Technical Document - Confidential

GSM PROTOCOL STACK

MESSAGE SEQUENCE CHARTS

SIM

Document Number:	6301.201.98.102
Version:	0.4
Status:	Draft
Approval Authority:	
Creation Date:	1997-Oct-16
Last changed:	2015-Mar-08 by XINTEGRA
File Name:	Sim.doc

Important Notice

Change History

Date	Changed by	Approved by	Version	Status	Notes
1997-Oct-16	Stefan Lemke et al.		0.1		1
1998-Oct-27	Stefan Lemke et al.		0.2		2
2002-Mar-14	STW		0.3		3
2003-Jun-11	XINTEGRA		0.4	Draft	

Notes:

1. Initial version
2. SIM Toolkit, Phase 2+
3. SAT class "c/e"

Table of Contents

List of Figures and Tables

List of References

[ISO 9000:2000] International Organization for Standardization. Quality management systems - Fundamentals and vocabulary. December 2000

1.1 References

[1] GSM 2.81, Line Identification Supplementary Services - Stage 1
ETS 300 514, ETSI, September 1994
[2] GSM 2.82, Call Forwarding Supplementary Services - Stage 1
ETS 300 515, ETSI, September 1994
[3] GSM 2.83, Call Waiting and Call Hold Supplementary Services - Stage 1
ETS 300 516, ETSI, September 1994
[4] GSM 2.84, Multi Party Supplementary Services - Stage 1
ETS 300 517, ETSI, September 1994
[5] GSM 2.85, Closed User Group Supplementary Services - Stage 1
ETS 300 518, ETSI, September 1994
[6] GSM 2.86, Advice of Charge Supplementary Services - Stage 1
ETS 300 519, ETSI, September 1994
[7] GSM 2.88, Call Barring Supplementary Services - Stage 1
ETS 300 520, ETSI, September 1994
[8] GSM 3.14, Support of Dual Tone Multi Frequency Signalling via the GSM System
ETS 300 532, ETSI, April 1994
[9] GSM 3.40, Technical Realization of the Short Message Service Point-to-Point
ETS 300 536, ETSI, January 1996
[10] GSM 3.41, Technical Realization of Short Message Service Cell Broadcast
ETS 300 537, ETSI, June 1995
[11] GSM 3.81, Line Identification Supplementary Services - Stage 2
ETS 300 542, ETSI, February 1995
[12] GSM 3.82, Call Forwarding Supplementary Services - Stage 2
ETS 300 543, ETSI, February 1995
[13] GSM 3.83, Call Waiting and Call Hold Supplementary Services - Stage 2
ETS 300 544, ETSI, November 1994
[14] GSM 3.84, Multi Party Supplementary Services - Stage 2
ETS 300 545, ETSI, November 1994
[15] GSM 3.85, Closed User Group Supplementary Services - Stage 2
ETS 300 546, ETSI, January 1996
[16] GSM 3.86, Advice of Charge Supplementary Services - Stage 2
ETS 300 547, ETSI, March 1995
[17] GSM 3.88, Call Barring Supplementary Services - Stage 2
ETS 300 548, ETSI, November 1994
[18] GSM 4.01, MS-BSS Interface General Aspects and Principles
ETS 300 550, ETSI, September 1994
[18a] GSM 4.03, MS-BSS Interface Channel Structures and Access Capabilities
ETS 300 552, ETSI, September 1994
[19] GSM 4.05, Data Link Layer General Aspects
ETS 300 554, ETSI, September 1994
[20] GSM 4.06, MS-BSS Interface Data Link Layer Specification
ETS 300 555, ETSI, September 1994
[21] GSM 4.07, Mobile Radio Interface Signalling Layer 3 General Aspects
ETS 300 556, ETSI, February 1995
[22] GSM 4.08, Mobile Radio Interface Layer 3 Specification
ETS 300 557, ETSI, January 1996
[23] GSM 4.10, Mobile Radio Interface Layer 3 Supplementary Services Specification
General Aspects
ETS 300 558, ETSI, February 1995
[24] GSM 4.11, Point-to-Point Short Message Service Support on Mobile Radio Interface
ETS 300 559, ETSI, October 1995
[25] GSM 4.12, Short Message Service Cell Broadcast Support on Mobile Radio Interface
ETS 300 560, ETSI, January 1996
[26] GSM 4.80, Mobile Radio Interface Supplementary Services Specification Formats and Coding
ETS 300 564, ETSI, February 1995

[27] GSM 4.81, Line Identification Supplementary Services - Stage 3
ETS 300 565, ETSI, February 1995
[28] GSM 4.82, Call Forwarding Supplementary Services - Stage 3
ETS 300 566, ETSI, February 1995
[29] GSM 4.83, Call Waiting and Call Hold Supplementary Services - Stage 3
ETS 300 567, ETSI, February 1995
[30] GSM 4.84, Multi Party Supplementary Services - Stage 3
ETS 300 568, ETSI, February 1995
[31] GSM 4.85, Closed User Group Supplementary Services - Stage 3
ETS 300 569, ETSI, February 1995
[32] GSM 4.86, Advice of Charge Supplementary Services - Stage 3
ETS 300 570, ETSI, February 1995
[33] GSM 4.88, Call Barring Supplementary Services - Stage 3
ETS 300 571, ETSI, February 1995
[34] GSM 5.01, Physical Layer on the Radio Path General Description
ETS 300 573, ETSI, October 1995
[35] GSM 5.02, Multiplexing and Multiple Access on the Radio Path
ETS 300 574, ETSI, January 1996
[36] GSM 5.08, Radio Sub-system Link Control
ETS 300 578, ETSI, January 1996
[37] GSM 5.10, Radio Sub-system Synchronisation
ETS 300 579, ETSI, October 1995
[38] Service Access Point MMREG
6147.100.96.100; Condat GmbH
[39] Service Access Point MNCC
6147.101.96.100; Condat GmbH
[40] Service Access Point MNSS
6147.102.96.100; Condat GmbH
[41] Service Access Point MNSMS
6147.103.96.100; Condat GmbH
[42] Service Access Point MMCC
6147.104.97.100; Condat GmbH
[43] Service Access Point MMSS
6147.105.97.100; Condat GmbH
[44] Service Access Point MMSMS
6147.106.97.100; Condat GmbH
[45] Service Access Point RR
6147.107.97.100; Condat GmbH
[46] Service Access Point SIM
6147.108.97.100; Condat GmbH
[47] Service Access Point MPH
6147.109.96.100; Condat GmbH
[48] Service Access Point DL
6147.110.96.100; Condat GmbH
[49] Service Access Point MDL
6147.111.96.100; Condat GmbH
[50] Service Access Point PH
6147.112.97.100; Condat GmbH
[51] Service Access Point MMI
6147.113.96.100; Condat GmbH
[52] Message Sequence Charts CC
6147.200.97.100; Condat GmbH
[53] Message Sequence Charts SS
6147.201.97.100; Condat GmbH
[54] Message Sequence Charts SMS
6147.202.97.100; Condat GmbH
[55] Message Sequence Charts MM
6147.203.97.100; Condat GmbH

[56] Message Sequence Charts RR
6147.204.96.100; Condat GmbH
[57] Message Sequence Charts DL
6147.205.96.100; Condat GmbH
[58] Users Guide
6147.300.96.100; Condat GmbH
[59] Test Specification CC
6147.400.97.100; Condat GmbH
[60] Test Specification SS
6147.401.97.100; Condat GmbH
[61] Test Specification SMS
6147.402.97.100; Condat GmbH
[62] Test Specification MM
6147.403.97.100; Condat GmbH
[63] Test Specification RR
6147.404.97.100; Condat GmbH
[64] Test Specification DL
6147.405.97.100; Condat GmbH
[65] Test Specification CCD
6147.406.97.100; Condat GmbH
[66] SDL Specification CC
6147.500.97.100; Condat GmbH
[67] SDL Specification SS
6147.501.97.100; Condat GmbH
[68] SDL Specification SMS
6147.502.97.100; Condat GmbH
[69] SDL Specification MM
6147.503.97.100; Condat GmbH
[70] SDL Specification RR
6147.504.97.100; Condat GmbH
[71] SDL Specification DL
6147.505.97.100; Condat GmbH
[72] Message Specification CC
6147.600.97.100; Condat GmbH
[73] Message Specification SS
6147.601.97.100; Condat GmbH
[74] Message Specification SMS
6147.602.97.100; Condat GmbH
[75] Message Specification MM
6147.603.97.100; Condat GmbH
[76] Message Specification RR
6147.604.97.100; Condat GmbH
[77] Message Specification DL
6147.605.97.100; Condat GmbH
[78] Technical Documentation CC
6147.700.97.100; Condat GmbH
[79] Technical Documentation SS
6147.701.97.100; Condat GmbH
[80] Technical Documentation SMS
6147.702.97.100; Condat GmbH
[81] Technical Documentation MM
6147.703.97.100; Condat GmbH
[82] Technical Documentation RR
6147.704.97.100; Condat GmbH
[83] Technical Documentation DL
6147.705.97.100; Condat GmbH
[84] Technical Documentation CCD
6147.706.97.100; Condat GmbH

1.2 Abbreviations

AGCH Access Grant Channel
BCCH Broadcast Control Channel
BS Base Station
BSIC Base Station Identification Code
CBCH Cell Broadcast Channel
CBQ Cell Bar Qualify
CC Call Control
CCCH Common Control Channel
CCD Condat Coder Decoder
CKSN Ciphering Key Sequence Number
C/R Command / Response
C1 Path Loss Criterion
C2 Reselection Criterion
DCCH Dedicated Control Channel
DISC Disconnect Frame
DL Data Link Layer
DM Disconnected Mode Frame
EA Extension Bit Address Field
EL Extension Bit Length Field
EMMI Electrical Man Machine Interface
F Final Bit
FACCH Fast Associated Control Channel
FHO Forced Handover
GP Guard Period
GSM Global System for Mobile Communication
HPLMN Home Public Land Mobile Network
I Information Frame
IMEI International Mobile Equipment Identity
IMSI International Mobile Subscriber Identity
Kc Authentication Key
L Length Indicator
LAI Location Area Information
LPD Link Protocol Discriminator
M More Data Bit
MCC Mobile Country Code
MM Mobility Management
MMI Man Machine Interface
MNC Mobile Network Code
MS Mobile Station
NCC National Colour Code
NECI New Establishment Causes included
N(R) Receive Number
N(S) Send Number
OTD Observed Time Difference
P Poll Bit
PCH Paging Channel
PDU Protocol Description Unit
P/F Poll / Final Bit
PL Physical Layer
PLMN Public Land Mobile Network
RACH Random Access Channel
REJ Reject Frame
RNR Receive Not Ready Frame
RR Radio Resource Management
RR Receive Ready Frame
RTD Real Time Difference

SABM Set Asynchronous Balanced Mode
SACCH Slow Associated Control Channel
SAP Service Access Point
SAPI Service Access Point Identifier
SDCCH Slow Dedicated Control Channel
SIM Subscriber Identity Module
SMS Short Message Service
SMSCB Short Message Service Cell Broadcast
SS Supplementary Services
TCH Traffic Channel
TCH/F Traffic Channel Full Rate
TCH/H Traffic Channel Half Rate
TDMA Time Division Multiple Access
TMSI Temporary Mobile Subscriber Identity
UA Unnumbered Acknowledgement Frame
UI Unnumbered Information Frame
VPLMN Visiting Public Land Mobile Network
V(A) Acknowledgement State Variable
V(R) Receive State Variable
V(S) Send State Variable

1.3 Terms

Entity: Program which executes the functions of a layer

Message: A message is a data unit which is transferred between the entities of the same layer (peer-to-peer) of the mobile and infrastructure side. Message is used as a synonym to protocol data unit (PDU). A message may contain several information elements.

Primitive: A primitive is a data unit which is transferred between layers on one component (mobile station or infrastructure). The primitive has an operation code which identifies the primitive and its parameters.

Service Access Point: A Service Access Point is a data interface between two layers on one component (mobile station or infrastructure).

2 Overview

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

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

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

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

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

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

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

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

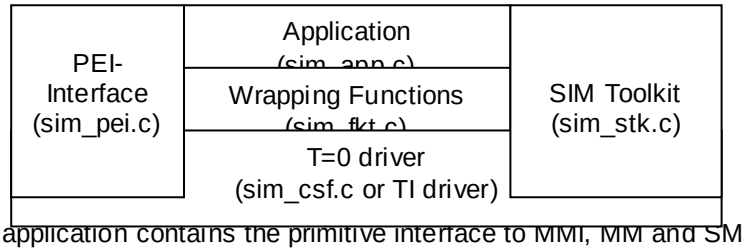
The man machine interface (MMI) is the interface to the user. Normally it is connected with a keypad as input device and a display as output device.

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

The SIM card used by MM, MMI and SMS in several ways. This document describes the services needed for the SIM application.

3 Structure

The SIM software is divided in the following parts:



The SIM application contains the primitive interface to MMI, MM and SMS. It contains several structured services based on the basic functions of the SIM driver.
The wrapping functions in module sim_fkt.c are use to wrap the original SIM driver functions. It contains converting of the original SIM driver error codes.
The interface of the SIM functionality to the target frame is handled in the PEI interface module sim_pei.c. It contains the PEI interface, primitive distribution and timeout handling.
The SIM Driver has a functional interface and covers the functions according to GSM 11.11. This includes the T=0 protocol and the hardware control. In the target version this part of software is delivered by the customer. In the PC-environment this part is the module sim_csf.c. It contains a SIM driver simulation with predefined contents and a lot of test modes for multilayer tests and component tests of the SIM application.
Optional the SIM toolkit functionality can be added to the SIM application. It is located in the sim_stk.c module. If SIM toolkit is integrated, all other modules must be compiled with the constant SIM_TOOLKIT.

3.1 SIM Driver

The SIM driver contains the T=0 protocol and has several function calls mapped to GSM 11.11.

SIM_Init
Initializes the SIM driver. This function must be called before any other access to the SIM.

SIM_Reset
The SIM driver reads the ATR signal and checks whether a SIM card is available.

SIM_Select
This function selects a file in the SIM. After a successful selection the record pointer in a linear fixed file is undefined. The record pointer in a cyclic file shall address the last record which has been updated or increased.

SIM_Status
This function returns information concerning the current directory. A current elementary field is not affected by the STATUS function.

SIM_ReadBinary
This function reads a string of bytes from the current transparent elementary field. This function shall only be performed if the READ access condition for this elementary field is satisfied.

SIM_UpdateBinary
This function updates the current transparent elementary field with a string of bytes. This function shall only be performed if the UPDATE access condition for this elementary field is satisfied. An update can be considered as a replacement of the string already present in the elementary by the string given in the update command.

SIM_ReadRecord
This function reads one complete record in the current linear fixed or cyclic elementary field. The record to be read is described by the modes below. This function shall only be performed if the READ access condition for this elementary field is satisfied. The record pointer shall not be changed by an unsuccessful READ RECORD function. Only the absolute mode is used.

SIM_UpdateRecord
This function updates one complete record in the current linear fixed or cyclic elementary field. This function shall only be performed if the UPDATE access condition for this elementary field is satisfied. The UPDATE can be considered as a replacement of the relevant record data of the elementary field by the record data given in the command. The record pointer shall not be changed by an unsuccessful UPDATE RECORD function. Only the absolute mode is used.

SIM_Increase
This function adds the value given by the mobile station to the value of the last increased/updated record of the current cyclic elementary field, and stores the result into the oldest record. The record pointer is set to this record and this record becomes record number 1. This function shall be used only if this elementary field has an INCREASE access condition assigned and this condition is fulfilled (see bytes 8 and 10 in the response parameters/data of the current elementary field). The SIM shall not perform the increase if the result would exceed the maximum value of the record (represented by all bytes set to 'FF').

SIM_VerifyCHV
This function verifies the CHV presented by the mobile station by comparing it with the relevant one stored in the SIM. The verification process is subject to the following conditions being fulfilled:

- CHV is not disabled;
- CHV is not blocked.

If the access condition for a function to be performed on the last selected file is CHV1 or CHV2, then a successful verification of the relevant CHV is required prior to the use of the function on this file unless the CHV is disabled.
If the CHV presented is correct, the number of remaining CHV attempts for that CHV shall be reset to its initial value 3.
If the CHV presented is false, the number of remaining CHV attempts for that CHV shall be decremented. After 3 consecutive false CHV presentations, not necessarily in the same card session, the respective CHV shall be blocked and the access condition can never be fulfilled until the UNBLOCK CHV function has been successfully performed on the respective CHV.

SIM_ChangeCHV
This function assigns a new value to the relevant CHV subject to the following conditions being fulfilled:

- CHV is not disabled;
- CHV is not blocked.

The old and new CHV shall be presented.
If the old CHV presented is correct, the number of remaining CHV attempts for that CHV shall be reset to its initial value 3 and the new value for the CHV becomes valid.
If the old CHV presented is false, the number of remaining CHV attempts for that CHV shall be decremented and the value of the CHV is unchanged. After 3 consecutive false CHV presentations, not necessarily in the same card session, the respective CHV shall be blocked and the access condition can never be fulfilled until the UNBLOCK CHV function has been performed successfully on the respective CHV.

SIM_DisableCHV
This function may only be applied to CHV1. The successful execution of this function has the effect that files protected by CHV1 are now accessible as if they were marked "ALWAYS". The function DISABLE CHV shall not be executed by the SIM when CHV1 is already disabled or blocked.
If the CHV1 presented is correct, the number of remaining CHV1 attempts shall be reset to its initial value 3 and CHV1 shall be disabled.
If the CHV1 presented is false, the number of remaining CHV1 attempts shall be decremented and CHV1 remains enabled. After 3 consecutive false CHV1 presentations, not necessarily in the same card session, CHV1 shall be blocked and the access condition can never be fulfilled until the UNBLOCK CHV function has been successfully performed on CHV1.

SIM_EnableCHV
This function may only be applied to CHV1. It is the reverse function of DISABLE CHV. The function ENABLE CHV shall not be executed by the SIM when CHV1 is already enabled or blocked.
If the CHV1 presented is correct, the number of remaining CHV1 attempts shall be reset to its initial value 3 and CHV1 shall be enabled.
If the CHV1 presented is false, the number of remaining CHV1 attempts shall be decremented and CHV1 remains disabled. After 3 consecutive false CHV1 presentations, not necessarily in the same card session, CHV1 shall be blocked and the access condition can never be fulfilled until the UNBLOCK CHV function has been successfully performed on CHV1.

SIM_UnblockCHV
This function unblocks a CHV which has been blocked by 3 consecutive wrong CHV presentations. This function may be performed whether or not the relevant CHV is blocked.
If the UNBLOCK CHV presented is correct, the value of the CHV, presented together with the UNBLOCK CHV, is assigned to that CHV, the number of remaining UNBLOCK CHV attempts for that UNBLOCK CHV is reset to its initial value 10 and the number of remaining CHV attempts for that CHV is reset to its initial value 3. After a successful unblocking attempt the CHV is enabled and the relevant access condition level is satisfied.
If the presented UNBLOCK CHV is false, the number of remaining UNBLOCK CHV attempts for that UNBLOCK CHV shall be decremented. After 10 consecutive false UNBLOCK CHV presentations, not necessarily in the same card session, the respective UNBLOCK CHV shall be blocked. A false UNBLOCK CHV shall have no effect on the status of the respective CHV itself.

SIM_Invalidate
This function invalidates the current elementary field. After an INVALIDATE function the respective flag in the file status shall be changed accordingly. This function shall only be performed if the INVALIDATE access condition for the current elementary field is satisfied.
An invalidated file shall no longer be available within the application for any function except for the SELECT and the REHABILITATE functions.

SIM_Rehabilitate
This function rehabilitates the invalidated current elementary field. After a REHABILITATE function the respective flag in the file status shall be changed accordingly. This function shall only be performed if the REHABILITATE access condition for the current elementary field is satisfied.

