



Technical Document

SIXTIES SAT COMMAND SUPPORT

HIGH LEVEL DESIGN

Document Number:	8462.744.04
Version:	1.0
Status:	Submitted
Approval Authority:	
Creation Date:	2004-Jul-30
Last changed:	2015-Mar-08 by Nick Holt TI Employee
File Name:	hld_SixTies_sat_cust.doc

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
2004-Jul-30	Nick Holt		0.1	Draft	1
2004-Jul-30	Nick Holt		1	Draft	2

Notes:

1. Initial version
2. Updates following review by Frank Kaiser and Thomas Lüttig

Table of Contents

1	Introduction	6
2	Requirements to support SixTies application suite	7
2.1	Proactive SIM Commands	7
2.2	SIM Events	7
2.3	Power Up	7
3	New Requirements	8
3.1	AT command AT%CUST	8
3.2	AT command AT%SATCC	8
3.3	Field cust_mode in the SIM_ACTIVATE_REQ primitive	9
3.4	Primitive from ACI to SIM : MMI_SIM_REFRESH_USER_RESP	9
4	%SAT commands in modem part	10
4.1	%SATC: Configuration for SIM application toolkit (SIM-Profile)	10
4.2	%SATE: Send SAT envelope command	11
4.3	%SATR: Send SAT command response	11
4.4	%SATI processing	12
4.4.1	Current Behaviour	12
4.4.2	Customized SixTies Proactive Command Behaviour	13
5	Terminal profile handling	14
6	Power on behavior with SixTies Application	19
7	Event Handling	20
7.1	Current Behaviour	20
7.1.1	Notifying the SIM of Supported events	20
7.1.2	Processing the Setup Event List Proactive Command	21
7.2	Modifications	21
7.2.1	Notifying the SIM of Supported events	21
7.2.2	Processing the Setup Event List Proactive Command	21
8	Refresh proactive SIM command Details	22
8.1	Refresh Types and Handling	22
8.2	Current Behaviour	22
8.2.1	Refresh - File Change Notification (FCN) Handling	22
8.2.2	Refresh – Sim Initialisation (SI) Handling	24
8.2.3	Refresh – Sim Reset (SR) Handling	24
8.3	Modifications	24
8.3.1	SIM Entity	24
8.3.2	ACI / MMI behaviour	25
9	Mobile Originated Call Handling	27
9.1	User Initiated Call Handling	27
9.1.1	Allowed with No Modifications	27
9.1.2	Allowed, with modifications (still a call)	28
9.1.3	Allowed, Modified to Send SS or Send USSD	28

9.1.4	Not Allowed	29
9.2	SetUpCall Proactive SIM Command Handling	29
9.2.1	Allowed, without modifications	30
9.2.2	Allowed, with modifications (still a call)	31
9.2.3	Allowed, Modified to Send SS/USSD	32
9.2.4	Not Allowed	33
10	Supplementary Service and USSD Handling	34
10.1	User Initiated SS or USSD Requests	34
10.2	SIM SendSS or SendUSSD Requests	34
11	Mobile Originated Short Message Handling	35
11.1	User Initiated SMS Handling	35
11.1.1	Allowed with No Modifications	35
11.1.2	Allowed, with Modifications	36
11.1.3	Not Allowed	36
11.2	SendSm Proactive SIM Command Handling	37
11.2.1	Allowed, with no modifications	38
11.2.2	Allowed, with modifications	39
11.2.3	Not Allowed	40
A.	Appendices	41
B.	Acronyms	41
C.	Glossary	41

List of Figures and Tables

FIGURE 1 : PROACTIVE COMMAND PROCESSING CHART	13
FIGURE 2 : POWER UP SEQUENCE	19
FIGURE 3 : SIM TOOLKIT EVENT SUPPORT WITHIN THE ME	20
FIGURE 4 : REFRESH (FCN)	22
FIGURE 5 : REFRESH (FCN) MESSAGE HANDLING WITHIN THE MODEM PART	23
FIGURE 6 : REFRESH (FCN) : FLOWCHART	23
FIGURE 7 : REFRESH (FCN) MODIFIED FOR SIXTIES	25
FIGURE 8 : REFRESH FCN : UPDATED MSG SEQ. CHART FOR SIXTIES	26
FIGURE 9 : USER INITIATED MO CALL (NO MODIFICATIONS)	27
FIGURE 10 : USER INITIATED MO CALL (ALLOWED WITH MODIFICATIONS)	28
FIGURE 11 : USER INITIATED MO CALL (MODIFIED TO SS OR USSD REQUEST)	28
FIGURE 12 : USER INITIATED MO CALL (NOT ALLOWED)	29
FIGURE 13 : SIM INITIATED MO CALL (NO MODIFICATIONS)	30
FIGURE 14 : SIM INITIATED MO CALL (ALLOWED WITH MODIFICATIONS)	31
FIGURE 15 : SIM INITIATED MO CALL (MODIFIED TO SS OR USSD REQUEST)	32
FIGURE 16 : SIM INITIATED MO CALL (NOT ALLOWED)	33
FIGURE 17 : USER INITIATED SHORT MESSAGE (NO MODIFICATIONS)	35
FIGURE 18 : USER INITIATED SHORT MESSAGE (ALLOWED WITH MODS)	36
FIGURE 19 : USER INITIATED SHORT MESSAGE (NOT ALLOWED)	36
FIGURE 20 : SIM INITIATED SHORT MESSAGE (NO MODIFICATIONS)	38
FIGURE 21 : SIM INITIATED SHORT MESSAGE (ALLOWED WITH MODS)	39
FIGURE 22 : SIM INITIATED SHORT MESSAGE (NOT ALLOWED)	40

List of References

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

1 Introduction

This document provides the high-level design overview of changes required to implement SixTies application specific handling of certain proactive SIM commands and SIM events. Currently some of these requested proactive commands are handled by the modem part only.

Additional AT%... commands will be implemented in order to:

- indicate that the SixTies application is being used and that it is connected as MMI. In this case the affected proactive commands will be forward to the SixTies application via %SATI command. (AT%*CUST*)
- Disable or Enable Call or Short Message Control by SIM (AT%*SATCC*)

Furthermore, the SixTies will deliver a profile list to the modem part, which will compile a 'mixed profile list' to be sent to the SIM. The 'mixed profile list' will be created by

- Masking the SixTies profile list to ensure that only the agreed events will be supported in the event list.
- Performing a logical OR of the modem part profile list with the *masked* SixTies profile list.

2 Requirements to support SixTies application suite

2.1 Proactive SIM Commands

The following proactive commands were previously supported either entirely, or partly, within the ACI without any involvement from the MMI. However, for the SixTies, the commands must be forwarded to the MMI, which will perform any required processing.

- SendSm
- SendSs
- SendUssd
- SetUpCall
- Refresh (support currently in ACI, SMS and MM)
- SendDtmf

Previously the SetUpEventList proactive command was supported in both the ACI and in the MMI, and the handling of this command will continue to be treated in exactly the same manner as the current functionality meets the requirements.

2.2 SIM Events

Only the following events may be forwarded to SixTies in contrast to the current implementation:

- 'UserActivity' (Class3)
- 'IdleScreen' (Class3)
- 'Language Selection' (Class3)

Only these three Events will be supported from SixTies, any other events listed in the SixTies profile list will be masked out, and will be reported as unsupported. Eg. If the SixTies application reported that it supported the Browser Termination event, then the profile list will be masked to reset that bit to 0. Other events supported within the ACI will continue to be supported

2.3 Power Up

During startup, the MMI will need to be able to indicate that it is a SixTies application and that the special processing outlined in this high-level design and in the Requirements document is to apply. For this purpose a new AT command AT% CUST is to be provided.

3 New Requirements

3.1 AT command AT%CUST

The purpose of the AT%CUST command is to provide a mechanism for customer specific modifications to be applied in the ACI and other entities, without impacting on the generic nature of the software. The modifications required for Atelier SixTies include:

- Modifications to the standard MMI profile handling for Event Management
- Activation of special handling for the refresh proactive SIM commands
- Suspension of the ACI management of the SetUpCall proactive SIM command
- Suspension of the ACI management of the SendSS proactive SIM command
- Suspension of the ACI management of the SendUSSD proactive SIM command
- Suspension of the ACI management of the SendSm proactive SIM command
- Suspension of the ACI management of the SendDTMF proactive SIM command

With this command the modem part switches to another mode with different SAT behavior than before. This AT command must be received in the modem part earlier than either the AT%SATC command or the AT+CFUN command.

Syntax:

Command	Possible response(s)
%CUST=<n>	
%CUST?	%CUST: <n>
%CUST=?	%CUST: (list of supported <n>s)

Defined Values:

<n>: '0' → Default value, normal behaviour to apply
 '1' → SixTies application as MMI, special behaviour to apply

Other values of <n> may be added in the future to provide further customizations for specific MMI's

3.2 AT command AT%SATCC

The purpose of the AT%SATCC command is to allow the MMI to enable or disable Call Control By SIM, or the Short Message Control By SIM, functionality within the modem part. While this provides for a way in which the MMI may contravene the ETSI GSM SIM Application Toolkit Specifications, that is not the intention. However the conditions under which this command is used is left to the discretion of the MMI producer.

Syntax:

Command	Possible response(s)
%SATCC=<n>	
%SATCC?	%SATCC: <n>

Defined Values:

<n>: '0' → Disable the Call/Short Message Control By SIM mechanism in the modem part
 '1' → Enable the Call/Short Message Control By SIM mechanism in the modem part.

Note: *Enabling Call Control By SIM where it is not supported in the SIM Service Table will have no effect. It will remain inactive.*

3.3 Field `cust_mode` in the `SIM_ACTIVATE_REQ` primitive

The field `cust_mode` passes the customization value of the ACI channel that starts the SIM and places the handset into Fully Functional mode. This value will then apply to all processing within the ACI and SIM regardless of whether a different customization mode is requested on another channel. The value is a UBYTE and may contain the values listed in Para 3.1

3.4 Primitive from ACI to SIM : `SIM_REFRESH_USER_RES`

This primitive will be sent from the ACI to the SIM on receipt of a User Response from the SixTies application as a result of a Refresh Proactive SIM Command. See Para 8 for further implementation details.

Primitive Fields:

