Technical Document

GSM PROTOCOL STACK

G23

SIM CARD

DRIVER INTERFACE

Document Number:	8415.222.00.002
Version:	0.4
Status:	Draft
Approval Authority:	
Creation Date:	2000-Sep-08
Last changed:	2015-Mar-08 by XINTEGRA
File Name:	8415_222.doc

Copyright © 2003 Texas Instruments
Texas Instruments Proprietary Information – Internal Data

Important Notice

Texas Instruments Incorporated and/or its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products, software and services at any time and to discontinue any product, software or service without notice. Customers should obtain the latest relevant information during product design and before placing orders and should verify that such information is current and complete.

All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment. TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI products, software and/or services. To minimize the risks associated with customer products and applications, customers should provide adequate design, testing and operating safeguards.

Any access to and/or use of TI software described in this document is subject to Customers entering into formal license agreements and payment of associated license fees. TI software may solely be used and/or copied subject to and strictly in accordance with all the terms of such license agreements.

Customer acknowledges and agrees that TI products and/or software may be based on or implement industry recognized standards and that certain third parties may claim intellectual property rights therein. The supply of products and/or the licensing of software does not convey a license from TI to any third party intellectual property rights and TI expressly disclaims liability for infringement of third party intellectual property rights.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products, software or services are used.

Information published by TI regarding third–party products, software or services does not constitute a license from TI to use such products, software or services or a warranty, endorsement thereof or statement regarding their availability. Use of such information, products, software or services may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

No part of this document may be reproduced or transmitted in any form or by any means, electronically or mechanically, including photocopying and recording, for any purpose without the express written permission of TI.

Change History

Date	Changed by	Approved by	Version	Status	Notes
2000-Sep-08	GSP		0.1		1
2000-Sep-28	GSP		0.2		2
2001-Nov-07	GSP		0.3		3
2003-May-20	XINTEGRA		0.4	Draft	

Notes:

1. Initial version
2. SIM_Update Binary/offset description, SIM_Terminal/Profile, SIM_Terminal/Response, SIM_Fetch description
3. New document number, SIM_Status_Extended revised, Some BYTE changed to UBYTE

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

[C_8415.0026] 8415.026.99.012; March 19, 1999
Generic Driver Interface – Functional Specification; Condat

1.2 Abbreviations

A3 Algorithm 3, authentication algorithm; used for authenticating the subscriber
A5 Algorithm 5, cipher algorithm; used for enciphering/deciphering data
A8 Algorithm 8, cipher key generator; used to generate Kc
A38 A single algorithm performing the functions of A3 and A8
ACM Accumulated Call Meter
AND Abbreviated Dialling Number
ADM Access condition to an EF which is under the control of the authority which creates this file
ALW ALWays
AoC Advice of Charge
APDU Application Protocol Data Unit
ATR Answer To Reset
BCCH Broadcast Control CHannel
BCD Binary Coded Decimal
BDN Barred Dialling Number
BTS Base Transmitter Station
CB Cell Broadcast
CBMI Cell Broadcast Message Identifier
CCP Capability/Configuration Parameter
CHV Card Holder Verification information; access condition used by the SIM for the verification of the
identity of the user
CLA CLAss
CNL Co-operative Network List
DCK De-personalization Control Keys
DCS Digital Cellular System
DF Dedicated File (abbreviation formerly used for Data Field)
DTMF Dual Tone Multiple Frequency
ECC Emergency Call Code
EF Elementary File
eMLPP enhanced Multi-Level Precedence and Preemption Service
etu elementary time unit
FDN Fixed Dialling Number
HPLMN Home PLMN
ID IDentifier
IMSI International Mobile Subscriber Identity
ISO International Organization for Standardization
Kc Cryptographic key; used by the cipher A5
Ki Subscriber authentication key; the cryptographic key used by the authentication algorithm, A3, and cipher key
generator, A8
LAI Location Area Information; information indicating a cell or a set of cells lgth The (specific) length of a data
unit
LND Last Number Dialled
LSB Least Significant Bit
MCC Mobile Country Code
ME Mobile Equipment
MF Master File
MMI Man Machine Interface
MNC Mobile Network Code
MS Mobile Station
MSISDN Mobile Station international ISDN number
MSB Most Significant Bit
NET NETwork
NEV NEVer
NPI Numbering Plan Identifier

