

GSM Protocol Stack

Test Specification BTI (Part ACI/DTI)

Confidential

Author: Condat DV-Beratung
Organisation Software GmbH
Alt-Moabit 91d
D-10559 Berlin
Germany

Date: 15 January 2001
Document No.: 8445.400.01.001
File: BTI.DOC

Table of Contents

0	DOCUMENT CONTROL	3
0.1	Document History	3
0.2	References	3
0.3	Abbreviations	7
0.4	Terms	9
1	OVERVIEW	10
2	PARAMETERS	12
3	TEST CASES	17
3.1	Routing (internal)	17
3.1.1	BTI001: Setup the Routing and the PCO view for the MMI test	18
3.1.2	BTI002: Setup the Routing and the PCO view for the MMI test	20
3.2	MMI Component Tests	21
3.2.1	BTI100: Power On Sequence	22
3.3	BTI-ACI Testcases	23
3.3.1	BTI200: BT terminal sends deinit indication	24
3.3.2	BTI201: BT terminal requests to open port 1	25
3.3.3	BTI202: BT terminal requests to open port 2	26
3.3.4	BTI203: BT terminal requests to close port 1	27
3.3.5	BTI204: BT terminal requests to close port 2	28
3.3.6	BTI205: BT terminal requests to open then close port 1	29
3.3.7	BTI206: BT terminal requests to open then close port 2	30
3.3.8	BTI207: BT terminal requests to open pt 1 then pt 2 (and closing pt2)	31
3.3.9	BTI208: BT terminal requests to open invalid port	32
3.3.10	BTI209: BT terminal requests to open port 1 after port 1 already opened	33
3.4	AT Commands	33
3.4.1	BTI301: BT terminal sends AT command (+CLIR=, +COLP=) single answer	34
3.4.2	BTI302: BT terminal sends AT command (+CMOD=?) multi answer	35
3.4.3	BTI303: BT terminal sends AT command (ATD123;) unfinished TEST	36
3.4.4	BTI304: BT terminal triggers AT command (ATD+COPN) with long output	37
3.4.5	BTI305: BT terminal sends AT command abort req	41
3.5	Data Transfer	41
3.5.1	BTI400: initiated DTI data transfer (waits for DTI_GETDATA_REQ)	42
3.5.2	BTI401: initiated DTI data transfer GSM-BT (DEBUG)	43
3.5.3	BTI403: initiated DTI data transfer BT-GSM (DEBUG)	44

0 Document Control

© Copyright Condat DV-Beratung Organisation und Software GmbH, 1997.

All rights reserved.

Every effort has been made to ensure that the information contained in this document is accurate at the time of printing. However, the software described in this document is subject to continuous development and improvement. Condat GmbH reserves the right to change the specification of the software. Information in this document is subject to change without notice and does not represent a commitment on the part of Condat GmbH. Condat GmbH accepts no liability for any loss or damage arising from the use of any information contained in this document.

The software described in this document is furnished under a licence agreement and may be used or copied only in accordance with the terms of the agreement. It is an offence to copy the software in any way except as specifically set out in the agreement. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Condat GmbH.

Condat DV-Beratung

Organisation und Software GmbH

Alt Moabit 91d

10559 Berlin

Germany

Telephone: +49.30.39094-0

Fax: +49.30.39094-300

Internet: <http://www.condat.de>

E-mail: gsm@condat.de

0.1 Document History

Document Id.	Date	Author	Remarks
yyyy.xxx.98.100	15-Jan-01	RM	Initial

0.2 References

- [1] GSM 2.81, Line Identification Supplementary Services - Stage 1
ETS 300 514, ETSI, September 1994
- [2] GSM 2.82, Call Forwarding Supplementary Services - Stage 1
ETS 300 515, ETSI, September 1994
- [3] GSM 2.83, Call Waiting and Call Hold Supplementary Services - Stage 1
ETS 300 516, ETSI, September 1994
- [4] GSM 2.84, Multi Party Supplementary Services - Stage 1
ETS 300 517, ETSI, September 1994

- [5] GSM 2.85, Closed User Group Supplementary Services - Stage 1
ETS 300 518, ETSI, September 1994
- [6] GSM 2.86, Advice of Charge Supplementary Services - Stage 1
ETS 300 519, ETSI, September 1994
- [7] GSM 2.88, Call Barring Supplementary Services - Stage 1
ETS 300 520, ETSI, September 1994
- [8] GSM 3.14, Support of Dual Tone Multi Frequency Signalling via the GSM System
ETS 300 532, ETSI, April 1994
- [9] GSM 3.40, Technical Realization of the Short Message Service Point-to-Point
ETS 300 536, ETSI, January 1996
- [10] GSM 3.41, Technical Realization of Short Message Service Cell Broadcast
ETS 300 537, ETSI, June 1995
- [11] GSM 3.81, Line Identification Supplementary Services - Stage 2
ETS 300 542, ETSI, February 1995
- [12] GSM 3.82, Call Forwarding Supplementary Services - Stage 2
ETS 300 543, ETSI, February 1995
- [13] GSM 3.83, Call Waiting and Call Hold Supplementary Services - Stage 2
ETS 300 544, ETSI, November 1994
- [14] GSM 3.84, Multi Party Supplementary Services - Stage 2
ETS 300 545, ETSI, November 1994
- [15] GSM 3.85, Closed User Group Supplementary Services - Stage 2
ETS 300 546, ETSI, January 1996
- [16] GSM 3.86, Advice of Charge Supplementary Services - Stage 2
ETS 300 547, ETSI, March 1995
- [17] GSM 3.88, Call Barring Supplementary Services - Stage 2
ETS 300 548, ETSI, November 1994
- [18] GSM 4.01, MS-BSS Interface General Aspects and Principles
ETS 300 550, ETSI, September 1994
- [18a] GSM 4.03, MS-BSS Interface Channel Structures and Access Capabilities
ETS 300 552, ETSI, September 1994
- [19] GSM 4.05, Data Link Layer General Aspects
ETS 300 554, ETSI, September 1994
- [20] GSM 4.06, MS-BSS Interface Data Link Layer Specification
ETS 300 555, ETSI, September 1994
- [21] GSM 4.07, Mobile Radio Interface Signalling Layer 3 General Aspects
ETS 300 556, ETSI, February 1995
- [22] GSM 4.08, Mobile Radio Interface Layer 3 Specification
ETS 300 557, ETSI, January 1996
- [23] GSM 4.10, Mobile Radio Interface Layer 3 Supplementary Services Specification
General Aspects
ETS 300 558, ETSI, February 1995
- [24] GSM 4.11, Point-to-Point Short Message Service Support on Mobile Radio Interface
ETS 300 559, ETSI, October 1995