SIM_RunGSMAlgo
This function is used during the procedure for authenticating the SIM to a GSM network and to calculate a cipher key. The card runs the specified algorithms A3 and A8 using a 16 byte random number and the subscriber authentication key Ki, which is stored in the SIM. The function returns the calculated response SRES and the cipher key Kc.

The function shall not be executable unless DF_{GSM} has been selected as the Current Directory and a successful CHV1 verification procedure has been performed.
The contents of Kc shall be presented to algorithm A5 by the mobile station in its full 64 bit format as delivered by the SIM.

SIM_GetResponse
The response data depends on the preceding command. Response data is available after the commands RUN GSM ALGORITHM, SEEK (type 2), SELECT, and INCREASE. If the command GET RESPONSE is executed, it is required that it is executed immediately after the command it is related to (no other command shall come between the command/response pair and the command GET RESPONSE). If the sequence is not respected, the SIM shall send the status information "technical problem with no diagnostic given" as a reaction to the GET RESPONSE.
Since the master file is implicitly selected after activation of the SIM, GET RESPONSE is also allowed as the first command after activation.

SIM_PowerOff
The driver call is used to switch off the SIM driver during power off.

SIM_TerminalProfile (SIM Toolkit only)
This function is used by the ME to transmit to the SIM its capabilities concerning the SIM Application Toolkit functionality.

SIM_TerminalResponse (SIM Toolkit only)
This function is used to transfer from the ME to the SIM the response to a previously fetched SIM Application Toolkit command.

SIM_Fetch (SIM Toolkit only)
This function is used to transfer an Application Toolkit command from the SIM to the ME.

SIM_Envelope (SIM Toolkit only)
This function is used to transfer data to the SIM Application Toolkit applications in the SIM.

4 General Aspects

The Application has a primitive interface to the protocol stack. It uses one or a combination of SIM driver calls for the requested services by the protocol stack. It is mapping the result codes of the SIM driver and handles error conditions.

4.1 Error Causes

Each driver call has a result buffer. The two last bytes of the result buffer contains the error variables SW1 and SW2. The following table defines the mapping of this variables to the error codes of the SIM application:

SW 1	SW 2	Error Code	Remark
0x90	-	No error	
0x94	0,4	Unknown data field	
	2	Invalid offset	
	8	Invalid access	
0x98	2,4,8	Invalid PIN	See table 1
	0x40	Invalid PUK	See table 2
	0x50	Max value	
0x91	-	No error Fatal error	If proactive SIM card SW 2 Bytes Proactive SIM data available If no proactive SIM card
0x9F	-	No error	SW 2 Byte Response data available
ELSE	-	Fatal error	

If an invalid PIN is detected the error code depends on the last requested PIN number according to table 1:

Last requested PIN number	Error code
None	Invalid PIN 1
PIN 1	Invalid PIN 1
PIN 2	Invalid PIN 2
PUK 1	Invalid PUK 1
PUK 2	Invalid PUK 2

If an invalid PUK (unblocking key) is detected the error code depends on the last requested PIN number according to table 2:

Last requested PIN number	Error code
None	Invalid PUK 1
PIN 1	Invalid PUK 1
PIN 2	Invalid PUK 2
PUK 1	Invalid card
PUK 2	Invalid card

4.2 Access Conditions

For each SIM field the GSM 11.11 defines access conditions for read and update functions. The following access conditions are possible:

ALWAYS	The access to the SIM field is possible without restrictions.
PIN 1	The access to the SIM field is possible after entering PIN 1. This is normally done after Power on.
PIN 2	The access to the SIM field is possible after entering PIN 2.
PIN 1 or 2	The access condition must be detemined by the SIM application (using GET RESPONSE). The needed PIN number is signalled to the MMI. The use of PIN 1 or 2 is a choice of the SIM card manufacturer.
NEVER	The access to the SIM field is not allowed.

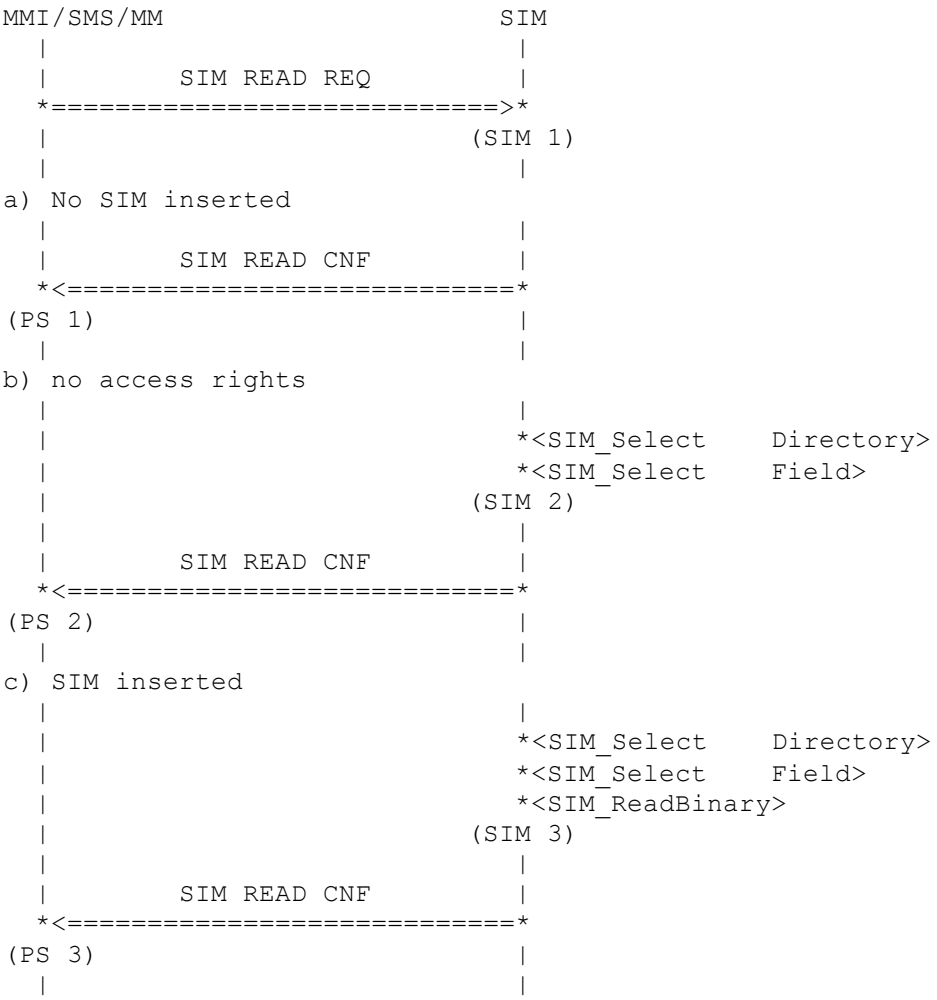
The following access conditions are used:

SIM field	Read Operation	Update Operation
Preferred Language	ALWAYS	PIN 1
Authentication Parameter Kc	PIN 1	PIN 1
Cell Broadcast Message Identifier	PIN 1	PIN 1
BCCH information	PIN 1	PIN 1
Forbidden PLMN list	PIN 1	PIN 1
Preferred PLMN list	PIN 1	PIN 1
Automatic Answer for eMLPP Service	PIN 1	PIN 1
Cell broadcast message identifier range selection	PIN 1	PIN 1
Short messages	PIN 1	PIN 1
Capability configuration parameters	PIN 1	PIN 1
MSISDN	PIN 1	PIN 1
Short message service parameters	PIN 1	PIN 1
SMS status	PIN 1	PIN 1
Last number dialled	PIN 1	PIN 1
Extension1	PIN 1	PIN 1
Location information	PIN 1	PIN 1
Abbreviated dialling numbers	PIN 1	PIN 1
ACM maximum value	PIN 1	PIN 1 or 2
Price per unit and currency table	PIN 1	PIN 1 or 2
Accumulated call meter	PIN 1	PIN 1 or 2

Service Provider Name	ALWAYS	NEVER
Phase identification	ALWAYS	NEVER
Administrative data	ALWAYS	NEVER
Emergency Call Codes	ALWAYS	NEVER
Fixed dialling numbers	PIN 1	PIN 2
Extension 2	PIN 1	PIN 2
Extension 4	PIN 1	PIN 2
Barred dialling numbers	PIN 1	PIN 2
Else	PIN 1	NEVER

5 Basic Functions

5.1 Read Binary File



(SIM 1)

A protocol stack component requests reading of a data field.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause SIM_FATAL_ERROR.

(SIM 2)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded. The access rights are checked. If it is not allowed to read the field the reading is not processed.

(PS 2)

The result of the reading attempt is forwarded to the requesting procotol stack component (error cause SIM_INVALID_ACCESS).

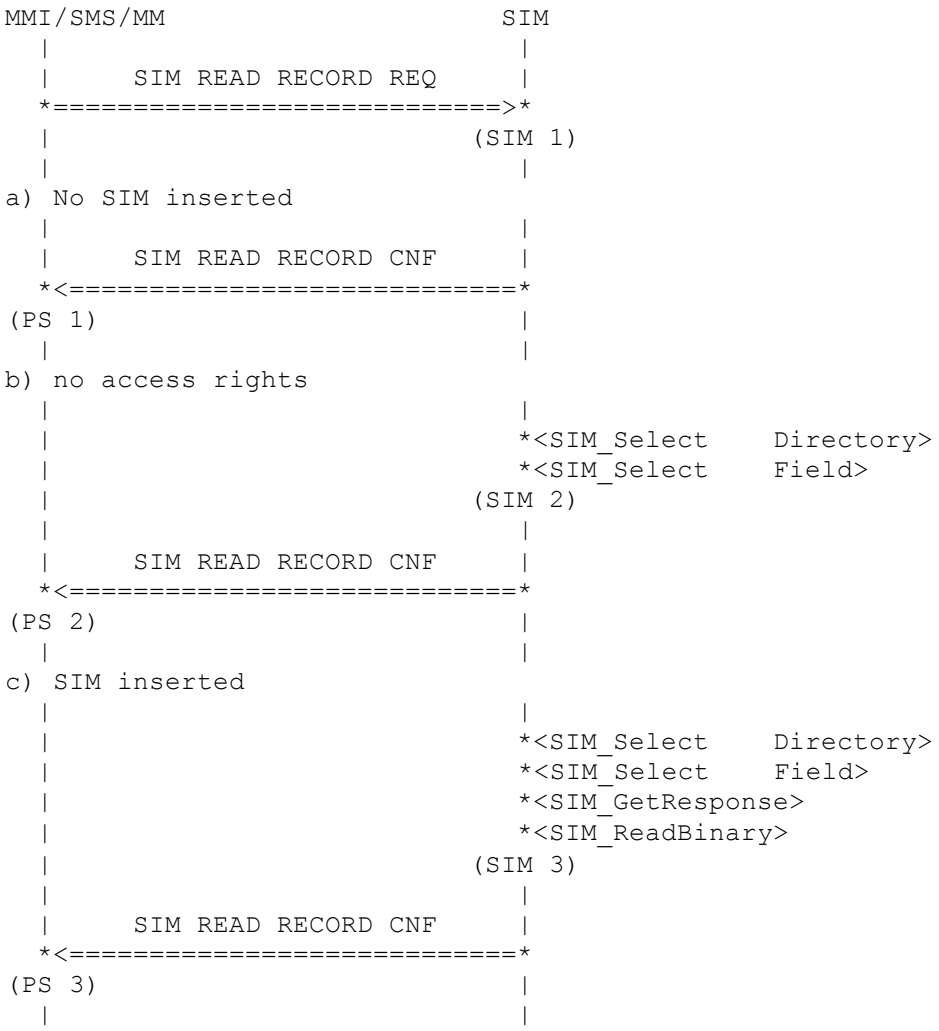
(SIM 3)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded. After successfull selection the content of the field is read with the SIM driver call SIM_ReadBinary.

(PS 3)

The result of the reading attempt is forwarded to the requesting procotol stack component.

5.2 Read Record File



(SIM 1)

A protocol stack component requests reading of a record of a cyclic or linear fixed data field.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause SIM_FATAL_ERROR.

(SIM 2)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded. The access rights are checked. If it is not allowed to read the field the reading is not processed.

(PS 2)

The result of the reading attempt is forwarded to the requesting procotol stack component (error cause SIM_INVALID_ACCESS).

(SIM 3)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded.

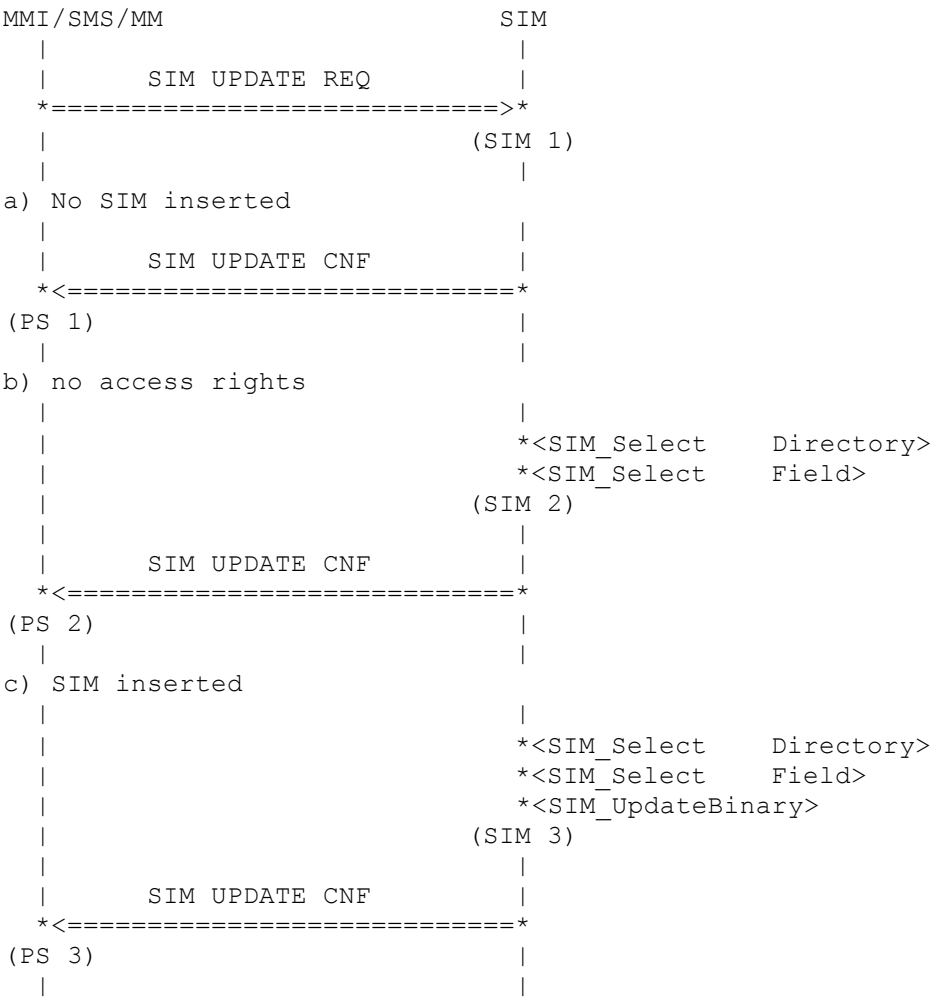
If the record number is one, that means the first record, additionally the maximum number of records will be calculated. Therefore the SIM_Get_Response driver call is used. The maximum number of records will be calculated and is part of the response to the requesting component. Else the maximum number of records will not be calculated and is set to zero in the response.

After successfull selection the content of the record is read with the SIM driver call SIM_ReadRecord.

(PS 3)

The result of the reading attempt is forwarded to the requesting procotol stack component.

5.3 Update Binary File



(SIM 1)

A protocol stack component requests updating of a data field.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause SIM_FATAL_ERROR.

(SIM 2)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded. The access rights are checked. If it is not allowed to update the field the updating is not processed.

(PS 2)

The result of the updating attempt is forwarded to the requesting procotol stack component (error cause SIM_INVALID_ACCESS).

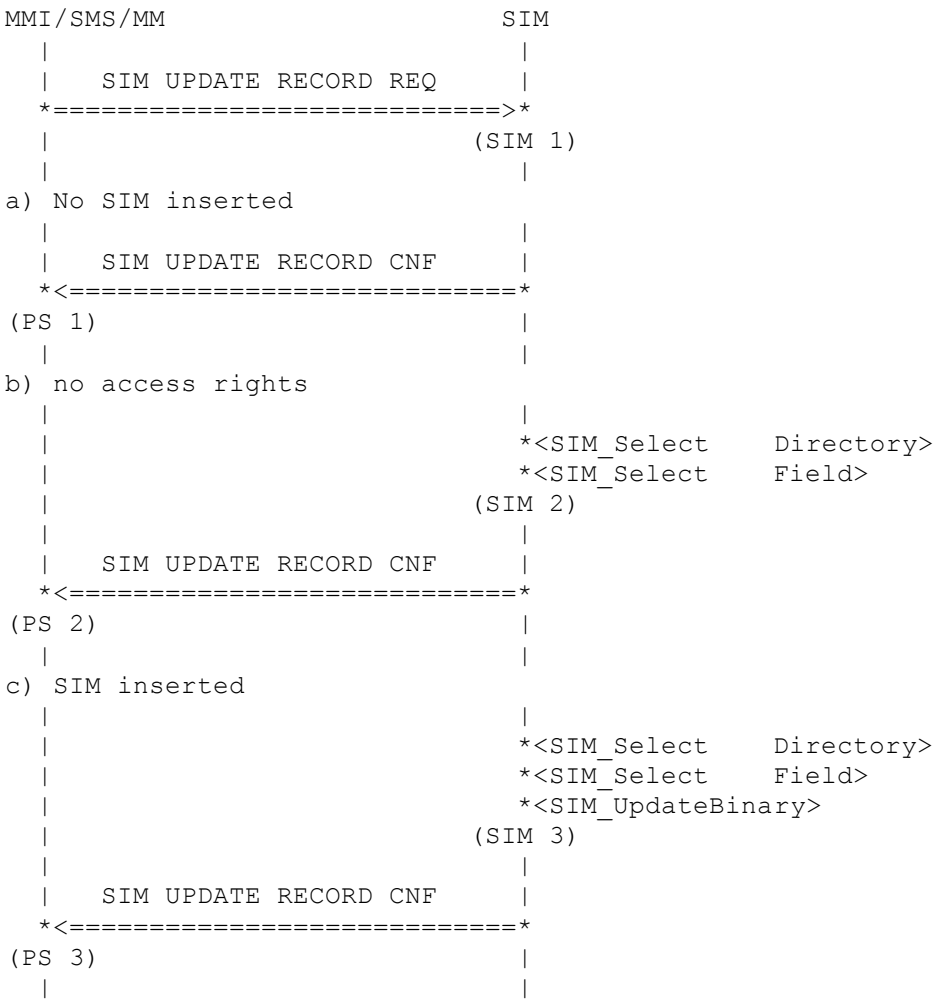
(SIM 3)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded. After successfull selection the content of the field is updated with the SIM driver call SIM_UpdateBinary.

(PS 3)

The result of the updating attempt is forwarded to the requesting procotol stack component.

5.4 Update Record File



(SIM 1)

A protocol stack component requests update of a record of a cyclic or linear fixed data field.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause SIM_FATAL_ERROR.

(SIM 2)

The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded. The access rights are checked. If it is not allowed to update the field the updating is not processed.

(PS 2)

The result of the updating attempt is forwarded to the requesting procotol stack component (error cause SIM_INVALID_ACCESS).

