MD8480A/B
W-CDMA Signalling Tester
Easy-to-understand
Signalling Tester

13th Edition

Read this manual before using the equipment.
Keep this manual with the equipment.

ANRITSU CORPORATION

Safety Symbols

To prevent the risk of personal injury or loss related to equipment malfunction, Anritsu Corporation uses the following safety symbols to indicate safety-related information. Insure that you clearly understand the meanings of the symbols BEFORE using the equipment.
Some or all of the following five symbols may not be used on all Anritsu equipment. In addition, there may be other labels attached to products which are not shown in the diagrams in this manual.

Symbols used in manual

DANGER ⚠ This indicates a very dangerous procedure that could result in death or serious injury if not performed properly.

WARNING ⚠ This indicates a hazardous procedure that could result in death or serious injury if not performed properly.

CAUTION ⚠ This indicates a hazardous procedure or danger that could result in light-to-severe injury, or loss related to equipment malfunction, if proper precautions are not taken.

Safety Symbols Used on Equipment and in Manual

The following safety symbols are used inside or on the equipment near operation locations to provide information about safety items and operation precautions. Insure that you clearly understand the meanings of the symbols and take the necessary precautions BEFORE using the equipment.

This indicates a prohibited operation. The prohibited operation is indicated symbolically in or near the barred circle.

This indicates an obligatory safety precaution. The obligatory operation is indicated symbolically in or near the circle.

This indicates warning or caution. The contents are indicated symbolically in or near the triangle.

This indicates a note. The contents are described in the box.

These indicate that the marked part should be recycled.

MD8480A/B W-CDMA Signalling Tester
Easy-to-understand Signalling Tester

1 June 2001 (First Edition)
22 December 2005 (13th Edition)

Modification History

Version	Year/Month/Day	Modification	Remarks
First Edition	2001/6/20	First Edition	-
Second Edition	2001/8/13	Chapter 2.5,Chapter 3.4, Appendix A, Appendix B, Appendix C are modified.	-
Third Edition	2002/02/27	Chapter 3.4, Chapter 3.5, Chapter 3.6, Chapter 3.7, Appendix A, Appendix B, Appendix C are modified. Corresponded to MD8480A/B.	-
Fourth Edition	2002/7/19	Chapter 1.1, Chapter 2.5,Chapter 2.6, Chapter 3.4, Chapter 3.5, Chapter 3.6,Chapter 3.7, Appendix A, Appendix B, Appendix C is modified. Appendix E(GSM/GPRS Option) is added.	-
Fifth Edition	2002/11/15	Chapter 3.6, Chapter 3.7, Appendix A, Appendix E is modified. Chapter 1.3 is added.	-
Sixth Edition	2002/12/20	Chapter 3.4, Chapter 3.7, Appendix A, Appendix C, Appendix E is modified.	-
Seventh Edition	2003/11/29	Chapter 2.4, Chapter3.7, Appendix A, Appendix C, Appendix D, Appendix E is modified.Appendix F is added.	-
Eighth	2004/05/31	Chapter 2.3, Chapter 3.4, Chapter 3.6,Chapter 3.7, Appendix A, Appendix C, Appendix E is modified. Chapter 1.4, Chapter 1.5 is added.	-
Ninth	2004/09/30	Chapter 1.4, Chapter 1.5, Chapter 2.4, Chapter 3.4,Chapter 3.7, Appendix A, Appendix E is modified.	-
Tenth	2004/12/27	Chapter 1.4, Chapter 2.4, Chapter 2.5, Chapter 3.6, Chapter 3.7, Appendix A, Appendix E is modified.	-
Eleventh	2005/01/31	Chapter 3.7, Appendix A, Appendix E is modified.	-
Twelfth	2005/06/17	Chapter 1.4, Chapter 3.2, Chapter 3.4, Chapter 3.6, Chapter 3.7, Appendix A, Appendix B, Appendix C, Appendix E are modified.	-

Thirteenth	2005/12/22	In Chapter 2.2, add explanation about the install/uninstall procedure for MX848000A. Chapter 1.3, Chapter1.4, Chapter1.5, Chapter2.3, Chapter2.4, Chapter2.5, Chapter 3.7, Appendix A, Appendix C, Appendix E, Support are modified.	