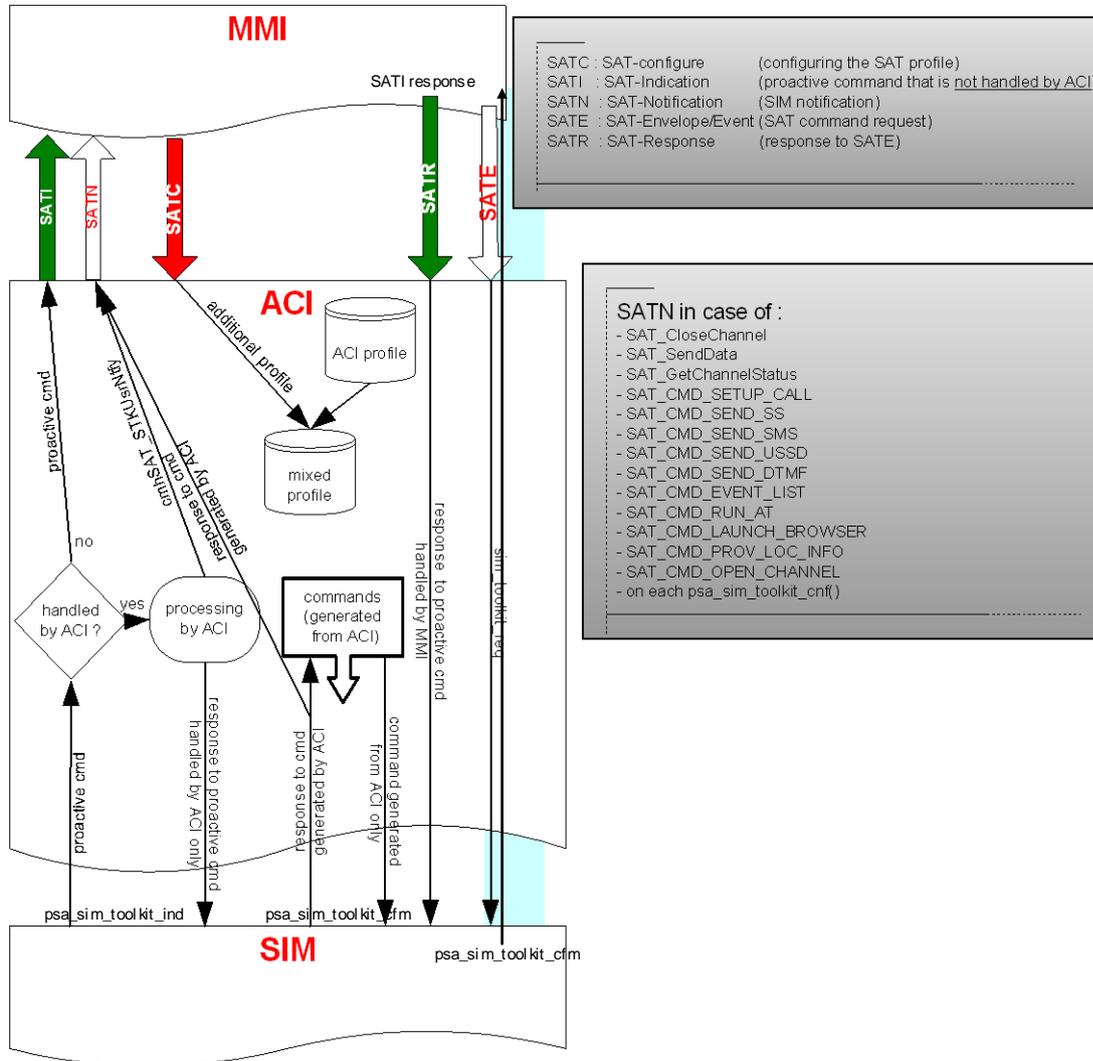
 BOOL `user_resp`
 STRUCT `stk_cmd`

'`user_accepts`' will be TRUE if the user has accepted the refresh command, FALSE if they have rejected it.

'`stk_cmd`' will be valid only if the '`user_accepts`' field is set to FALSE (ie the User has rejected the request), at which time the `stk_cmd` field will contain the Terminal Response to be sent to the SIM.

4 %SAT commands in modem part

Following picture shows the standard processing of SAT<x> commands and corresponding returns.



4.1 %SATC: Configuration for SIM application toolkit (SIM-Profile)

This command is used to indicate to the SIM the features that the ME is capable of.

Syntax:

Command	Possible response(s)
%SATC=<n>,<satProfile>	
%SATC?	%SATC: <n>,<satProfile>
%SATC=?	%SATC: (list of supported <n>s),(<profileLen>)

Defined Values

<n> : (Parameter sets/shows the result code presentation status in the TA)
 <satProfile> : String type; SIM application toolkit profile (hexadecimal format; refer +CSCS) starting with first byte of the profile.

Furthermore see '8415_052.doc' Chapter 2.6.

4.2 %SATE: Send SAT envelope command

This command provides the possibility to send a command to the SIM, using the envelope mechanism of SIM application toolkit as described in GSM 11.14.

Syntax:

Command	Possible response(s)
%SATE=<satCmd>	%SATE: <satRsp>
%SATE=?	

Defined Values:

<satCmd>: String type; SIM application toolkit command (hexadecimal format; refer +CSCS) starting with command tag.

<satRsp>: String type; SIM application toolkit response (hexadecimal format; refer +CSCS) starting with first byte of response data.

4.3 %SATR: Send SAT command response

This command provides the possibility to send a response to previous received SAT command. If a SIM application toolkit command was indicated to TE using the result %SATI: <satCmd>, the TE should send an appropriate response using the %SATR command. If <satRsp> is present the contents is converted and send directly to SIM. The coding of the SIM response is the task of TE, no checking is done by ME.

Command	Possible response(s)
%SATR=<satRsp>	
%SATR=?	

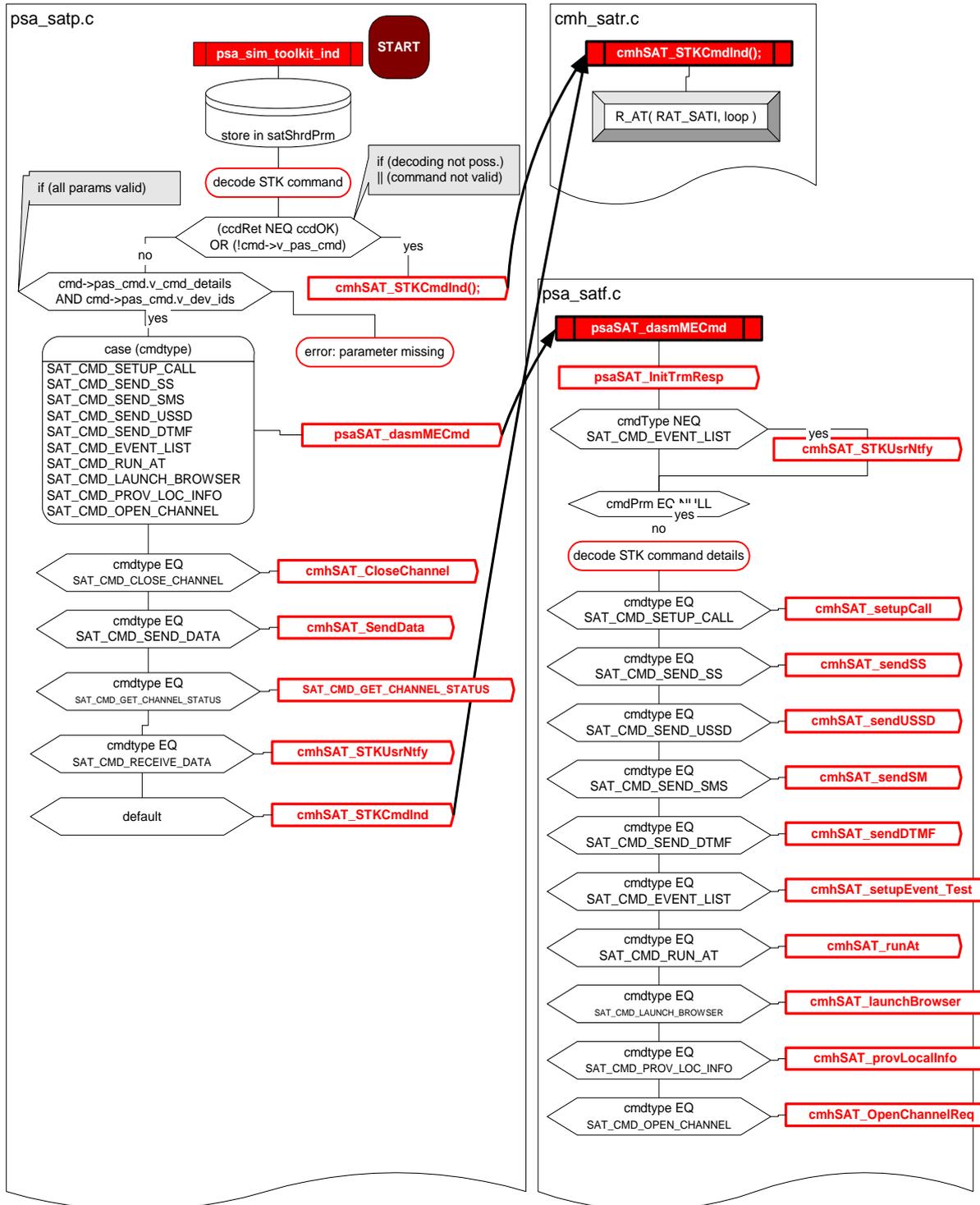
Defined Values:

<satRsp> : String type; SIM application toolkit response (hexadecimal format; refer +CSCS) starting with first byte of response data.

4.4 %SATI processing

4.4.1 Current Behaviour

All proactive SIM commands go to the ACI_pei 'gate'. The function `psa_sim_toolkit_ind()` initially receives all proactive commands. The current behavior looks like as following picture.



4.4.2 Customized SixTies Proactive Command Behaviour

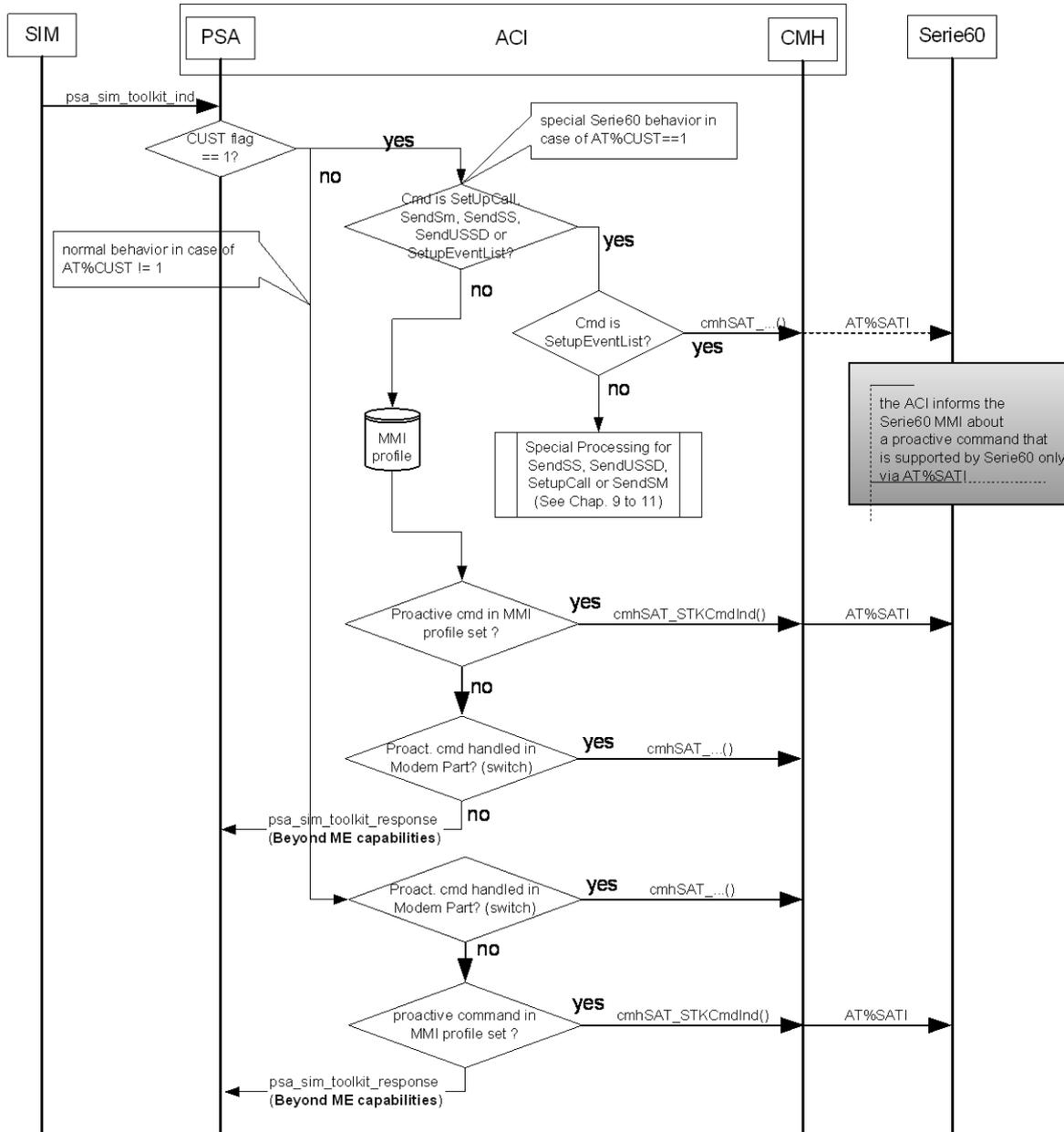


Figure 1 : Proactive Command processing Chart

If the CUST flag is set to '0' the function `psa_sim_toolkit_ind()` will continue processing as it currently does, otherwise the following functionality will apply. The function will initially determine whether the requested command is `SendSS`, `SendUSSD`, `SendSM`, `SetupCall` or `SetupEventList`. If so, then for `SetupEventList`, the proactive command will be processed in the normal manner, being processed in the ACI before being forwarded to the MMI if necessary. For the `SendSS`, `SendUSSD`, `SendSM` and `SetUpCall` commands the processing will be as shown in Paras 9 through 11. For all other commands, the function will determine whether the proactive command is support in MMI, and if so it will be forwarded in an `AT%SATI` indication otherwise it will be processed within the ACI with an `AT%SATN` notification being sent to the MMI if required.

