

# I

## Test Suite Overview

Test Suite Structure			
<b>Suite Name</b> : CFNRc_N_v001 <b>Standards Ref</b> : EN 302 094–1, EN 301 144–1 <b>PICS Ref</b> : EN 302 094–2, EN 301 144–2 <b>PIXIT Ref</b> : EN 302 094–6 <b>Test Method(s)</b> : Remote single layer test method <b>Comments</b> : Applicable to both Basic and Primary Rate Access.			
Test Group Reference	Selection Ref	Test Group Objective	Page Nr
ServedUser/	CFNRc_Network		154
ServedUser/Registration/			154
ServedUser/Erasure/			156
ServedUser/Activation/			158
ServedUser/Deactivation/			160
ServedUser/Interrogation/			162
ServedUser/Operation/			164
ServedUser/ReminderNotification/			167
CallingUser/			168
CallingUser/NotifDiv/			168
CallingUser/IdentificationDivertedTo /			170
DivertedToUser/	CFNRc_DivertedTo		171
<b>Detailed Comments</b> :			

Test Case Index				
Test Group Reference	Test Case Id	Selection Ref	Description	Page Nr
ServedUser/Registration/	CFNRc_N01_001			154
ServedUser/Erasure/	CFNRc_N02_001			156
ServedUser/Activation/	CFNRc_N03_001			158
ServedUser/Deactivation/	CFNRc_N04_001			160
ServedUser/Interrogation/	CFNRc_N05_001			162
ServedUser/Operation/	CFNRc_N06_001			164
ServedUser/Operation/	CFNRc_N06_002			165
ServedUser/Operation/	CFNRc_N06_003			166
ServedUser/ReminderNotification/	CFNRc_N07_001			167
CallingUser/NotifDiv/	CFNRc_N08_001			CFNRc_notif_calling
CallingUser/Identification DivertedTo/	CFNRc_N09_001	CFNRc_Network	170	
DivertedToUser/	CFNRc_N10_001			171
DivertedToUser/	CFNRc_N10_002			172
Detailed Comments :				

Test Step Index			
Test Step Group Reference	Test Step Id	Description	Page Nr
Preambles/	PR_N00	Preamble to the Null call state N00 to be used with non-NCICS connections.	173
Preambles/	PR_N00_Init	Preamble to the Null call state N00 with subscription registration (if supported) and location registration.	174
Preambles/	PR_N04_Init	Preamble to the Null call state N00 with subscription registration (if supported) and location registration.	175
Steps/	ST_Activate	To activate IUT to CFNRc SS.	176
Steps/	ST_CTMAccessRightRequest	To proceed to the subscription registration with the IUT.	177
Steps/	ST_CTMLocationRegistration	To proceed to the location registration procedure with the IUT.	179
Steps/	ST_Erase	To register IUT to CFNRc SS.	181
Steps/	ST_GSMLocationRegistration	Performing the GSM Location Registration procedure, accounting for possible embedded procedures	182
Steps/	ST_GSM_EMB_PROCS	Handling of embedded procedures following a Location Registration request	183
Steps/	ST_N02_CTMOutgoing	To bring IUT to state N02 for a CTM outgoing call.	185
Steps/	ST_Receive_SU_Incoming		186
Steps/	ST_Register	To register IUT to CFNRc SS.	187
Steps/	ST_Stop_PTCs		188
Steps/	ST_U06_To_U10	To complete the establishment of an outgoing NCICS call. Bring IUT from U06 to U10.	188
PTC1_Steps/	PO1_IC_OC	To bring the IUT back to the Null call state U00.	189
PTC1_Steps/	PO1_IC_OC_Stop	To bring the IUT back to the Null call state U00 and stop the test.	189
PTC1_Steps/	PO1_NCICS	To bring the IUT back to the Null call state U00 (for an NCICS).	190
PTC1_Steps/	PR1_N00	Preamble to the Null call state N00 to be used with non-NCICS connections for PTC1.	191
PTC1_Steps/	PR1_N00_Init	Preamble to the Null call state N00 with subscription registration (if supported) and location registration.	192
PTC1_Steps/	ST1_Activate	To activate IUT to CFNRc SS.	193

Continued on next page

Continued from previous page

Test Step Index			
Test Step Group Reference	Test Step Id	Description	Page Nr
PTC1_Steps/	ST1_CTMAccessRightRequest	To proceed to the subscription registration with the IUT on PTC1.	194
PTC1_Steps/	ST1_CTMLocationRegistration	To proceed to the location registration procedure with the IUT on PTC1.	196
PTC1_Steps/	ST1_GSM_EMB_PROCS	Handling of embedded procedures following a Location Registration request	198
PTC1_Steps/	ST1_GSMLocationRegistration	Performing the GSM Location Registration procedure, accounting for possible embedded procedures on PTC1	200
PTC1_Steps/	ST1_N02_CTMOutgoing	To bring IUT to state N02 for a CTM outgoing call.	201
PTC1_Steps/	ST1_PTC1_Activate		202
PTC1_Steps/	ST1_Receive_SU_Incoming		203
PTC1_Steps/	ST1_Register	To register IUT to CFNRc SS.	204
PTC1_Steps/	ST1_U06_To_U10	To complete the establishment of an outgoing NCICS call. Bring IUT from U06 to U10 on PTC1.	205
PTC2_Steps/	PO2_NCICS	To bring the IUT back to the Null call state U00 (for an NCICS).	205
PTC2_Steps/	PR2_N00	Preamble to the Null call state N00 to be used with non-NCICS connections for PTC2.	206
PTC2_Steps/	PR2_N00_Init	Preamble to the Null call state N00 with subscription registration (if supported) and location registration.	207
PTC2_Steps/	ST2_CTMAccessRightRequest	To proceed to the subscription registration with the IUT on PTC2.	208
PTC2_Steps/	ST2_CTMLocationRegistration	To proceed to the location registration procedure with the IUT on PTC2.	210
PTC2_Steps/	ST2_GSM_EMB_PROCS	Handling of embedded procedures following a Location Registration request	212
PTC2_Steps/	ST2_GSMLocationRegistration	Performing the GSM Location Registration procedure, accounting for possible embedded procedures on PTC2	214
PTC2_Steps/	ST2_PTC_IN		215
PTC2_Steps/	ST2_PTC2_IN		216
PTC2_Steps/	ST2_PTC4_IN		217

Continued on next page

Continued from previous page

Test Step Index			
Test Step Group Reference	Test Step Id	Description	Page Nr
PTC2_Steps/	ST2_Receive_SU_Incoming		218
PTC2_Steps/	ST2_U06_To_U10	To complete the establishment of an outgoing NCICS call. Bring IUT from U06 to U10 on PTC2.	219
Postambles/	PO_IC_OC	To bring the IUT back to the Null call state U00.	219
Postambles/	PO_IC_OC_Stop	To bring the IUT back to the Null call state U00 and stop the test.	220
Postambles/	PO_NCICS	To bring the IUT back to the Null call state U00 (for an NCICS).	220
Detailed Comments :			

Default Index			
Default Group Reference	Default Id	Description	Page Nr
	DEF_CALL	Default subtree for outgoing call and incoming call test cases.	221
	DEF_NCICS	Default behaviour for NCICS calls.	223
	DEF1_NCICS	Default behaviour for NCICS calls.	225
	DEF2_NCICS	Default behaviour for NCICS calls.	226
	DEF1_CALL	Same as Default from basic call	227
	DEF2_CALL	Same as Default from basic call	228
Detailed Comments :			

## **II**

### **Declarations Part**



Simple Type Definitions			
Type Name	Type Definition	Type Encoding	Comments
ANY_MT_LIST	BITSTRING ( '0000001'B,'00000010'B,'0 0000111'B,'00001111'B,'00 000011'B,'00000101'B,'000 01101'B,'00100110'B,'0010 1110'B,'00100010'B,'00100 101'B,'00101101'B,'001000 01'B,'00100000'B,'0100010 1'B,'01000110'B,'01001101' B,'01001110'B,'01011010'B , '01100010'B,'01101101'B,' 01101110'B,'01110101'B,'0 1111001'B,'01111011'B,'01 111101'B,'00100100'B,'001 01000'B,'00110000'B,'0011 0001'B,'00110011'B,'00110 111'B,'01100100'B)		BITSTRING[8] any messages that may be rece ived and ignored. RESTART is not included here. HOLD/RETRIEVE messages are included, also FACILITY and REGISTER.BCAP
BCAP_I	BITSTRING('00000100'B)		Bearer capability identifier type
BIT7OR15	BITSTRING[7 .. 15]		Used in Call Reference
CAU_I	BITSTRING('00001000'B)		Cause identifier type
CDPN_I	BITSTRING('01110000'B)		Called party number identifier type
CDPS_I	BITSTRING('01110001'B)		Called party subaddress identifier type
CGPN_I	BITSTRING('01101100'B)		Calling party number identifier type
CGPS_I	BITSTRING('01101101'B)		Calling party subaddress identifier type
CHI_I	BITSTRING('00011000'B)		Channel id identifier type
CODN_I	BITSTRING('01001100'B)		Connected number identifier type
CODS_I	BITSTRING('01001101'B)		Spare Connected subaddress identifier type
DATI_I	BITSTRING('00101001'B)		Date/time identifier type
DSP_I	BITSTRING('00101000'B)		Display identifier type
EFAC_I	BITSTRING ('00001101'B)		Extended Facility id type
FAC_I	BITSTRING('00011100'B)		Facility identifier type
FAC_PP_INV	OCTETSTRING('91A1'O)		Protocol profile, invoke component tag

Continued on next page

Continued from previous page

Simple Type Definitions			
Type Name	Type Definition	Type Encoding	Comments
GFP_MT_LIST	BITSTRING ( '00100100'B, '00101000'B, '00110000'B, '00110001'B, '00110011'B, '00110111'B, '01100100'B)		ETS 300 196, subclause 11 HOLD HOLD_ACK HOLD_REJ RETRIEVE RETRIEVE_ACK RETRIEVE_REJ REGISTER
HLC_I	BITSTRING('01111101'B)		High layer compatibility identifier type
IE_LIST	OCTETSTRING[0..255]		Any sequence of information elements
KPF_I	BITSTRING('00101100'B)		Keypad facility identifier type
LLC_I	BITSTRING('01111100'B)		Low layer compatibility identifier type
LOCK_SHIFT_TYPE	OCTETSTRING [1]		locking shift type
MT	BITSTRING[8]		Message Type
NSF_I	BITSTRING('00100000'B)		Network-specific fac. identifier type
NOID_I	BITSTRING('00100111'B)		Notification indicator identifier type
PD	BITSTRING('00001000'B)		Protocol discriminator
PI_I	BITSTRING('00011110'B)		Progress indicator identifier type
RI_I	BITSTRING('01111001'B)		Restart indicator identifier type
RNGN_I	BITSTRING('01110100'B)		Redirecting number identifier type
RONN_I	BITSTRING('01110110'B)		Redirection number identifier type
SCI	BITSTRING('10100001'B)		Sending complete information
TNS_I	BITSTRING('01111000'B)		Transit network selection identifier type
UUI_I	BITSTRING('01111110'B)		User-user identifier type
Detailed Comments :			

Structured Type Definition			
<b>Type Name</b> : BCAP (BEARER CAPABILITY IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Bearer CAPability EN 300 403-1 & Q.931 subclause 4.5.5			
Element Name	Type Definition	Field Encoding	Comments
bcap_i	BCAP_I		Identifier
bcap_l	OCTETSTRING[1]		Length
bcap_oct3	BITSTRING[8]		Contents of the octet 3 of the bearer capability information element
bcap_con	OCTETSTRING[1..11]		Contents of the bearer capability information element
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CAU (Cause) <b>Encoding Variation</b> : <b>Comments</b> : Info Element CAUse ETS 300 403-1 subclause 4.5.12			
Element Name	Type Definition	Field Encoding	Comments
cau_i	CAU_I		Identifier
cau_l	BITSTRING[8]		Length
cau_e3_eb	BITSTRING[1]		Extension bit
cau_e3_cs	BITSTRING[3]		Coding standard
cau_e3_loc	BITSTRING[4]		Location
cau_e4_rec	OCTETSTRING[0..1]		Recommendation
cau_e5_eb	BITSTRING[1]		Extension bit
cau_e5_cv	BITSTRING[7]		Cause value
cau_di	OCTETSTRING[0..28]		Diagnostics
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CDPN (CALLED PARTY NUMBER IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Called Party Number EN 300 403-1 & Q.931 subclause 4.5.8			
Element Name	Type Definition	Field Encoding	Comments
cdpn_i	CDPN_I		Identifier
cdpn_l	OCTETSTRING [1]		Length
cdpn_e3_npi	OCTETSTRING [1]		Numbering plan id.
cdpn_e4_nd	OCTETSTRING [1 TO 20]		Number digits
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CDPS (CALLED PARTY SUBADDRESS IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Called Party Subaddress EN 300 403-1 & Q.931 subclause 4.5.9			
Element Name	Type Definition	Field Encoding	Comments
cdps_i	CDPS_I		Identifier
cdps_l	BITSTRING [8]		Length
cdps_e3_tos	BITSTRING [8]		Type of subaddress
cdps_e4_si	OCTETSTRING [1 TO 20]		Subaddress information
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CGPN (CALLING PARTY NUMBER IE)			
<b>Encoding Variation</b> :			
<b>Comments</b> : Info Element CallinG Party Number EN 300 403-1 & Q.931 subclause 4.5.10			
Element Name	Type Definition	Field Encoding	Comments
cgpn_i	CGPN_I		Identifier
cgpn_l	BITSTRING [8]		Length
cgpn_e3_ton	BITSTRING [4]		Type of number
cgpn_e3_npi	BITSTRING [4]		Numbering plan id.
cgpn_e4_pi	BITSTRING [3]		Presentation indicator
cgpn_e4_si	BITSTRING [5]		Screening indicator
cgpn_e5_nd	OCTETSTRING [0 TO 20]		Number digits
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CGPS (CALLING PARTY SUBADDRESS IE)			
<b>Encoding Variation</b> :			
<b>Comments</b> : Info Element CallinG Party Subaddress EN 300 403-1 & Q.931 subclause 4.5.11			
Element Name	Type Definition	Field Encoding	Comments
cgps_i	CGPS_I		Identifier
cgps_l	BITSTRING [8]		Length
cgps_e3_tos	BITSTRING [4]		Type of subaddress
cgps_e3_oei	BITSTRING [1]		Odd/even indicator
cgps_e3_sp	BITSTRING [3]		Spare
cgps_e4_si	OCTETSTRING [1 TO 20]		Subaddress information
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CHI (Channel identification) <b>Encoding Variation</b> : <b>Comments</b> : Information Element CHannel Identification EN 300 403-1 subclause 4.5.13			
Element Name	Type Definition	Field Encoding	Comments
chi_i (Identifier)	CHI_I		
chi_l (Length)	BITSTRING[8]		
chi_e3_p1	BITSTRING[4]		(1)
chi_e3_pe	BITSTRING[1]		(1)
chi_e3_p3	BITSTRING[3]		(1)
chi_e3_cs	BITSTRING[8]		(2) BITSTRING[8]
chi_e4	BITSTRING[8]		(3)
chi_e5_ch1 (Extension bit)	BITSTRING[1]		(3)
chi_e5_ch2 (Channel number)	BITSTRING[7]		(3) BITSTRING[7]
<b>Detailed Comments</b> : (1) Coding of octet 3 used for primary rate access. (2) Coding of octet 3 used for basic access. (3) The octets 4 and 5 are only used in primary rate access configurations.			

Structured Type Definition			
<b>Type Name</b> : CHI_RS (Channel identification) <b>Encoding Variation</b> : <b>Comments</b> : Information Element CHannel Identification used in RESTART and RESTART ACK PDUs ETS 300 403-1 subclause 4.5.13			
Element Name	Type Definition	Field Encoding	Comments
chi_i	CHI_I		Identifier
chi_l	BITSTRING[8]		Length
chi_e3_p1	BITSTRING[4]		(1)
chi_e3_pe	BITSTRING[1]		(1)
chi_e3_p3	BITSTRING[3]		(1)
chi_e3_cs	OCTETSTRING[1]		(2)
chi_e4	BITSTRING[8]		(3)
chi_e5_ch	OCTETSTRING[1..30]		(3)
<b>Detailed Comments</b> : (1) Coding of octet 3 used for primary rate access. (2) Coding of octet 3 used for basic access. (3) The octets 4 and 5 are only used in primary rate access configurations.			

Structured Type Definition			
<b>Type Name</b> : CODN (Connected number) <b>Encoding Variation</b> : <b>Comments</b> : Information Element COnnectedD Number ETS 300 097-1 subclause 7.1			
Element Name	Type Definition	Field Encoding	Comments
codn_i	CODN_I		Identifier
codn_l	OCTETSTRING[1]		Length
codn_e3_ton	BITSTRING[4]		Type of number
codn_e3_npi	BITSTRING[4]		Numbering plan identifier
codn_e4_pi	BITSTRING[3]		Presentation indicator
codn_e4_si	BITSTRING[5]		Screening indicator
codn_e5_nd	OCTETSTRING[1..20]		Number digits
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CODS (Connected subaddress) <b>Encoding Variation</b> : <b>Comments</b> : Information Element COnnectedD Subaddress ETS 300 097-1 subclause 7.2			
Element Name	Type Definition	Field Encoding	Comments
cods_i	CODS_I		Identifier
cods_l	OCTETSTRING[1]		Length
cods_e3_tos	BITSTRING[4]		Type of subaddress
cods_e3_oei	BITSTRING[1]		Odd/even indicator
cods_e3_sp	BITSTRING[3]		Spare
cods_e4_si	OCTETSTRING[1..20]		Subaddress information
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : CR (CALL REFERENCE) <b>Encoding Variation</b> : <b>Comments</b> : Call Reference EN 300 403-1 & Q.931 subclause 4.3			
Element Name	Type Definition	Field Encoding	Comments
cr_l	BITSTRING [8]		Length
cr_f	BITSTRING [1]		Flag
cr_r	BIT7OR15		Call reference value
<b>Detailed Comments</b> : The call reference is of type BITSTRING[7] for basic access and of type BITSTRING[15] for primary rate access.			

Structured Type Definition			
<b>Type Name</b> : DATI (DATE/TIME IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element DAtE/TIme EN 300 403-1 & Q.931 subclause 4.6.1			
Element Name	Type Definition	Field Encoding	Comments
dati_i	DATI_I		Identifier
dati_l	BITSTRING [8]		Length
dati_dt	OCTETSTRING [0 TO 5]		Date/time value
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : DSP (DISPLAY IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element DiSPlay EN 300 403-1 & Q.931 subclause 4.5.15			
Element Name	Type Definition	Field Encoding	Comments
dsp_i	DSP_I		Identifier
dsp_l	BITSTRING [8]		Length
dsp_di	OCTETSTRING [0 TO 32]		Display information
<b>Detailed Comments</b> :			



Structured Type Definition			
<b>Type Name</b> : EFAC (Extended facility) <b>Encoding Variation</b> : <b>Comments</b> : Extended FACility ETS 300 196–1 subclause 11.2.2.4			
Element Name	Type Definition	Field Encoding	Comments
efac_i	EFAC_I		Identifier
efac_l	OCTETSTRING[2 TO 250]		Length
efac_e3_pp	BITSTRING[8]		Protocol profile
efac_comp	OCTETSTRING[0..250]		Components
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : FA (FACILITY IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element FACility ETS 300 196–1 subclause 11			
Element Name	Type Definition	Field Encoding	Comments
fac_i	FAC_I		Identifier
fac_l	OCTETSTRING[1]		Length
fac_pp_inv	FAC_PP_INV		Protocol profile, invoke component tag
fac_con	OCTETSTRING[6..250]		Rest of component
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : HLC (HIGH LAYER COMPATIBILITY IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element High Layer Compatibility EN 300 403–1 & Q.931 subclause 4.5.16			
Element Name	Type Definition	Field Encoding	Comments
hlc_i	HLC_I		Identifier
hlc_l	OCTETSTRING[1]		Length
hlc_con	OCTETSTRING[0..3]		Contents of the high layer compatibility information element
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : KPF (KEYPAD FACILITY IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element KeyPad Facility EN 300 403-1 & Q.931 subclause 4.5.17			
Element Name	Type Definition	Field Encoding	Comments
kpf_i	KPF_I		Identifier
kpf_l	BITSTRING [8]		Length
kpf_ki	OCTETSTRING [0 TO 32]		Keypad information
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : LLC (LOW LAYER COMPATIBILITY IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Low Layer Compatibility EN 300 403-1 & Q.931 subclause 4.5.18			
Element Name	Type Definition	Field Encoding	Comments
llc_i	LLC_I		Identifier
llc_l	OCTETSTRING[1]		Length
llc_con	OCTETSTRING[0..14]		Contents of the low layer compatibility information element
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : NSF (NETWORK-SPECIFIC FACILITIES IE)			
<b>Encoding Variation</b> :			
<b>Comments</b> : Info Element Network-Specific Facilities ETS 300 102-1 clause 4.5.20			
Element Name	Type Definition	Field Encoding	Comments
nsf_i	NSF_I		Identifier
nsf_l	BITSTRING [8]		Length
nsf_lni	BITSTRING [8]		Length of network identification
nsf_toni	BITSTRING [4]		Type of network identification
nsf_nip	BITSTRING [4]		Network identification plan
nsf_ni	OCTETSTRING		Network identification
nsf_nsfs	OCTETSTRING		Network-specific facility specification
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : NOID (NOTIFICATION INDICATOR IE)			
<b>Encoding Variation</b> :			
<b>Comments</b> : Info Element NOTification InDicator EN 300 403-1 & Q.931 subclause 4.5.21			
Element Name	Type Definition	Field Encoding	Comments
noid_i	NOID_I		Identifier
noid_l	BITSTRING [8]		Length
noid_e3_nd	BITSTRING [8]		Notification description
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : PI (PROGRESS INDICATOR IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Progress Indicator EN 300 403-1 & Q.931 subclause 4.5.22			
Element Name	Type Definition	Field Encoding	Comments
pi_i	PI_I		Identifier
pi_l	BITSTRING [8]		Length
pi_e3_loc	BITSTRING [8]		Location
pi_e4_pd	BITSTRING [8]		Progress description
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : RI (RESTART INDICATOR IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Restart Indicator ETS 300 102-1 clause 4.5.24			
Element Name	Type Definition	Field Encoding	Comments
ri_i	RI_I		Identifier
ri_l	BITSTRING [8]		Length
ri_cl	BITSTRING [5]		Fixed value '10000'B
ri_cl1	BITSTRING [3]		Class
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : RNGN (Redirecting number) <b>Encoding Variation</b> : <b>Comments</b> : Info Element RedirectiNG Number ETS 300 207 subclause 7.2.2			
Element Name	Type Definition	Field Encoding	Comments
rngn_i	RNGN_I		Identifier
rngn_l	OCTETSTRING[1]		Length
rngn_e3_ton	BITSTRING[4]		Type of number
rngn_e3_npi	BITSTRING[4]		Numbering plan identifier
rngn_e4_pi	BITSTRING[3]		Presentation indicator
rngn_e4_sp	BITSTRING[5]		Spare
rngn_e5_sp	BITSTRING[4]		Spare
rngn_e5_rfd	BITSTRING[4]		Reason for diversion
rngn_e6_nd	OCTETSTRING[1 TO 20]		Number digits
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : RONN (Redirection number) <b>Encoding Variation</b> : <b>Comments</b> : Info Element RedirectiON Number ETS 300 207-1 subclause 7.2.3			
Element Name	Type Definition	Field Encoding	Comments
ronn_i	RONN_I		Identifier
ronn_l	OCTETSTRING[1]		Length
ronn_e3_ton	BITSTRING[4]		Type of number
ronn_e3_npi	BITSTRING[4]		Numbering plan identifier
ronn_e4_pi	BITSTRING[3]		Presentation indicator
ronn_e4_sp	BITSTRING[5]		Spare
ronn_e5_nd	OCTETSTRING[1 TO 20]		Number digits
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : TNS (TRANSIT NETWORK SELECTION IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element Transit Network Selection EN 300 403-1 & Q.931 subclause 4.5.28			
Element Name	Type Definition	Field Encoding	Comments
tns_i	TNS_I		Identifier
tns_l	BITSTRING [8]		Length
tns_toni	BITSTRING [4]		Type of network identification
tns_nip	BITSTRING [4]		Network identification plan
tns_ni	OCTETSTRING [0 TO 251]		Network identification
<b>Detailed Comments</b> :			

Structured Type Definition			
<b>Type Name</b> : UUI (USER USER INFORMATION IE) <b>Encoding Variation</b> : <b>Comments</b> : Info Element User–User EN 300 403-1 & Q.931 subclause 4.5.29			
Element Name	Type Definition	Field Encoding	Comments
uui_i	UUI_I		Identifier
uui_l	BITSTRING [8]		Length
uui_pd	BITSTRING [8]		Protocol discriminator
uui_uic	OCTETSTRING [0 TO 128]		User information
<b>Detailed Comments</b> :			

ASN.1 Type Definition	
<b>Type Name</b>	: AddressInformation
<b>Encoding Variation</b>	:
<b>Comments</b>	: EN 301 144–1 annex E and EN 300 196–1 annex D.3
Type Definition	
<pre> CHOICE { unknownPartyNumber [0] IMPLICIT NumberDigits,   publicPartyNumber [1] IMPLICIT PublicPartyNumber,   dataPartyNumber [3] IMPLICIT NumberDigits,   telexPartyNumber [4] IMPLICIT NumberDigits,   privateNumber [5] IMPLICIT PrivatePartyNumber,   nationalStandardPartyNumber [8] IMPLICIT NumberDigits }  PublicPartyNumber ::= SEQUENCE { publicTypeOfNumber PublicTypeOfNumber,   publicNumberDigits NumberDigits }  PrivatePartyNumber ::= SEQUENCE { privateTypeOfNumber PrivateTypeOfNumber,   privateNumberDigits NumberDigits }  PublicTypeOfNumber ::= TypeOfNumber ( unknownNumber  internnationalNumber    nationalNumber    networkSpecificNumber    subscriberNumber    abbreviatedNumber )  PrivateTypeOfNumber ::= TypeOfNumber ( unknownNumber    level2RegionalNumber    level1RegionalNumber    pTNSpecificNumber    localNumber    abbreviatedNumber )  NumberDigits ::= NumericString (SIZE(1..20)) </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: AllocType
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5, 7.7.2
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: AuthType
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: BasicService
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5, 7.6.4
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: CipherInfo
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5, 7.7.10 with applicable GSM mappings as defined in ETS 300 370, 6.1.8.2.13 or 6.1.7.1.3
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:



ASN.1 Type Definition	
<b>Type Name</b>	: CipherKey
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175-7, 4.4.3.3 with applicable GSM mappings as defined in ETS 300 370, Annex A
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: Component
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER The collection of all possible components for CTM alpha user side
Type Definition	
<pre> CHOICE {   encapsulatedStimulus_Components      EncapsulatedStimulus_Components,   cTMAccessRightsRequest_Components     CTMAccessRightsRequest_Components,   cTMAccessRightsTerminate_Components   CTMAccessRightsTerminate_Components,   cTMLocationRegistration_Components     CTMLocationRegistration_Components,   cTMLocationCancellation_Components     CTMLocationCancellation_Components,   cTMLocationRegistrationSuggest_Components CTMLocationRegistrationSuggest_Components,   gSMLocationRegistration_Components     GSMLocationRegistration_Components,   cTMTerminalAuthentication_Components   CTMTerminalAuthentication_Components,   gSMAssignIdentity_Components           GSMAssignIdentity_Components,   gSMTerminalAuthentication_Components   GSMTerminalAuthentication_Components,   gSMTerminalAuthenticationReject_Components GSMTerminalAuthenticationReject_Components,   cTMNetworkAuthentication_Components    CTMNetworkAuthentication_Components,   cTMCiphering_Components                CTMCiphering_Components,   gSMCiphering_Components                GSMCiphering_Components,   cTMCipheringSuggest_Components         CTMCipheringSuggest_Components,   cTMKeyAllocate_Components              CTMKeyAllocate_Components,   cTMIdentityRequest_Components           CTMIdentityRequest_Components,   cTMOutgoingCallMMInfo_Components       CTMOutgoingCallMMInfo_Components,   cTMIncomingCallMMInfo_Components       CTMIncomingCallMMInfo_Components,   gSMOutgoingCallMMInfo_Components       GSMOutgoingCallMMInfo_Components,   gSMIncomingCallMMInfo_Components       GSMIncomingCallMMInfo_Components,   gSMIdentityRequest_Components           GSMIdentityRequest_Components,   gSMLinkedAssignIdentity_Components      GSMLinkedAssignIdentity_Components,   general_Components                     General_Components   -- required to cope with the receipt of "other" components which are ignored. } </pre>	
<b>Detailed Comments</b>	: plural (components) as each type represents invoke, return result, return error etc.

## ASN.1 Type Definition

**Type Name** : CTMAccessRightsRequest\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  cTMAccessRightsRequest_InvokeCpt [1] IMPLICIT CTMAccessRightsRequest_InvokeCpt,
  cTMAccessRightsRequest_RRCpt      [2] IMPLICIT CTMAccessRightsRequest_RRCpt,
  cTMAccessRightsRequest_RECpt      [3] IMPLICIT CTMAccessRightsRequest_RECpt,
  cTMAccessRightsRequest_RejectCpt  [4] IMPLICIT RejectComponent }
```

-- This is the CTMAccessRightsRequest InvokeComponent --

```
CTMAccessRightsRequest_InvokeCpt ::= SEQUENCE {
```

```
  invokeID      InvokeIDType,
```

```
  -- note "linkedID" not used for this operation
```

```
  operation_value Operation,
```

```
  argument      Argument_Invoke OPTIONAL }
```

-- This is the CTMAccessRightsRequest Return Result Component --

```
CTMAccessRightsRequest_RRCpt ::= SEQUENCE {
```

```
  invokeID      InvokeIDType,
```

```
  argument      SEQUENCE
```

```
  {
```

```
    operation_value Operation,
```

```
    result      Argument_RR OPTIONAL
```

```
  }
```

```
}
```

-- This is the CTMAccessRightsRequest ReturnErrorComponent --

```
CTMAccessRightsRequest_RECpt ::= SEQUENCE {
```

```
  invokeID      InvokeIDType,
```

```
  error         CTMAccessRightsRequestError }
```

-- Common (local) type elements --

```
Argument_Invoke ::= SEQUENCE {
```

```
  cTMPortableIdentity [0] PortableIdentity,
```

```
  cTMAuthType         [1] AuthType,
```

```
  cTMPortablesCapabilities [2] PortableCapabilities }
```

```
Argument_RR ::= SEQUENCE {
```

```
  cTMPortableIdentity [0] PortableIdentity,
```

```
  cTMFixedIdentity    [1] FixedIdentity,
```

```
  cTMServiceClass     [2] ServiceClass OPTIONAL }
```

```
CTMAccessRightsRequestError ::= Error
```

**Detailed Comments** :

ASN.1 Type Definition	
<b>Type Name</b>	: CTMAccessRightsTerminate_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMAccessRightsTerminate_InvokeCpt  [1] IMPLICIT CTMAccessRightsTerminate_InvokeCpt,   cTMAccessRightsTerminate_RRCpt      [2] IMPLICIT CTMAccessRightsTerminate_RRCpt,   cTMAccessRightsTerminate_RECpt      [3] IMPLICIT CTMAccessRightsTerminate_RECpt,   cTMAccessRightsTerminate_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the CTMAccessRightsTerminate InvokeComponent -- CTMAccessRightsTerminate_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the CTMAccessRightsTerminate Return Result Component -- CTMAccessRightsTerminate_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType }  -- This is the CTMAccessRightsTerminate ReturnErrorComponent -- CTMAccessRightsTerminate_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMAccessRightsTerminateError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity  [0] PortableIdentity,   cTMFixedIdentity     [1] FixedIdentity}  CTMAccessRightsTerminateError ::= Error </pre>	
<b>Detailed Comments</b>	:

## ASN.1 Type Definition

**Type Name** : CTMCiphering\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  cTMCiphering_InvokeCpt [1] IMPLICIT CTMCiphering_InvokeCpt,
  cTMCiphering_RRCpt     [2] IMPLICIT CTMCiphering_RRCpt,
  cTMCiphering_RECpt     [3] IMPLICIT CTMCiphering_RECpt,
  cTMCiphering_RejectCpt [4] IMPLICIT RejectComponent }
```

-- This is the CTMCiphering InvokeComponent --

```
CTMCiphering_InvokeCpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  -- note "linkedID" not used for this operation
  operation_value Operation,
  argument      Argument_Invoke OPTIONAL
}
```

-- This is the CTMCiphering Return Result Component --

```
CTMCiphering_RRCpt ::= SEQUENCE {
  invokeID      InvokeIDType
}
```

-- This is the CTMCiphering ReturnErrorComponent --

```
CTMCiphering_RECpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  error         CTMCipheringError }
```

-- Common (local) type elements --

```
Argument_Invoke ::= SEQUENCE {
  cTMPortableIdentity [0] PortableIdentity OPTIONAL,
  cTMCipherInfo [1] CipherInfo,
  cTMCipherKey [2] CipherKey
}
```

```
CTMCipheringError ::= Error
```

**Detailed Comments** :

ASN.1 Type Definition	
<b>Type Name</b>	: CTMCipheringSuggest_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMCipheringSuggest_InvokeCpt  [1] IMPLICIT CTMCipheringSuggest_InvokeCpt,   cTMCipheringSuggest_RECpt      [3] IMPLICIT CTMCipheringSuggest_RECpt,   cTMCipheringSuggest_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the CTMCipheringSuggest InvokeComponent -- CTMCipheringSuggest_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the CTMCipheringSuggest ReturnErrorComponent -- CTMCipheringSuggest_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMCipheringSuggestError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity [0] PortableIdentity OPTIONAL,   cTMCipherInfo [1] CipherInfo }  CTMCipheringSuggestError ::= Error </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: CTMIncomingCallMMInfo_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMICallMMInfo_InvokeCpt  [1] IMPLICIT CTMCallMMInfo_InvokeCpt,   cTMICallMMInfo_RECpt      [3] IMPLICIT CTMCallMMInfo_RECpt,   cTMICallMMInfo_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the CTMCallMMInfo InvokeComponent -- CTMCallMMInfo_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the CTMCallMMInfo ReturnErrorComponent -- CTMCallMMInfo_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMCallMMInfoError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity [0] PortableIdentity,   cTMSignal [1] Signal }  CTMCallMMInfoError ::= Error </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: CTMIdentityRequest_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMIdentityRequest_InvokeCpt  [1] IMPLICIT CTMIdentityRequest_InvokeCpt,   cTMIdentityRequest_RRCpt      [2] IMPLICIT CTMIdentityRequest_RRCpt,   cTMIdentityRequest_RECpt      [3] IMPLICIT CTMIdentityRequest_RECpt,   cTMIdentityRequest_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the CTMIdentityRequest InvokeComponent -- CTMIdentityRequest_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the CTMIdentityRequest Return Result Component -- CTMIdentityRequest_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType,   argument      SEQUENCE   {     operation_value Operation,     result      Argument_RR OPTIONAL   } }  -- This is the CTMIdentityRequest ReturnErrorComponent -- CTMIdentityRequest_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMIdentityRequestError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity [0] PortableIdentity OPTIONAL,   cTMIdentityType [1] IdentityType }  Argument_RR ::= SEQUENCE {   cTMPortableIdentity [0] PortableIdentity }  CTMIdentityRequestError ::= Error </pre>	
<b>Detailed Comments</b>	:

## ASN.1 Type Definition

**Type Name** : CTMKeyAllocate\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  cTMKeyAllocate_InvokeCpt [1] IMPLICIT CTMKeyAllocate_InvokeCpt,
  cTMKeyAllocate_RRCpt [2] IMPLICIT CTMKeyAllocate_RRCpt,
  cTMKeyAllocate_RECpt [3] IMPLICIT CTMKeyAllocate_RECpt,
  cTMKeyAllocate_RejectCpt [4] IMPLICIT RejectComponent }
```

-- This is the CTMKeyAllocate InvokeComponent --

```
CTMKeyAllocate_InvokeCpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  linkedID      [0] IMPLICIT InvokeIDType OPTIONAL,
  operation_value Operation,
  argument      Argument_Invoke OPTIONAL
}
```

-- This is the CTMKeyAllocate Return Result Component --

```
CTMKeyAllocate_RRCpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  argument      SEQUENCE
  {
    operation_value Operation,
    result      Argument_RR OPTIONAL
  }
}
```

-- This is the CTMKeyAllocate ReturnErrorComponent --

```
CTMKeyAllocate_RECpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  error         CTMKeyAllocateError
}
```

-- Common (local) type elements --

```
Argument_Invoke ::= SEQUENCE {
  cTMPortableIdentity [0] PortableIdentity OPTIONAL,
  cTMAllocType [1] AllocType,
  cTMRand [2] Rand,
  cTMRs [3] Rs
}
```

```
Argument_RR ::= SEQUENCE {
  cTMRes [0] Res
}
```

```
CTMKeyAllocateError ::= Error
```

**Detailed Comments** :



ASN.1 Type Definition	
<b>Type Name</b>	: CTMLocationAreIdentity
<b>Encoding Variation</b>	:
<b>Comments</b>	: containing an RFPI as encoded in ETS 300 175-6 subclause 5 figure 4, truncated to the length corresponding to the relevant LAL.
Type Definition	
BIT STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: CTMLocationCancellation_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMLocationCancellation_InviteCpt [1] IMPLICIT CTMLocationCancellation_InviteCpt,   cTMLocationCancellation_RRCpt     [2] IMPLICIT CTMLocationCancellation_RRCpt,   cTMLocationCancellation_RECpt     [3] IMPLICIT CTMLocationCancellation_RECpt,   cTMLocationCancellation_RejectCpt [4] IMPLICIT RejectComponent }  -- This is the CTMLocationCancellation InvokeComponent -- CTMLocationCancellation_InviteCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke OPTIONAL }  -- This is the CTMLocationCancellation Return Result Component -- CTMLocationCancellation_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType }  -- This is the CTMLocationCancellation ReturnErrorComponent -- CTMLocationCancellation_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMLocationCancellationError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity [0] PortableIdentity }  CTMLocationCancellationError ::= Error </pre>	
<b>Detailed Comments</b>	:

## ASN.1 Type Definition

**Type Name** : CTMLocationRegistration\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  cTMLocationRegistration_InvokeCpt [1] IMPLICIT CTMLocationRegistration_InvokeCpt,
  cTMLocationRegistration_RRCpt [2] IMPLICIT CTMLocationRegistration_RRCpt,
  cTMLocationRegistration_RECpt [3] IMPLICIT CTMLocationRegistration_RECpt,
  cTMLocationRegistration_RejectCpt [4] IMPLICIT RejectComponent }
```

-- This is the CTMLocationRegistration InvokeComponent --

```
CTMLocationRegistration_InvokeCpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  -- note "linkedID" not used for this operation
  operation_value Operation,
  argument      Argument_Invoke OPTIONAL }
```

-- This is the CTMLocationRegistration Return Result Component --

```
CTMLocationRegistration_RRCpt ::= SEQUENCE {
  invokeID      InvokeIDType }
```

-- This is the CTMLocationRegistration ReturnErrorComponent --

```
CTMLocationRegistration_RECpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  error         CTMLocationRegistrationError }
```

-- Common (local) type elements --

```
Argument_Invoke ::= SEQUENCE {
  cTMPortableIdentity [0] PortableIdentity,
  cTMOldLocationAreaIdentity [1] CTMLocationAreaIdentity,
  cTMNewLocationAreaIdentity [2] CTMLocationAreaIdentity,
  cTMPortableCapabilities [3] PortableCapabilities
}
```

```
CTMLocationRegistrationError ::= Error
```

**Detailed Comments** :

ASN.1 Type Definition	
<b>Type Name</b>	: CTMLocationRegistrationSuggest_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMLocationRegistrationSuggest_InvokeCpt  [1] IMPLICIT CTMLocationRegistrationSuggest_InvokeCpt,   cTMLocationRegistrationSuggest_RECpt      [3] IMPLICIT CTMLocationRegistrationSuggest_RECpt,   cTMLocationRegistrationSuggest_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the CTMLocationRegistrationSuggest InvokeComponent -- CTMLocationRegistrationSuggest_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the CTMLocationRegistrationSuggest ReturnErrorComponent -- CTMLocationRegistrationSuggest_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMLocationRegistrationSuggestError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity      [0] PortableIdentity }  CTMLocationRegistrationSuggestError ::= Error </pre>	
<b>Detailed Comments</b>	:

## ASN.1 Type Definition

**Type Name** : CTMNetworkAuthentication\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  cTMNetworkAuthentication_InvokeCpt [1] IMPLICIT CTMNetworkAuthentication_InvokeCpt,
  cTMNetworkAuthentication_RRCpt [2] IMPLICIT CTMNetworkAuthentication_RRCpt,
  cTMNetworkAuthentication_RECpt [3] IMPLICIT CTMNetworkAuthentication_RECpt,
  cTMNetworkAuthentication_RejectCpt [4] IMPLICIT RejectComponent }
```

-- This is the CTMNetworkAuthentication InvokeComponent --

CTMNetworkAuthentication\_InvokeCpt ::= SEQUENCE {

```
  invokeID      InvokeIDType,
  -- note "linkedID" not used for this operation
  operation_value Operation,
  argument      Argument_Invoke OPTIONAL }
```

-- This is the CTMNetworkAuthentication Return Result Component --

CTMNetworkAuthentication\_RRCpt ::= SEQUENCE {

```
  invokeID      InvokeIDType,
  argument      SEQUENCE
  {
    operation_value Operation,
    result      Argument_RR OPTIONAL
  }
}
```

-- This is the CTMNetworkAuthentication ReturnErrorComponent --

CTMNetworkAuthentication\_RECpt ::= SEQUENCE {

```
  invokeID      InvokeIDType,
  error         CTMNetworkAuthenticationError }
```

-- Common (local) type elements --

Argument\_Invoke ::= SEQUENCE {

```
  cTMPortableIdentity [0] PortableIdentity OPTIONAL,
  cTMAuthType [1] AuthType,
  cTMRand [2] Rand
}
```

Argument\_RR ::= SEQUENCE {

```
  cTMRes [0] Res,
  cTMRs [1] Rs OPTIONAL
}
```

CTMNetworkAuthenticationError ::= Error

**Detailed Comments** :

ASN.1 Type Definition	
<b>Type Name</b>	: CTMOutgoingCallMMInfo_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   cTMOCallMMInfo_InvokeCpt  [1] IMPLICIT CTMOCallMMInfo_InvokeCpt,   cTMOCallMMInfo_RECpt     [3] IMPLICIT CTMOCallMMInfo_RECpt,   cTMOCallMMInfo_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the CTMOCallMMInfo InvokeComponent -- CTMOCallMMInfo_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the CTMOCallMMInfo ReturnErrorComponent -- CTMOCallMMInfo_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         CTMOCallMMInfoError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   cTMPortableIdentity [0] PortableIdentity,   cTMFixedIdentity [1] FixedIdentity,   cTMBasicService [2] BasicService }  CTMOCallMMInfoError ::= Error </pre>	
<b>Detailed Comments</b>	:

## ASN.1 Type Definition

**Type Name** : CTMTerminalAuthentication\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  cTMTerminalAuthentication_InvokeCpt [1] IMPLICIT CTMTerminalAuthentication_InvokeCpt,
  cTMTerminalAuthentication_RRCpt [2] IMPLICIT CTMTerminalAuthentication_RRCpt,
  cTMTerminalAuthentication_RECpt [3] IMPLICIT CTMTerminalAuthentication_RECpt,
  cTMTerminalAuthentication_RejectCpt [4] IMPLICIT RejectComponent }
```

-- This is the CTMTerminalAuthentication InvokeComponent --

CTMTerminalAuthentication\_InvokeCpt ::= SEQUENCE {

```
  invokeID      InvokeIDType,
  -- note "linkedID" not used for this operation
  operation_value Operation,
  argument      Argument_Invoke OPTIONAL }
```

-- This is the CTMTerminalAuthentication Return Result Component --

CTMTerminalAuthentication\_RRCpt ::= SEQUENCE {

```
  invokeID      InvokeIDType,
  argument      SEQUENCE
  {
    operation_value Operation,
    result      Argument_RR OPTIONAL
  }
}
```

-- This is the CTMTerminalAuthentication ReturnErrorComponent --

CTMTerminalAuthentication\_RECpt ::= SEQUENCE {

```
  invokeID      InvokeIDType,
  error         CTMTerminalAuthenticationError }
```

-- Common (local) type elements --

Argument\_Invoke ::= SEQUENCE {

```
  cTMPortableIdentity [0] PortableIdentity OPTIONAL,
  cTMAuthType [1] AuthType,
  cTMRand [2] Rand,
  cTMRs [3] Rs
}
```

Argument\_RR ::= SEQUENCE {

```
  cTMRes [0] Res,
  cTMServiceClass [1] ServiceClass OPTIONAL
}
```

CTMTerminalAuthenticationError ::= Error

**Detailed Comments** :

ASN.1 Type Definition	
<b>Type Name</b>	: DisplayIE
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 302 094-1
Type Definition	
SEQUENCE { displayInf [0] DisplayInf }	
DisplayInf ::= VisibleString (SIZE (1..82))	
<b>Detailed Comments</b> :	

ASN.1 Type Definition	
<b>Type Name</b>	: KeypadIE
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 302 094-1
Type Definition	
SEQUENCE { keypadInf [0] KeypadInf }	
KeypadInf ::= VisibleString (SIZE(1..34)) (FROM ("0" "1" "2" "3" "4" "5" "6" "7" "8" "9" "*" "#"))	
<b>Detailed Comments</b> :	

ASN.1 Type Definition	
<b>Type Name</b>	: EncapsulatedStimulus_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 302 094-1
Type Definition	
CHOICE { encapsulatedStimulus_InvokeCpt [1] IMPLICIT EncapsulatedStimulus_InvokeCpt, encapsulatedStimulus_RejectCpt [4] IMPLICIT RejectComponent }  -- This is the EncapsulatedStimulus InvokeComponent -- EncapsulatedStimulus_InvokeCpt ::= SEQUENCE { invokeID InvokeIDType, -- note "linkedID" not used for this operation operation_value Operation, argument Argument OPTIONAL }  Argument ::= CHOICE { keypad [0] IMPLICIT KeypadIE, display [1] IMPLICIT DisplayIE } 	
<b>Detailed Comments</b> :	

ASN.1 Type Definition	
<b>Type Name</b>	: EntityType
<b>Encoding Variation</b>	:
<b>Comments</b>	: EN 301 144-1 annex E
Type Definition	
<pre> ENUMERATED {   endIntraNetworkNode (0),   anyIntraNetworkNode (1),   endNode (2),   anyNode (3),   endTerminal (4) } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: Error
<b>Encoding Variation</b>	:
<b>Comments</b>	: Ref: EN 300 196, Annex D
Type Definition	
<pre> CHOICE {   localValue    INTEGER,   globalValue   OBJECT IDENTIFIER ,   globalValueP  SEQUENCE {     globalValue  OBJECT IDENTIFIER,     parameter RejectReason OPTIONAL } } </pre>	
<b>Detailed Comments</b>	:



ASN.1 Type Definition	
<b>Type Name</b>	: FIE
<b>Encoding Variation</b>	:
<b>Comments</b>	: Facility information element taken from EN 301 144–1 Annex E and EN 301 061–1
Type Definition	
<pre> SEQUENCE {   informationElementId    FIE_I,   length                  FIE_LengthType,   extBit                  BIT STRING (SIZE (1)),   spareBits                BIT STRING (SIZE (2)),   protocolProfile          BIT STRING (SIZE (5)),   nfe                     NetworkFacilityExtension OPTIONAL,   networkProtocolProfile  NetworkProtocolProfile OPTIONAL,   interpretationAPDU      InterpretationAdu OPTIONAL,   components              SET OF Component } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: FIE_I
<b>Encoding Variation</b>	:
<b>Comments</b>	: Identifier for the Facility information element.
Type Definition	
BIT STRING('00011100'B)	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: FIE_LengthType
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
BIT STRING(SIZE(8))	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
Type Name	: FIES
Encoding Variation	:
Comments	: This type carries a SET OF FIE. The order of the element is of no interest.
Type Definition	
SET OF FIE	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: FixedIdentity
Encoding Variation	:
Comments	:
Type Definition	
BIT STRING	
Detailed Comments	:

ASN.1 Type Definition	
<b>Type Name</b>	: General_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: Non specified components must match this type definition.
Type Definition	
<pre> CHOICE {   general_InvokeComp    [1] IMPLICIT General_InvokeCpt,   general_ReturnResultComp [2] IMPLICIT General_ReturnResultCpt,   general_ReturnErrorComp [3] IMPLICIT General_ReturnErrorCpt,   general_RejectComp    [4] IMPLICIT RejectComponent }  -- This is the General InvokeComponent -- General_InvokeCpt ::= SEQUENCE {   invokeID          InvokeIDType,   linked_ID         [0] IMPLICIT InvokeIDType OPTIONAL,   operation_value    Operation,   argument          ANY  OPTIONAL }  -- This is the General ReturnResultComponent -- General_ReturnResultCpt ::= SEQUENCE {   invokeID          InvokeIDType,   valueAndResult SEQUENCE {     operation_value Operation,     result          ANY  } OPTIONAL }  -- This is the General ReturnErrorComponent -- General_ReturnErrorCpt ::= SEQUENCE {   invokeID InvokeIDType,   error    ANY } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GeneralProblem
<b>Encoding Variation</b>	:
<b>Comments</b>	: from ETS 300 196 D.1
Type Definition	
<pre> ROSE_Problems ( unrecognizedComponent     mistypedComponent     badlyStructuredComponent ) </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMAssignIdentity_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMAssignIdentity_InviteCpt  [1] IMPLICIT GSMAssignIdentity_InviteCpt,   gSMAssignIdentity_RRCpt     [2] IMPLICIT GSMAssignIdentity_RRCpt,   gSMAssignIdentity_RejectCpt [4] IMPLICIT RejectComponent }  -- This is the GSMAssignIdentity InviteComponent -- GSMAssignIdentity_InviteCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invite  OPTIONAL }  -- This is the GSMAssignIdentity Return Result Component -- GSMAssignIdentity_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType }  -- Common (local) type elements -- Argument_Invite ::= SEQUENCE {   gSMPortableIdentity [0] PortableIdentity OPTIONAL,   gSMLocationAreaIdentity [1] GSMLocationAreaIdentity,   gSMNewTMSI [2] PortableIdentity } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMCiphering_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMCiphering_InvokeCpt  [1] IMPLICIT GSMCiphering_InvokeCpt,   gSMCiphering_RRCpt      [2] IMPLICIT GSMCiphering_RRCpt,   gSMCiphering_RECpt      [3] IMPLICIT GSMCiphering_RECpt,   gSMCiphering_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMCiphering InvokeComponent -- GSMCiphering_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the GSMCiphering Return Result Component -- GSMCiphering_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType }  -- This is the GSMCiphering ReturnErrorComponent -- GSMCiphering_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         GSMCipheringError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   gSMPortableIdentity [0] PortableIdentity OPTIONAL,   gSMCipherKey [1] CipherKey }  GSMCipheringError ::= Error </pre>	
<b>Detailed Comments</b>	:

## ASN.1 Type Definition

**Type Name** : GSMIdentityRequest\_Components

**Encoding Variation** :

**Comments** : from EN 301 144-1

## Type Definition

```
CHOICE {
  gSMIdentityRequest_InvokeCpt [1] IMPLICIT GSMIdentityRequest_InvokeCpt,
  gSMIdentityRequest_RRCpt [2] IMPLICIT GSMIdentityRequest_RRCpt,
  gSMIdentityRequest_RECpt [3] IMPLICIT GSMIdentityRequest_RECpt,
  gSMIdentityRequest_RejectCpt [4] IMPLICIT RejectComponent }
```

-- This is the GSMIdentityRequest InvokeComponent --

```
GSMIdentityRequest_InvokeCpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  -- note "linkedID" not used for this operation
  operation_value Operation,
  argument      Argument_Invoke OPTIONAL }
```

-- This is the GSMIdentityRequest Return Result Component --

```
GSMIdentityRequest_RRCpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  argument      SEQUENCE
  {
    operation_value Operation,
    result      Argument_RR OPTIONAL
  }
}
```

-- This is the GSMIdentityRequest ReturnErrorComponent --

```
GSMIdentityRequest_RECpt ::= SEQUENCE {
  invokeID      InvokeIDType,
  error         GSMIdentityRequestError }
```

-- Common (local) type elements --

```
Argument_Invoke ::= SEQUENCE {
  gSMPortableIdentity [0] PortableIdentity OPTIONAL,
  gSMIdentityType [1] IdentityType
}
```

```
Argument_RR ::= SEQUENCE {
  gSMPortableIdentity [0] PortableIdentity
}
```

```
GSMIdentityRequestError ::= Error
```

**Detailed Comments** :

ASN.1 Type Definition	
<b>Type Name</b>	: GSMIncomingCallMMInfo_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMICallMMInfo_InvokeCpt  [1] IMPLICIT GSMICallMMInfo_InvokeCpt,   gSMICallMMInfo_RECpt      [3] IMPLICIT GSMICallMMInfo_RECpt,   gSMICallMMInfo_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMICallMMInfo InvokeComponent -- GSMICallMMInfo_InvokeCpt ::= SEQUENCE {   invokeID          InvokeIDType,   -- note "linkedID" not used for this operation   operation_value   Operation,   argument          Argument_Invoke  OPTIONAL }  -- This is the GSMICallMMInfo ReturnErrorComponent -- GSMICallMMInfo_RECpt ::= SEQUENCE {   invokeID          InvokeIDType,   error             GSMICallMMInfoError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   gSMPortableIdentity [0] PortableIdentity,   gSMSignal [1] Signal }  GSMICallMMInfoError ::= Error </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMLinkedAssignIdentity_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMLinkedAssignIdentity_InviteCpt  [1] IMPLICIT GSMLinkedAssignIdentity_InviteCpt,   gSMLinkedAssignIdentity_RRCpt      [2] IMPLICIT GSMLinkedAssignIdentity_RRCpt,   gSMLinkedAssignIdentity_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMLinkedAssignIdentity InviteComponent -- GSMLinkedAssignIdentity_InviteCpt ::= SEQUENCE {   invokeID      InvokeIDType,   linkedID      InvokeIDType,   operation_value Operation,   argument      Argument_Invite  OPTIONAL }  -- This is the GSMLinkedAssignIdentity Return Result Component -- GSMLinkedAssignIdentity_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType }  -- Common (local) type elements -- Argument_Invite ::= SEQUENCE {   gSMNewTMSI [0] PortableIdentity } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMLocationAreaIdentity
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in Location Area ETS 300 175-5, 7.7.25
Type Definition	
BIT STRING	
<b>Detailed Comments</b>	:



ASN.1 Type Definition	
<b>Type Name</b>	: GSMLocationRegistration_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMLocationRegistration_InvokeCpt  [1] IMPLICIT GSMLocationRegistration_InvokeCpt,   gSMLocationRegistration_RRCpt      [2] IMPLICIT GSMLocationRegistration_RRCpt,   gSMLocationRegistration_RECpt      [3] IMPLICIT GSMLocationRegistration_RECpt,   gSMLocationRegistration_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMLocationRegistration InvokeComponent -- GSMLocationRegistration_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   linkedID      [0] IMPLICIT InvokeIDType OPTIONAL,   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the GSMLocationRegistration Return Result Component -- GSMLocationRegistration_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType,   argument      SEQUENCE   {     operation_value Operation,     result        Argument_RR OPTIONAL   } }  -- This is the GSMLocationRegistration ReturnErrorComponent -- GSMLocationRegistration_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         GSMLocationRegistrationError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   gSMPortableIdentity      [0] PortableIdentity,   gSMLocationRegistrationType [1] LocationRegistrationType,   gSMLocationAreaIdentity  [2] GSMLocationAreaIdentity,   gSMCipherInfo            [3] CipherInfo,   gSMPortableCapabilities  [4] PortableCapabilities }  Argument_RR ::= SEQUENCE {   gSMLocationAreaIdentity [0] GSMLocationAreaIdentity }  GSMLocationRegistrationError ::= Error </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMOutgoingCallMMInfo_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMOCallMMInfo_InvokeCpt  [1] IMPLICIT GSMOCallMMInfo_InvokeCpt,   gSMOCallMMInfo_RECpt      [3] IMPLICIT GSMOCallMMInfo_RECpt,   gSMOCallMMInfo_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMOCallMMInfo InvokeComponent -- GSMOCallMMInfo_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the GSMOCallMMInfo ReturnErrorComponent -- GSMOCallMMInfo_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         GSMOCallMMInfoError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   gSMPortableIdentity [0] PortableIdentity,   gSMBasicService [1] BasicService }  GSMOCallMMInfoError ::= Error </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMTerminalAuthentication_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMTerminalAuthentication_InvokeCpt  [1] IMPLICIT GSMTerminalAuthentication_InvokeCpt,   gSMTerminalAuthentication_RRCpt      [2] IMPLICIT GSMTerminalAuthentication_RRCpt,   gSMTerminalAuthentication_RECpt      [3] IMPLICIT GSMTerminalAuthentication_RECpt,   gSMTerminalAuthentication_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMTerminalAuthentication InvokeComponent -- GSMTerminalAuthentication_InvokeCpt ::= SEQUENCE {   invokeID      InvokeIDType,   -- note "linkedID" not used for this operation   operation_value Operation,   argument      Argument_Invoke  OPTIONAL }  -- This is the GSMTerminalAuthentication Return Result Component -- GSMTerminalAuthentication_RRCpt ::= SEQUENCE {   invokeID      InvokeIDType,   argument      SEQUENCE   {     operation_value Operation,     result        Argument_RR OPTIONAL   } }  -- This is the GSMTerminalAuthentication ReturnErrorComponent -- GSMTerminalAuthentication_RECpt ::= SEQUENCE {   invokeID      InvokeIDType,   error         GSMTerminalAuthenticationError }  -- Common (local) type elements -- Argument_Invoke ::= SEQUENCE {   gSMPortableIdentity [0] PortableIdentity OPTIONAL,   gSMRand [1] Rand,   gSMCipherInfo [2] CipherInfo }  Argument_RR ::= SEQUENCE {   gSMRes [0] Res }  GSMTerminalAuthenticationError ::= Error </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: GSMTerminalAuthenticationReject_Components
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
<pre> CHOICE {   gSMTerminalAuthenticationReject_InviteCpt  [1] IMPLICIT GSMTerminalAuthenticationReject_InviteCpt,   gSMTerminalAuthenticationReject_RejectCpt  [4] IMPLICIT RejectComponent }  -- This is the GSMTerminalAuthentication InviteComponent -- GSMTerminalAuthenticationReject_InviteCpt ::= SEQUENCE {   invokeID          InvokeIDType,   -- note "linkedID" not used for this operation   operation_value   Operation } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: IdentityType
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
ENUMERATED{imsi(0), tmsi(1), imei(2), imeisv(3), ipui(4), ipei(5)}	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: IMEI
<b>Encoding Variation</b>	:
<b>Comments</b>	: ETS 300 599
Type Definition	
OCTET STRING (SIZE(3..8))	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: IMSI
<b>Encoding Variation</b>	:
<b>Comments</b>	: ETS 300 599
Type Definition	
OCTET STRING (SIZE(3..8))	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: InterpretationApdu
<b>Encoding Variation</b>	:
<b>Comments</b>	: Reference EN 300 196
Type Definition	
[11] IMPLICIT ENUMERATED { discardAnyUnrecognisedInvokePdu (0), clrCallIfAnyInvPduNotRecognised (1), rejectAnyUnrecognisedInvokePdu (2) }	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: InvokeIDType
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
INTEGER (−32768 .. 32767)	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: InvokeProblem
<b>Encoding Variation</b>	:
<b>Comments</b>	: from ETS 300 196 D.1
Type Definition	
ROSE_Problems ( duplicateInvocation   unrecognizedOperation   mistypedArgument   resourceLimitation   initiatorReleasing   unrecognizedLinkedID   linkedResponseUnexpected   unexpectedChildOperation )	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: IPUI
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–6, 6.2
Type Definition	
BIT STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: LocationRegistrationType
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
ENUMERATED {normal_updating(0), periodic_updating(1), imsi_attach(2)}	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: NetworkFacilityExtension
<b>Encoding Variation</b>	:
<b>Comments</b>	: References ETS 300 196, EN 301 061
Type Definition	
<pre>[10] IMPLICIT SEQUENCE { sourceEntity      [0] IMPLICIT EntityType,   sourceEntityAddress [1] AddressInformation OPTIONAL,   destEntity        [2] IMPLICIT EntityType,   destEntityAddress  [3] AddressInformation OPTIONAL,   serviceFunction    [4] IMPLICIT OBJECT IDENTIFIER OPTIONAL }</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: NetworkProtocolProfile
<b>Encoding Variation</b>	:
<b>Comments</b>	: reference EN 300 196
Type Definition	
<pre>[18] IMPLICIT INTEGER {   acse (19),   dse(32) } (0..254)</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: OID
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
OBJECT IDENTIFIER	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: Operation
<b>Encoding Variation</b>	:
<b>Comments</b>	: from ETS 300 196 (table E.1) & CCITT X.219 (figure 4).
Type Definition	
CHOICE { localValue INTEGER, globalValue OBJECT IDENTIFIER}	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: PortableCapabilities
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5, 7.7.41 with applicable GSM mappings as defined in ETS 300 370, table 7.
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: PortableIdentity
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Type Definition	
CHOICE { iPUI [0] IPUI, iMSI [1] IMSI, iMEI [2] IMEI, tMSI [3] TMSI}	
<b>Detailed Comments</b>	:



ASN.1 Type Definition	
<b>Type Name</b>	: Rand
<b>Encoding Variation</b>	:
<b>Comments</b>	: as either encoded in ETS 300 175–5, 7.7.32 with applicable GSM mappings as defined in ETS 300 370, 6.1.8.1.9 or 6.1.7.1.2.
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: RejectComponent
<b>Encoding Variation</b>	:
<b>Comments</b>	: Reject Component is not specific to any particular operation. The invokeID may be used to identify a specific operation.
Type Definition	
<pre> SEQUENCE {   invokeID CHOICE {     invokeID InvokeIDType,     null NULL },   problem CHOICE {     generalProblem [0] IMPLICIT GeneralProblem,     invokeProblem [1] IMPLICIT InvokeProblem,     returnResultProblem [2] IMPLICIT ReturnResultProblem,     returnErrorProblem [3] IMPLICIT ReturnErrorProblem } } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: RejectReason
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5 subclause 7.7.34
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: Res
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5, 7.7.35 with applicable GSM mapping as defined in ETS 300 370, 6.1.7.25 and 6.1.8.2.14
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: ReturnErrorProblem
<b>Encoding Variation</b>	:
<b>Comments</b>	: from ETS 300 196 D.1
Type Definition	
ROSE_Problems ( unrecognizedInvocation   errorResponseUnexpected   unrecognizedError   unexpectedError   mistypedParameter )	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: ReturnResultProblem
<b>Encoding Variation</b>	:
<b>Comments</b>	: from ETS 300 196 D.1
Type Definition	
ROSE_Problems ( unrecognizedInvocation   resultResponseUnexpected   mistypedResult )	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: ROSE_Problems
<b>Encoding Variation</b>	:
<b>Comments</b>	: from ETS 300 196 D.1
Type Definition	
<pre> INTEGER {   unrecognizedComponent (0),  -- GeneralProblem   unrecognizedInvocation (0), -- ReturnResultProblem, ReturnErrorProblem   duplicateInvocation (0),  -- InvokeProblem   mistypedComponent (1),  -- GeneralProblem   errorResponseUnexpected (1), -- ReturnErrorProblem   resultResponseUnexpected (1), -- ReturnResultProblem   unrecognizedOperation (1), -- InvokeProblem   badlyStructuredComponent (2), -- GeneralProblem   unrecognizedError (2), -- ReturnErrorProblem   mistypedArgument (2), -- InvokeProblem   mistypedResult (2), -- ReturnResultProblem   resourceLimitation (3), -- InvokeProblem   unexpectedError (3), -- ReturnErrorProblem   mistypedParameter (4), -- ReturnErrorProblem   initiatorReleasing (4), -- InvokeProblem   unrecognizedLinkedID (5), -- InvokeProblem   linkedResponseUnexpected (6), -- InvokeProblem   unexpectedChildOperation (7) -- InvokeProblem } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: Rs
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175-5, 7.7.36
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: ServiceClass
<b>Encoding Variation</b>	:
<b>Comments</b>	: from EN 301 144-1
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: Signal
<b>Encoding Variation</b>	:
<b>Comments</b>	: as encoded in ETS 300 175–5, 7.6.8
Type Definition	
OCTET STRING	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: TMSI
<b>Encoding Variation</b>	:
<b>Comments</b>	: ETS 300 599
Type Definition	
OCTET STRING (SIZE(3..8))	
<b>Detailed Comments</b>	:

ASN.1 Type Definition	
<b>Type Name</b>	: TypeOfNumber
<b>Encoding Variation</b>	:
<b>Comments</b>	: EN 301 144–1, annex D
Type Definition	
ENUMERATED { unknownNumber (0), internationalNumber (1), level2RegionalNumber (1), nationalNumber (2), level1RegionalNumber (2), networkSpecificNumber (3), pTNSpecificNumber (3), subscriberNumber (4), localNumber (4), level3RegionalNumber (5), abbreviatedNumber (6) }	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_ASSIGN_CHI(basic, primary : CHI; basic_flag : BOOLEAN)
<b>Result Type</b>	: CHI
<b>Comments</b>	: This operation is used to assign a correct Channel identification information element to PDUs dependant on the type of access that is tested.
Description	
CHI TSO_ASSIGN_CHI(basic,primary,basic_flag)	
If the value of the basic_flag is set to TRUE, the result of the operation ASSIGN_CHI will be the value represented by the parameter basic which is of type CHI. Else the operation results in the value represented by the parameter primary.	
Examples: TSO_ASSIGN_CHI(CHI1b_R1, CHI1p_R1, TRUE) = CHI1b_R1 TSO_ASSIGN_CHI(CHI1b_R1, CHI1p_R1, FALSE) = CHI1p_R1	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_CALC_FIE_LENGTH
<b>Result Type</b>	: FIE_LengthType
<b>Comments</b>	: This operation is used to calculate the length of a Facility information element after BER encoding of the ASN.1 parts.
Description	
The return value represents the length of the contents of a Facility information element in which this test suite operation is called depending on its content.	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_CTM_CALC_RES (rs_ : Rs ; rand: Rand ; alloctype_ : AllocType)
<b>Result Type</b>	: Res
<b>Comments</b>	: This operation calculates the xRes1 value (that will be embedded in the return result component) needed for key allocation procedure.
Description	
This operation calculates the xRes1 value needed for key allocation procedure.	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_CTM_CALC_RES2 (rs_ : Rs ; rand: Rand ; authtype_ : AuthType)
<b>Result Type</b>	: Res
<b>Comments</b>	: This operation calculates the Res value (that will be embedded in the return result component) needed for terminal authentication procedure.
Description	
This operation calculates the Res value needed for terminal authentication procedure.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_CIPHERINFO
<b>Result Type</b>	: CipherInfo
<b>Comments</b>	: This operation finds the cipher info of the specified terminal authentication component in the received Facility information element(s)
Description	
This operation finds the component specified in the COMP parameter within the Facility information element(s) specified in the FAC_IE parameter and returns the value of the cipher info value of that component.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_CTM_alloctype
<b>Result Type</b>	: AllocType
<b>Comments</b>	: This operation finds the AllocType field of the specified component in the received Facility information element(s)
Description	
This operation extracts the AllocType field of the component included in the facility message currently processed in the receive queue.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_CTM_authtype
<b>Result Type</b>	: AuthType
<b>Comments</b>	: This operation finds the AuthType field of the specified component in the received Facility information element(s)
Description	
This operation extracts the AuthType field of the component included in the facility message currently processed in the receive queue.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_CTM_RS
<b>Result Type</b>	: Rs
<b>Comments</b>	: This operation finds the Rs field of the specified component in the received Facility information element(s)
Description	
This operation extrracts the Rs filed of the component included in the facility message currently processed in the receive queue.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_IDTYPE
<b>Result Type</b>	: IdentityType
<b>Comments</b>	: This operation finds the identity type contained in the specified identity request component in the received Facility information element(s)
Description	
This operation finds the component specified in the COMP parameter within the Facility information element(s) specified in the FAC_IE parameter and returns the value of the identity type value of that component.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_INVOKEID
<b>Result Type</b>	: INTEGER
<b>Comments</b>	: This operation finds the invoke Id of the specified component in the received Facility information element(s)
Description	
This operation finds the component specified in the COMP parameter within the Facility information element(s) specified in the FAC_IE parameter and returns the value of the invoke identifier of that component.	
<b>Detailed Comments</b> :	

