crh, mtmcgsm, bcchflgsm, bcflaltgsm, Re, bcc, ncc, nccp)			
87		[sysinfonb = C_Test_rgsm]			EGSM
88		+SysInfoSending_r (sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, cchdgsm, crh, mtmcgsm, bcchflgsm, bcchfle, Re, bcc, ncc, nccp)			
89		[sysinfonb = C_Test_cbms]			CBMS
90		+SysInfoSending_cbms_R (sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, crh, mtmcgsm, bcchflgsm, Re, bcc, ncc, nccp, subch)			
		ltree_chconf(conf : B_3)			
91		[activ_cell = C_CellA]			
92		+ltree_chconf_A(conf)			Config LT
93		[activ_cell = C_CellB]			
94		+ltree_chconf_B(conf)			Config LT
95		[activ_cell = C_CellC]			
96		+ltree_chconf_C(conf)			Config LT
		ltree_chconf_A(conf_a : B_3)			
97		[conf_a = C_cch_1nonComb]			
98		+NonCombinedBCCH_A_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_a, bpm)			1 non combined CCCH, cell A
99		(TCV_Upd := C_NotCombined)			
100		[conf_a = C_cch_1Comb]			
101		[sysinfonb <> C_Test_cbms]			
102		+CombinedBCCH_A_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_a, bpm)			1 combined CCCH, cell A
103		(TCV_Upd := C_Combined)			
104		[sysinfonb = C_Test_cbms]			
105		+CombinedBCCH_A_CBMS_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_a, bpm)			1 combined CCCH + CBMS, cell A
106		(TCV_Upd := C_Combined)			
107		[conf_a = C_cch_2nonComb]			2 CCCHs

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
108		+NonCombinedBCCH_A_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_a, bpm)			1st non combined CCCH, cell A
109		+NonCombinedBCCH_A_2_R(acttype, slot2, tsc, ta, babr, conf_a, bpm)			2nd non combined CCCH, cell A
110		(TCV_Upd := C_NotCombined)			
111		[conf_a = C_cch_3nonComb]			
112		+NonCombinedBCCH_A_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_a, bpm)			1st non combined CCCH, cell A
113		+NonCombinedBCCH_A_2_R(acttype, slot2, tsc, ta, babr, conf_a, bpm)			2nd non combined CCCH, cell A
114		+NonCombinedBCCH_A_3_R(acttype, slot3, tsc, ta, babr, conf_a, bpm)			3rd non combined CCCH, cell A
115		(TCV_Upd := C_NotCombined)			
116		[conf_a = C_cch_4nonComb]			
117		+NonCombinedBCCH_A_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_a, bpm)			1st non combined CCCH, cell A
118		+NonCombinedBCCH_A_2_R(acttype ,slot2, tsc, ta, babr, conf_a, bpm)			2nd non combined CCCH, cell A
119		+NonCombinedBCCH_A_3_R(acttype, slot3, tsc, ta, babr, conf_a, bpm)			3rd non combined CCCH, cell A
120		+NonCombinedBCCH_A_4_R(acttype, slot4, tsc, ta, babr, conf_a, bpm)			4th non combined CCCH, cell A
121		(TCV_Upd := C_NotCombined)			
122		[(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
123		[conf_b = C_cch_1nonComb]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
124		+NonCombinedBCCH_B_R (par_bspwr, mtmcgsm, arfcngsm, acttype,slot,tsc, ta, babr, conf_b, bpm)			1 non combined CCCH, cell B
125		(TCV_Upd := C_NotCombined)			
126		[conf_b = C_cch_1Comb]			
127		+CombinedBCCH_B_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_b, bpm)			1 combined CCCH, cell B
128		(TCV_Upd := C_Combined)			
129		[(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)			
130		[conf_c = C_cch_1Comb]			
131		+ CombinedBCCH_C_R(par_bspwr, mtmcgsm, arfcngsm, acttype, slot, tsc, ta, babr, conf_c, bpm)			1 combined CCCH, cell C
132		(TCV_Upd := C_Combined)			
133		[conf_c <> C_cch_1Comb]		I	
		ltree_sysinfo_other(attach : INTEGER)			
134		[(cch_con = C_cch_1nonComb) OR (cch_con = C_cch_1Comb)]			
135		[cch_con = C_cch_2nonComb]			
136		+SysInfoSending_fh_R(sch, C_BCCH_A_2, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, crh, mtmcgsm, BcchFreqLst_01, Re, bcc, ncc, nccp)			SIs on 2nd CCCH
137		[cch_con = C_cch_3nonComb]			
138		+SysInfoSending_fh_R(sch, C_BCCH_A_2, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, crh, mtmcgsm, BcchFreqLst_01, Re, bcc, ncc, nccp)			SIs on 2nd CCCH
139		+SysInfoSending_fh_R(sch, C_BCCH_A_3, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, crh, mtmcgsm, BcchFreqLst_01, Re, bcc, ncc, nccp)			SIs on 3rd CCCH
140		[cch_con = C_cch_4nonComb]			
141		+SysInfoSending_fh_R(sch, C_BCCH_A_2, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, crh, mtmcgsm, BcchFreqLst_01, Re, bcc, ncc, nccp)			SIs on 2nd CCCH
142		+SysInfoSending_fh_R(sch, C_BCCH_A_3, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, crh, mtmcgsm, BcchFreqLst_01, Re, bcc, ncc, nccp)			SIs on 3rd CCCH