(SIM 3)

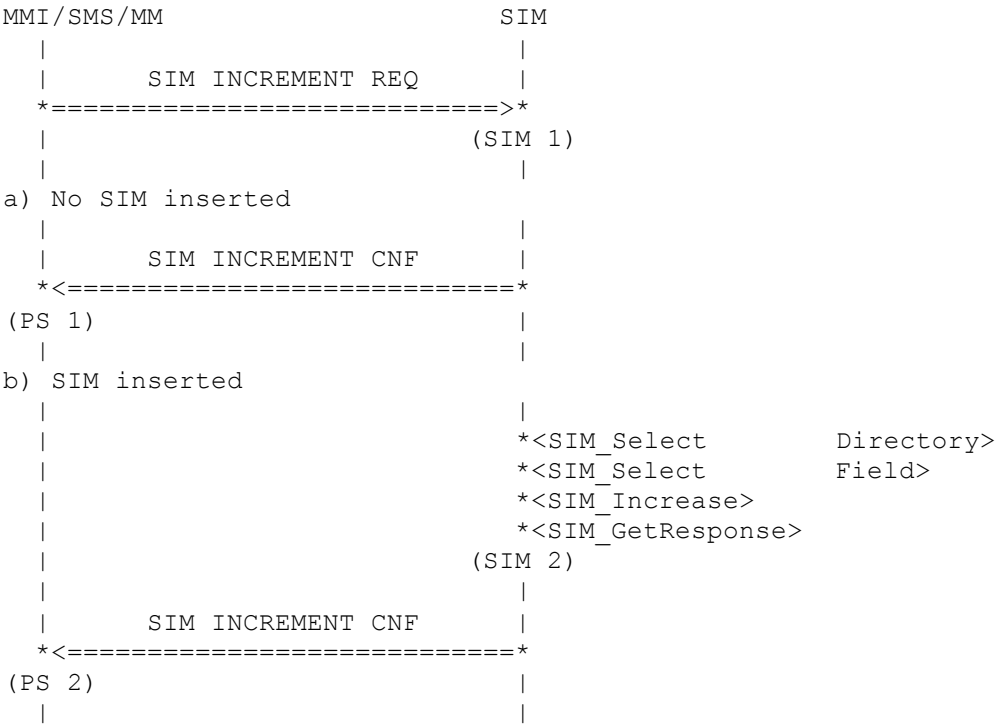
The requested data field is selected using the SIM driver calls SIM_Select. If the actual directory differs from the needed directory a SIM_Select is carried out for the needed directory. If the actual field differs from the needed field, a SIM_Select is carried out for the needed field. The dependency between field and directory is hardcoded.

After successfull selection the content of the record is updated with the SIM driver call SIM_UpdateRecord.

(PS 3)

The result of the updating attempt is forwarded to the requesting procotol stack component.

5.5 Increment



(SIM 1)

One protocol stack component requests incrementing of a cyclic data field.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause.

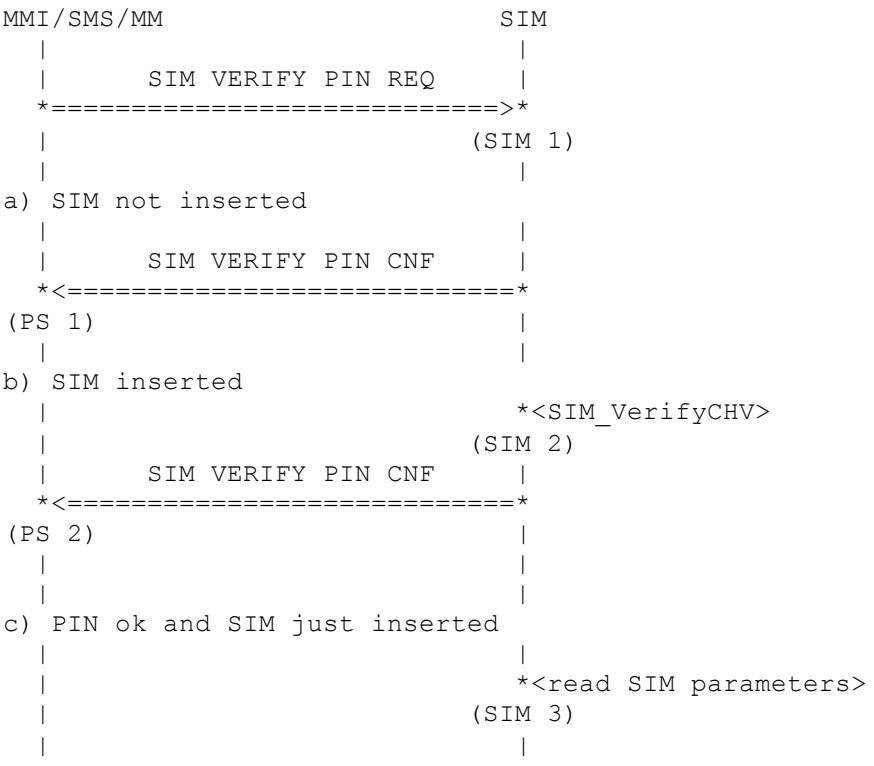
(SIM 2)

The requested data field is selected using the SIM driver calls SIM_Select. After successfull selection the field is incremented using the SIM driver call SIM_Increase. The result of the operation is requested by the SIM driver call SIM_GetResponse.

(PS 2)

The result of the operation is forwarded to the requesting procotol stack component.

5.6 Verify PIN



(SIM 1)

MMI requests verification of the PIN.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause.

(SIM 2)

The PIN is verified using the SIM driver call SIM_VerifyCHV.

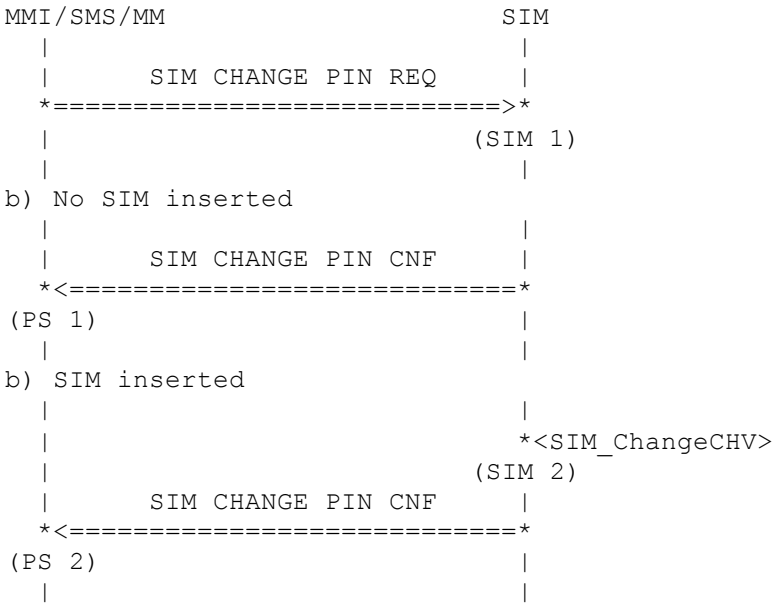
(PS 2)

The result of the operation is forwarded to the requesting procotol stack component.

(SIM 3)

If the PIN verify is successfull and the SIM was just inserted, the SIM application starts reading the SIM parameters as described into another chapter of this specification.

5.7 Change PIN



(SIM 1)

MMI requests the change of a PIN.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause.

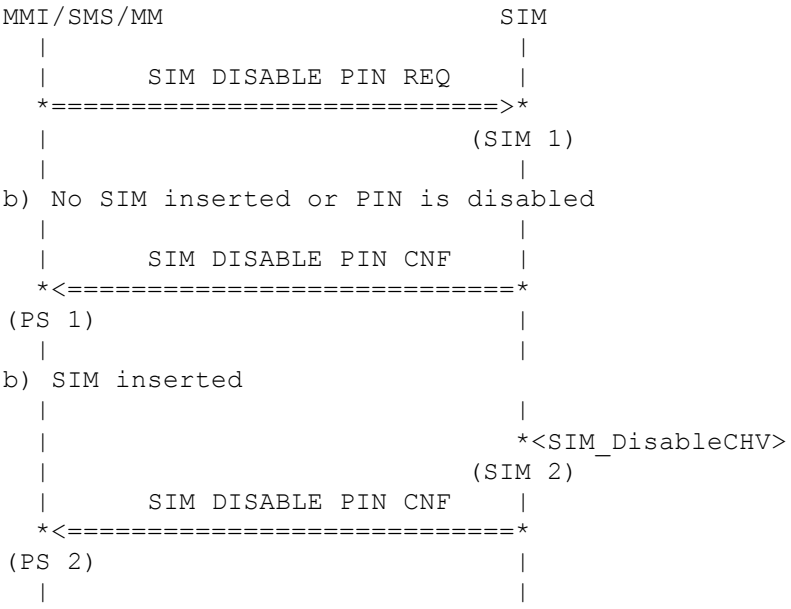
(SIM 2)

The PIN is changed using the SIM driver call SIM_ChangeCHV.

(PS 2)

The result of the operation is forwarded to the requesting procotol stack component.

5.8 Disable PIN



(SIM 1)

MMI requests disabling of a PIN.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause. If a SIM is inserted and the card is already disabled, the procedure ends successful immediately.

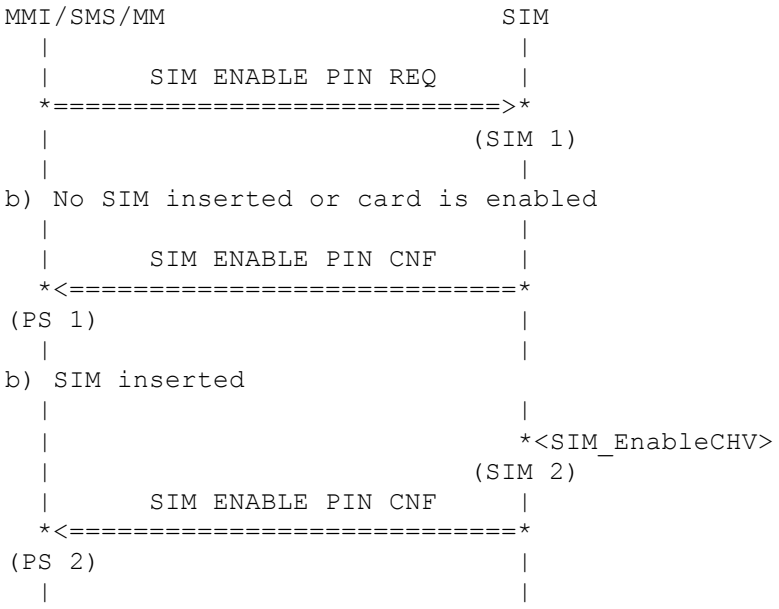
(SIM 2)

The PIN is disabled using the SIM driver call SIM_DisableCHV.

(PS 2)

The result of the operation is forwarded to the requesting procotol stack component.

5.9 Enable PIN



(SIM 1)

MMI requests enabling of a PIN.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause. If a SIM card is inserted and the card is enabled, the procedure ends successful immediately.

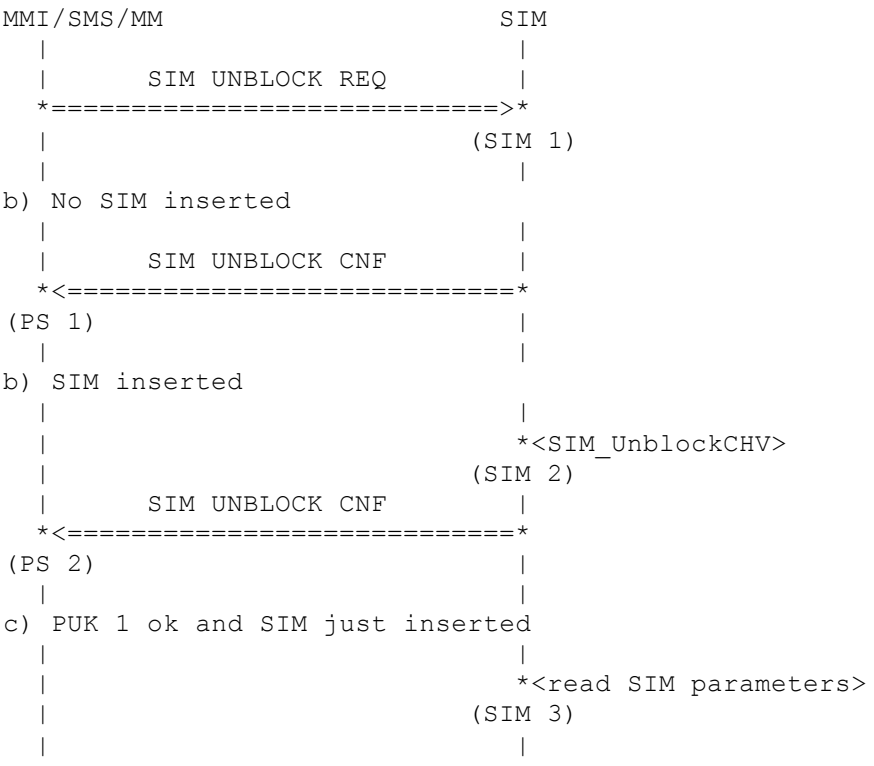
(SIM 2)

The PIN is enabled using the SIM driver call SIM_EnableCHV.

(PS 2)

The result of the operation is forwarded to the requesting procotol stack component.

5.10 Unblock PIN



(SIM 1)

MMI requests unblocking of a PIN with the PUK.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause.

(SIM 2)

The PIN is unblocked using the SIM driver call SIM_UnblockCHV.

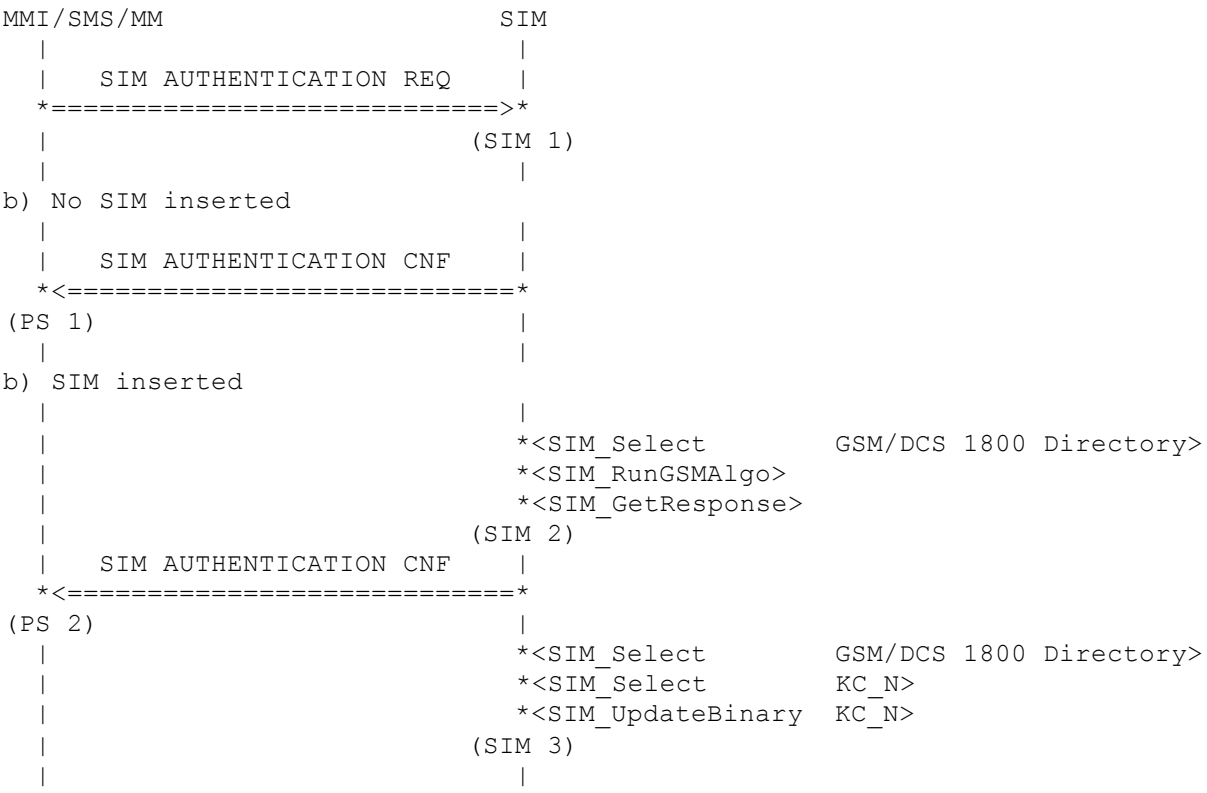
(PS 2)

The result of the operation is forwarded to the requesting procotol stack component.

(SIM 3)

If the PUK verify is successfull and the SIM was just inserted, the SIM application starts reading the SIM parameters as described into another chapter of this specification.

5.11 Authentication



(SIM 1)

MM requests running of the GSM algorithm.

(PS 1)

If no SIM is inserted, the SIM application rejects this request with an error cause.

(SIM 2)

The GSM directory is selected by the SIM driver call SIM_Select. After successfull selection the GSM algoritm is triggered using the SIM driver call SIM_RunGSMAlgo. The result parameters are requested by the SIM driver call SIM_GetResponse.

(PS 2)

The result of the operation (authentication parameter Sres and Kc) is forwarded to the requesting procotol stack component.

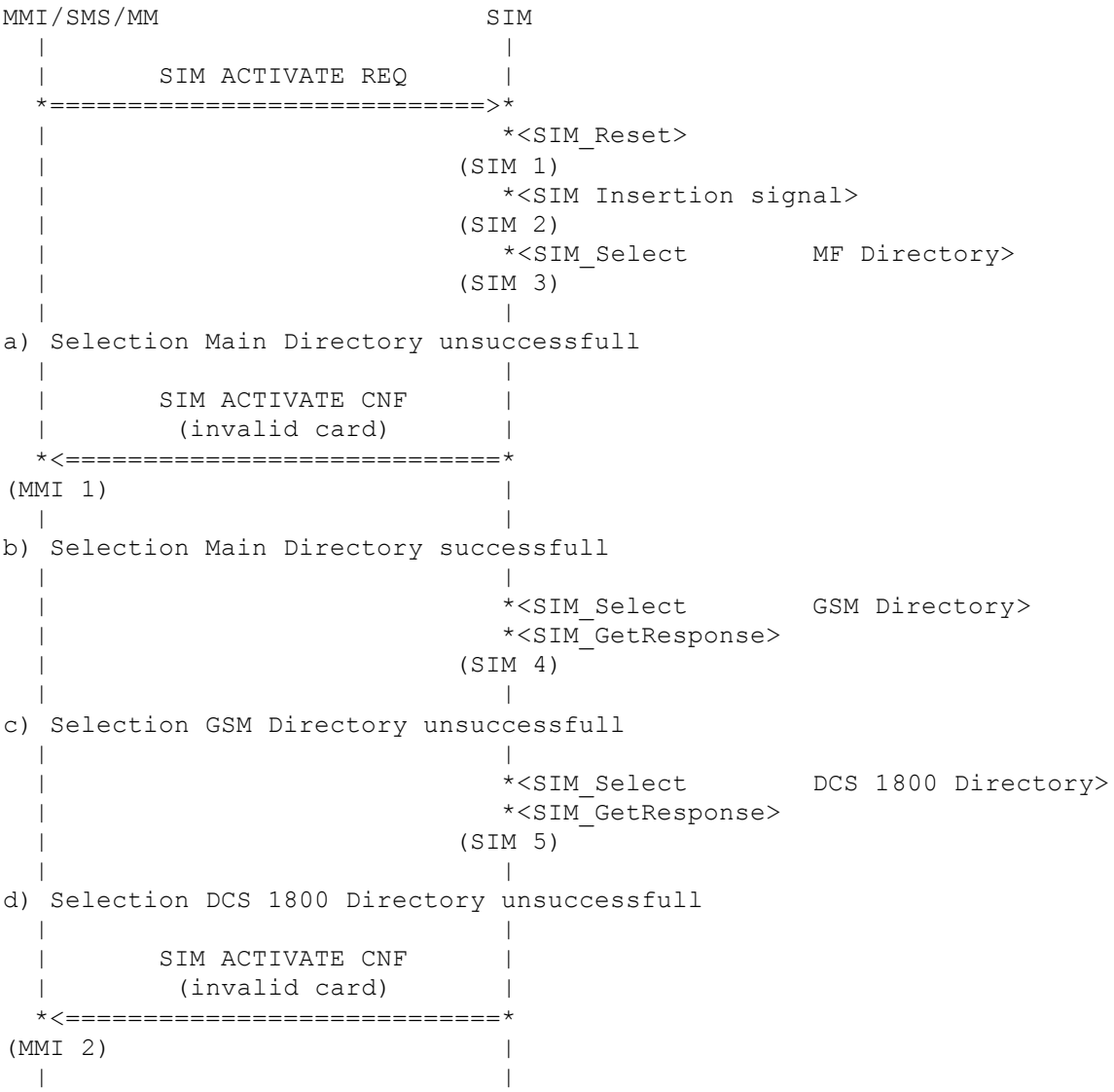
(SIM 3)

The calculated Kc value is stored on the SIM card. Therefore the KC_N data field is selected using the SIM driver calls SIM_Select and written by the SIM driver call SIM_UpdateBinary.