Table of Contents

Modification History iii

1. INTRODUCTION..................................... 1-1

1.1 Configuration of this Manual.. 1-1
1.2 Outline.. 1-2
1.2.1 Concept of Signalling Tester................................ 1-2
1.2.2 What can be done
by the MD8480A/B W-CDMA Signalling Tester.... 1-4
1.3 3GPP Specifications referred to in this document............ 1-4
1.4 The functions for support Service..................................... 1-5
1.5 Restriction items 1-6

2. BASIC OPERATION 2-1

2.1 Flow of Operation 2-1
2.2 Connection of Equipment 2-2
2.2.1 Connection of Equipment.................................... 2-2
2.2.2 Installation of “ MX848000A Control Software” 2-2
2.2.3 Confirmation of connection 2-8
2.2.4 UnInstallation of Control Software 2-10
2.3 Setting.. 2-13
2.3.1 Setting List 2-13
2.3.2 Parameter Setup window..................................... 2-14
2.3.3 BTS setup window 2-16
2.3.4 TX RF/RX RF setup window................................ 2-18
2.3.5 Timing setup window 2-19
2.4 Scenario... 2-21
2.4.1 About scenario 2-21
2.4.2 Description method 2-22
2.4.3 Scenario library 2-24
2.4.4 Scenario compilation.. 2-25
2.4.5 Scenario execution 2-25
2.5 Trace.. 2-26
2.5.1 Description of Trace Screen 2-26
2.5.2 Execution example... 2-32
2.6 Monitor... 2-34
2.6.1 Monitor screen 2-35
2.6.2 Main unit front LEDs 2-36
2.6.3 Information obtained from Monitor connector....... 2-38

3. APPLICATION TESTS 3-1

3.1 Harikiri (Physical Layer) Test 3-1
3.1.1 About Harikiri (physical layer) test......................... 3-1
3.1.2 Scenario to be Used... 3-2
3.1.3 Method for Harikiri (physical layer) Test............... 3-2
3.2 Call Processing Protocol Test.. 3-3
3.2.1 Method for call processing protocol test................ 3-3
3.2.2 Sample Scenarios 3-3
3.3 TE Connection Test 3-5
3.3.1 ISDN test 3-6
3.3.2 AMR Voice test... 3-6
3.3.3 Userdata test 3-6
3.3.4 IP Packet test 3-6
3.3.5 PPP (Serial connection) test 3-7
3.3.6 PPP (Built-in server) test...................................... 3-7
3.3.7 MS-to-MS test 3-7
3.4 TX Diversity... 3-8
3.4.1 Setup for Tx Diversity... 3-9
3.4.2 Parameter setup.. 3-10
3.4.3 Scenario setup 3-11
3.4.4 Various operations 3-12
3.4.5 Transmission power setting 3-15
3.5 Handover Test.. 3-17
3.5.1 Setup for handover.. 3-18
3.5.2 SHO test procedure.. 3-18
3.5.3 HHO test procedure 3-21
3.5.4 GSM Handover test 3-26
3.6 Compressed Mode Test... 3-27
3.6.1 What is Compressed Mode.................................. 3-27
3.6.2 Compressed Mode of MD8480A/B 3-28
3.6.3 Operation and how to use Compressed Mode...... 3-30
3.6.4 Compressed Mode Test sample scenario............. 3-42
3.7 OTHER TESTS 3-63
3.7.1 BLER and BTFD Measurement feature 3-63
3.7.2 AWGN 3-67
3.7.3 OCNS Test 3-68
3.7.4 FACH Measurement Occasion Function............... 3-74
3.7.5 DPCH/PRACH simultaneous boot 3-76
3.7.6 SSDT Function 3-85
3.7.7 RLC UM Reconfiguration Function 3-89
3.7.8 Variable rate AMR 3-91
3.7.9 PDCP function 3-97
3.7.10 Cmac Restriction function 3-104
3.7.11 SS_Interface Function.. 3-105
3.7.12 TM Segmentation Function 3-112