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
143		+SysInfoSending_fh_R(sch, C_BCCH_A_4, t, retr, neci, attach, babr, cch_con, bpm, t3212, ci, mcc, mnc, lac, co, CellChDes_02, crh, mtmcgsm, BcchFreqLst_01, Re, bcc, ncc, nccp)			SIs on 4th CCCH
		ltree_rmvPLMNfbdnlist			
144		(TCV_Null := OO_PLMNselModeMan())			
145		(TCV_Null := OO_SelPLMN(mnc))			
146		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
147		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
148		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, subch, slot, tsc, TCV_chdescr_arfcn, ta))		
149		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(C_norm_period_attach))		Any location update type
150		ACTIVATE(OtherEvents)			Restore Normal default
151		[StartType = C_Start_PLMNnot]			The PLMN is already in forbidden list on short SIM
152		L!DL_DatRqLupRej	LocRej(TCV_ch, LocUpdtRej_01(C_rc_plmn_not))		Delete LAI, tmsi, cksn, PLMN still in list
153		+PostMainLinkRel(TCV_ch)			Release Ch
154		(TCV_Null := OO_PLMNselModeAuto())			
155		[StartType <> C_Start_PLMNnot]			
156		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiTmsi_r02iei, mcc, mnc, TCV_lac))		Location Accept with no TMSI
157		+PostMainLinkRel(TCV_ch)			Release Ch
158		(TCV_Null := OO_PLMNselModeAuto())			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
159		(TCV_Null := OM_StopCell(activ_cell))			
160		(TCV_Cnt2 := 1)			2nd time

Detailed Comments : Formal Parameter Type Comments & Explanations Possible values

par_bspwr INTEGER Base station power level C_E_default, C_E_suitable

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

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

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

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.
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

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

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_rgsm
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 : IdleUpdated_R_CS (par_bspwr : INTEGER; sch, bcch : LOGICCH; acttype : BITSTRING; slot, slot2, slot3, slot4 : SN; tsc : TSC; t, retr, neci, att : INTEGER; ta : TA; babr, bpm : B_3; t3212 : OCTETSTRING; ci : CI; mcc, mnc, lac : OCTETSTRING; co : CO; cchdgsm : CCHD; crh, mtmcgsm : INTEGER; bcchflgsm, bcflaltgsm, bcchflegsm : NCD; Re : B_1; bcc : BCC; ncc : NCC; subch : B_2; cksn : B_3; rand : RAND; arfcngsm : 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_Cnt2 := 1)			
2		REPEAT l_idleUpdated UNTIL [TCV_Cnt2 = 0]			
		l_idleUpdated			
3		(TCV_Cnt2 := 0)			Only once
4		(TCV_slot := slot, TCV_tsc := tsc, TCV_Ccd0H := CntrlChDscrp(att, babr, C_cch_1Comb, bpm, t3212))			
5		+MM_PwrOrSimOff (C_SIMneedRmv)			
6		+Varinit_fix_R (C_CellA, lac, subch, cksn, rand, arfcngsm, TCV_Ccd0H, imsi)			
7		+ltree_chconf(C_cch_1Comb)			Config LT
8		+localtree1			
		localtree1			
9		+ltree_sysinfo(1, C_cch_1Comb)			ATT= '1'B
10		+MM_PwrOrSimOn (C_SIMneedRmv)			
11		+localtree_Lup_Auth			
		localtree_Lup_Auth			
12		START T_dly(C_T_Wait1stChReq)			
13		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
14		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
15		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, subch, slot, tsc, TCV_chdescr_arfcn, ta))		
16		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(C_norm_period_attach))		Any location update type
17		ACTIVATE(OtherEvents)			Restore Normal default
18		+Authentication(TCV_ch, cksn, rand)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
19		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiTmsi_01iei, mcc, mnc, TCV_lac))		Location Accept with TMSI
20		L?DL_DatInTmsireCom	TmsiReallocCmp(TCV_ch)		
21		+PostMainLinkRel(TCV_ch)			Release Ch
22		+localtree2			
23		?TIMEOUT T_dly			
24		+ltree_rmvPLMNfbdnlist			
		localtree2			
25		+localtree3			
26		+ltree_PgReorg			
		localtree3			
27		(TCV_Ccd0H := CntrlChDscrp(att, babr, C_cch_1nonComb, bpm, t3212))			
28		+Varinit_fix_R(C_CellA, lac, subch, cksn, rand, arfcngsm, TCV_Ccd0H, imsi)			
29		+ltree_chconf (C_cch_1nonComb)			Config LT
30		+ltree_sysinfo(att, C_cch_1nonComb)			
		ltree_PgReorg			
31		(TCV_Null:=OM_PgFill(C_CellA, PgReqTp1Reorg))			
32		START T_dly(5000)			
33		?TIMEOUT T_dly			
34		(TCV_Null:=OM_PgFill(C_CellA, PgReqTp1Norm))			
		ltree_sysinfo(attach : INTEGER; con_ch : B_3)			
35		+SysInfoSending(attach, con_ch)			SI1−4, cellA
36		(TCV_Ccd0A := TCV_Ccd0H, TCV_sysinfo5 := TCV_sysinf5, TCV_sysinfo6 := TCV_sysinf6)			
		SysInfoSending(attachb : INTEGER; con_chs : B_3)			
37		[sysinfonb = C_Test_fh]			Hopping
38		+ SysInfoSending_fh_R (sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, cchdgsm, crh, mtmcgsm, bcchflgsm, Re, bcc, ncc, nccp)			
39		[sysinfonb = C_Test_rgsm]			EGSM
40		+SysInfoSending_e (sch, bcch, t, retr, neci, attachb, babr, con_chs, bpm, t3212, ci, mcc, mnc, lac, co, cchdgsm, crh, mtmcgsm, bcchflgsm, bcchflegsm, Re, bcc, ncc, nccp)			
		ltree_chconf(conf : B_3)			
41		+ltree_chconf_A(conf)			Config LT
		ltree_chconf_A(conf_a : B_3)			
42		[conf_a = C_cch_1nonComb]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
43		+Config_BCCH_CCCH_A_1(par_bspwr, mtmcgsm, ChMod_sign, acttype, slot, tsc, FreqBCCH(arfcngsm), C_FCCH_SCH_BCCH_CCCH, ta, babr, conf_a, bpm)			1 non combined CCCH, cell A
44		(TCV_Upd := C_NotCombined)			
45		[conf_a = C_cch_1Comb]			
46		[(conf_a <> C_cch_1nonComb) AND (conf_a <> C_cch_1Comb)]		I	A wrong CCCH−CONF value, cell A
		ltree_rmvPLMNfbdnlist			
47		(TCV_Null := OO_PLMNselModeMan())			
48		(TCV_Null := OO_SelPLMN(mnc))			
49		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
50		ACTIVATE(OtherEvents_02)			To match ChReq retrans.
51		L!DL_UdatRqImmass	ImmAss(TCV_agch, ImmAsgn_01Def(TCV_Rr, TCV_Fn, subch, slot, tsc, TCV_chdescr_arfcn, ta))		
52		L?DL_EstInLupRq	LocUp(TCV_ch, LocUpdtReq_01(C_norm_period_attach))		Any location update type
53		ACTIVATE(OtherEvents)			Restore Normal default
54		L!DL_DatRqLupAcp	LocAcp(TCV_ch, LocUpdtAcp_01(MiTmsi_r02iei, mcc, mnc, TCV_lac))		Location Accept with no TMSI
55		+PostMainLinkRel(TCV_ch)			Release Ch
56		(TCV_Null := OO_PLMNselModeAuto())			
57		(TCV_Null := OM_StopCell(C_CellA))			
58		(TCV_Cnt2 := 1)			2nd time