6 Structured Procedures

6.1 SIM Insertion

6.1.1 Access to Main Directory



(SIM 1)

MMI sends the primitive to start activation of the SIM card. The SIM driver is resetted. The callback functions for SIM insertion and SIM remove are installed.

(SIM 2)

The SIM driver has detected a SIM card and the callback function for SIM insertion is called by the SIM driver.

(SIM 3)

The SIM application selectes the Main File Directory.

(MMI 1)

If is it impossible to select the Main File Directory, the SIM card is invalid and the MMI is informed.

(SIM 4)

After successful selection of the Main File Directory the SIM application selects the GSM Directory. If this is successful it reads the response data of this field.

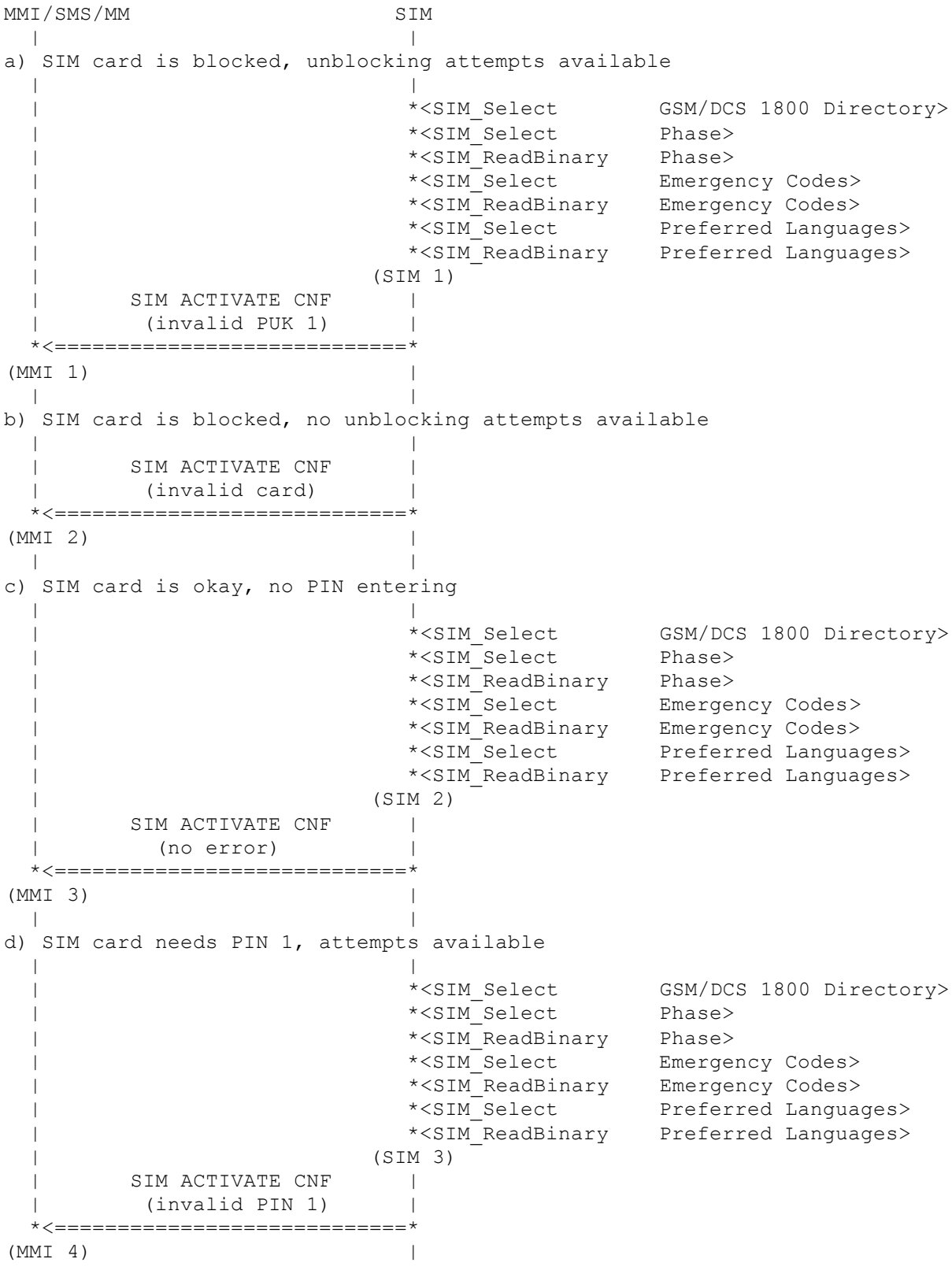
(SIM 5)

If the selection of the GSM directory is unsuccessful, the SIM application starts the selection of the DCS 1800 directory instead. This is done for backward compatibility for Phase 1 cards for DCS 1800.

(MMI 2)

If also the selection of the DCS 1800 directory fails the SIM card is invalid and this is signalled to the MMI.

6.1.2 Check of PIN / PUK counter



(SIM 1)

The response data from selecting the GSM / DCS 1800 directory contains the blocking status and the number of remaining PIN / PUK attempts. If the SIM card is blocked, but unblocking attempts are still available, the SIM application reads the

elementary fields PHASE, EMERGENCY CODES and PREFERRED LANGUAGES. The field PHASE must be available, the two other fields are optional.

(MMI 1)

The MMI is informed that the SIM card is blocked with the cause Invalid PUK 1. This is the trigger for entering PUK 1 in the MMI. The primitive to MMI contains additional the parameters read from the SIM card.

(MMI 2)

The response data from selecting the GSM / DCS 1800 directory contains the blocking status and the number of remaining PIN / PUK attempts. The SIM card is blocked and no unblocking attempts are available. The SIM card is invalid. This is signalled to MMI.

(SIM 2)

The response data from selecting the GSM / DCS 1800 directory contains the PIN / PUK status and the number of remaining PIN / PUK attempts. If the SIM card is unblocked and needs no PIN entering, the SIM application reads the elementary fields PHASE, EMERGENCY CODES and PREFERRED LANGUAGES. The field PHASE must be available, the two other fields are optional.

(MMI 3)

The MMI is informed that the SIM card is okay and no PIN entering is needed with the cause No Error. This is the trigger for starting registration in the MMI. The primitive to MMI contains additional the parameters read from the SIM card.

(SIM 3)

The response data from selecting the GSM / DCS 1800 directory contains the PIN / PUK status and the number of remaining PIN / PUK attempts. If the SIM card is unblocked but PIN 1 entering is needed, the SIM application reads the elementary fields PHASE, EMERGENCY CODES and PREFERRED LANGUAGES. The field PHASE must be available, the two other fields are optional.

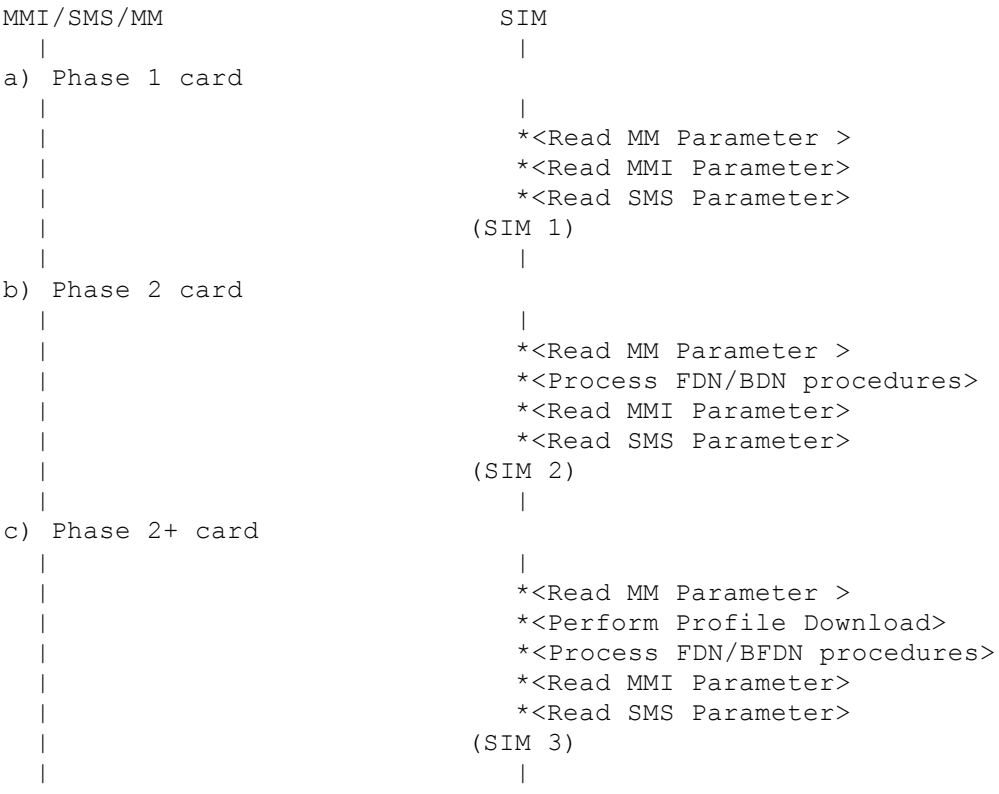
(MMI 4)

The MMI is informed that the SIM card is okay and PIN 1 entering is needed with the cause Invalid Pin 1. This is the trigger for requesting PIN 1 in the MMI. The primitive to MMI contains additional the parameters read from the SIM card.

6.1.3 Reading of SIM Parameters

The SIM parameters are read

- after activation, if PIN entering is disabled, or
- after successful verification of PIN1 or
- after successful unblocking the SIM card with PUK 1



(SIM 1)

If the SIM card is a phase 1 card, the SIM application reads the MM parameter and forwards them to MM, reads the MMI parameter and forwards them to MMI and reads the SMS parameter and forwards them to SMS.

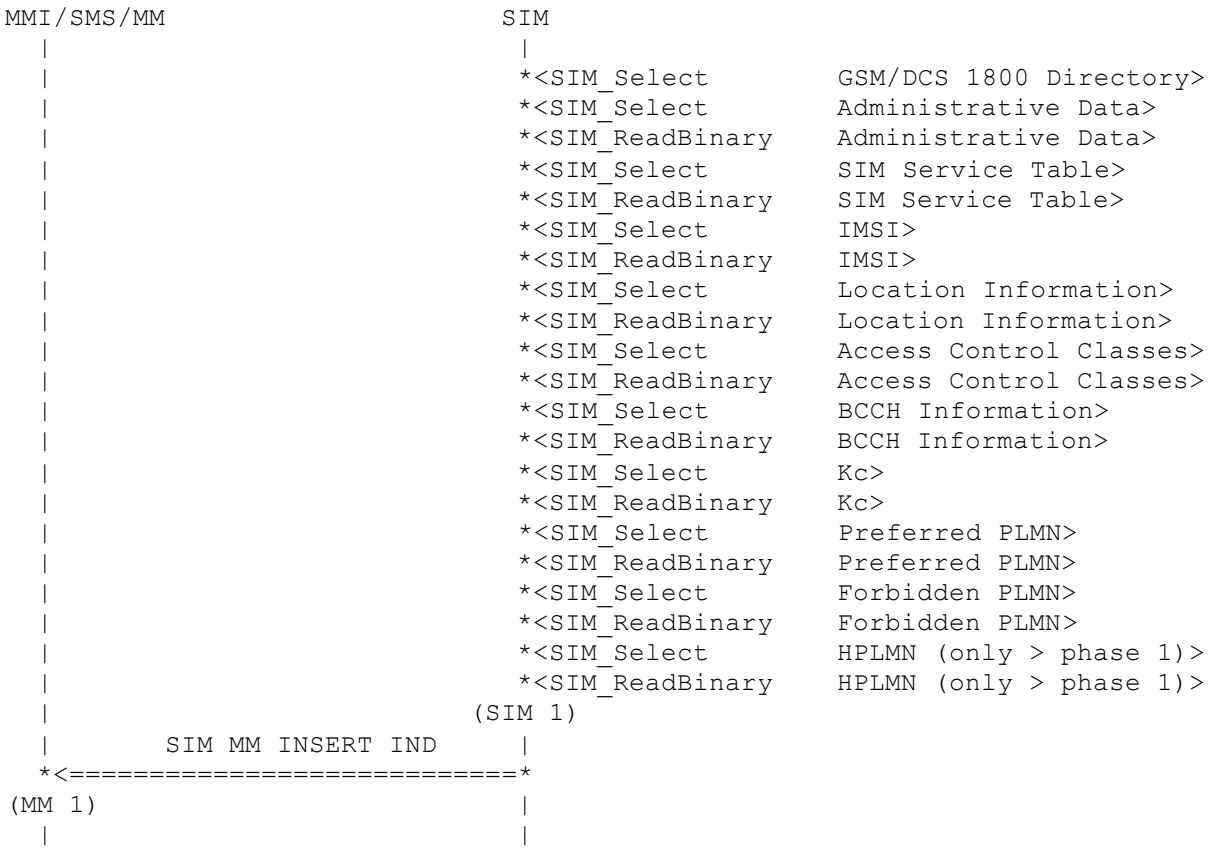
(SIM 2)

If the SIM card is a phase 2 card, the SIM application reads the MM parameter and forwards them to MM. Then it checks the capabilities of restricted or unrestricted operation as described in the chapter Process FDN / BDN procedures. Then it reads the MMI parameter and forwards them to MMI and reads the SMS parameter and forwards them to SMS.

(SIM 3)

A phase 2+ card processes the same procedures like a phase 2 card. It additionally process the profile download for SIM toolkit.

6.1.4 Read MM Parameter



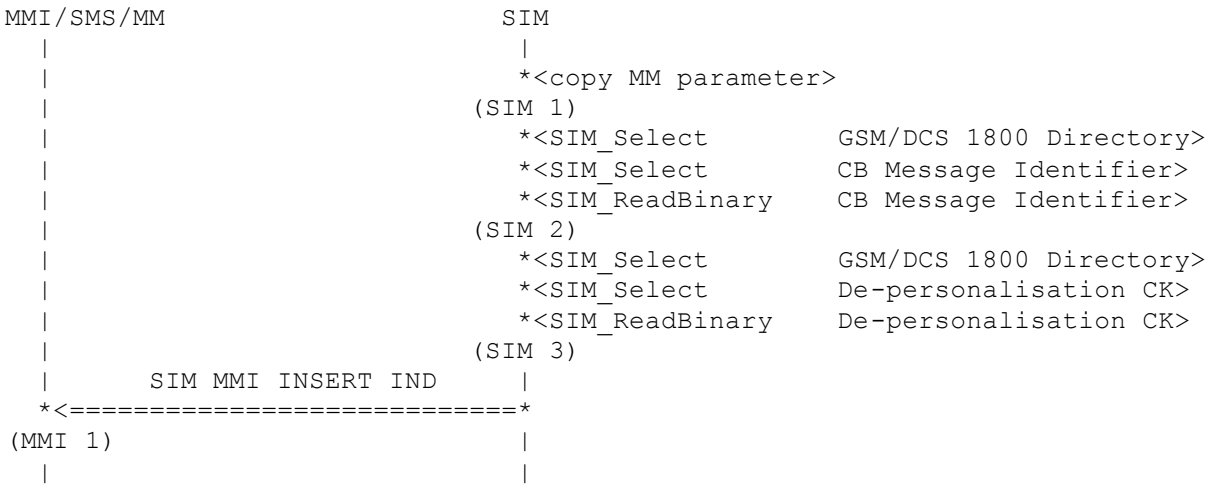
(SIM 1)

MM needs the content of several fields of the SIM card. This data are used for registration purposes in the protocol stack. The SIM application reads the following fields: ADMINISTRATIVE DATA, SIM SERVICE TABLE, IMSI, LOCATION INFORMATION, ACCESS CONTROL CLASSES, BCCH INFORMATION, KC, PREFERRED PLMN LIST, FORBIDDEN PLMN and HPLMN if the SIM card is not a phase 1 card.

(MM 1)

All parameters are filled in a SIM MM INSERT IND primitive and forwarded to MM.

6.1.5 Read MMI Parameter



(SIM 1)

Some of the MM parameters are also needed by MMI. The SIM application copies this parameters to the SIM MMI INSERT IND primitive.

(SIM 2)

If the SIM service table indicates that service 14 is activated and allocated, the SIM application selects the cell broadcast message identifier field and reads the content.

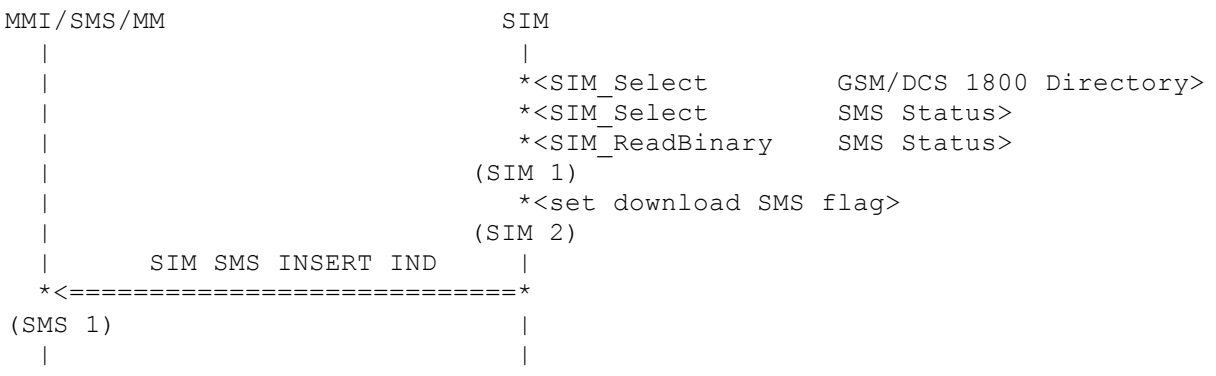
(SIM 3)

If the SIM service table indicates that service 33 is activated and allocated, the SIM application selects the de-personalisation key field and reads the content.

(MMI 1)

The content of the several datafields relevant to MMI are send with the primitive SIM MMI INSERT IND to MMI.

6.1.6 Read SMS Parameter



(SIM 1)

The SIM application reads the SMS status field for the initialization of SMS. This field contains the last used message reference and a flag whether short message memory is available on the SIM card.

