TEXAS INSTRUMENTS

Design Document

Dual SIM Feature Support

Department:	Berlin Wireless Center		
Creation Date:	2004-01-12		
Last Modified:	2005-01-02 by Karthik Ramamurthy		
ID:		Version:	001
Status:	Draft	ECCN:	Not applicable

0 Document Control

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 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 do 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, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of TI.

0.1 Export Control Statement

Recipient agrees that it will not knowingly export or re-export, directly or indirectly, any product or technical data (as defined by the U.S, EU and other Export Administration Regulations) including software, or any controlled product restricted by other applicable national regulations, received from Disclosing party under this Agreement, or any direct product of such technology, to any destination to which such export or re-export is restricted or prohibited by U.S or other applicable laws, without obtaining prior authorisation from U.S. Department of Commerce and other competent Government authorities to the extent required by those laws. This provision shall survive termination or expiration of this Agreement.

According to our best knowledge of the state and end-use of this product or technology, and in compliance with the export control regulations of dual-use goods in force in the origin and exporting countries, this technology is classified as given on the front page.

This product or technology may require export or re-export license for shipping it in compliance with certain countries regulations.

0.2 Document History

Date	Version	Status	Author
2004-01-12	001	Draft	Karthik Ramamurthy

0.3 References, Abbreviations, Terms

LCR	Low Cost Router
HPLMN	Home Public Land Mobile network
IMSI	International Mobile Subscriber Identity

Table of Contents

1 Introduction

This document describes the implementation of the support for the Dual SIM feature in ACI and SIM modules of the protocol stack.

This Dual SIM feature allows the user to have two SIMs on the Mobile Equipment the same time. At a given point of time, one SIM will be powered on. This feature is primarily aimed at the following two use cases.

a. Frequent traveler between two countries: It is assumed that a traveller has operator SIM cards of both visiting countries. Thus, the traveller can use the HPLMN services in both countries by changing the SIM cards. Instead of doing this manually this should be done automatically by the LCR feature using the Dual-SIM Feature.

b. Traveller between various countries: It is assumed that a traveller has one SIM card from an operator of his home country and another one for roaming. Thus the traveller can use the HPLMN services while he is at home country or he uses the other one elsewhere. Again instead of doing this manually this should be done automatically by the LCR feature using the Dual-SIM feature.

The Dual SIM feature support implemented in the protocol stack provides the interface to the LCR to select a particular SIM at power on and also to switch between the two SIMs dynamically (without switching off the mobile phone). The currently implemented operations related to the SIM will be performed on the currently powered on SIM.

In the next chapter, various use cases and the expected LCR behavior is discussed. In chapter 3, the design of the support implemented in the protocol stack is described. In chapter 4, the implementation in the ACI module to support the Dual SIM feature is discussed. In chapter 5, the implementation in the SIM module to support the Dual SIM feature is described. In chapter 6, message sequences for various use cases are described.

2 Dual SIM Support Design

The implementation for the support for the Dual SIM feature has been designed in such a way as to not affect any other functionality of the protocol stack.

A new TI proprietary AT command %SIM is implemented to enable the LCR to select a particular SIM. This new AT command will have set, query and test options.

The SIM primitive SIM_ACTIVATE_REQ will be expanded to accommodate the SIM select/switch operation. Two new optional elements, one indicating Sim select/switch operation and the other carrying the actual number of the selected SIM are added.

The support is available to the LCR application, for activation/deactivation of the selected SIM in the form of +CFUN AT command and in the form of +CRSM to enable reading of the ICC id and IMSI files on the selected SIM.

2.1 %SIM AT Command Syntax

Command	Possible response(s)
%SIM= <sim_num>	+CME ERROR: <err>
%SIM?	%SIM: <sim_num> +CME ERROR: <err>
%SIM =?	%SIM: (0-2) +CME ERROR: <err>

Description

The set option will allow the user to select a particular SIM to be powered on. In case, the other SIM is powered on and not activated, the set option of the %SIM command will power off the power on SIM and then powers on the selected SIM. However, the selection of the SIM with the set option of the %SIM command will fail, if there is a SIM active and registered to the network and an attempt is made to select another SIM.

The query option of the %SIM command enables the user to get information as to which SIM is currently powered on. If no SIM is powered on, the query option will return 0.

The test option will return the allowable options that can be used as the parameter with the set option of the %SIM command. Currently a standard “(0-2)” will be returned.