Detailed Comments : Formal Parameter Type Comments & Explanations Possible values

par_bspwr INTEGER Base station power level C_E_default, C_E_suitable

C_CellA CellID Activated cell identity C_CellA, C_CellB, C_CellC

sch LOGICCH Sync. channel identity
If formal para C_CellA = C_CellA, C_SCH_A
If formal para C_CellA = C_CellB, C_SCH_B
If formal para C_CellA = C_CellC, C_SCH_C

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

bcch LOGICCH 1st broadcast channel id
If formal para C_CellA = C_CellA, C_BCCH_A_1
If formal para C_CellA = C_CellB, C_BCCH_B_1
If formal para C_CellA = 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

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

C_cch_1nonComb 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

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

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 C_CellA = C_CellA C_ci_CellA
If formal para C_CellA = C_CellB C_ci_CellB
If formal para C_CellA = 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.
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

C_Start_Tmsi 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

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

subch BITSTRING Selected sub−channel
If Formal para C_cch_1nonComb = 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 C_CellA = C_CellA
C_arfcnA_x If C_CellA = C_CellB C_arfcnB_x If
C_CellA = C_CellC C_arfcnC_x

arfcndcs INTEGER BCCH channel ARFCN (DCS)
If sysinfonb = C_Test_fh C_Noarfcn
Otherwise: If C_CellA = C_CellA
C_arfcnAd_x If C_CellA = C_CellB C_arfcnBd_x If
C_CellA = 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_rgsm
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 : PreEnterCCstateU10_r01_R (Ta : TA; sub : B_2; powerlevelg : INTEGER; cksn : B_3; rand : RAND; slot : SN; tsc : TSC; tch : B_1; rate : IA5String)

Group : Preambles/

Objective : To establish a mobile terminating call for the supported bearer capability and put the MS in the CC state U10.

Default : OtherEvents

Comments : The supported bearer capability is specified in the input partameter setup, test case variable TCV_TI holds the transaction ID and the transaction ID =0. This is generic call set up procedure. The calling tree shall prepare three variables for the step: TCV_ch for SDCCH4 subchannel, TCV_chTch for traffic channel, TCV_CphKey for ciphering key. The timing advance is parameter. For 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, sub, TCV_slot, TCV_tsc, TCV_chdescr_arfcn, Ta))		
5		L?DL_EstInPgRes	PagingRes(PagingRes_01)		
6		ACTIVATE(OtherEvents)			Restore Normal default
7		L!DL_DatRqAuthRq	AuthReq(TCV_ch, AuthRequest(cksn, rand))		
8		L?DL_DatInAuthRes (TCV_Sres := DL_DatInAuthRes.msg.sres)	AuthRes(AuthResponse)		
9		(TCV_Res := OC_ChkSRES(TCV_Sres, TSPX_Ki, rand))			
10		[NOT TCV_Res]		I	1.
11		[TCV_Res]			
12		+Ciphering_on(TCV_ch)			
13		+localtree			
		localtree			
14		(TCV_Setup_mt.sig := Signal_01)			
15		L!DL_DatRqSetup (TCV_TI.ti_f := '0'B, TCV_TI.ti_v := '000'B)	SetupSnd(TCV_ch, TCV_Setup_mt)		
16		L?DL_DatInCallCo (TCV_CallCfm:=DL_DatInCallCo.msg)	CallCfm(CallConfirm_01(TI_01))		
17		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
18		+localtree1			
19		L?DL_DatInAlert	AlertRcv(AlertingInd_02(TI_01))		
20		(TCV_Null := OO_HookOff())			
21		L?DL_DatInConn	ConnRcv(Connect_01(TI_01))		
22		+localtree1			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
		localtree1			
23		+AssCmdGen(TCV_cellid, rate, slot, tsc, tch)			
24		+Adjust_gsmanddcs_powerlvl_R(powerlevelg)			
25		+AssCh_complete(TCV_ch, TCV_chTch, TCV_AssCmd)			
26		L!DL_DatRqConnAck	ConnAckSnd(TCV_chTch, ConnectAck_01(TI_02))		
27		+CCstatuschk_02(TCV_chTch, C_U10, TI_02, TI_01)			

Detailed Comments : 1. Authentication fails, inconclusive.