5 Terminal profile handling

The STK profile is stored in the shared parameter variables for the SIM card:

simShrdPrm.setPrm[SrcId].STKprof

The profile contains at maximum 20*8 entries. See at GSM 11.14 for more details. Some entries are not predefined. The next table shows the entire supported profile table supported from modem part and the according MMI profile mask.

The resulting Terminal Profile is created with the following formula:

simShrdPrm.setPrm[SrcId].actualMMIPrf = (actualMMIPrf1 & satMaskMMIPrf1)
resultingPrf1 = ***simShrdPrm.setPrm[SrcId].actualMMIPrf*** | satDefPrf1

The 'actualMMIPrf' is sent by the MMI in an AT%SATC command and merged directly into the SIM Toolkit profile. It is also stored in the global shared parameters at *simShrdPrm.setPrm[SrcId].actualMMIPrf* and used in order to determine which proactive command must be sent to MMI via AT%SATI.

Please note: If the customer mode is set to '1' (SixTies) by AT%CUST=1, the 'satMaskMMIPrf1' includes only the requested events for SixTies.

Note:

In the following table the 'green' fields are "agreed" by Atelier/Nokia concerning some E-Mails received from Atelier/Nokia. The purple fields are the event fields. The 'red' fields will be changed in case of SixTies mode is on (AT%CUST=1).

Please agree the other fields also !!!

Service Name (define)	Value	Service explanation	Behavior AT%CUST=0 is set			Behavior AT%CUST=1 is set		
			Support in mode m part (satDefPrfl)	MMI profile (actualMMIPrfl)	MMI mask (satMaskMMIPrfl)	Support in mode m part (satDefPrfl)	MMI profile (actualMMIPrfl)	MMI mask (satMaskMMIPrfl)
SAT_TP1_PRF_D NL	(0x1)	Profile dow nload (redundant definition)	X		X	X		X
SAT_TP1_SMS_ DNL	(0x2)	SMS-PP data dow nload						
SAT_TP1_CB_DN L	(0x4)	Cell Broadcast data dow nload	X			X		
SAT_TP1_MENU _SEL	(0x8)	Menu selection			X			X
SAT_TP1_9E_XX	(0x10)	'9E XX' response code for SIM data dow nload error						
SAT_TP1_TIMER _EXP	(0x20)	Timer expiration (Class 3)						
SAT_TP1_CC_US SD	(0x40)	USSD string data object supported in Call Control	X			X		
SAT_TP1_CC_O N_REDIAL	(0x80)	Call Control by SIM alw ays performed on automatic redial						
SAT_TP2_CMD_ RES	(0x1)	Command result (generic)	X		X	X		X
SAT_TP2_CC	(0x2)	Call Control by SIM	X		X	X		X
SAT_TP2_CC_CE LL_ID	(0x4)	Cell Identity included in Call Control by SIM	X			X		
SAT_TP2_MOSM C	(0x8)	MO SM Control by SIM (Class 3)	X		X	X		X
SAT_TP2_ALPHA _ID	(0x10)	Alpha identifier according to clause 9			X			X
SAT_TP2_UCS2_ ENTRY	(0x20)	UCS2 entry supported			X			X
SAT_TP2_UCS2_ DSPL	(0x40)	UCS2 display supported			X			X
SAT_TP2_DSPL_ EXT	(0x80)	extended display length supported (Class 3)			X			X
SAT_TP3_DSPL_ TXT	(0x1)	Proactive SIM: Display Text		X	X		X	X
SAT_TP3_GET_I NKEY	(0x2)	Proactive SIM: Get Inkey		X	X		X	X
SAT_TP3_GET_I NPUT	(0x4)	Proactive SIM: Get Input		X	X		X	X
SAT_TP3_MORE _TIME	(0x8)	Proactive SIM: More Time						
SAT_TP3_PLAY_ TONE	(0x10)	Proactive SIM: Play Tone		X	X		X	X
SAT_TP3_POLL_I TV	(0x20)	Proactive SIM: Poll Interval						
SAT_TP3_POLL_ OFF	(0x40)	Proactive SIM: Polling Off						
SAT_TP3_REFRE SH	(0x80)	Proactive SIM: Refresh	X	X (via SATI)	X	X	X (via SATI)	X

Service Name +(define)	Value	Service explanation	Behavior AT%CUST=0 is set			Behavior AT%CUST=1 is set		
			Support in modem part (satDefPrfl)	MMI profile (actualMMIPrf)	MMI mask (satMaskMMIPrf)	Support in modem part (satDefPrfl)	MMI profile (actualMMIPrf)	MMI mask (satMaskMMIPrf)
SAT_TP4_SEL_IT EM	(0x1)	Proactive SIM: Select Item		X	X		X	X
SAT_TP4_SEND_ SMS	(0x2)	Proactive SIM: Send SM	X			X	X	
SAT_TP4_SEND_ SS	(0x4)	Proactive SIM: Send SS	X		X	X	X	X
SAT_TP4_SEND_ USSD	(0x8)	Proactive SIM: Send USSD (Class 3)	X		X	X	X	X
SAT_TP4_SETUP_ CALL	(0x10)	Proactive SIM: Set up Call	X		X	X	X	X
SAT_TP4_SETUP_ MENU	(0x20)	Proactive SIM: Set up Menu		X	X		X	X
SAT_TP4_PLI_PL MN_IMEI	(0x40)	Proactive SIM: Provide Local Informaton (PLMN, Cell ID, IMEI)						
SAT_TP4_PLI_N MR	(0x80)	Proactive SIM: Provide Local Informaton (NMR)						
SAT_TP5_EVENT_ LIST	(0x1)	Proactive SIM: Set up Event List (Class 3)	X	X	X	X	X	X
SAT_TP5_MT_CA LL	(0x2)	Event: Mobile Terminated Call (Class 3)	X			X		
SAT_TP5_CALL_ CONN	(0x4)	Event: Call connected (Class 3)	X			X		
SAT_TP5_CALL_ DISC	(0x8)	Event: Call disconnected (Class 3)	X			X		
SAT_TP5_LOC_S TATUS	(0x10)	Event: Location status (Class 3)	X			X		
SAT_TP5_USER_ ACT	(0x20)	Event: User activity (Class 3)			X		X	X
SAT_TP5_SCR_A VAIL	(0x40)	Event: Idle screen available (Class 3)			X		X	X
SAT_TP5_CDR_S TATUS	(0x80)	Event: Card reader status (Class a)						
SAT_TP6_LANG_ SEL	(0x1)	Event: Language Selection (Class 3)					X	X
SAT_TP6_BROW S_TERM	(0x2)	Event: Browser Termination (Class c)			X			No support
SAT_TP6_DATA_ AVAIL	(0x4)	Event: Data available (Class e)						
SAT_TP6_CHAN NEL_STAT	(0x8)	Event: Channel status (Class e)						
SAT_TP7_PW_O N_CD	(0x1)	Proactive SIM: Pow er on Card (Class a)						
SAT_TP7_PW_O FF_CD	(0x2)	Proactive SIM: Pow er off Card (Class a)						
SAT_TP7_PF_CD _ADPU	(0x4)	Proactive SIM: Perform Card ADPU (Class a)						
SAT_TP7_GET_C DR_STATUS	(0x8)	Proactive SIM: Get Card Reader Status (Class a)						
SAT_TP7_GET_C	(0x10)	Proactive SIM:						

Service Name +(define)	Value	Service explanation	Behavior AT%CUST=0 is set			Behavior AT%CUST=1 is set		
			Support in modem part (satDefPrfl)	MMI profile (actualMMIPrfl)	MMI mask (satMaskMMIPrfl)	Support in modem part (satDefPrfl)	MMI profile (actualMMIPrfl)	MMI mask (satMaskMMIPrfl)
DR_IDENT		Get Card Reader Identifier (Class a)						
SAT_TP8_TMNG_ST	(0x1)	Proactive SIM: Timer Management (start, stop) (Class 3)						
SAT_TP8_TMNG_VAL	(0x2)	Proactive SIM: Timer Management (get current value) (Class 3)						
SAT_TP8_PLI_DT	(0x4)	Proactive SIM: Provide Local Information (date, time, tz) (Class 3)	X			X		
SAT_TP8_BIN_GET_INKEY	(0x8)	Binary choice in Get Inkey (Class 3)			X			X
SAT_TP8_IDLE_TXT	(0x10)	Proactive SIM: Set up Idle Mode Text (Class 3)		X	X		X	X
SAT_TP8_AT_CMD	(0x20)	Proactive SIM: Run AT command (Class b)	X			X		
SAT_TP8_A12_SETUP_CALL	(0x40)	2nd alpha identifier in Set up Call (Class 3)			X			X
SAT_TP8_CCP2_CC	(0x80)	2nd capability configuration parameter with Call Control (Class 3)	X			X		
SAT_TP9_SUST_DSP_TXT	(0x1)	Sustained Display Text (Class 3)			X			X
SAT_TP9_DTMF_CMD	(0x2)	Send DTMF Command (Class 3)	X			No support	X	X
SAT_TP9_BCCH_COD	(0x4)	BCCH coding according to subclause 12.19 (Class 3)						
SAT_TP9_PLI_LANG	(0x8)	Proactive SIM: Provide Local Information (Language) (Rel. 99)						
SAT_TP9_PLI_TIMING_ADV	(0x10)	Proactive SIM: Provide Local Information (Timing Advance) (Rel. 99)						
SAT_TP9_LANG_NOTIFY	(0x20)	Proactive SIM: Language Notification (Class 3)						
SAT_TP9_LAUNCH_BROWSER	(0x40)	Proactive SIM: Launch Browser (Class c)		X	(X)		X	(X)
SAT_TP10_SOFTKEY_SELECT_ITEM	(0x1)	Soft key support for SELECT ITEM			X			X
SAT_TP10_SOFTKEY_SETUP_MENU	(0x2)	Soft key support for SET UP MENU			X			X
SAT_TP11_MAX_NR_SOFTKEY	(0xff)	Maximum number of soft keys			X			X

Service Name +(define)	Value	Service explanation	Behavior AT% <i>CUST</i> =0 is set			Behavior AT% <i>CUST</i> =1 is set		
			Support in modem part (satDefPrfl)	MMI profile (actualMMIPrf)	MMI mask (satMaskMMIPrf)	Support in modem part (satDefPrfl)	MMI profile (actualMMIPrf)	MMI mask (satMaskMMIPrf)
		available						
SAT_TP12_OPEN_CHANNEL	(0x1)	Proactive SIM: OPEN CHANNEL (Class e)	(X)			(X)		
SAT_TP12_CLOSE_CHANNEL	(0x2)	Proactive SIM: CLOSE CHANNEL (Class e)	(X)			(X)		
SAT_TP12_RECEIVE_DATA	(0x4)	Proactive SIM: RECEIVE DATA (Class e)						
SAT_TP12_SEND_DATA	(0x8)	Proactive SIM: SEND DATA (Class e)						
SAT_TP12_GET_CHANNEL_STAT	(0x10)	Proactive SIM: GET CHANNEL STATUS (Class e)	(X)			(X)		
SAT_TP13_CSD_SUPP_BY_ME	(0x1)	CSD supported by ME (Class e)	(X)			(X)		
SAT_TP13_GPRS_SUPP_BY_ME	(0x2)	GPRS supported by ME (Class e)	(X)			(X)		
SAT_TP13_NR_OF_CHAN_SUPP	(0xe0)	Number of channels supported by ME (Class e)	(X)			(X)		
SAT_TP14_NR_OF_CHAR_DSPL_DWN	(0x1f)	Number of characters supported down the ME display			X			X
SAT_TP14_SCRN_SIZE_PARAM	(0x80)	Screen sizing parameters supported by ME			X			X
SAT_TP15_NR_OF_CHAR_DSPL_ACRS	(0x7f)	Number of characters supported across the ME display						
SAT_TP15_VAR_SIZE_FONT	(0x80)	Variable size fonts supported						
SAT_TP16_DSPL_RESIZE	(0x1)	Display can be resized			X			X
SAT_TP16_TEXT_WRAP	(0x2)	Text wrapping supported			X			X
SAT_TP16_TEXT_SCROLL	(0x4)	Text scrolling supported			X			X
SAT_TP16_WIDTH_RDCT_MENU	(0xe0)	Width reduction when in Menu			X			X
SAT_TP17_BEARER_IND_SUPP_TCP	(0x1)	Bearer independent protocol supported TCP (Class e)						
SAT_TP17_BEARER_IND_SUPP_UDP	(0x2)	Bearer independent protocol supported UDP (Class e)	(X)			(X)		
SAT_TP19_PROTOCOL_VERSION	(0xf)	Protocol version coded as TIA/EIA 136-123						

Table 1

The SIM is informed about the profile via `PSENDX(SIM, sim_activate_req);` in `psaSIM_ActivateSIM()` [`psa_sims.c`]. This function is called from `AT+CFUN`, which means the 'sum'-profile (MMI- & modem part-profile) must be available before `AT+CFUN` is called.