PIN Personal Identification Number (obsolete terms for CHV1 and HV2, respectively)
PIN2 Personal Identification Number 2 (obsolete terms for CHV1 and HV2, respectively)
PLMN Public Land Mobile Network
PTS Protocol Type Select (response to the ATR)
PUK PIN Unblocking Key (obsolete terms for UNBLOCK CHV1 and UNBLOCK CHV2, respectively)
PUK2 PIN2 Unblocking Key (obsolete terms for UNBLOCK CHV1 and UNBLOCK CHV2, respectively)
RAND A RANDom challenge issued by the network
RFU Reserved for Future Use
SDN Service Dialling Number
SIM Subscriber Identity Module
SMS Short Message Service
SRES Signed RESponse calculated by a SIM
SSC Supplementary Service Control string
SW1 Status Word 1
SW2 Status Word 2
TMSI Temporary Mobile Subscriber Identity
TON Type Of Number
TP Transfer layer Protocol
TPDU Transfer Protocol Data Unit
TS Technical Specification
UNBLOCK CHV1/2 value to unblock CHV1/CHV2
VBS Voice Broadcast Service
VGCS Voice Group Call Service
VPLMN Visited PLMN

2 Introduction

G23 is a software package implementing Layers 2 and 3 of the ETSI-defined GSM air interface signaling protocol, and as such represents the part of a GSM mobile station's protocol software which is both, platform and manufacturer independent. Therefore, G23 can be viewed as a building block providing standardized functionality through generic interfaces for easy integration.

The G23 suite of products consists of the following items:

- Layers 2 and 3 for speech & short message services,
- Layers 2 and 3 for fax & data services,
- Application Control Interface,
- Slim MMI [02.30] and
- Test and integration support tools.

This document describes the functional interface of the SIM card driver interface.

3 Interface description of SIM card driver

3.1 Function overview

Name	Description
SIM_Init	Initialize data structures and install interrupt handler
SIM_Reset	Reset SIM card
SIM_Select	Select a DF or a EF
SIM_Status	Request status from SIM card
SIM_Status_Extended	Request status from SIM card
SIM_PowerOff	Switch of the SIM card
SIM_UpdateBinary	Store data in the current transparent EF
SIM_UpdateRecord	Store a record in the current linear fixed or cyclic EF
SIM_Increase	Add value to a record of a cyclic EF
SIM_ReadBinary	Read data from the current EF
SIM_ReadRecord	Read a record from the current linear fixed or cyclic EF
SIM_VerifyCHV	Verify the specified CHV
SIM_ChangeCHV	Change the specified CHV
SIM_DisableCHV	Disable the specified CHV
SIM_EnableCHV	Enable CHV 1
SIM_UnblockCHV	Unblock the specified CHV and store a new CHV
SIM_Invalidate	Invalidate the current EF
SIM_Rehabilitate	Rehabilitate the current EF
SIM_RunGSMAlgo	Authentication procedure
SIM_GetResponse	Get response data from the card
SIM_TerminalProfile	Used by ME to send its toolkit capabilities to SIM
SIM_Envelope	ME or NET command for SIM toolkit
SIM_Fetch	Used by ME to inquiry of what SIM toolkit need to do
SIM_TerminalResponse	Response to an envelope command

3.2 Functions

3.2.1 SIM_Init

Definition:

```
VOID SIM_Init
(
    VOID            (* insert)(void),
    VOID            (* remove)(void)
) ;
```

Parameters:

Name	Description
(* insert)(void)	Pointer to callback function when SIM is inserted
(* remove)(void)	Pointer to callback function when SIM is removed

Return values:

Name	Description

None

Description

Initialize data structures and install interrupt handler.

3.2.2 SIM_Reset

Definition:

```
int SIM_Reset
(
    T_SIM_CARD*          Info
) ;
```

Parameters:

Name	Description
T_SIM_CARD*	Pointer on a structure containing polarity of the card (direct/inverse), size of the ATR and ATR bytes

Return values:

Name	Description
0	Ok
1	No SIM inserted
2	No SIM inserted
others	Invalid card

Description

Reset SIM card

3.2.3 SIM_PowerOff

Definition:

```
USHORT SIM_PowerOff
(
    VOID
) ;
```

Parameters:

Name	Description

None

Return values:

Name	Description

None

Description

This function is used to deactivate the SIM card.