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_TMSI
<b>Result Type</b>	: PortableIdentity
<b>Comments</b>	: This operation finds the TM SI contained in the specified assignment request or linked assignment request component in the received Facility information element(s)
Description	
This operation finds the component specified in the COMP parameter within the Facility information element(s) specified in the FAC_IE parameter and returns the value of the new TMSI value of that component.\	
The value is stored as a PortableIdentity of the TMSI type.	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GET_RAND
<b>Result Type</b>	: Rand
<b>Comments</b>	: This operation finds the random number of the specified component in the received Facility information element(s)
Description	
This operation extrracts the Rand filed of the component included in the facility message currently processed in the receive queue.	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GSM_CALC_RES (rand: Rand)
<b>Result Type</b>	: Res
<b>Comments</b>	: This operation calculates the response requested for terminal authentication. Input is the random number received in the terminal authentication invoke component in the previously received Facility information element(s)
Description	
This operation calculates the response to be input into component specified in the COMP parameter within the Facility information element(s) specified in the FAC_IE parameter .	
<b>Detailed Comments</b>	:



Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_GSM_CALC_PID (id_type_ : IdentityType)
<b>Result Type</b>	: PortableIdentity
<b>Comments</b>	: This operation calculates the portable identity to be sent in response to an identity request. Input is the identity type provided in the identity request
Description	
<p>This operation calculates the response to be input into component specified in the COMP parameter within the Facility information element(s) specified in the FAC_IE parameter .</p> <p>f id_type_ is unknown then error</p> <p>if id_type_ is IPUI then error</p> <p>if id_type_ is IMSI then PX_IMSI</p> <p>if id_type_ is IMEI then PX_IMEI</p> <p>if id_type_ is TMSI then current value of tsv_tmsi which is updated with every new successful temporary assignment</p> <p>NOTE:</p> <p>The TMSI management requires that the testing is continuous so that the tester can keep track of the assignments. It may be necessary to perform Successful Temporary Assignments in order to "synchronize" tester and IUT. (This means that a test case is executed which ensures that a new TMSI is successfully delivered to the IUT.)</p>	
<b>Detailed Comments</b>	:

Test Suite Operation Definition	
<b>Operation Name</b>	: TSO_RANDOM_INVOKE_ID
<b>Result Type</b>	: INTEGER
<b>Comments</b>	: This operation is used to generate a random invoke identifier value..
Description	
<p>This test suite operation returns a random value of type InvokeIDType.</p> <p>The result of this operation is restricted to values between -32768 and 32767.</p>	
<b>Detailed Comments</b>	:

Test Suite Parameter Declarations			
Parameter Name	Type	PICS/PIXIT Ref	Comments
PC_Network	BOOLEAN	EN 302 094-2, R2.2	Support of network requirements
PC_CFNRC_CTM	BOOLEAN	EN 302 094-2, R1.1	Support the CFRNc SS for the CTM mode
PC_CFNRC_GSM	BOOLEAN	EN 302 094-2, R1.2	Support the CFRNc SS for the DectToGsm mode
PC_CFNRC_NOTIF_DIV	BOOLEAN	I-2, MC 8	support the procedure associated with the notification of the diversion at the calling network
PC_CFNRC_DivertedTo	BOOLEAN	EN 302 094-2, MC 9	support the procedures associated with the release of the diverting number to the diverted-to user
PC_CTM_SR	BOOLEAN	EN 301 144-2, MC5	Support of the subscription registration procedure for CTM.
PX_BASIC	BOOLEAN	PIXIT, Table 1.1	TRUE -> basic access FALSE -> primary rate access
PX_CR_LENGTH	BITSTRING	PIXIT, Table 1.2	Value for bits 8- 1 of the call reference length, Bitstring[8]
PX_CH_NUM	INTEGER	PIXIT, Table 1.3	Preferred channel number (Integer) Possible value: Basic access 1 .. 2 Primary rate access 1 .. 30.
PX_DG_LR_EachTest	BOOLEAN	PIXIT, Table 2.1	True if the location registration procedure shall be executed at the beginning of each test.
PX_CTM_SR_EachTest	BOOLEAN	PIXIT, Table 2.2	True if the CTM Subscription registration procedure shall be executed at the beginning of each test.
PX_CTM_LR_EachTest	BOOLEAN	PIXIT, Table 2.3	True if the CTM location registration procedure shall be executed at the beginning of each test.
PX_WAIT_RESTART	BOOLEAN	PIXIT, Table 2.4	TRUE, if the IUT sends RESTART messages after re-establishment of the multiple frame operation
PX_L2_INIT	BOOLEAN	PIXIT, Table 2.5	TRUE if layer 2 initialisation should take place at the start of each test case

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	Type	PICS/PIXIT Ref	Comments
PX_TWAIT	INTEGER	PIXIT, Table 3.1	Value for timer that controls test events initiated at the IUT via the test operator. (Value in seconds)
PX_TAC	INTEGER	PIXIT, Table 3.2	Value for timer that controls test events initiated by stimuli sent by the tester. (Value in seconds)
PX_TNOAC	INTEGER	PIXIT, Table 3.3	Value for timer that controls the inactivity of the IUT. (Value in seconds)
PX_T_RESTART	INTEGER	PIXIT, Table 3.4	Value for timer that is used to wait for RESTART messages. (Value in seconds). timer used in the initialisation Step only.
PX_ADDR_SERVING_NODE	AddressInformation	PIXIT, Table 4.1	Value of the address of the serving node.
PX_GSM_LA_ID	GSMLocationAreaIdentity	PIXIT, Table 4.2	GSM Location area identity to be used by the tester to send/receive a DectToGsm component
PX_GSM_BASIC_SERVICE	OCTETSTRING	PIXIT, Table 4.3	GSM basic service to be used by the tester to send a DectToGsm component
PX_GSM_TMSI	PortableIdentity	PIXIT, Table 4.4	GSM portable identity, IMSI to be used by the tester as initial TMSI value
PX_GSM_LR_TYPE	LocationRegistrationType	PIXIT, Table 4.5	GSM Location registration type area to be used by the tester to send a DectToGsm component
PX_GSM_CIPHER_INFO	CipherInfo	PIXIT, Table 4.6	GSM CipherInfo to be used by the tester to send a DectToGsm component
PX_GSM_PORT_CAP	PortableCapabilities	PIXIT, Table 4.7	GSM portable capabilities to be used by the tester to send a DectToGsm component
PX_GSM_PORTABLE_ID	PortableIdentity	PIXIT, Table 4.8	GSM portable identity to be used by the tester to send a DectToGsm component

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	Type	PICS/PIXIT Ref	Comments
PX_GSM_PORTABLE_ID1	PortableIdentity	PIXIT, Table 4.9	GSM portable identity to be used by the tester (ON PTC1) to send a DectToGsm component
PX_GSM_PORTABLE_ID2	PortableIdentity	PIXIT, Table 4.10	GSM portable identity to be used by the tester (ON PTC2) to send a DectToGsm component
PX_GSM_KEYPAD_REGISTRATION	KeypadIE	PIXIT, Table 4.11	GSM keypad information to be sent to the IUT to request registration to CFNRc.
PX_GSM_KEYPAD_REGISTRATION1	KeypadIE	PIXIT, Table 4.12	GSM keypad information to be sent to the IUT to request registration to CFNRc for PTC1.
PX_GSM_KEYPAD_ERASURE	KeypadIE	PIXIT, Table 4.13	GSM keypad information to be sent to the IUT to request erasure to CFNRc.
PX_GSM_KEYPAD_ACTIVATION	KeypadIE	PIXIT, Table 4.14	GSM keypad information to be sent to the IUT to request activation to CFNRc.
PX_GSM_KEYPAD_DEACTIVATION	KeypadIE	PICXIT, Table 4.15	GSM keypad information to be sent to the IUT to request deactivation to CFNRc.
PX_GSM_KEYPAD_INTERROGATION	KeypadIE	PIXIT, Table 4.16	GSM keypad information to be sent to the IUT to request interrogation to CFNRc.
PX_CTM_FIXED_IDENTITY	FixedIdentity	PIXIT, Table 5.1	Value of the CTM fixed Identity
PX_CTM_IPUI	IPUI	PIXIT, Table 5.2	Value of the CTM IPUI of the user
PX_CTM_IPUI1	IPUI	PIXIT, Table 5.3	Value of the CTM IPUI for PTC1
PX_CTM_IPUI2	IPUI	PIXIT, Table 5.4	Value of the CTM IPUI for PTC2
PX_CTM_BASIC_SERVICE	OCTETSTRING	PIXIT, Table 5.5	CTM basic service to be used by the tester to send a a setup message with CTMOutgoingCallMMInfo invoke component.

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	Type	PICS/PIXIT Ref	Comments
PX_CTM_AuthType	OCTETSTRING	PIXIT, Table 5.6	value of the authentication type supported by IUT (used in a CTMNetworkAuthentication procedure, see encoding in ETS 300 175–5, 7.7.4)
PX_CTM_PORT_CAP	PortableCapabilities	PIXIT, Table 5.7	CTM portable capabilities to be used by the tester to send a CTM component
PX_CTM_Rand	OCTETSTRING	PIXIT, Table 5.8	value of a Rand supported by IUT (used in a CTMNetworkAuthentication procedure, see encoding in ETS 300 175–5, 7.7.7.32)
PX_CTM_NEW_LA	CTMLocationAreaIdentity	PIXIT, Table 5.9	CTM new location area used by the tester to send a CTM component
PX_CTM_KEYPAD_REGISTRATION	KeypadIE	PICXIT, Table 5.10	CTM keypad information to be sent to the IUT to request registration to CFNRc.
PX_CTM_KEYPAD_REGISTRATION1	KeypadIE	PIXIT, Table 5.11	CTM keypad information to be sent to the IUT to request registration to CFNRc for PTC1.
PX_CTM_KEYPAD_ERASURE	KeypadIE	PIXIT, Table 5.12	CTM keypad information to be sent to the IUT to request erasure to CFNRc.
PX_CTM_KEYPAD_ACTIVATION	KeypadIE	PIXIT, Table 5.13	CTM keypad information to be sent to the IUT to request activation to CFNRc.
PX_CTM_KEYPAD_DEACTIVATION	KeypadIE	PIXIT, Table 5.14	CTM keypad information to be sent to the IUT to request deactivation to CFNRc.
PX_CTM_KEYPAD_INTERROGATION	KeypadIE	PIXIT, Table 5.15	CTM keypad information to be sent to the IUT to request interrogation to CFNRc.
PX_CPN	OCTETSTRING	PIXIT, Table 6.1	Number digits (IA5) for the Called party number information element to be sent to the IUT

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	Type	PICS/PIXIT Ref	Comments
PX_CDPN_OCTET3	OCTETSTRING	PIXIT, Table 6.2	Octet 3 (Type of number, Numbering plan identification) of the Called party number information elements to be sent to the IUT
PX_LCPN	OCTETSTRING	PIXIT, Table 6.3	Length of the Called party number information element to be sent to the IUT including the number digits
PX_BCAPL	OCTETSTRING	PIXIT, Table 6.4	Length of the bearer capability to be sent by the test system (speech or 3,1kHz audio)
PX_BCAPV	OCTETSTRING	PIXIT, Table 6.5	Value of the bearer capability to be sent by the test system (speech or 3,1kHz audio)
Detailed Comments :			

Test Case Selection Expression Definitions		
Expression Name	Selection Expression	Comments
CFNRc_Network	PC_Network AND ( PC_CFNRc_CTM OR PC_CFNRc_GSM )	TRUE if Network requirement and CFNRc for CTM orDectToGsm is supported
CFNRc_notif_calling	PC_Network AND PC_CFNRc_NOTIF_DIV	TRUE if the IUT support notification to the calling n user
CFNRc_DivertedTo	PC_Network AND PC_CFNRc_DivertedTo	TRUE if IUT support the procedures associated with the release of the diverting number to the diverted-to user
Detailed Comments :		

Test Suite Constant Declarations			
Constant Name	Type	Value	Comments
TSC_CTMAccessRightRq	Operation	globalValue {TSC_mMOID (1)}	
TSC_CTMTerminalAuth	Operation	globalValue {TSC_mMOID (6)}	
TSC_CTM Ciphering	Operation	globalValue {TSC_mMOID (8)}	
TSC_CTMIncomingCallMM	Operation	globalValue { TSC_mMOID ( 13)}	
TSC_CTMKeyAllocate	Operation	globalValue {TSC_mMOID (10)}	
TSC_CTMLocationReg	Operation	globalValue {TSC_mMOID (3)}	
TSC_CTMNetworkAuth	Operation	globalValue {TSC_mMOID (7)}	
TSC_CTMOutgoingCallMM	Operation	globalValue { TSC_mMOID ( 12)}	
TSC_GSMAssignIdentity	Operation	globalValue { TSC_mMOID ( 26)}	
TSC_GSMIncomingCallMM	Operation	globalValue { TSC_mMOID ( 30)}	
TSC_GSMLinkedAssignId	Operation	globalValue { TSC_mMOID ( 27)}	
TSC_GSMLocationReg	Operation	globalValue { TSC_mMOID ( 20)}	
TSC_GSMOutgoingCallMM	Operation	globalValue { TSC_mMOID ( 29)}	
TSC_GSMTerminalAuth	Operation	globalValue { TSC_mMOID ( 23)}	
TSC_GSM Ciphering	Operation	globalValue { TSC_mMOID ( 25)}	
TSC_GSMIdentityRequest	Operation	globalValue { TSC_mMOID ( 28)}	
TSC_E_Congestion	Error	globalValue {TSC_mMOID (44)}	
TSC_E_PagingFailure	Error	globalValue {TSC_mMOID (46)}	
TSC_E_RadioConnectionFailure	Error	globalValue {TSC_mMOID (47)}	
TSC_EncapsulatedStimulus	Operation	globalValue { TSC_cFNRCOID ( 1)}	
TSC_cFNRCOID	OID	{ 0 4 0 2094 1 }	
TSC_mMOID	OID	{ 0 4 0 1144 1 }	
TSC_ID_BCAP	BITSTRING	'00000100'B	Bearer capability
TSC_ID_CAU	BITSTRING	'00001000'B	Cause

Continued on next page



Continued from previous page

Test Suite Constant Declarations			
Constant Name	Type	Value	Comments
TSC_ID_CDPN	BITSTRING	'01110000'B	Called party number
TSC_ID_CHI	BITSTRING	'00011000'B	Channel identification
TSC_ID_NOID	BITSTRING	'00100111'B	Notification indicator
TSC_ID_RI	BITSTRING	'01111001'B	Restart indicator
TSC_ID_SCI	BITSTRING	'10100001'B	Sending complete
TSC_MT_ALERTING	BITSTRING	'00000001'B	
TSC_MT_CALL_PROC	BITSTRING	'00000010'B	
TSC_MT_CONNECT	BITSTRING	'00000111'B	
TSC_MT_CONNECT_ACK	BITSTRING	'00001111'B	
TSC_MT_DISCONNECT	BITSTRING	'01000101'B	
TSC_MT_FAC	BITSTRING	'01100010'B	
TSC_MT_INFORMATION	BITSTRING	'01111011'B	
TSC_MT_NOTIFY	BITSTRING	'01101110'B	
TSC_MT_PROGRESS	BITSTRING	'00000011'B	
TSC_MT_RELEASE	BITSTRING	'01001101'B	
TSC_MT_RELEASE_COM	BITSTRING	'01011010'B	
TSC_MT_RESTART	BITSTRING	'01000110'B	
TSC_MT_RESTART_ACK	BITSTRING	'01001110'B	
TSC_MT_SETUP	BITSTRING	'00000101'B	
TSC_MT_SETUP_ACK	BITSTRING	'00001101'B	
TSC_MT_STATUS_ENQ	BITSTRING	'01110101'B	
TSC_Q931_PD	BITSTRING	'00001000'B	
Detailed Comments :			

Test Case Variable Declarations			
Variable Name	Type	Value	Comments
tcv_tmsi	PortableIdentity	PX_GSM_TMSI	Value of the TMSI received, initialized from PIXIT.
tcv_cipherinfo	CipherInfo		Value of the cipherinfo parameter received
tcv_config1	BOOLEAN	FALSE	TRUE if the test case is in CONFIG1 if the test case is in CONFIG 0 then FALSE
tcv_ctm_authtype	AuthType		Value of the AuthType field received .
tcv_ctm_alloctype	AllocType		Value of the AllocType field received .
tcv_ctm_rs	Rs		Value of the Rs field received .
tcv_ctm_xres1	Res		Value of the computed xres1 .
tcv_cref	BIT7OR15		Call reference value
tcv_cref1	BIT7OR15		Call reference value for PTC1
tcv_cref2	BIT7OR15		Call reference value for PTC2
tcv_glob_cref	BIT7OR15		Global call reference value
tcv_glob_cref1	BIT7OR15		Global call reference value for PTC1
tcv_idtype	IdentityType		Value of the identity type received .
tcv_invokeld	INTEGER		Value of the invoke Identifier.
tcv_invokeld2	INTEGER		Value of the invoke Identifier.
tcv_invokeld3	INTEGER		Value of the third invoke Identifier for embedded procedures.
tcv_flag_r	INTEGER		Value of the flag in a PDU received by the tester.
tcv_flag_s	INTEGER		Value of the flag in a PDU sent by the tester.
tcv_rand	Rand		Value of the random number received .
tcv_rcv_idreq	BOOLEAN	FALSE	Value of the "Identity Request invoke received" as embedded procedure
tcv_rcv_lidass	BOOLEAN	FALSE	Value of the "Linked Identity Assignment invoke received" as embedded procedure

Continued on next page

Continued from previous page

Test Case Variable Declarations			
Variable Name	Type	Value	Comments
tcv_rcv_locreg	BOOLEAN	FALSE	Value of "Location Registration component received" as embedded procedure. RR, RJ or RE depending upon the context.
tcv_rcv_niciph	BOOLEAN	FALSE	Value of the "Network Initiated Ciphering invoke received" as embedded procedure
tcv_rcv_termauth	BOOLEAN	FALSE	Value of "Terminal Authentication invoke received" as embedded procedure
tcv_rcv_timeout	BOOLEAN	FALSE	Value of "Timeout received" during embedded procedure
tcv_rcv_tmpidass	BOOLEAN	FALSE	Value of "Temporary Identity Assignment invoke received" as embedded procedure
tcv_stop_flag	BOOLEAN	FALSE	Value used to stop a loop
tcv_bch_num	BITSTRING		B-channel for call, BITSTRING[7..8] (1)
tcv_bch_num1	BITSTRING		B-channel for call, BITSTRING[7..8] (1) for PTC1
tcv_bch_num2	BITSTRING		B-channel for call, BITSTRING[7..8] (1) for PTC2
Detailed Comments :			

PCO Type Declarations		
PCO Type	Role	Comments
OPERATOR SAP	UT LT	
Detailed Comments :		

PCO Declarations			
PCO Name	PCO Type	Role	Comments
O	OPERATOR	UT	(2)
O1	OPERATOR	UT	(2) used by PTC1
O2	OPERATOR	UT	(2) used by PTC2
L	SAP	LT	SAP at the lower tester controlling and observing the exchange of call control PDUs (messages) on the layer 3 D-channel. The lower tester is the user of the data link layer service.
L1	SAP	LT	SAP at the lower tester controlling and observing the exchange of call control PDUs (messages) on the ISDN layer 3 D-channel. The lower tester is the user of the data link layer service.
L2	SAP	LT	SAP at the lower tester controlling and observing the exchange of call control PDUs (messages) on the ISDN layer 3 D-channel. The lower tester is the user of the data link layer service.
<b>Detailed Comments :</b> (1) SAP at the lower tester controlling and observing the exchange of call control PDUs (messages) on the ISDN layer 3 D-channel. The lower tester is the user of the data link layer service. (2) SAP at the upper tester observing messages displayed on the lower tester's screen and controlling the initiation of test events at the IUT. The upper tester is the test operator.			

Coordination Point Declarations	
CP Name	Comments
CPA1	CP: MTCA – PTC1
CPA2	CP: MTCA – PTC2
Detailed Comments :	

Timer Declarations			
Timer Name	Duration	Unit	Comments
T_AC	PX_TAC	s	to control an event initiated by a stimuli sent by the tester.
T_NOAC	PX_TNOAC	s	to control the inactivity of the IUT. (Value in seconds)
T_RESTART	PX_T_RESTART	s	to control the receipt of a RESTART message
T_WAIT	PX_TWAIT	s	to control test events initiated at the IUT by the test operator.
Detailed Comments :			

Test Component Declarations				
Component Name	Component Role	Nr PCOs	Nr CPs	Comments
MTCA	MTC	2	2	main test component
PTC1	PTC	2	1	1st parallel test component
PTC2	PTC	2	1	2nd parallel test component
Detailed Comments :				



Test Components Configuration Declaration			
<b>Configuration Name</b> : CONFIG0			
<b>Comments</b> :			
Components Used	PCOs Used	CPs Used	Comments
MTCA	L,O		
<b>Detailed Comments</b> : &COMMON_N12			

Test Components Configuration Declaration			
<b>Configuration Name</b> : CONFIG1			
<b>Comments</b> :			
Components Used	PCOs Used	CPs Used	Comments
MTCA	L,O	CPA1, CPA2	
PTC1	L1,O1	CPA1	
PTC2	L2,O2	CPA2	
<b>Detailed Comments</b> : &COMMON_N12			

ASP Type Definition		
<b>ASP Name</b> : DL_EST_RQ (DL-ESTABLISH-REQUEST) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to request the establishment of multiple frame operation (L3 ----> L2).		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_EST_IN (DL-ESTABLISH-INDICATION) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to indicate the establishment of multiple frame operation (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_EST_CO (DL-ESTABLISH-CONFIRM) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to confirm the establishment of multiple frame operation (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_REL_RQ (DL-RELEASE-REQUEST) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to request the termination of an established multiple frame operation (L3 ----> L2).		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_REL_IN (DL-RELEASE-INDICATION) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to confirm the termination of an established multiple frame operation or to report an unsuccessful establishment attempt (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_REL_CO (DL-RELEASE-CONFIRM) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to confirm the termination of an established multiple frame operation (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_DAT_RQ (DL-DATA-REQUEST) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to request the transmission of layer 3 PDUs using acknowledged operation (L3 ----> L2).		
Parameter Name	Parameter Type	Comments
mun (Message unit)	PDU	Network layer (peer-to-peer message) PDU.
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_DAT_IN (DL-DATA-INDICATION) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to indicate the receipt of layer 3 PDUs using acknowledged operation (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
mun (Message unit)	PDU	Network layer (peer-to-peer message) PDU.
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_DAT_IN_RESTART (DL-DATA-INDICATION) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to indicate the receipt of RESTART PDUs using acknowledged operation (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
mun (Message unit)	RESTART_PDU	Network layer (peer-to-peer message) PDU.
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_DAT_IN_SETUP (DL-DATA-INDICATION) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) This ASP is used to indicate the receipt of SETUP PDUs using acknowledged operation (L2 ----> L3).		
Parameter Name	Parameter Type	Comments
mun (Message unit)	SETUP_PDU	Network layer (peer-to-peer message) PDU.
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DL_UDAT_IN (DL-UDATA-INDICATION) <b>PCO Type</b> : SAP <b>Comments</b> : CEId: = (SAPI,CES) mapped onto DLCI: = (SAPI,TEI) ASP is used to indicate the receipt of layer 3 pdus using unacknowledged operation (L2 ----> L3)		
Parameter Name	Parameter Type	Comments
mun (Message unit)	PDU	Network layer (peer-to-peer message) PDU.
<b>Detailed Comments</b> :		

ASP Type Definition		
<b>ASP Name</b> : DISPLAY		
<b>PCO Type</b> : OPERATOR		
<b>Comments</b> : This ASP is used to indicate to operator actions which are to be performed on the SUT.		
Parameter Name	Parameter Type	Comments
msg	IA5String	message destined to the operator
<b>Detailed Comments</b> :		

PDU Type Definition			
<b>PDU Name</b> : ALERTING_PDU (ALERTING) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.1			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
bcap	BCAP		Bearer capability Direction: both, type: O, length: 4 – 12 octets
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
chi	CHI		Channel identification Direction: u>n , type: O, length: 2 – 34 octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
pi	PI		Progress indicator Direction: both, type: O, length: 2 – 4 octets
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
ronn	RONN		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
hlc	HLC		High layer compatibility Direction: both, type: O, length: 2 – 4 octets

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	Field Type	Field Encoding	Comments
uui	UUI		User-user Direction: both, type: O, length: 2 – * octets
Detailed Comments :			

PDU Type Definition			
<b>PDU Name</b> : ANY_MSG_PDU <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : u -> n			
Field Name	Field Type	Field Encoding	Comments
pd	PD		protocol discriminator M
cr	CR		call reference M OCTETSTRING[1..3]
mt	ANY_MT_LIST		message type M
ie_list	OCTETSTRING		information element(s)
Detailed Comments : &COMMON_N12			

PDU Type Definition			
<b>PDU Name</b> : CALL_PROC_PDU (CALL PROCEEDING) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.1.2			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
bcap	BCAP		Bearer capability Direction: both, type: O, length: 4 – 12 octets
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
chi	CHI		Channel identification Direction: both, type: O, length: 2 – 34 octets (1)
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
pi	PI		Progress indicator Direction: both, type: O, length: 2 – 4 octets
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
hlc	HLC		High layer compatibilty Direction: both, type: O, length: 2 – 4 octets
<b>Detailed Comments</b> : (1) Mandatory in the network-to-user direction.			



PDU Type Definition			
<b>PDU Name</b> : CONNECT_PDU (CONNECT) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.3			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
bcap	BCAP		Bearer capability Direction: both, type: O, length: 4 – 12 octets
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
chi	CHI		Channel identification Direction: u>n , type: O, length: 2 – 34 octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
pi	PI		Progress indicator Direction: both, type: O, length: 2 – 4 octets
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
dati	DATI		Date/time Direction: n>u , type: O, length: 2 – 7 octets
codn	CODN		Connected number Direction: both, type: O, length: 2 – 24 octets

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	Field Type	Field Encoding	Comments
cods	CODS		Connected subaddress Direction: both, type: O, length: 2 – 23 octets
ronn	RONN		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
llc	LLC		Low layer compatibilty Direction: both, type: O, length: 2 – 16 octets
hlc	HLC		High layer compatibilty Direction: both, type: O, length: 2 – 4 octets
uui	UUI		User-user Direction: both, type: O, length: 2 – * octets
Detailed Comments :			

PDU Type Definition			
<b>PDU Name</b> : CONNECT_ACK_PDU (CONNECT ACKNOWLEDGE) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.4			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
<b>Detailed Comments</b> :			

PDU Type Definition			
<b>PDU Name</b> : DISCONNECT_PDU (DISCONNECT) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.5			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
cau	CAU		Cause Direction: both, type: M, length: 4 – 32 octets
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
pi	PI		Progress indicator Direction: n>u , type: O, length: 2 – 4 octets
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
uui	UUI		User-user Direction: both, type: O, length: 2 – * octets
<b>Detailed Comments</b> :			

PDU Type Definition			
<b>PDU Name</b> : GFP_MSG_PDU <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : ETS 300 196-1 subclause 11			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	GFP_MT_LIST		Message type Direction: both, type: M, length: 1 octet
ie_list	IE_LIST		Direction: both, type: O, length: 1 – * octets
<b>Detailed Comments</b> :			

PDU Type Definition			
<b>PDU Name</b> : INFORMATION_PDU (INFORMATION) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.1.6			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
sci	SCI		Sending complete Direction: both, type: O, length: 1 octet (1)
cau	CAU		Cause Direction: n>u , type: O, length: 4 – 32 octets
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
kpf	KPF		Keypad facility Direction: u>n , type: O, length: 2 – 34 octets
cdpn	CDPN		Called party number Direction: both, type: O, length: 2 – 23 octets
ronn	RONN		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
<b>Detailed Comments</b> : (1) The Sending complete information element may be located at any position in the message.			

PDU Type Definition			
<b>PDU Name</b> : NOTIFY_PDU (NOTIFY) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: access Direction: both ETS 300 403-1 subclause 3.1.7			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
noid	NOID		Notification indicator Direction: both, type: M, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
ronn	RONN		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
<b>Detailed Comments</b> :			

PDU Type Definition			
<b>PDU Name</b> : PROGRESS_PDU (PROGRESS) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.8			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 2 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
bcap	BCAP		Bearer capability Direction: both, type: O, length: 2 – 12 octets
cau	CAU		Cause Direction: both, type: O, length: 2 – 32 octets
pi	PI		Progress indicator Direction: n>u , type: M, length: 4 octets
dsp	DSP		Direction: n>u , type: O, length: 2 – 82 octets
ronn	RONN		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
hlc	HLC		High layer compatilbity Direction: both, type: O, length: 2 – 5 octets
<b>Detailed Comments</b> :			



PDU Type Definition			
<b>PDU Name</b> : RELEASE_PDU (RELEASE) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.1.9			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
cau	CAU		Cause Direction: both, type: O, length: 4 – 32 octets (1)
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
uui	UUI		User-user Direction: both, type: O, length: 2 – * octets
<b>Detailed Comments</b> : (1) Mandatory in the first call clearing message, including when the RELEASE message is sent as a result of an error handling condition.			

PDU Type Definition			
<b>PDU Name</b> : RELEASE_COM_PDU (RELEASE COMPLETE) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.1.10			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
cau	CAU		Cause Direction: both, type: O, length: 4 – 32 octets (1)
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
uui	UUI		User-user Direction: both, type: O, length: 2 – * octets
<b>Detailed Comments</b> : (1) Mandatory in the first call clearing message, including when the RELEASE message is sent as a result of an error handling condition.			

PDU Type Definition			
<b>PDU Name</b> : RESTART_ACK_PDU (RESTART ACKNOWLEDGE) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.4.2			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
chi	CHI		Channel identification Direction: both, type: O, length: 2 – 34 octets
chi_rs	CHI_RS		Channel identification Direction: both, type: O, length: 2 – 34 octets (1)
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
ri	RI		Restart indicator Direction: both, type: M, length: 3 octets
<b>Detailed Comments</b> : (1) This special Channel identification information element type is used to handle the restart procedures.			

PDU Type Definition			
<b>PDU Name</b> : RESTART_PDU (RESTART) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.4.1			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
chi	CHI		Channel identification Direction: both, type: O, length: 2 – 34 octets
chi_rs	CHI_RS		Channel identification Direction: both, type: O, length: 2 – 34 octets (1)
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
ri	RI		Restart indicator Direction: both, type: M, length: 3 octets
<b>Detailed Comments</b> : (1) This special Channel identification information element type is used to handle the restart procedures.			

PDU Type Definition			
<b>PDU Name</b> : SETUP_PDU (SETUP) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.14			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
sci	SCI		Sending complete Direction: both, type: O, length: 1 octet (1)
bcap	BCAP		Bearer capability Direction: both, type: M, length: 4 – 12 octets
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
chi	CHI		Channel identification Direction: both, type: O, length: 2 – 34 octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
pi	PI		Progress indicator Direction: both, type: O, length: 2 – 4 octets
nsf	NSF		Network-specific facilities Direction: both, type: O, length: 2 – * octets
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets

Continued on next page

Continued from previous page

PDU Type Definition			
Field Name	Field Type	Field Encoding	Comments
kpf	KPF		Keypad facility Direction: u>n , type: O, length: 2 – 34 octets
cgpn	CGPN		Calling party number Direction: both, type: O, length: 2 – 24 octets
cgps	CGPS		Calling party subaddress Direction: both, type: O, length: 2 – 23 octets
cdpn	CDPN		Called party number Direction: both, type: O, length: 2 – 23 octets
cdps	CDPS		Called party subaddress Direction: both, type: O, length: 2 – 23 octets
rngn	RNGN		Redirecting number Direction: n>u , type: O, length: 2 – 24 octets
tns	TNS		Transfer network selection Direction: u>n , type: O, length: 2 – * octets
llc	LLC		Low layer compatibility Direction: both, type: O, length: 2 – 16 octets
hlc	HLC		High layer compatibility Direction: both, type: O, length: 2 – 4 octets
uui	UUI		User-user Direction: both, type: O, length: 2 – * octets
lsh4	LOCK_SHIFT_TYPE		Locking shift Direction: both , type: O, length: 1 octets
<b>Detailed Comments</b> : (1) The Sending complete information element may be located at any position in the message. (2) Bearer capability and High layer compatibility information elements may be repeated, if fallback to an alternative service is allowed. For the repeated Bearer capability information element two different types are used for sending and receiving.			

