



Low Level Design Specification

ACI/LLD Interface % CSSD - Supplementary Service Diagnostic

Department:	WTBU - Cellular Systems		
Creation Date:	2004-10-01		
Last Modified:	2004-10-01 by Marc Droste-Franke		
ID:	8462.741.04	Version:	001
Status:	Draft	ECCN:	Not Applicable

© 2004 Texas Instruments Incorporated. All rights reserved.

Texas Instruments Proprietary Information

Strictly Private

0 Document Control

© 2004 Texas Instruments Incorporated. All rights reserved.

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
2004-10-01	001	Draft	Marc Droste-Franke
Initial version.			

0.3 References, Abbreviations, Terms

[1] 3GPP TS 24.008 V3.19.0 (2004-06)

[TI 7010.801] 7010.801, References and Vocabulary, Texas Instruments.

Table of Contents

1	Overview.....	6
1.1	Introduction.....	6
1.2	Detailed description	6
1.3	Command syntax: Supplementary Service Diagnostic %CSSD	6
1.4	Internal behavior (MSC's).....	7
1.5	Calling %CSSD from Terminal outside	7
1.6	Storing of SS diagnostic during call disconnection	8
2	Modifications in ACI.....	9
2.1	Interface Changes	9
2.2	Modifications in ATI	9
2.2.1	Module ati_cmd.c	9
2.2.2	Module ati_err.c	9
2.3	Modification in CMH.....	9
2.3.1	Module aci_cmh.h	9
2.3.2	Module cmh_ccq.c.....	9
2.4	Modification in PSA	10
2.4.1	Module psa_cc.h.....	10
2.4.2	Module psa_ccf.c.....	10
2.4.3	Module psa_ccp.c	10
3	Test.....	11
3.1	Simulation Test.....	11
3.2	Target Test.....	11

1 Overview

1.1 Introduction

This document is a Low Level Design (LLD) document, which describes the implementation of the supplementary service diagnostic feature in ACI.

The cause information element may also contain diagnostic bytes¹. These bytes allow to substantiate the meaning of the given cause value. Diagnostic bytes are not supported for every specified cause value. Additionally, the specification for the diagnostic byte differs for the causes that can be enhanced with this kind of additional information.

3 GPP TS 24.008 section 10.5.4.11 Note 1 describes the handling of diagnostic bytes for supplementary service. These supplementary service diagnostic bytes are can be part of cause information elements with the following cause values:

- User busy
- Facility rejected
- No circuit/channel available
- Requested facility not subscribed
- Incoming calls barred within the CUG
- Requested facility not implemented
- User not member of CUG

1.2 Detailed description

The introduced command %CSSD provides the SS diagnostic to the user if this has been delivered within the last call disconnect process. The SS diagnostic is always defined in the first byte of the diagnostic bytes array. For this command, additional bytes given in the diagnostic bytes array are ignored.

The SS diagnostic value given during the last call disconnection is stored. The user has to call the command %CSSD actively to retrieve the information from ACI.

1.3 Command syntax: Supplementary Service Diagnostic %CSSD

Command	Possible response(s)
%CSSD	%CSSD: <ss_diagnostic> ERROR

Description

The aim of this command is to provide the information of the supplementary service byte sent by the network to the user <ss_diagnostic>. This byte provides a deeper analysis of the cause of the disconnection of the last call.

¹ Refer to Figure 10.5.95/3GPP TS 24.008.

This byte is only set in for special disconnect causes. Please refer to 3GPP TS 24.008 V3.19.0 section 10.5.4.11 (2004-06) specification for further details.

In the cases that there has no diagnostic value been delivered by the network the ss diagnostic value will be set to default 255 (0xFF).