3.2.4 SIM_Select

Definition:

```
USHORT SIM_Select
(
    USHORT          id,
    UBYTE*          dat,
    UBYTE*          size,
) ;
```

Parameters:

Name	Description
id	File ID
dat	Pointer to the buffer for SIM card response
size	Pointer to the size of data in the buffer

Return values:

Name	Description
0x6Bxx	Incorrect parameter P1 or P2
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9404	File ID not found/Pattern not found
0x9Fxx	Length 'XX' of the response data

Description

This function selects a file according to GSM 11.11/6.5. 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.

3.2.5 SIM_Status

Definition:

```
USHORT SIM_Status
(
    UBYTE*          result,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
size	Pointer to the size of data in the buffer

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Bxx	Incorrect parameter P1 or P2
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x9240	Memory problem
0x9404	File ID not found/Pattern not found

Description

This function returns information concerning the current directory. A current EF is not affected by the STATUS function. It is also used to give an opportunity for a pro-active SIM to indicate that the SIM wants to issue a SIM Application Toolkit command to the ME.

3.2.6 SIM_Status_Extended

Definition:

```
USHORT SIM_Status_Extended
(
    UBYTE*          result,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
len	Number of returned bytes
size	Pointer to the size of data in the buffer

Return values:

Name	Description
0x9000	Normal ending of the command
0x91xx	Indication of SIM toolkit data. SELECT ITEM with xx bytes.
0x9404	File ID not found/Pattern not found

Description

This function returns information concerning the current directory. A current EF is not affected by the STATUS function. It is also used to give an opportunity for a pro-active SIM to indicate that the SIM wants to issue a SIM Application Toolkit command to the ME.

3.2.7 SIM_UpdateBinary

Definition:

```
USHORT SIM_UpdateBinary
(
    UBYTE*          result,
    UBYTE*          dat,
    USHORT          offset,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
dat	Pointer to data to write to SIM
offset	Offset from where you start to write in a file
	It composes parameter P1 and P2
len	Number of bytes to be written (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem
0x9400	No EF selected Status
0x9404	File ID not found/Pattern not found
0x9408	File is inconsistent with the command
0x9808	In contradiction with CHV status

Description

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

3.2.8 SIM_UpdateRecord

Definition:

```
USHORT SIM_UpdateRecord
(
    BYTE*       result,
    BYTE*       dat,
    BYTE        mode,
    BYTE        recNum,
    BYTE        len,
    BYTE*       size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
dat	Pointer to data to write to SIM card
mode	Update mode (1: next record, 2: previous, 4: current)
recNum	The record which will be written
len	Number of bytes to be written
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem
0x9400	No EF selected
0x9402	Out of range (invalid address)
0x9408	File is inconsistent with the command
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least one attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left
0x9808	In contradiction with CHV status

Description

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

3.2.9 SIM_Increase

Definition:

```
USHORT SIM_Increase
(
        UBYTE*      result,
        UBYTE*      dat,
        UBYTE*      size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
dat	Pointer to data to write to SIM card
size	Pointer to the number of return bytes

Return values:

Name	Description
0x9000	Normal ending of the command
0x9840	Negative result

Description

This function adds the value given by the ME to the value of the last increased/updated record of the current cyclic EF,
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 EF has an INCREASE access condition assigned and this condition
is fulfilled (see bytes 8 and 10 in the response parameters/data of the current EF, clause 9). 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').

3.2.10 SIM_ReadBinary

Definition:

```
USHORT SIM_ReadBinary
(
    UBYTE*      result,
    USHORT      offset,
    UBYTE       len,
    UBYTE*      size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
offset	Start of writing data
len	Number of bytes to read (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x9240	Memory problem
0x9400	No EF selected
0x9402	Out of range (invalid address)
0x9408	File is inconsistent with the command
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least on attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left
0x9808	In contradiction with CHV status
0x9404	File ID not found/Pattern not found

Description

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

3.2.11 SIM_ReadRecord

Definition:

```
USHORT SIM_ReadRecord
(
    BYTE*       result,
    BYTE        mode,
    BYTE        recNum,
    BYTE        len,
    BYTE*       size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
mode	Update mode (1: next record, 2: previous, 4: current)
recNum	The record which will be written
len	Number of bytes to be written (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x9240	Memory problem
0x9400	No EF selected
0x9402	Out of range (invalid address)
0x9408	File is inconsistent with the command
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least one attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left
0x9808	In contradiction with CHV status

Description

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

3.2.12 SIM_VerifyCHV

Definition:

```
USHORT SIM_VerifyCHV
(
    UBYTE*          result,
    UBYTE*          chv,
    UBYTE           chvType,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
chv	Points to the PIN
chvType	1: CHV1, 2: CHV2
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem
0x9400	Negative result
0x9802	No CHV initialized
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least one attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left
0x9808	In contradiction with CHV status
0x9840	Unsuccessful CHV verification, no attempt left/unsuccessful UNBLOCK CHV verification, no attempt left/CHV blocked/UNBLOCK CHV blocked

Description

This function verifies the CHV presented by the ME by comparing it with the relevant one stored in the SIM if the CHV is not disabled and 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 success-fully performed on the respective CHV.