PDU Type Definition			
<b>PDU Name</b> : SETUP_ACK_PDU (SETUP ACKNOWLEDGE) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: global Direction: both ETS 300 403-1 subclause 3.1.15			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
efac	EFAC		Extended facility Direction: both, type: O, length: 2 – * octets
chi	CHI		Channel identification Direction: both, type: O, length: 2 – 34 octets
fac	FIES		Facility Direction: both, type: O, length: 2 – * octets
fac2	FA		facility in tabular form
pi	PI		Progress indicator Direction: both, type: O, length: 2 – 4 octets
noid	NOID		Notification indicator Direction: both, type: O, length: 2 – * octets
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
<b>Detailed Comments</b> :			

PDU Type Definition			
<b>PDU Name</b> : STATUS_ENQ_PDU (STATUS ENQUIRY) <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Significance: local Direction: both ETS 300 403-1 subclause 3.1.17			
Field Name	Field Type	Field Encoding	Comments
pd	PD		Protocol discriminator Direction: both, type: M, length: 1 octet
cr	CR		Call reference Direction: both, type: M, length: 1 – 3 octets
mt	MT		Message type Direction: both, type: M, length: 1 octet
dsp	DSP		Display Direction: n>u , type: O, length: 2 – 82 octets
<b>Detailed Comments</b> :			

PDU Type Definition			
<b>PDU Name</b> : FAC_PDU <b>PCO Type</b> : SAP <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : FACility u <-> n, EN 301 061-1, subclause 11.1.1.1			
Field Name	Field Type	Field Encoding	Comments
pd	PD		protocol discriminator M
cr	CR		call reference M
mt	MT		message type M
fac	FIES		facility M
fac2	FA		facility in tabular form
noid	NOID		notification indicator O
<b>Detailed Comments</b> :			



CM Type Definition		
<b>CM Name</b> : CP_M <b>Comments</b> : coordination message		
Parameter Name	Parameter Type	Comments
CM_content	IA5String [0 TO 26]	message content in clear text
<b>Detailed Comments</b> :		

Alias Definitions		
Alias Name	Expansion	Comments
PDU <sub>s</sub>	DL_DAT_RQ	PDU sent, point-to-point data link
PDU <sub>r</sub>	DL_DAT_IN	PDU received
RESTART <sub>r</sub>	DL_DAT_IN_RESTART	RESTART received
SETUP <sub>r</sub>	DL_DAT_IN_SETUP	SETUP received, point-to-point
UPDU <sub>r</sub>	DL_UDAT_IN	PDU received in unacknowledge mode
Detailed Comments :		

# **III**

## **Constraints Part**

Structured Type Constraint Declaration			
<b>Constraint Name</b> : BCAP_S1 <b>Structured Type</b> : BCAP <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send constraint; values as given in the test suite parameters			
Element Name	Element Value	Element Encoding	Comments
bcap_i	TSC_ID_BCAP		Identifier
bcap_l	'02'O		
bcap_con	'A880'O		(1)
<b>Detailed Comments</b> : (1) coding standard = ISO/IEC, transfer capability = unrestricted digital information transfer mode = NCICS, transfer rate = NCICS.			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : BCAP_S2 <b>Structured Type</b> : BCAP <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send constraint; values as given in the test suite parameters			
Element Name	Element Value	Element Encoding	Comments
bcap_i	TSC_ID_BCAP		Identifier
bcap_l	PX_BCAPL		
bcap_con	PX_BCAPV		
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CallRef_1(flag_: INTEGER; call_ref_: BIT7OR15) <b>Structured Type</b> : CR <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Constraint for sending and receiving			
Element Name	Element Value	Element Encoding	Comments
cr_l	PX_CR_LENGTH		Length value
cr_f	INT_TO_BIT(flag_,1)		Parametrized flag
cr_r	call_ref_		Parametrized value
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CallRef_R1 <b>Structured Type</b> : CR <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive constraint with any call reference value			
Element Name	Element Value	Element Encoding	Comments
cr_l	PX_CR_LENGTH		Length value, bits 4 – 1 (1)
cr_f	'0'B		Originator
cr_r	?		Call reference value present
<b>Detailed Comments</b> : (1) PX_CR_LENGTH is a test suite parameter.			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : Cause_R2 (cval_: INTEGER) <b>Structured Type</b> : CAU <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send constraint with parametrized cause value			
Element Name	Element Value	Element Encoding	Comments
cau_i	TSC_ID_CAU		Cause identifier
cau_l	?		Length value present
cau_e3_eb	?		Extension bit present
cau_e3_cs	'000'B		CCITT standardised coding
cau_e3_loc	?		Location value present
cau_e4_rec	*		Any or no recommendation value
cau_e5_eb	'1'B		Extension bit present
cau_e5_cv	INT_TO_BIT(cval_, 7)		Parametrized cause value
cau_di	*		Any or no diagnostics
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : Cause_S1(cval_: INTEGER) <b>Structured Type</b> : CAU <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send constraint with parametrized cause value			
Element Name	Element Value	Element Encoding	Comments
cau_i	TSC_ID_CAU		Cause identifier
cau_l	'00000010'B		Length value present
cau_e3_eb	'1'B		Extension bit present
cau_e3_cs	'000'B		CCITT standardised coding
cau_e3_loc	'0000'B		Location user
cau_e4_rec	–		No recommendation value
cau_e5_eb	'1'B		Extension bit present
cau_e5_cv	INT_TO_BIT(cval_, 7)		Parametrized cause value
cau_di	–		No diagnostics value
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CDPN_S1 <b>Structured Type</b> : CDPN <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send constraint; values as given in the test suite parameters			
Element Name	Element Value	Element Encoding	Comments
cdpn_i	TSC_ID_CDPN		Identifier
cdpn_l	PX_LCPN		Length present
cdpn_e3_npi	PX_CDPN_OCTET3		Type of number and Numbering plan identification present
cdpn_e4_nd	PX_CPN		Number digits present
<b>Detailed Comments</b> : PX_LCPN, PX_CDPN_OCTET3 and PX_CPN are test suite parameters			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CHI_basic_S1 <b>Structured Type</b> : CHI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Channel Id IE for a NCICS with basic rate interface.			
Element Name	Element Value	Element Encoding	Comments
chi_i	TSC_ID_CHI		Identifier
chi_l	'00000001'B		Length value present
chi_e3_p1	'1000'B		Extension bit present
chi_e3_pe	'1'B		(1)
chi_e3_p3	'100'B		no channel
chi_e3_cs	–		
chi_e4	–		Not present
chi_e5_ch1	–		Not present
chi_e5_ch2	–		Not present
<b>Detailed Comments</b> : (1) Interface implicitly identified, basic interface, exclusive: only the indicated channel is acceptable, the channel identified is the D–channel			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CHI_primary_S1 <b>Structured Type</b> : CHI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Channel Id IE for a NCICS with primary rate interface.			
Element Name	Element Value	Element Encoding	Comments
chi_i	TSC_ID_CHI		Identifier
chi_l	'00000001'B		Length value present
chi_e3_p1	'1010'B		Extension bit present
chi_e3_pe	'1'B		(1)
chi_e3_p3	'100'B		no channel
chi_e3_cs	–		
chi_e4	–		Not present
chi_e5_ch1	–		Not present
chi_e5_ch2	–		Not present
<b>Detailed Comments</b> : (1) Interface implicitly identified, primary interface, exclusive: only the indicated channel is acceptable, the channel identified is the D–channel			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CHI_basic_S2 ( bch_ : BITSTRING) <b>Structured Type</b> : CHI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Channel Id IE for basic access with parametrized channel selection			
Element Name	Element Value	Element Encoding	Comments
chi_i	TSC_ID_CHI		Identifier
chi_l	'00000001'B		Length value present
chi_e3_p1	–		Extension bit present
chi_e3_pe	–		(1)
chi_e3_p3	–		Parametrized channel selection
chi_e3_cs	bch_		
chi_e4	–		Not present
chi_e5_ch1	–		Not present
chi_e5_ch2	–		Not present
<b>Detailed Comments</b> : (1) Interface implicitly identified, basic interface, exclusive: only the indicated channel is acceptable, the channel identified is not the D–channel			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CHI_primary_S2 ( bch_ : BITSTRING) <b>Structured Type</b> : CHI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Channel Id IE for a primary rate interface.			
Element Name	Element Value	Element Encoding	Comments
chi_i	TSC_ID_CHI		Identifier
chi_l	'00000011'B		Length value present
chi_e3_p1	'1010'B		Extension bit present
chi_e3_pe	'1'B		(1)
chi_e3_p3	'001'B		Channel as indicated
chi_e3_cs	–		
chi_e4	'10000011'B		(2)
chi_e5_ch1	'1'B		Extension bit present
chi_e5_ch2	bch_		Parametrized channel number
<b>Detailed Comments</b> : (1) Interface implicitly identified, other interface, exclusive: only the indicated channel is acceptable, the channel identified is not the D–channel (2) CCITT standardized coding, channel(s) is/are indicated by the number(s) in the following octet(s), B–channel units			



Structured Type Constraint Declaration			
<b>Constraint Name</b> : CHI_basic_R1 <b>Structured Type</b> : CHI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> :			
Element Name	Element Value	Element Encoding	Comments
chi_i	'00011000'B		Identifier
chi_l	'00000001'B		Length value present
chi_e3_p1	–		Not present
chi_e3_pe	–		Not present
chi_e3_p3	–		Not present
chi_e3_cs	?		Channel selection present
chi_e4	–		Not present
chi_e5_ch1	–		Not present
chi_e5_ch2	–		Not present
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : CHI_primary_R1 <b>Structured Type</b> : CHI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> :			
Element Name	Element Value	Element Encoding	Comments
chi_i	'00011000'B		Identifier
chi_l	'00000011'B		Length value present
chi_e3_p1	'1010'B		
chi_e3_pe	'1'B		
chi_e3_p3	'001'B		
chi_e3_cs	–		Not present
chi_e4	'10000011'B		
chi_e5_ch1	'1'B		Extension bit present
chi_e5_ch2	?		Channel selection present
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : NOID_R1 <b>Structured Type</b> : NOID <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive constraint containing any notification description			
Element Name	Element Value	Element Encoding	Comments
noid_i	TSC_ID_NOID		Identifier
noid_l	?		Length present
noid_e3_nd	?		Notification description present
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : NOID_2 (val : BITSTRING) <b>Structured Type</b> : NOID <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Notification indicator information element			
Element Name	Element Value	Element Encoding	Comments
noid_i	'00100111'B		
noid_l	'00000001'B		
noid_e3_nd	val		(1)
<b>Detailed Comments</b> : &COMMON_N12 (1) Any value acceptable for the Notification description.			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : RestartInd1(class_val_: INTEGER) <b>Structured Type</b> : RI <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Constraint with parametrized class value used for sending and receiving.			
Element Name	Element Value	Element Encoding	Comments
ri_i	TSC_ID_RI		Identifier
ri_l	'00000001'B		Length present
ri_cl	'10000'B		Spare value
ri_cl1	INT_TO_BIT(class_val_,3)		Parametrized class value
<b>Detailed Comments</b> :			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : RNGN1 <b>Structured Type</b> : RNGN <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Info Element RedirectiNG Number EN 300 207-1 subclause 7.2.2			
Element Name	Element Value	Element Encoding	Comments
rngn_i	'01110100'B		Identifier
rngn_l	?		Length
rngn_e3_ton	( '0000'B , '0001'B , '0010'B )		Type of number
rngn_e3_npi	( '0000'B , '0001'B )		Numbering plan identifier
rngn_e4_pi	'000'B		Presentation indicator – allowed
rngn_e4_sp	'00000'B		Spare
rngn_e5_sp	'1000'B		Spare
rngn_e5_rfd	'1111'B		Reason for diversion
rngn_e6_nd	?		Number digits
<b>Detailed Comments</b> : &COMMON_N12			

Structured Type Constraint Declaration			
<b>Constraint Name</b> : RNGN2 <b>Structured Type</b> : RNGN <b>Derivation Path</b> : <b>Encoding Variation</b> : <b>Comments</b> : Info Element RedirectiNG Number EN 300 207-1 subclause 7.2.2			
Element Name	Element Value	Element Encoding	Comments
rngn_i	'01110100'B		Identifier
rngn_l	'03'O		Length
rngn_e3_ton	'0000'B		Type of number
rngn_e3_npi	( '0000'B , '0001'B )		Numbering plan identifier
rngn_e4_pi	'001'B		Presentation indicator – restricted
rngn_e4_sp	'00000'B		Spare
rngn_e5_sp	'1000'B		Spare
rngn_e5_rfd	'1111'B		Reason for diversion
rngn_e6_nd	–		Number digits
<b>Detailed Comments</b> : &COMMON_N12			

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_AccessRightRq_Invoke ( invokeld_ : INTEGER ; ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMAccessRightsRequest_Components   cTMAccessRightsRequest_InvokeCpt {   invokeID      invokeld_,   operation_value TSC_CTMAccessRightRq,   argument      {     cTMPortableIdentity  iPUI ipui_ ,     cTMAuthType          PX_CTM_AuthType,     cTMPortablesCapabilities PX_CTM_PORT_CAP   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_AccessRightRq_RR (invokeld_:INTEGER)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER. Return result component with a cTMSERVICEClass parameter
Constraint Value	
<pre> cTMAccessRightsRequest_Components   cTMAccessRightsRequest_RRCpt {   invokeID      invokeld_,   argument      {     {       operation_value TSC_CTMAccessRightRq,       result          {         {           cTMPortableIdentity  iPUI PX_CTM_IPUI,           cTMFixedIdentity     ?,           cTMSERVICEClass     *         }       }     }   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_Ciphering_Invoke ( ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMCiphering_Components   cTMCiphering_InvokeCpt {   invokeID      ?,   operation_value TSC_CTMCiphering,   argument      {     cTMPortableIdentity  iPUI ipui_ ,     cTMCipherInfo  ?,     cTMCipherKey  ?   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_Ciphering_RR ( invokeId_ : INTEGER )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMCiphering_Components   cTMCiphering_RRCpt {   invokeID      invokeId_ } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_IncomingCallMMInfoRq_Invoke ( ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Constraint Value	
<pre> cTMIncomingCallMMInfo_Components   cTMICallMMInfo_InvokeCpt {   invokeID      ? ,   operation_value TSC_CTMIncomingCallMM ,   argument      {     cTMPortableIdentity iPUI ipui_,     cTMSignal ?   } } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_IncomingCallMMInfo_RE ( invokeId_ : INTEGER ; error_ : Error )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Constraint Value	
<pre> cTMIncomingCallMMInfo_Components   cTMICallMMInfo_RECpt {   invokeID      invokeId_ ,   error         error_ } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_KeyAllocate_Invoke ( ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMKeyAllocate_Components   cTMKeyAllocate_InvokeCpt {   invokeID      ?,   operation_value TSC_CTMKeyAllocate,   argument      {     cTMPortableIdentity iPUI ipui_,     cTMAllocType      ?,     cTMRand            ?,     cTMRs              ?   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_KeyAllocate_RR ( invokeld_ : INTEGER ; res_ : OCTETSTRING)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMKeyAllocate_Components   cTMKeyAllocate_RRcpt {   invokeID      invokeld_ ,   argument      {     operation_value TSC_CTMKeyAllocate,     result          {       cTMRes res_     }   } } </pre>	
<b>Detailed Comments</b> :	



ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_LocationReg_Invoke ( invokeId_ : INTEGER ; ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMLocationRegistration_Components   cTMLocationRegistration_InvokeCpt {   invokeID      invokeId_,   operation_value TSC_CTMLocationReg,   argument      {     cTMPortableIdentity      IPUI ipui_ ,     cTMOldLocationAreaIdentity PX_CTM_NEW_LA,     cTMNewLocationAreaIdentity PX_CTM_NEW_LA,     cTMPortableCapabilities PX_CTM_PORT_CAP   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_LocationReg_RR (invokeid_:INTEGER)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMLocationRegistration_Components   cTMLocationRegistration_RRCpt {   invokeID      invokeid_ } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_NetworkAuth_Invoke ( invokeld_ : INTEGER ; ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
cTMNetworkAuthentication_Components cTMNetworkAuthentication_InvokeCpt { invokeID        invokeld_, operation_value TSC_CTMNetworkAuth, argument       { cTMPortableIdentity iPUI ipui_, cTMAuthType PX_CTM_AuthType, cTMRand       PX_CTM_Rand } } }	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_NetworkAuth_RR ( invokeld_ : INTEGER )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
cTMNetworkAuthentication_Components cTMNetworkAuthentication_RRCpt { invokeID        invokeld_, argument       { { operation_value TSC_CTMNetworkAuth, result        { cTMRes ? } } } } }	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_OutgoingCallMMInfoRq_Invoke ( invokeId_ : INTEGER ; ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMOutgoingCallMMInfo_Components   cTMOCallMMInfo_InvokeCpt {   invokeID      invokeId_,   operation_value TSC_CTMOutgoingCallMM ,   argument      {     cTMPortableIdentity iPUI ipui_,     cTMFixedIdentity PX_CTM_FIXED_IDENTITY,     cTMBasicService PX_CTM_BASIC_SERVICE   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_TerminalAuth_Invoke ( ipui_ : IPUI )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMTerminalAuthentication_Components   cTMTerminalAuthentication_InvokeCpt {   invokeID      ?,   operation_value TSC_CTMTerminalAuth,   argument      {     cTMPortableIdentity iPUI ipui_ ,     cTMAuthType ?,     cTMRand ?,     cTMRs ?   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: CTM_TerminalAuth_RR ( invokeld_ : INTEGER ; res_ : OCTETSTRING)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> cTMTerminalAuthentication_Components   cTMTerminalAuthentication_RRCpt   {     invokeID      invokeld_,     argument     {       operation_value TSC_CTMTerminalAuth,       result {         cTMRes      res_,         cTMSERVICEClass OMIT       }     }   } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: EncapsulatedStimulus_Invoke_R
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN.1 Encoding: BER
Constraint Value	
<pre> encapsulatedStimulus_Components   encapsulatedStimulus_InvokeCpt   {     invokeID      ?,     operation_value TSC_EncapsulatedStimulus,     argument      display ?   } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: EncapsulatedStimulus_Invoke_S ( inv_ : INTEGER; arg_ : KeypadIE )
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN.1 Encoding: BER
Constraint Value	
<pre> encapsulatedStimulus_Components   encapsulatedStimulus_InvokeCpt {   invokeID      inv_,   operation_value TSC_EncapsulatedStimulus,   argument      keypad arg_ } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: FIE_CTM_R01(comp_:Component)
<b>ASN1 Type</b>	: FIE
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Received Facility IE containing a RosePDU service component
Constraint Value	
<pre> {   informationElementId '00011100'B,   length ?,   extBit '1'B,   spareBits '00'B,   protocolProfile '11111'B, -- Networking Extensions value   nfe NFE_CTM_R01,   networkProtocolProfile OMIT,   interpretationAPDU *,   components SUPERSET ({comp_}) } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: FIE_CTM_S01(comp_:Component)
<b>ASN1 Type</b>	: FIE
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Received Facility IE containing a RosePDU service component
Constraint Value	
<pre>{   informationElementId '00011100'B,   length TSO_CALC_FIE_LENGTH(),   extBit '1'B,   spareBits '00'B,   protocolProfile '11111'B, -- Networking Extensions value   nfe NFE_CTM_S01,   networkProtocolProfile OMIT,   interpretationAPDU OMIT,   components {comp_} }</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: FIE_GSM_R01 ( comp_ : Component)
<b>ASN1 Type</b>	: FIE
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Received Facility IE containing a RosePDU service component
Constraint Value	
<pre>{   informationElementId '00011100'B,   length ?,   extBit '1'B,   spareBits '00'B,   protocolProfile '11111'B, -- Networking Extensions value   nfe NFE_GSM_R01,   networkProtocolProfile OMIT,   interpretationAPDU *,   components SUPERSET ({comp_}) }</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: FIE_GSM_S01 ( comp_ : Component)
<b>ASN1 Type</b>	: FIE
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Received Facility IE containing a RosePDU service component
Constraint Value	
<pre>{   informationElementId '00011100'B,   length TSO_CALC_FIE_LENGTH(),   extBit '1'B,   spareBits '00'B,   protocolProfile '11111'B, -- Networking Extensions value   nfe NFE_GSM_S01,   networkProtocolProfile OMIT,   interpretationAPDU OMIT,   components { comp_} }</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: NFE_CTM_R01
<b>ASN1 Type</b>	: NetworkFacilityExtension
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Network facility extension received in a Bearer unrelated PDU with a CTM service function.
Constraint Value	
<pre>{ sourceEntity      endTerminal,   sourceEntityAddress --,   destEntity        anyNode,   destEntityAddress  ?,   serviceFunction    { 0 4 0 1144 3 1} -- CTM }</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: NFE_CTM_S01
<b>ASN1 Type</b>	: NetworkFacilityExtension
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Network facility extension sent in a Bearer unrelated PDU with a CTM service function.
Constraint Value	
<pre>{ sourceEntity      anyNode,   sourceEntityAddress PX_ADDR_SERVING_NODE,   destEntity        endTerminal,   destEntityAddress OMIT,   serviceFunction    { 0 4 0 1144 3 1}  -- CTM }</pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: NFE_GSM_R01
<b>ASN1 Type</b>	: NetworkFacilityExtension
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Network facility extension received in a Bearer unrelated PDU with a DECTAccessToGSM service function.
Constraint Value	
<pre>{ sourceEntity      endTerminal,   sourceEntityAddress -,   destEntity        anyNode,   destEntityAddress ?,   serviceFunction    { 0 4 0 1144 3 2}  -- GSM }</pre>	
<b>Detailed Comments</b>	:



ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: NFE_GSM_S01
<b>ASN1 Type</b>	: NetworkFacilityExtension
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: Network facility extension sent in a Bearer unrelated PDU with a DECTAccessToGSM service function.
Constraint Value	
<pre> { sourceEntity      anyNode,   sourceEntityAddress PX_ADDR_SERVING_NODE,   destEntity        endTerminal,   destEntityAddress OMIT,   serviceFunction   { 0 4 0 1144 3 2}  -- GSM } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_LinkedAssignId_Invoke
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> gSMLinkedAssignIdentity_Components gSMLinkedAssignIdentity_InvokeCpt {   invokeID      ?,   linkedID      ?,   operation_value TSC_GSMLinkedAssignId,   argument      {     gSMNewTMSI      tMSI ?   } } </pre>	
<b>Detailed Comments</b>	: EN 301 144-1 9.3.6.1.2

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_LinkedAssignId_RR ( inv_id_ : INTEGER)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMLinkedAssignIdentity_Components gSMLinkedAssignIdentity_RRCpt { invokeID            inv_id_ } 	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_AssignIdentity_Invoke ( port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMAssignIdentity_Components gSMAssignIdentity_InvokeCpt { invokeID            ?, operation_value    TSC_GSMAssignIdentity, argument            { gSMPortableIdentity    port_id_ , gSMLocationArealIdentity    PX_GSM_LA_ID, gSMNewTMSI            tMSI ? } } 	
<b>Detailed Comments</b> : EN 301 144–1 9.3.6.1.1	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_AssignIdentity_RR ( inv_id_ : INTEGER)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMAssignIdentity_Components gSMAssignIdentity_RRCpt { invokeID          inv_id_ } 	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_Ciphering_Invoke ( port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMCiphering_Components gSMCiphering_InvokeCpt { invokeID          ?, operation_value   TSC_GSMCiphering, argument          { gSMPortableIdentity   port_id_ , gSMCipherKey          ? } } 	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_Ciphering_RR ( inv_id_ : INTEGER)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMCiphering_Components gSMCiphering_RRCpt { invokeID            inv_id_ } 	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_IdentityReq_Invoke (port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMIdentityRequest_Components gSMIdentityRequest_InvokeCpt { invokeID            ?, operation_value    TSC_GSMIdentityRequest, argument           { gSMPortableIdentity    port_id_, gSMIdentityType        ? } } 	
<b>Detailed Comments</b> : in the normal case the IMSI is used as PortableIdentity (EN 301 144–1 9.3.8.1)	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_IdentityReq_RR ( inv_id_ : INTEGER; port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> gSMIdentityRequest_Components gSMIdentityRequest_RRCpt {   invokeID      inv_id_,   argument {     operation_value TSC_GSMIdentityRequest,     result {       gSMPortableIdentity      port_id_     }   } } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_IncomingCallMMInfo_Invoke (port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Constraint Value	
<pre> gSMIncomingCallMMInfo_Components gSMICallMMInfo_InvokeCpt {   invokeID      ?,   operation_value TSC_GSMIncomingCallMM,   argument {     gSMPortableIdentity      port_id_ ,     gSMSignal                ?   } } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_IncomingCallMMInfo_RE ( inv_id_ : INTEGER ; error_ : Error)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	:
Constraint Value	
gSMIncomingCallMMInfo_Components gSMICallMMInfo_RECpt { invokeID            inv_id_, error               error_ } 	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_LocationRegistration_Invoke( inv_id_ : INTEGER; port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
gSMLocationRegistration_Components gSMLocationRegistration_InvokeCpt { invokeID            inv_id_, operation_value    TSC_GSMLocationReg, argument           { gSMPortableIdentity               port_id_, gSMLocationRegistrationType       PX_GSM_LR_TYPE, gSMLocationArealIdentity          PX_GSM_LA_ID, gSMCipherInfo                     PX_GSM_CIPHER_INFO, gSMPortableCapabilities           PX_GSM_PORT_CAP } } 	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_LocationRegistration_RR ( inv_id_ : INTEGER)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> gSMLocationRegistration_Components   gSMLocationRegistration_RRCpt   {     invokeID      inv_id_,     argument      {       operation_value TSC_GSMLocationReg,       result {         gSMLocationAreaidentity PX_GSM_LA_ID       }     }   } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_OutgoingCallIMMInfo_Invoke ( inv_id_ : INTEGER; port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> gSMOutgoingCallIMMInfo_Components   gSMOCallIMMInfo_InvokeCpt   {     invokeID      inv_id_,     operation_value TSC_GSMOutgoingCallIMM,     argument      {       gSMPortableIdentity      port_id_,       gSMBasicService          PX_GSM_BASIC_SERVICE     }   } </pre>	
<b>Detailed Comments</b>	:

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_TerminalAuth_Invoke ( port_id_ : PortableIdentity)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> gSMTerminalAuthentication_Components gSMTerminalAuthentication_InvokeCpt {   invokeID      ?,   operation_value TSC_GSMTerminalAuth,   argument      {     gSMPortableIdentity port_id_ ,     gSMRand            ?,     gSMCipherInfo      ?   } } </pre>	
<b>Detailed Comments</b> :	

ASN.1 Type Constraint Declaration	
<b>Constraint Name</b>	: GSM_TerminalAuth_RR ( inv_id_ : INTEGER; res_:Res)
<b>ASN1 Type</b>	: Component
<b>Derivation Path</b>	:
<b>Encoding Variation</b>	:
<b>Comments</b>	: ASN1_Encoding: BER
Constraint Value	
<pre> gSMTerminalAuthentication_Components gSMTerminalAuthentication_RRCpt {   invokeID      inv_id_,   argument      {     operation_value TSC_GSMTerminalAuth,     result         {       gSMRes       res_     }   } } </pre>	
<b>Detailed Comments</b> :	



ASP Constraint Declaration		
<b>Constraint Name</b> : Ds ( param : IA5String )		
<b>ASP Type</b> : DISPLAY		
<b>Derivation Path</b> :		
<b>Comments</b> : ASP to send information to the operator.		
Parameter Name	Parameter Value	Comments
msg	param	Information to be sent to the operator
<b>Detailed Comments</b> :		

ASP Constraint Declaration		
<b>Constraint Name</b> : Ms(PARAM: PDU)		
<b>ASP Type</b> : DL_DAT_RQ		
<b>Derivation Path</b> :		
<b>Comments</b> : ASP to request the sending of layer 3 messages.		
Parameter Name	Parameter Value	Comments
mun	PARAM	PDU to be sent
<b>Detailed Comments</b> :		

ASP Constraint Declaration		
<b>Constraint Name</b> : Mr(PARAM: PDU)		
<b>ASP Type</b> : DL_DAT_IN		
<b>Derivation Path</b> :		
<b>Comments</b> : ASP to indicate the receipt of layer 3 messages.		
Parameter Name	Parameter Value	Comments
mun	PARAM	PDU to be received
<b>Detailed Comments</b> :		

ASP Constraint Declaration		
<b>Constraint Name</b> : RSr(PARAM: RESTART_PDU)		
<b>ASP Type</b> : DL_DAT_IN_RESTART		
<b>Derivation Path</b> :		
<b>Comments</b> : ASP to indicate the receipt of RESTART messages.		
Parameter Name	Parameter Value	Comments
mun	PARAM	RESTART to be received
<b>Detailed Comments</b> :		

ASP Constraint Declaration		
<b>Constraint Name</b> : Sr(PARAM: SETUP_PDU)		
<b>ASP Type</b> : DL_DAT_IN_SETUP		
<b>Derivation Path</b> :		
<b>Comments</b> : ASP to indicate the receipt of SETUP messages.		
Parameter Name	Parameter Value	Comments
mun	PARAM	SETUP to be received
<b>Detailed Comments</b> :		

ASP Constraint Declaration		
<b>Constraint Name</b> : UMr(PARAM: PDU)		
<b>ASP Type</b> : DL_UDAT_IN		
<b>Derivation Path</b> :		
<b>Comments</b> : ASP to indicate the receipt of layer 3 messages.		
Parameter Name	Parameter Value	Comments
mun	PARAM	PDU to be received
<b>Detailed Comments</b> :		

PDU Constraint Declaration			
<b>Constraint Name</b> : AL_R1 ( flag_ : INTEGER; cref_ : BIT7OR15) <b>PDU Type</b> : ALERTING_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_ALERTING		
bcap	*		
efac	*		
chi	*		
fac	—		
fac2	*		
pi	*		
noid	*		
dsp	*		
ronn	*		
hlc	*		
uui	*		
<b>Detailed Comments</b> : PDU with "don't care" values;			

PDU Constraint Declaration			
<b>Constraint Name</b> : AL_S1 ( flag_ : INTEGER; cref_ : BIT7OR15) <b>PDU Type</b> : ALERTING_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_ALERTING		
bcap	—		
efac	—		
chi	—		
fac	—		
fac2	—		
pi	—		
noid	—		
dsp	—		
ronn	—		
hlc	—		
uui	—		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : ANY_MSG <b>PDU Type</b> : ANY_MSG_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	'00001000'B		
cr	CallRef_R1		
mt	?		
ie_list	*		
<b>Detailed Comments</b> : &COMMON_N12 PDU with a valid CREF. Used for test cases where PDUs must be absorbed by the tester;			

PDU Constraint Declaration			
<b>Constraint Name</b> : CA_S1(flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CONNECT_ACK_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CONNECT_ACK		
efac	—		
fac	—		
fac2	—		
noid	—		
dsp	—		
<b>Detailed Comments</b> : PDU without optional information elements.			

PDU Constraint Declaration			
<b>Constraint Name</b> : CN_R1(flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CONNECT_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received CONNECT message for a NCICS connection			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CONNECT		
bcap	—		
efac	—		
chi	—		
fac	—		
fac2	*		
pi	—		
noid	*		
dsp	—		
dati	—		
codn	—		
cods	—		
ronn	—		
llc	—		
hlc	—		
uui	—		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : CN_R2 (flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CONNECT_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received CONNECT message			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CONNECT		
bcap	*		
efac	—		
chi	*		
fac	*		
fac2	*		
pi	*		
noid	*		
dsp	*		
dati	*		
codn	*		
cods	*		
ronn	*		
llc	*		
hlc	*		
uui	*		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : CN_R3 (flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CONNECT_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received CONNECT message without redirection number information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CONNECT		
bcap	*		
efac	—		
chi	*		
fac	*		
fac2	*		
pi	*		
noid	*		
dsp	*		
dati	*		
codn	*		
cods	*		
ronn	—		
llc	*		
hlc	*		
uui	*		
<b>Detailed Comments</b> :			



PDU Constraint Declaration			
<b>Constraint Name</b> : CN_S1(flag_: INTEGER; cref_: BIT7OR15)			
<b>PDU Type</b> : CONNECT_PDU			
<b>Derivation Path</b> :			
<b>Encoding Rule Name</b> :			
<b>Encoding Variation</b> :			
<b>Comments</b> : Sent CONNECT message			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		(1)
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CONNECT		
bcap	–		
efac	–		
chi	–		
fac	–		
fac2	–		
pi	–		
noid	–		
dsp	–		
dati	–		
codn	–		
cods	–		
ronn	–		
llc	–		
hlc	–		
uui	–		
<b>Detailed Comments</b> : (1) The Channel identification information element is mandatory in messages sent as a first response to a SETUP message unless the user accepts the B–channel indicated in the SETUP message.			

PDU Constraint Declaration			
<b>Constraint Name</b> : CP_R1(flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CALL_PROC_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received Call proceeding message for a NCICS call (no information elements are included).			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CALL_PROC		
bcap	—		
efac	—		
chi	—		
fac	—		
pi	—		
noid	*		
dsp	—		
hlc	—		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : CP_R2(flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CALL_PROC_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received Call proceeding message without notification information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CALL_PROC		
bcap	–		
efac	–		
chi	TSO_ASSIGN_CHI (CHI_basic_R1,CHI_primary _R1, PX_BASIC) IF_PRESENT		
fac	–		
pi	–		
noid	–		
dsp	–		
hlc	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : CP_R3 (flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : CALL_PROC_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received Call proceeding message with a notification information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_CALL_PROC		
bcap	–		
efac	–		
chi	TSO_ASSIGN_CHI (CHI_basic_R1,CHI_primary _R1, PX_BASIC) IF_PRESENT		
fac	–		
pi	–		
noid	NOID_2('11111011'B )		CALL IS DIVERTING
dsp	–		
hlc	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : CP_S1 ( flag : INTEGER; cref : BIT7OR15; bch_ : BITSTRING) <b>PDU Type</b> : CALL_PROC_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Sent CALL PROCEEDING message (ChannelId IE present).			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag, cref)		
mt	TSC_MT_CALL_PROC		
bcap	–		
efac	–		
chi	TSO_ASSIGN_CHI ( CHI_basic_S2 ( bch_), CHI_primary_S2 ( bch_), PX_BASIC )		
fac	–		
pi	–		
noid	–		
dsp	–		
hlc	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : DI_S1 ( flag_ : INTEGER; cref_ : BIT7OR15; cval_ : INTEGER) <b>PDU Type</b> : DISCONNECT_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_DISCONNECT		
cau	Cause_S1(cval_)		
efac	–		
fac	–		
fac2	–		
pi	–		
noid	–		
dsp	–		
uui	–		
<b>Detailed Comments</b> : PDU with Dect access to GSM facility IE.			

PDU Constraint Declaration			
<b>Constraint Name</b> : DI_S2 ( flag_ : INTEGER; cref_ : BIT7OR15; cval_ : INTEGER; fie_ : FIE) <b>PDU Type</b> : DISCONNECT_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_DISCONNECT		
cau	Cause_S1(cval_)		
efac	—		
fac	{ fie_ }		
fac2	—		
pi	—		
noid	—		
dsp	—		
uui	—		
<b>Detailed Comments</b> : PDU with Dect access to GSM facility IE.			

PDU Constraint Declaration			
<b>Constraint Name</b> : FC_R1(flag_ : INTEGER ; cref_ : BIT7OR15 ; fie_ : FIE ) <b>PDU Type</b> : FAC_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Facility message to be received. The Facility IE to be received is passed within a formal parameter.			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_FAC		
fac	SUPERSET({fie_})		
fac2	—		
noid	*		
<b>Detailed Comments</b> : FAC_IE shall be included in the SET OF facility IE received.			

PDU Constraint Declaration			
<b>Constraint Name</b> : FC_S1(flag_ : INTEGER ; cref_ : BIT7OR15 ; fie_ : FIE ) <b>PDU Type</b> : FAC_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Facility message to be sent. The Facility IE to be sent is passed within a formal parameter.			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_FAC		
fac	{fie_}		
fac2	–		
noid	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : GFP_R1 ( flag_ : INTEGER ; cref_ : BIT7OR15) <b>PDU Type</b> : GFP_MSG_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	?		
ie_list	*		
<b>Detailed Comments</b> : PDU with a valid CREF. Used for test cases where PDUs must be absorbed by the tester.			



PDU Constraint Declaration			
<b>Constraint Name</b> : IN_R1 ( flag_ : INTEGER; cref_ : BIT7OR15) <b>PDU Type</b> : INFORMATION_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_INFORMATION		
sci	*		
cau	*		
efac	*		
fac	—		
fac2	*		
noid	*		
dsp	*		
kpf	*		
cdpn	*		
ronn	*		
<b>Detailed Comments</b> : PDU with "don't care" values;			

PDU Constraint Declaration			
<b>Constraint Name</b> : NO_R1 ( flag_ : INTEGER ; cref_ : BIT7OR15) <b>PDU Type</b> : NOTIFY_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_NOTIFY		
noid	NOID_R1		
dsp	*		
ronn	*		
<b>Detailed Comments</b> : PDU with "don't care" values in noid.			

PDU Constraint Declaration			
<b>Constraint Name</b> : NO_R2 ( flag_ : INTEGER ; cref_ : BIT7OR15)			
<b>PDU Type</b> : NOTIFY_PDU			
<b>Derivation Path</b> :			
<b>Encoding Rule Name</b> :			
<b>Encoding Variation</b> :			
<b>Comments</b> : Receive PDU with a notification information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		CALL IS DIVERTING
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_NOTIFY		
noid	NOID_2('11111011'B )		
dsp	*		
ronn	*		
<b>Detailed Comments</b> : PDU with "don't care" values in noid.			