Defined Values

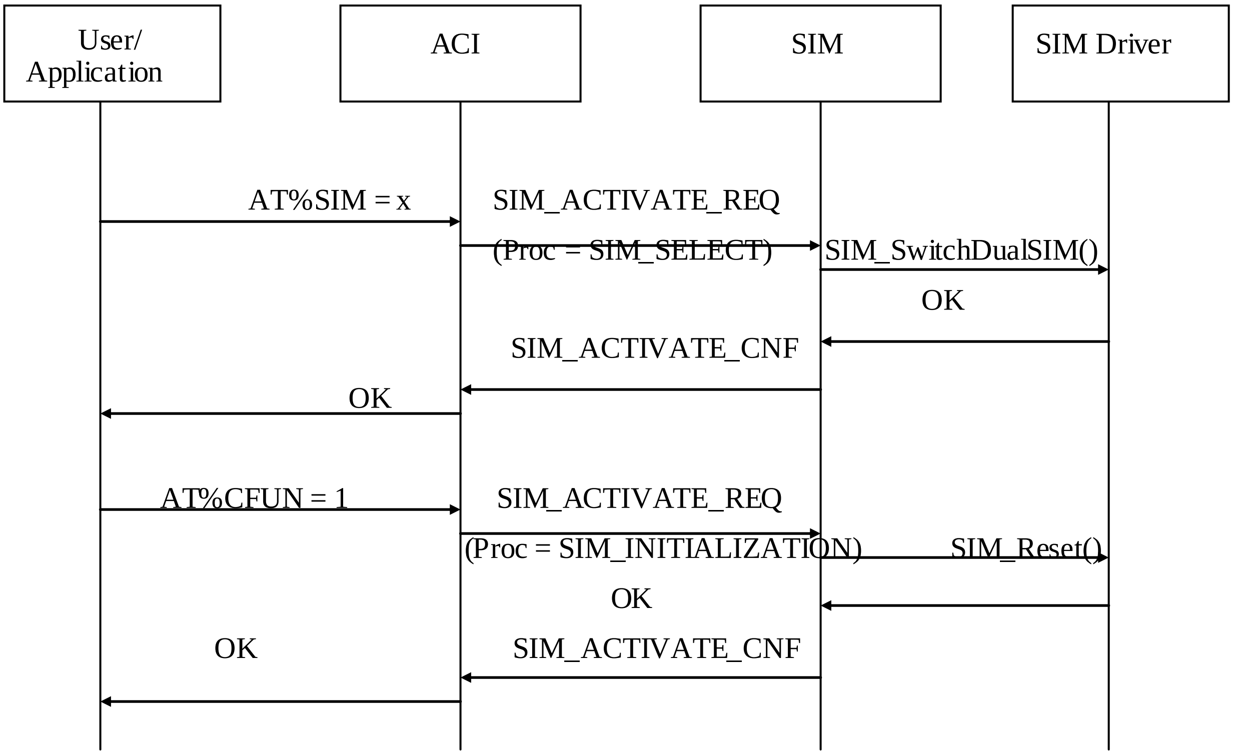
<sim_num> : Integer value. Can take up values 0, 1 or 2.

3 Use Case Sequences

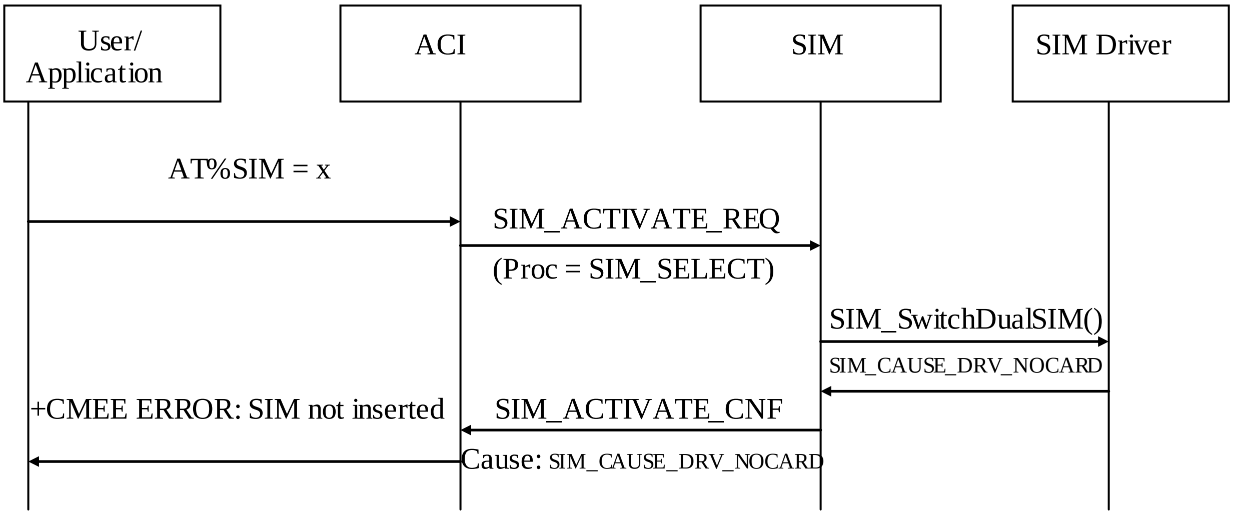
The various use cases are given below.

3.1 Select SIM

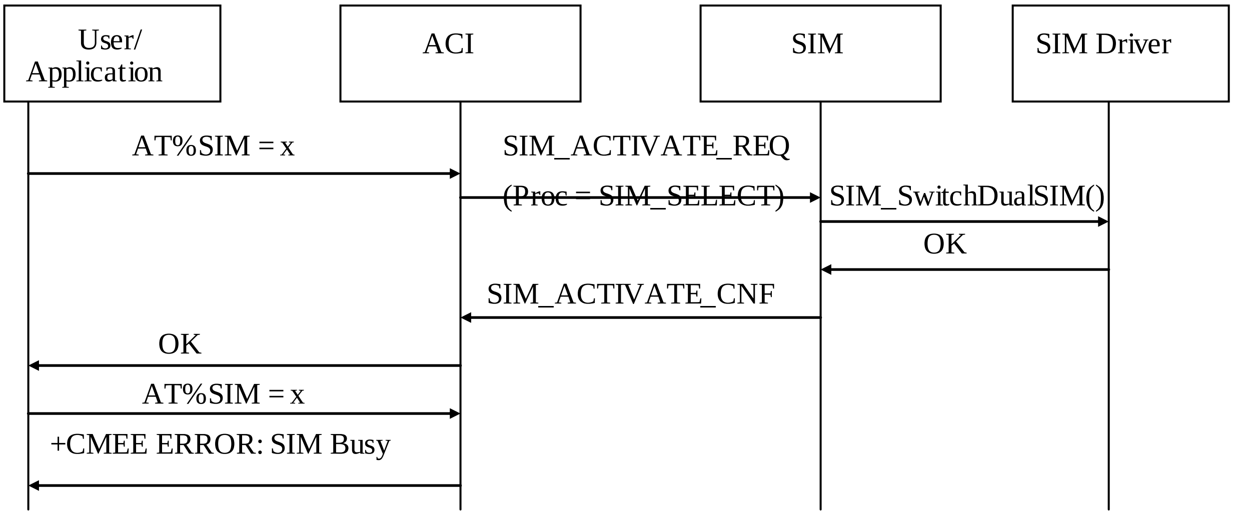
This use case occurs when there is no SIMs active. In this case, firstly a SIM selection is made by %SIM and then it is initialized/activated by using +CFUN