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 : StartCellB_R (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 : CCHD; bcchflgsm : NCD; Re : BITSTRING; bcc : BCC; ncc : NCC)

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_R(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, C_CellReselectHys12, C_MaxPwrLvlG, bcchflgsm, 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		[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	
		ltree_chconf			
9		[cch_con = C_cch_1Comb]			
10		+CombinedBCCH_B_R(par_bspwr, C_MaxPwrLvlG, arfcn1, acttype, slot, tsc, ta, babr, cch_con, bpm)			
11		[cch_con = C_cch_1nonComb]			
12		+NonCombinedBCCH_B_R(par_bspwr, C_MaxPwrLvlG, arfcn1, acttype, slot, tsc, ta, babr, cch_con, bpm)			
13		[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : StartCellB_rgsm (par_bspwr, arfcn1 : 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 : CCHD; bcchfl1, bcchfl2 : NCD; Re : BITSTRING; bcc : BCC; ncc : NCC)

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 EGSM 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_r(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, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl1, bcchfl2, Re, bcc, ncc, C_NCCP_2)			
4		(TCV_Ccd0B := TCV_Ccd0H, TCV_sysinfo5_B := TCV_sysinf5, TCV_sysinfo5bis_B := TCV_sysinf5bis, 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		[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	
		ltree_chconf			
9		[cch_con = C_cch_1Comb]			
10		+CombinedBCCH_B_R(par_bspwr, C_MaxPwrLvlG, arfcn1, acttype, slot, tsc, ta, babr, cch_con, bpm)			
11		[cch_con = C_cch_1nonComb]			
12		+NonCombinedBCCH_B_R (par_bspwr, C_MaxPwrLvlG, arfcn1, acttype, slot, tsc, ta, babr, cch_con, bpm)			
13		[(cch_con <> C_cch_1nonComb) AND (cch_con <> C_cch_1Comb)]		I	

Detailed Comments :

Test Step Dynamic Behaviour

Test Step Name : StartMultiCells_02_R (bcchfl_gsm : NCD; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; att : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, mnc, lacB, lacC, lacD, lacE, lacF, lacG, lacH : OCTETSTRING)

Group : Preambles/

Objective : To broadcast SYSTEM INFORMATION messages 1, 2, 3, 4, 5 and 6 in multiple cells for measurement testing. Neighbour cells description for cell S1 is a formal parameter.

Default : OtherEvents

Comments : The parameters for SYSTEM INFORMATION messages are defined in section 26.6.3 of GSM 11.10

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+cellN1			
2		+cellN2			
3		+cellN3			
4		+cellN4			
5		+cellN5			
6		+cellN6			
7		+cellN7			
		cellN1			
8		+CombinedBCCH_B_R (28, C_MaxPwrLvlG, 8, acttype, slot, tsc, ta, babr, cch_con, bpm)			
9		+SysInfoSending_fh_R(C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellB, mcc, mnc, lacB, CellOpt_01, CellChDes_05, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC, C_NCC, C_NCCP_2)			
10		(TCV_Ccd0B := TCV_Ccd0H)			
		cellN2			
11		+CombinedBCCH_C_R (33, C_MaxPwrLvlG, 14, acttype, slot, tsc, ta, babr, cch_con, bpm)			
12		+SysInfoSending_fh_R(C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellC, mcc, mnc, lacC, CellOpt_01, CellChDes_06, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_7, C_NCC, C_NCCP_2)			
13		(TCV_Ccd0C := TCV_Ccd0H)			
		cellN3			
14		+CombinedBCCH_D_R (38, C_MaxPwrLvlG, 20, acttype, slot, tsc, ta, babr, cch_con, bpm)			
15		+SysInfoSending_fh_R(C_SCH_D, C_BCCH_D_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellD, mcc, mnc, lacD, CellOpt_01, CellChDes_07, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_1, C_NCC, C_NCCP_2)			ncc = 1
16		(TCV_Ccd0D := TCV_Ccd0H)			
		cellN4			
17		+CombinedBCCH_E_R (58, C_MaxPwrLvlG, 26, acttype, slot, tsc, ta, babr, cch_con, bpm)			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
18		+SysInfoSending_fh_R(C_SCH_E, C_BCCH_E_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellE, mcc, mnc, lacE, CellOpt_01, CellChDes_08, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_3, C_NCC, C_NCCP_2)			ncc = 1
19		(TCV_Ccd0E := TCV_Ccd0H)			
		cellN5			
20		+CombinedBCCH_F_R (63, C_MaxPwrLvlG, 32, acttype, slot, tsc, ta, babr, cch_con, bpm)			
21		+SysInfoSending_fh_R(C_SCH_F, C_BCCH_F_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellF, mcc, mnc, lacF, CellOpt_01, CellChDes_09, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC, C_NCC, C_NCCP_2)			ncc = 1
22		(TCV_Ccd0F := TCV_Ccd0H)			
		cellN6			
23		+CombinedBCCH_G_R (68, C_MaxPwrLvlG, 38, acttype, slot, tsc, ta, babr, cch_con, bpm)			
24		+SysInfoSending_fh_R (C_SCH_G, C_BCCH_G_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellG, mcc, mnc, lacG, CellOpt_01, CellChDes_10, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_7, C_NCC, C_NCCP_2)			
25		(TCV_Ccd0G := TCV_Ccd0H)			
		cellN7			
26		+CombinedBCCH_H_R (73, C_MaxPwrLvlG, 44, acttype, slot, tsc, ta, babr, cch_con, bpm)			
27		+SysInfoSending_fh_R(C_SCH_H, C_BCCH_H_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellH, mcc, mnc, lacH, CellOpt_01, CellChDes_11, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_1, C_NCC, C_NCCP_2)			
Detailed Comments :					