3.7.13 AM-RLC Asymmetric payload size setting functionality 3-115
3.7.14 Specifying feature for a start timing of TE transmission/the number of transmission 3-118
3.7.15 CFN Timing Indication feature 3-122
3.7.16 RLC Stop, RLC Continue function 3-127
3.7.17 W-CDMA CSD 3-130
3.7.18 SFN Timing Indication function 3-133
3.7.19 Real time display on TRACE window 3-137
3.7.20 PipeLine Delay Reduction Function 3-139
3.7.21 DPCCH indication function 3-148
3.7.22 Preamble acquisition function 3-153
3.7.23 Lossless InterRAT Cell Change Function 3-158
3.7.24 W-CDMA CBS Function 3-167
3.7.25 RLC AM Reconfiguration Function 3-176
3.7.26 RLC UM Special Length Indicator 3-178

A. SCENARIO LIBRARIES A-1

A.1 DETAILS OF EACH FUNCTION A-3
A.2 VARIABLE USED IN EACH FUNCTION A-45
A.2.1 Basic types A-45
A.2.2 Structure A-45
A.2.3 Common variables A-72
A.3 Range of Layer 1 Parameter A-89
A.3.1 DPCH A-90
A.3.2 PDSCH A-97
A.3.3 S-CCPCH A-97
A.3.4 PRACH A-97

B. DETAILS OF HARIKIRI (PHYSICAL LAYER) TEST B-1

B.1 HARIKIRI (PHYSICAL LAYER) TEST LIST B-1
B.2 OPERATION IN EACH SCENARIO B-4
B.2.1 Start_xxx.dl B-4
B.2.2 xxx_Data(xxx).dll B-4
B.3 LAYER 1 OPERATION PARAMETER WHEN THE ATTACHED SCENARIOA IS USED B-5
B.3.1 Procedure and Operation Common to Uplink B-6
B.3.2 Procedure and Operation Common to Downlink .. B-7
B.3.3 P_SCH, S_SCH, P_CPICH, and P_CCPCH Settings B-8
B.3.4 S_CCPCH(PCH+FACH) Setting B-9
B.3.5 RACH receive and AICH transmit Settings B-11
B.3.6 Stand-Alone DCCH Setting B-12

B.3.7 Stand-Alone DCCH(TTI=80m) Setting B-14
B.3.8 UDI+DCCH Setting B-16
B.3.9 AMR Voice(12.2[kbps])+DCCH Setting B-18
B.3.10 DCH 8Channel Setting B-20
B.3.11 Packet Convolutional Coding(Downlink 8[kbps]/Uplink 32[kbps])+DCCH Setting B-25
B.3.12 Packet 32[kbps](Downlink TTI=20[ms]) +DCCH Setting B-27
B.3.13 Packet 64[kbps](Downlink TTI=10[ms]) +DCCH Setting B-29
B.3.14 Packet 128[kbps](Downlink TTI=10[ms]) +DCCH Setting B-31
B.3.15 Packet 384[kbps](Downlink TTI=10[ms]) +DCCH Setting B-34
B.3.16 AudioVisual(32[kbps])+DCCH Setting B-36
B.3.17 AudioVisual(64[kbps])+DCCH Setting B-38
B.3.18 AMR+Packet 32[kbps](Downlink TTI=10[ms]) +DCCH Setting B-40
B.3.19 AMR+Packet 64[kbps](Downlink TTI=10[ms]) +DCCH Setting B-43
B.3.20 AMR+Packet 384[kbps] +DCCH(TTI=20ms) Setting B-47
B.3.21 AMR+UDI+DCCH Setting B-53
B.3.22 Reference Measurement Channel (12.2[kbps]) Setting B-56
B.3.23 Reference Measurement Channel (64[kbps]) Setting B-58
B.3.24 Reference Measurement Channel (144[kbps]) Setting B-60
B.3.25 Reference Measurement Channel (384[kbps]) Setting B-62
B.3.26 Reference Measurement Channel (BTFD) Setting B-64
B.3.27 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH Setting B-68
B.3.28 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH Setting...................... B-73
B.3.29 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH Setting B-79
B.3.30 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet [64kbps] + DCCH Setting B-90

B.3.31 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 128[kbps] + DCCH Setting........................ B-96
B.4 HOW TO CHECK THE HARIKIRI (PHYSICAL LAYER) TEST UPLINK OPERATION B-102
B.4.1 DPCH Demodulation.. B-102
B.4.2 RACH Demodulation.. B-104
B.4.3 Signals from the RxBB Front Panel B-106