- [25] GSM 4.12, Short Message Service Cell Broadcast Support on Mobile Radio Interface
ETS 300 560, ETSI, January 1996
- [26] GSM 4.80, Mobile Radio Interface Supplementary Services Specification Formats and Coding
ETS 300 564, ETSI, February 1995
- [27] GSM 4.81, Line Identification Supplementary Services - Stage 3
ETS 300 565, ETSI, February 1995
- [28] GSM 4.82, Call Forwarding Supplementary Services - Stage 3
ETS 300 566, ETSI, February 1995
- [29] GSM 4.83, Call Waiting and Call Hold Supplementary Services - Stage 3
ETS 300 567, ETSI, February 1995
- [30] GSM 4.84, Multi Party Supplementary Services - Stage 3
ETS 300 568, ETSI, February 1995
- [31] GSM 4.85, Closed User Group Supplementary Services - Stage 3
ETS 300 569, ETSI, February 1995
- [32] GSM 4.86, Advice of Charge Supplementary Services - Stage 3
ETS 300 570, ETSI, February 1995
- [33] GSM 4.88, Call Barring Supplementary Services - Stage 3
ETS 300 571, ETSI, February 1995
- [34] GSM 5.01, Physical Layer on the Radio Path General Description
ETS 300 573, ETSI, October 1995
- [35] GSM 5.02, Multiplexing and Multiple Access on the Radio Path
ETS 300 574, ETSI, January 1996
- [36] GSM 5.08, Radio Sub-system Link Control
ETS 300 578, ETSI, January 1996
- [37] GSM 5.10, Radio Sub-system Synchronisation
ETS 300 579, ETSI, October 1995
- [38] Service Access Point MMREG
6147.100.96.100; Condat GmbH
- [39] Service Access Point MNCC
6147.101.96.100; Condat GmbH
- [40] Service Access Point MNSS
6147.102.96.100; Condat GmbH
- [41] Service Access Point MNSMS
6147.103.96.100; Condat GmbH
- [42] Service Access Point MMCC
6147.104.97.100; Condat GmbH
- [43] Service Access Point MMSS
6147.105.97.100; Condat GmbH
- [44] Service Access Point MMSMS
6147.106.97.100; Condat GmbH
- [45] Service Access Point RR
6147.107.97.100; Condat GmbH
- [46] Service Access Point SIM
6147.108.97.100; Condat GmbH

- [47] Service Access Point MPH
6147.109.96.100; Condat GmbH
- [48] Service Access Point DL
6147.110.96.100; Condat GmbH
- [49] Service Access Point MDL
6147.111.96.100; Condat GmbH
- [50] Service Access Point PH
6147.112.97.100; Condat GmbH
- [51] Service Access Point MMI
6147.113.96.100; Condat GmbH
- [52] Service Access Point BTP
8445.100.00.002; Condat GmbH
- [53] Message Sequence Charts CC
6147.200.97.100; Condat GmbH
- [54] Message Sequence Charts SS
6147.201.97.100; Condat GmbH
- [55] Message Sequence Charts SMS
6147.202.97.100; Condat GmbH
- [56] Message Sequence Charts MM
6147.203.97.100; Condat GmbH
- [57] Message Sequence Charts RR
6147.204.96.100; Condat GmbH
- [58] Message Sequence Charts DL
6147.205.96.100; Condat GmbH
- [59] Users Guide
6147.300.96.100; Condat GmbH
- [60] Test Specification CC
6147.400.97.100; Condat GmbH
- [61] Test Specification SS
6147.401.97.100; Condat GmbH
- [62] Test Specification SMS
6147.402.97.100; Condat GmbH
- [63] Test Specification MM
6147.403.97.100; Condat GmbH
- [64] Test Specification RR
6147.404.97.100; Condat GmbH
- [65] Test Specification DL
6147.405.97.100; Condat GmbH
- [66] Test Specification CCD
6147.406.97.100; Condat GmbH
- [67] SDL Specification CC
6147.500.97.100; Condat GmbH
- [68] SDL Specification SS
6147.501.97.100; Condat GmbH

- [69] SDL Specification SMS
6147.502.97.100; Condat GmbH
- [70] SDL Specification MM
6147.503.97.100; Condat GmbH
- [71] SDL Specification RR
6147.504.97.100; Condat GmbH
- [72] SDL Specification DL
6147.505.97.100; Condat GmbH
- [73] Message Specification CC
6147.600.97.100; Condat GmbH
- [74] Message Specification SS
6147.601.97.100; Condat GmbH
- [75] Message Specification SMS
6147.602.97.100; Condat GmbH
- [76] Message Specification MM
6147.603.97.100; Condat GmbH
- [77] Message Specification RR
6147.604.97.100; Condat GmbH
- [78] Message Specification DL
6147.605.97.100; Condat GmbH
- [79] Technical Documentation CC
6147.700.97.100; Condat GmbH
- [80] Technical Documentation SS
6147.701.97.100; Condat GmbH
- [81] Technical Documentation SMS
6147.702.97.100; Condat GmbH
- [82] Technical Documentation MM
6147.703.97.100; Condat GmbH
- [83] Technical Documentation RR
6147.704.97.100; Condat GmbH
- [84] Technical Documentation DL
6147.705.97.100; Condat GmbH
- [85] Technical Documentation CCD
6147.706.97.100; Condat GmbH
- [86] GTI-BTI Interface Description
8445.200.00.008; Condat GmbH