6 Power on behavior with SixTies Application

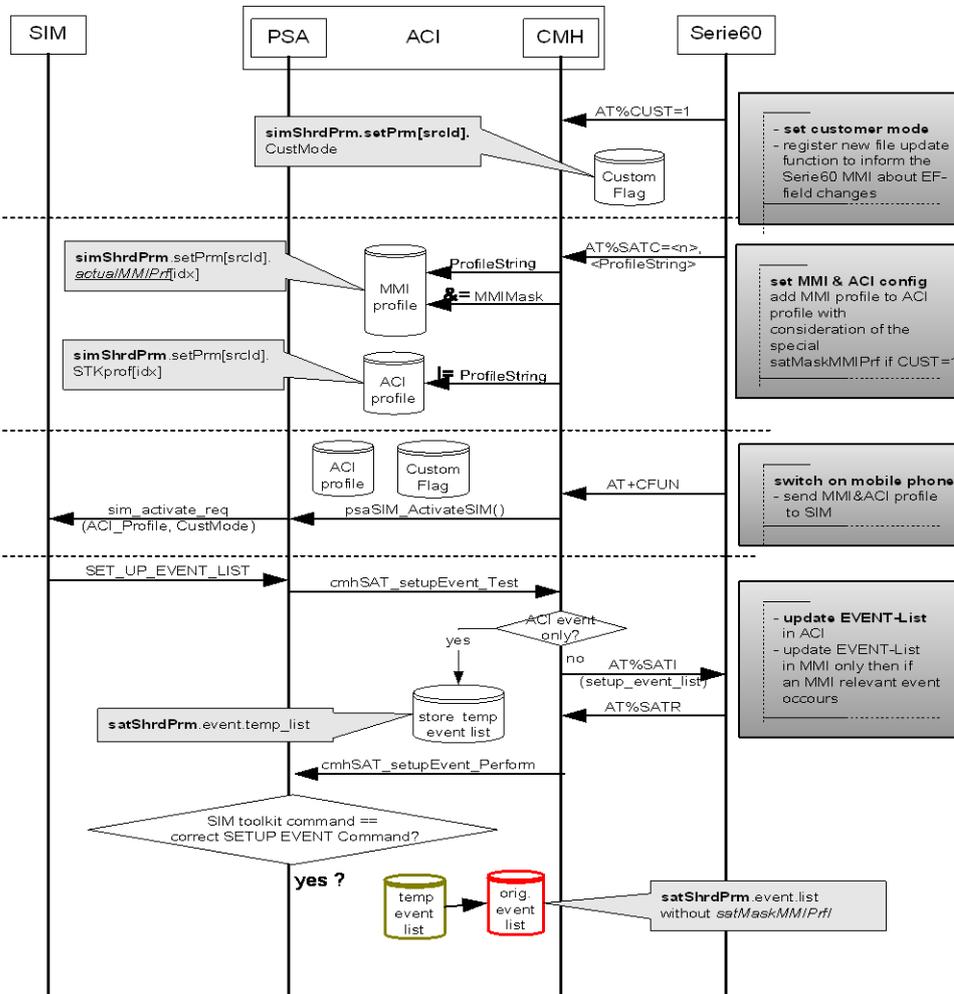


Figure 2 : Power Up Sequence

7 Event Handling

7.1 Current Behaviour

SIM Event reporting has two functional parts:

- Notifying the SIM of the Events which are supported by the ME
- Processing the SetUp Event List proactive command from the SIM

This of course ignores the processing required in order to recognize when one of the subscribed events has occurred, which is implementation specific and beyond the scope of this document.

7.1.1 Notifying the SIM of Supported events

The ME on startup performs the first part, notifying the SIM of which events can be supported. The supported events are passed to the SIM as a bit mask in the TERMINAL PROFILE. However, not all events are necessarily supported, and, of the events that are supported, some may be supported by the MMI, while others are supported only in the ACI. (see the diagram below)

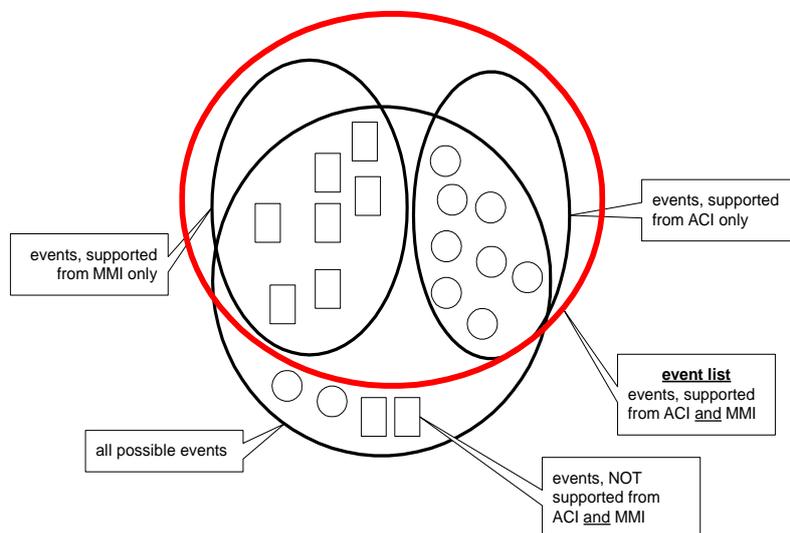


Figure 3 : SIM Toolkit Event Support within the ME

On startup, the MMI sends its terminal Profile to the ACI. The ACI combines its own Terminal Profile, with that provided by the MMI in order to create a superset of Commands and Events supported by them both. This is then sent to the SIM.

7.1.2 Processing the Setup Event List Proactive Command

When the SIM sends a SetUpEventList proactive command, it is received by the ACI. The ACI currently parses the list of requested events, and configures triggers for those events it supports. If there are any events, which the ACI processes, that are duplicated in the list then it shall, in accordance with GSM 11.14, reject the entire list and send a “Command beyond ME’s capabilities” response and the command will NOT be forwarded to the MMI. If the ACI has successfully processed the list of events, without failure, then it will regenerate the requested list, but it will remove the events supported by the ACI. This, possibly smaller, list will be forward to the MMI for processing. In the MMI, the list is checked for any events that are unsupported or any duplicated events, and if either is found the command rejected with a “Command beyond ME’s capabilities” response. Otherwise the event triggers are configured and a “Command Completed” response returned back to the ACI.

When the ACI receives the response from the MMI, if it is an error, any configured events will be cleared before the error is returned to the SIM Entity. Otherwise, the “Command Completed” response is forwarded on to the SIM Entity.

7.2 Modifications

7.2.1 Notifying the SIM of Supported events

When the MMI Terminal Profile is received in the ACI, if the %CUST value is 1 then the supported events should be masked to ensure that no other events are supported by the MMI other than ‘User Activity’, ‘Idle Screen’ or ‘Language Selection’. Once this has been done, all other processing will continue as before.

7.2.2 Processing the Setup Event List Proactive Command

There are no changes required to the current handling.

8 Refresh proactive SIM command Details

8.1 Refresh Types and Handling

In contrast to the present behavior, SixTies wants to be notified of all refresh processing:

- SIM initialization	0x00	soft reset
- File Change notification	0x01	soft reset
- SIM initialization and file change notification	0x02	soft reset
- SIM initialization and full file change notification	0x03	soft reset
- SIM Reset	0x04	hard reset

Additionally SixTies wants to have the possibility to allow the user to confirm or reject the refresh, which means, if the user rejects the refresh (including a SIM reset) a Terminal Response will be returned to the SIM with the value "ME currently unable to process command"

For each refresh type the original PDU must be delivered, in a SIM_TOOLKIT_IND, to the modem part and forwarded, in a %SATI indication, to the SixTies MMI.

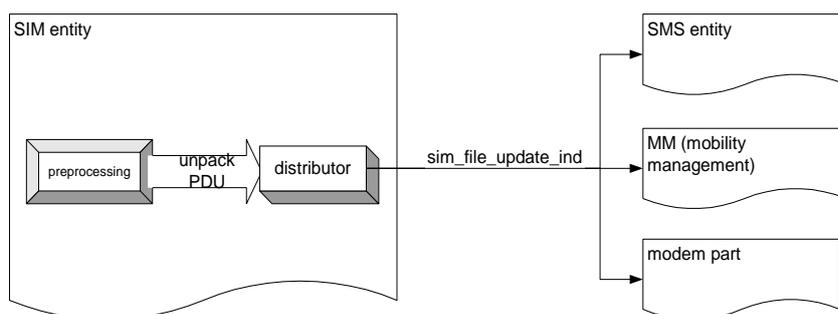
8.2 Current Behaviour

8.2.1 Refresh - File Change Notification (FCN) Handling

8.2.1.1 SIM Entity

On receipt of a Refresh FCN Command, the SIM Entity currently parses the command, and sends a `sim_file_update_ind` primitive to the ACI, MM and SMS Entities. Each Entity receives the primitive and performs the necessary processing to update the cached SIM data, and returns a `sim_toolkit_resp` to the SIM Entity. The SIM Entity collates the responses and on receipt of all three returns the appropriate response to the SIM, along with an empty `sim_toolkit_ind` to the other 3 Entities to indicate that the refresh processing has been completed.

The refresh command PDU is never sent to the ACI, MM or SMS Entities.



For a Refresh (Sim Initialisation) the primitive sent from the SIM Entity is a *sim_mmi_insert_ind*
 For a Refresh (Sim Reset) the primitive sent from the SIM Entity is a *sim_remove_ind*

Figure 4 : Refresh (FCN)

8.2.1.2 ACI / Modem Part

The following Message Sequence Chart and flow diagram represent the current interaction between the ACI and the SIM Entity, and the ACI behaviour. The SIM sends a `file_update_ind` primitive to the ACI, which refreshes the SIM fields it has cached and calls the function `'psaSAT_FUConfirm'` to notify the SIM entity with an appropriate Envelope Response. (Either Command Completed or Me currently unable to process command). The SIM Entity sends an empty SIM toolkit indication to signal that the process has completed.