Test Step Dynamic Behaviour

Test Step Name : StartMultiCells_02r_R (bcchfl_gsm : NCD; acttype : BITSTRING; slot : SN; tsc : TSC; ta : TA; att : INTEGER; babr, cch_con, bpm : B_3; t3212, mcc, mnc, lacA, lacB, lacC, lacD, lacE, lacF, lacG, lacH : OCTETSTRING)

Group : Preambles/

Objective : To broadcast SYSTEM INFORMATION messages 1, 2, 3, 4, 5 and 6 in multiple cells for measurement testing. Neighbour cells description for cell S1 is a formal parameter.

Default : OtherEvents

Comments : The parameters for SYSTEM INFORMATION messages are defined in section 26.10.2 of GSM 11.10

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+cellN1			
2		+cellN2			
3		+cellN3			
4		+cellN4			
5		+cellN5			
6		+cellN6			
7		+cellN7			
		cellN1			
8		+CombinedBCCH_B_R (28, C_MaxPwrLvlG, C_arfcnEgsm, acttype, slot, tsc, ta, babr, cch_con, bpm)			
9		+SysInfoSending_fh_R (C_SCH_B, C_BCCH_B_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellB, mcc, mnc, lacB, CellOpt_01, CellChDes_05, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC, C_NCC, C_NCCP_2)			
10		(TCV_Ccd0B := TCV_Ccd0H)			
		cellN2			
11		+CombinedBCCH_C_R (33, C_MaxPwrLvlG, 970, acttype, slot, tsc, ta, babr, cch_con, bpm)			
12		+SysInfoSending_fh_R (C_SCH_C, C_BCCH_C_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellC, mcc, mnc, lacC, CellOpt_01, CellChDes_06, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_7, C_NCC, C_NCCP_2)			
13		(TCV_Ccd0C := TCV_Ccd0H)			
		cellN3			
14		+CombinedBCCH_D_R (38, C_MaxPwrLvlG, 0, acttype, slot, tsc, ta, babr, cch_con, bpm)			
15		+SysInfoSending_fh_R (C_SCH_D, C_BCCH_D_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellD, mcc, mnc, lacD, CellOpt_01, CellChDes_07, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_1, C_NCC, C_NCCP_2)			ncc = 1
16		(TCV_Ccd0D := TCV_Ccd0H)			
		cellN4			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
17		+CombinedBCCH_E_R (58, C_MaxPwrLvlG, 26, acttype, slot, tsc, ta, babr, cch_con, bpm)			
18		+SysInfoSending_fh_R (C_SCH_E, C_BCCH_E_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellE, mcc, mnc, lacE, CellOpt_01, CellChDes_08, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_3, C_NCC, C_NCCP_2)			ncc = 1
19		(TCV_Ccd0E := TCV_Ccd0H)			
		cellN5			
20		+CombinedBCCH_F_R (63, C_MaxPwrLvlG, 1020, acttype, slot, tsc, ta, babr, cch_con, bpm)			
21		+SysInfoSending_fh_R (C_SCH_F, C_BCCH_F_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellF, mcc, mnc, lacF, CellOpt_01, CellChDes_09, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC, C_NCC, C_NCCP_2)			ncc = 1
22		(TCV_Ccd0F := TCV_Ccd0H)			
		cellN6			
23		+CombinedBCCH_G_R (68, C_MaxPwrLvlG, 38, acttype, slot, tsc, ta, babr, cch_con, bpm)			
24		+SysInfoSending_fh_R (C_SCH_G, C_BCCH_G_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellG, mcc, mnc, lacG, CellOpt_01, CellChDes_10, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_7, C_NCC, C_NCCP_2)			
25		(TCV_Ccd0G := TCV_Ccd0H)			
		cellN7			
26		+CombinedBCCH_H_R (73, C_MaxPwrLvlG, 960, acttype, slot, tsc, ta, babr, cch_con, bpm)			
27		+SysInfoSending_fh_R (C_SCH_H, C_BCCH_H_1, C_TxInt_5, C_Max_1, C_NECI_0, att, babr, cch_con, bpm, t3212, C_ci_cellH, mcc, mnc, lacH, CellOpt_01, CellChDes_11, C_CellReselectHys12, C_MaxPwrLvlG, bcchfl_gsm, C_noRestablishment, C_BCC_1, C_NCC, C_NCCP_2)			
Detailed Comments :					