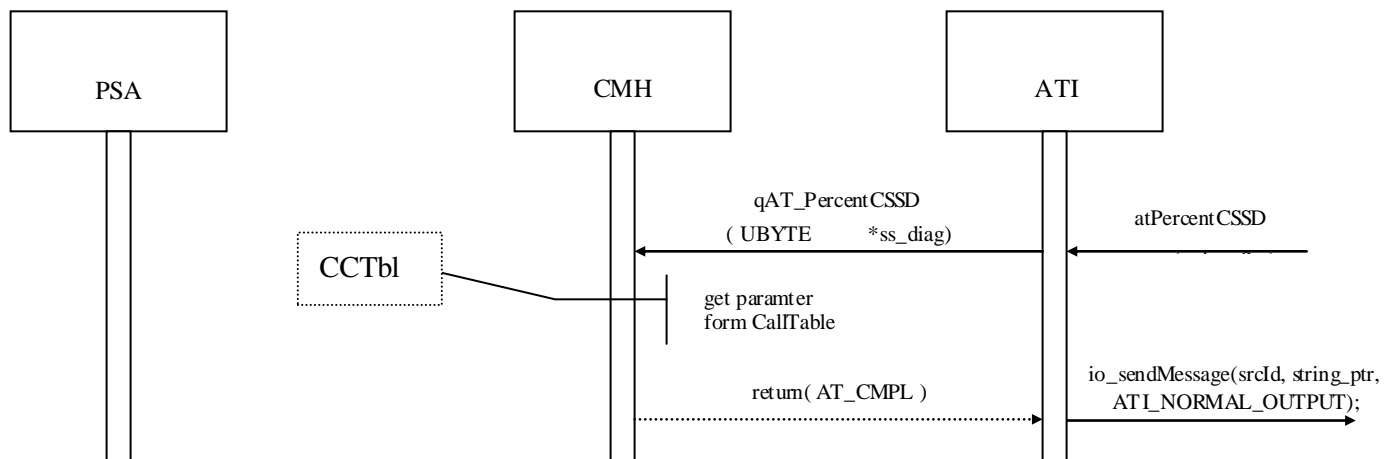
Defined values

<ss_diagnostic> :

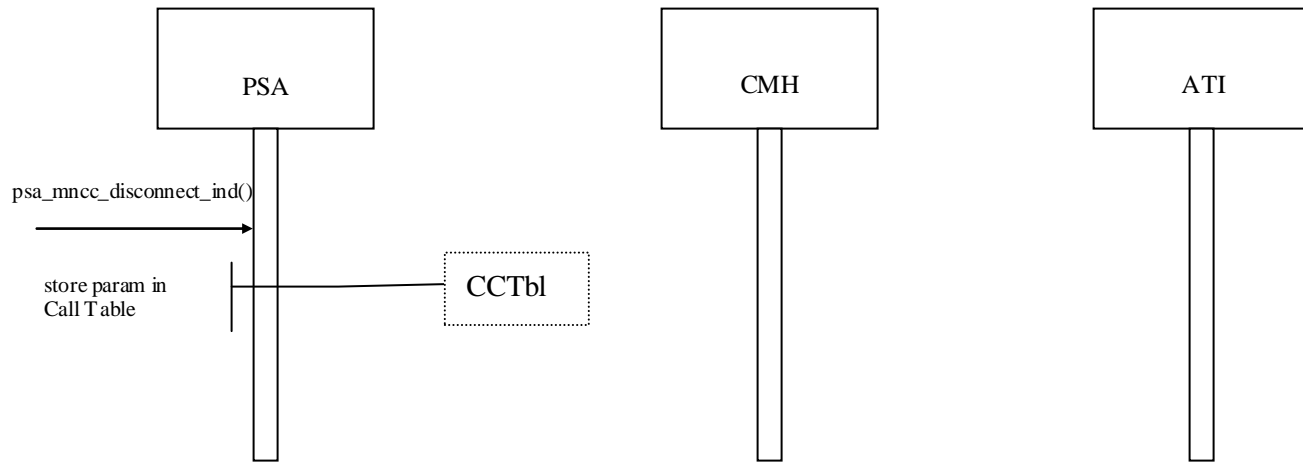
- 1 Outgoing call bared within CUG
- 2 No CUG selected
- 3 Unknown CUG index
- 4 CUG index incompatible with requested basic service
- 5 CUG call failure, unspecified
- 6 CLIR not subscribed
- 7 CCBS possible
- 8 CCBS not possible
- 255 No information provided

1.4 Internal behavior (MSC's)

1.5 Calling %CSSD from Terminal outside



1.6 Storing of SS diagnostic during call disconnection



2 Modifications in ACI

2.1 Interface Changes

The new command %CSSD is defined as described above.