3.2.13 SIM_ChangeCHV

Definition:

```
USHORT SIM_ChangeCHV
(
    UBYTE*          result,
    UBYTE*          oldCHV,
    UBYTE*          newCHV,
    UBYTE           chvType,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
oldCHV	Points to the currently active PIN
newCHV	Points to the new desired PIN
chvType	1: CHV1, 2: CHV2
size	Pointer to the number of return bytes

Return values:

Name	Description
0x9000	Normal ending of the command

Description

This function assigns a new value to the relevant CHV subject if the CHV is not disabled and not blocked. 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 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 conditions can never be fulfilled until the UNBLOCK CHV function has been performed successfully on the respective CHV.

3.2.14 SIM_DisableCHV

Definition:

```
USHORT SIM_DisableCHV
(
    UBYTE*          result,
    UBYTE*          CHV,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
CHV	Points to the currently active PIN
size	Pointer to the number of return bytes

Return values:

Name	Description
0x9000	Normal ending of the command

Description

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

3.2.15 SIM_EnableCHV

Definition:

```
USHORT SIM_EnableCHV
(
    UBYTE*          result,
    UBYTE*          CHV,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
CHV	Points to the currently active PIN
size	Pointer to the number of return bytes

Return values:

Name	Description
0x9000	Normal ending of the command

Description

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 may optionally be set to "enabled". Once blocked, the CHV1 can only be unblocked using the UNBLOCK CHV function. If the CHV1 is blocked and "disabled", the access condition shall remain granted. If the CHV1 is blocked and "enabled", the access condition can never be fulfilled until the UNBLOCK CHV function has been successfully performed on CHV1.

3.2.16 SIM_UnblockCHV

Definition:

```
USHORT SIM_UnblockCHV
(
    UBYTE*          result,
    UBYTE*          unblockCHV,
    UBYTE*          newCHV,
    UBYTE*          chvType,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
unblockCHV	Points to the PUK
newCHV	Points to the new desired PIN
chvType	1: CHV1, 2: CHV2
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	Normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem
0x9404	File ID not found/Pattern not found
0x9802	No CHV initialized
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least one attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left
0x9808	In contradiction with CHV status
0x9840	Unsuccessful CHV verification, no attempt left/unsuccessful UNBLOCK CHV verification, no attempt left/CHV blocked/UNBLOCK CHV blocked

Description

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.

3.2.17 SIM_Invalidate

Definition:

```
USHORT SIM_Invalidate
(
    UBYTE*          result,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
size	Pointer to the number of return bytes

Return values:

Name	Description
0x9000	Normal ending of the command

Description

This function invalidates the current EF. 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 EF
is satisfied. An invalidated file shall no longer be available within the application for any function except for the SELECT and the REHABILITATE functions unless the file status of the EF indicates that READ and UPDATE may also be performed.

3.2.18 SIM_Rehabilitate

Definition:

```
USHORT SIM_Rehabilitate
(
    UBYTE*          result,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem
0x9400	No EF selected
0x9404	File ID not found/Pattern not found
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least one attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left
0x9810	In contradiction with invalidation status

Description

This function rehabilitates the invalidated current EF. 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 EF is satisfied. If BDN is enabled then the REHABILITATE function shall not rehabilitate the invalidated EF IMSI and EF LOCI until the PROFILE DOWNLOAD procedure is performed indicating that the ME supports the "Call control by SIM" facility.