PDU Constraint Declaration			
<b>Constraint Name</b> : NO_R3 ( flag_ : INTEGER ; cref_ : BIT7OR15)			
<b>PDU Type</b> : NOTIFY_PDU			
<b>Derivation Path</b> :			
<b>Encoding Rule Name</b> :			
<b>Encoding Variation</b> :			
<b>Comments</b> : Receive PDU with a notification information element without the redirection number information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_NOTIFY		
noid	NOID_R1		
dsp	*		
ronn	—		
<b>Detailed Comments</b> : PDU with "don't care" values in noid.			

PDU Constraint Declaration			
<b>Constraint Name</b> : PR_R1 ( flag_ : INTEGER ; cref_ : BIT7OR15) <b>PDU Type</b> : PROGRESS_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_PROGRESS		
bcap	*		
cau	*		
pi	?		
dsp	*		
ronn	*		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
hlc	*		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : PR_R2 ( flag_ : INTEGER ; cref_ : BIT7OR15) <b>PDU Type</b> : PROGRESS_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU without redirection number information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_PROGRESS		
bcap	*		
cau	*		
pi	?		
dsp	*		
ronn	—		Redirection number Direction: n>u , type: O, length: 2 – 24 octets
hlc	*		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : RC_R1(flag_: INTEGER; call_ref_: BIT7OR15) <b>PDU Type</b> : RELEASE_COM_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, call_ref_)		
mt	TSC_MT_RELEASE_COM		
cau	*		
efac	*		
fac	—		
fac2	*		
noid	*		
dsp	*		
uui	*		
<b>Detailed Comments</b> : PDU with "don't care" values.			

PDU Constraint Declaration			
<b>Constraint Name</b> : RC_S1(flag_: INTEGER; cref_: BIT7OR15) <b>PDU Type</b> : RELEASE_COM_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_RELEASE_COM		
cau	—		
efac	—		
fac	—		
fac2	—		
noid	—		
dsp	—		
uui	—		
<b>Detailed Comments</b> : PDU without optional information elements.			

PDU Constraint Declaration			
<b>Constraint Name</b> : RL_R1(flag_: INTEGER; cref_: BIT7OR15)			
<b>PDU Type</b> : RELEASE_PDU			
<b>Derivation Path</b> :			
<b>Encoding Rule Name</b> :			
<b>Encoding Variation</b> :			
<b>Comments</b> : Received RELEASE message . (No facility information element is allowed)			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		mandatory for NCICS
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_RELEASE		
cau	?		
efac	—		
fac	—		
fac2	*		
noid	—		
dsp	—		
uui	—		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : RL_R2 (flag_: INTEGER; cref_: BIT7OR15; cval_: INTEGER)			
<b>PDU Type</b> : RELEASE_PDU			
<b>Derivation Path</b> :			
<b>Encoding Rule Name</b> :			
<b>Encoding Variation</b> :			
<b>Comments</b> : Received RELEASE message . (No facility information element is allowed)			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		mandatory for NCICS
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_RELEASE		
cau	Cause_R2 (cval_)		
efac	–		
fac	–		
fac2	–		
noid	–		
dsp	–		
uui	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : RL_S1(flag:INTEGER; call_ref: BIT7OR15; cval: INTEGER) <b>PDU Type</b> : RELEASE_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag, call_ref)		
mt	TSC_MT_RELEASE		
cau	Cause_S1(cval)		
efac	–		
fac	–		
noid	–		
dsp	–		
uui	–		
<b>Detailed Comments</b> : PDU with optional information element cau.			

PDU Constraint Declaration			
<b>Constraint Name</b> : RSA_S1(flag_: INTEGER; call_ref_: BIT7OR15; class_val_: INTEGER) <b>PDU Type</b> : RESTART_ACK_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Send PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, call_ref_)		
mt	TSC_MT_RESTART_ACK		
chi	–		
chi_rs	–		
dsp	–		
ri	RestartInd1(class_val_)		
<b>Detailed Comments</b> : PDU without optional parameters; PDU that indicates "All interfaces" or "Single interface".			

PDU Constraint Declaration			
<b>Constraint Name</b> : RST_R1(flag_: INTEGER; call_ref_: BIT7OR15; class_val_: INTEGER) <b>PDU Type</b> : RESTART_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received RESTART message for NCICS (no Channel Id IE is included, because only the codepoint "single" and "all" interfaces are supported).			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, call_ref_)		
mt	TSC_MT_RESTART		
chi	—		
chi_rs	—		
dsp	*		
ri	RestartInd1(class_val_)		
<b>Detailed Comments</b> : PDU with "don't care" values; PDU that indicates "All interfaces" or "Single interface".			

PDU Constraint Declaration			
<b>Constraint Name</b> : SQ_R1( flag_ : INTEGER ; cref_ : BIT7OR15) <b>PDU Type</b> : STATUS_ENQ_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_STATUS_ENQ		
dsp	*		
<b>Detailed Comments</b> : PDU with "don't care" values.			

PDU Constraint Declaration			
<b>Constraint Name</b> : SU_R1 (fie_ : FIE) <b>PDU Type</b> : SETUP_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received SETUP message for a call.			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_R1		
mt	TSC_MT_SETUP		
sci	'10100001'B IF_PRESENT		
bcap	BCAP_S1		
efac	—		
chi	TSO_ASSIGN_CHI ( CHI_basic_S1, CHI_primary_S1, PX_BASIC)		
fac	SUPERSET({fie_})		
fac2	—		
pi	—		
nsf	—		
noid	—		
dsp	—		
kpf	—		
cgpn	*		
cgps	—		
cdpn	*		
cdps	—		
rngn	—		
tns	—		
llc	—		
hlc	—		
uui	—		
lsh4	'94'O IF_PRESENT		
<b>Detailed Comments</b> :			



PDU Constraint Declaration			
<b>Constraint Name</b> : SU_R3 (fie_ : FIE) <b>PDU Type</b> : SETUP_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received SETUP message for a call.			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_R1		
mt	TSC_MT_SETUP		
sci	'10100001'B IF_PRESENT		
bcap	BCAP_S1		
efac	—		
chi	TSO_ASSIGN_CHI ( CHI_basic_S1, CHI_primary_S1, PX_BASIC)		
fac	SUPERSET ({fie_})		
fac2	—		
pi	—		
nsf	—		
noid	—		
dsp	—		
kpf	—		
cgpn	*		
cgps	—		
cdpn	*		
cdps	—		
rngn	RNGN1		
tns	—		
llc	—		
hlc	—		
uui	—		
lsh4	'94'O IF_PRESENT		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : SU_R4 (fie_ : FIE) <b>PDU Type</b> : SETUP_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Received SETUP message for a call.			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_R1		
mt	TSC_MT_SETUP		
sci	'10100001'B IF_PRESENT		
bcap	BCAP_S1		
efac	—		
chi	TSO_ASSIGN_CHI ( CHI_basic_S1, CHI_primary_S1, PX_BASIC)		
fac	SUPERSET ({{fie_}})		
fac2	—		
pi	—		
nsf	—		
noid	—		
dsp	—		
kpf	—		
cgpn	*		
cgps	—		
cdpn	*		
cdps	—		
rngn	RNGN2		
tns	—		
llc	—		
hlc	—		
uui	—		
lsh4	'94'O IF_PRESENT		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : SU_S1 ( flag_ : INTEGER; cref_ : BIT7OR15; fie_ : FIE ) <b>PDU Type</b> : SETUP_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Sent SETUP message for a NCICS connection with a facility information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_, cref_)		
mt	TSC_MT_SETUP		
sci	TSC_ID_SCI		
bcap	BCAP_S1		
efac	–		
chi	TSO_ASSIGN_CHI ( CHI_basic_S1, CHI_primary_S1, PX_BASIC)		
fac	{fie_}		
fac2	–		
pi	–		
nsf	–		
noid	–		
dsp	–		
kpf	–		
cgpn	–		
cgps	–		
cdpn	CDPN_S1		Address related to an End configuration.
cdps	–		
rngn	–		
tns	–		
llc	–		
hlc	–		
uui	–		
lsh4	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : SU_S2(flag_: INTEGER; cref_: BIT7OR15; fie_ : FIE) <b>PDU Type</b> : SETUP_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Sent SETUP message for a outgoing call without CDPN IE with a facility information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1 ( flag_ , cref_ )		
mt	TSC_MT_SETUP		
sci	TSC_ID_SCI		
bcap	BCAP_S2		
efac	–		
chi	TSO_ASSIGN_CHI (		
	CHI_basic_S2 (		
	tcv_bch_num ),		
	CHI_primary_S2 (		
	tcv_bch_num ), PX_BASIC )		
fac	{ fie_ }		
fac2	–		
pi	–		
nsf	–		
noid	–		
dsp	–		
kpf	–		
cgpn	–		
cgps	–		
cdpn	–		
cdps	–		
rngn	–		
tns	–		
llc	–		
hlc	–		
uui	–		
lsh4	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : SU_S3(flag_: INTEGER; cref_: BIT7OR15; fie_:FIE; bch_ : BITSTRING)			
<b>PDU Type</b> : SETUP_PDU			
<b>Derivation Path</b> :			
<b>Encoding Rule Name</b> :			
<b>Encoding Variation</b> :			
<b>Comments</b> : Sent SETUP message for a outgoing call with CDPN IE within a facility information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		Address related to an End configuration.
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_SETUP		
sci	TSC_ID_SCI		
bcap	BCAP_S1		
efac	–		
chi	TSO_ASSIGN_CHI ( CHI_basic_S2(bch_), CHI_primary_S2(bch_), PX_BASIC)		
fac	{fie_}		
fac2	–		
pi	–		
nsf	–		
noid	–		
dsp	–		
kpf	–		
cgpn	–		
cgps	–		
cdpn	CDPN_S1		
cdps	–		
rngn	–		
tns	–		
llc	–		
hlc	–		
uui	–		
lsh4	–		
<b>Detailed Comments</b> :			

PDU Constraint Declaration			
<b>Constraint Name</b> : SUA_R1 ( flag_ : INTEGER; cref_ : BIT7OR15) <b>PDU Type</b> : SETUP_ACK_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_SETUP_ACK		
efac	*		
chi	*		
fac	—		
fac2	*		
pi	*		
noid	*		
dsp	*		
<b>Detailed Comments</b> : PDU without optional information elements			

PDU Constraint Declaration			
<b>Constraint Name</b> : SUA_R2 ( flag_ : INTEGER; cref_ : BIT7OR15) <b>PDU Type</b> : SETUP_ACK_PDU <b>Derivation Path</b> : <b>Encoding Rule Name</b> : <b>Encoding Variation</b> : <b>Comments</b> : Receive PDU, without notification information element			
Field Name	Field Value	Field Encoding	Comments
pd	TSC_Q931_PD		
cr	CallRef_1(flag_, cref_)		
mt	TSC_MT_SETUP_ACK		
efac	*		
chi	*		
fac	—		
fac2	*		
pi	*		
noid	—		
dsp	*		
<b>Detailed Comments</b> : PDU without optional information elements			

CM Constraint Declaration		
<b>Constraint Name</b> : N00_Ready <b>CM Type</b> : CP_M <b>Derivation Path</b> : <b>Comments</b> : to indicate that the preamble to N00 is finished at the PTC		
Parameter Name	Parameter Value	Comments
CM_content	"N00_Ready"	
<b>Detailed Comments</b> : &COMMON_N09		

CM Constraint Declaration		
<b>Constraint Name</b> : STOP_PTC <b>CM Type</b> : CP_M <b>Derivation Path</b> : <b>Comments</b> : to stop the PTC test step		
Parameter Name	Parameter Value	Comments
CM_content	"STOP_PTC"	
<b>Detailed Comments</b> : &COMMON_N12		

CM Constraint Declaration		
<b>Constraint Name</b> : R_SU <b>CM Type</b> : CP_M <b>Derivation Path</b> : <b>Comments</b> : to trigger off the sending of a SETUP message		
Parameter Name	Parameter Value	Comments
CM_content	"RECEIVED_SETUP"	
<b>Detailed Comments</b> : &COMMON_N12		

CM Constraint Declaration		
<b>Constraint Name</b> : R_SU_Rgnb <b>CM Type</b> : CP_M <b>Derivation Path</b> : <b>Comments</b> : to trigger off the receipt of a SETUP message with a redirecting number information element		
Parameter Name	Parameter Value	Comments
CM_content	"RECEIVE_SETUP_RGNB"	
<b>Detailed Comments</b> : &COMMON_N12		

CM Constraint Declaration		
<b>Constraint Name</b> : R_SU_RgnbR <b>CM Type</b> : CP_M <b>Derivation Path</b> : <b>Comments</b> : to trigger off the receipt of a SETUP message with a redirecting number information element indicating presentation restricted		
Parameter Name	Parameter Value	Comments
CM_content	"RECEIVE_SETUP_RGNBR"	
<b>Detailed Comments</b> : &COMMON_N12		

CM Constraint Declaration		
<b>Constraint Name</b> : S_SU <b>CM Type</b> : CP_M <b>Derivation Path</b> : <b>Comments</b> : to trigger off the sending of a SETUP message		
Parameter Name	Parameter Value	Comments
CM_content	"SEND_SETUP"	
<b>Detailed Comments</b> : &COMMON_N12		



# **IV**

## **Dynamic Part**

Test Case Dynamic Behaviour								
<b>Test Case Name</b> : CFNRc_N01_001								
<b>Group</b> : ServedUser/Registration/								
<b>Purpose</b> : Ensure that IUT in call state N2 for a CTM outgoing call, on receipt of a FACILITY message including an EncapsulatedStimulus invoke component containing encapsulated a keypad information requesting registration, Sends the FACILITY message, including an EncapsulatedStimulus invoke component with the Display information								
<b>Configuration</b> : CONFIG0								
<b>Default</b> : DEF_CALL								
<b>Comments</b> : EN 302 094–1, subclause 9.1.1								
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments			
1	L1	+ PR_N00_Init	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))	(F)				
2		+ ST_N02_CTMOutgoing						
3		+ ltree_send_facility						
4		START T_AC						
5		L ? PDUr CANCEL T_AC						
6		GOTO L1						
7		L? PDUr CANCEL T_AC						
8		L!PDUr						
9		GOTO L1						
10		+ ltree_receive_fac						
11		+ ltree_release						
12		? TIMEOUT T_AC						
13		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )						
		ltree_receive_fac						
14		[ PC_CFNRc_GSM ]					(1)	
15		L?PDUr CANCEL T_AC				Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invoke_R ) ) )	(P)	(3)
16		[ PC_CFNRc_CTM ]					(2)	
17		L?PDUr CANCEL T_AC				Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invoke_R ) ) )	(P)	(3)
		ltree_send_facility						
18	( tcv_invokeld := TSO_RANDOM_INVOKE_ID ( ) )							
19	[ PC_CFNRc_CTM ]			(1)				

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
20		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_CTM_KEYPAD_REGISTRATION ) ) ) )		(4)
21		[ PC_CFNRC_GSM ]			(2)
22		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_GSM_KEYPAD_REGISTRATION ) ) ) )		(4)
23		ltree_release + PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
24		+ ST_Erase			(4)
<b>Detailed Comments :</b> 1 IUT supports DectToGsm mode 2 IUT supports CTM mode 3 Receipt of a FACILITY message including an Encapsulatedstimulus invoke component including a display information 4 Send a FACILITY message containing an EncapsulatedStimulus invoke component with a keypad information					

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N02_001					
<b>Group</b> : ServedUser/Erasure/					
<b>Purpose</b> : Ensure that IUT in call state N2 for a CTM outgoing call, on receipt of a FACILITY message including an EncapsulatedStimulus invoke component containing encapsulated a keypad information requesting erasure, Sends the FACILITY message, including an EncapsulatedStimulus invoke component with the Display information					
<b>Configuration</b> : CONFIG0					
<b>Default</b> : DEF_CALL					
<b>Comments</b> : EN 302 094-1, subclause 9.1.1					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+ PR_N00_Init	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))	(F)	
2		+ ST_Register			
3		+ST_N02_CTMOutgoing			
4		+ ltree_send_facility			
5		START T_AC			
6		L ? PDUr CANCEL T_AC			
7		GOTO L1			
8		L? PDUr CANCEL T_AC	Mr (CN_R2 ( tcv_flag_r, tcv_cref ))		
9		L!PDUs	Ms ( CA_S1 ( tcv_flag_s, tcv_cref ) )		
10		GOTO L1			
11		+ ltree_receive_fac			
12		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
13		? TIMEOUT T_AC			
14		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
15		+ ST_Erase			
16		ltree_receive_fac			(1)
17		[ PC_CFNRc_GSM ]			(3)
		L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	
18		[ PC_CFNRc_CTM ]			(2)
19		L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)
		ltree_send_facility			
20		( tcv_invokeId := TSO_RANDOM_INVOKE_ID ())			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		[ PC_CFNRc_CTM ]			(1)
22		L!PDU <sub>s</sub> START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_CTM_KEYPAD_ERASU RE ) ) ) )		(4)
23		[ PC_CFNRc_GSM ]			(2)
24		L!PDU <sub>s</sub> START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_GSM_KEYPAD_ERASU RE ) ) ) )		(4)
<b>Detailed Comments :</b> 1 IUT supports DectToGsm mode 2 IUT supports CTM mode 3 Receipt of a FACILITY message including an Encapsulatedstimulus invoke component including a display information 4 Send a FACILITY message containing an EncapsulatedStimulus invoke component with a keypad information					

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N03_001					
<b>Group</b> : ServedUser/Activation/					
<b>Purpose</b> : Ensure that IUT in call state N2 for a CTM outgoing call, on receipt of a FACILITY message including an EncapsulatedStimulus invoke component containing encapsulated a keypad information requesting activation, Sends the FACILITY message, including an EncapsulatedStimulus invoke component with the Display information					
<b>Configuration</b> : CONFIG0					
<b>Default</b> : DEF_CALL					
<b>Comments</b> : EN 302 094-1, subclause 9.1.1					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+ PR_N00_Init	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))	(F)	
2		+ ST_Register			
3		+ST_N02_CTMOutgoing			
4		+ ltree_send_facility			
5		START T_AC			
6		L ? PDUr CANCEL T_AC			
7		GOTO L1			
8		L? PDUr CANCEL T_AC	Mr (CN_R2 ( tcv_flag_r, tcv_cref ))		
9		L!PDUs	Ms ( CA_S1 ( tcv_flag_s, tcv_cref ) )		
10		GOTO L1			
11		+ ltree_receive_fac			
12		+ ltree_release			
13		? TIMEOUT T_AC			
14		+ ltree_release			
15		ltree_receive_fac	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(1)
16	[ PC_CFNRc_GSM ]	L?PDUr CANCEL T_AC			(3)
17	[ PC_CFNRc_CTM ]				(2)
18	L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )			(3)
19		ltree_send_facility			
20		( tcv_invokeId := TSO_RANDOM_INVOKE_ID ( ) )			
	[ PC_CFNRc_CTM ]				(1)

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_CTM_KEYPAD_ACTIVATION ) ) ) )		(4)
22		[ PC_CFNRC_GSM ]			(2)
23		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_GSM_KEYPAD_ACTIVATION ) ) ) )		(4)
24		ltree_release			
25		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
		+ ST_Erase			
<b>Detailed Comments :</b> 1 IUT supports DectToGsm mode 2 IUT supports CTM mode 3 Receipt of a FACILITY message including an Encapsulatedstimulus invoke component including a display information 4 Send a FACILITY message containing an EncapsulatedStimulus invoke component with a keypad information					

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N04_001					
<b>Group</b> : ServedUser/Deactivation/					
<b>Purpose</b> : Ensure that IUT in call state N2 for a CTM outgoing call, on receipt of a FACILITY message including an EncapsulatedStimulus invoke component containing encapsulated a keypad information requesting deactivation, Sends the FACILITY message, including an EncapsulatedStimulus invoke component with the Display information					
<b>Configuration</b> : CONFIG0					
<b>Default</b> : DEF_CALL					
<b>Comments</b> : EN 302 094-1, subclause 9.1.1					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+ PR_N00_Init	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))	(F)	
2		+ ST_Register			
3		+ST_N02_CTMOutgoing			
4		+ ltree_send_facility			
5		START T_AC			
6		L ? PDUr CANCEL T_AC			
7		GOTO L1			
8		L? PDUr CANCEL T_AC	Mr (CN_R2 ( tcv_flag_r, tcv_cref ))		
9		L!PDUs	Ms ( CA_S1 ( tcv_flag_s, tcv_cref ) )		
10		GOTO L1			
11		+ ltree_receive_fac			
12		+ ltree_release			
13		? TIMEOUT T_AC			
14		+ ltree_release			
15		ltree_receive_fac			
15	[ PC_CFNRc_GSM ]				(1)
16	L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)	
17	[ PC_CFNRc_CTM ]				(2)
18	L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)	
19		ltree_send_facility			
19	( tcv_invokeId := TSO_RANDOM_INVOKE_ID ( ) )				
20	[ PC_CFNRc_CTM ]				(1)

Continued on next page



Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_CTM_KEYPAD_DEACTIVATION ) ) ) )		(4)
22		[ PC_CFNRC_GSM ]			(2)
23		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_GSM_KEYPAD_DEACTIVATION ) ) ) )		(4)
24		ltree_release + PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
25		+ ST_Erase			
<b>Detailed Comments :</b> 1 IUT supports DectToGsm mode 2 IUT supports CTM mode 3 Receipt of a FACILITY message including an Encapsulatedstimulus invoke component including a display information 4 Send a FACILITY message containing an EncapsulatedStimulus invoke component with a keypad information					

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N05_001					
<b>Group</b> : ServedUser/Interrogation/					
<b>Purpose</b> : Ensure that IUT in call state N2 for a CTM outgoing call, on receipt of a FACILITY message including an EncapsulatedStimulus invoke component containing encapsulated a keypad information requesting interrogation, Sends the FACILITY message, including an EncapsulatedStimulus invoke component with the Display information					
<b>Configuration</b> : CONFIG0					
<b>Default</b> : DEF_CALL					
<b>Comments</b> : EN 302 094-1, subclause 9.1.1					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+ PR_N00_Init	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))	(F)	
2		+ ST_Register			
3		+ST_N02_CTMOutgoing			
4		+ ltree_send_facility			
5		START T_AC			
6		L ? PDUr CANCEL T_AC			
7		GOTO L1			
8		L? PDUr CANCEL T_AC	Mr (CN_R2 ( tcv_flag_r, tcv_cref ))		
9		L!PDUs	Ms ( CA_S1 ( tcv_flag_s, tcv_cref ) )		
10		GOTO L1			
11		+ ltree_receive_fac			
12		+ ltree_release			
13		? TIMEOUT T_AC			
14		+ ltree_release			
15		ltree_receive_fac			
15	[ PC_CFNRc_GSM ]				(1)
16	L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)	
17	[ PC_CFNRc_CTM ]				(2)
18	L?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)	
19		ltree_send_facility			
19		( tcv_invokeId := TSO_RANDOM_INVOKE_ID ())			
20	[ PC_CFNRc_CTM ]				(1)

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_CTM_KEYPAD_INTERROGATION ) ) ) )		(4)
22		[ PC_CFNRC_GSM ]			(2)
23		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invokes_S ( tcv_invokeld, PX_GSM_KEYPAD_INTERROGATION ) ) ) )		(4)
		ltree_release			
24		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
25		+ ST_Erase			
<b>Detailed Comments :</b> 1 IUT supports DectToGsm mode 2 IUT supports CTM mode 3 Receipt of a FACILITY message including an Encapsulatedstimulus invoke component including a display information 4 Send a FACILITY message containing an EncapsulatedStimulus invoke component with a keypad information					

### Test Case Dynamic Behaviour

**Test Case Name** : CFNRc\_N06\_001

**Group** : ServedUser/Operation/

**Purpose** : Ensure that IUT, in call state N02 for a CTM incoming call destined to user having previously successfully register and activated CFNRc, upon receipt of a CTMIncomingCallManagementInfo or a GSMIncomingCallManagementInfo return error with the congestion error value, does not send any indication to the served user.

**Configuration** : CONFIG0

**Default** : DEF\_CALL

**Comments** : EN 302 094-1, subclause 9.2.4

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR_N00_Init			
2		+ ST_Register			
3		+ ST_Activate			
4		O! DISPLAY START T_WAIT	Ds( "Initiate Incoming call invoke component")		
5		+ ST_Receive_SU_Incoming			
6		L ! PDUs	Ms ( CP_S1 ( tcv_flag_s, tcv_cref, tcv_bch_num))		
7		+ ltree_send_disconnect			
8		L ? PDUr CANCEL T_AC	Mr( RL_R2 ( tcv_flag_r, tcv_cref, 31))	(P)	5
9		L ! PDUs	Ms( RC_S1 ( tcv_flag_s, tcv_cref))		
10		? TIMEOUT T_AC		(F)	
11		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref)			
		ltree_send_disconnect			
12		[PC_CFNRc_CTM]			1
13		L ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref, 29, FIE_CTM_S01( CTM_IncomingCallMMInfo_R E ( tcv_invokeld, TSC_E_Congestion ) )))		3
14		[PC_CFNRc_GSM]			2
15		L ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref, 29, FIE_GSM_S01( GSM_IncomingCallMMInfo_ RE ( tcv_invokeld, TSC_E_Congestion ) )))		3

**Detailed Comments** : 1 IUT supports CTM mode

2 IUT supports DectToGsm mode

3 Receipt of a SETUP message including a CTMIncomingCallMMInfo or GSMIncomingCallMMInfo  
invoke component

4 Send a DISCONNECT message containing a CTMIncomingCallMMInfo or  
GSMIncomingCallMMInfo return error component

5 Receipt of a RELEASE message with indication of diversion (in a facility information element)

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N06_002 <b>Group</b> : ServedUser/Operation/ <b>Purpose</b> : Ensure that IUT, in call state N02 for a CTM incoming call destined to user having previously successfully register and activated CFNRc, upon receipt of a CTMIncomingCallManagementInfo or a GSMIncomingCallManagementInfo return error with the pagingFailure error value, does not send any indication to the served user. <b>Configuration</b> : CONFIG0 <b>Default</b> : DEF_CALL <b>Comments</b> : EN 302 094-1, subclause 9.2.4					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR_N00_Init			
2		+ ST_Register			
3		+ ST_Activate			
4		O! DISPLAY START T_WAIT	Ds( "Initiate Incoming call invoke component")		
5		+ ST_Receive_SU_Incoming			
6		L ! PDUs	Ms ( CP_S1 ( tcv_flag_s, tcv_cref, tcv_bch_num))		
7		+ ltree_send_disconnect			
8		L ? PDUr CANCEL T_AC	Mr( RL_R2 ( tcv_flag_r, tcv_cref, 31))	(P)	5
9		L ! PDUs	Ms( RC_S1 ( tcv_flag_s, tcv_cref))		
10		? TIMEOUT T_AC		(F)	
11		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref)			
		ltree_send_disconnect			
12		[PC_CFNRc_CTM]			1
13		L ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref, 29, FIE_CTM_S01( CTM_IncomingCallIMMInfo_RE ( tcv_invokeld, TSC_E_PagingFailure ) )))		3
14		[PC_CFNRc_GSM]			2
15		L ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref, 29, FIE_GSM_S01( GSM_IncomingCallIMMInfo_RE ( tcv_invokeld, TSC_E_PagingFailure ) )))		3
<b>Detailed Comments</b> : 1 IUT supports CTM mode 2 IUT supports DectToGsm mode 3 Receipt of a SETUP message including a CTMIncomingCallIMMInfo or GSMIncomingCallIMMInfo invoke component 4 Send a DISCONNECT message containing a CTMIncomingCallIMMInfo or GSMIncomingCallIMMInfo return error component 5 Receipt of a RELEASE message with indication of diversion (in a facility information element)					

### Test Case Dynamic Behaviour

**Test Case Name** : CFNRc\_N06\_003

**Group** : ServedUser/Operation/

**Purpose** : Ensure that IUT, in call state N02 for a CTM incoming call destined to user having previously successfully register and activated CFNRc, upon receipt of a CTMIncomingCallManagementInfo or a GSMIncomingCallManagementInfo return error with the radioConnectionFailure error value, does not send any indication to the served user.

**Configuration** : CONFIG0

**Default** : DEF\_CALL

**Comments** : EN 302 094-1, subclause 9.2.4

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR_N00_Init			
2		+ ST_Register			
3		+ ST_Activate			
4		O! DISPLAY START T_WAIT	Ds( "Initiate Incoming call invoke component")		
5		+ ST_Receive_SU_Incoming			
6		L ! PDUs	Ms ( CP_S1 ( tcv_flag_s, tcv_cref, tcv_bch_num))		
7		+ ltree_send_disconnect			
8		L ? PDUr CANCEL T_AC	Mr( RL_R2 ( tcv_flag_r, tcv_cref, 31))	(P)	5
9		L ! PDUs	Ms( RC_S1 ( tcv_flag_s, tcv_cref))		
10		? TIMEOUT T_AC		(F)	
11		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref)			
		ltree_send_disconnect			
12		[PC_CFNRc_CTM]			1
13		L ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref, 29, FIE_CTM_S01( CTM_IncomingCallMMInfo_R E ( tcv_invokeld, TSC_E_RadioConnectionFai lure ))))		3
14		[PC_CFNRc_GSM]			2
15		L ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref, 29, FIE_GSM_S01( GSM_IncomingCallMMInfo_ RE ( tcv_invokeld, TSC_E_RadioConnectionFai lure ))))		3

**Detailed Comments** : 1 IUT supports CTM mode

2 IUT supports DectToGsm mode

3 Receipt of a SETUP message including a CTMIncomingCallMMInfo or GSMIncomingCallMMInfo  
invoke component

Continued on next page

*Continued from previous page*

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	4 Send a DISCONNECT message containing a CTMIncomingCallMMInfo or GSMIncomingCallMMInfo return error component
	5 Receipt of a RELEASE message with indication of diversion (in a facility information element)

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N07_001 <b>Group</b> : ServedUser/ReminderNotification/ <b>Purpose</b> : Ensures that IUT, in state N00, on receipt of a valid SETUP message for a CTM outgoing call, returns a SETUP ACKNOWLEDGE or a CALL PROCEEDING message without the notification information element <b>Configuration</b> : CONFIG0 <b>Default</b> : DEF_CALL <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR_N00_Init			
2		+ ltree_send_SETUP			
3		L?PDUr CANCEL T_AC	Mr ( SUA_R2 ( tcv_flag_r, tcv_cref ))	(P)	1
4		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref)			
5		L?PDUr CANCEL T_AC	Mr ( CP_R2 ( tcv_flag_r, tcv_cref ))	(P)	2
6		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref)			
7		ltree_send_SETUP ( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld := TSO_RANDOM_INVOKE_ID() )			
8		[ PC_CFNRc_CTM ]			3
9		L! PDUs START T_AC	Ms ( SU_S2 ( 0, tcv_cref, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoRq_Invoke ( tcv_invokeld, PX_CTM_IPUI ) )))		5
10		[ PC_CFNRc_GSM ]			4
11		L! PDUs START T_AC	Ms ( SU_S2 ( 0, tcv_cref, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_Invoke ( tcv_invokeld, PX_GSM_PORTABLE_ID ) )))		5
<b>Detailed Comments</b> : 1 Receipt of a SETUP ACKNOWLEDGE message without notification information element 2 Receipt of a CALL PROCEEDING message without the Notification information element 3 IUT supports CTM mode 4 IUT supports DectToGsm mode 5 Receipt of a SETUP message including a CTMIncomingCallMMInfo or GSMIncomingCallMMInfo invoke component					



Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N08_001 <b>Group</b> : CallingUser/NotifDiv/ <b>Purpose</b> : Ensure that IUT in call state N01, to indicate that the first call diversion has occurred, sends a CALL PROCEEDING or NOTIFY message containing a Notification indicator information element coded "call is diverting" to the calling user. <b>Configuration</b> : CONFIG1 <b>Default</b> : DEF_CALL <b>Comments</b> : EN 302 094, subclause 9.2.2.1					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		CREATE ( PTC1 : ST1_PTC1_Activate )			
2		CREATE ( PTC2 : ST2_PTC2_IN )			
3		( tcv_config1 := TRUE )			
4		+ PR_N00_Init			
5		CPA1!CP_M	N00_Ready		
6		CPA1?CP_M	S_SU		
7		+ ltree_send_setup			
8		CPA2?CP_M	R_SU		
9		( tcv_stop_flag := FALSE )			
10		REPEAT ltree_Notification UNTIL [			
11		tcv_stop_flag ]			
12		+ PO_IC_OC ( tcv_flag_r,			
13		tcv_flag_s, tcv_cref )			
		? DONE ( PTC1)			
		? DONE ( PTC2)			
		ltree_Notification			
14		L?PDUr CANCEL T_AC	Mr ( CP_R3 ( tcv_flag_r,	(P)	
15		( tcv_stop_flag := TRUE )	tcv_cref ))		
16		L?PDUr CANCEL T_AC	Mr ( NO_R2 ( tcv_flag_r,	(P)	
17		( tcv_stop_flag := TRUE )	tcv_cref ))		
18		?TIMEOUT T_AC		(I)	
19		( tcv_stop_flag := TRUE )			
		ltree_send_setup			
20		( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld :=			
		TSO_RANDOM_INVOKE_ID() )			
21		[ PC_CFNRc_CTM ]			

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
22		L! PDUs START T_AC	Ms ( SU_S3 ( 0, tcv_cref, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoRq_Invoke ( tcv_invokeld, PX_CTM_IPUI )), tcv_bch_num ))		
23		[ PC_CFNRC_GSM ]			
24		L! PDUs START T_AC	Ms ( SU_S3 ( 0, tcv_cref, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_Invoke ( tcv_invokeld, PX_GSM_PORTABLE_ID )), tcv_bch_num ))		
<b>Detailed Comments</b> : 1 CTM supported 2 DestToGsm supported 3 Receipt of the notification information element in the CALL PROCEEDING of the the NOTIFY message					

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N09_001 <b>Group</b> : CallingUser/IdentificationDivertedTo/ <b>Purpose</b> : Ensure that IUT in the call state N04, to indicate completion of the call at the diverted-to user, sends a CONNECT message and possibly a NOTIFY or PROGRESS message none of which includes a redirection number information element. <b>Configuration</b> : CONFIG1 <b>Default</b> : DEF_CALL <b>Comments</b> : EN 032 094-1, subclause 9.2.3					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		CREATE ( PTC1 : ST1_PTC1_Activate )			
2		CREATE ( PTC2 : ST2_PTC4_IN )			
3		( tcv_config1 := TRUE, tcv_stop_flag := FALSE )			
4		+ PR_N04_Init			
5		START T_AC			
6		REPEAT ltree_Notification UNTIL [ tcv_stop_flag ]			
7		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
8		? DONE ( PTC1 )			
9		? DONE ( PTC2 )			
		ltree_Notification			
10		L?PDUr CANCEL T_AC	Mr ( CN_R3 ( tcv_flag_r, tcv_cref ) )	(P)	
11		( tcv_stop_flag := TRUE )			
12		L?PDUr CANCEL T_AC	Mr ( NO_R3 ( tcv_flag_r, tcv_cref ) )	(P)	
13		( tcv_stop_flag := TRUE )			
14		L?PDUr CANCEL T_AC	Mr ( PR_R2 ( tcv_flag_r, tcv_cref ) )	(P)	
15		( tcv_stop_flag := TRUE )			
16		?TIMEOUT T_AC		(F)	
17		( tcv_stop_flag := TRUE )			
<b>Detailed Comments</b> :					

### Test Case Dynamic Behaviour

**Test Case Name** : CFNRc\_N10\_001

**Group** : DivertedToUser/

**Purpose** : Ensure that the IUT in order to establish the call to the diverted-to user and if the indication is received that the presentation of the number is allowed and only one call diversion occurred,  
Sends a SETUP message containing one valid Redirecting number information element giving the reason for the call diversion with the presentation indicator set to "presentation restricted" .