If a SIM is selected using %SIM command and the SIM is not physically present, the a "SIM not inserted error" is displayed to the user.

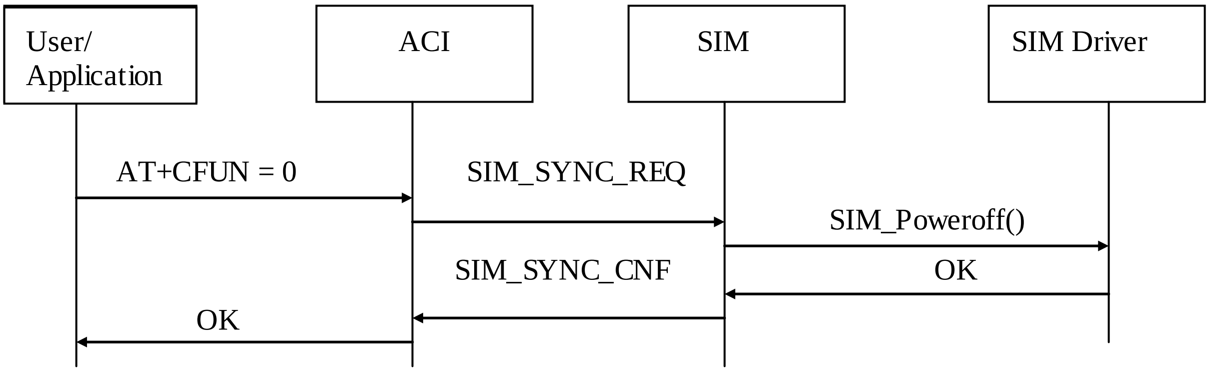


If a SIM has already been selected and if the same SIM is Selected again, a SIM Busy Error is displayed to the user



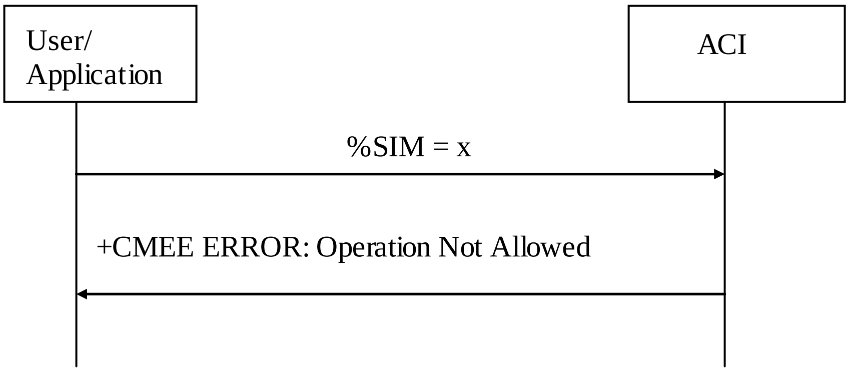
3.2 Switch SIM

It is assumed that currently a SIM is powered on and activated. In this case the currently active SIM is powered off with +CFUN command, another SIM is selected using %SIM and then it is activated using +CFUN.