3.2.19 SIM_RunGSMAlgo

Definition:

```
USHORT SIM_RunGSMAlgo
(
    UBYTE*          result,
    UBYTE*          rand,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
rand	Pointer to a 16 byte random value
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9Fxx	Length 'XX' of the response data
0x9240	Memory problem
0x9408	File is inconsistent with the command
0x9804	access condition not fulfilled/unsuccessful CHV verification, at least one attempt left/unsuccessful UNBLOCK CHV verification, at least one attempt left

Description

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 or any sub-directory under DFGSM has been selected as the Current Directory and a successful CHV1 verification procedure has been performed.

3.2.20 SIM_GetResponse

Definition:

```
USHORT SIM_GetResponse
(
    UBYTE*          result,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the select response data buffer
len	Number of bytes to read (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x9240	Memory problem

Description

The response data depends on the preceding command. Response data is available after the commands RUN GSM ALGORITHM, SEEK (type 2), SELECT, INCREASE and ENVELOPE. 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 MF is implicitly selected after activation of the SIM, GET RESPONSE is also allowed as the first command after activation.

3.2.21 SIM_TerminalProfile

Definition:

```
USHORT SIM_TerminalProfile
(
    UBYTE*          result,
    UBYTE*          dat,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
dat	Pointer to the data (profile) transmitted to the SIM card
len	Length of the data transmitted to the SIM card (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine'X' times
0x9240	Memory problem

Description

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

This command is part of the set used by SIM Application Toolkit. This function is used by the ME to transmit to the SIM its capabilities concerning the SIM Application Toolkit functionality. The ME completes the command data /parameters of the relevant command and sends the command to the SIM. The transmitted data is processed by the SIM in a specific way depending on the tag value in the command parameters.

A SIM or ME not supporting SIM Application Toolkit does not need to support this command.

3.2.22 SIM_Envelope

Definition:

```
USHORT SIM_Envelope
(
    UBYTE*          result,
    UBYTE*          dat,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
dat	Pointer to the data transmitted to the SIM card
len	Length of the data transmitted to the SIM card (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x9Exx	length 'XX' of the response data given in case of a SIM data download error
0x9Fxx	Length 'XX' of the response data
0x9300	SIM Application Toolkit is busy. Command cannot be executed at present, further normal commands are allowed
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem
0x9404	File ID not found/Pattern not found

Description

Used by Network or ME to transfer data download to the SIM in a transparent way for user.

This command is part of the set used by SIM Application Toolkit. This function is used to transfer data to the SIM Application Toolkit applications in the SIM. The ME completes the command parameters/data of the relevant command and sends the command to the SIM. The transmitted data is processed by the SIM in a specific way depending on the tag value in the command parameters.

A SIM or ME not supporting SIM Application Toolkit does not need to support this command.

3.2.23 SIM_Fetch

Definition:

```
USHORT SIM_Fetch
(
    UBYTE*          result,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
len	Length of required data (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x9240	Memory problem

Description

This function is used to transfer an Application Toolkit command from the SIM to the ME. The output data string containing an SIM Application Toolkit command for the ME.

This command is used by SIM Application Toolkit. It is similar in function to GET RESPONSE, in that it requests response parameters from the SIM, following a '91 XX' status response. The transmitted response data from the SIM is processed by the ME in a specific way depending on the tag value in the response parameters.

A SIM or ME not supporting SIM Application Toolkit does not need to support this command.

3.2.24 SIM_TerminalResponse

Definition:

```
USHORT SIM_TerminalResponse
(
    UBYTE*          result,
    UBYTE*          dat,
    UBYTE           len,
    UBYTE*          size
) ;
```

Parameters:

Name	Description
result	Pointer to the result buffer
dat	Pointer to the data (profile) transmitted to the SIM card
len	Length of required data (Parameter P3)
size	Pointer to the number of return bytes

Return values:

Name	Description
0x67xx	Incorrect parameter P3
0x6Dxx	Unknown instruction code given in the command
0x6Exx	Wrong instruction class given in the command
0x6Fxx	Technical problem with no diagnostic given
0x9000	Normal ending of the command
0x91xx	normal ending of the command, with extra information from the proactive SIM containing a command for the ME. Length 'XX' of the response data
0x920x	command successful but after using an internal update retry routine 'X' times
0x9240	Memory problem

Description

Used for ME to respond at a SIM toolkit command.

This command is part of the set used by SIM Application Toolkit. This function is used to transfer from the ME to the SIM the response to a previously fetched SIM Application Toolkit command. The ME completes the command parameters/data of the relevant command and sends the command to the SIM. The transmitted data is processed by the SIM in a specific way depending on the tag value in the command parameters.

A SIM or ME not supporting SIM Application Toolkit does not need to support this command.

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/>