C. DETAILS OF TE CONNECTION TEST.... C-1

C.1 ISDN TEST C-4
C.1.1 Sample Scenarios.. C-4
C.1.2 Scenario Configuration C-5
C.1.3 Handling Calling Party Number............................ C-9
C.1.4 Executing Scenarios C-10
C.2 AMR VOICE TEST C-11
C.2.1 Sample Scenarios.. C-11
C.2.2 Scenario Configuration C-12
C.2.3 Executing Scenarios C-17
C.3 USERDATA TEST C-18
C.3.1 Sample Scenarios.. C-18
C.3.2 Scenario Configuration C-18
C.3.3 Executing Scenarios C-20
C.4 IP PACKET TEST.. C-21
C.4.1 Sample Scenarios.. C-21
C.4.2 Scenario Configuration C-21
C.4.3 Executing Scenarios C-24
C.4.4 Setting IP address of Mobile Station side C-24
C.4.5 ServerPC IP Address Setting Function................. C-25
C.5 PPP (Serial connection) TEST C-26
C.5.1 Sample Scenarios.. C-26
C.5.2 Scenario Configuration C-26
C.5.3 Executing Scenarios C-29
C.5.4 Restriction C-29
C.6 PPP (Built-in SERVER) TEST C-30
C.6.1 Sample Scenarios.. C-30
C.6.2 Explanation of Scenarios C-31
C.6.3 Executing Scenarios C-32
C.6.4 Connection... C-33
C.6.5 Protocol Stack C-34
C.6.6 PPP Option Setting C-36
C.6.7 RFC1877 Setting Function................................... C-37
C.7 MS-TO-MS TEST C-41
C.7.1 Sample Scenarios.. C-41
C.7.2 Scenario Configuration C-41

C.7.3 Executing Scenarios... C-46

D. EXPLANATION OF SCENARIOS............ D-1

D.1 Explanation of Sample Scenario
for Harikiri (Physical Layer) Test...................................... D-1
D.1.1 "Start_AMR.c" D-1
D.1.2 "DCH_data(AMR).c".. D-4
D.2 Sample Scenario for Protocol Sequence.......................... D-6
D.2.1 "Idle.c" D-6
D.2.2 "Orig_AMR.c" D-8

E. GSM/GPRS OPTION E-1

E.1 Outline of GSM/GPRS option E-1
E.2 Option Composition needed for GSM/GPRS option.......... E-1
E.3 Required Software components for GSM/GPRS Option ... E-2
E.3.1 Software option E-2
E.4 GSM/GPRS Option operation.. E-3
E.4.1 Connection and Operation of equipment E-3
E.4.2 Control Software GUI specific
to GSM/GPRS option E-3
E.4.3 Monitor and LED specifications............................ E-8
E.5 GSM/GPRS tests E-11
E.5.1 GSM call origination and termination test E-11
E.5.2 GPRS data communication test
(Between Server and Client PC).......................... E-14
E.5.3 GSM CSD test... E-19
E.5.4 Handover test E-26
E.5.5 Frequency Hopping Test E-28
E.5.6 GPRS Harikiri (physical layer) test........................ E-30
E.6 Scenario Library... E-31
E.6.1 Scenario library function list E-31
E.6.2 Details of each function....................................... E-33
E.6.3 Variable used in each function E-70
E.6.4 Scheduling of BCCH and SACCH........................ E-84
E.6.5 Explanation regarding sample scenario E-86
E.6.6 Description of each sample scenario E-88
E.6.7 Restriction of Frequency and Timeslot settings
for RF channels.. E-97
E.7 Modification method for GPRS Scenarios E-99
E.7.1 About this manual... E-99
E.7.2 Scenario modification method
(Sample scenarios attached to v3.20 or before) ... E-99
E.7.3 Scenario modification method
(Sample scenarios attached to v5.11 or before) ... E-103

F. ROUTER CONNECTION F-1

F.1 Outline of specification.. F-1
F.2 Setting method.. F-2
F.3 Note item and Restriction item.. F-4
F.3.1 Note item.. F-4
F.3.2 Restriction item F-4

Support

About expansion of Option key

1. INTRODUCTION

This document describes the function and operation of the MD8480A/B W-CDMA Signalling Tester, mainly on how to use software and measurement method.