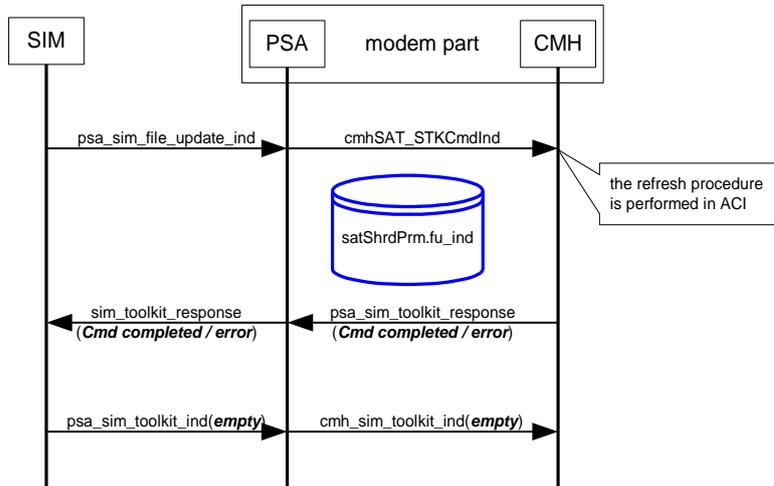


Figure 5 : Refresh (FCN) Message handling within the Modem Part

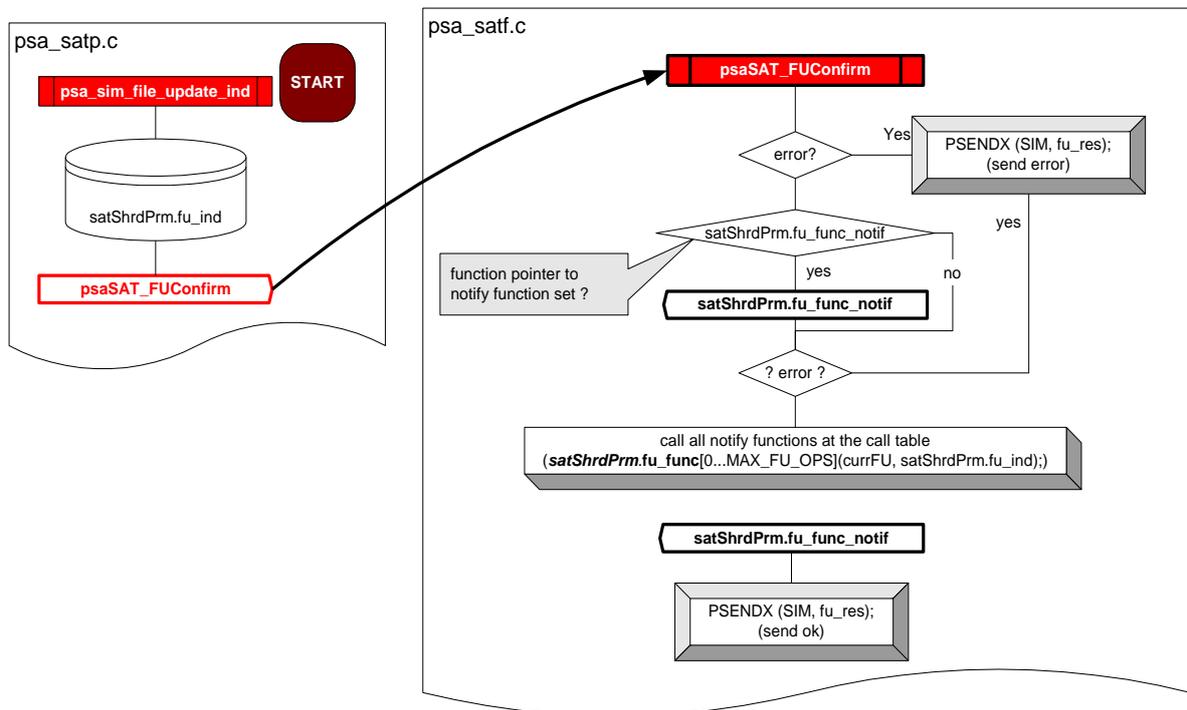


Figure 6 : Refresh (FCN) : Flowchart

8.2.2 Refresh – Sim Initialisation (SI) Handling

8.2.2.1 SIM Entity

On receipt of a Refresh SI Command, the SIM Entity currently parses the command, and sends a `sim_mmi_insert_ind` primitive to the ACI, MM and SMS Entities. The SIM Entity processing for the Refresh SI command differs from the FCN command in that, having sent the `sim_mmi_insert_ind` primitive the SIM Entity immediately sends a **Command Completed** Response Envelope to the SIM. Each Entity receives the primitive and performs its necessary processing, the ACI resets the entire system and processes the signal as if a new SIM has been inserted and verified. **No SIM toolkit response is sent from the ACI to the SIM Entity.**

The refresh command PDU is never sent to the ACI, MM or SMS Entities.
See Figure 4

8.2.2.2 ACI / Modem Part

The ACI treats the `sim_mmi_insert_ind` exactly the same regardless of whether the ME is starting up or whether it has been received as a Refresh SI Command. The ACI does not send a `sim_toolkit_response` to the SIM Entity on completion of the process.

8.2.3 Refresh – Sim Reset (SR) Handling

8.2.3.1 SIM Entity

On receipt of a Refresh SR Command, the SIM Entity currently parses the command, and sends a `sim_remove_ind` primitive to the ACI, MM and SMS Entities. No SIM Toolkit response is returned to the SIM, this is in accordance with GSM11.14. **No SIM toolkit response is received from the ACI.** The refresh command PDU is never sent to the ACI, MM or SMS Entities.
See Figure 4

8.2.3.2 ACI / Modem Part

The ACI treats the `sim_remove_ind` exactly the same regardless of whether the SIM has been removed or whether it has been received as a Refresh SR Command. The ACI does not send a `sim_toolkit_response` to the SIM Entity on completion of the process.

8.3 Modifications

8.3.1 SIM Entity

The modification to the SIM entity will be the same, regardless of the refresh type requested. If the `%CUST` value is 1, then the full refresh PDU will be sent to the ACI, to allow the user to be notified and to get a response as to whether the refresh may continue. The SIM entity will wait for a `SIM_REFRESH_USER_RES` primitive before continuing.

On receipt of the `SIM_REFRESH_USER_RES` primitive, the 'user_accepts' parameter will be read to determine whether the user has given permission for the refresh to continue. If the parameter is TRUE then the user has accepted the refresh command, and the processing will continue as for any other MMI, with the SIM entity sending one of `sim_file_update_ind`, `sim_mmi_insert_ind` or `sim_remove_ind` to the ACI, MM and SMS entities.

If the user rejects the refresh command then the SIM Entity will return an "ME currently unable to process command" Terminal Response to the SIM. The SIM will resend the Refresh Command to the SIM Entity, although the details of exactly how frequently are SIM specific.

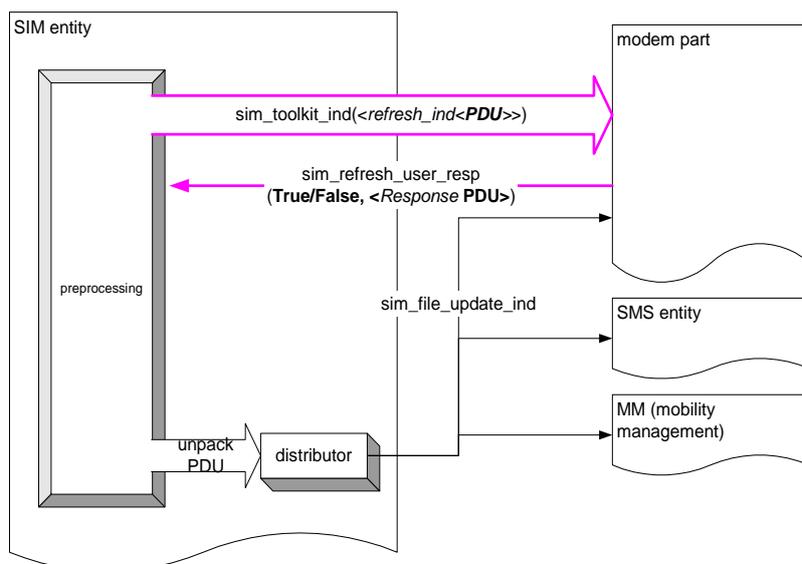
8.3.2 ACI / MMI behaviour

When the ACI receives a SIM Toolkit indication from the SIM Entity, the PDU is decoded to determine what type of proactive command has been received. If the command is a refresh command and the %CUST value is 1 then the full PDU will be sent to the MMI in a %SATI indication, and no further processing will be done until the response is received in a %SATR indication from the MMI. The response from the MMI will either be a “Command Complete” Envelope, which will indicate that the User has accepted the refresh command, or an “ME currently unable to process command” Envelope, which will indicate that the User has rejected the refresh command.

If the ACI receives a “Command Complete” Envelope then it will send the new SIM_REFRESH_USER_RES primitive to the SIM Entity with the ‘user_accepts’ parameter set to TRUE (the Response PDU will not be set), otherwise the same primitive will be sent but with the ‘user_accepts’ parameter set to FALSE, and the Response PDU from the MMI will be copied into the primitive.

All other ‘refresh’ behaviour in the ACI will remain the same.

These changes are reflected in the message sequence charts below (only the Refresh FCN command is shown).



For a Refresh (Sim Initialisation) the primitive sent from the SIM Entity is a *sim_mmi_insert_ind*
 For a Refresh (Sim Reset) the primitive sent from the SIM Entity is a *sim_remove_ind*

Figure 7 : Refresh (FCN) Modified for SixTies

9 Mobile Originated Call Handling

9.1 User Initiated Call Handling

The following sections show the current implementation for MO Call handling with regard to Call Control by SIM. They are included for completeness, although no modifications are required relating to this area of functionality.

At each SAT CC related confirm (sim_toolkit_cnf) an AT%SATN is sent. It includes the content of the sim_toolkit_cnf primitive (allowed, allowed with modifications, not allowed).

9.1.1 Allowed with No Modifications

In the following Message Sequence Chart, that the psaSAT_STKEnvelope (marked in red) was triggered by an ATD request from the MMI, causes the %SATN notification (marked in red) to be sent back to the MMI, however, the subsequent cmhSAT_ResCalCntrlBySIM and the resulting call setup (marked in blue) are handled entirely within the ACI.

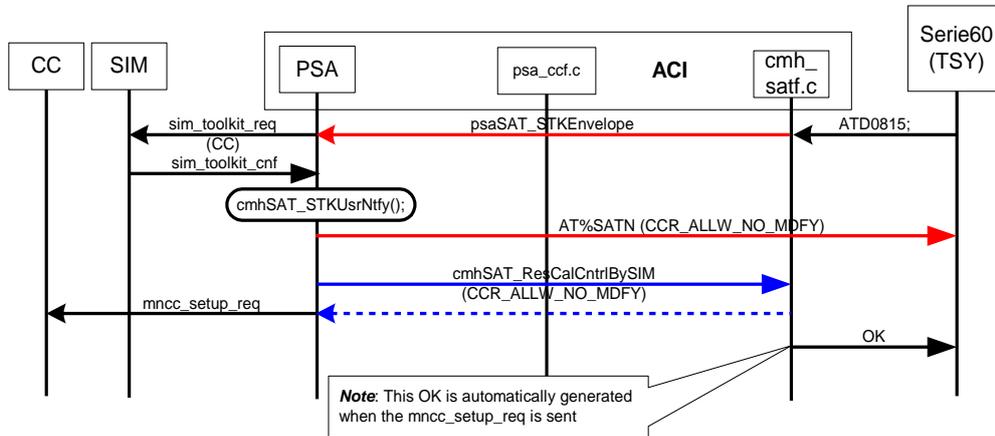


Figure 9 : User Initiated MO Call (No Modifications)

9.1.2 Allowed, with modifications (still a call)

In the following Message Sequence Chart, that the `psaSAT_STKEnvelope` (marked in red) was triggered by an ATD request from the MMI, causes the `%SATN` notification (marked in red) to be sent back to the MMI, however, the subsequent `cmhSAT_ResCalCntrlBySIM` and the resulting call setup (marked in blue) are handled entirely within the ACI.

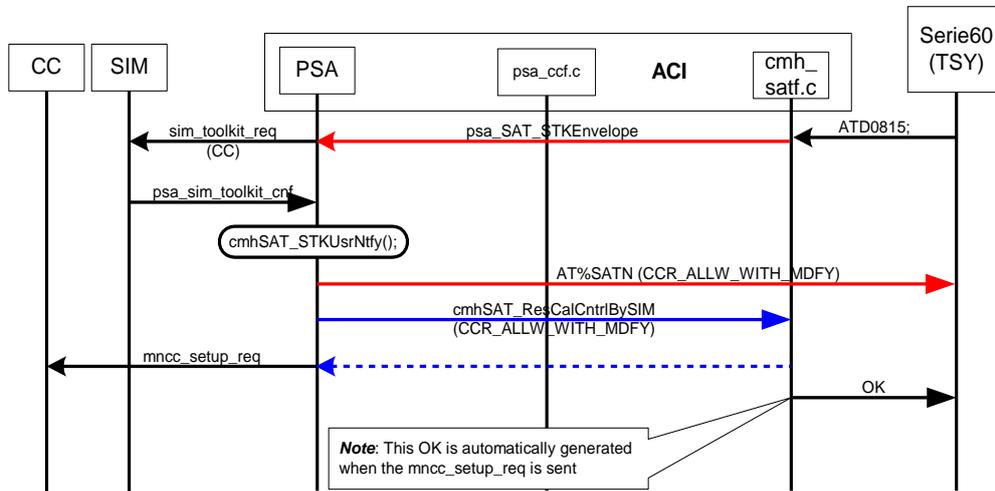


Figure 10 : User Initiated MO Call (Allowed with Modifications)

9.1.3 Allowed, Modified to Send SS or Send USSD

In the following Message Sequence Chart, that the `psaSAT_STKEnvelope` (marked in red) was triggered by an ATD request from the MMI, causes the `%SATN` notification (marked in red) to be sent back to the MMI, however, the subsequent `cmhSAT_ResCalCntrlBySIM` and the resulting SS or USSD request (marked in blue) are handled entirely within the ACI.