0.3 Abbreviations

AGCH	Access Grant Channel
BCCH	Broadcast Control Channel
BS	Base Station
BSIC	Base Station Identification Code

CBCH	Cell Broadcast Channel
CBQ	Cell Bar Qualify
CC	Call Control
CCCH	Common Control Channel
CCD	Condat Coder Decoder
CKSN	Ciphering Key Sequence Number
C/R	Command / Response
C1	Path Loss Criterion
C2	Reselection Criterion
DCCH	Dedicated Control Channel
DISC	Disconnect Frame
DL	Data Link Layer
DM	Disconnected Mode Frame
EA	Extension Bit Address Field
EL	Extension Bit Length Field
EMMI	Electrical Man Machine Interface
F	Final Bit
FACCH	Fast Associated Control Channel
FHO	Forced Handover
GP	Guard Period
GSM	Global System for Mobile Communication
HPLMN	Home Public Land Mobile Network
I	Information Frame
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
Kc	Authentication Key
L	Length Indicator
LAI	Location Area Information
LPD	Link Protocol Discriminator
M	More Data Bit
MCC	Mobile Country Code
MM	Mobility Management
MMI	Man Machine Interface
MNC	Mobile Network Code
MS	Mobile Station
NCC	National Colour Code
NECI	New Establishment Causes included
N(R)	Receive Number
N(S)	Send Number
OTD	Observed Time Difference
P	Poll Bit
PCH	Paging Channel
PDU	Protocol Description Unit
P/F	Poll / Final Bit
PL	Physical Layer
PLMN	Public Land Mobile Network
RACH	Random Access Channel
REJ	Reject Frame

RNR	Receive Not Ready Frame
RR	Radio Resource Management
RR	Receive Ready Frame
RTD	Real Time Difference
SABM	Set Asynchronous Balanced Mode
SACCH	Slow Associated Control Channel
SAP	Service Access Point
SAPI	Service Access Point Identifier
SDCCH	Slow Dedicated Control Channel
SIM	Subscriber Identity Module
SMS	Short Message Service
SMSCB	Short Message Service Cell Broadcast
SS	Supplementary Services
TCH	Traffic Channel
TCH/F	Traffic Channel Full Rate
TCH/H	Traffic Channel Half Rate
TDMA	Time Division Multiple Access
TMSI	Temporary Mobile Subscriber Identity
UA	Unnumbered Acknowledgement Frame
UI	Unnumbered Information Frame
VPLMN	Visiting Public Land Mobile Network
V(A)	Acknowledgement State Variable
V(R)	Receive State Variable
V(S)	Send State Variable

0.4 Terms

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

1 Overview

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

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

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

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

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

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

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

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

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

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

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

Error! Objects cannot be created from editing field codes.

Figure 1: Mobile-station protocol architecture

This document describes the tests for the whole protocol stack using the MMI Framework.

2 Parameters

```
#define PHASE_2_PIN_ENTERING 2
```

```
DECLARATION (PLMN_262_01)
DECLARATION (PLMN_262_02)
DECLARATION (PLMN_262_03)
DECLARATION (PLMN_NO_ID)
DECLARATION (MCC_NONE)
DECLARATION (MNC_NONE)
DECLARATION (MCC_262)
DECLARATION (MCC_765)
DECLARATION (MNC_01)
DECLARATION (MNC_02)
DECLARATION (MNC_03)
DECLARATION (IMSI)
DECLARATION (SEA_BR_DEF)
DECLARATION (SCA_TI_DEF)
DECLARATION (SCA_BR_DEF)
DECLARATION (ADDRESS_HEADSET)
DECLARATION (NAME_HEADSET1)
DECLARATION (NAME_HEADSET2)
DECLARATION (PIN)
DECLARATION (DATA_FIELD)
DECLARATION (DATA_RLP_0_A_1)
DECLARATION (A_ECC_FIELD)
DECLARATION (A_AD_FIELD_CI_DISABLED)
```

```
#define PIN_1                1
#define PIN_2                2
#define HEADSET              1
#define OK                   0
#define NOK                  1
#define SEA_TI_DEF           0x100
#define SEA_BR_DEF           0x100
#define SCA_TI_DEF           0x100
#define SCA_BR_DEF           0x100
#define PARK_DEF             0x01
```

```
/*---"OK"---(SUCCESSFUL OPERATION)*/
```

```
// Message:
STRING(M_OK, "OK" )
BYTE LM_OK 2
```

```
/*---"CLIR_COLP"---(CLIR_COLP_S)*/
```

```
// Command:
STRING(C_CLIR_COLP_S, "AT+CLIR=2;+COLP=1")
BYTE LC_CLIR_COLP_S 18
```

```
/*--- "+CMOD=?" (CMOD_T) ---*/
```

```
// Command:
STRING(C_CMOD_T, "AT+CMOD=?")
BYTE LC_CMOD_T 10
// Message:
STRING(M_CMOD_T, "+CMOD: (0-3)")
BYTE LM_CMOD_T 12
```

```
/*---"CLIR_COLP"—(CLIR_COLP_S)*/
```

```
// Command:
STRING(C_ATD_S, "ATD123;")
BYTE LC_ATD_S 8
```

```
/*---"CLIR_COLP"—(CLIR_COLP_S)*/
```

```
// Command:
STRING(C_COPN_S, "AT+COPN")
BYTE LC_COPN_S 8
```

```
/*
```

```
Command:      +CMEE
               extended error report mode
```

```
*/
```

```
STRING(C_PLUS_CMEE_VERB, "AT+CMEE=2 ")
BYTE LC_PLUS_CMEE_VERB 9
```

```
/*
```

```
/*
```

```
Command:      +CFUN
               set phone functionality
```

```
*/
```

```
STRING(C_PLUS_CFUN_FULL, "AT+CFUN=1 ")
BYTE LC_PLUS_CFUN_FULL 9
```

```
STRING(UL_NAME, "BTI" )
```

```
BYTE UL_NAME_L 4
```

```
BYTE TI_MO_0      0x00
```

```
BYTENUM_00
```