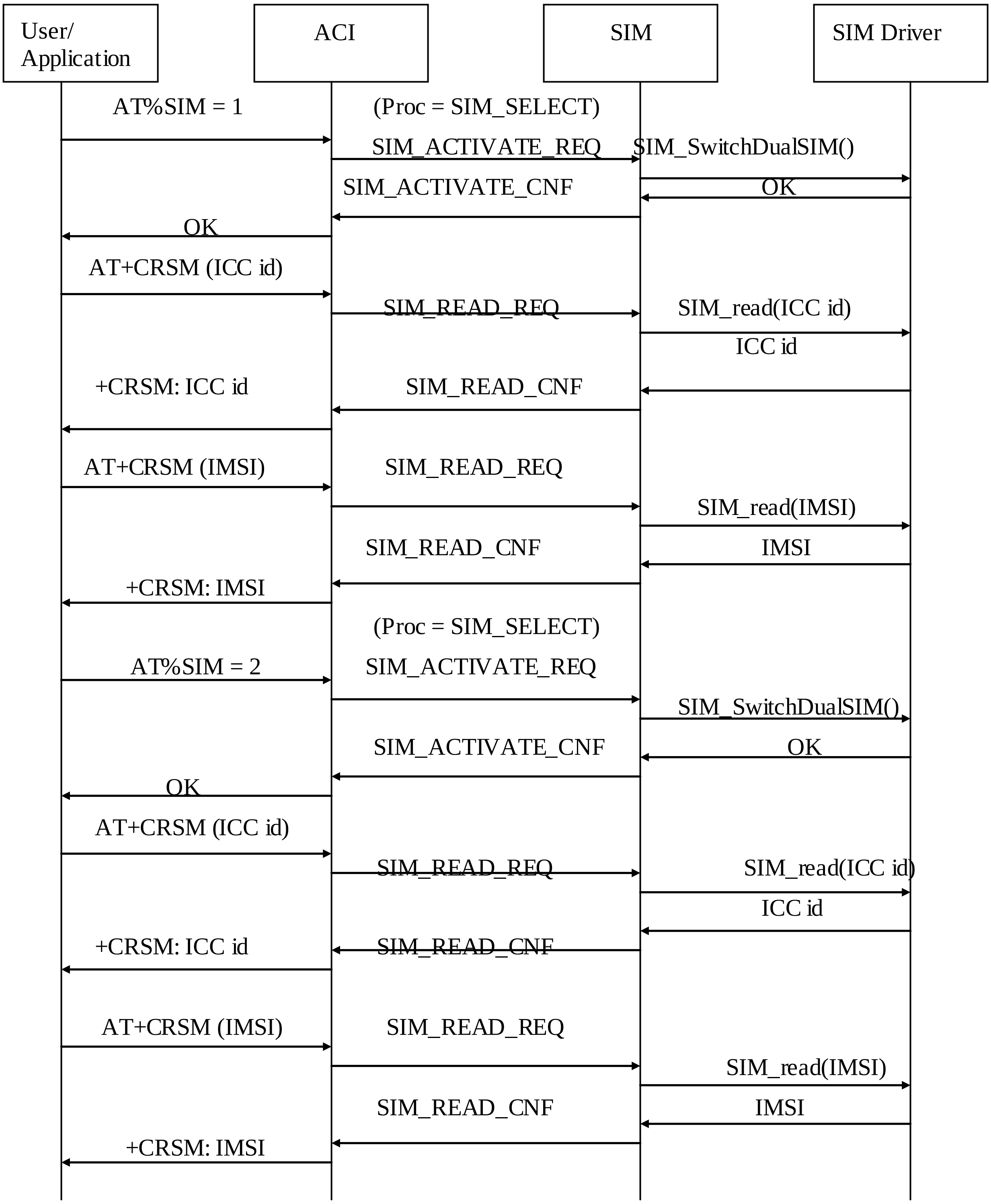
The rest of the procedure is same as in Select SIM use case.

If the %SIM is used to switch the SIM directly without first switching off the currently active SIM, an error is returned. The sequence is as shown below if currently a SIM is active.