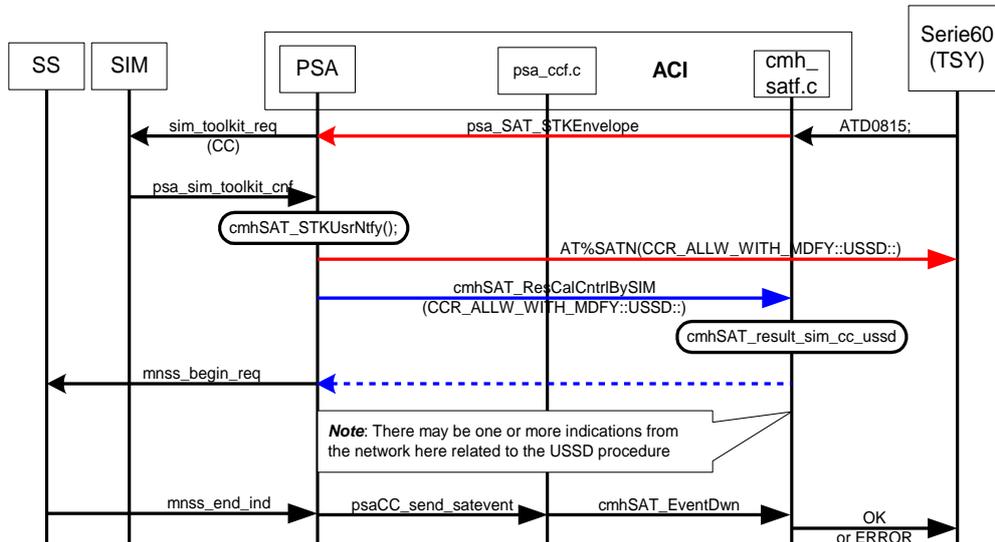


Figure 11 : User Initiated MO Call (Modified to SS or USSD Request)

In case of an error in the Send SS or Send USSD request, the response to the initial ATD command will be an error message.

9.1.4 Not Allowed

In the following Message Sequence Chart, that the `psaSAT_STKEnvelope` (marked in red) was triggered by an ATD request from the MMI, causes the `%SATN` notification (marked in red) to be sent back to the MMI, however, the subsequent `cmhSAT_ResCalCntrlBySIM` (marked in blue) is handled within the ACI.

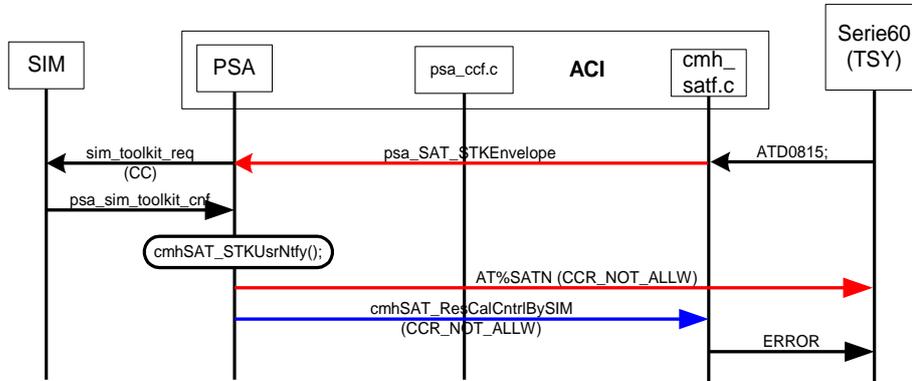


Figure 12 : User Initiated MO Call (Not Allowed)

9.2 SetUpCall Proactive SIM Command Handling

The expected behaviour, where the SetUpCall Command is handled by the MMI, would be that the SIM Toolkit PDU would be forwarded to the MMI in a `%SATI` indication, and an `ATD<number>` command would be expected as a result. This `ATD<number>` command would be subject to the Call Control By SIM procedure if it is active, with the MMI being notified of the result, but the necessary modifications being handled within the ACI as for a User Initiated call (See Para 9). The MMI would be responsible for the terminal response to the SIM indicating the outcome of the command.

For Atelier, the procedure is slightly different and is outlined below for each of the possible scenarios.

In the Message Sequence Charts below the following should be noted:

- All the `%SATI` commands described hold the PDU of the pending proactive command sent by the SIM.
- All the `%SATN` commands described hold the PDU of the result to the CC envelope cmd sent to the SIM.

9.2.1 Allowed, without modifications

The ACI captures the SetupCall Command and, provided Call Control By SIM is active, immediately sends a Call Control by SIM request to the SIM. The resultant response is then forwarded through to the MMI in a %SATN notification. The cmhSAT_ResCalCntrlBySIM, normally handled by the ACI, is forwarded to the MMI in a %SATI (SetupCall) indication. (It is important for the SixTies to receive the %SATN notification **before** it receives the %SATI indication).

On receipt of the %SATI indication the SixTies will send a %SATCC=0 command in order to disable the Call Control by SIM mechanism, and will follow it with an ATD command. Once the call setup is completed and the Terminal Response sent to the SIM, the Call Control By SIM will be re-activated with the command %SATCC=1.

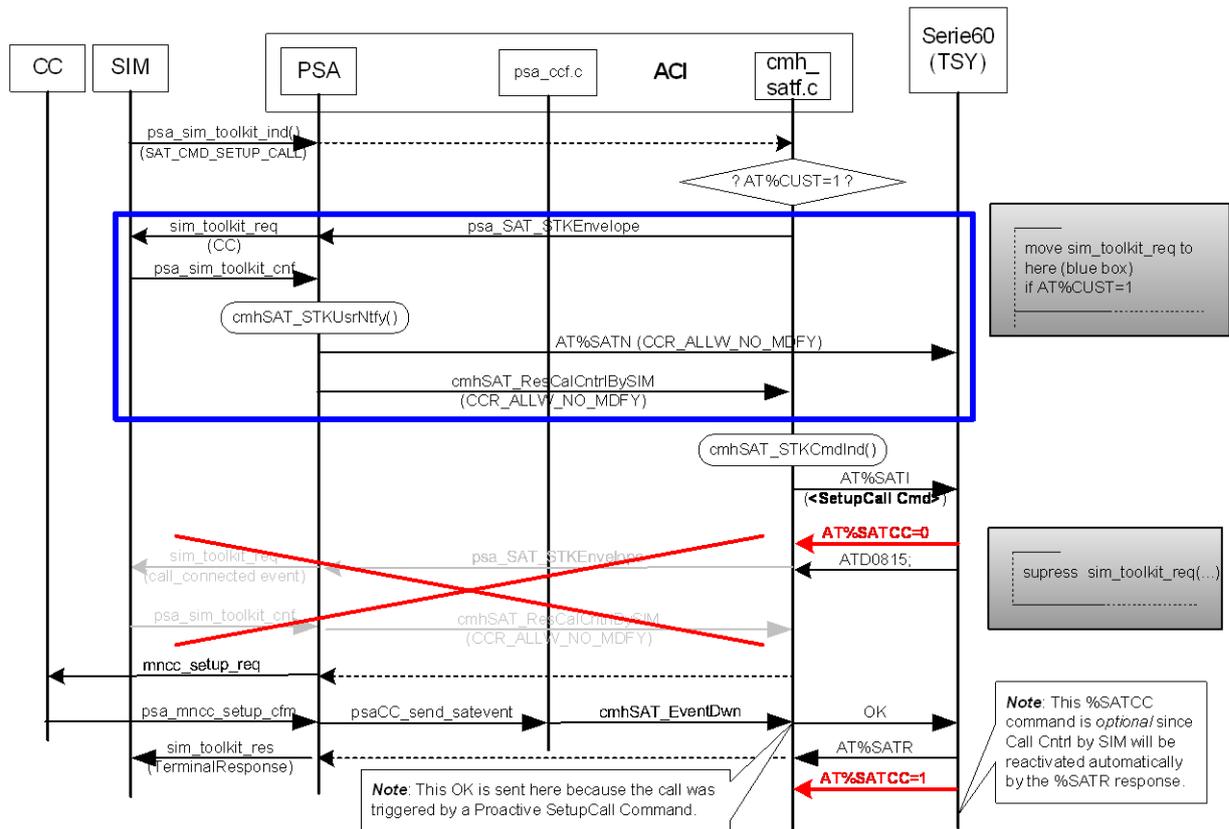


Figure 13 : SIM Initiated MO Call (No Modifications)

9.2.2 Allowed, with modifications (still a call)

This scenario is handled in exactly the same manner as a the SetUpcall with No Modifications by call control, except that the %SATN notification to the MMI is of a different type.

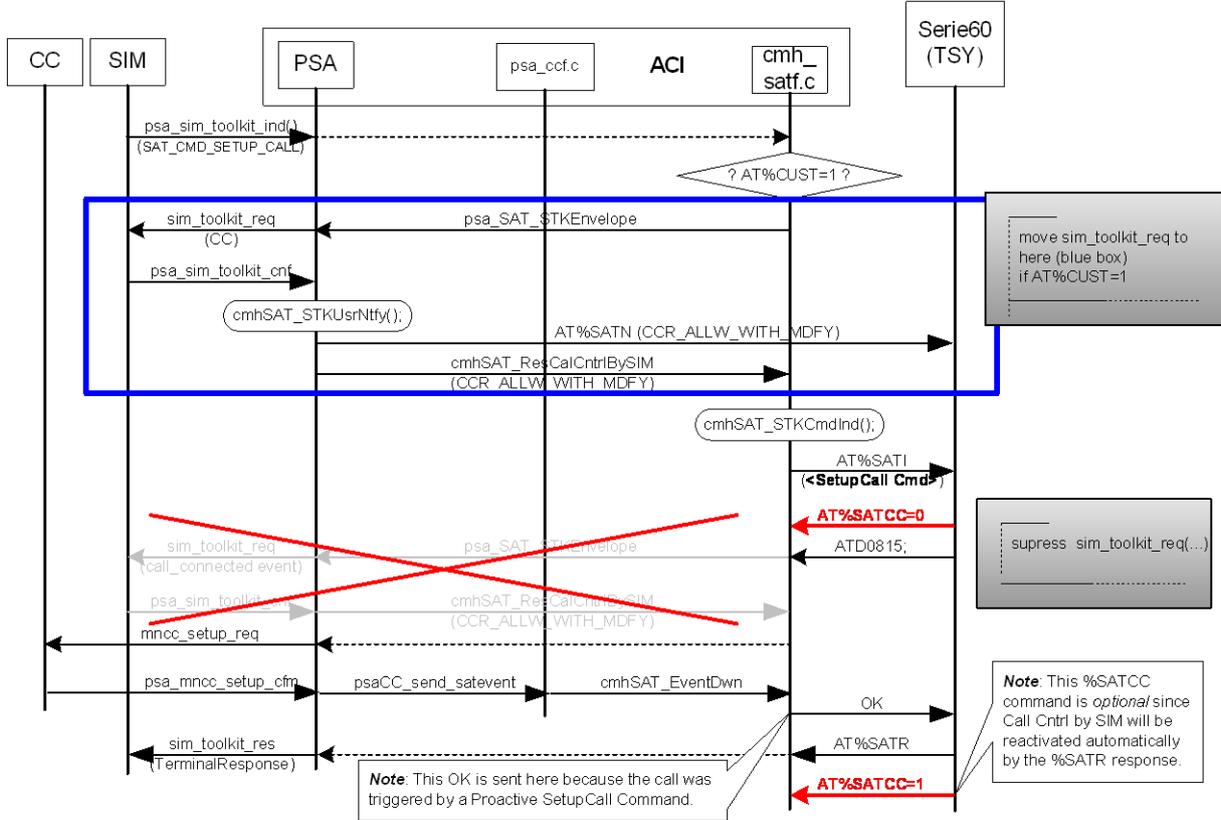


Figure 14 : SIM Initiated MO Call (Allowed with Modifications)

NOTE: It is important for SIXTIES to receive %SATN before %SATI !!