BYTE NUM_1 1

BYTE NUM_2 2

BYTE NUM_3 3

BYTE NUM_4 4

BYTE NUM_9 9

BYTE NUM_10 10

BYTE NUM_12 12

BYTE NUM_18 18

BYTE NUM_20 20

BYTE NUM_27 27

BYTE NUM_30 30

BYTE NUM_50 50

BYTE V_PLMN_PRES 1

BYTE V_PLMN_NOT_PRES 0

BYTE SECMOD_DEF 1

BYTE CONATT_DEF 0x10

BYTE MAXRIN_DEF 0x10

BYTE BOND_DEF 0x01

BYTE SECMOD_2 2

BYTE CONATT_5 0x05

BYTE MAXRIN_3 0x03

BYTE PARK_OFF 1

BYTE BOND_OFF 1

BYTE ESTABL 0x01

BYTE RELEASED 0x02

BYTE DUMMY_B 0

FIELD (PREF_PLMN)

0x32, 0xF8, 0x10,
 0x32, 0xF8, 0x20,
 0x02, 0xF8, 0x10,
 0x02, 0xF8, 0x01,
 0xFF, 0xFF, 0xFF,
 0xFF, 0xFF, 0xFF

ENDFIELD (PREF_PLMN, 30)

FIELD (PIN_1_VALUE)

0x31,0x32,0x33,0x34,0xFF,0xFF, 0xFF,0xFF

ENDFIELD (PIN_1_VALUE, 8)

FIELD (EC_CODES) 0X11, 0XF2, 0XFF,
 0X99, 0XF9, 0XFF,
 0XFF, 0XFF, 0XFF,
 0XFF, 0XFF, 0XFF,

```

        0XFF, 0XFF, 0XFF
    ENDFIELD (EC_CODES, 15)

```

```

/* SIM service table */

```

```

FIELD (F_SIM_SRV) 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
ENDFIELD (F_SIM_SRV, 10)

```

```

BEGINARRAY (MCC_262, 3) 0x02, 0x06, 0x02 ENDARRAY
BEGINARRAY (MCC_765, 3) 0x07, 0x06, 0x05 ENDARRAY
BEGINARRAY (MNC_01, 2) 0x00, 0x01 ENDARRAY
BEGINARRAY (MNC_02, 2) 0x00, 0x02 ENDARRAY
BEGINARRAY (MNC_03, 2) 0x00, 0x03 ENDARRAY
BEGINARRAY (MNC_NONE, 3) 0x0F, 0x0F, 0x0F ENDARRAY
BEGINARRAY (ADDRESS_HEADSET, 6) 0xbb, 0xbb, 0xbb, 0xbb, 0xbb, 0xbb ENDARRAY
BEGINARRAY (NAME_HEADSET1, 2) 0x41, 0x42 ENDARRAY
BEGINARRAY (NAME_HEADSET2, 2) 0x43, 0x44 ENDARRAY
BEGINARRAY (PIN, 6) 0x01, 0x02, 0x03, 0x04, 0x05, 0x06 ENDARRAY

```

```

BEGIN_PSTRUCT ("plmn", PLMN_262_01)
    SET_COMP ("v_plmn",          V_PLMN_PRESENT)
    SET_COMP ("mcc",            MCC_262)
    SET_COMP ("mnc",            MNC_01)
ENDSTRUCT

```

```

BYTE DATA_LEN 48

```

```

BEGINARRAY (DATA_FIELD, 48)

```

```

    0x41, 0x42, 0x43, 0x44, 0x45, 0x46, 0x47, 0x48,
    0x41, 0x42, 0x43, 0x44, 0x45, 0x46, 0x47, 0x48,
    0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38,
    0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38,
    0x41, 0x42, 0x43, 0x44, 0x45, 0x46, 0x47, 0x48,
    0x41, 0x42, 0x43, 0x44, 0x45, 0x46, 0x47, 0x48

```

```

ENDARRAY

```

```

BEGINARRAY (DATA_RLP_0_A_1, 31)

```

```

    0xD8, 0x00,
    0x00, 0x00,
    0x00, 0x00,
    0x1F,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

```

```
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
ENDARRAY
/* EF ECC field array */
BEGINARRAY(A_ECC_FIELD,12) 0x11, 0xF2, 0xFF, 0x99, 0xF9, 0xFF, 0x21, 0x43, 0x65, 0xFF, 0xFF, 0xFF ENDARRAY
/* EF AD field array , disable CI */
BEGINARRAY_PART(A_AD_FIELD_CI_DISABLED,4) 0x00, 0x00, 0x00, 0x02 ENDARRAY
/* Transmissiun unit identifier dummy (PPP)*/
SHORT TUI 0
/* Chanel Identifier dummy */
BYTE C_ID_DUMMY 1
BYTE OP_ACK 1
BYTE BTI_NAK 0
BYTE BTI_ACK 1
BYTE ABORT_CMD 1
LONG PORT1 1
LONG PORT2 2
LONG PORT_INV 99
BYTE ACL_SRC_ID1 2
BYTE ACL_SRC_ID2 3
BYTE RATE 2
SHORT BUF_SIZE 4096
SHORT L2R_TUI 0x4321
SHORT UL_TUI 0x4322
BYTE K_MS_IWF 61
BYTE K_IWF_MS 61
BYTE T1 48
BYTE T2 9
BYTE N2 6
BYTE PT 0
BYTE P0 0
SHORT P1 512
BYTE P2 6
BYTE BPP_25_2 50
BYTE DTI_C_ID_DEFAULT 0x5
BYTE RLP_VERS 1
BYTE RLP_ACK 0
```

BYTE RLP_NAK 1

SHORT DATA_SIZE_25 25

/*frame size for channel coding TCH/F9.6*/

3 TEST CASES

3.1 Routing (internal)

3.1.1 BTI001: Setup the Routing and the PCO view for the MMI test

Description:

Routings for the ACI tests are set

Preamble:

None

APL	ACI	PS
COMMAND (TAP RESET)		
COMMAND (CC RESET)		
COMMAND (MM RESET)		
COMMAND (SIM RESET)		
COMMAND (SS RESET)		
COMMAND (MMI RESET)		
COMMAND (SMS RESET)		
COMMAND (RA RESET)		
COMMAND (T30 RESET)		
COMMAND (L2R RESET)		
COMMAND (TAP REDIRECT CLEAR)		
COMMAND (CC REDIRECT CLEAR)		
COMMAND (MM REDIRECT CLEAR)		
COMMAND (SIM REDIRECT CLEAR)		
COMMAND (SS REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CLEAR)		
COMMAND (SMS REDIRECT CLEAR)		
COMMAND (RA REDIRECT CLEAR)		
COMMAND (T30 REDIRECT CLEAR)		
COMMAND (L2R REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CC TAP)		
COMMAND (MMI REDIRECT MM TAP)		
COMMAND (MMI REDIRECT SIM TAP)		
COMMAND (MMI REDIRECT SS TAP)		
COMMAND (MMI REDIRECT MMI TAP)		
COMMAND (MMI REDIRECT SMS TAP)		
COMMAND (MMI REDIRECT RA TAP)		
COMMAND (MMI REDIRECT T30 TAP)		
COMMAND (MMI REDIRECT L2R TAP)		
COMMAND (TAP REDIRECT TAP MMI)		
COMMAND (MMI REDIRECT MMI TAP)		
COMMAND (CST REDIRECT L1 TAP)		

Parametrization:

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

History: 15.01.01 RM Initial

3.1.2 BTI002: Setup the Routing and the PCO view for the MMI test

Description:

Routings for the ACI tests are set

Preamble:

None

APL	ACI	PS
COMMAND (TAP RESET)		
COMMAND (CC RESET)		
COMMAND (MM RESET)		
COMMAND (SIM RESET)		
COMMAND (SS RESET)		
COMMAND (MMI RESET)		
COMMAND (SMS RESET)		
COMMAND (RA RESET)		
COMMAND (T30 RESET)		
COMMAND (TAP REDIRECT CLEAR)		
COMMAND (CC REDIRECT CLEAR)		
COMMAND (MM REDIRECT CLEAR)		
COMMAND (SIM REDIRECT CLEAR)		
COMMAND (SS REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CLEAR)		
COMMAND (SMS REDIRECT CLEAR)		
COMMAND (RA REDIRECT CLEAR)		
COMMAND (T30 REDIRECT CLEAR)		
COMMAND (MMI REDIRECT CC TAP)		
COMMAND (MMI REDIRECT MM TAP)		
COMMAND (MMI REDIRECT SIM TAP)		
COMMAND (MMI REDIRECT SS TAP)		
COMMAND (MMI REDIRECT MMI TAP)		
COMMAND (MMI REDIRECT SMS TAP)		
COMMAND (MMI REDIRECT RA TAP)		
COMMAND (MMI REDIRECT T30 TAP)		
COMMAND (TAP REDIRECT TAP MMI)		
COMMAND (MMI REDIRECT MMI TAP)		
COMMAND (CST REDIRECT L1 TAP)		

Parametrization:

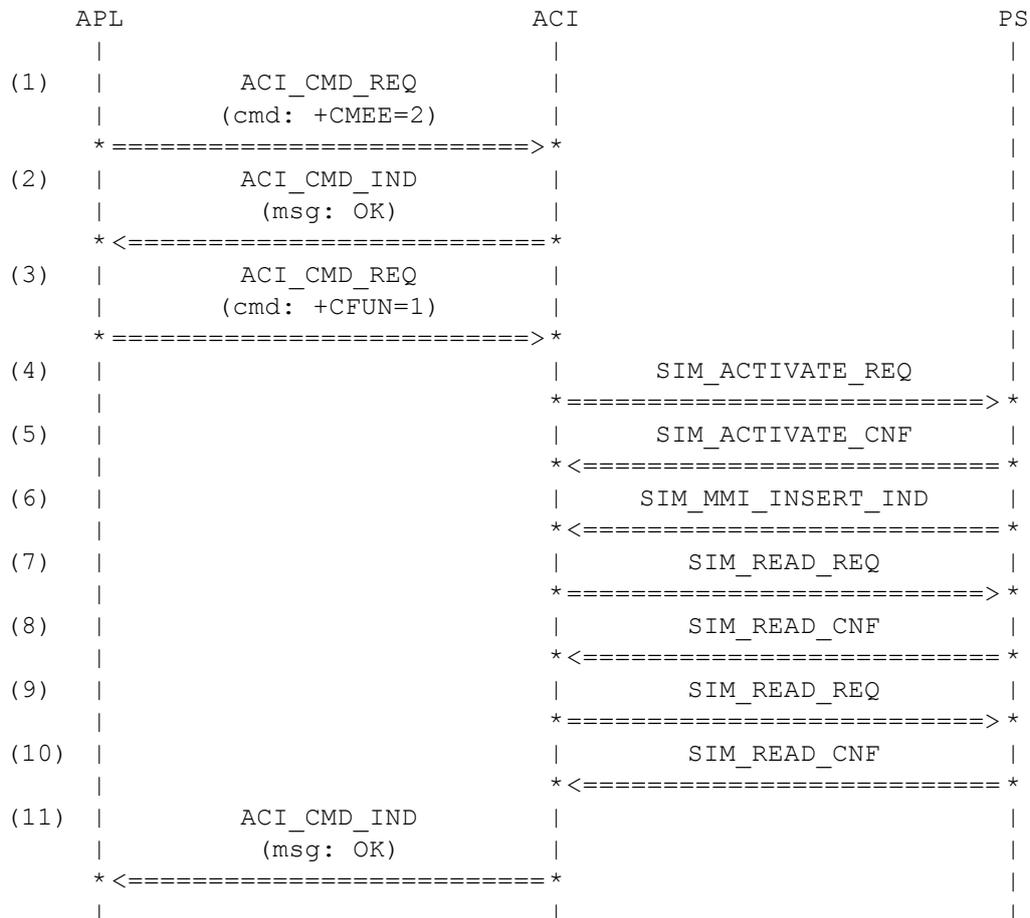
Primitive	Parameter	Value
History:	15.01.01	RM Initial