Test Step Dynamic Behaviour

Test Step Name : TS61MO(srv : SERVICES)
Group : Preambles/
Objective : To generate an MO SETUP message with appropriate Bcap for TS61 service.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_ur := TSPX_TS_61_ur1, TCV_CE := NT_BothT_BothNTorT_BothT_BothNT)			
2		+local_selectCe			
3		+local_bcap			get the right BC
4		(TCV_Setup_mo := Setup_11_3100(C_RI_alternate, Bcap_Speech_MO(C_Rchr_dontcare), TCV_BcapMO1, Llcmp_NotApplicable, Llcmp_AnyOrOmit, C_RI_alternate, C_RI_alternate, Hlcmp_NotApplicable, Hlcmp_TS61))			generate SETUP
5		+Chmod(TCV_ChRate, TCV_ce, TCV_ur, srv)			initiate channel mode variables
6		(TCV_Null := OO_IFsetup_TS6x_MO(TCV_srv, TCV_ur, TCV_ce, TCV_ChRate))			ask operator to configure the MS
		local_bcap			
7		[TCV_ce = C_transparent]			
8		[TCV_ur = C_9600bs]			
9		+local_TS61_FAX3_T(C_9600bs, C_ir_16kbs)			
10		[(TCV_ur = C_4800bs) OR (TCV_ur = C_2400bs)]			
11		+local_TS61_FAX3_T(TCV_ur, C_ir_8kbs)			
12		[(TCV_ur <> C_4800bs) AND (TCV_ur <> C_2400bs) AND (TCV_ur <> C_9600bs)]		I	IXIT parameter error
13		[TCV_ce = C_notransparent]			
14		+local_TS61_FAX3_NT(TCV_ur)			
		local_TS61_FAX3_T(ur : B_4; ir : B_2)			
15		(TCV_BcapMO1 := Bcap_MO(TCV_Rchr, C_FAX3, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, C_modemt_none))			11.8.2.7.1
		local_TS61_FAX3_NT(ur : B_4)			
16		[ur=C_9600bs]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
17		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_FAX3, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, TCV_CE.NTorBothTorBothNT, C_modemt_none, C_X25_flct))			11.8.2.7 .2
18		[(ur=C_2400bs) OR (ur = C_4800bs)]			
19		(TCV_BcapMO1 := BcapX_MO(TCV_Rchr, C_FAX3, C_SDUintegrity, C_nirr_dontcare, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, TCV_CE.NTorBothTorBothNT, C_modemt_none, C_X25_flct))			11.8.2.7 .2
20		[(ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)] local_selectCe		I	IXIT paramet er error
21		[TSPX_TS_61_T_NT]			
22		(TCV_ce := C_notransparent)			
23		[NOT TSPX_TS_61_T_NT]			
24		(TCV_ce := TSPX_TS_61_ce)			

Detailed Comments : The TS61 Bearer Capabilities and SETUP messages for MO is constructed according to section 11.8.2.7.1 and section 11.8.2.7.2 of GSM 11.10 and section B.1.10.2 and section B.2.10 of GSM 07.01.

The algorithm for selecting parameters:
IF (TSPX_BS_61_T_NT)
    TCV_ce := C_notransparent
ELSE
    TCV_ce := TSPX_BS_61_ce

use (TCV_ce, TSPX_BS_61_ur1, svc) for OO_IFsetup_TS6x_MO

Test Step Dynamic Behaviour

Test Step Name : TS62MO(srv : SERVICES)
Group : Preambles/
Objective : To generate an MO SETUP message with appropriate Bcap for TS62 service.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(TCV_srv := srv, TCV_ur := TSPX_TS_62_ur1, TCV_CE := NT_BothT_BothNTorT_BothT_BothNT)			variables used in InitCall
2		+local_selectCe			
3		+local_bcap			get the right BC
4		(TCV_Setup_mo := Setup_10_3100(TCV_BcapMO1, Llcmp_AnyOrOmit, Hlcmp_TS62))			
5		+Chmod(TCV_ChRate, TCV_ce, TCV_ur, srv)			
6		(TCV_Null := OO_IFsetup_TS6x_MO(TCV_srv, TCV_ur, TCV_ce, TCV_ChRate))			
		local_bcap			
7		[TCV_ce = C_transparent]			
8		[TCV_ur = C_9600bs]			
9		+local_TS62_FAX3_T(C_9600bs, C_ir_16kbs)			
10		[(TCV_ur = C_4800bs) OR (TCV_ur = C_2400bs)]			
11		+local_TS62_FAX3_T(TCV_ur, C_ir_8kbs)			
12		[(TCV_ur <> C_4800bs) AND (TCV_ur <> C_2400bs) AND (TCV_ur <> C_9600bs)]		I	IXIT paramet er error
13		[TCV_ce = C_notransparent]			
14		+local_TS62_FAX3_NT(TCV_ur)			
		local_TS62_FAX3_T(ur : B_4; ir : B_2)			
15		(TCV_BcapMO1 := Bcap_MO(C_Rchr_dontcare, C_FAX3, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, C_modemt_none))			11.8.2.7 .1
		local_TS62_FAX3_NT(ur : B_4)			
16		[ur = C_9600bs]			
17		(TCV_BcapMO1 := BcapX_MO(C_Rchr_dontcare, C_FAX3, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, TCV_CE.NTorBothTorBothNT, C_modemt_none, C_X25_flct))			11.8.2.7 .2
18		[(ur = C_2400bs) OR (ur = C_4800bs)]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
19		(TCV_BcapMO1 := BcapX_MO(C_Rchr_dontcare, C_FAX3, C_SDUintegrity, C_nirr_dontcare, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, TCV_CE.NTorBothTorBothNT, C_modemt_none, C_X25_flct))			11.8.2.7.2
20		[(ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)] local_selectCe		I	IXIT parameter error
21		[TSPX_TS_62_T_NT]			
22		(TCV_ce := C_notransparent)			
23		[NOT TSPX_TS_62_T_NT]			
24		(TCV_ce := TSPX_TS_62_ce)			

Detailed Comments : The TS62 Bearer Capabilities and SETUP messages for MO is constructed according to section 11.8.2.7.1 and section 11.8.2.7.2 of GSM 11.10 and section B.1.10.2 and section B.2.11 of GSM 07.01.