9.2.3 Allowed, Modified to Send SS/USSD

The ACI captures the SetupCall Command and, provided Call Control By SIM is active, immediately sends a Call Control by SIM request to the SIM. The resultant response is then forwarded through to the MMI in a %SATN notification. The cmhSAT_ResSSCntrlBySIM or cmhSAT_ResUSSDCntrlBySIM, normally handled by the ACI, is forwarded to the MMI in a %SATI (SendSS/SendUSSD) indication. (It is important for the SixTies to receive the %SATN notification **before** it receives the %SATI indication).

On receipt of the %SATI indication the SixTies will send a %SATCC=0 command in order to disable the Call Control by SIM mechanism, and will follow it with an ATD command. Once the SS / USSD transaction has completed and the Terminal Response sent to the SIM, the Call Control By SIM will be re-activated with the command %SATCC=1.

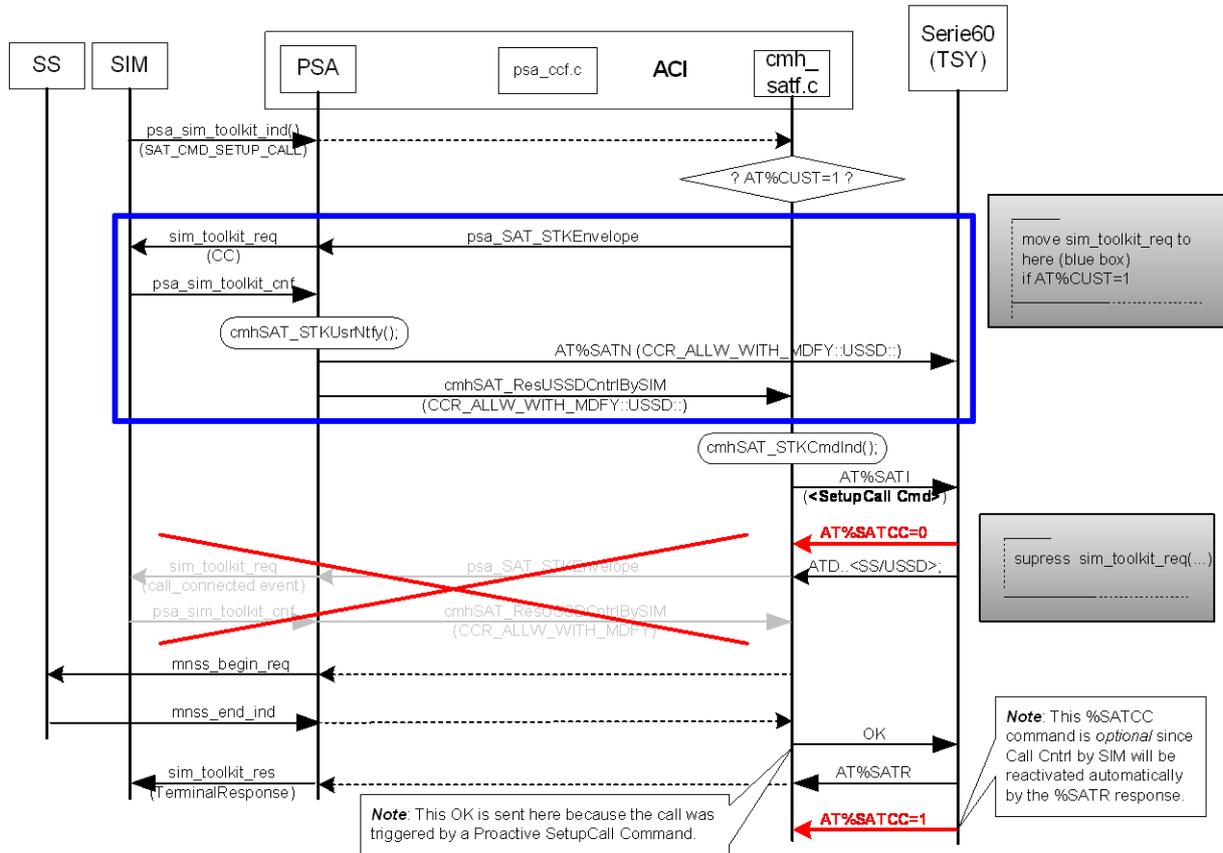


Figure 15 : SIM Initiated MO Call (Modified to SS or USSD Request)

9.2.4 Not Allowed

The ACI captures the SetUpCall Command and, provided Call Control By SIM is active, immediately sends a Call Control by SIM request to the SIM. The resultant response (NOT_ALLOWED) is NOT then forwarded through to the MMI, this is an exception from the other responses. The cmhSAT_ResCalCntrlBySIM (CCR_NOT_ALLW) is handled by the ACI and a Terminal Response sent to the SIM.

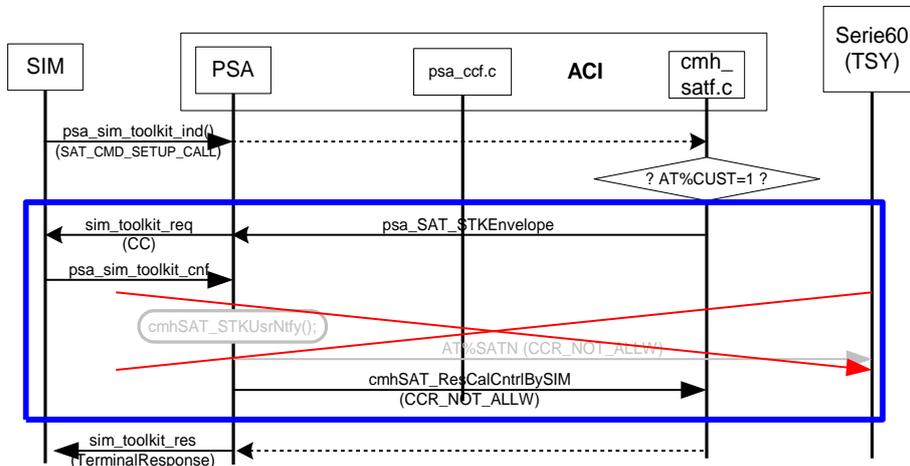


Figure 16 : SIM Initiated MO Call (Not Allowed)

10 Supplementary Service and USSD Handling

10.1 User Initiated SS or USSD Requests

User initiated Supplementary Service requests are performed using a ATD<SS/USSD String> request, and as such they are handled in exactly the same manner as a User Initiated Mobile Call (Para. 9.1) The message sequence charts shown in the reference paragraph are valid for SS and USSD requests also, however instead of an mncc_setup_req being sent from the Modem part, the primitive will be an mnss_begin_req (see Figure 11)

10.2 SIM SendSS or SendUSSD Requests

As for User Initiated SS or USSD Requests, the SIM SendSS and SendUSSD Requests are handled in exactly the same manner as the SIM Initiated MO Call (Para. 9.2)

The message sequence charts shown in the reference paragraph are valid for SendSS and SendUSSD requests also, however instead of an mncc_setup_req being sent from the Modem part, the primitive will be an mnss_begin_req (see Figure 15)

11 Mobile Originated Short Message Handling

11.1 User Initiated SMS Handling

The following sections show the current implementation for MO SMS handling with regard to MO Short Message Control by SIM. They are included for completeness, although no modifications are required relating to this area of functionality.

At each SAT SMC_ACT_MO related confirm (sim_toolkit_cnf) an AT%SATN is sent. It includes the content of the sim_toolkit_cnf primitive (allowed, allowed with modifications, not allowed).

11.1.1 Allowed with No Modifications

In the following Message Sequence Chart, that the psaSAT_STKEnvelope (marked in red) was triggered by an AT+CMGS request from the MMI, causes the %SATN notification (marked in red) to be sent back to the MMI, however, the subsequent cmhSAT_ResSMCntrlBySIM and the resulting Sms Submit (marked in blue) are handled entirely within the ACI.

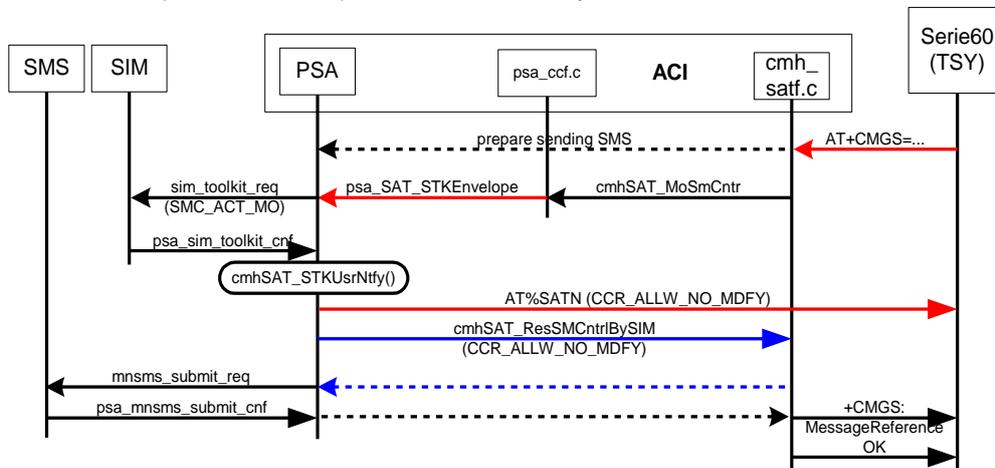


Figure 17 : User Initiated Short Message (No Modifications)

11.1.2 Allowed, with Modifications

This scenario is handled in exactly the same manner as the MO SMS with No Modifications by MO short message control, except that the %SATN notification to the MMI is of a different type.

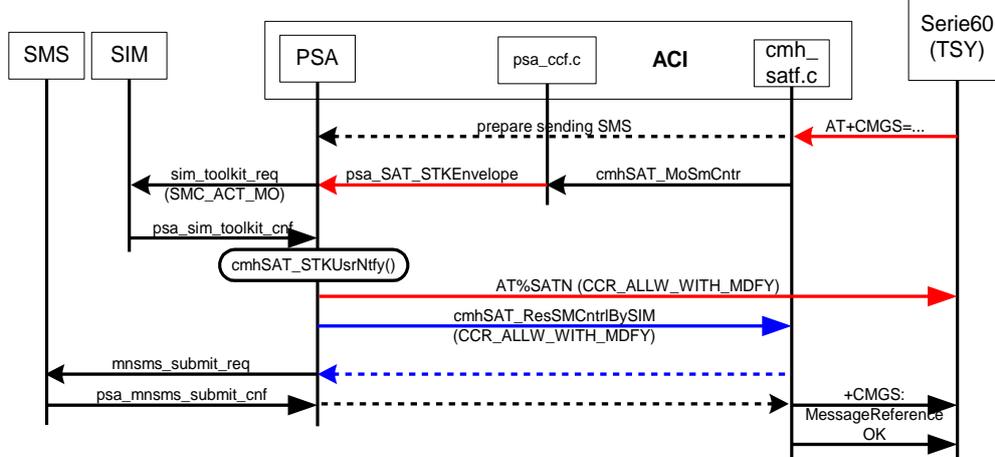


Figure 18 : User Initiated Short Message (Allowed with Mods)

11.1.3 Not Allowed

In the following Message Sequence Chart, that the `psaSAT_STKEvelope` (marked in red) was triggered by an `AT+CMGS` request from the MMI causes the %SATN notification (marked in red) to be sent back to the MMI. The subsequent `cmhSAT_ResSMCntrlBySIM` causes the ACI to send an ERROR indication to the MMI.