(SIM 2)

The download SMS flag indicates SMS whether the SIM card supports Data Download to the SIM card via SMS. This is only supported if

- The SIM card is a phase 2+ card,
- SIM toolkit is supported by the mobile and
- Service 26 of the SIM service table is allocated and activated.

(SMS 1)

The parameters are forwarded to SMS.

6.1.7 FDN/BDN Procedures

This chapter describes the implementation of FDN / BDN checking procedures according to annex B of GSM 11.11.

The result of this procedure is the actual operation mode of the mobile. The following operation modes are possible:

ADN ENABLED	The SIM card has enabled unrestricted operation (abbreviated dialling). Fixed dialling and barred dialling are not available or de-activated.
ADN/BDN ENABLED	The SIM card has enabled unrestricted operation (abbreviated dialling). Fixed dialling is not available or de-activated. Additionally barred dialling is activated.
FDN ENABLED	The SIM card has enabled restricted operation (fixed dialling). Abbreviated dialling and barred dialling are not available or de-activated.
FDN/BDN ENABLED	The SIM card has enabled restricted operation (fixed dialling). Abbreviated dialling is not available or de-activated. Additionally barred dialling is activated.
NO OPERATION	The SIM card is invalid and no operation is allowed.

The first check is IMSI LOCI VALIDATION. Therefore the invalidation flag of the response after selection of IMSI and Location Information is checked. The result is TRUE or FALSE. If the result is TRUE, the operation mode of the SIM card is ADN ENABLED.

If this check fails then the rest of the procedure must be carried out as described in the following:

The BDN and FDN capability are checked.

The BDN CAPABILITY REQUEST means whether the SIM card supports barred dialling and whether it is enabled or disabled. The following results are possible NO BDN SIM, BDN ENABLED or BDN DISABLED. If service 31 of the SIM service table is unequal allocated and activates the result is NO BDN SIM. Else the BDN field is selected and the response data are read. Any failure during this procedure leads to the result NO BDN SIM. Then the validation flag of the response data is checked. If the validation flag is 0 the result is BDN DISABLED else BDN ENABLED.

The FDN CAPABILITY REQUEST means whether the SIM card supports fixed dialling and whether it is enabled or disabled. The following results are possible NO FDN SIM, FDN ENABLED or FDN DISABLED. If service 3 of the SIM service table is unequal allocated and activates the result is NO FDN SIM. If the abbreviated dialling field (service 2) is not allocated and activated the result is FDN ENABLED. Else the ADN field is selected and the response data are read. Any failure during this procedure leads to the result FDN ENABLED. Then the validation flag of the response data is checked. If the validation flag is 0 the result is FDN ENABLED else FDN DISABLED.

During checking the operation mode it is may be necessary to try rehabilitation of IMSI and Location Information. Therefore the IMSI field is selected and rehabilitated (using SIM driver call SIM_Rehabilitate). After this the same is carried out for the field Location Information. Any failure during this procedure leads to the result FALSE. Else the result of this procedure is TRUE.

The resulting operation mode for mobile with support of call control with SIM toolkit is given by the following rules:

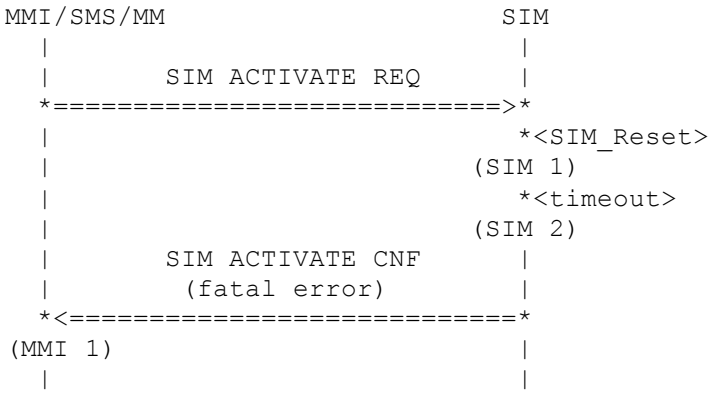
- If the mobile doesn´t support BDN, but the SIM card supports BDN, the resulting operation mode is NO OPERATION.
- If the mobile doesn´t support FDN, but the SIM card supports FDN, the resulting operation mode is NO OPERATION.
- If the rehabilitation of IMSI and location information fails, the resulting operation mode is NO OPERATION.
- If the rehabilitation of IMSI and location information is successful, the resulting operation mode is set according to the following table:

	NO BDN SIM	BDN ENABLED	BDN DISABLED
NO FDN SIM	ADN ENABLED	ADN BDN ENABLED	ADN ENABLED
FDN ENABLED	FDN ENABLED	FDN BDN ENABLED	FDN ENABLED
FDN DISABLED	ADN ENABLED	ADN BDN ENABLED	ADN ENABLED

The resulting operation mode for mobiles without supporting call control with the SIM card is given by the following rules:

- If the mobile doesn´t support FDN or FDN is not supported by the SIM the resulting operation mode is NO OPERATION.
- If rehabilitation of IMSI and location information fails, the resulting operation mode is NO OPERATION.
- If rehabilitation of IMSI and location information is successful, the resulting operation mode is FDN ENABLED.

6.2 No SIM Inserted



(SIM 1)

MMI sends the primitive to start activation of the SIM card. The SIM driver is resetted. The callback functions for SIM insertion and SIM remove are installed.

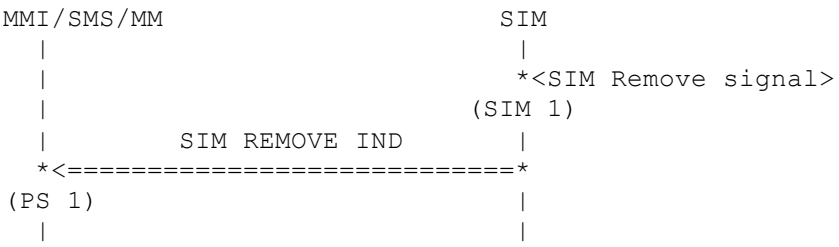
(SIM 2)

The SIM driver has not detected a SIM card and a timer in the SIM application times-out.

(MMI 1)

The MMI is informed with the cause fatal error about the absence of a SIM card.

6.3 SIM Removing



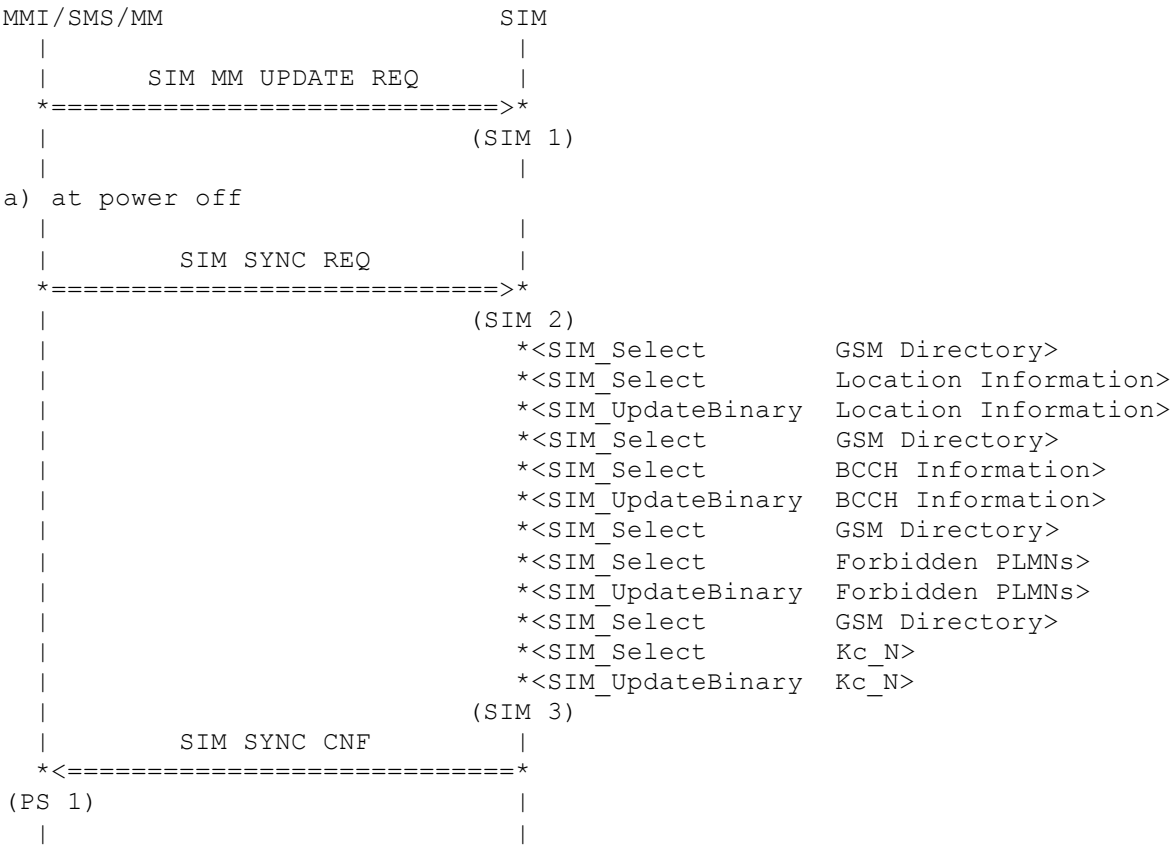
(SIM 1)

The SIM driver sends a signal that the SIM card is removed.

(PS 1)

Mobility management is informed about removing of the SIM card and starts de-registration.

6.4 SIM Updating



(SIM 1)

Whenever mobility management has changed location information, changed BCCH information, a changed forbidden PLMN list or a changed Kc or ciphering key sequence number it updates the SIM card content.

To avoid attempt to the SIM-Hardware the changes are stored only in the SIM application.

If SIM toolkit is supported by the mobile additionally the cell identity is stored by the SIM application.

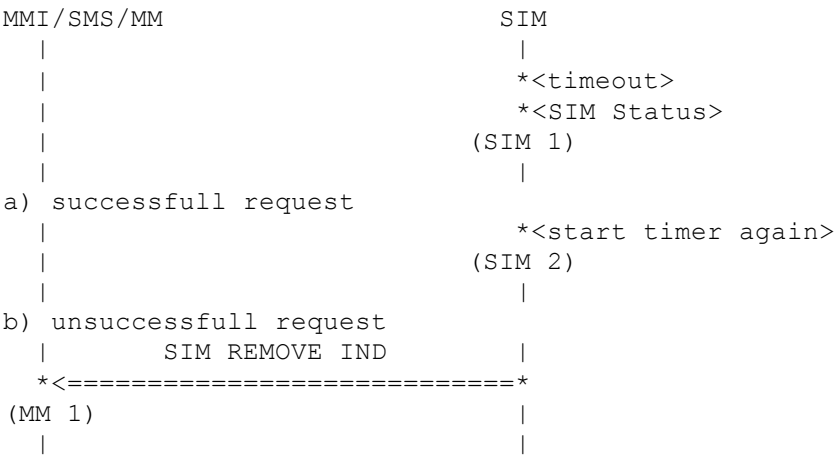
(SIM 2)

If the mobile station is switched off, MMI sends a SIM SYNC REQ to the SIM application. This triggers the SIM application to update this data fields on the SIM card.

(PS 1)

If the synchronisation is finished a confirmation is send to the MMI.

6.5 SIM Status



(SIM 1)

All thirty seconds the status timer times-out. The SIM driver call SIM status is used to check the presence of the SIM card.

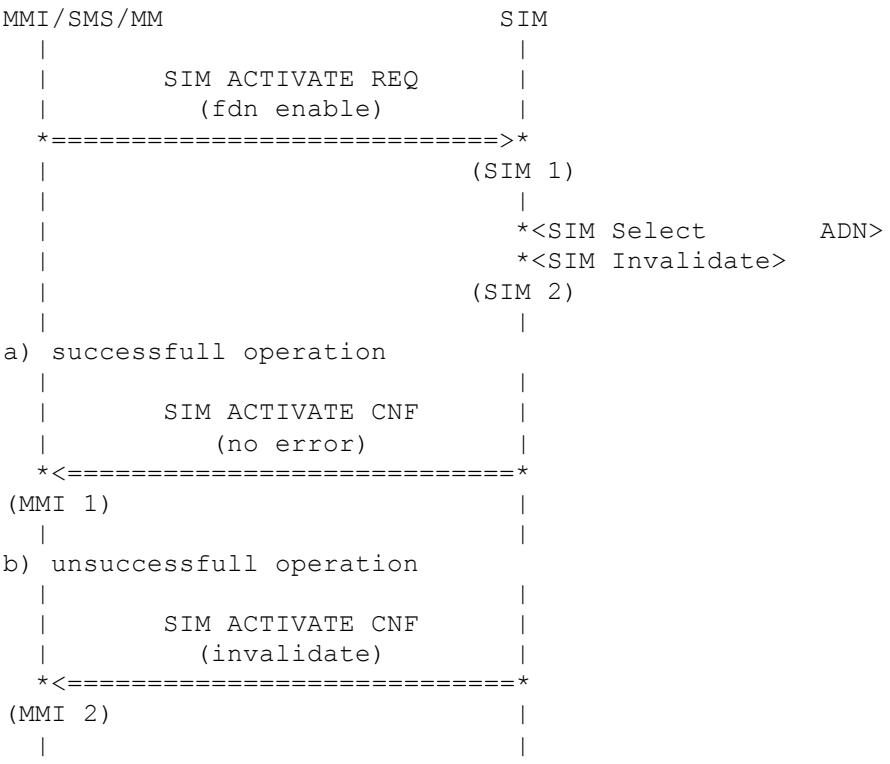
(SIM 2)

If the request is sucessful, the timer is started again.

(MMI 1)

An unsuccessful request leads to the remove indication to MM.

6.6 Change Operation Mode to Restricted Operation



(SIM 1)

MMI requests the change to restricted operation. This will be done only if

- The mobile is in unrestricted operation and
- Service 3 is allocated and activated (that means restricted operation is possible)

(SIM 2)

The SIM application selects the ADN field and invalidates it. This is implicit the rehabilitation of FDN and restricted operation.

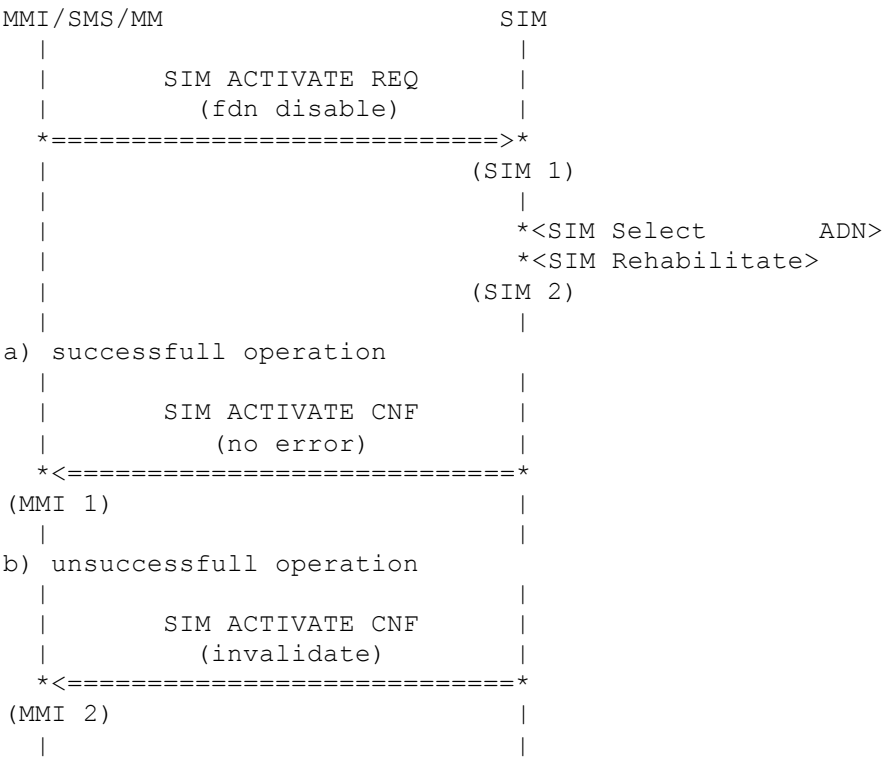
(MMI 1)

The successful end of the procedure is signalled with the cause no error to MMI. The SIM service table inside of SIM application is changed.

(MMI 2)

An unsuccessful request leads to the error cause invalidate to MMI.

6.7 Change Operation Mode to Unrestricted Operation



(SIM 1)

MMI requests the change to unrestricted operation. This will be done only if

- The mobile is in restricted operation and
- Service 2 is allocated and activated (that means unrestricted operation is possible)

(SIM 2)

The SIM application selects the ADN field and rehabilitates it.

(MMI 1)

The successful end of the procedure is signalled with the cause no error to MMI. The SIM service table inside of SIM application is changed.

(MMI 2)

An unsuccessful request leads to the error cause invalidate to MMI.

7 SIM Toolkit

The SIM toolkit package is an upgrade of the protocol stack to support the SIM toolkit functionality according GSM 11.11. To include this functionality the SIM application must be compiled with the compile switch SIM_TOOLKIT and the sim toolkit object sim_stk.obj must be linked.

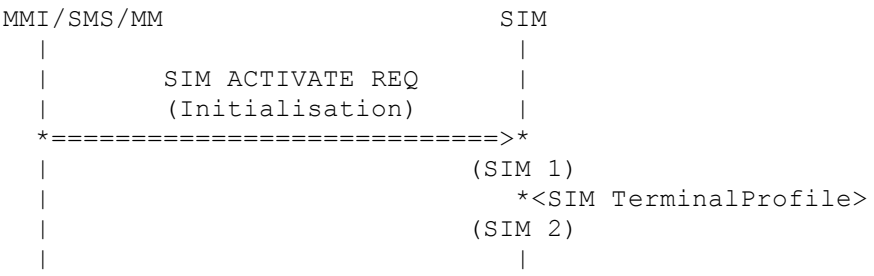
Additional support must be available by the MMI.

The following procedures / pro-active commands / SIM Toolkit Commands are supported:

- Profile Download
- REFRESH
- MORE TIME

- POLL INTERVALL
- POLLING OFF
- SET UP CALL
- SEND SS
- SEND SMS
- PLAY TONE
- DISPLAY TEXT
- GET INKEY
- GET INPUT
- SELECT ITEM
- SET UP MENU
- PROVIDE LOCAL INFO
- LAUNCH BROWSER
- OPEN CHANNEL
- CLOSE CHANNEL
- RECEIVE DATA
- SEND DATA
- GET CHANNEL STATUS
- Terminal Response
- Envelope

7.1 Profile Download



(SIM 1)

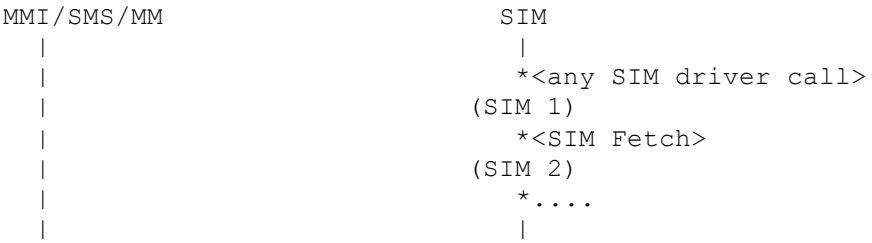
During Activation of the mobile MMI requests the initialisation of the SIM application. One parameter of the initiating SIM ACTIVATE REQ primitive is the SIM toolkit capability parameter. This parameter is a bitmap according GSM 11.11 describing the SIM toolkit capabilities of the mobile.

(SIM 2)

If a proactive SIM card is detected (phase 2+ card) during initialisation of the SIM application the profile download is performed. Therefore the mobile capabilities for SIM toolkit are forwarded to the SIM card using the SIM TerminalProfile driver call.