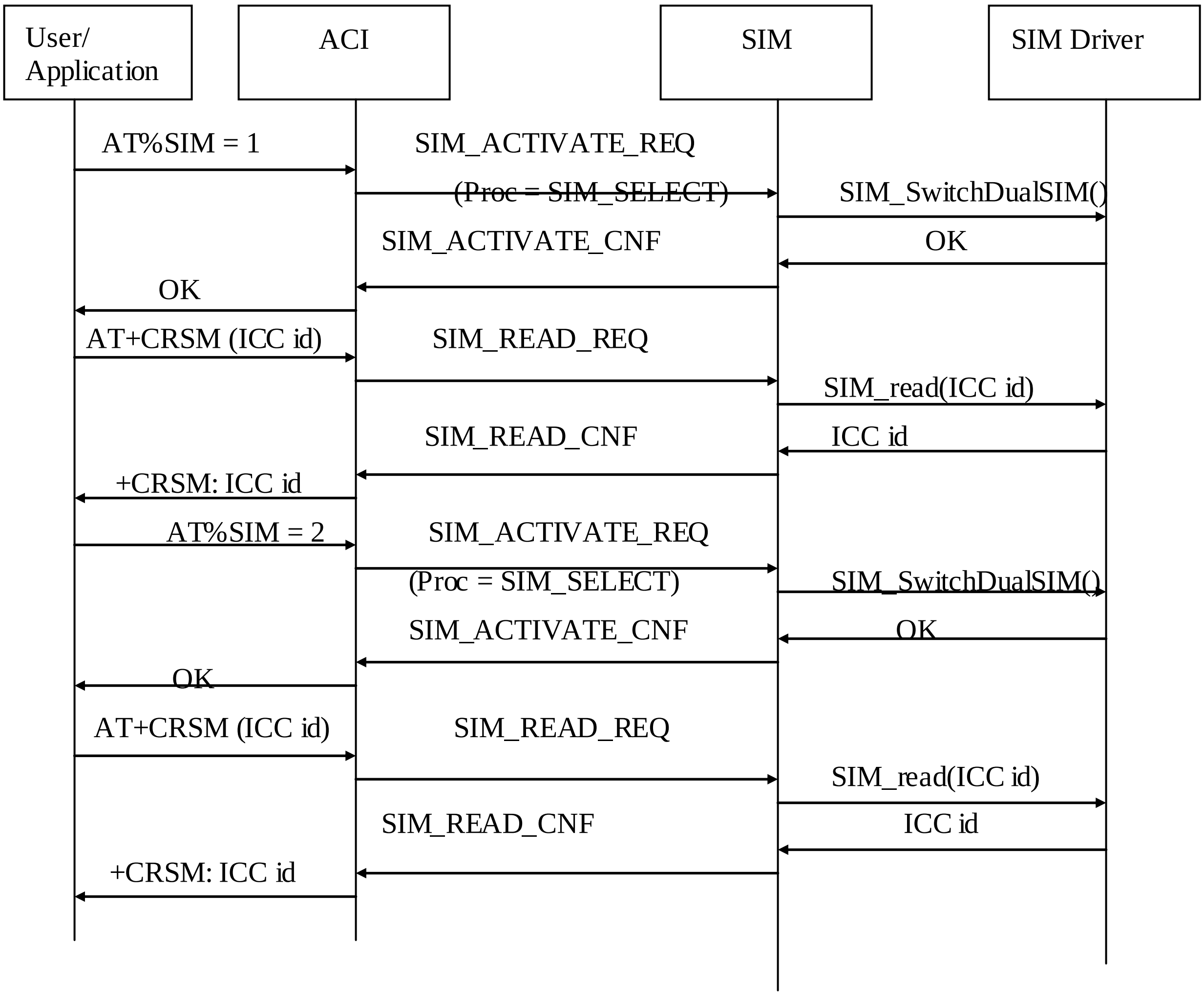
3.3 Introduce SIM

This Use case condition arises when the SIMs are new and their ICC id and IMSI are not known to the application. Neither of the SIMs are active. ICC id and IMSI of both the new SIMs are read and stored by the application for identifying the SIMs.



3.4 Get SIM Info

This use case scenario is faced when the SIMs are known to the application. In this use case, only ICC id is read and then compared with the one stored by the application before activating it.



4 Implementation in ACI Entity

This chapter describes the ACI implementation part of the Dual SIM support. The processing of the new AT command is implemented in ACI.

4.1 Variables

Three new variables are introduced in the structure T_SIM_SHRD_PRM. The first variable is SIM_Selected of type UBYTE which is introduced in the structure T_SIM_SET_PRM setPrm which in turn is a part of the structure T_SIM_SHRD_PRM. This variable is loaded with the value of the parameter in the set option of the %SIM command. The second variable is SIM_Powered_on of type UBYTE which has been added as the new element of T_SIM_SHRD_PRM structure. This element holds the currently powered on SIM number and is used to respond to the user's query option of the %SIM command. The last one is SIM_Selection which is a flag set during a SIM selection.

4.2 Functions

New functions have been added to implement the new AT command. A few existing functions are also modified.

4.2.1 New Functions

The following new functions have been added.

4.2.1.1 setatPercentSIM ()

Prototype: GLOBAL T_ATI_RSLT setatPercentSIM (char *cl, UBYTE srcId);

Parameters:

Name	Comment	
cl	Command line	IN
srcId	Originator of command	IN

Returns:

Symbolic constant	Comment
ATI_EXCT	Executing
ATI_FAIL	Failed

Description:

This function recovers the SIM number in the set option of the %SIM command. It calls the function sAT_PercentSIM() with the SIM number as its argument.

4.2.1.2 queatPercentSIM()

Prototype: GLOBAL T_ATI_RSLT queatPercentSIM (char *cl, UBYTE srcId);

Parameters:

Name	Comment	
cl	Command line	IN
srcId	Originator of command	IN

Returns:

Symbolic constant	Comment
ATI_CMPL	Successfully completed
ATI_FAIL	Failed

Description:

This function obtains the currently powered on SIM number by calling the function qAT_PercentSIM() for the AT interpreter interface.

4.2.1.3 sAT_PercentSIM()

Prototype: GLOBAL T_ACI_RETURN sAT_PercentSIM(T_ACI_CMD_SRC srcId, UBYTE sim_num);

Parameters:

Name	Comment	
srcId	Originator of command	IN
sim_num	Index of the SIM to be selected	IN

Returns:

Symbolic constant	Comment
AT_EXCT	Executing
AT_FAIL	Failed

Description:

This function checks the SIM activation status and also the validity of the selected SIM number. If either SI is in full functionality or if the SIM requested to be selected is invalid, an "operation not allowed" error is returned to the user. Otherwise, the SIM_Selected variable of simShrdPrm.setPrm is updated with the selection and the function psaSIM_SelectSIM() is called for further processing.

4.2.1.4 psaSIM_SelectSIM()

Prototype: GLOBAL SHORT psaSIM_SelectSIM (void);

Returns:

Value	Comment
-1	Error
0	Successful

Description:

This function updates the primitive elements and sends the primitive to the SIM module for a SIM selection.

4.2.1.5 qAT_PercentSIM()

Prototype: GLOBAL T_ACI_RETURN qAT_PercentSIM (T_ACI_CMD_SRC srcId, UBYTE *sim_num);

Parameters:

Name	Comment	
srcId	Originator of command	IN
sim_num	Index of the SIM selected	IN/OUT

Returns:

Symbolic constant	Comment
AT_CMPL	Successfully Completed
AT_FAIL	Failed

Description:

This function returns the currently powered on SIM number stored in simShrdPrm.SIM_Powered_on.

4.2.1.6 cmhSIM_SIMSelected()

Prototype: GLOBAL void cmhSIM_SIMSelected (void);

Description:

This function is called by the primitive handling function psa_sim_activate_cnf() when a confirm primitive is received from the SIM module for a SIM select request. This function sends the response of the select request to the user.

4.2.2 Modified functions

The following functions have been modified to support the Dual Sim feature.