**Configuration** : CONFIG0

**Default** : DEF\_CALL

**Comments** : EN 302 094-1 subclause 9.2.5.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		CREATE ( PTC1 : ST1_PTC1_Activate )			
2		CREATE ( PTC2 : ST2_PTC_IN )			
3		( tcv_config1 := TRUE )			
4		+ PR_N00_Init			
5		CPA1!CP_M	N00_Ready		
6		CPA1?CP_M	S_SU		
7		CPA2!CP_M	R_SU_Rgnb		
8		+ ltree_send_setup			
9		CPA2?CP_M	R_SU	(P)	
10		L? PDUr CANCEL T_WAIT	Mr( RC_R1 ( tcv_flag_r, tcv_cref))		
11		? DONE ( PTC1 )			
12		? DONE ( PTC2 )			
		ltree_send_setup			
13		( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld := TSO_RANDOM_INVOKE_ID() )			
14		[ PC_CFNRc_CTM ]			
15		L! PDUs START T_WAIT	Ms ( SU_S3 ( 0, tcv_cref, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoR q_Invoke ( tcv_invokeld, PX_CTM_IPUI )), tcv_bch_num ))		
16		[ PC_CFNRc_GSM ]			
17		L! PDUs START T_WAIT	Ms ( SU_S3 ( 0, tcv_cref, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_I nvoke ( tcv_invokeld, PX_GSM_PORTABLE_ID )), tcv_bch_num ))		

**Detailed Comments** :

Test Case Dynamic Behaviour					
<b>Test Case Name</b> : CFNRc_N10_002 <b>Group</b> : DivertedToUser/ <b>Purpose</b> : Ensure that the IUT in order to establish the call to the diverted-to user and if the indication is received that the presentation of the number is allowed and only one call diversion occurred, Sends a SETUP message containing one valid Redirecting number information element giving the reason for the call diversion with the presentation indicator set to "presentation allowed" and the redirecting number information provided in the number digits field. <b>Configuration</b> : CONFIG0 <b>Default</b> : DEF_CALL <b>Comments</b> : EN 302 094-1 subclause 9.2.5.1					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		CREATE ( PTC1 : ST1_PTC1_Activate )			
2		CREATE ( PTC2 : ST2_PTC_IN )			
3		( tcv_config1 := TRUE )			
4		+ PR_N00_Init			
5		CPA1!CP_M	N00_Ready		
6		CPA1?CP_M	S_SU		
7		CPA2!CP_M	R_SU_RgnbR		
8		+ ltree_send_setup			
9		CPA2?CP_M	R_SU	(P)	
10		L? PDUr CANCEL T_WAIT	Mr( RC_R1 ( tcv_flag_r, tcv_cref))		
11		? DONE ( PTC1 )			
12		? DONE ( PTC2 )			
		ltree_send_setup			
13		( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld := TSO_RANDOM_INVOKE_ID() )			
14		[ PC_CFNRc_CTM ]			
15		L! PDUs START T_WAIT	Ms ( SU_S3 ( 0, tcv_cref, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoR q_Invoke ( tcv_invokeld, PX_CTM_IPUI ) ), tcv_bch_num ) )		
16		[ PC_CFNRc_GSM ]			
17		L! PDUs START T_WAIT	Ms ( SU_S3 ( 0, tcv_cref, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_I nvoke ( tcv_invokeld, PX_GSM_PORTABLE_ID ) ), tcv_bch_num ) )		
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PR_N00 <b>Group</b> : Preambles/ <b>Objective</b> : Preamble to the Null call state N00 to be used with non-NCICS connections. <b>Default</b> : DEF_CALL <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+LT_INIT_VARIABLES			(1)
2		[PX_L2_INIT]			
3		L!DL_REL_RQ START T_AC			(2)
4		L?DL_REL_CO CANCEL T_AC		(P)	(3)
5		L!DL_EST_RQ START T_AC			(4)
6		L?DL_EST_CO CANCEL T_AC		(P)	(5)
7		+WAIT_RESTART			(6)
8		L?DL_REL_IN START T_NOAC			(7)
9		L?DL_EST_IN CANCEL T_AC , CANCEL T_NOAC		(P)	(8)
10		+WAIT_RESTART			(6)
11		?TIMEOUT T_NOAC			
12		L!DL_EST_RQ			(4)
13		GOTO L1			
14		L?OTHERWISE		I	
15		L?DL_EST_IN CANCEL T_AC , START T_NOAC			(8)
16		L?DL_EST_CO CANCEL T_NOAC		(P)	(9)
17		+WAIT_RESTART			
18		?TIMEOUT T_NOAC		I	no respons e
19		L?OTHERWISE		I	(10)
20		?TIMEOUT T_AC		I	no respons e
21		L?OTHERWISE		I	(10)
22		?TIMEOUT T_AC		I	no respons e
23		L?OTHERWISE		I	(10)
24		[NOT PX_L2_INIT] LT_INIT_VARIABLES			
25		[PX_BASIC]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
26		(tcv_cref:='0000001'B, tcv_glob_cref:='0000000'B, tcv_invokeld := TSO_RANDOM_INVOKE_ID(), tcv_bch_num := INT_TO_BIT( PX_CH_NUM ,2))			Basic access
27		[NOT PX_BASIC]			
28		(tcv_cref:='0000000000000001'B, tcv_glob_cref:='000000000000000'B, tcv_invokeld := TSO_RANDOM_INVOKE_ID(), tcv_bch_num := INT_TO_BIT( PX_CH_NUM ,7))			Primary rate access
		WAIT_RESTART			
29		[PX_WAIT_RESTART]			
30		START T_RESTART			
31		L?RESTARTr CANCEL T_RESTART	RSr(RST_R1(0,tcv_glob_cref ,6))		Single interface
32		L!PDUs	Ms(RSA_S1(1,tcv_glob_cref, 6))		
33		?TIMEOUT T_RESTART			
34		[NOT PX_WAIT_RESTART]			
<p><b>Detailed Comments :</b> The layer 2 of the IUT must have a TEI assigned value before the execution of this preamble.</p> <p>The procedure to assign the TEI value to the IUT is a matter for the test laboratory.</p> <p>(1) The local subtree INIT_VARIABLES is used to assign initial values to test case variables taking into account the used interface configuration.</p> <p>(2) Termination of the multiple frame operation is requested (A DISC frame is sent).</p> <p>(3) Termination of the multiple frame operation is confirmed (A UA or a DM frame is received).</p> <p>(4) Establishment of the multiple frame operation is requested (A SABME frame is sent).</p> <p>(5) Establishment of the multiple frame operation is confirmed (A UA frame is received).</p> <p>(6) The local subtree WAIT_RESTART is used to deal with the receipt of RESTART messages that may be sent by the IUT after the re-establishment of the multiple frame operation.</p> <p>(7) An unsuccessful establishment attempt is reported (A DM frame is received).</p> <p>(8) Establishment of the multiple frame operation is indicated (A SABME frame is received and a UA frame is sent).</p> <p>(9) Establishment of the multiple frame operation (requested in line 4) is confirmed (A UA frame is received).</p> <p>(10) Any other event occurred.</p>					

### Test Step Dynamic Behaviour

**Test Step Name** : PR\_N00\_Init

**Group** : Preambles/

**Objective** : Preamble to the Null call state N00 with subscription registration (if supported) and location registration.

**Default** : DEF\_NCICS

**Comments** :

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+PR_N00			(1)
2		[ PC_CFNRC_CTM ]			
3		[PC_CTM_SR AND PX_CTM_SR_EachTest]			
4		+ ST_CTMAccessRightRequest			
5		[ PX_CTM_LR_EachTest ]			
6		+ ST_CTMLocationRegistration			
7		[ NOT ( PX_CTM_LR_EachTest ) ]			
8		[NOT (PC_CTM_SR ) OR NOT (PX_CTM_SR_EachTest )]			
9		[ PX_CTM_LR_EachTest ]			
10		+ ST_CTMLocationRegistration			
11		[ NOT ( PX_CTM_LR_EachTest ) ]			
12		[ PC_CFNRC_GSM ]			
13		[ PX_DG_LR_EachTest ]			
14		+ST_GSMLocationRegistration			
15		[ NOT ( PX_DG_LR_EachTest ) ]			

**Detailed Comments** :



Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PR_N04_Init <b>Group</b> : Preambles/ <b>Objective</b> : Preamble to the Null call state N04 with subscription registration (if supported) and location registration. <b>Default</b> : DEF_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR_N00_Init			
2		CPA1!CP_M	N00_Ready		
3		CPA1?CP_M	S_SU		
4		+ ltree_send_setup			
5		CPA2?CP_M START T_AC	R_SU		
6		L?PDUr CANCEL T_AC, START T_AC	Mr ( CP_R3 ( tcv_flag_r, tcv_cref ))		
7		L?PDUr CANCEL T_AC	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))		
		ltree_send_setup			
8		( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld := TSO_RANDOM_INVOKE_ID() )			
9		[ PC_CFNRC_CTM ]			
10		L! PDUr START T_AC	Ms ( SU_S3 ( 0, tcv_cref, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoR q_Invoke ( tcv_invokeld, PX_CTM_IPUI )), tcv_bch_num ))		
11		[ PC_CFNRC_GSM ]			
12		L! PDUr START T_AC	Ms ( SU_S3 ( 0, tcv_cref, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_I nvoke ( tcv_invokeld, PX_GSM_PORTABLE_ID )), tcv_bch_num ))		
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_Activate					
<b>Group</b> : Steps/					
<b>Objective</b> : To activate IUT to CFNRc SS.					
<b>Default</b> : DEF_CALL					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	ST1	+ ST_N02_CTMOutgoing	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))		
2		+ ltree_send_facility			
3		START T_AC			
4		L? PDUr CANCEL T_AC			
5		GOTO ST1			
6		L? PDUr CANCEL T_AC			
7		L!PDUr			
8		GOTO ST1			
9		+ ltree_receive_fac			
10		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
11		? TIMEOUT T_AC			
12		+ PO_IC_OC_Stop ( tcv_flag_r, tcv_flag_s, tcv_cref )			
		ltree_receive_fac			
13		[ PC_CFNRc_GSM ]			
14		L?PDUr CANCEL T_AC			
15		[ PC_CFNRc_CTM ]			
16		L?PDUr CANCEL T_AC			
		ltree_send_facility			
17	( tcv_invokeld := TSO_RANDOM_INVOKE_ID ())				
18	[ PC_CFNRc_CTM ]				
19	L!PDUr START T_AC				
20	[ PC_CFNRc_GSM ]				
		Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(1)	
		Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)	
		Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invoked_S ( tcv_invokeld, PX_CTM_KEYPAD_ACTIVATION ) ) ) )		(2)	
				(4)	

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_GSM_KEYPAD_ACTIV ATION ) ) ) )		(4)
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_CTMAccessRightRequest <b>Group</b> : Steps/ <b>Objective</b> : <b>Default</b> : DEF_NCICS <b>Comments</b> : To proceed to the subscription registration with the IUT.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L! PDUs START T_AC	Ms ( SU_S1 ( 0 , tcv_cref , FIE_CTM_S01 ( CTM_AccessRightRq_Invok e ( tcv_invokeld , PX_CTM_IPUI))))		(1)
2		( tcv_flag_r := 1, tcv_flag_s := 0)			
3		+ ST_U06_To_U10 ( tcv_flag_r , tcv_flag_s , tcv_cref )			
4		O ! DISPLAY START T_WAIT	Ds( "Initiate subscription registration – Access right request return result component")		
5		+LTS_KA_Invoke			
6		+LTS_TA_Invoke			
7		+LTS_NIC_Invoke			
8		+LTS_AR_ReturnResult			
9		+LTS_timeout			
		LTS_NIC_Invoke			
10		L? PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID()) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_Ciphering_Invoke ( PX_CTM_IPUI ))))		(4)
11		L! PDUs START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 ( CTM_Ciphering_RR ( tcv_invokeld2 ))))		
12		+LTS_AR_ReturnResult			
13		+LTS_timeout			
		LTS_TA_Invoke			
14		L? PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_authtype := TSO_GET_CTM_authtype () ) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_TerminalAuth_Invoke ( PX_CTM_IPUI ))))		(5)
15		( tcv_ctm_xres1 := TSO_CTM_CALC_RES2 ( tcv_ctm_rs, tcv_rand , tcv_ctm_authtype ) )			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L! PDU <sub>s</sub> START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 ( CTM_TerminalAuth_RR ( tcv_invokeld2 , tcv_ctm_xres1 ))))		
17		+LTS_NIC_Invoke			
18		+LTS_AR_ReturnResult			
19		+LTS_timeout			
		LTS_KA_Invoke			
20		L? PDU <sub>r</sub> ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_alloctype := TSO_GET_CTM_alloctype () ) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_KeyAllocate_Invoke ( PX_CTM_IPUI ))))		(6)
21		( tcv_invokeld3 := TSO_RANDOM_INVOKE_ID () , tcv_ctm_xres1 := TSO_CTM_CALC_RES ( tcv_ctm_rs, tcv_rand , tcv_ctm_alloctype ) )			
22		L! PDU <sub>s</sub>	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_NetworkAuth_Invoke ( tcv_invokeld3, PX_CTM_IPUI ))))		
23		L! PDU <sub>s</sub> START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_KeyAllocate_RR ( tcv_invokeld2 , tcv_ctm_xres1 ))))		
24		L? PDU <sub>r</sub> START T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_NetworkAuth_RR ( tcv_invokeld3 ))))		
25		+LTS_TA_Invoke			
26		+LTS_timeout			
		LTS_AR_ReturnResult			
27		L? PDU <sub>r</sub> CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_AccessRightRq_RR ( tcv_invokeld ))))		(2)
28		+ PO_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
		LTS_timeout			
29		?TIMEOUT T_WAIT			(3)
30		+ PO_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
31		+ ST_Stop_PTCs			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
32		CANCEL		I	
<b>Detailed Comments</b> : (1) sending of a SETUP message including a CTM_AccessRightsRequest invoke component. (2) receipt of a FACILITY message with a CTM_AccessRightsRequest return result component. (3) Nothing received, cancel all running timers if present , immediat verdict Inconclusive (4) receipt of a FACILITY message with a CTM_Ciphering invoke component. (5) receipt of a FACILITY message with a CTM_TerminalAuth invoke component. (6) receipt of a FACILITY message with a CTM_KeyAllocate invoke component.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_CTMLocationRegistration <b>Group</b> : Steps/ <b>Objective</b> : <b>Default</b> : DEF_NCICS <b>Comments</b> : To proceed to the location registration procedure with the IUT.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L! PDUs START T_AC	Ms ( SU_S1 ( 0 , tcv_cref , FIE_CTM_S01 (CTM_LocationReg_Invoke ( tcv_invokeld , PX_CTM_IPUI))))		(1)
2		( tcv_flag_r := 1, tcv_flag_s := 0)			
3		+ ST_U06_To_U10 ( tcv_flag_r , tcv_flag_s , tcv_cref )			
4		O ! DISPLAY START T_WAIT	Ds( "Initiate location registration – CTMLocationRegistration return result component")		
5		+LTS_KA_Invoke			
6		+LTS_TA_Invoke			
7		+LTS_LR_ReturnResult			
8		+LTS_timeout			
9		LTS_TA_Invoke L? PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_authtype := TSO_GET_CTM_authtype () ) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_TerminalAuth_Invoke ( PX_CTM_IPUI ) )))		(4)
10		( tcv_ctm_xres1 := TSO_CTM_CALC_RES2 ( tcv_ctm_rs, tcv_rand , tcv_ctm_authtype ) )			
11		L! PDUs START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_TerminalAuth_RR ( tcv_invokeld2 , tcv_ctm_xres1 ) )))		
12		+LTS_LR_ReturnResult			
13		+LTS_timeout			
14		LTS_KA_Invoke L? PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_alloctype := TSO_GET_CTM_alloctype () ) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_KeyAllocate_Invoke ( PX_CTM_IPUI ) )))		(5)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
15		( tcv_invokeld3 := TSO_RANDOM_INVOKE_ID(), tcv_ctm_xres1 := TSO_CTM_CALC_RES (			
16		tcv_ctm_rs, tcv_rand , tcv_ctm_alloctype ) )  L! PDUs	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_NetworkAuth_Invoke (		
17		L! PDUs START T_WAIT	tcv_invokeld3, PX_CTM_IPUI ))))		
18		L? PDUr START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_KeyAllocate_RR (		
19		L? PDUr START T_WAIT	tcv_invokeld2 , tcv_ctm_xres1 ))))		
20		+LTS_TA_Invoke	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_NetworkAuth_RR (		
21		+LTS_timeout LTS_LR_ReturnResult	tcv_invokeld3 ))))		
22		L? PDUr CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_LocationReg_RR (		(2)
23		+ PO_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )	tcv_invokeld ))))		
24		LTS_timeout			
25		?TIMEOUT T_WAIT			(3)
26		+ PO_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
		+ ST_Stop_PTCs			
		CANCEL		I	
<b>Detailed Comments</b> : (1) sending of a SETUP message including a CTMLocationregistration invoke component. (2) receipt of a FACILITY message with a CTMLocationregistration return result component. (3) Nothing received, cancel all running timers if present , immediat verdict Inconclusive (4) receipt of a FACILITY message with a CTM_TerminalAuth invoke component. (5) receipt of a FACILITY message with a CTM_KeyAllocate invoke component.					



Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_Erase					
<b>Group</b> : Steps/					
<b>Objective</b> : To erase IUT to CFNRc SS.					
<b>Default</b> :					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	ST1	+ ST_N02_CTMOutgoing	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))		
2		+ ltree_send_facility			
3		START T_AC			
4		L? PDUr CANCEL T_AC			
5		GOTO ST1			
6		L? PDUr CANCEL T_AC			
7		L!PDUr			
8		GOTO ST1			
9		+ ltree_receive_fac			
10		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
11		? TIMEOUT T_AC			
12		+ PO_IC_OC_Stop ( tcv_flag_r, tcv_flag_s, tcv_cref )			
13		ltree_receive_fac			
14		[ PC_CFNRc_GSM ]			
15		L?PDUr CANCEL T_AC			
16		[ PC_CFNRc_CTM ]			
17		L?PDUr CANCEL T_AC			
18		ltree_send_facility			
19		( tcv_invokeld := TSO_RANDOM_INVOKE_ID () )			
20		[ PC_CFNRc_CTM ]			
	L!PDUr START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invok e_S ( tcv_invokeld, PX_CTM_KEYPAD_ERASU RE ) ) ) )	(F)	(1)	
		Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( EncapsulatedStimulus_Invo ke_R ) ) )	(P)	(3)	
		Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_CTM_R01 ( EncapsulatedStimulus_Invo ke_R ) ) )	(P)	(3)	
				(2)	
				(4)	
				(1)	
				(2)	

Continued on next page

*Continued from previous page*

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L!PDU <sub>s</sub> START T <sub>AC</sub>	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_GSM_KEYPAD_ERASU RE ) ) ) )		(4)
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_GSMLocationRegistration <b>Group</b> : Steps/ <b>Objective</b> : Performing the GSM Location Registration procedure, accounting for possible embedded procedures <b>Default</b> : DEF_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		O! DISPLAY	Ds( "Initiate location registration – GSMLocationRegistration return result component")		
2		( tcv_invokeId := TSO_RANDOM_INVOKE_ID ())			
3		L! PDUs START T_AC	Ms ( SU_S1( 0, tcv_cref,FIE_GSM_S01( GSM_LocationRegistration_I nvoke ( tcv_invokeId, PX_GSM_PORTABLE_ID))))		(1)
4		( tcv_flag_r := 1, tcv_flag_s := 0)			
5		+ ST_U06_To_U10 ( tcv_flag_r, tcv_flag_s, tcv_cref)			
6		REPEAT ST_GSM_EMB_PROCS (tcv_flag_r, tcv_flag_s, tcv_cref) UNTIL [tcv_rcv_locreg OR tcv_rcv_timeout]			(2)
7		+PO_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref)			(3)
8		[ tcv_rcv_timeout ]		(I)	(4)
9		+ ST_Stop_PTCs			
10		[ tcv_rcv_locreg ]		(P)	(5)
11		?TIMEOUT T_WAIT		I	
<b>Detailed Comments</b> : (1) sending a SETUP message including a GSMLocationRegistration invoke component. (2) handling of one or more embedded procedures and finally receiving a GSMLocationRegistration return result component (3) releasing the connection (4) if a timeout had occurred the test ends with an INCONCLUSIVE verdict (5) otherwise the location registration had been successful?					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_GSM_EMB_PROCS (flag_r, flag_s: INTEGER ; cref: BIT7OR15) <b>Group</b> : Steps/ <b>Objective</b> : Handling of embedded procedures following a Location Registration request <b>Default</b> : DEF_NCICS <b>Comments</b> : EN 301 144-1 subclause 9.2.1.1.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_rand := TSO_GET_RAND(), tcv_cipherinfo := TSO_GET_CIPHERINFO())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( GSM_TerminalAuth_Invoke (OMIT))))		(1)
2		L!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 ( GSM_TerminalAuth_RR ( tcv_invokeld2, TSO_GSM_CALC_RES(tcv_ rand))))))	(P)	(2)
3		[NOT tcv_rcv_termauth]		(P)	(3)
4		(tcv_rcv_termauth := TRUE)			
5		[tcv_rcv_termauth]		(I)	(4)
6		L?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_idtype := TSO_GET_IDTYPE())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_IdentityReq_Invoke (OMIT))))	(P)	(5)
7		L!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_IdentityReq_RR ( tcv_invokeld2, TSO_GSM_CALC_PID (tcv_idtype))))))	(P)	(6)
8		[NOT tcv_rcv_idreq]		(P)	(7)
9		(tcv_rcv_idreq := TRUE)			
10		[tcv_rcv_idreq]		(I)	(8)
11		L?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_Ciphering_Invoke (OMIT))))	(P)	(9)
12		L!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_Ciphering_RR ( tcv_invokeld2))))	(P)	(10)
13		[NOT tcv_rcv_niciph]		(P)	(11)
14		(tcv_rcv_niciph := TRUE)			
15		[tcv_rcv_niciph]		(I)	(12)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_tmsi := TSO_GET_TMSI())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_AssignIdentity_Invok e (OMIT))))	(P)	(13)
17		L!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_AssignIdentity_RR ( tcv_invokeld2))))	(P)	(14)
18		[NOT tcv_rcv_tmpidass]		(P)	(15)
19		(tcv_rcv_tmpidass := TRUE)			
20		[tcv_rcv_tmpidass]		(I)	(16)
21		L?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_tmsi := TSO_GET_TMSI())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LinkedAssignId_Invo ke)))	(P)	(17)
22		L?PDUr (tcv_rcv_locreg := TRUE) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LocationRegistration_ RR ( tcv_invokeld))))	(P)	(18)
23		L!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_LinkedAssignId_RR ( tcv_invokeld2))))	(P)	(19)
24		[NOT tcv_rcv_tmpidass]		(P)	(20)
25		(tcv_rcv_lidass := TRUE)			
26		[tcv_rcv_tmpidass]		(F)	(21)
27		L?PDUr (tcv_rcv_locreg := TRUE) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LocationRegistration_ RR ( tcv_invokeld))))	(P)	(18)
28		?TIMEOUT T_WAIT (tcv_rcv_timeout := TRUE)		(I)	(22)

**Detailed Comments : ST\_EMB\_PROCS:**

- handles embedded procedures during Location Registration
- one or more of the 4 allowed procedures may be applied by the IUT
- if one of these procedures is used more than once an INCONC verdict is assigned, but the test is continued
- if the Linked Assignment procedure is applied it is checked that it has been used instead of the Temporary Assignment procedure, otherwise a FAIL verdict is assigned
- this step runs until the Location Registration has been accepted by the network, or until a timeout supervising the test occurs

- (1) receiving of a FACILITY message including a GSMTerminalAuthentication invoke component. PortableIdentity has to be omitted. Store InvokeID, Random number and CipherInfo.
- (2) sending a FACILITY message including a GSMTerminalAuthentication return result component
- (3) PASS if procedure was activated the first time
- (4) INCONC if procedure was activated more than once

Continued on next page

*Continued from previous page***Test Step Dynamic Behaviour****Detailed Comments : ...**

- (5) receiving of a FACILITY message including a GSMIdentityRequest invoke component.  
PortableIdentity has to be omitted.
- (6) sending a FACILITY message including a GSMIdentityRequest return result component
- (7) PASS if procedure was activated the first time
- (8) INCONC if procedure was activated more than once
- (9) receiving of a FACILITY message including a GSMCIPHERING invoke component.  
PortableIdentity has to be omitted.
- (10) sending a FACILITY message including a GSMCIPHERING return result component
- (11) PASS if procedure was activated the first time
- (12) INCONC if procedure was activated more than once
- (13) receiving of a FACILITY message including a GSMAssignIdentity invoke component.  
PortableIdentity has to be omitted.
- (14) sending a FACILITY message including a GSMAssignIdentity return result component
- (15) PASS if procedure was activated the first time
- (16) INCONC if procedure was activated more than once
- (17) receiving of a FACILITY message including a GSMLinkedAssignIdentity invoke component.
- (18) receiving a FACILITY message including a GSMLocationRegistration return result component
- (19) sending a FACILITY message including a GSMLinkedAssignIdentity return result component
- (20) PASS if the TEMPORARY IDENTITY ASSIGNMENT procedure had been not activated
- (21) FAIL !! if the TEMPORARY IDENTITY ASSIGNMENT procedure had been activated
- (22) Timer T\_WAIT has elapsed

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_N02_CTMOutgoing					
<b>Group</b> : Steps/					
<b>Objective</b> : To bring IUT to state N02 for a CTM outgoing call.					
<b>Default</b> : DEF_CALL					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ ltree_send_SETUP	Mr ( SUA_R1 ( tcv_flag_r, tcv_cref ))		
2		L?PDUr CANCEL T_AC			
		ltree_send_SETUP			
3		( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld := TSO_RANDOM_INVOKE_ID() )			
4		[ PC_CFNRC_CTM ]			
5		L! PDUs START T_AC	Ms ( SU_S2 ( 0, tcv_cref, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoRq_Invoke ( tcv_invokeld, PX_CTM_IPUI )))		
6		[ PC_CFNRC_GSM ]			
7		L! PDUs START T_AC	Ms ( SU_S2 ( 0, tcv_cref, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_Invoke ( tcv_invokeld, PX_GSM_PORTABLE_ID )))		
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_Receive_SU_Incoming <b>Group</b> : Steps/ <b>Objective</b> : To receive SETUP message for a CTM incoming call depending on the CTM/DectToGsm and basic/primary option. <b>Default</b> : DEF_CALL <b>Comments</b> : To receive SETUP message depending on the CTM/DectToGsm and basic/primary option.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[PC_CFNRC_CTM]			
2		L? SETUPr [NOT PX_BASIC] ( tcv_cref := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI ))))	(P)	
3		L? SETUPr [PX_BASIC] ( tcv_cref := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI ))))	(P)	
4		[PC_CFNRC_GSM]			
5		L? SETUPr [NOT PX_BASIC] ( tcv_cref := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID))) )	(P)	
6		L? SETUPr [PX_BASIC] ( tcv_cref := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID))) )	(P)	
<b>Detailed Comments</b> :					



Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_Register					
<b>Group</b> : Steps/					
<b>Objective</b> : To register IUT to CFNRc SS.					
<b>Default</b> :					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	ST1	+ ST_N02_CTMOutgoing	Mr ( AL_R1 ( tcv_flag_r, tcv_cref ))		
2		+ ltree_send_facility			
3		START T_AC			
4		L? PDUr CANCEL T_AC			
5		GOTO ST1			
6		L? PDUr CANCEL T_AC			
7		L!PDUr			
8		GOTO ST1			
9		+ ltree_receive_fac			
10		+ PO_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref )			
11		? TIMEOUT T_AC			
12		+ PO_IC_OC_Stop ( tcv_flag_r, tcv_flag_s, tcv_cref )			
13		ltree_receive_fac			
14		[ PC_CFNRc_GSM ]			
15		L?PDUr CANCEL T_AC			
16		[ PC_CFNRc_CTM ]			
17		L?PDUr CANCEL T_AC			
18		ltree_send_facility			
19		( tcv_invokeld := TSO_RANDOM_INVOKE_ID ())			
20		[ PC_CFNRc_CTM ]			
	L!PDUr START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (EncapsulatedStimulus_Invok e_S ( tcv_invokeld, PX_CTM_KEYPAD_REGIST RATION ) ) ) )	(F)	(1)	(3)
			(P)	(2)	(3)
				(P)	(3)
					(1)
					(4)
					(1)
					(2)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L!PDU <sub>s</sub> START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref, FIE_GSM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_GSM_KEYPAD_REGIS TRATION ) ) ) )		(4)
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_Stop_PTCs <b>Group</b> : Steps/ <b>Objective</b> : <b>Default</b> : <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[tcv_config1=TRUE]			
2		CPA1!CP_M	STOP_PTC		
3		CPA2!CP_M	STOP_PTC	R	
4		[TRUE]		R	
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST_U06_To_U10 (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : Steps/ <b>Objective</b> : To complete the establishment of an outgoing NCICS call. Bring IUT from U06 to U10. <b>Default</b> : DEF_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L? PDUr CANCEL T_AC	Mr ( CP_R1(flag_r, cref) )		(1)
2		O!DISPLAY START T_WAIT	Ds("IUT send a CONNECT message")		
3		L? PDUr CANCEL T_WAIT	Mr ( CN_R1(flag_r, cref) )	(P)	(1)
4		L!PDUr	Ms (CA_S1(flag_s, cref))		
5		?TIMEOUT T_WAIT		(I)	
6		+PO_NCICS(flag_r, flag_s, cref)			
7		?TIMEOUT T_AC		(I)	
8		+PO_NCICS(flag_r, flag_s, cref)			
<b>Detailed Comments</b> : (1) sends a CALL PROCEEDING message for a NCICS call. (2) sends a CONNECT message for a NCICS call. (3) receipt of a CONNECT ACKNOWLEDGE message.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO1_IC_OC (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : PTC1_Steps/ <b>Objective</b> : To bring the IUT back to the Null call state U00. <b>Default</b> : DEF1_CALL(flag_s) <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1! PDUr START T_AC	Ms( DI_S1 ( flag_s, cref, 16))		(1)
2		L1? PDUr CANCEL T_AC	Mr( RL_R1 ( flag_r, cref))		(2)
3		L1! PDUr	Ms( RC_S1 ( flag_s, cref))		(3)
4		? TIMEOUT T_AC		(I)	(4)
5		L1? OTHERWISE		(I)	(5)
<b>Detailed Comments</b> : (1) A valid DISCONNECT message indicating the cause value 16 "Normal call clearing" is sent. (2) A valid RELEASE message is received from the IUT. (3) A RELEASE COMPLETE message is sent to the IUT. (4) No response. (5) An invalid event occurred.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO1_IC_OC_Stop (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : PTC1_Steps/ <b>Objective</b> : To bring the IUT back to the Null call state U00 and stop the test. <b>Default</b> : DEF1_CALL (flag_s) <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1! PDUs START T_AC	Ms( DI_S1 ( flag_s, cref, 16))		(1)
2		L1? PDUr CANCEL T_AC	Mr( RL_R1 ( flag_r, cref))		(2)
3		L1! PDUs	Ms( RC_S1 ( flag_s, cref))	R	(3)
4		? TIMEOUT T_AC		R	(4)
5		L1? OTHERWISE		R	(5)
<b>Detailed Comments</b> : (1) A valid DISCONNECT message indicating the cause value 16 "Normal call clearing" is sent. (2) A valid RELEASE message is received from the IUT. (3) A RELEASE COMPLETE message is sent to the IUT. (4) No response. (5) An invalid event occurred.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO1_NCICS (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : PTC1_Steps/ <b>Objective</b> : To bring the IUT back to the Null call state U00. <b>Default</b> : DEF1_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1!PDUs START T_AC	Ms( RL_S1 ( flag_s, cref, 16))		(1)
2		L1?PDUr CANCEL T_AC	Mr( RC_R1 ( flag_r, cref))		(2)
3		?TIMEOUT T_AC		(I)	(4)
4		L1?OTHERWISE		(I)	(3)
<b>Detailed Comments</b> : (1) A valid RELEASE message indicating the cause value 16 "Normal call clearing" is sent. (2) A RELEASE COMPLETE message is received from the IUT. (3) An invalid event occurred. (4) No response.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PR1_N00 <b>Group</b> : PTC1_Steps/ <b>Objective</b> : Preamble to the Null call state N00 to be used with non-NCICS connections for PTC1. <b>Default</b> : DEF1_CALL(0) <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+LT_INIT_VARIABLES			(1)
2		[PX_L2_INIT]			
3		L1!DL_REL_RQ START T_AC			(2)
4		L1?DL_REL_CO CANCEL T_AC		(P)	(3)
5		L1!DL_EST_RQ START T_AC			(4)
6		L1?DL_EST_CO CANCEL T_AC		(P)	(5)
7		+WAIT_RESTART			(6)
8		L1?DL_REL_IN START T_NOAC			(7)
9		L1?DL_EST_IN CANCEL T_AC , CANCEL T_NOAC		(P)	(8)
10		+WAIT_RESTART			(6)
11		?TIMEOUT T_NOAC			
12		L1!DL_EST_RQ			(4)
13		GOTO L1			
14		L1?OTHERWISE		I	
15		L1?DL_EST_IN CANCEL T_AC , START T_NOAC			(8)
16		L1?DL_EST_CO CANCEL T_NOAC		(P)	(9)
17		+WAIT_RESTART			
18		?TIMEOUT T_NOAC		I	no respons e
19		L1?OTHERWISE		I	(10)
20		?TIMEOUT T_AC		I	no respons e
21		L1?OTHERWISE		I	(10)
22		?TIMEOUT T_AC		I	no respons e
23		L1?OTHERWISE		I	(10)
24		[NOT PX_L2_INIT] LT_INIT_VARIABLES			
25		[PX_BASIC]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
26		(tcv_cref1:='0000001'B, tcv_cref:='0000001'B, tcv_glob_cref1:='0000000'B, tcv_bch_num1 := INT_TO_BIT( PX_CH_NUM ,2))			Basic access
27		[NOT PX_BASIC]			
28		(tcv_cref1:='0000000000000001'B, tcv_cref:='0000001'B, tcv_glob_cref1:='0000000000000000'B, tcv_bch_num1 := INT_TO_BIT( PX_CH_NUM ,7))			Primary rate access
		WAIT_RESTART			
29		[PX_WAIT_RESTART]			
30		START T_RESTART			
31		L1?RESTARTr CANCEL T_RESTART	RSr(RST_R1(0,tcv_glob_cref ,6))		Single interface
32		L1!PDUs	Ms(RSA_S1(1,tcv_glob_cref, 6))		
33		?TIMEOUT T_RESTART			
34		[NOT PX_WAIT_RESTART]			
<p><b>Detailed Comments</b> : The layer 2 of the IUT must have a TEI assigned value before the execution of this preamble.</p> <p>The procedure to assign the TEI value to the IUT is a matter for the test laboratory.</p> <p>(1) The local subtree INIT_VARIABLES is used to assign initial values to test case variables taking into account the used interface configuration.</p> <p>(2) Termination of the multiple frame operation is requested (A DISC frame is sent).</p> <p>(3) Termination of the multiple frame operation is confirmed (A UA or a DM frame is received).</p> <p>(4) Establishment of the multiple frame operation is requested (A SABME frame is sent).</p> <p>(5) Establishment of the multiple frame operation is confirmed (A UA frame is received).</p> <p>(6) The local subtree WAIT_RESTART is used to deal with the receipt of RESTART messages that may be sent by the IUT after the re-establishment of the multiple frame operation.</p> <p>(7) An unsuccessful establishment attempt is reported (A DM frame is received).</p> <p>(8) Establishment of the multiple frame operation is indicated (A SABME frame is received and a UA frame is sent).</p> <p>(9) Establishment of the multiple frame operation (requested in line 4) is confirmed (A UA frame is received).</p> <p>(10) Any other event occurred.</p>					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PR1_N00_Init					
<b>Group</b> : PTC1_Steps/					
<b>Objective</b> : Preamble to the Null call state N00 with subscription registration (if supported) and location registration.					
<b>Default</b> : DEF1_NCICS					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+PR1_N00			(1)
2		[ PC_CFNrc_CTM ]			
3		[PC_CTM_SR AND PX_CTM_SR_EachTest]			
4		+ ST1_CTMAccessRightRequest			
5		[ PX_CTM_LR_EachTest ]			
6		+ ST1_CTMLocationRegistration			
7		[ NOT ( PX_CTM_LR_EachTest ) ]			
8		[NOT (PC_CTM_SR ) OR NOT (PX_CTM_SR_EachTest )]			
9		[ PX_CTM_LR_EachTest ]			
10		+ ST1_CTMLocationRegistration			
11		[ NOT ( PX_CTM_LR_EachTest ) ]			
12		[ PC_CFNrc_GSM ]			
13		[ PX_DG_LR_EachTest ]			
14		+ST1_GSMLocationRegistration			
15		[ NOT ( PX_DG_LR_EachTest ) ]			
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_Activate					
<b>Group</b> : PTC1_Steps/					
<b>Objective</b> : To activate IUT to CFNRc SS.					
<b>Default</b> : DEF1_CALL(0)					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	ST1	+ ST1_N02_CTMOutgoing	Mr ( AL_R1 ( tcv_flag_r, tcv_cref1 ))	(F)	
2		+ ltree_send_facility			
3		START T_AC			
4		L1? PDUr CANCEL T_AC			
5		GOTO ST1			
6		L1? PDUr CANCEL T_AC			
7		L1!PDUs			
8		GOTO ST1			
9		+ ltree_receive_fac			
10		+ PO1_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref1 )			
11		? TIMEOUT T_AC			
12		+ PO1_IC_OC_Stop ( tcv_flag_r, tcv_flag_s, tcv_cref1 )			
13		ltree_receive_fac			
14		[ PC_CFNRc_GSM ]			
15		L1?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref1, FIE_GSM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(1)
16		[ PC_CFNRc_CTM ]			(2)
17		L1?PDUr CANCEL T_AC	Mr ( FC_R1 ( tcv_flag_r, tcv_cref1, FIE_CTM_R01 ( EncapsulatedStimulus_Invoked_R ) ) )	(P)	(3)
18		ltree_send_facility			
19		( tcv_invokeld := TSO_RANDOM_INVOKE_ID ( ))			
20		[ PC_CFNRc_CTM ]			(1)
21	L1!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref1, FIE_CTM_S01 (EncapsulatedStimulus_Invoked_S ( tcv_invokeld, PX_CTM_KEYPAD_ACTIVATION ) ) ) )		(4)	
22	[ PC_CFNRc_GSM ]			(2)	