If this operation is successfull, the SIM application supports SIM toolkit during this session.

7.2 Pro-active Commands



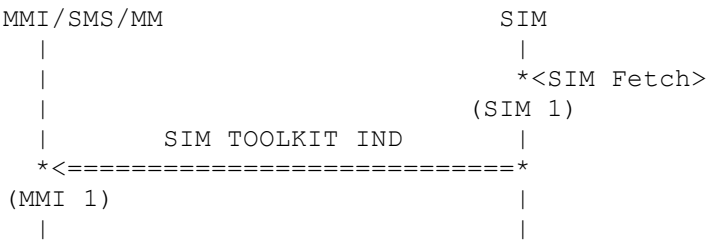
(SIM 1)

Any SIM driver Call has two result parameters SW1 and SW2. If the parameter SW1 is 0x91 or 0x9F this is an indication that the pro-active SIM card has a SIM Toolkit command to send. In this case the parameter SW2 indicates the length of the data.

(SIM 2)

Then the pro-active command is read from the SIM card using the driver call SIM Fetch. The following procedure depends on the pro-command type.

7.2.1 Display Text



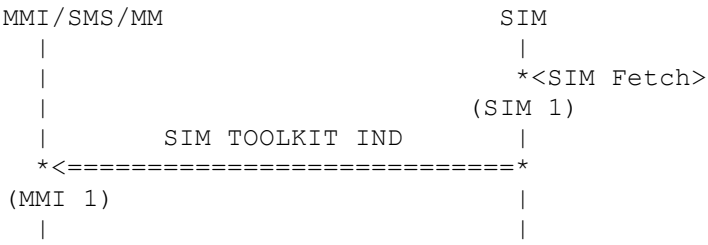
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the DISPLAY TEXT command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to display text in the MMI display.

7.2.2 Get Inkey



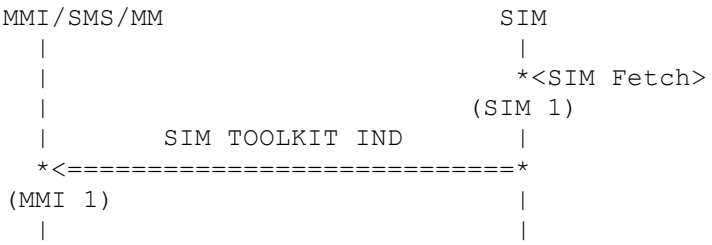
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the GET INKEY command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to request a key input from the user.

7.2.3 Get Input



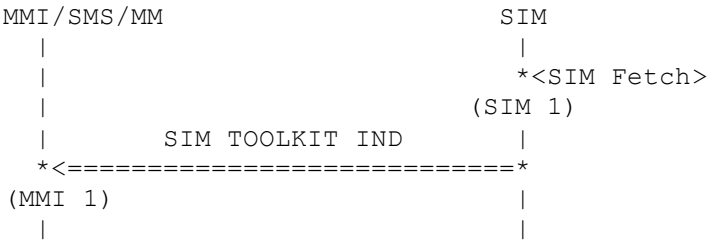
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the GET INPUT command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to get input (string) from the user.

7.2.4 Play Tone



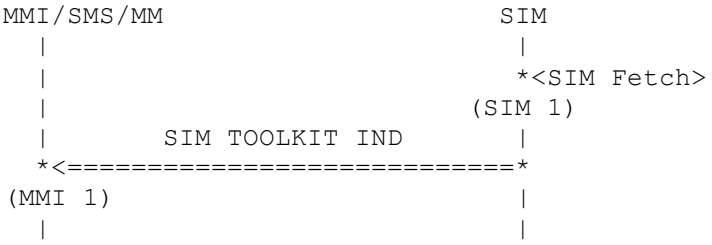
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the PLAY TONE command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to play a tone in the earpiece.

7.2.5 Refresh



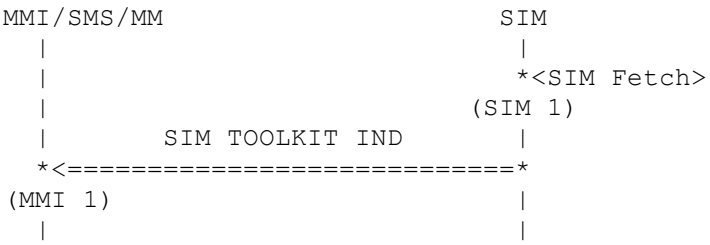
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the Refresh command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to initiate a refresh of some SIM fields up to a complete reset of the mobile.

7.2.6 Set Up Menu



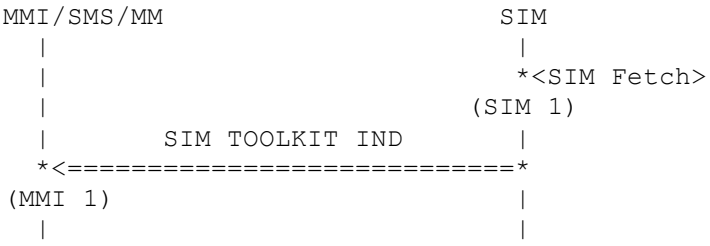
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the SET UP MENU command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to install a menu in the MMI menu system.

7.2.7 Select Item



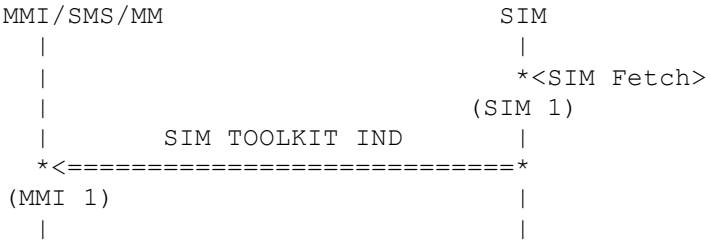
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the SELECT ITEM command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used forward a item list to the MMI to select one of the items by the user.

7.2.8 Send SMS



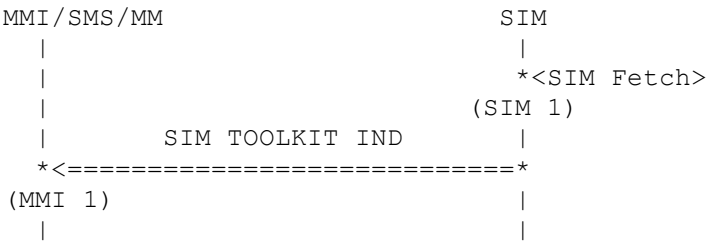
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the SEND SMS command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to send a short message initiated from the SIM card.

7.2.9 Send SS



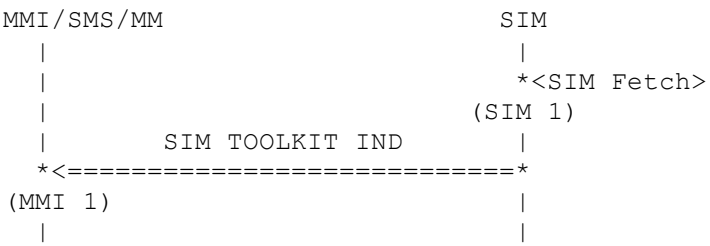
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the SEND SS command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to initiate a call independent supplementary service transaction by the SIM card.

7.2.10 Set Up Call



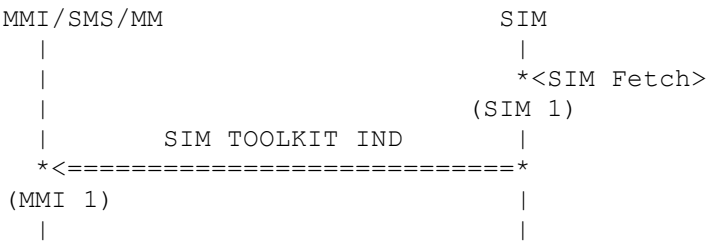
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the SET UP CALL command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to initiate a call establishment by the SIM card.

7.2.11 Launch Browser



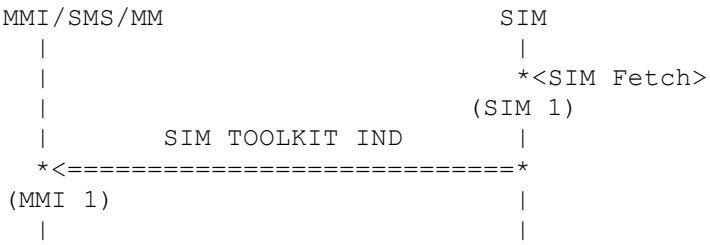
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the LAUNCH BROWSER command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to load a page on the WAP browser.

7.2.12 Set Up Event List



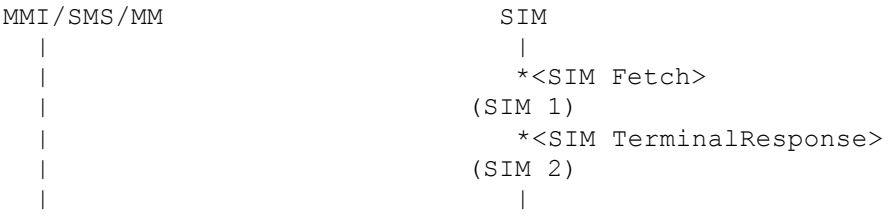
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the SET UP EVENT LIST command. This is processed by MMI.

(MMI 1)

The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. A response of MMI is expected as described later. The command is used to define requested ME events.

7.2.13 More Time



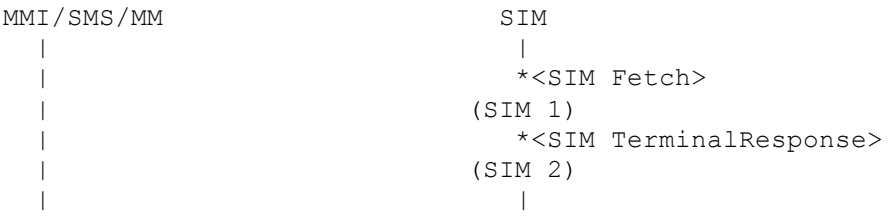
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the MORE TIME command. This is processed by the SIM application internally.

(SIM 2)

The MORE TIME command indicates that the SIM card needs more time for operation. This has no affect to the SIM application. Using the SIM driver call SIM TerminalResponse an OK indication is send to the SIM card.

7.2.14 Poll Intervall



(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the POLL INTERVALL command. This is processed by the SIM application internally.

(SIM 2)

The POLL INTERVALL command indicates that the SIM card will change the interval for the status requests. The new time is send with the unit minutes, seconds or ten seconds and will be calculated to the internal time unit. Using the SIM driver call SIM TerminalResponse an OK indication is send to the SIM card.

7.2.15 Polling Off

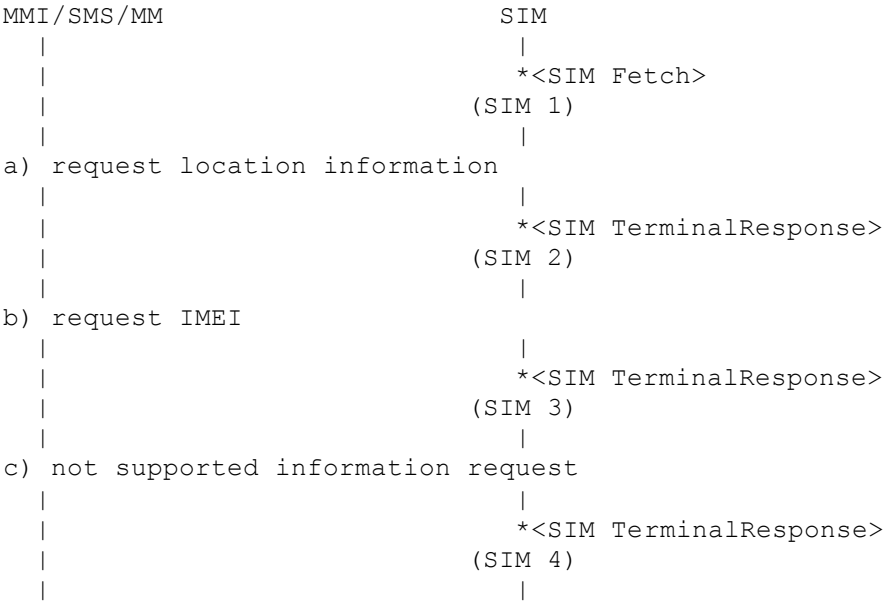
(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the POLLING OFF command. This is processed by the SIM application internally.

(SIM 2)

The POLLING OFF command indicates that the SIM shall use the default value of thirty seconds between to status requests. The used timer value in the SIM application is resetted to thirty seconds.

7.2.16 Provide Local Information



(SIM 1)

Using the SIM driver call SIM Fetch the pro-active command is read from the SIM card. It is the PROVIDE LOCAL INFORMATION command. This is processed by the SIM application internally.

(SIM 2)

The PROVIDE LOCAL INFORMATION command indicates that the SIM card requests location information. The SIM application builds a terminal response to the SIM card containing the mobile country code, the mobile network code, the location area identification and the cell identity.

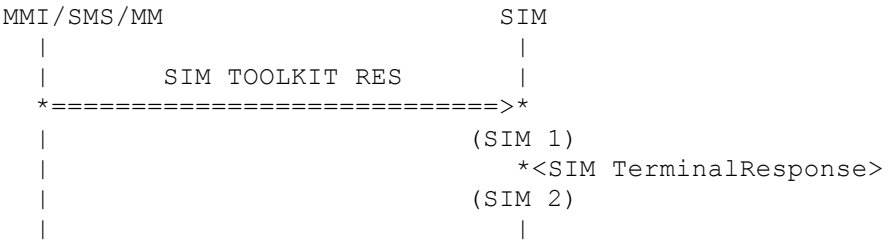
(SIM 3)

The PROVIDE LOCAL INFORMATION command indicates that the SIM card requests the IMEI. The IMEI is read from the permanent configuration memory. The SIM application builds a terminal response to the SIM card containing the IMEI.

(SIM 4)

The PROVIDE LOCAL INFORMATION command indicates that the SIM card requests a not supported information. The SIM application builds a terminal response to the SIM card indicating that the SIM application has no capability to provide the requested information.

7.3 Terminal Response



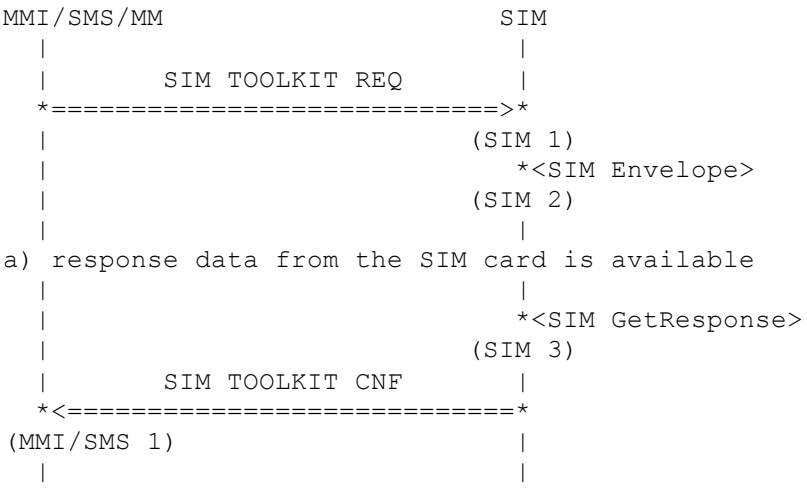
(SIM 1)

Most of the pro-active commands are forwarded to MMI and expect a response of the MMI. This response is send by MMI to the SIM application using the SIM TOOLKIT RES primitive.

(SIM 2)

The content of the primitive is forwarded to the SIM card using the SIM driver call SIM TerminalResponse.

7.4 Envelope



(SIM 1)

MMI or SMS requests a SIM toolkit operation. An example is the Data Download via SMS to the SIM card.

(SIM 2)

The requested operation is forwarded to the SIM card using the SIM driver call SIM Envelope.

(SIM 3)

The result code of this Envelope SIM driver call (SW1/SW2) indicates whether response data of the SIM card are available. If this happens the data are read with the SIM driver call SIM GetResponse. Else the response data are not read and will be cleared in the response primitive to the requesting layer.

(MMI/SMS 1)

A response primitive is build by the SIM application and send to the requesting layer (MMI or SMS).

7.5 SAT class e

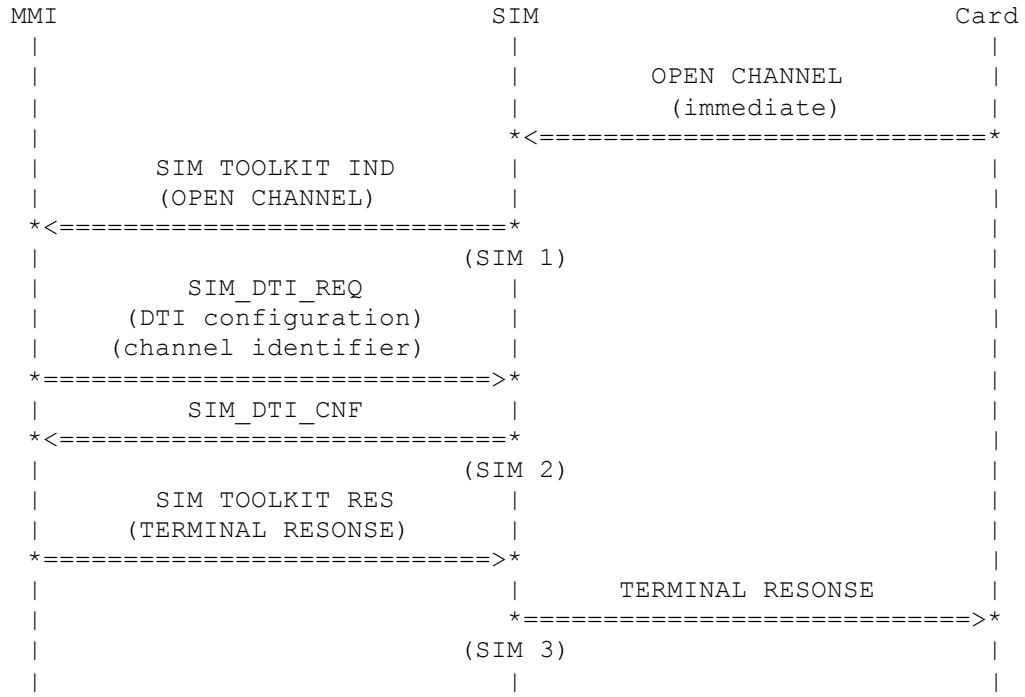
7.5.1 Open Channel

There are two types of link establishment:

- immediate link establishment
- on demand link establishment

If immediate link establishment is requested, MMI sets up the link, connects the SIM entity via DTI with an appropriate entity and sends a TERMINAL RESPONSE message to SIM.
If link establishment on demand is requested, MMI just stores the data and generates a TERMINAL RESPONSE message for SIM. The actual link establishment will be done if SIM sends a SEND DATA message to MMI with the information "send data immediately". Link establishment includes DTI configuration.
If MMI requests a DTI connection to UDP entity then the UDP SAP is also used by SIM entity.

7.5.1.1 Immediate link establishment



(SIM 1)

SIM receives an OPEN CHANNEL command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command is used to open a channel for bearer independent protocol.