The algorithm for selecting parameters:

```
IF (TSPX_BS_62_T_NT)
        TCV_ce := C_notransparent
ELSE
        TCV_ce := TSPX_BS_62_ce

use (TCV_ce, TSPX_BS_62_ur1, svc) for OO_IFsetup_TS6x_MO
```

Test Step Dynamic Behaviour

Test Step Name : TS61MT(srv : SERVICES)
Group : Preambles/
Objective : To generate an MT SETUP message with appropriate Bcap for TS61 service.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_maintree			
2		+local_IFsetup			
		local_maintree			
3		(TCV_srv := srv, TCV_2ndtest := C_Yes)			
4		(TCV_BcapMT1 := Bcap_Speech_MT)			1st Bcap
5		+local_first_SETUP_msg			
6		+local_second_SETUP_msg			
		local_first_SETUP_msg			
7		(TCV_ur := TSPX_TS_61_ur1)			
8		[TSPX_TS_61_T_NT OR (TSPX_TS_61_ce = C_notransparent)]			
9		+local_bcap2(TCV_ur, C_notransparent)			
10		(TCV_Setup_mt := Setup_21(C_RI_alternate, TCV_BcapMT1, TCV_BcapMT2, TI_02))			
11		+Chmod(TCV_ChRate, C_notransparent, TCV_ur, srv)			2nd channel mode
12		(TCV_ChMod.mode := C_ChMod_r)			Speech mode
13		[NOT TSPX_TS_61_T_NT AND (TSPX_TS_61_ce = C_transparent)]			
14		+local_bcap2(TCV_ur, C_transparent)			
15		(TCV_Setup_mt := Setup_21(C_RI_alternate, TCV_BcapMT1, TCV_BcapMT2, TI_02))			
16		+Chmod(TCV_ChRate, C_transparent, TCV_ur, srv)			2nd channel mode
17		(TCV_ChMod.mode := C_ChMod_r)			Speech mode
		local_second_SETUP_msg			
18		+local_userrate			
19		[TSPX_TS_61_T_NT]			
20		+local_bcap2(TCV_ur2, C_transparent)			2nd Bcap
21		(TCV_Setup_mt1 := Setup_21(C_RI_alternate, TCV_BcapMT1, TCV_BcapMT2, TI_02))			
22		[NOT TSPX_TS_61_T_NT]			
23		+local_bcap2(TCV_ur2, C_BothT)			
24		(TCV_Setup_mt1 := Setup_21(C_RI_alternate, TCV_BcapMT1, TCV_BcapMT2, TI_02))			
		local_bcap2(ur : B_4; ce : B_2)			
25		[ce = C_transparent]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
26		[(ur = C_2400bs) OR (ur = C_4800bs)]			
27		+local_TS61_FAX3_T(ur, C_ir_8kbs)			
28		[ur = C_9600bs]			
29		+local_TS61_FAX3_T(ur, C_ir_16kbs)			
30		[(ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
31		[ce = C_BothT]			
32		+local_TS61_FAX3_NT(ur, C_BothT)			
33		[ce = C_notransparent]			
34		+local_TS61_FAX3_NT(ur, C_notransparent)			
35		[(ce <> C_transparent) AND (ce <> C_BothT) AND (ce<> C_notransparent)]		I	
		local_TS61_FAX3_T(ur : B_4; ir : B_2)			
36		(TCV_BcapMT2 := Bcap_MT(C_FAX3, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, C_modemt_none))			11.8.1.5 .1
		local_TS61_FAX3_NT(ur : B_4; ce1 : B_2)			
37		(TCV_BcapMT2 := Bcap_MT(C_FAX3, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, ce1, C_modemt_none))			11.8.1.5 .2
		local_userrate			
38		[TSPX_TS_61_more_ur]			
39		(TCV_ur2 := TSPX_TS_61_ur2)			
40		[NOT TSPX_TS_61_more_ur]			
41		(TCV_ur2 := TSPX_TS_61_ur1)			
		local_IFsetup			
42		[TCV_supported]			
43		(TCV_Null := OO_IFsetup_TS6x_MT(TCV_srv, TCV_ur))			
44		[NOT TCV_supported]			
45		[TCV_SpecialCase]			
46		START T_dly(C_T_Wait)			
47		?TIMEOUT T_dly			
48		[NOT TCV_SpecialCase]		I	Paramet er error

Detailed Comments : The algorithm for derivation of TS61 Bearer Capabilities:

The 1st facsimile group 3 Bearer Capability (used for an MT call):
ur= TSPX_TS_61_ur1
IF TSPX_TS_61_T_NT OR (TSPX_TS_61_ce = NT)
use coding 11.8.1.5.2 with (ur & ce=NT)
ELSE
use coding 11.8.1.5.1 with (ur & ce=T)

The 2nd facsimile group 3 Bearer Capabiility (used for TC_11_1_1):
IF TSPX_TS_61_more_ur

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Detailed Comments : ...