Continued on next page



Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L1!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref1, FIE_GSM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_GSM_KEYPAD_ACTIV ATION ) ) ) )		(4)
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_CTMAccessRightRequest <b>Group</b> : PTC1_Steps/ <b>Objective</b> : <b>Default</b> : DEF1_NCICS <b>Comments</b> : To proceed to the subscription registration with the IUT on PTC1.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1! PDU <sub>s</sub> START T <sub>_AC</sub>	Ms ( SU_S1 ( 0 , tcv_cref , FIE_CTM_S01 ( CTM_AccessRightRq_Invok e ( tcv_invokeld , PX_CTM_IPUI1))))		(1)
2		( tcv_flag_r := 1, tcv_flag_s := 0)			
3		+ ST1_U06_To_U10 ( tcv_flag_r , tcv_flag_s , tcv_cref )			
4		O1 ! DISPLAY START T <sub>_WAIT</sub>	Ds( "Initiate subscription registration for PTC1 – Access right request return result component")		
5		+LTS_KA_Invoke			
6		+LTS_TA_Invoke			
7		+LTS_NIC_Invoke			
8		+LTS_AR_ReturnResult			
9		+LTS_timeout			
		LTS_NIC_Invoke			
10		L1? PDU <sub>r</sub> ( tcv_invokeld2 := TSO_GET_INVOKEID()) CANCEL T <sub>_WAIT</sub>	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_Ciphering_Invoke ( PX_CTM_IPUI1 ))))		(4)
11		L1! PDU <sub>s</sub> START T <sub>_WAIT</sub>	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 ( CTM_Ciphering_RR ( tcv_invokeld2 ))))		
12		+LTS_AR_ReturnResult			
13		+LTS_timeout			
		LTS_TA_Invoke			
14		L1? PDU <sub>r</sub> ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_authtype := TSO_GET_CTM_authtype () ) CANCEL T <sub>_WAIT</sub>	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_TerminalAuth_Invoke ( PX_CTM_IPUI ))))		(5)
15		( tcv_ctm_xres1 := TSO_CTM_CALC_RES2 ( tcv_ctm_rs, tcv_rand , tcv_ctm_authtype ) )			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L1! PDUs START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 ( CTM_TerminalAuth_RR ( tcv_invokeld2 , tcv_ctm_xres1 ))))		
17		+LTS_NIC_Invoke			
18		+LTS_AR_ReturnResult			
19		+LTS_timeout			
		LTS_KA_Invoke			
20		L1? PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_alloctype := TSO_GET_CTM_alloctype () ) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_KeyAllocate_Invoke ( PX_CTM_IPUI1 ))))		(6)
21		( tcv_invokeld3 := TSO_RANDOM_INVOKE_ID () , tcv_ctm_xres1 := TSO_CTM_CALC_RES ( tcv_ctm_rs, tcv_rand , tcv_ctm_alloctype ) )			
22		L1! PDUs	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_NetworkAuth_Invoke ( tcv_invokeld3, PX_CTM_IPUI1 ))))		
23		L1! PDUs START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_KeyAllocate_RR ( tcv_invokeld2 , tcv_ctm_xres1 ))))		
24		L1? PDUr START T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_NetworkAuth_RR ( tcv_invokeld3 ))))		
25		+LTS_TA_Invoke			
26		+LTS_timeout			
		LTS_AR_ReturnResult			
27		L1? PDUr CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_AccessRightRq_RR ( tcv_invokeld ))))		(2)
28		+ PO1_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
		LTS_timeout			
29		?TIMEOUT T_WAIT			(3)
30		+ PO1_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
31		CANCEL		I	

Continued on next page

*Continued from previous page*

Test Step Dynamic Behaviour
-----------------------------

<p><b>Detailed Comments</b> : (1) sending of a SETUP message including a CTM_AccessRightsRequest invoke component. (2) receipt of a FACILITY message with a CTM_AccessRightsRequest return result component. (3) Nothing received, cancel all running timers if present , immediat verdict Inconclusive (4) receipt of a FACILITY message with a CTM_Ciphering invoke component. (5) receipt of a FACILITY message with a CTM_TerminalAuth invoke component. (6) receipt of a FACILITY message with a CTM_KeyAllocate invoke component.</p>
---

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_CTMLocationRegistration <b>Group</b> : PTC1_Steps/ <b>Objective</b> : <b>Default</b> : DEF1_NCICS <b>Comments</b> : To proceed to the location registration procedure with the IUT on PTC1.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1! PDU <sub>s</sub> START T <sub>AC</sub>	Ms ( SU <sub>S1</sub> ( 0 , tcv <sub>cref</sub> , FIE <sub>CTM_S01</sub> (CTM_LocationReg_Invoke ( tcv <sub>invokeld</sub> , PX <sub>CTM_IPUI1</sub> ))))		(1)
2		( tcv <sub>flag_r</sub> := 1, tcv <sub>flag_s</sub> := 0)			
3		+ ST1_U06_To_U10 ( tcv <sub>flag_r</sub> , tcv <sub>flag_s</sub> , tcv <sub>cref</sub> )			
4		O1 ! DISPLAY START T <sub>WAIT</sub>	Ds( "Initiate location registration on PTC1 – CTMLocationRegistration return result component")		
5		+LTS_KA_Invoke			
6		+LTS_TA_Invoke			
7		+LTS_LR_ReturnResult			
8		+LTS_timeout			
9		LTS_TA_Invoke L1? PDU <sub>r</sub> ( tcv <sub>invokeld2</sub> := TSO_GET_INVOKEID() , tcv <sub>ctm_rs</sub> := TSO_GET_CTM_RS () , tcv <sub>rand</sub> := TSO_GET_RAND () , tcv <sub>ctm_authtype</sub> := TSO_GET_CTM_authtype () ) CANCEL T <sub>WAIT</sub>	Mr ( FC <sub>R1</sub> ( tcv <sub>flag_r</sub> , tcv <sub>cref</sub> , FIE <sub>CTM_R01</sub> ( CTM_TerminalAuth_Invoke ( PX <sub>CTM_IPUI1</sub> ) )))		(4)
10		( tcv <sub>ctm_xres1</sub> := TSO_CTM_CALC_RES2 ( tcv <sub>ctm_rs</sub> , tcv <sub>rand</sub> , tcv <sub>ctm_authtype</sub> ) )			
11		L1! PDU <sub>s</sub> START T <sub>WAIT</sub>	Ms ( FC <sub>S1</sub> ( tcv <sub>flag_s</sub> , tcv <sub>cref</sub> , FIE <sub>CTM_S01</sub> (CTM_TerminalAuth_RR ( tcv <sub>invokeld2</sub> , tcv <sub>ctm_xres1</sub> ) )))		
12		+LTS_LR_ReturnResult			
13		+LTS_timeout			
14		LTS_KA_Invoke L1? PDU <sub>r</sub> ( tcv <sub>invokeld2</sub> := TSO_GET_INVOKEID() , tcv <sub>ctm_rs</sub> := TSO_GET_CTM_RS () , tcv <sub>rand</sub> := TSO_GET_RAND () , tcv <sub>ctm_alloctype</sub> := TSO_GET_CTM_alloctype () ) CANCEL T <sub>WAIT</sub>	Mr ( FC <sub>R1</sub> ( tcv <sub>flag_r</sub> , tcv <sub>cref</sub> , FIE <sub>CTM_R01</sub> (CTM_KeyAllocate_Invoke ( PX <sub>CTM_IPUI1</sub> ) )))		(5)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
15		( tcv_invokeld3 := TSO_RANDOM_INVOKE_ID(), tcv_ctm_xres1 := TSO_CTM_CALC_RES (			
16		tcv_ctm_rs, tcv_rand , tcv_ctm_alloctype ) )  L1! PDUs	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_NetworkAuth_Invoke (		
17		L1! PDUs START T_WAIT	tcv_invokeld3, PX_CTM_IPUI ))))  Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_KeyAllocate_RR (		
18		L1? PDUr START T_WAIT	tcv_invokeld2 , tcv_ctm_xres1 ))))  Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_NetworkAuth_RR (		
19		+LTS_TA_Invoke	tcv_invokeld3 ))))		
20		+LTS_timeout			
21		LTS_LR_ReturnResult L1? PDUr CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_LocationReg_RR (		(2)
22		+ PO1_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )  LTS_timeout	tcv_invokeld ))))		
23		?TIMEOUT T_WAIT			(3)
24		+ PO1_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
25		CANCEL		I	
<b>Detailed Comments</b> : (1) sending of a SETUP message including a CTMLocationregistration invoke component. (2) receipt of a FACILITY message with a CTMLocationregistration return result component. (3) Nothing received, cancel all running timers if present , immediat verdict Inconclusive (4) receipt of a FACILITY message with a CTM_TerminalAuth invoke component. (5) receipt of a FACILITY message with a CTM_KeyAllocate invoke component.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_GSM_EMB_PROCS (flag_r, flag_s: INTEGER ; cref: BIT7OR15) <b>Group</b> : PTC1_Steps/ <b>Objective</b> : Handling of embedded procedures following a Location Registration request <b>Default</b> : DEF1_NCICS <b>Comments</b> : EN 301 144-1 subclause 9.2.1.1.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_rand := TSO_GET_RAND(), tcv_cipherinfo := TSO_GET_CIPHERINFO())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( GSM_TerminalAuth_Invoke (OMIT))))		(1)
2		L1!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 ( GSM_TerminalAuth_RR ( tcv_invokeld2, TSO_GSM_CALC_RES(tcv_ rand))))))	(P)	(2)
3		[NOT tcv_rcv_termauth]		(P)	(3)
4		(tcv_rcv_termauth := TRUE)			
5		[tcv_rcv_termauth]		(I)	(4)
6		L1?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_idtype := TSO_GET_IDTYPE())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_IdentityReq_Invoke (OMIT))))	(P)	(5)
7		L1!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_IdentityReq_RR ( tcv_invokeld2, TSO_GSM_CALC_PID (tcv_idtype))))))	(P)	(6)
8		[NOT tcv_rcv_idreq]		(P)	(7)
9		(tcv_rcv_idreq := TRUE)			
10		[tcv_rcv_idreq]		(I)	(8)
11		L1?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_Ciphering_Invoke (OMIT))))	(P)	(9)
12		L1!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_Ciphering_RR ( tcv_invokeld2))))	(P)	(10)
13		[NOT tcv_rcv_niciph]		(P)	(11)
14		(tcv_rcv_niciph := TRUE)			
15		[tcv_rcv_niciph]		(I)	(12)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L1?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_tmsi := TSO_GET_TMSI())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_AssignIdentity_Invok e (OMIT))))	(P)	(13)
17		L1!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_AssignIdentity_RR ( tcv_invokeld2))))	(P)	(14)
18		[NOT tcv_rcv_tmpidass]		(P)	(15)
19		(tcv_rcv_tmpidass := TRUE)			
20		[tcv_rcv_tmpidass]		(I)	(16)
21		L1?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_tmsi := TSO_GET_TMSI())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LinkedAssignId_Invo ke)))	(P)	(17)
22		L1?PDUr (tcv_rcv_locreg := TRUE) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LocationRegistration_ RR ( tcv_invokeld))))	(P)	(18)
23		L1!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_LinkedAssignId_RR ( tcv_invokeld2))))	(P)	(19)
24		[NOT tcv_rcv_tmpidass]		(P)	(20)
25		(tcv_rcv_lidass := TRUE)			
26		[tcv_rcv_tmpidass]		(F)	(21)
27		L1?PDUr (tcv_rcv_locreg := TRUE) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LocationRegistration_ RR ( tcv_invokeld))))	(P)	(18)
28		?TIMEOUT T_WAIT (tcv_rcv_timeout := TRUE)		(I)	(22)

**Detailed Comments** : ST\_EMB\_PROCS:

- handles embedded procedures during Location Registration
  - one or more of the 4 allowed procedures may be applied by the IUT
  - if one of these procedures is used more than once an INCONC verdict is assigned, but the test is continued
  - if the Linked Assignment procedure is applied it is checked that it has been used instead of the Temporary Assignment procedure, otherwise a FAIL verdict is assigned
  - this step runs until the Location Registration has been accepted by the network, or until a timeout supervising the test occurs
- (1) receiving of a FACILITY message including a GSMTerminalAuthentication invoke component. PortableIdentity has to be omitted. Store InvokeID, Random number and CipherInfo.
- (2) sending a FACILITY message including a GSMTerminalAuthentication return result component
- (3) PASS if procedure was activated the first time
- (4) INCONC if procedure was activated more than once

Continued on next page



Continued from previous page

### Test Step Dynamic Behaviour

**Detailed Comments :** ...

- (5) receiving of a FACILITY message including a GSMIdentityRequest invoke component.  
PortableIdentity has to be omitted.
- (6) sending a FACILITY message including a GSMIdentityRequest return result component
- (7) PASS if procedure was activated the first time
- (8) INCONC if procedure was activated more than once
- (9) receiving of a FACILITY message including a GSMCIPHERING invoke component.  
PortableIdentity has to be omitted.
- (10) sending a FACILITY message including a GSMCIPHERING return result component
- (11) PASS if procedure was activated the first time
- (12) INCONC if procedure was activated more than once
- (13) receiving of a FACILITY message including a GSMAssignIdentity invoke component.  
PortableIdentity has to be omitted.
- (14) sending a FACILITY message including a GSMAssignIdentity return result component
- (15) PASS if procedure was activated the first time
- (16) INCONC if procedure was activated more than once
- (17) receiving of a FACILITY message including a GSMLinkedAssignIdentity invoke component.
- (18) receiving a FACILITY message including a GSMLocationRegistration return result component
- (19) sending a FACILITY message including a GSMLinkedAssignIdentity return result component
- (20) PASS if the TEMPORARY IDENTITY ASSIGNMENT procedure had been not activated
- (21) FAIL !! if the TEMPORARY IDENTITY ASSIGNMENT procedure had been activated
- (22) Timer T\_WAIT has elapsed

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_GSMLocationRegistration <b>Group</b> : PTC1_Steps/ <b>Objective</b> : Performing the GSM Location Registration procedure, accounting for possible embedded procedures on PTC1 <b>Default</b> : DEF1_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		O1! DISPLAY	Ds( "Initiate location registration for PTC1–GSMLocationRegistration return result component")		
2		( tcv_invokeId := TSO_RANDOM_INVOKE_ID ())			
3		L1! PDUs START T_AC	Ms ( SU_S1( 0, tcv_cref,FIE_GSM_S01( GSM_LocationRegistration_I nvoke ( tcv_invokeId, PX_GSM_PORTABLE_ID1)) ))		(1)
4		( tcv_flag_r := 1, tcv_flag_s := 0)			
5		+ ST1_U06_To_U10 ( tcv_flag_r, tcv_flag_s, tcv_cref)			
6		REPEAT ST1_GSM_EMB_PROCS (tcv_flag_r, tcv_flag_s, tcv_cref) UNTIL [tcv_rcv_locreg OR tcv_rcv_timeout]			(2)
7		+PO1_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref)			(3)
8		[ tcv_rcv_timeout ]		(I)	(4)
9		[ tcv_rcv_locreg ]		(P)	(5)
10		?TIMEOUT T_WAIT		I	
<b>Detailed Comments</b> : (1) sending a SETUP message including a GSMLocationRegistration invoke component. (2) handling of one or more embedded procedures and finally receiving a GSMLocationRegistration return result component (3) releasing the connection (4) if a timeout had occurred the test ends with an INCONCLUSIVE verdict (5) otherwise the location registration had been successful?					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_N02_CTMOutgoing					
<b>Group</b> : PTC1_Steps/					
<b>Objective</b> : To bring IUT to state N02 for a CTM outgoing call.					
<b>Default</b> : DEF1_CALL (0)					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ ltree_send_SETUP	Mr ( SUA_R1 ( tcv_flag_r, tcv_cref1 ))		
2		L1?PDUr CANCEL T_AC			
		ltree_send_SETUP			
3		( tcv_flag_r := 1, tcv_flag_s := 0, tcv_invokeld := TSO_RANDOM_INVOKE_ID() )			
4		[ PC_CFNRC_CTM ]			
5		L1! PDUs START T_AC	Ms ( SU_S2 ( 0, tcv_cref1, FIE_CTM_S01 ( CTM_OutgoingCallMMInfoRq_Invoke ( tcv_invokeld, PX_CTM_IPUI1 ))))		
6		[ PC_CFNRC_GSM ]	Ms ( SU_S2 ( 0, tcv_cref1, FIE_GSM_S01 ( GSM_OutgoingCallMMInfo_Invoke ( tcv_invokeld, PX_GSM_PORTABLE_ID1 ))))		
7		L1! PDUs START T_AC			
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_PTC1_Activate <b>Group</b> : PTC1_Steps/ <b>Objective</b> : To handle the PTC1 actions. <b>Default</b> : DEF1_CALL(1) <b>Comments</b> : To handle the PTC1 actions.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR1_N00_Init			
2		+ST1_Register			
3		+ST1_Activate			
4		CPA1?CP_M	N00_Ready		
5		CPA1!CP_M START T_WAIT	S_SU		
6		+ ST1_Receive_SU_Incoming			
7		+ ltree1_send_disconnect			
8		L1 ? PDUr CANCEL T_AC	Mr( RL_R2 ( tcv_flag_r, tcv_cref1, 31))		
9		L1 ! PDUs	Ms( RC_S1 ( tcv_flag_s, tcv_cref1))		
10		? TIMEOUT T_WAIT		(F)	
		ltree1_send_disconnect			
11		[PC_CFNRC_CTM]			1
12		L1 ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref1, 29, FIE_CTM_S01( CTM_IncomingCallMMInfo_R E ( tcv_invokeld, TSC_E_Congestion ))))		
13		[PC_CFNRC_GSM]			2
14		L1 ! PDUs START T_AC	Ms ( DI_S2 ( tcv_flag_s, tcv_cref1, 29, FIE_GSM_S01( GSM_IncomingCallMMInfo_ RE ( tcv_invokeld, TSC_E_Congestion ))))		
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_Receive_SU_Incoming <b>Group</b> : PTC1_Steps/ <b>Objective</b> : To receive SETUP message for a CTM incoming call depending on the CTM/DectToGsm and basic/primary option. <b>Default</b> : DEF1_CALL(1) <b>Comments</b> : To receive SETUP message depending on the CTM/DectToGsm and basic/primary option.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[PC_CFNRC_CTM]			
2		L1? SETUPr [NOT PX_BASIC] ( tcv_cref1 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI1 )))	(P)	
3		L1? SETUPr [PX_BASIC] ( tcv_cref1 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI1 )))	(P)	
4		[PC_CFNRC_GSM]			
5		L1? SETUPr [NOT PX_BASIC] ( tcv_cref1 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID1) )))	(P)	
6		L1? SETUPr [PX_BASIC] ( tcv_cref1 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID1) )))	(P)	
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_Register					
<b>Group</b> : PTC1_Steps/					
<b>Objective</b> : To register IUT to CFNRc SS.					
<b>Default</b> : DEF1_CALL (0)					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	ST1	+ ST1_N02_CTMOutgoing	Mr ( AL_R1 ( tcv_flag_r, tcv_cref1 ))	(F)	
2		+ ltree_send_facility			
3		START T_AC			
4		L1? PDUr CANCEL T_AC			
5		GOTO ST1			
6		L1? PDUr CANCEL T_AC			
7		L1!PDUs			
8		GOTO ST1			
9		+ ltree_receive_fac			
10		+ PO1_IC_OC ( tcv_flag_r, tcv_flag_s, tcv_cref1 )			
11		? TIMEOUT T_AC			
12		+ PO1_IC_OC_Stop ( tcv_flag_r, tcv_flag_s, tcv_cref1 )			
13		ltree_receive_fac			
14		[ PC_CFNRc_GSM ]			
15		L1?PDUr CANCEL T_AC			
16		[ PC_CFNRc_CTM ]			
17		L1?PDUr CANCEL T_AC			
18		ltree_send_facility			
19		( tcv_invokeld := TSO_RANDOM_INVOKE_ID ( ))			
20		[ PC_CFNRc_CTM ]			
	L1!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref1, FIE_CTM_S01 (EncapsulatedStimulus_Invoke_S ( tcv_invokeld, PX_CTM_KEYPAD_REGISTRATION1 ) ) ) )	(1)	(4)	
	[ PC_CFNRc_GSM ]			(2)	

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		L1!PDUs START T_AC	Ms (FC_S1 ( tcv_flag_s, tcv_cref1, FIE_GSM_S01 (EncapsulatedStimulus_Invo ke_S ( tcv_invokeld, PX_GSM_KEYPAD_REGIS TRATION1 ) ) ) )		(4)
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST1_U06_To_U10 (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : PTC1_Steps/ <b>Objective</b> : To complete the establishment of an outgoing NCICS call. Bring IUT from U06 to U10 on PTC1. <b>Default</b> : DEF1_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1? PDUR CANCEL T_AC	Mr ( CP_R1(flag_r, cref) )		(1)
2		O1!DISPLAY START T_WAIT	Ds("IUT send a CONNECT message on PTC1")		
3		L1? PDUR CANCEL T_WAIT	Mr ( CN_R1(flag_r, cref) )	(P)	(1)
4		L1!PDUs	Ms (CA_S1(flag_s, cref))		
5		?TIMEOUT T_WAIT		(I)	
6		+PO1_NCICS(flag_r, flag_s, cref)			
7		?TIMEOUT T_AC		(I)	
8		+PO1_NCICS(flag_r, flag_s, cref)			
<b>Detailed Comments</b> : (1) sends a CALL PROCEEDING message for a NCICS call. (2) sends a CONNECT message for a NCICS call. (3) receipt of a CONNECT ACKNOWLEDGE message.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO2_NCICS (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : PTC2_Steps/ <b>Objective</b> : To bring the IUT back to the Null call state U00. <b>Default</b> : DEF2_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L2!PDU <sub>s</sub> START T_AC	Ms( RL_S1 ( flag_s, cref, 16))		(1)
2		L2?PDU <sub>r</sub> CANCEL T_AC	Mr( RC_R1 ( flag_r, cref))		(2)
3		?TIMEOUT T_AC		(I)	(4)
4		L1?OTHERWISE		(I)	(3)
<b>Detailed Comments</b> : (1) A valid RELEASE message indicating the cause value 16 "Normal call clearing" is sent. (2) A RELEASE COMPLETE message is received from the IUT. (3) An invalid event occurred. (4) No response.					



Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PR2_N00 <b>Group</b> : PTC2_Steps/ <b>Objective</b> : Preamble to the Null call state N00 to be used with non-NCICS connections for PTC2. <b>Default</b> : DEF2_CALL(0) <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	+LT_INIT_VARIABLES			(1)
2		[PX_L2_INIT]			
3		L2!DL_REL_RQ START T_AC			(2)
4		L2?DL_REL_CO CANCEL T_AC		(P)	(3)
5		L2!DL_EST_RQ START T_AC			(4)
6		L2?DL_EST_CO CANCEL T_AC		(P)	(5)
7		+WAIT_RESTART			(6)
8		L2?DL_REL_IN START T_NOAC			(7)
9		L2?DL_EST_IN CANCEL T_AC , CANCEL T_NOAC		(P)	(8)
10		+WAIT_RESTART			(6)
11		?TIMEOUT T_NOAC			
12		L2!DL_EST_RQ			(4)
13		GOTO L1			
14		L2?OTHERWISE		I	
15		L2?DL_EST_IN CANCEL T_AC , START T_NOAC			(8)
16		L2?DL_EST_CO CANCEL T_NOAC		(P)	(9)
17		+WAIT_RESTART			
18		?TIMEOUT T_NOAC		I	no respons e
19		L2?OTHERWISE		I	(10)
20		?TIMEOUT T_AC		I	no respons e
21		L2?OTHERWISE		I	(10)
22		?TIMEOUT T_AC		I	no respons e
23		L2?OTHERWISE		I	(10)
24		[NOT PX_L2_INIT] LT_INIT_VARIABLES			
25		[PX_BASIC]			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
26		(tcv_cref2:='0000001'B, tcv_cref:='0000001'B, tcv_glob_cref:='0000000'B, tcv_bch_num2 := INT_TO_BIT( PX_CH_NUM, 2))			Basic access
27		[NOT PX_BASIC]			
28		(tcv_cref2:='0000000000000001'B, tcv_cref:='0000001'B, tcv_glob_cref:='000000000000000'B, tcv_bch_num2 := INT_TO_BIT( PX_CH_NUM, 7))			Primary rate access
		WAIT_RESTART			
29		[PX_WAIT_RESTART]			
30		START T_RESTART			
31		L2?RESTARTr CANCEL T_RESTART	RSr(RST_R1(0,tcv_glob_cref ,6))		Single interface
32		L2!PDUs	Ms(RSA_S1(1,tcv_glob_cref, 6))		
33		?TIMEOUT T_RESTART			
34		[NOT PX_WAIT_RESTART]			
<p><b>Detailed Comments</b> : The layer 2 of the IUT must have a TEI assigned value before the execution of this preamble.</p> <p>The procedure to assign the TEI value to the IUT is a matter for the test laboratory.</p> <p>(1) The local subtree INIT_VARIABLES is used to assign initial values to test case variables taking into account the used interface configuration.</p> <p>(2) Termination of the multiple frame operation is requested (A DISC frame is sent).</p> <p>(3) Termination of the multiple frame operation is confirmed (A UA or a DM frame is received).</p> <p>(4) Establishment of the multiple frame operation is requested (A SABME frame is sent).</p> <p>(5) Establishment of the multiple frame operation is confirmed (A UA frame is received).</p> <p>(6) The local subtree WAIT_RESTART is used to deal with the receipt of RESTART messages that may be sent by the IUT after the re-establishment of the multiple frame operation.</p> <p>(7) An unsuccessful establishment attempt is reported (A DM frame is received).</p> <p>(8) Establishment of the multiple frame operation is indicated (A SABME frame is received and a UA frame is sent).</p> <p>(9) Establishment of the multiple frame operation (requested in line 4) is confirmed (A UA frame is received).</p> <p>(10) Any other event occurred.</p>					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PR2_N00_Init					
<b>Group</b> : PTC2_Steps/					
<b>Objective</b> : Preamble to the Null call state N00 with subscription registration (if supported) and location registration.					
<b>Default</b> : DEF2_NCICS					
<b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR2_N00			(1)
2		[ PC_CFNrc_CTM ]			
3		[PC_CTM_SR AND PX_CTM_SR_EachTest]			
4		+ ST2_CTMAccessRightRequest			
5		[ PX_CTM_LR_EachTest ]			
6		+ ST2_CTMLocationRegistration			
7		[ NOT ( PX_CTM_LR_EachTest ) ]			
8		[NOT (PC_CTM_SR ) OR NOT (PX_CTM_SR_EachTest )]			
9		[ PX_CTM_LR_EachTest ]			
10		+ ST2_CTMLocationRegistration			
11		[ NOT ( PX_CTM_LR_EachTest ) ]			
12		[ PC_CFNrc_GSM ]			
13		[ PX_DG_LR_EachTest ]			
14		+ST2_GSMLocationRegistration			
15		[ NOT ( PX_DG_LR_EachTest ) ]			
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_CTMAccessRightRequest <b>Group</b> : PTC2_Steps/ <b>Objective</b> : <b>Default</b> : DEF2_NCICS <b>Comments</b> : To proceed to the subscription registration with the IUT on PTC2.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L2! PDU <sub>s</sub> START T <sub>_AC</sub>	Ms ( SU_S1 ( 0 , tcv_cref , FIE_CTM_S01 ( CTM_AccessRightRq_Invok e ( tcv_invokeld , PX_CTM_IPUI1))))		(1)
2		( tcv_flag_r := 1, tcv_flag_s := 0)			
3		+ ST2_U06_To_U10 ( tcv_flag_r , tcv_flag_s , tcv_cref )			
4		O2 ! DISPLAY START T <sub>_WAIT</sub>	Ds( "Initiate subscription registration for PTC2 – Access right request return result component")		
5		+LTS_KA_Invoke			
6		+LTS_TA_Invoke			
7		+LTS_NIC_Invoke			
8		+LTS_AR_ReturnResult			
9		+LTS_timeout			
		LTS_NIC_Invoke			
10		L2? PDU <sub>r</sub> ( tcv_invokeld2 := TSO_GET_INVOKEID()) CANCEL T <sub>_WAIT</sub>	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_Ciphering_Invoke ( PX_CTM_IPUI1 ))))		(4)
11		L2! PDU <sub>s</sub> START T <sub>_WAIT</sub>	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 ( CTM_Ciphering_RR ( tcv_invokeld2 ))))		
12		+LTS_AR_ReturnResult			
13		+LTS_timeout			
		LTS_TA_Invoke			
14		L2? PDU <sub>r</sub> ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_authtype := TSO_GET_CTM_authtype () ) CANCEL T <sub>_WAIT</sub>	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_TerminalAuth_Invoke ( PX_CTM_IPUI ))))		(5)
15		( tcv_ctm_xres1 := TSO_CTM_CALC_RES2 ( tcv_ctm_rs, tcv_rand , tcv_ctm_authtype ) )			

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L2! PDU <sub>s</sub> START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 ( CTM_TerminalAuth_RR ( tcv_invokeld2 , tcv_ctm_xres1 ))))		
17		+LTS_NIC_Invoke			
18		+LTS_AR_ReturnResult			
19		+LTS_timeout			
		LTS_KA_Invoke			
20		L2? PDU <sub>r</sub> ( tcv_invokeld2 := TSO_GET_INVOKEID() , tcv_ctm_rs := TSO_GET_CTM_RS () , tcv_rand := TSO_GET_RAND () , tcv_ctm_alloctype := TSO_GET_CTM_alloctype () ) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_KeyAllocate_Invoke ( PX_CTM_IPUI1 ))))		(6)
21		( tcv_invokeld3 := TSO_RANDOM_INVOKE_ID () , tcv_ctm_xres1 := TSO_CTM_CALC_RES ( tcv_ctm_rs, tcv_rand , tcv_ctm_alloctype ) )			
22		L2! PDU <sub>s</sub>	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_NetworkAuth_Invoke ( tcv_invokeld3, PX_CTM_IPUI1 ))))		
23		L2! PDU <sub>s</sub> START T_WAIT	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_KeyAllocate_RR ( tcv_invokeld2 , tcv_ctm_xres1 ))))		
24		L2? PDU <sub>r</sub> START T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_NetworkAuth_RR ( tcv_invokeld3 ))))		
25		+LTS_TA_Invoke			
26		+LTS_timeout			
		LTS_AR_ReturnResult			
27		L2? PDU <sub>r</sub> CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 ( CTM_AccessRightRq_RR ( tcv_invokeld ))))		(2)
28		+ PO2_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
		LTS_timeout			
29		?TIMEOUT T_WAIT			(3)
30		+ PO2_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
31		CANCEL		I	