Also see MD8480A/B Operation Manual, Download Manual and Q&A Book available as documents for the MD8480A/B W-CDMA Signalling Tester. Figure 1-1 shows the document configurations related to the MD8480A/B W-CDMA Signalling Tester.

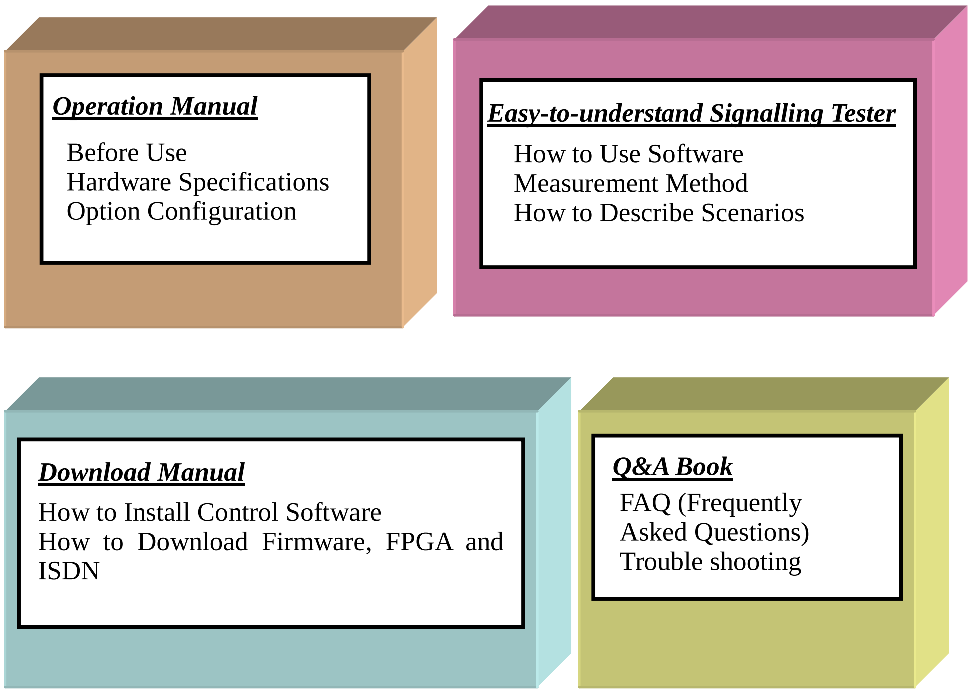


Figure 1-1 Configuration of MD8480A/B W-CDMA Signalling Tester documents

1.1 Configuration of this Manual

Configuration of this manual is as follows:

CHAPTER 1: INTRODUCTION

Describes the concept and features of the MD8480A/B Signalling Tester.

CHAPTER 2: BASIC OPERATION

Describes basic operation of the MD8480A/B Signalling Tester.

CHAPTER 3: APPLICATION TESTS

Describes the main test items of the MD8480A/B Signalling Tester.

APPENDIX A~D:

Describes parameters necessary for using the MD8480A/B Signalling Tester and details on how to use the MD8480A/B Signalling Tester.

APPENDIX E:

Describes the GSM/GPRS Option.

APPENDIX F:

Describes the ROUTER CONNECTION.

1.2 Outline

1.2.1 Concept of Signalling Tester

1.2.1.1 About base station simulator

When a communications equipment maker develops a mobile station, etc., it cannot conduct a connection test using a base station in operation. Also, in recent years, data transfer speed of communications systems is accelerating and the communications systems are updated more frequently. This leads to development of a mobile station and a base station in parallel thus inevitably requiring the need for a base station simulator. In this situation, a measurement system to work equivalently to a base station is required, and it is generically called as a base station simulator. Figure 1-2 shows a system diagram of a base station simulator, and a system diagram including an actual W-CDMA system and a network system.