4.2.2.1 AtiHandleTestCOPS()

This function has been modified to allow a network scan request without checking for the SIM activation status.

4.2.2.2 psaG_MM_CMD_NET_SRCH()

This function has been modified to send a GMMREG_NET_REQ to the GMM module instead of GMMREG_ATTACH_REQ, if the SIM is not in active state.

4.2.2.3 psaMM_NetSrch()

This function contains the same modification as above but for a stack built without GPRS.

4.2.2.4 psa_sim_activate_cnf()

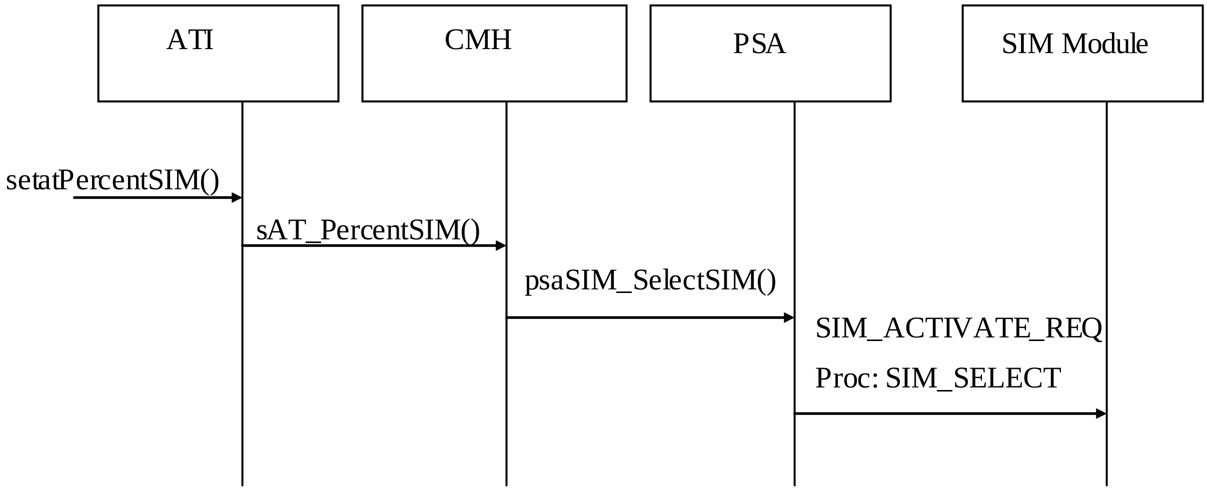
This function has been modified to handle the response from the SIM module for a SIM select request using the primitive SIM_ACTIVATE_REQ.

4.3 Internal Control Flow Behavior

The control flow for the %SIM AT command is as given below.

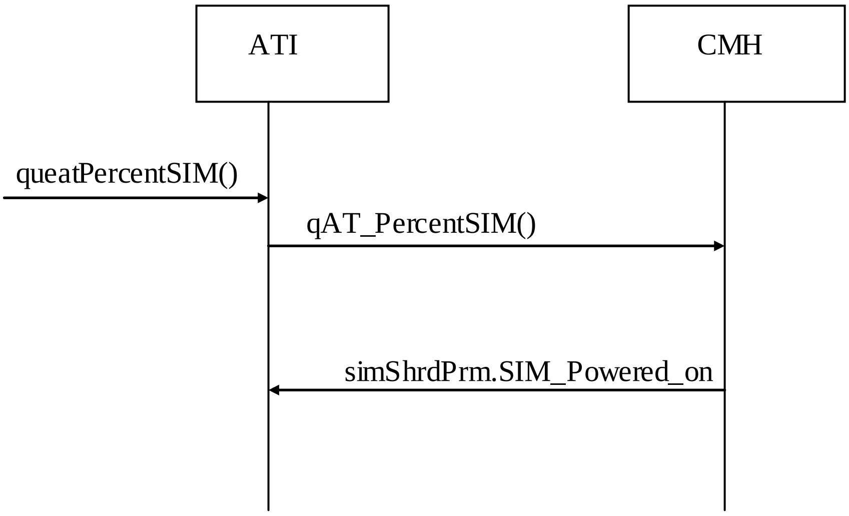
4.3.1 %SIM Set Command

After enter the %SIM command at AT-Interface, the parameter will be stored into the shared parameter inside the command handler and a primitive will send to the SIM-entity to power on the Selected SIM.



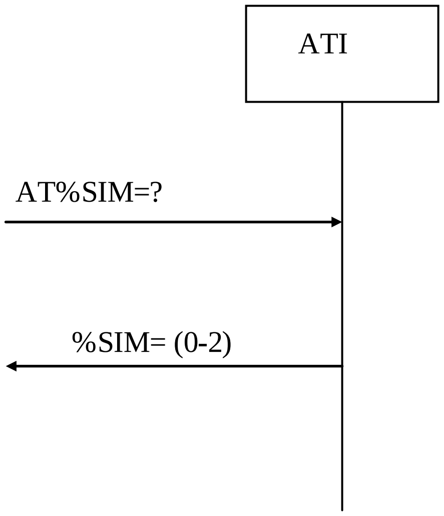
4.3.2 %SIM Query Command

After enter the %SIM? Command at AT-Interface, the parameter stored into the shared parameter inside the command handler will be returned.



4.3.3 %SIM Test Command

After enter the %SIM=? Command at AT-Interface, a standard response as given in 2.1 will be returned.