Continued on next page

*Continued from previous page*

Test Step Dynamic Behaviour
-----------------------------

<p><b>Detailed Comments</b> : (1) sending of a SETUP message including a CTM_AccessRightsRequest invoke component. (2) receipt of a FACILITY message with a CTM_AccessRightsRequest return result component. (3) Nothing received, cancel all running timers if present , immediat verdict Inconclusive (4) receipt of a FACILITY message with a CTM_Ciphering invoke component. (5) receipt of a FACILITY message with a CTM_TerminalAuth invoke component. (6) receipt of a FACILITY message with a CTM_KeyAllocate invoke component.</p>
---

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_CTMLocationRegistration <b>Group</b> : PTC2_Steps/ <b>Objective</b> : <b>Default</b> : DEF2_NCICS <b>Comments</b> : To proceed to the location registration procedure with the IUT on PTC2.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L2! PDU <sub>s</sub> START T <sub>AC</sub>	Ms ( SU <sub>S1</sub> ( 0 , tcv <sub>cref</sub> , FIE <sub>CTM_S01</sub> (CTM_LocationReg_Invoke ( tcv <sub>invokeld</sub> , PX <sub>CTM_IPUI1</sub> ))))		(1)
2		( tcv <sub>flag_r</sub> := 1, tcv <sub>flag_s</sub> := 0)			
3		+ ST2_U06_To_U10 ( tcv <sub>flag_r</sub> , tcv <sub>flag_s</sub> , tcv <sub>cref</sub> )			
4		O2 ! DISPLAY START T <sub>WAIT</sub>	Ds( "Initiate location registration on PTC1 – CTMLocationRegistration return result component")		
5		+LTS_KA_Invoke			
6		+LTS_TA_Invoke			
7		+LTS_LR_ReturnResult			
8		+LTS_timeout			
		LTS_TA_Invoke			
9		L2? PDU <sub>r</sub> ( tcv <sub>invokeld2</sub> := TSO_GET_INVOKEID() , tcv <sub>ctm_rs</sub> := TSO_GET_CTM_RS () , tcv <sub>rand</sub> := TSO_GET_RAND () , tcv <sub>ctm_auth</sub> := TSO_GET_CTM_authtype () ) CANCEL T <sub>WAIT</sub>	Mr ( FC <sub>R1</sub> ( tcv <sub>flag_r</sub> , tcv <sub>cref</sub> , FIE <sub>CTM_R01</sub> ( CTM_TerminalAuth_Invoke ( PX <sub>CTM_IPUI1</sub> ) )))		(4)
10		( tcv <sub>ctm_xres1</sub> := TSO_CTM_CALC_RES2 ( tcv <sub>ctm_rs</sub> , tcv <sub>rand</sub> , tcv <sub>ctm_auth</sub> ) )			
11		L2! PDU <sub>s</sub> START T <sub>WAIT</sub>	Ms ( FC <sub>S1</sub> ( tcv <sub>flag_s</sub> , tcv <sub>cref</sub> , FIE <sub>CTM_S01</sub> (CTM_TerminalAuth_RR ( tcv <sub>invokeld2</sub> , tcv <sub>ctm_xres1</sub> ) )))		
12		+LTS_LR_ReturnResult			
13		+LTS_timeout			
		LTS_KA_Invoke			
14		L2? PDU <sub>r</sub> ( tcv <sub>invokeld2</sub> := TSO_GET_INVOKEID() , tcv <sub>ctm_rs</sub> := TSO_GET_CTM_RS () , tcv <sub>rand</sub> := TSO_GET_RAND () , tcv <sub>ctm_alloct</sub> := TSO_GET_CTM_alloctype () ) CANCEL T <sub>WAIT</sub>	Mr ( FC <sub>R1</sub> ( tcv <sub>flag_r</sub> , tcv <sub>cref</sub> , FIE <sub>CTM_R01</sub> (CTM_KeyAllocate_Invoke ( PX <sub>CTM_IPUI1</sub> ) )))		(5)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
15		( tcv_invokeld3 := TSO_RANDOM_INVOKE_ID(), tcv_ctm_xres1 := TSO_CTM_CALC_RES (			
16		tcv_ctm_rs, tcv_rand , tcv_ctm_alloctype ) )  L2! PDUs	Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_NetworkAuth_Invoke (		
17		L2! PDUs START T_WAIT	tcv_invokeld3, PX_CTM_IPUI ))))  Ms ( FC_S1 ( tcv_flag_s, tcv_cref, FIE_CTM_S01 (CTM_KeyAllocate_RR (		
18		L2? PDUr START T_WAIT	tcv_invokeld2 , tcv_ctm_xres1 ))))  Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_NetworkAuth_RR (		
19		+LTS_TA_Invoke	tcv_invokeld3 ))))		
20		+LTS_timeout			
21		LTS_LR_ReturnResult L2? PDUr CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r , tcv_cref , FIE_CTM_R01 (CTM_LocationReg_RR (		(2)
22		+ PO2_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )  LTS_timeout	tcv_invokeld ))))		
23		?TIMEOUT T_WAIT			(3)
24		+ PO2_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref )			
25		CANCEL		I	
<b>Detailed Comments</b> : (1) sending of a SETUP message including a CTMLocationregistration invoke component. (2) receipt of a FACILITY message with a CTMLocationregistration return result component. (3) Nothing received, cancel all running timers if present , immediat verdict Inconclusive (4) receipt of a FACILITY message with a CTM_TerminalAuth invoke component. (5) receipt of a FACILITY message with a CTM_KeyAllocate invoke component.					



Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_GSM_EMB_PROCS (flag_r, flag_s: INTEGER ; cref: BIT7OR15) <b>Group</b> : PTC2_Steps/ <b>Objective</b> : Handling of embedded procedures following a Location Registration request <b>Default</b> : DEF2_NCICS <b>Comments</b> : EN 301 144-1 subclause 9.2.1.1.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L2?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_rand := TSO_GET_RAND(), tcv_cipherinfo := TSO_GET_CIPHERINFO())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 ( GSM_TerminalAuth_Invoke (OMIT))))		(1)
2		L2!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 ( GSM_TerminalAuth_RR ( tcv_invokeld2, TSO_GSM_CALC_RES(tcv_ rand))))))	(P)	(2)
3		[NOT tcv_rcv_termauth]		(P)	(3)
4		(tcv_rcv_termauth := TRUE)			
5		[tcv_rcv_termauth]		(I)	(4)
6		L2?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_idtype := TSO_GET_IDTYPE())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_IdentityReq_Invoke (OMIT))))	(P)	(5)
7		L2!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_IdentityReq_RR ( tcv_invokeld2, TSO_GSM_CALC_PID (tcv_idtype))))))	(P)	(6)
8		[NOT tcv_rcv_idreq]		(P)	(7)
9		(tcv_rcv_idreq := TRUE)			
10		[tcv_rcv_idreq]		(I)	(8)
11		L2?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_Ciphering_Invoke (OMIT))))	(P)	(9)
12		L2!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_Ciphering_RR ( tcv_invokeld2))))	(P)	(10)
13		[NOT tcv_rcv_niciph]		(P)	(11)
14		(tcv_rcv_niciph := TRUE)			
15		[tcv_rcv_niciph]		(I)	(12)

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L2?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_tmsi := TSO_GET_TMSI())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_AssignIdentity_Invok e (OMIT))))	(P)	(13)
17		L2!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_AssignIdentity_RR ( tcv_invokeld2))))	(P)	(14)
18		[NOT tcv_rcv_tmpidass]		(P)	(15)
19		(tcv_rcv_tmpidass := TRUE)			
20		[tcv_rcv_tmpidass]		(I)	(16)
21		L2?PDUr ( tcv_invokeld2 := TSO_GET_INVOKEID(), tcv_tmsi := TSO_GET_TMSI())	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LinkedAssignId_Invo ke)))	(P)	(17)
22		L2?PDUr (tcv_rcv_locreg := TRUE) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LocationRegistration_ RR ( tcv_invokeld))))	(P)	(18)
23		L2!PDUs	Ms ( FC_S1 ( tcv_flag_r, tcv_cref, FIE_GSM_S01 (GSM_LinkedAssignId_RR ( tcv_invokeld2))))	(P)	(19)
24		[NOT tcv_rcv_tmpidass]		(P)	(20)
25		(tcv_rcv_lidass := TRUE)			
26		[tcv_rcv_tmpidass]		(F)	(21)
27		L2?PDUr (tcv_rcv_locreg := TRUE) CANCEL T_WAIT	Mr ( FC_R1 ( tcv_flag_r, tcv_cref, FIE_GSM_R01 (GSM_LocationRegistration_ RR ( tcv_invokeld))))	(P)	(18)
28		?TIMEOUT T_WAIT (tcv_rcv_timeout := TRUE)		(I)	(22)

**Detailed Comments** : ST\_EMB\_PROCS:

- handles embedded procedures during Location Registration
  - one or more of the 4 allowed procedures may be applied by the IUT
  - if one of these procedures is used more than once an INCONC verdict is assigned, but the test is continued
  - if the Linked Assignment procedure is applied it is checked that it has been used instead of the Temporary Assignment procedure, otherwise a FAIL verdict is assigned
  - this step runs until the Location Registration has been accepted by the network, or until a timeout supervising the test occurs
- (1) receiving of a FACILITY message including a GSMTerminalAuthentication invoke component. PortableIdentity has to be omitted. Store InvokeID, Random number and CipherInfo.
- (2) sending a FACILITY message including a GSMTerminalAuthentication return result component
- (3) PASS if procedure was activated the first time
- (4) INCONC if procedure was activated more than once

Continued on next page

Continued from previous page

### Test Step Dynamic Behaviour

**Detailed Comments :** ...

- (5) receiving of a FACILITY message including a GSMIdentityRequest invoke component.  
PortableIdentity has to be omitted.
- (6) sending a FACILITY message including a GSMIdentityRequest return result component
- (7) PASS if procedure was activated the first time
- (8) INCONC if procedure was activated more than once
- (9) receiving of a FACILITY message including a GSMCIPHERING invoke component.  
PortableIdentity has to be omitted.
- (10) sending a FACILITY message including a GSMCIPHERING return result component
- (11) PASS if procedure was activated the first time
- (12) INCONC if procedure was activated more than once
- (13) receiving of a FACILITY message including a GSMAssignIdentity invoke component.  
PortableIdentity has to be omitted.
- (14) sending a FACILITY message including a GSMAssignIdentity return result component
- (15) PASS if procedure was activated the first time
- (16) INCONC if procedure was activated more than once
- (17) receiving of a FACILITY message including a GSMLinkedAssignIdentity invoke component.
- (18) receiving a FACILITY message including a GSMLocationRegistration return result component
- (19) sending a FACILITY message including a GSMLinkedAssignIdentity return result component
- (20) PASS if the TEMPORARY IDENTITY ASSIGNMENT procedure had been not activated
- (21) FAIL !! if the TEMPORARY IDENTITY ASSIGNMENT procedure had been activated
- (22) Timer T\_WAIT has elapsed

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_GSMLocationRegistration <b>Group</b> : PTC2_Steps/ <b>Objective</b> : Performing the GSM Location Registration procedure, accounting for possible embedded procedures on PTC2 <b>Default</b> : DEF2_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		O2! DISPLAY	Ds( "Initiate location registration for PTC1–GSMLocationRegistration return result component")		
2		( tcv_invokeId := TSO_RANDOM_INVOKE_ID ())			
3		L2! PDUs START T_AC	Ms ( SU_S1( 0, tcv_cref,FIE_GSM_S01( GSM_LocationRegistration_I nvoke ( tcv_invokeId, PX_GSM_PORTABLE_ID1)) ))		(1)
4		( tcv_flag_r := 1, tcv_flag_s := 0)			
5		+ ST2_U06_To_U10 ( tcv_flag_r, tcv_flag_s, tcv_cref)			
6		REPEAT ST2_GSM_EMB_PROCS (tcv_flag_r, tcv_flag_s, tcv_cref) UNTIL [tcv_rcv_locreg OR tcv_rcv_timeout]			(2)
7		+PO2_NCICS ( tcv_flag_r, tcv_flag_s, tcv_cref)			(3)
8		[ tcv_rcv_timeout ]		(I)	(4)
9		[ tcv_rcv_locreg ]		(P)	(5)
10		?TIMEOUT T_WAIT		I	
<b>Detailed Comments</b> : (1) sending a SETUP message including a GSMLocationRegistration invoke component. (2) handling of one or more embedded procedures and finally receiving a GSMLocationRegistration return result component (3) releasing the connection (4) if a timeout had occurred the test ends with an INCONCLUSIVE verdict (5) otherwise the location registration had been successful?					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_PTC_IN <b>Group</b> : PTC2_Steps/ <b>Objective</b> : To handle the PTC2 actions for receiving an incoming call. <b>Default</b> : DEF2_CALL(1) <b>Comments</b> : To handle the PTC1 actions.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR2_N00_Init			
2		START T_WAIT			
3		CPA2?CP_M	R_SU_Rgnb		
4		+ ltree_Receive_SU_Rgnb			
5		CPA2!CP_M	R_SU		
6		L2!PDU_s	Ms(RC_S1(tcv_flag_s, tcv_cref2))		
7		CPA2?CP_M	R_SU_RgnbR		
8		+ ltree_Receive_SU_RgnbR			
9		CPA2!CP_M	R_SU		
10		L2!PDU_s	Ms(RC_S1(tcv_flag_s, tcv_cref2))		
		ltree_Receive_SU_Rgnb			
11		[PC_CFNRC_CTM]			
12		L2? SETUPr [NOT PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeId := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R3 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI2 ))))	(P)	
13		L2? SETUPr [PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeId := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R3 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI2 ))))	(P)	
14		[PC_CFNRC_GSM]			
15		L2? SETUPr [NOT PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeId := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R3 ( FIE_GSM_R01 (GSM_IncomingCallMMInfoR Invoke (PX_GSM_PORTABLE_ID2 )))	(P)	

Continued on next page

Continued from previous page

Test Step Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
16		L2? SETUPr [PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT  ltree_Receive_SU_RgnbR	Sr ( SU_R3 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID2 )))	(P)	
17		[PC_CFNRC_CTM]			
18		L2? SETUPr [NOT PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R4 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI2 )))	(P)	
19		L2? SETUPr [PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R4 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI2 )))	(P)	
20		[PC_CFNRC_GSM]			
21		L2? SETUPr [NOT PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R4 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID2 )))	(P)	
22		L2? SETUPr [PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R4 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID2 )))	(P)	
Detailed Comments :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_PTC2_IN <b>Group</b> : PTC2_Steps/ <b>Objective</b> : To handle the PTC2 actions for receiving an incoming call. <b>Default</b> : DEF2_CALL(1) <b>Comments</b> : To handle the PTC1 actions.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR2_N00_Init			
2		START T_WAIT			
3		+ ST2_Receive_SU_Incoming			
4		CPA2!CP_M START T_WAIT	R_SU		
5		L2?PDUr CANCEL T_WAIT	Mr(RL_R1 (tcv_flag_r, tcv_cref2))		
6		L2!PDU	Ms(RC_S1(tcv_flag_s, tcv_cref2))		
7		? TIMEOUT T_WAIT		(F)	
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_PTC4_IN <b>Group</b> : PTC2_Steps/ <b>Objective</b> : To handle the PTC2 actions for receiving an incoming call. <b>Default</b> : DEF2_CALL(1) <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ PR2_N00_Init			
2		START T_WAIT			
3		+ ST2_Receive_SU_Incoming			
4		CPA2!CP_M	R_SU		
5		L2!PDU	Ms ( CP_S1 ( tcv_flag_s, tcv_cref2, tcv_bch_num2 ))		
6		L2!PDU	Ms ( AL_S1 ( tcv_flag_s, tcv_cref2 ))		
7		L2!PDU START T_WAIT	Ms ( CN_S1 ( tcv_flag_s, tcv_cref2 ))		
8		L2?PDUr CANCEL T_WAIT	Mr (RL_R1 ( tcv_flag_r, tcv_cref2 ))		
9		L2!PDU	Ms ( RC_S1 ( tcv_flag_s, tcv_cref2 ))		
10		? TIMEOUT T_WAIT		(F)	
11		? TIMEOUT T_WAIT		(F)	
<b>Detailed Comments</b> :					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_Receive_SU_Incoming <b>Group</b> : PTC2_Steps/ <b>Objective</b> : To receive SETUP message for a CTM incoming call depending on the CTM/DectToGsm and basic/primary option. <b>Default</b> : DEF1_CALL(1) <b>Comments</b> : To receive SETUP message depending on the CTM/DectToGsm and basic/primary option.					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		[PC_CFNRC_CTM]			
2		L2? SETUPr [NOT PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI2 )))	(P)	
3		L2? SETUPr [PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_CTM_R01 (CTM_IncomingCallMMInfoR q_Invoke ( PX_CTM_IPUI2 )))	(P)	
4		[PC_CFNRC_GSM]			
5		L2? SETUPr [NOT PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e5_ch2) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID2) )))	(P)	
6		L2? SETUPr [PX_BASIC] ( tcv_cref2 := SETUPr.mun.cr.cr_r, tcv_invokeld := TSO_GET_INVOKEID (), tcv_flag_r := BIT_TO_INT( SETUPr.mun.cr.cr_f), tcv_flag_s := ( (tcv_flag_r + 1) MOD 2), tcv_bch_num := SETUPr.mun.chi.chi_e3_cs) CANCEL T_WAIT	Sr ( SU_R1 ( FIE_GSM_R01 (GSM_IncomingCallMMInfo_ Invoke (PX_GSM_PORTABLE_ID2) )))	(P)	
<b>Detailed Comments</b> :					



Test Step Dynamic Behaviour					
<b>Test Step Name</b> : ST2_U06_To_U10 (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : PTC2_Steps/ <b>Objective</b> : To complete the establishment of an outgoing NCICS call. Bring IUT from U06 to U10 on PTC2. <b>Default</b> : DEF2_NCICS <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L2? PDUr CANCEL T_AC	Mr ( CP_R1(flag_r, cref) )		(1)
2		O2!DISPLAY START T_WAIT	Ds("IUT send a CONNECT message on PTC1")		
3		L2? PDUr CANCEL T_WAIT	Mr ( CN_R1(flag_r, cref) )	(P)	(1)
4		L2!PDUs	Ms (CA_S1(flag_s, cref))		
5		?TIMEOUT T_WAIT		(I)	
6		+PO2_NCICS(flag_r, flag_s, cref)			
7		?TIMEOUT T_AC		(I)	
8		+PO2_NCICS(flag_r, flag_s, cref)			
<b>Detailed Comments</b> : (1) sends a CALL PROCEEDING message for a NCICS call. (2) sends a CONNECT message for a NCICS call. (3) receipt of a CONNECT ACKNOWLEDGE message.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO_IC_OC (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : Postambles/ <b>Objective</b> : To bring the IUT back to the Null call state U00. <b>Default</b> : DEF_CALL <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L! PDUs START T_AC	Ms( DI_S1 ( flag_s, cref, 16))		(1)
2		L? PDUr CANCEL T_AC	Mr( RL_R1 ( flag_r, cref))		(2)
3		L! PDUs	Ms( RC_S1 ( flag_s, cref))		(3)
4		? TIMEOUT T_AC		(I)	(4)
5		L? OTHERWISE		(I)	(5)
<b>Detailed Comments</b> : (1) A valid DISCONNECT message indicating the cause value 16 "Normal call clearing" is sent. (2) A valid RELEASE message is received from the IUT. (3) A RELEASE COMPLETE message is sent to the IUT. (4) No response. (5) An invalid event occurred.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO_IC_OC_Stop (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : Postambles/ <b>Objective</b> : To bring the IUT back to the Null call state U00 and stop the test. <b>Default</b> : DEF_CALL <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L! PDUs START T_AC	Ms( DI_S1 ( flag_s, cref, 16))		(1)
2		L? PDUr CANCEL T_AC	Mr( RL_R1 ( flag_r, cref))		(2)
3		L! PDUs	Ms( RC_S1 ( flag_s, cref))	R	(3)
4		? TIMEOUT T_AC		R	(4)
5		L? OTHERWISE		R	(5)
<b>Detailed Comments</b> : (1) A valid DISCONNECT message indicating the cause value 16 "Normal call clearing" is sent. (2) A valid RELEASE message is received from the IUT. (3) A RELEASE COMPLETE message is sent to the IUT. (4) No response. (5) An invalid event occurred.					

Test Step Dynamic Behaviour					
<b>Test Step Name</b> : PO_NCICS (flag_r, flag_s:INTEGER; cref:BIT7OR15) <b>Group</b> : Postambles/ <b>Objective</b> : To bring the IUT back to the Null call state U00. <b>Default</b> : <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L!PDUs START T_AC	Ms( RL_S1 ( flag_s, cref, 16))		(1)
2		L?PDUr CANCEL T_AC	Mr( RC_R1 ( flag_r, cref))		(2)
3		?TIMEOUT T_AC		(I)	(4)
4		L?OTHERWISE		(I)	(3)
<b>Detailed Comments</b> : (1) A valid RELEASE message indicating the cause value 16 "Normal call clearing" is sent. (2) A RELEASE COMPLETE message is received from the IUT. (3) An invalid event occurred. (4) No response.					

Default Dynamic Behaviour					
<b>Default Name</b> : DEF_CALL <b>Group</b> : <b>Objective</b> : Default subtree for outgoing call and incoming call test cases. <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L? DL_REL_IN		(I)	DL failure
2		+ ltree_PTC			
3		CANCEL		R	
4		L? DL_EST_IN		(I)	DL reset
5		+RELEASE_CALL			(1)
6		L? PDUR	Mr ( IN_R1 ( tcv_flag_r, tcv_cref ))		ignore
7		RETURN			(2)
8		L? PDUR	Mr ( NO_R1 ( tcv_flag_r, tcv_cref ))		ignore
9		RETURN			(2)
10		L? PDUR	Mr ( PR_R1 ( tcv_flag_r, tcv_cref ))		ignore
11		RETURN			(2)
12		L? PDUR	Mr ( SQ_R1 ( tcv_flag_r, tcv_cref ))		ignore
13		RETURN			(2)
14		L? PDUR	Mr ( GFP_R1 ( tcv_flag_r, tcv_cref ))		ignore
15		RETURN			(2)
16		L? OTHERWISE		(F)	(3)
17		+RELEASE_CALL			(1)
		RELEASE_CALL			
18		L! PDUs START T_AC	Ms ( RL_S1 ( tcv_flag_s, tcv_cref, 16 ))		(4)
19	L1	L? PDUR CANCEL T_AC	Mr ( RC_R1 ( tcv_flag_r, tcv_cref ))		(5)
20		+ ltree_PTC			
21		? TIMEOUT T_AC			(6)
22		+ ltree_PTC			
23		CANCEL		R	
24		L? PDUR	Mr ( IN_R1 ( tcv_flag_r, tcv_cref ))		ignore
25		GOTO L1			
26		L? PDUR	Mr ( NO_R1 ( tcv_flag_r, tcv_cref ))		ignore
27		GOTO L1			

Continued on next page

Continued from previous page

Default Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
28		L? PDUr	Mr ( SQ_R1 ( tcv_flag_r, tcv_cref ))	R	ignore
29		GOTO L1			
30		L? PDUr	Mr ( GFP_R1 ( tcv_flag_r, tcv_cref ))		ignore
31		GOTO L1			
32		L? OTHERWISE			(3)
33		+ ltree_PTC			
34		CANCEL			
		ltree_PTC			
35		[tcv_config1=TRUE]			
36		CPA1!CP_M	STOP_PTC		
37		CPA2!CP_M	STOP_PTC		
38		[TRUE]			
<b>Detailed Comments :</b> (1) Subtree to release the call. (2) Return to the test body. (3) An invalid event occurred. (4) A valid RELEASE message with cause #16 is sent. (5) A RELEASE COMPLETE message is received from the IUT. (6) No response.					

Default Dynamic Behaviour					
<b>Default Name</b> : DEF_NCICS <b>Group</b> : <b>Objective</b> : Default behaviour for NCICS calls. <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	L? DL_REL_IN	Mr ( NO_R1 ( tcv_flag_r, tcv_cref))  Mr ( SQ_R1 ( tcv_flag_r, tcv_cref))  Mr ( GFP_R1 ( tcv_flag_r, tcv_cref))  Ms ( RL_S1 ( tcv_flag_s, tcv_cref, 16))  Mr ( RC_R1 ( tcv_flag_r, tcv_cref))	(I)	DL failure
2		+ ltree_PTC		R	
3		CANCEL			
4		L? DL_EST_IN		(I)	DL reset
5		+RELEASE_CALL			(1)
6		L? PDUr			ignore
7		RETURN			(2)
8		L? PDUr			ignore
9		RETURN			(2)
10		L? PDUr			ignore
11		RETURN			(2)
12		L? OTHERWISE		(F)	(3)
13		+RELEASE_CALL			(1)
		RELEASE_CALL			
14		L! PDUs START T_AC			(4)
15		L? PDUr CANCEL T_AC			(5)
16		+ ltree_PTC			
17		? TIMEOUT T_AC			(6)
18		+ ltree_PTC			
19		CANCEL		R	
20		L? PDUr			ignore
21		GOTO L1			
22		L? PDUr			ignore
23		GOTO L1			
24		L? PDUr			ignore
25		GOTO L1			
26		L? OTHERWISE			(3)
27		+ ltree_PTC			

Continued on next page

Continued from previous page

Default Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
28		CANCEL	STOP_PTC	R	
		ltree_PTC			
29		[tcv_config1=TRUE]			
30		CPA1!CP_M			
31		[TRUE]			
<b>Detailed Comments</b> : (1) Subtree to release the call. (2) Return to the test body. (3) An invalid event occurred. (4) A valid RELEASE message with cause #16 is sent. (5) A RELEASE COMPLETE message is received from the IUT. (6) No response.					

Default Dynamic Behaviour					
<b>Default Name</b> : DEF1_NCICS <b>Group</b> : <b>Objective</b> : Default behaviour for NCICS calls. <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L1? DL_REL_IN		(I)	DL failure
2		CANCEL		R	
3		L1? DL_EST_IN		(I)	DL reset
4		+RELEASE_CALL			(1)
5		L1? PDUr	Mr ( NO_R1 ( tcv_flag_r, tcv_cref))		ignore
6		RETURN			(2)
7		L1? PDUr	Mr ( SQ_R1 ( tcv_flag_r, tcv_cref))		ignore
8		RETURN			(2)
9		L1? PDUr	Mr ( GFP_R1 ( tcv_flag_r, tcv_cref))		ignore
10		RETURN			(2)
11		L1? OTHERWISE		(F)	(3)
12		+RELEASE_CALL			(1)
		RELEASE_CALL			
13		L1! PDUs START T_AC	Ms ( RL_S1 ( tcv_flag_s, tcv_cref, 16))		(4)
14	L1	L1? PDUr CANCEL T_AC	Mr ( RC_R1 ( tcv_flag_r, tcv_cref))		(5)
15		? TIMEOUT T_AC			(6)
16		CANCEL		R	
17		L1? PDUr	Mr ( NO_R1 ( tcv_flag_r, tcv_cref))		ignore
18		GOTO L1			
19		L1? PDUr	Mr ( SQ_R1 ( tcv_flag_r, tcv_cref))		ignore
20		GOTO L1			
21		L1? PDUr	Mr ( GFP_R1 ( tcv_flag_r, tcv_cref))		ignore
22		GOTO L1			
23		L1? OTHERWISE			(3)
24		CANCEL		R	
<b>Detailed Comments</b> : (1) Subtree to release the call. (2) Return to the test body. (3) An invalid event occurred. (4) A valid RELEASE message with cause #16 is sent.					

Continued on next page

*Continued from previous page*

Default Dynamic Behaviour	
Detailed Comments : ...	
	(5) A RELEASE COMPLETE message is received from the IUT.
	(6) No response.



Default Dynamic Behaviour					
<b>Default Name</b> : DEF2_NCICS <b>Group</b> : <b>Objective</b> : Default behaviour for NCICS calls. <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		L2? DL_REL_IN		(I)	DL failure
2		CANCEL		R	
3		L2? DL_EST_IN		(I)	DL reset
4		+RELEASE_CALL			(1)
5		L2? PDUr	Mr ( NO_R1 ( tcv_flag_r, tcv_cref))		ignore
6		RETURN			(2)
7		L2? PDUr	Mr ( SQ_R1 ( tcv_flag_r, tcv_cref))		ignore
8		RETURN			(2)
9		L2? PDUr	Mr ( GFP_R1 ( tcv_flag_r, tcv_cref))		ignore
10		RETURN			(2)
11		L2? OTHERWISE		(F)	(3)
12		+RELEASE_CALL			(1)
		RELEASE_CALL			
13		L2! PDUs START T_AC	Ms ( RL_S1 ( tcv_flag_s, tcv_cref, 16))		(4)
14	L1	L2? PDUr CANCEL T_AC	Mr ( RC_R1 ( tcv_flag_r, tcv_cref))		(5)
15		? TIMEOUT T_AC			(6)
16		CANCEL		R	
17		L2? PDUr	Mr ( NO_R1 ( tcv_flag_r, tcv_cref))		ignore
18		GOTO L1			
19		L2? PDUr	Mr ( SQ_R1 ( tcv_flag_r, tcv_cref))		ignore
20		GOTO L1			
21		L2? PDUr	Mr ( GFP_R1 ( tcv_flag_r, tcv_cref))		ignore
22		GOTO L1			
23		L2? OTHERWISE			(3)
24		CANCEL		R	
<b>Detailed Comments</b> : (1) Subtree to release the call. (2) Return to the test body. (3) An invalid event occurred. (4) A valid RELEASE message with cause #16 is sent.					

Continued on next page

*Continued from previous page*

Default Dynamic Behaviour	
Detailed Comments : ...	
	(5) A RELEASE COMPLETE message is received from the IUT.
	(6) No response.

Default Dynamic Behaviour					
<b>Default Name</b> : DEF1_CALL (FL:INTEGER) <b>Group</b> : <b>Objective</b> : Default subtree for PTC. <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	CPA1?CP_M	STOP_PTC		
2		L1?DL_REL_IN			DL failure
3		L1?DL_EST_IN			DL reset
4		L1!PDU <sub>s</sub> START T_AC	Ms ( RL_S1 (FL, tcv_cref1, 16))		(2)
5		L1?PDU <sub>r</sub> CANCEL T_AC	Mr ( RC_R1 ((FL+1) MOD 2, tcv_cref1))		
6		?TIMEOUT T_AC			no response
7		L1?PDU <sub>r</sub>	Mr ( ANY_MSG )		ignore
8		GOTO L1			
9		L1?OTHERWISE			inv. event
10		L1?PDU <sub>r</sub>	Mr ( ANY_MSG )		ignore
11		RETURN			return to test case
12		L1?PDU <sub>r</sub>	Mr ( GFP_R1((FL+1) MOD 2, tcv_cref1))		ignore
13		RETURN			return to test case
14		L1?UPDU <sub>r</sub>	UMr ( GFP_R1((FL+1) MOD 2, tcv_cref1))		ignore
15		RETURN			return to test case
16		L1?OTHERWISE			
17		L1!PDU <sub>s</sub> START T_AC	Ms ( RL_S1 (FL, tcv_cref1, 16))		valid RELEASE
18	L2	L1?PDU <sub>r</sub> CANCEL T_AC	Mr ( RC_R1 ((FL+1) MOD 2, tcv_cref1))		valid REL_CO M

Continued on next page

Continued from previous page

Default Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
19		?TIMEOUT T_AC	Mr ( ANY_MSG)		no respons e
20		L1?PDUr			ignore
21		GOTO L2			
22		L1?OTHERWISE			inv. event
<b>Detailed Comments</b> : &COMMON_N09 <div>(1) (FL+1) MOD 2 is ususally used to store the inverted flag. This behaviour line is inserted to allow the assignment of a final verdict R. It is mandatory to assign a final verdict to each leaf of a default behaviour tree.</div> <div>(2) A valid RELEASE message with cause #16 is sent.</div>					

Default Dynamic Behaviour					
<b>Default Name</b> : DEF2_CALL (FL:INTEGER) <b>Group</b> : <b>Objective</b> : Default subtree for PTC. <b>Comments</b> :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	L1	CPA2?CP_M	STOP_PTC		
2		L2?DL_REL_IN			DL failure
3		L2?DL_EST_IN			DL reset
4		L2!PDU <sub>s</sub> START T_AC	Ms ( RL_S1 (FL, tcv_cref2, 16))		(2)
5		L2?PDU <sub>r</sub> CANCEL T_AC	Mr ( RC_R1 ((FL+1) MOD 2, tcv_cref2))		
6		?TIMEOUT T_AC			no response
7		L2?PDU <sub>r</sub>	Mr ( ANY_MSG )		ignore
8		GOTO L1			
9		L2?OTHERWISE			inv. event
10		L2?PDU <sub>r</sub>	Mr ( ANY_MSG )		ignore
11		RETURN			return to test case
12		L2?PDU <sub>r</sub>	Mr ( GFP_R1((FL+1) MOD 2, tcv_cref2))		ignore
13		RETURN			return to test case
14		L2?UPDU <sub>r</sub>	UMr ( GFP_R1((FL+1) MOD 2, tcv_cref2))		ignore
15		RETURN			return to test case
16		L2?OTHERWISE			
17		L2!PDU <sub>s</sub> START T_AC	Ms ( RL_S1 (FL, tcv_cref2, 16))		valid RELEASE
18	L2	L2?PDU <sub>r</sub> CANCEL T_AC	Mr ( RC_R1 ((FL+1) MOD 2, tcv_cref2))		valid REL_CO M

Continued on next page

Continued from previous page

Default Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
19		?TIMEOUT T_AC	Mr ( ANY_MSG)		no respons e
20		L2?PDUr			ignore
21		GOTO L2			
22		L2?OTHERWISE			inv. event
<b>Detailed Comments</b> : &COMMON_N09 <div>(1) (FL+1) MOD 2 is ususally used to store the inverted flag. This behaviour line is inserted to allow the assignment of a final verdict R. It is mandatory to assign a final verdict to each leaf of a default behaviour tree.</div> <div>(2) A valid RELEASE message with cause #16 is sent.</div>					