2.2 Modifications in ATI

2.2.1 Module ati_cmd.c

- Additional prototypes
 - **EXTERN T_ATI_RSLT atPercentCSSD (CHAR *cl, UBYTE srcId);**
- Additional row in

```
LOCAL const ATCommand cmds [] = {
```

```
:
```

```
{"%CSSD",    AT_CMD_CSSD, atPercentCSSD, test_gen,    0,        0},
```

```
:
```

```
}
```

2.2.2 Module ati_err.c

- Implemented new function
 - **GLOBAL T_ATI_RSLT atPercentCSSD(char *cl, UBYTE srcId){**
:
qAT_PercentCSSD(srcId, &ss_diag);
:
}

2.3 Modification in CMH

2.3.1 Module aci_cmh.h

- New command id added
 - **typedef enum**
{
:
AT_CMD_CSSD = 222, /* %CSSD command id */
AT_CMD_MAX /* maximum command id */
}T_ACI_AT_CMD;
- New prototypes for command handler control functions
 - **EXTERN T_ACI_RETURN qAT_PercentCSSD (T_ACI_CMD_SRC srcId**
UBYTE *ss_diag);

2.3.2 Module cmh_ccq.c

- Implementation of functional counterpart for AT command

- GLOBAL T_ACI_RETURN qAT_PercentCSSD (T_ACI_CMD_SRC srcId
UBYTE *ss_diag)

2.4 Modification in PSA

2.4.1 Module psa_cc.h

- Added a new parameter in call table structure
 - typedef struct CCCallTabl
{
:
UBYTE ssDiag; /* SS diagnostic (CQ 23619 - % DIAG) */
:
}

2.4.2 Module psa_ccf.c

- Added line into function for proper initialization of SS diagnostic byte
 - GLOBAL void psaCC_InitCtbNtry (SHORT idx)
{
:
ccShrdPrm.ctb[idx].ssDiag = SS_DIAG_NOT_PROVIDED;
:
}

2.4.3 Module psa_ccp.c

- Copied SS diagnostic delivered by MNCC_DISCONNECT_IND to respective field in call table
 - GLOBAL const void psa_mncc_disconnect_ind
(T_MNCC_DISCONNECT_IND *mncc_disconnect_ind)
{
:
pCtbNtry->ssDiag = mncc_disconnect_ind->ss_diag;
:
}

3 Test

3.1 Simulation Test

The test cases ACI350A-I have been written to test %CSSD.

3.2 Target Test

Functional tests of the command have been done.

A Modifications in CC

A.1 SAP MNCC

- Added new parameter SS diagnostic parameter to MNCC_DISCONNECT_IND
 - **U8 ss_diag;**
- Added respective values
 - **VAL_ss_diag**

A.2 CC Sources

A.2.1 Module cc.h

- Added prototype for function that extracts the SS diagnostic value from primitive
 - **EXTERN UBYTE cc_get_ss_diag (USHORT curr_cause,
T_D_DISCONNECT * disc);**

A.2.2 Module cc_cfk.c

- Implemented function that extracts the SS diagnostic value from primitive
 - **GLOBAL UBYTE cc_get_ss_diag (USHORT curr_cause,
T_D_DISCONNECT * disc) {...}**

A.2.3 Module cc_rel.c

- Added handling to forward SS diagnostic byte to ACI within MNCC_DISCONNECT_IND
 - **GLOBAL void cc_disconnect (T_CC_DATA * cc_data,
T_D_DISCONNECT * disconnect)**
{
:
disc_ind->ss_diag = SS_DIAG_NOT_PROVIDED;
:
disc_ind->ss_diag = cc_get_ss_diag(curr_cause, disconnect);
:
}