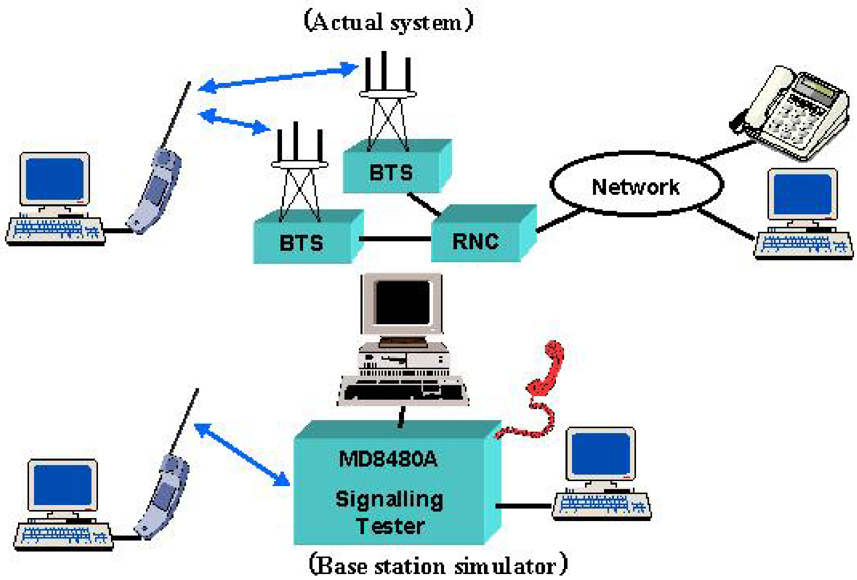


Figure 1-2 Concept of base station simulator

1.2.1.2　About Signalling Tester

As well as a feature corresponding to a base station simulator, the signalling tester is a generic term for measuring instruments equipped with a protocol test feature such as execution of a call connection test sequence and other sequences, and trace feature to check details of stepwise information exchange. The signalling tester also has a feature to identify any abnormality in specific steps of a connection sequence in the course of a procedure to establish a communication line in case a fault has taken place in establishing the call communication.　See Figure 1-3 that shows the outline of the feature operation of the signalling tester.