ur=TSPX_TS_61_ur2
ELSE
ur=TSPX_TS_61_ur1
IF TSPX_TS_61_T_NT
use coding 11.8.1.5.1 with (ur & ce=T)
ELSE
use coding 11.8.1.5.2 with(ur & ce=both/T)

Test Step Dynamic Behaviour

Test Step Name : TS62MT(srv : SERVICES)
Group : Preambles/
Objective : To generate an MT SETUP message with appropriate Bcap for TS62 service.
Default : OtherEvents
Comments :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+local_maintree			
2		(TCV_Null := OO_IFsetup_TS6x_MT(TCV_srv, TCV_ur))			
		local_maintree			
3		(TCV_srv := srv, TCV_2ndtest := C_Yes)			
4		+local_first_SETUP_msg			
5		+local_second_SETUP_msg			
		local_first_SETUP_msg			
6		(TCV_ur := TSPX_TS_62_ur1)			
7		[TSPX_TS_62_T_NT OR (TSPX_TS_62_ce = C_notransparent)]			
8		+local_bcap2(TCV_ur, C_notransparent)			
9		(TCV_Setup_mt := Setup_20(TCV_BcapMT2, TI_02))			
10		+Chmod(TCV_ChRate, C_notransparent, TCV_ur, srv)			
11		[NOT TSPX_TS_62_T_NT AND (TSPX_TS_62_ce = C_transparent)]			
12		+local_bcap2(TCV_ur, C_transparent)			
13		(TCV_Setup_mt := Setup_20(TCV_BcapMT2, TI_02))			
14		+Chmod(TCV_ChRate, C_transparent, TCV_ur, srv)			
		local_second_SETUP_msg			
15		+local_userrate			
16		[TSPX_TS_62_T_NT]			
17		+local_bcap2(TCV_ur2, C_transparent)			
18		(TCV_Setup_mt1 := Setup_20(TCV_BcapMT2, TI_02))			
19		[NOT TSPX_TS_62_T_NT]			
20		+local_bcap2(TCV_ur2, C_BothT)			
21		(TCV_Setup_mt1 := Setup_20(TCV_BcapMT2, TI_02))			
		local_bcap2(ur : B_4; ce : B_2)			
22		[ce = C_transparent]			
23		[(ur = C_2400bs) OR (ur = C_4800bs)]			
24		+local_TS62_FAX3_T(ur, C_ir_8kbs)			
25		[ur = C_9600bs]			
26		+local_TS62_FAX3_T(ur, C_ir_16kbs)			
27		[(ur <> C_2400bs) AND (ur <> C_4800bs) AND (ur <> C_9600bs)]		I	
28		[ce = C_BothT]			
29		+local_TS62_FAX3_NT(ur, C_BothNT)			
30		[ce = C_notransparent]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
31		+local_TS62_FAX3_NT(ur, C_notransparent)			
32		[(ce <> C_transparent) AND (ce <> C_BothT) AND (ce<> C_notransparent)]		I	
		local_TS62_FAX3_T(ur : B_4; ir : B_2)			
33		(TCV_BcapMT2 := Bcap_MT(C_FAX3, C_Unstructured, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, ir, C_parity_NA, C_transparent, C_modemt_none))			11.8.1.5 .1
		local_TS62_FAX3_NT(ur : B_4; ce1 : B_2)			
34		(TCV_BcapMT2 := Bcap_MT(C_FAX3, C_SDUintegrity, C_nirr_nomeaning, C_no_rate_adaption, C_sacp_NA, C_Synchronous, C_stopbit_NA, C_databit_NA, ur, C_ir_16kbs, C_parity_NA, ce1, C_modemt_none))			11.8.1.5 .2
		local_userrate			
35		[TSPX_TS_62_more_ur]			
36		(TCV_ur2 := TSPX_TS_62_ur2)			
37		[NOT TSPX_TS_62_more_ur]			
38		(TCV_ur2 := TSPX_TS_62_ur1)			
		local_IFsetup			
39		[TCV_supported]			
40		(TCV_Null := OO_IFsetup_TS6x_MT(TCV_srv, TCV_ur))			
41		[NOT TCV_supported]			
42		[TCV_SpecialCase]			
43		START T_dly(C_T_Wait)			
44		?TIMEOUT T_dly			
45		[NOT TCV_SpecialCase]		I	Paramet er error

Detailed Comments : The algorithm for derivation of TS62 Bearer Capabilities:

The 1st Bearer Capability (used for an MT call):
ur= TSPX_TS_62_ur1
IF TSPX_TS_62_T_NT OR (TSPX_TS_62_ce = NT)
use coding 11.8.1.5.2 with (ur & ce=NT)
ELSE
use coding 11.8.1.5.1 with (ur & ce=T)

The 2nd Bearer Capabiility (used for TC_11_1_1):
IF TSPX_TS_62_more_ur
ur=TSPX_TS_62_ur2
ELSE
ur=TSPX_TS_62_ur1
IF TSPX_TS_62_T_NT
use coding 11.8.1.5.1 with (ur & ce=T)
ELSE
use coding 11.8.1.5.2 with(ur & ce=both/T)

Default Dynamic Behaviour

Default Name : OtherEvents
Group :
Objective : To match unexpected events and asign a 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		CANCEL		I	3.
8		L?OTHERWISE			2.
9		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		CANCEL		I	3.
8		L?OTHERWISE			2.
9		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		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		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		CANCEL		I	4.
10		L?OTHERWISE			3.
11		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_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		CANCEL		I	4.
10		L?OTHERWISE			3.
11		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 :