5 Implementation in SIM Entity

This chapter describes the SIM module part of the Dual SIM support implementation in the protocol stack. A few new variables and a few changes in the functions have been incorporated in the SIM module. They are as described below.

5.1 Variables

Two new variables have been introduced in the structure T_SIM_DATA sim_data. The variable “SIM_Selection” is a flag, which is set during a SIM selection process. The other variable is the “sim_num” which contains the number of the SIM powered on.

5.2 Functions

There have been changes in a few functions to include the processing of the SIM select request from the ACI module.

5.2.1 app_sim_activate_req()

The handling of the SIM select request in the SIM_ACTIVATE_REQ is done here. A new procedure SIM_SELECT is introduced here. In this case, SIM_SwitchDualSIM (), which is a driver API for SIM selection is called and if the SIM selection is successful, the SIM is powered on by calling the SIM_Reset () function and the SIM selected is reported back to ACI or else an error is reported to the ACI module.

5.2.2 app_sim_card_error()

This function has been modified to send the error to the ACI module in case of a SIM selection failure. The error sent to the ACI module will be the one received from the SIM driver.

5.2.3 app_sim_insert()

This function is modified to send a response to the ACI about the SIM selected in case of a successful SIM selection. In a SIM selection operation, this function will not send the insert indications to other modules.

5.3 Driver APIs

There are two driver APIs to support the Dual SIM handling.

5.3.1 SIM_SwitchDualSIM()

Prototype: UINT16 SIM_SwitchDualSIM(UINT16 sim);

Parameters:

Name	Comment	
sim	Index of the SIM selected	IN

Returns:

Type	Comment
UINT16	Error value

Description:

This driver API does the operation of selecting the SIM. It takes the value of the SIM selected as argument and returns the outcome of the operation (success or error).

5.3.2 SIM_GetSIM()

Prototype: UINT16 SIM_GetSIM();

Returns:

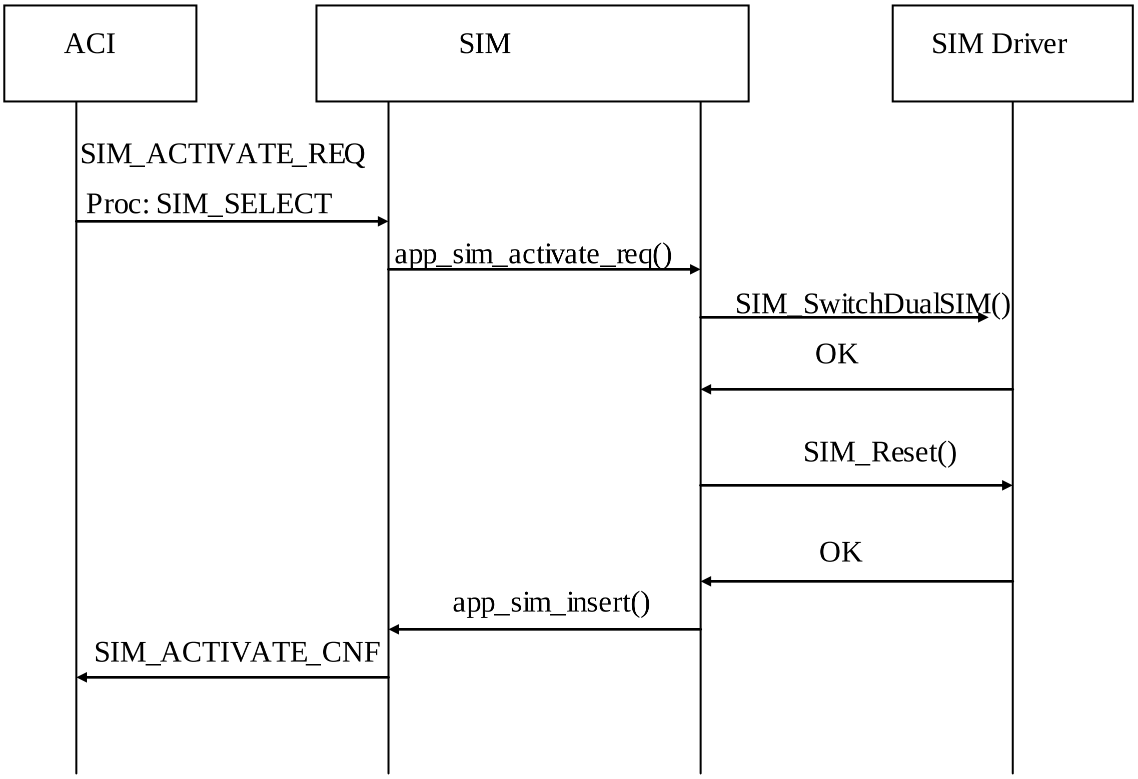
Value	Comment
0	No SIM selected
1	SIM 1 selected
2	SIM 2 selected

Description:

This driver API returns the currently powered on SIM value.

5.4 Internal Flow Control Behavior

This describes the internal flow control behavior of the SIM module in response to a %SIM set command.



6 State/Action Sequences

The various states of the SIM and the various triggers/actions are given below. The AT command issued by the LCR application acts as the trigger and the State Machine moves to another State.