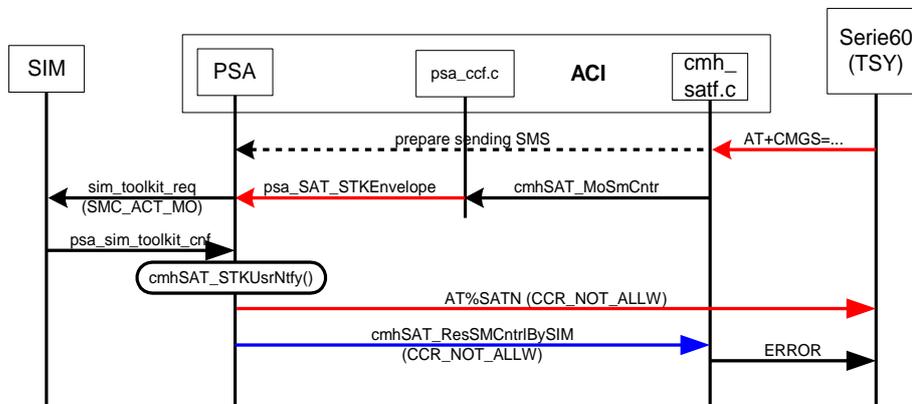


Figure 19 : User Initiated Short Message (Not Allowed)

11.2 SendSm Proactive SIM Command Handling

The expected behaviour, where the SendSm Command is handled by the MMI, would be that the SIM Toolkit PDU would be forwarded to the MMI in a %SATI indication, and an AT+CMGS command would be expected as a result. This AT+CMGS command would be subject to the MO Short Message Control By SIM procedure if it were active, with the MMI being notified of the result, but the necessary modifications being handled within the ACI as for a User Initiated SMS (See Para 10). The MMI would be responsible for the terminal response to the SIM indicating the outcome of the command.

For Atelier, the procedure is slightly different and is outlined below for each of the possible scenarios.

In the Message Sequence Charts below the following should be noted:

- All the %SATI commands described hold the PDU of the pending proactive command sent by the SIM.
- All the %SATN commands described hold the PDU of the result to the SMC envelope cmd sent to the SIM.

11.2.1 Allowed, with no modifications

The ACI captures the SendSm Command and, provided MO Short Message Control By SIM is active, immediately sends an MO Sm Control by SIM request to the SIM. The resultant response is then forwarded through to the MMI in a %SATN notification. The cmhSAT_ResSMCntrlBySIM, is handled by the ACI. The MMI is sent a %SATI (SendSm) indication. (It is important for the SixTies to receive the %SATN notification **before** it receives the %SATI indication).

On receipt of the %SATI indication the SixTies will send a %SATCC=0 command in order to disable the MO Short Message Control by SIM mechanism, and will follow it with an AT+CMGS command. Once the Sms has been sent the Terminal Response is forwarded to the SIM and the MO Short Message Control By SIM will be re-activated with the command %SATCC=1.

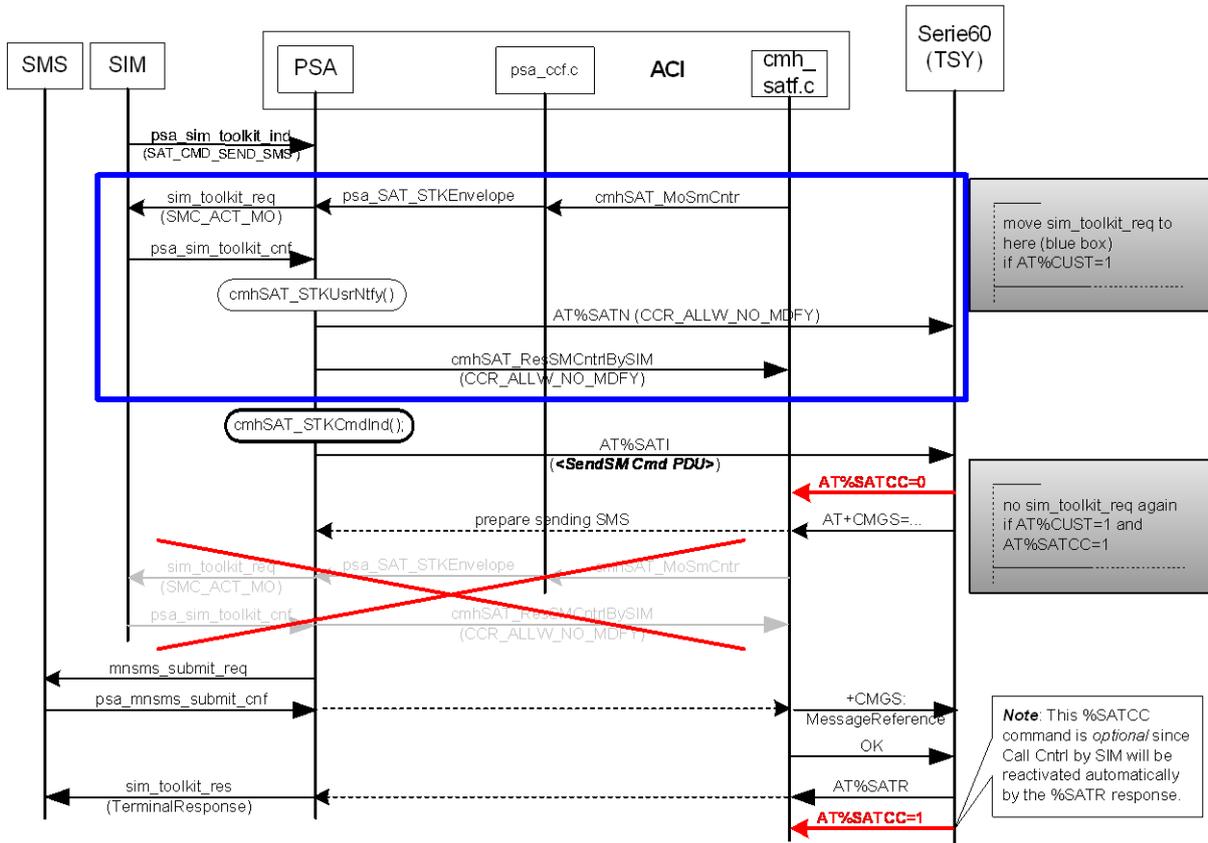


Figure 20 : SIM Initiated Short Message (No Modifications)

11.2.2 Allowed, with modifications

This scenario is handled in exactly the same manner as the MO SMS with No Modifications by MO short message control, except that the %SATN notification to the MMI is of a different type.

NOTE: It is important for SIXTIES to receive %SATN before %SATI !!

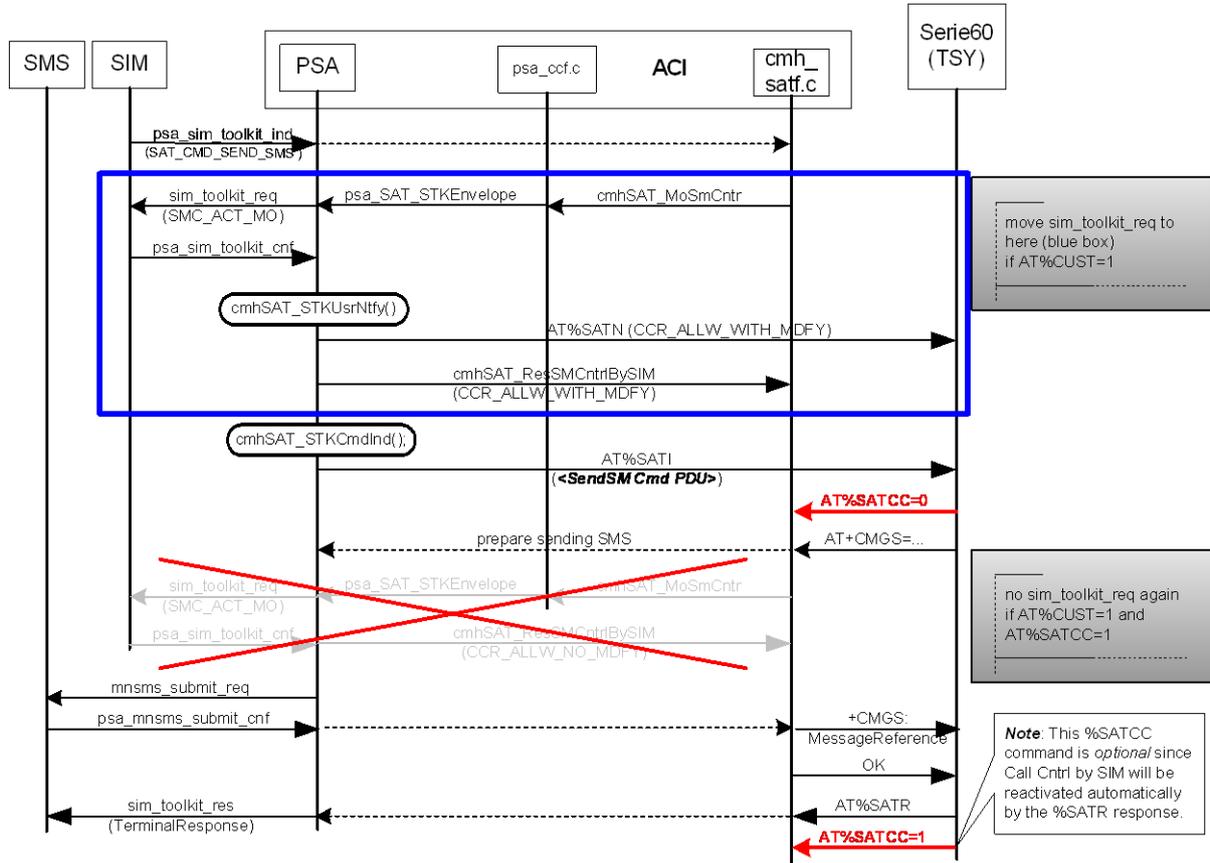


Figure 21 : SIM Initiated Short Message (Allowed with Mods)

11.2.3 Not Allowed

The ACI captures the SendSm Command and, provided MO Short Message Control By SIM is active, immediately sends an MO Sm Control by SIM request to the SIM. The resultant response is then forwarded through to the MMI in a %SATN notification. The cmhSAT_ResSMCntrlBySIM, is handled by the ACI, and the ACI sends the appropriate Terminal Response to the SIM.

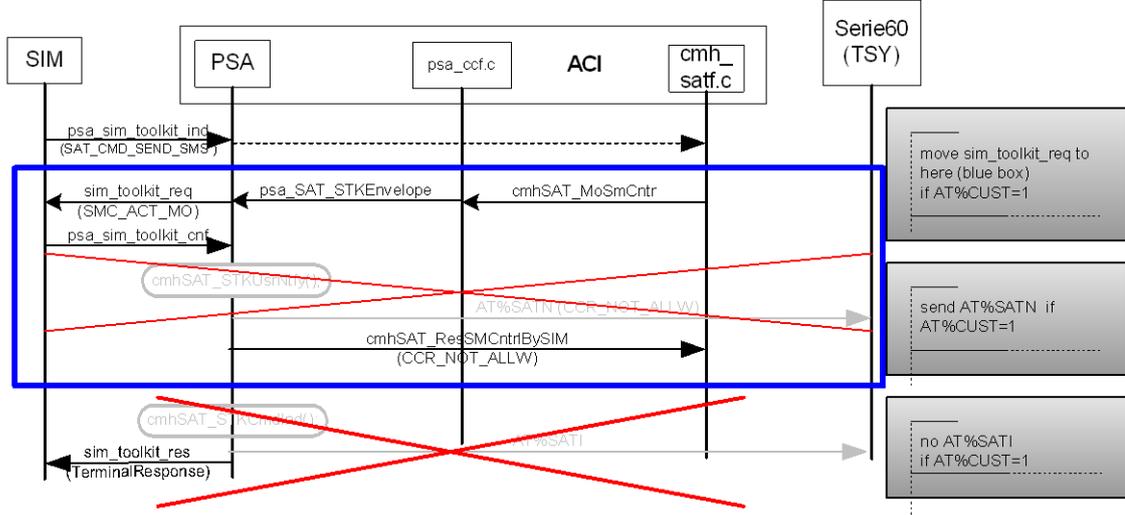


Figure 22 : SIM Initiated Short Message (Not Allowed)

A. Appendices

B. Acronyms

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

C. 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/>>