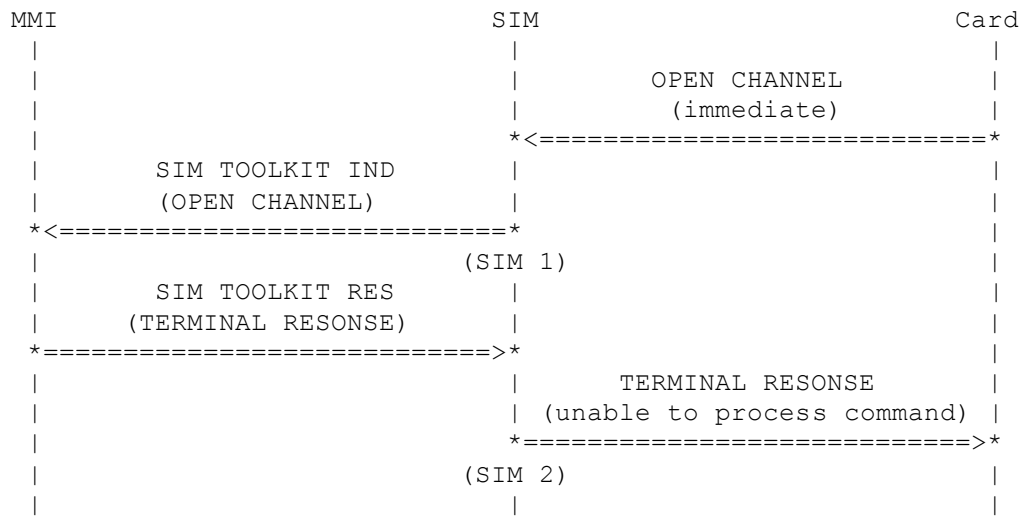
(SIM 2)

MMI requests a DTI connection and a bearer independent protocol channel (bits SIM_DTI_CONNECT and SIM_BIP_OPEN_CHANNEL are set in dti_conn parameter). It provides all necessary information including BIP channel identifier. If a UDP connection is requested (con_type parameter is set to SIM_CON_TYPE_UDP) then MMI also provides UDP relevant information (local_ip, destination_ip, destination_port). SIM opens the DTI connection and the BIP channel. After that SIM sends a confirm primitive to MMI.

(SIM 3)

MMI sends a TERMINAL RESPONSE message. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM TerminalResponse.

7.5.1.2 Immediate link establishment fails



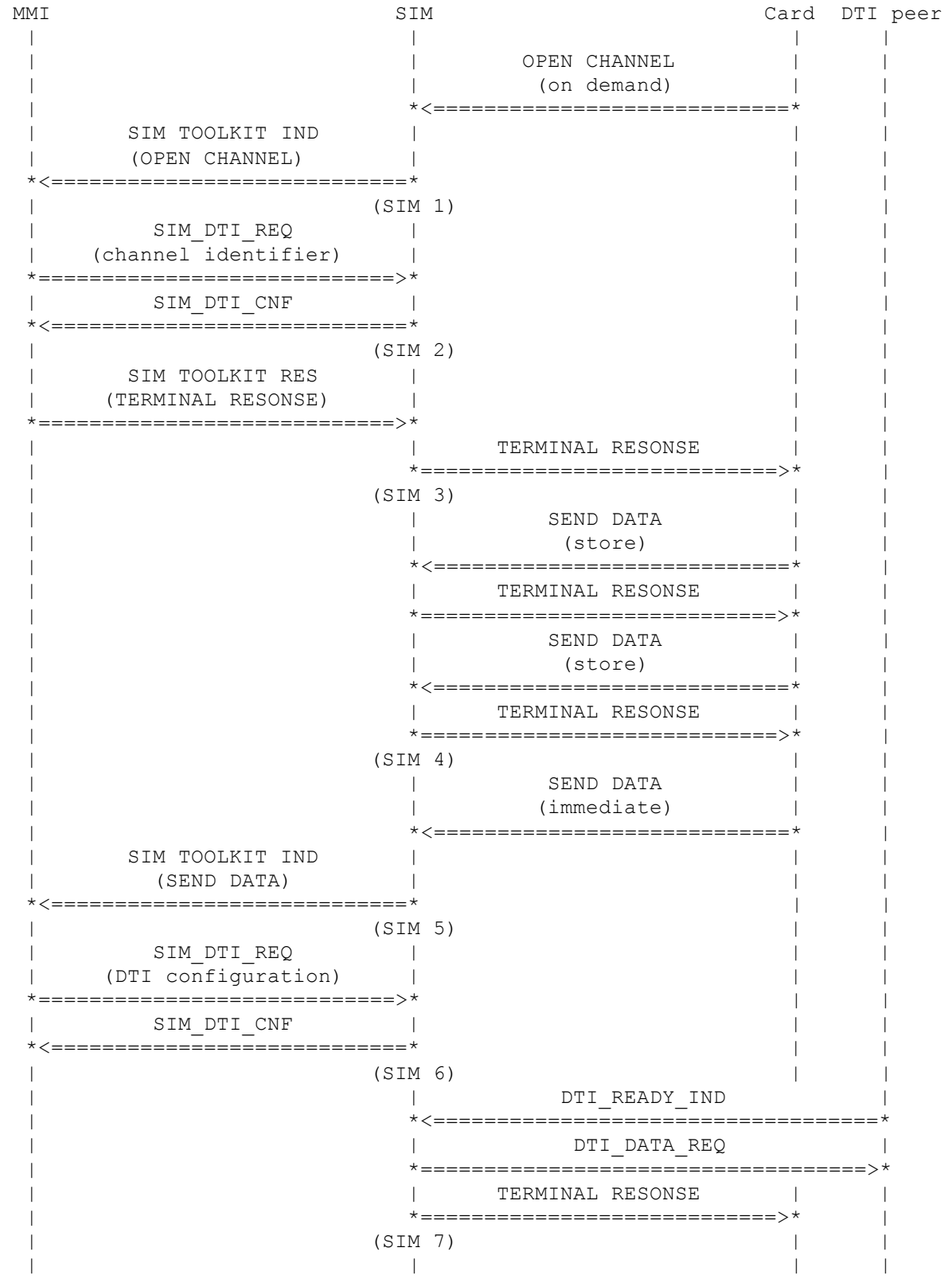
(SIM 1)

SIM receives an OPEN CHANNEL command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command is used to open a channel for bearer independent protocol.

(SIM 2)

MMI sends a TERMINAL RESPONSE message because it is not able to establish the link. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM TerminalResponse.

7.5.1.3 Link establishment on demand



(SIM 1)

SIM receives an OPEN CHANNEL command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command is used to open a channel for bearer independent protocol.

(SIM 2)

MMI requests a bearer independent protocol channel (bit SIM_BIP_OPEN_CHANNEL is set in dti_conn parameter). It provides the BIP channel identifier. SIM opens the BIP channel. After that SIM sends a confirm primitive to MMI.

(SIM 3)

MMI sends a TERMINAL RESPONSE message. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM TerminalResponse.

(SIM 4)

The SIM card gives data to send to the SIM entity. The SIM entity stores the data and expects more data because of the "store" flag. The SIM entity informs about the number of remaining bytes in the send buffer by a SIM TerminalResponse call.

(SIM 5)

If SIM gets a SEND DATA command with a set "immediate" flag then it forwards this message to MMI to indicate the need of link establishment.

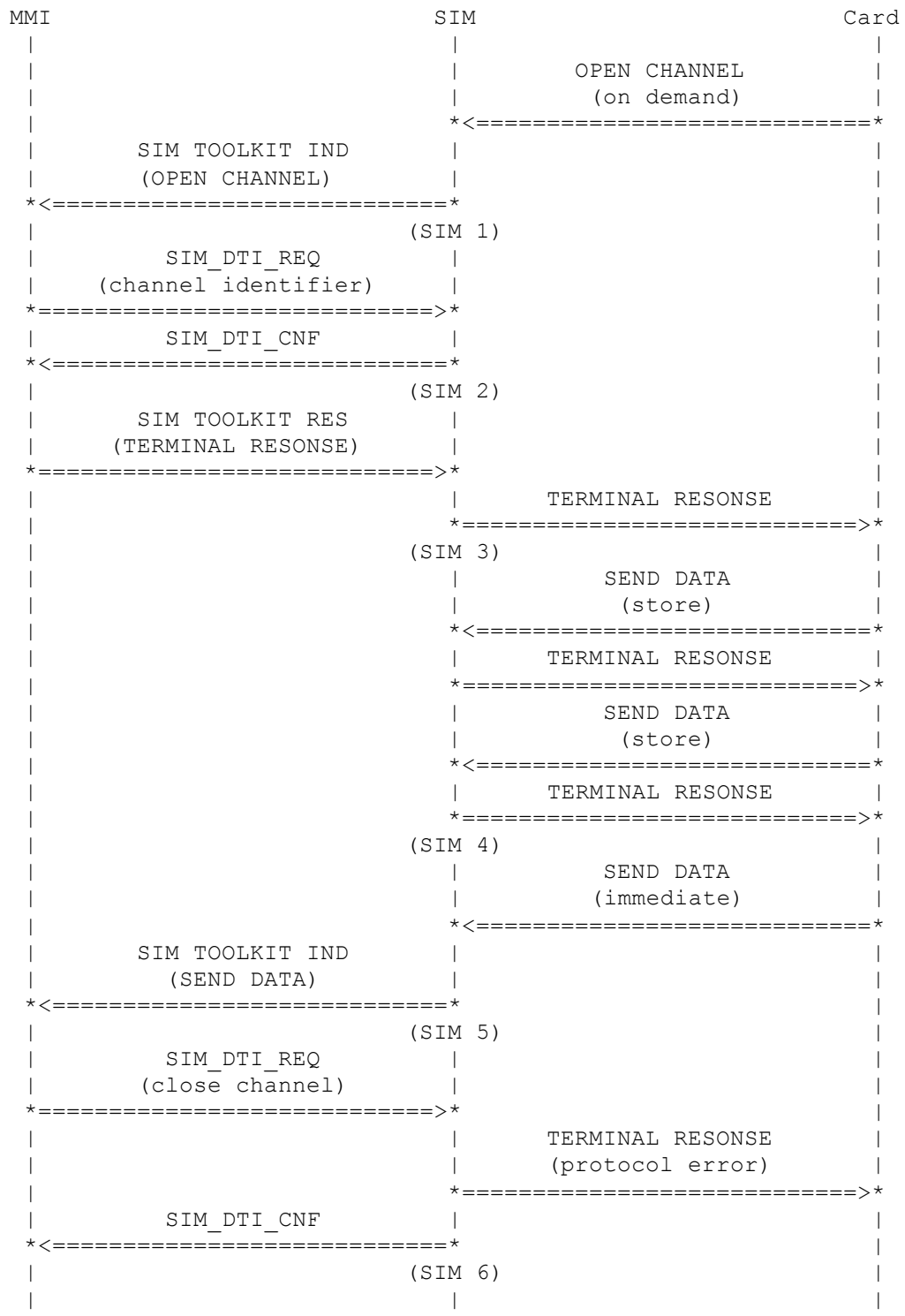
(SIM 6)

MMI requests a DTI connection (bit SIM_DTI_CONNECT is set in dti_conn parameter). It provides all necessary information. If a UDP connection is requested (con_type parameter is set to SIM_CON_TYPE_UDP) then MMI also provides UDP relevant information (local_ip, destination_ip and destination_port). SIM opens the DTI connection. After that SIM sends a confirm primitive to MMI.

(SIM 7)

SIM receives a DTI flow control primtive from the DTI peer. Now it is able to send the stored data to the DTI peer. After that SIM informs about a successful link establishment and data sending by a SIM TerminalResponse call.

7.5.1.4 Link establishment on demand fails



(SIM 1)

SIM receives an OPEN CHANNEL command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command is used to open a channel for bearer independent protocol.

(SIM 2)

MMI requests a bearer independent protocol channel (bit SIM_BIP_OPEN_CHANNEL is set in dti_conn parameter). It provides the BIP channel identifier. SIM opens the BIP channel. After that SIM sends a confirm primitive to MMI.

(SIM 3)

MMI sends a TERMINAL RESPONSE message. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM_TerminalResponse.

(SIM 4)

The SIM card gives data to send to the SIM entity. The SIM entity stores the data and expects more data because of the "store" flag. The SIM entity informs about the number of remaining bytes in the send buffer by a SIM_TerminalResponse call.

(SIM 5)

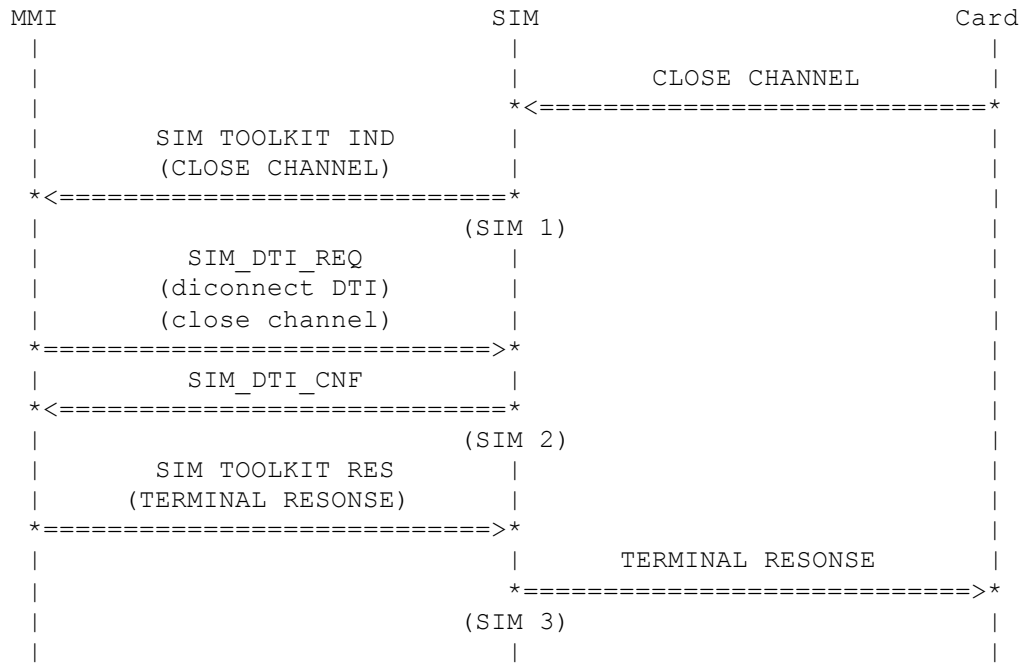
If SIM gets a SEND DATA command with a set "immediate" flag then it forwards this message to MMI to indicate the need of link establishment.

(SIM 6)

MMI is not able to establish the link. It requests a close of a bearer independent protocol channel (bit SIM_BIP_CLOSE_CHANNEL is set in dti_conn parameter). It provides the BIP channel identifier and the result codes. SIM closes the BIP channel and sends a Terminal Response message to the SIM card. After that SIM sends a confirm primitive to MMI.

7.5.2 Close Channel

7.5.2.1 SIM card initiated



(SIM 1)

SIM receives a CLOSE CHANNEL command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command is used to close a channel for bearer independent protocol.

(SIM 2)

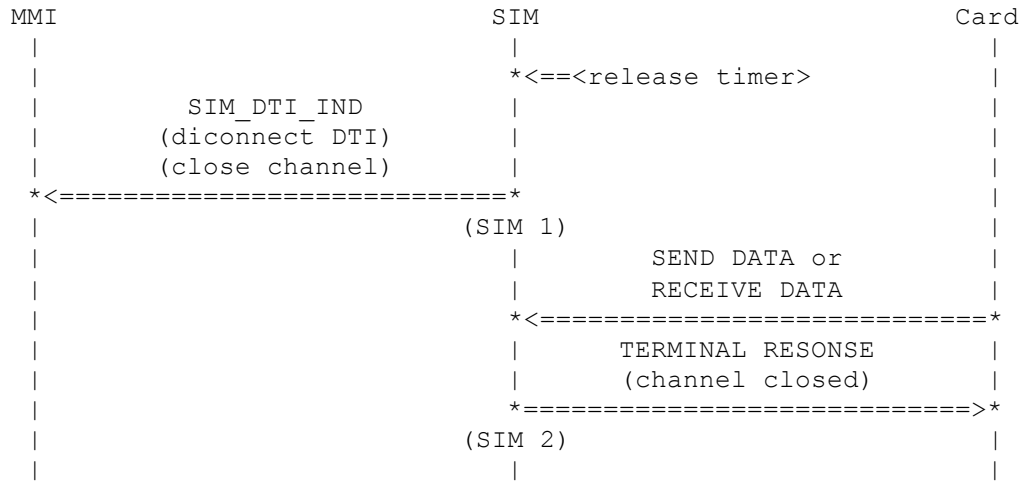
MMI requests a DTI disconnection and a close of a bearer independent protocol channel (bits SIM_DTI_DISCONNECT and SIM_BIP_CLOSE_CHANNEL are set in dti_conn parameter). It provides the BIP channel identifier. SIM closes the DTI connection and the BIP channel. After that SIM sends a confirm primitive to MMI.

(SIM 3)

MMI sends a TERMINAL RESPONSE message. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM_TerminalResponse.

7.5.2.2 SIM entity initiated

The release timer is used to release the link if there is no data exchange on the link. The value of the release timer is given in the SIM_DTI_REQ primitive. The use of the release timer can be turned off.



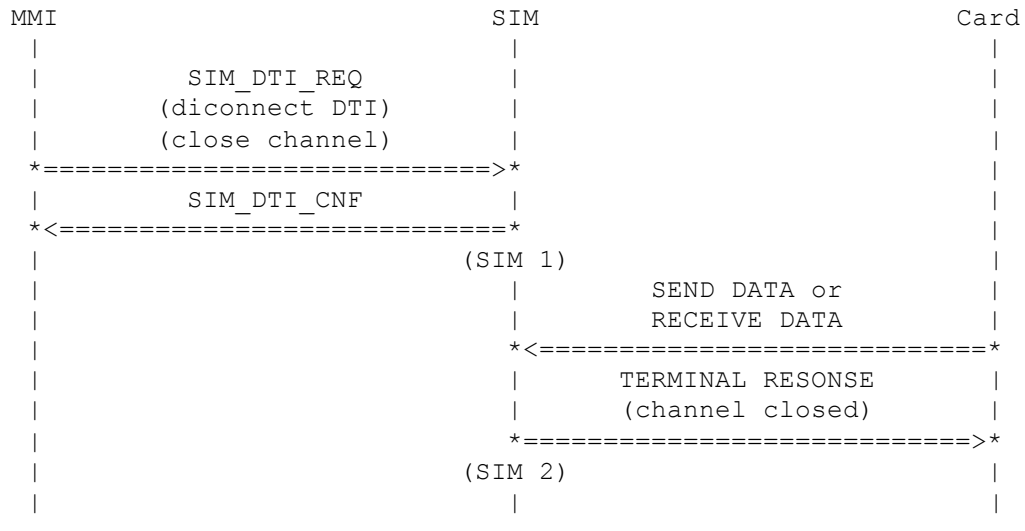
(SIM 1)

The release timer in SIM expirers and SIM closes the DTI connection and the bearer independent protocol (BIP) channel. After that SIM sends an indication primitive to MMI to inform about the closed DTI connection and the colsed BIP channel (bits SIM_DTI_DISCONNECT and SIM_BIP_CLOSE_CHANNEL are set in dti_conn parameter). It also provides the BIP channel identifier.

(SIM 2)

SIM receives a SEND DATA command or a RECEIVE DATA command from the SIM card. It answers with a TERMINAL RESPONSE message containing the information that the link is already closed an not usable any more.