3.2 MMI Component Tests

3.2.1 BTI100: Power On Sequence

Description:

Preamble: BTI001



Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CMEE_VERB
	cmd_seq	C_PLUS_CMEE_VERB
(2) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK
(3) ACI_CMD_REQ	cmd_src	CMD_SRC_EXT
	cmd_len	LC_PLUS_CFUN_FULL
	cmd_seq	C_PLUS_CFUN_FULL
(4) SIM_ACTIVATE_REQ	proc	SIM_INITIALISATION

	mimi_pro_file	NOT_USED
	stk_pro_file	NOT_USED
(5) SIM_ACTIVATE_CNF	error	SIM_INIT_NO_ERROR
	pin_cnt	NUM_3
	puk_cnt	NUM_10
	pin2_cnt	NUM_3
	puk2_cnt	NUM_10
	ec_code	EC_CODES
	pref_lang	NOT_USED
(6) SIM_MMI_INSERT_IND	func	SIM_ADN_ENABLED
	sim_serv	F_SIM_SRV
	imsi_field	NOT_USED
	pref_plmn	NOT_USED
	phase	PHASE_2_SIM
	access_acm	NOT_USED
	access_acmmax	NOT_USED
	access_puct	NOT_USED
(7) SIM_READ_REQ	source	SRC_MMI
	offset	NOT_USED
	datafield	SIM_ECC
	length	NOT_PRESENT_8BIT
	max_length	NOT_USED
(8) SIM_READ_CNF	datafield	SIM_ECC
	error	SIM_NO_ERROR
	length	NOT_USED
	trans_data	NOT_USED
(9) SIM_READ_REQ	source	SRC_MMI
	offset	NOT_USED
	datafield	SIM_AD
	length	NOT_PRESENT_8BIT
	max_length	NOT_USED
(10) SIM_READ_CNF	datafield	SIM_AD
	error	SIM_NO_ERROR
	length	NOT_USED
	trans_data	NOT_USED
(11) ACI_CMD_IND	cmd_len	LM_OK
	cmd_seq	M_OK
History:	17.10.01	RM Initial

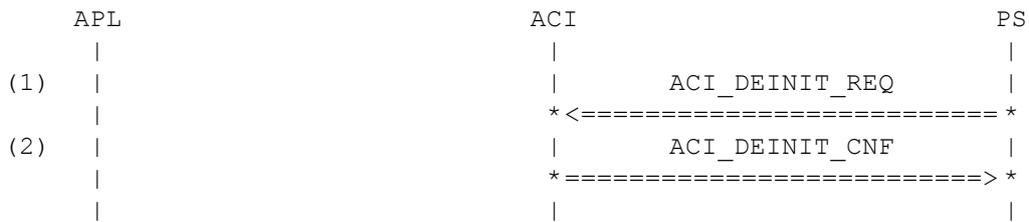
3.3 BTI-ACI Testcases

3.3.1 BTI200: BT terminal sends deinit indication

Description:

Preamble:

BTI100



Parametrization:

Primitive	Parameter	Value
(1) ACI_DEINIT_REQ		
(2) ACI_DEINIT_CNF		
History:	24.01.01	KK Initial

3.3.2 BTI201: BT terminal requests to open port 1

Description:

Preamble:

BTI100

APL		ACI		PS
(1)			ACI_OPEN_PORT_REQ	
		*	<=====	*
(2)			ACI_OPEN_PORT_CNF	
		*	=====	*

Parametrization:

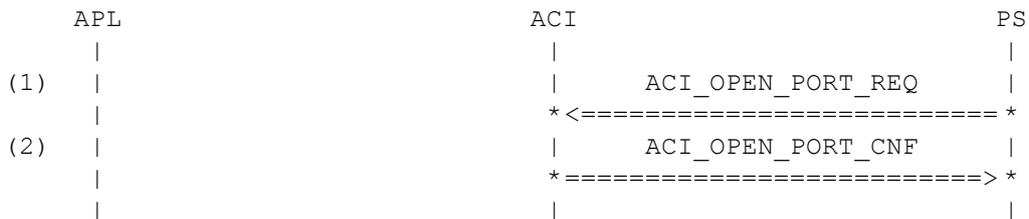
Primitive	Parameter	Value
(1) ACI_OPEN_PORT_REQ	port_nb	PORT1
(2) ACI_OPEN_PORT_CNF	port_nb	PORT1
	ackflg	BTI_ACK
History:	24.01.01	KK Initial

3.3.3 BTI202: BT terminal requests to open port 2

Description:

Preamble:

BTI100



Parametrization:

Primitive	Parameter	Value
(1) ACI_OPEN_PORT_REQ	port_nb	PORT2
(2) ACI_OPEN_PORT_CNF	port_nb	PORT2
	ackflg	BTI_ACK
History:	24.01.01	KK Initial

3.3.4 BTI203: BT terminal requests to close port 1

Description:

Preamble:

BTI100

	APL		ACI		PS
(1)				ACI_CLOSE_PORT_REQ	
				<=====	
(2)				ACI_CLOSE_PORT_CNF	
				=====>	

Parametrization:

Primitive	Parameter	Value
(1) ACI_CLOSE_PORT_REQ	port_nb	PORT1
(2) ACI_CLOSE_PORT_CNF	port_nb	PORT1
History:	24.01.01	KK Initial

3.3.5 BTI204: BT terminal requests to close port 2

Description:

Preamble:

BTI100

	APL		ACI		PS
(1)				ACI_CLOSE_PORT_REQ	
				<=====	
(2)				ACI_CLOSE_PORT_CNF	
				=====>	

Parametrization:

	Primitive	Parameter	Value
(1)	ACI_CLOSE_PORT_REQ	port_nb	PORT2
(2)	ACI_CLOSE_PORT_CNF	port_nb	PORT2
History:		24.01.01	KK Initial

3.3.6 BTI205: BT terminal requests to open then close port 1

Description:

Preamble:

BTI100

	APL		ACI		PS
(1)				ACI_OPEN_PORT_REQ	
				<=====	
(2)				ACI_OPEN_PORT_CNF	
				=====>	
(3)				ACI_CLOSE_PORT_REQ	
				<=====	
(4)				ACI_CLOSE_PORT_CNF	
				=====>	

Parametrization:

Primitive	Parameter	Value
(1) ACI_OPEN_PORT_REQ	port_nb	PORT1
(2) ACI_OPEN_PORT_CNF	port_nb ackflg	PORT1 BTI_ACK
(3) ACI_CLOSE_PORT_REQ	port_nb	PORT1
(4) ACI_CLOSE_PORT_CNF	port_nb	PORT1
History:	24.01.01	KK Initial

3.3.7 BTI206: BT terminal requests to open then close port 2

Description:

Preamble:

BTI100

APL	ACI	PS
(1)	ACI_OPEN_PORT_REQ	
	* <=====*	
(2)	ACI_OPEN_PORT_CNF	
	* =====>*	
(3)	ACI_CLOSE_PORT_REQ	
	* <=====*	
(4)	ACI_CLOSE_PORT_CNF	
	* =====>*	

Parametrization:

Primitive	Parameter	Value
(5) ACI_OPEN_PORT_REQ	port_nb	PORT2
(6) ACI_OPEN_PORT_CNF	port_nb ackflg	PORT2 BTI_ACK
(7) ACI_CLOSE_PORT_REQ	port_nb	PORT2
(8) ACI_CLOSE_PORT_CNF	port_nb	PORT2
History:	24.01.01	KK Initial

3.3.8 BTI207: BT terminal requests to open pt 1 then pt 2 (and closing pt2)

Description:

Preamble:

BTI201

APL	ACI	PS
(1)	ACI_OPEN_PORT_REQ	
	* <=====*	
(2)	ACI_OPEN_PORT_CNF	
	* =====>*	
(3)	ACI_CLOSE_PORT_REQ	
	* <=====*	
(4)	ACI_CLOSE_PORT_CNF	
	* =====>*	

Parametrization:

Primitive	Parameter	Value
(9) ACI_OPEN_PORT_REQ	port_nb	PORT2
(10) ACI_OPEN_PORT_CNF	port_nb ackflg	PORT2 BTI_ACK
(11) ACI_CLOSE_PORT_REQ	port_nb	PORT2
(12) ACI_CLOSE_PORT_CNF	port_nb	PORT2
History:	24.01.01	KK Initial

3.3.9 BTI208: BT terminal requests to open invalid port

Description:

Preamble:

BTI100

APL		ACI		PS
(1)			ACI_OPEN_PORT_REQ	
			* <=====*	
(2)			ACI_OPEN_PORT_CNF	
			* =====>*	

Parametrization:

Primitive	Parameter	Value
(1) ACI_OPEN_PORT_REQ	port_nb	PORT_INV
(2) ACI_OPEN_PORT_CNF	port_nb ackflg	PORT_INV BTI_NAK
History:	24.01.01	KK Initial

3.3.10 BTI209: BT terminal requests to open port 1 after port 1 already opened

Description:

Preamble:

BTI201

APL		ACI		PS
(1)				
			ACI_OPEN_PORT_REQ	
			<=====	
(2)				
			ACI_OPEN_PORT_CNF	
			=====>	

Parametrization:

Primitive	Parameter	Value
(3) ACI_OPEN_PORT_REQ	port_nb	PORT1
(4) ACI_OPEN_PORT_CNF	port_nb	PORT1
	ackflg	BTI_NAK
History:	24.01.01	KK Initial

3.4 AT Commands

3.4.1 BTI301: BT terminal sends AT command (+CLIR=, +COLP=) single answer

Description:

setting calling line id present and connection line id restriction

Preamble:

```

BTI201
      APL                               ACI                               PS
      |                                 |                                 |
(1)  |                                 | ACI_CMD_REQ_BT                 |
      |                                 | (cmd: +CLIR=, +COLP=)         |
      |                                 | *<=====*>                   |
(2)  |                                 | ACI_CMD_CNF_BT                 |
      |                                 | (cmd: OK)                       |
      |                                 | *=====*>                     |
      |                                 |                                 |
  
```

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ_BT	cmd_src	CMD_SRC_EXT
	cmd_len	LC_CLIR_COLP_S
	cmd_seq	C_CLIR_COLP_S
	src_id	ACI_SRC_ID1
(2) ACI_CMD_CNF_BT	cmd_len	LM_OK
	cmd_seq	M_OK
	src_id	ACI_SRC_ID1
History:	24.01.01	KK Initial

3.4.2 BTI302: BT terminal sends AT command (+CMOD=?) multi answer

Description:

Call Mode, test of supported call modes

Preamble:

```

BTI201
APL                               ACI                               PS
|                                 |                                 |
(1) | ACI_CMD_REQ_BT              |                                 |
    | (cmd: +CMOD=?)              |                                 |
    | *=====>*                  |                                 |
(2) | ACI_CMD_IND_BT              |                                 |
    | (cmd: +CMOD: (0-1))         |                                 |
    | *<=====*                  |                                 |
(3) | ACI_CMD_RES_BT              |                                 |
    | *=====>*                  |                                 |
(4) | ACI_CMD_CNF_BT              |                                 |
    | (cmd: OK)                   |                                 |
    | *<=====*                  |                                 |
|                                 |                                 |

```

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ_BT	cmd_src	CMD_SRC_EXT
	cmd_len	LC_CMOD_T
	cmd_seq	C_CMOD_T
	src_id	ACI_SRC_ID1
(2) ACI_CMD_IND_BT	cmd_len	LM_CMOD_T
	cmd_seq	M_CMOD_T
	src_id	ACI_SRC_ID1
(3) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(4) ACI_CMD_CNF_BT	cmd_len	LM_OK
	cmd_seq	M_OK
	src_id	ACI_SRC_ID1
History:	25.01.01	KK Initial