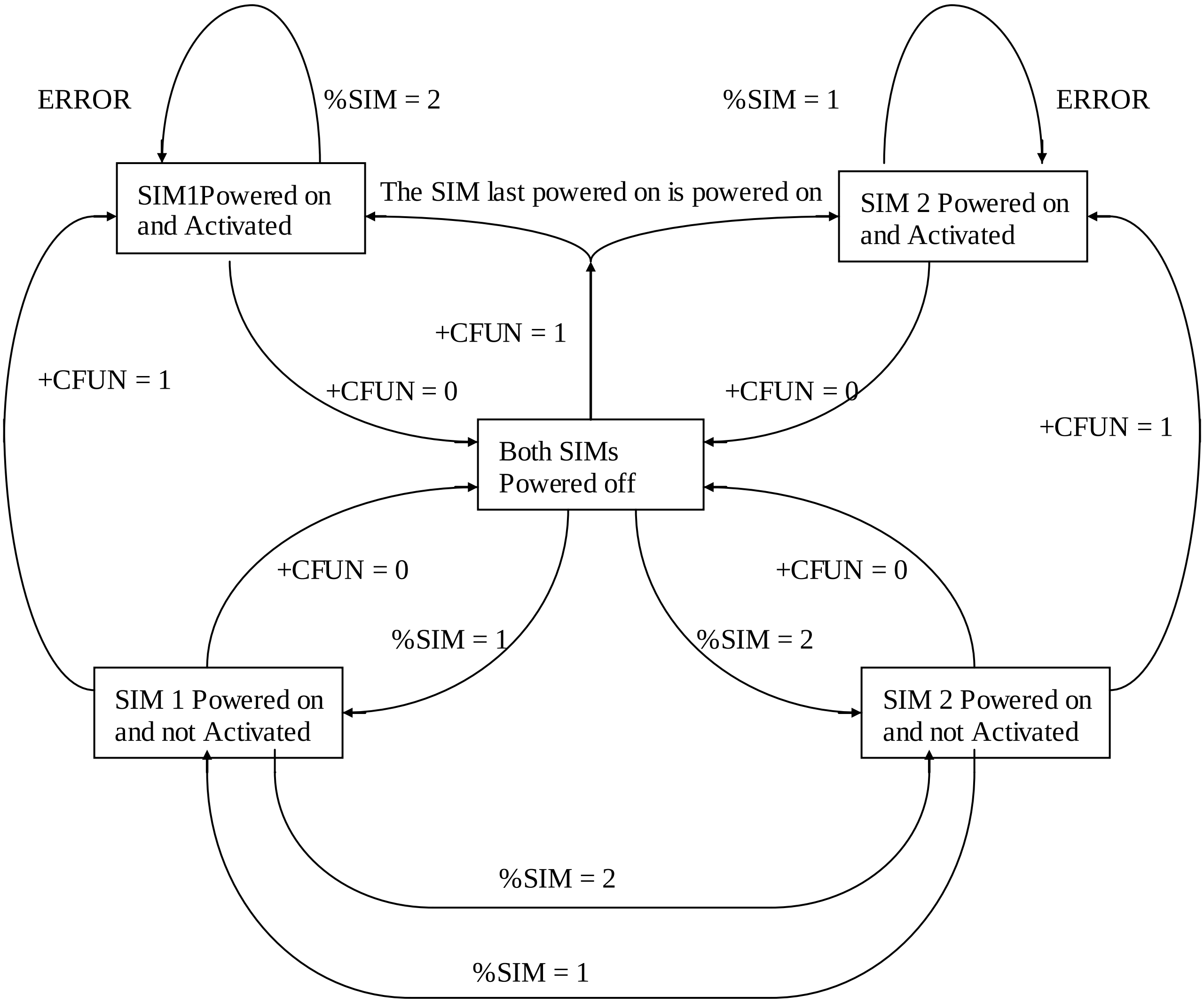
6.1 State Transition Table

The various triggers, action taken and the new states are given in the table below.

Current State	AT Command	Action	New State
Both SIMs Powered off	%SIM = 1	SIM 1 powered on	SIM 1 Powered on and not Activated
	%SIM = 2	SIM 2 powered on	SIM 2 Powered on and not Activated
	+CFUN = 1	Last powered on SIM is powered on and Initialized/Activated	SIM 1/2 Powered on and Activated
SIM 1 Powered on and not Activated	+CFUN = 1	SIM 1 is Initialized/Activated	SIM 1 Powered on and Activated
	%SIM = 2	SIM 1 powered off and SIM 2 powered on	SIM 2 Powered on and not Activated
SIM 2 Powered on and not Activated	+CFUN = 1	SIM 2 is Initialized/Activated	SIM 2 Powered on and Activated
	%SIM = 1	SIM 2 powered off and SIM 1 powered on	SIM 1 Powered on and not Activated
SIM 1 Powered on and Activated	+CFUN = 0	SIM 1 will be powered off	Both SIMs Powered off
	%SIM = 2	Error Operation Not Allowed is returned	No change in State
SIM 2 Powered on and Activated	+CFUN = 0	SIM 2 will be powered off	Both SIMs Powered off
	%SIM = 1	Error Operation Not Allowed is returned	No change in State

6.2 State Diagram

The above state transition table is depicted in the state diagram as shown below.


ERROR
%SIM = 2
%SIM = 1
ERROR
SIM1Powered on and Activated
The SIM last powered on is powered on
SIM 2 Powered on and Activated
+CFUN = 1
+CFUN = 1
+CFUN = 0
+CFUN = 0
+CFUN = 1
Both SIMs Powered off
+CFUN = 0
+CFUN = 0
%SIM = 1
%SIM = 2
SIM 1 Powered on and not Activated
SIM 2 Powered on and not Activated
%SIM = 2
%SIM = 1