7.5.2.3 MMI initiated



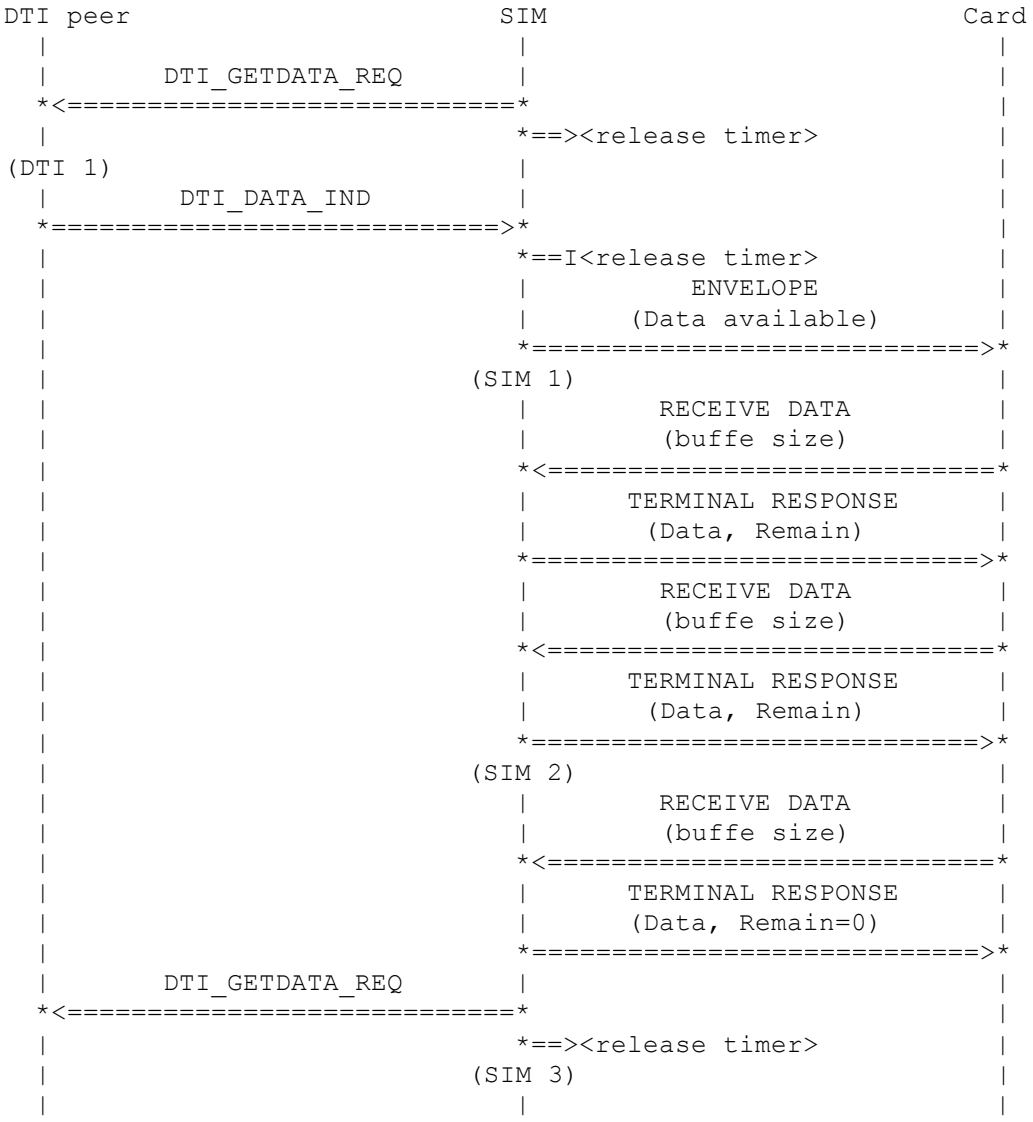
(SIM 1)

MMI requests a DTI disconnection and a close of a bearer independent protocol channel (bits SIM_DTI_DISCONNECT and SIM_BIP_CLOSE_CHANNEL are set in dti_conn parameter). It provides the BIP channel identifier. SIM closes the DTI connection and the BIP channel. After that SIM sends a confirm primitive to MMI.

(SIM 2)

SIM receives a SEND DATA command or a RECEIVE DATA command from the SIM card. It answers with a TERMINAL RESPONSE message containing the information that the link is already closed an not usable any more.

7.5.3 Receive Data



(DTI 1)

SIM indicates to the DTI peer that it is ready to receive data. It starts the release timer to watch for the data exchange.

(SIM 1)

The DTI peer entity sends bearer independent protocol data to SIM. SIM stops the release timer, because of the received data. If possible SIM informs the SIM application on the SIM card about available data by calling the driver function SIM Envelope.

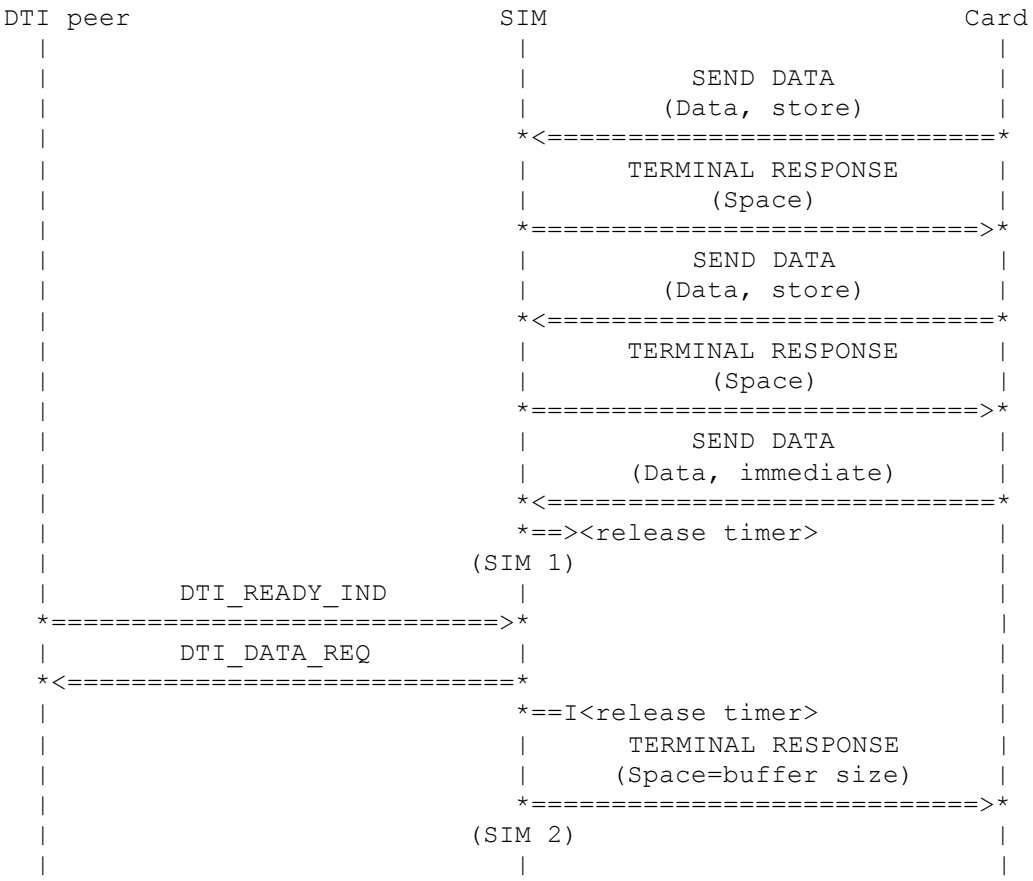
(SIM 2)

The SIM card requests the received data and provides its internal buffer size by a RECEIVE DATA message. The received data is forwarded to the SIM card in a SIM TerminalResponse call. The number of remaining bytes in the reception buffer is also forwarded, if it is not possible to forward all data in this function call.

(SIM 3)

The SIM card requests the rest of the received data by a RECEIVE DATA message. The rest of the received data is forwarded to the SIM card in a SIM TerminalResponse call. The number of remaining bytes is 0 to indicate that the reption buffer is empty now. SIM indicates to the DTI peer that it is ready to receive data again. It starts the release timer again to watch for the data exchange.

7.5.4 Send Data



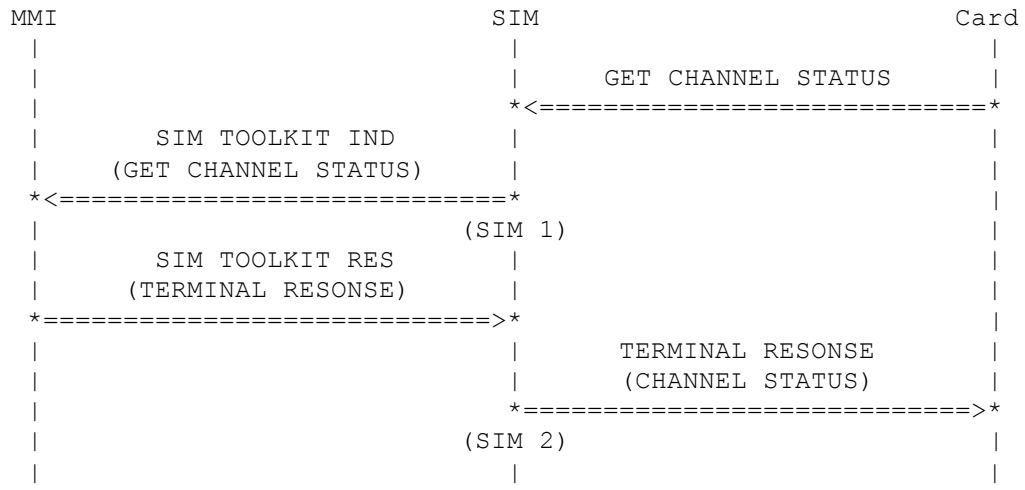
(SIM 1)

The SIM card gives data to send to the SIM entity by a SEND DATA message. The SIM entity stores the data and expects more data because of the "store" flag. The SIM entity informs about the number of bytes of empty space in the send buffer by a SIM TerminalResponse call. If the "immediate" flag is set in a SEND DATA message then the SIM entity stores the data, but does not expect more data. SIM does not call SIM TerminalResponse until the stored data is sent. It starts the release timer to watch for the data exchange.

(SIM 2)

The DTI peer entity indicates to SIM that it is ready to receive data. SIM stops the release timer, forwards the whole send buffer content to the DTI peer entity and calls SIM TerminalResponse to indicate a successful send operation and an empty send buffer to the SIM card.

7.5.5 Get Channel Status



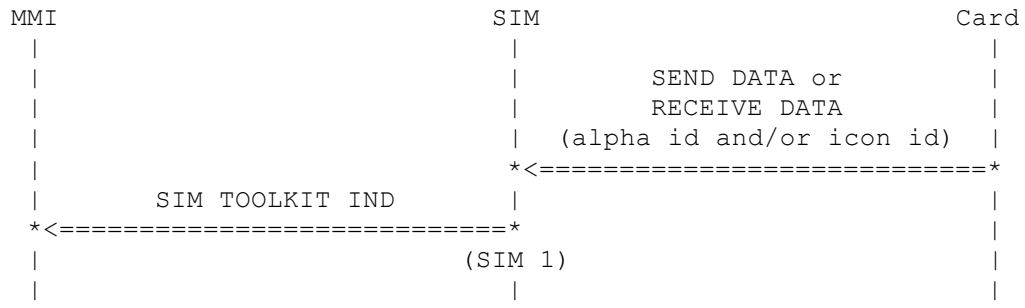
(SIM 1)

SIM receives a GET CHANNEL STATUS command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command is used to get the current channel status.

(SIM 2)

MMI sends a TERMINAL RESPONSE message. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM TerminalResponse.

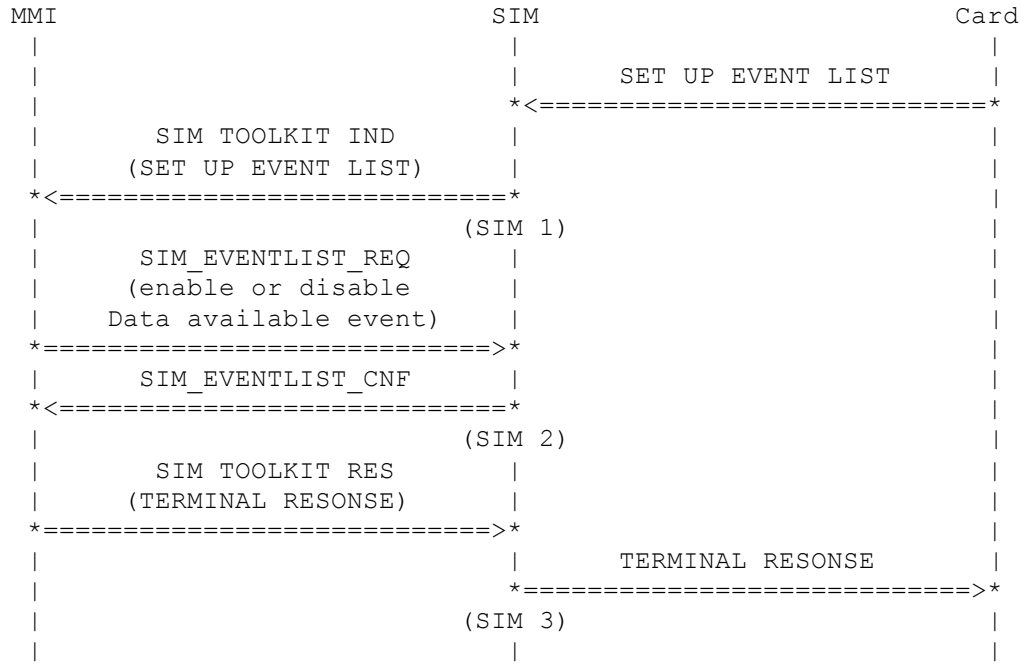
7.5.6 Forward of Alpha identifier and Icon identifier



(SIM 1)

If the SIM entity receives a SEND DATA message or a RECEIVE DATA message that contains an Alpha identifier or an Icon identifier then a duplicate of this message is forwarded to MMI.

7.5.7 Set Up Event List



(SIM 1)

SIM receives a SET UP EVENT LIST command. The content of the pro-active command is forwarded to MMI with the SIM TOOLKIT IND primitive. The command requests notification of the SIM card on specific events in the ME.

(SIM 2)

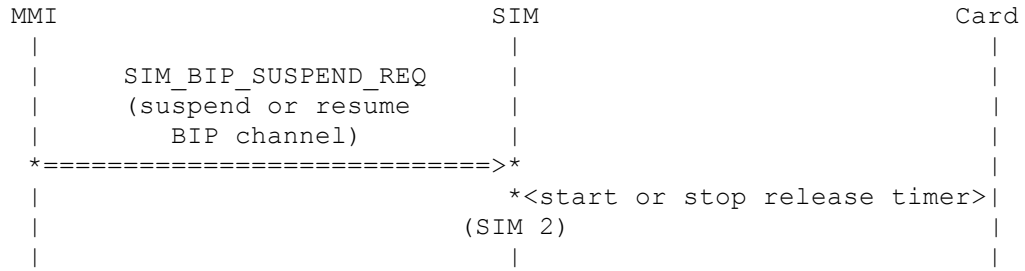
MMI informs about a change of the status of the Data available event. SIM stores the new status and sends a confirm primitive to MMI.

(SIM 3)

MMI sends a TERMINAL RESPONSE message. The content of the primitive is forwarded to the SIM card using the SIM driver call SIM_TerminalResponse.

7.5.8 Suspend and Resume channel

The release timer is used to release the link if there is no data exchange on the link. The value of the release timer is given in the SIM_DTI_REQ primitive. The use of the release timer can be turned off. MMI informs SIM if the link is suspended. In this case SIM stops the release timer.



(SIM 1)

MMI informs about a change of the status of a bearer independent protocol (BIP) channel. SIM reacts with a start or stop of the release timer.

8 SIM Driver Simulation

The SIM driver simulation replaces the SIM driver in the Windows-Environment. Using the principle of dynamic configuration a lot of modes are defined to have a specific behaviour of the SIM driver / SIM card to test all procedures of the SIM application.

A test mode is configured using the dynamic configuration for the SIM:

SIM CONFIG MODE=n

The parameter n indicates the test mode.

The SIM driver simulation is located in the module sim_csf.c. This object will be linked in the windows environment to the executable and replaces the SIM driver of the target version. The SIM driver simulation is not linked to the target version

8.1 SIM Test modes

The following table show the different test modes. Default values are:

- PIN enabled
- Phase 2 card
- Emergency Call Codes available
- Preferred Languages available
- SIM Service Table Phase 2
- IMSI and Location Information validated
- No SIM toolkit capability

No	Test Mode	Remark
0	PIN disabled	Default mode
1		
2	Blocked SIM Card	
3		
4	PIN disabled, Phase 1, no Emergency Call Codes, no Preferred Languages, SIM Service Table Phase 1	
5	Phase 1, no Emergency Call Codes, no Preferred Languages, SIM Service Table Phase 1	
6	Phase 1, no Emergency Call Codes, no Preferred Languages, SIM Service Table Phase 1	
7		
8	PIN disabled, Phase 2+, SIM Service Table Phase 2+	
9	Phase 2+, SIM Service Table Phase 2+	
10	IMSI/Location Information invalidated, SIM Service Table Phase 2 with FDN	
11	IMSI/Location Information invalidated	
12	IMSI/Location Information invalidated, SIM Service Table Phase 2 with FDN	

No	Test Mode	Remark
14	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN	
15	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN and BDN	
16	IMSI/Location Information invalidated, Phase 2+	
17	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with ADN and BDN	
18	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN	
19	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN and BDN	
20	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN and BDN	
21	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN and BDN	
22	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN, rehabilitation fails	
23	IMSI/Location Information invalidated, Phase 2+, SIM Service Table Phase 2 with FDN and BDN, rehabilitation fails	
24	PIN disabled, SIM Service Table Phase 2 with ADN and FDN	
25	Negative status request	
26	PIN disabled, Phase 2+, SIM Service Table Phase 2+, SIM-Toolkit enabled	
27	Pro-active Command Display text (short)	Only with SIM-Toolkit
28	Pro-active Command Display text (long)	Only with SIM-Toolkit
29	Pro-active Command Get Inkey	Only with SIM-Toolkit
30	Pro-active Command Get Input	Only with SIM-Toolkit
31	Pro-active Command More Time, TerminalResponse with OK expected	Only with SIM-Toolkit
32	Pro-active Command Play Tone	Only with SIM-Toolkit
33	Pro-active Command Poll Interval TerminalResponse with OK and duration expected	Only with SIM-Toolkit
34	Pro-active Command Polling Off	Only with SIM-Toolkit
35	Pro-active Command Refresh	Only with SIM-Toolkit

No	Test Mode	Remark
36	Pro-active Command Set Up Menu	Only with SIM-Toolkit
37	Pro-active Command Select Item	Only with SIM-Toolkit
38	Pro-active Command Send SMS	Only with SIM-Toolkit
39	Pro-active Command Send SS	Only with SIM-Toolkit
40	Pro-active Command Set Up Call	Only with SIM-Toolkit
41	Pro-active Command Provide Local Information (location information), TerminalResponse with OK and location information expected	Only with SIM-Toolkit
42	Pro-active Command Provide Local Information (location information), TerminalResponse with negative result for location information expected	Only with SIM-Toolkit
43	Pro-active Command Provide Local Information (IMEI), TerminalResponse with OK and IMEI expected	Only with SIM-Toolkit
44	Pro-active Command Provide Local Information (not supported), TerminalResponse with negative result no capability expected	Only with SIM-Toolkit
45	Blocked SIM card, no unblock attempts	
46	Response data for Envelope Command	Only with SIM-Toolkit
47	Pro-active Command Display Text (short), TerminalResponse with OK expected	Only with SIM-Toolkit

Appendices

A. Acronyms

DS-WCDMA Direct Sequence/Spread Wideband Code Division Multiple Access

B. Glossary

International Mobile Telecommunication 2000 (IMT-2000/ITU-2000) Formerly referred to as FPLMTS (Future Public Land-Mobile Telephone System), this is the ITU's specification/family of standards for 3G. This initiative provides a global infrastructure through both satellite and terrestrial systems, for fixed and mobile phone users. The family of standards is a framework comprising a mix/blend of systems providing global roaming. <URL: http://www.imt-2000.org/>