3.4.3 BTI303: BT terminal sends AT command (ATD123;) unfinished TEST

Description:

setting calling line id present and connection line id restriction

Preamble:

```

BTI201
      APL                               ACI                               PS
      |                                 |                                 |
(1)  |                                 |   ACI_CMD_RES_BT           |
      |                                 | *<=====*>                |
(2)  |                                 |   ACI_CMD_REQ_BT           |
      |                                 |   (cmd: ATD123)           |
      |                                 | *<=====*>                |
      |                                 |                                 |
  
```

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(1) ACI_CMD_REQ_BT	cmd_src	CMD_SRC_EXT
	cmd_len	LC_ATD_S
	cmd_seq	C_ATD_S
	src_id	ACI_SRC_ID1

History: 25.01.01 KK Initial

3.4.4 BTI304: BT terminal triggers AT command (ATD+COPN) with long output

Description:

COPN

Preamble:

BT1201

	APL	ACI	PS
(1)		ACI_CMD_REQ_BT	
		(cmd: AT+COPN)	
		*<=====	
(2)		ACI_CMD_IND_BT	
		*=====>	
(3)		ACI_CMD_RES_BT	
		*<=====	
(4)		ACI_CMD_IND_BT	
		*=====>	
(5)		ACI_CMD_RES_BT	
		*<=====	
(6)		ACI_CMD_IND_BT	
		*=====>	
(7)		ACI_CMD_RES_BT	
		*<=====	
(8)		ACI_CMD_IND_BT	
		*=====>	
(9)		ACI_CMD_RES_BT	
		*<=====	
(10)		ACI_CMD_IND_BT	
		*=====>	
(11)		ACI_CMD_RES_BT	
		*<=====	
(12)		ACI_CMD_IND_BT	
		*=====>	
(13)		ACI_CMD_RES_BT	
		*<=====	
(14)		ACI_CMD_IND_BT	
		*=====>	
(15)		ACI_CMD_RES_BT	
		*<=====	
(16)		ACI_CMD_IND_BT	
		*=====>	
(17)		ACI_CMD_RES_BT	
		*<=====	
(18)		ACI_CMD_IND_BT	
		*=====>	
(19)		ACI_CMD_RES_BT	
		*<=====	
(20)		ACI_CMD_IND_BT	
		*=====>	
(21)		ACI_CMD_RES_BT	
		*<=====	
(22)		ACI_CMD_IND_BT	
		*=====>	
(23)		ACI_CMD_RES_BT	
		*<=====	
(24)		ACI_CMD_IND_BT	

```

(25) |                                     *=====>*
      |          ACI_CMD_RES_BT          |
      |                                     *<=====*
```

```

(26) |                                     *=====>*
      |          ACI_CMD_IND_BT          |
      |                                     *<=====*
```

```

(27) |                                     *=====>*
      |          ACI_CMD_RES_BT          |
      |                                     *<=====*
```

```

(28) |                                     *=====>*
      |          ACI_CMD_CNF_BT          |
      |                                     *<=====*
```

Parametrization:

Primitive	Parameter	Value
(1) ACI_CMD_REQ_BT	cmd_src	CMD_SRC_EXT
	cmd_len	LC_COPN_S
	cmd_seq	C_COPN_S
	src_id	ACI_SRC_ID1
(2) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(3) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(4) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(5) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(6) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(7) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(8) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(9) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(10) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(11) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(12) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1

(13) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(14) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(15) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(16) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(17) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(18) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(19) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(20) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(21) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(22) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(23) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(24) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(25) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(26) ACI_CMD_IND_BT	cmd_len	NOT_USED
	cmd_seq	NOT_USED
	src_id	ACI_SRC_ID1
(27) ACI_CMD_RES_BT	src_id	ACI_SRC_ID1
(28) ACI_CMD_CNF_BT	cmd_len	LM_OK
	cmd_seq	M_OK
	src_id	ACI_SRC_ID1

History: 25.01.01 KK Initial

3.4.5 BTI305: BT terminal sends AT command abort req

Description:

COPN

Preamble:

BTI304

APL		ACI		PS
(1)			ACI_ABORT_REQ_BT	
			<=====	
(2)			ACI_ABORT_CNF_BT	
			=====>	

Parametrization:

Primitive	Parameter	Value
(1) ACI_ABORT_REQ_BT	cmd_src	CMD_SRC_EXT
	cause	ABORT_CMD
	src_id	ACI_SRC_ID1
(2) ACI_ABORT_CNF_BT	src_id	ACI_SRC_ID1

History: 25.01.01 KK Initial

3.5 Data Transfer

3.5.1 BTI400: initiated DTI data transfer (waits for DTI_GETDATA_REQ)

Description:

test

Preamble:

```

BTI201
  APL          ACI          PS
  |            |            |
(1) |          |          BTI_SEND_DATA_TEST |
    |          |          *<=====*>
(2) |          |          DTI_GETDATA_REQ   |
    |          |          *=====*>
    |            |            |

```

Parametrization:

Primitive	Parameter	Value
(1) BTI_SEND_DATA_TEST	len	DATA_LEN
	port_nb	PORT1
	ind_data	DATA_FIELD
(2) DTI_GETDATA_REQ	tui	TUI
	c_id	C_ID_DUMMY
	op_ack	OP_ACK

History:

25.01.01

KK Initial

3.5.2 BTI401: initiated DTI data transfer GSM-BT (DEBUG)

Description:

test

Preamble:

```

BTI201
  APL          ACI          PS
  |            |            |
(1) |          |          BTI_SEND_DATA_TEST |
  |            |          *<-----*
  |            |          |
  
```

Parametrization:

Primitive	Parameter	Value
(1) BTI_SEND_DATA_TEST	len	DATA_LEN
	port_nb	PORT1
	ind_data	DATA_FIELD

History: 25.01.01 KK Initial