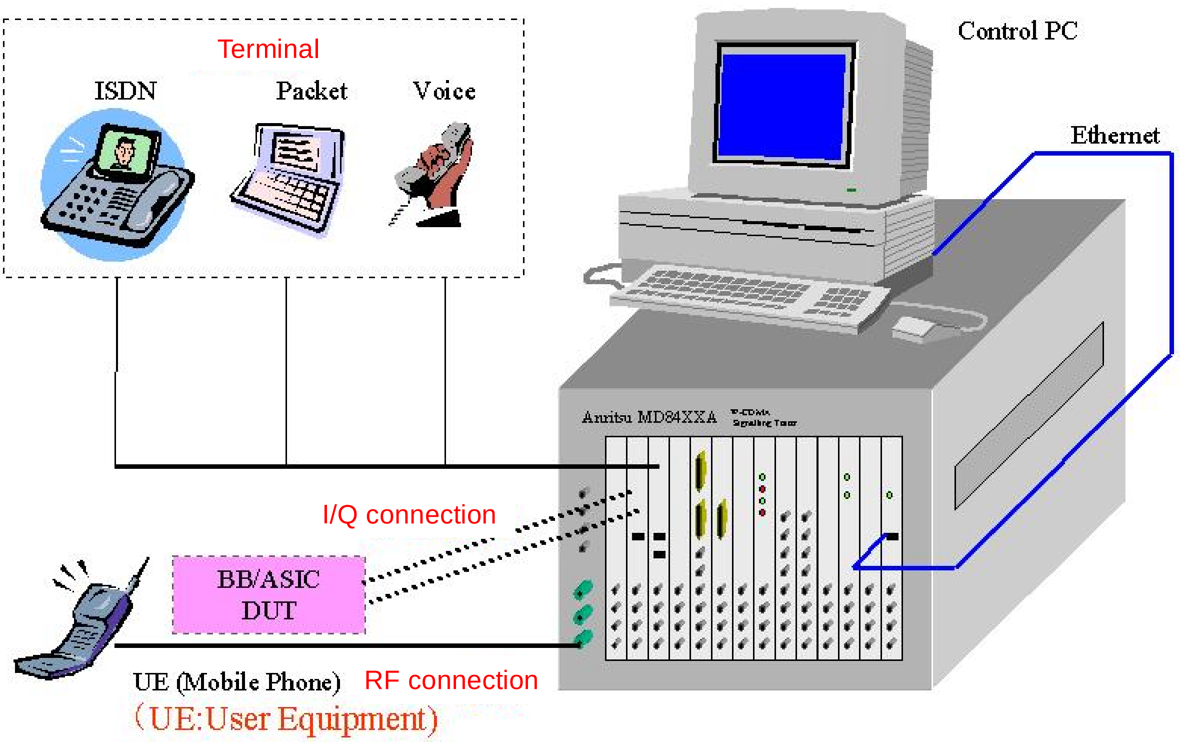


Figure 1-3　Signalling Tester system diagram

1.2.2 What can be done by the MD8480A/B W-CDMA Signalling Tester

The MD8480A/B W-CDMA Signalling Tester (hereinafter referred to as the Signalling Tester) is equipped with features necessary for feature test of a mobile station used in the W-CDMA, a third-generation mobile telecommunication system. The Signalling Tester can be used as a base station simulator. Available tests are feature test of modulation/demodulation of a mobile station, call processing sequence tests covering location registration, origination, termination, handover, mobile station disconnection and network disconnection, and a variety of application tests covering voice communications, packet communications and the MS-to-MS test. The Signalling Tester is an optimum tool for developing a mobile station and application software for a mobile station.

- Allows feature test of modulation/demodulation of W-CDMA mobile station
- Allows protocol sequence test of W-CDMA mobile station
  (Parameters and sequences used for the call processing protocol test can be defined arbitrarily.)
- Allows application tests covering voice communications, packet communications, the MS-to-MS test, etc.

1.3 3GPP Specifications referred to in this document

3GPP Specifications referred to in this document are versions listed below. (All are Release99.)

3GPP Specifications number	Version number
TS25.101	3.a.0
TS25.211	3.a.0
TS25.212	3.9.0
TS25.213	3.7.0
TS25.214	3.a.0
TS25.215	3.9.0
TS25.306	3.5.0
TS25.322	3.a.0
TS25.324	3.9.0
TS25.331	3.a.0
TS34.108	3.7.1
TS34.109	3.5.0
TS34.121	3.8.0
TR25.925	3.5.0

1.4 The functions for support Service

If you contract support service, Ver. 5.xx become able to be used.
The functions which become usable by using Ver. 5.xx are shown as follows.

Function Name	Section Number	Usable version
DPCH /PRACH simultaneous boot	3.7.5	Ver.5.00 or later
SSDT Function	3.7.6	Ver.5.00 or later
RLC UM Reconfiguration Function	3.7.7	Ver.5.00 or later
Variable rate AMR	3.7.8	Ver.5.00 or later
Cmac Restriction function	3.7.10	Ver.5.00 or later
SS-Interface function	3.7.11	Ver.5.00 or later
TM Segmentation Function	3.7.12	Ver.5.10 or later
MultiTGP	3.6	Ver.5.20 or later
AM-RLC Asymmetric payload size setting functionality	3.7.13	Ver.5.20 or later
Specifying feature for a start timing of TE transmission/the number of transmission	3.7.14	Ver.5.20 or later
CFN Timing Indication feature	3.7.15	Ver.5.20 or later
RLC Stop, RLC Continue function	3.7.16	Ver.5.20 or later
W-CDMA CSD	3.7.17	Ver.5.21 or later
BtsReadSFN()	A.1	Ver.5.30 or later
SFN Timing Indication function	3.7.18	Ver.5.30 or later
PipeLine Delay Reduction Function	3.7.20	Ver.5.30 or later
DPCCH indication function	3.7.21	Ver.5.30 or later
Preamble acquisition function	3.7.22	Ver.5.30 or later
Lossless InterRAT Cell Change Function	3.7.23	Ver.5.30 or later
W-CDMA CBS Function	3.7.24	Ver5.40 or later
RLC AM Reconfiguration Function	3.7.25	Ver5.40 or later

1.5 Restriction items

There are the following restrictions in both Ver. 3.xx and Ver.5.xx.
These restrictions are not scheduled to be released.

No	Layer	Content
(1)	Total	Operation of Monitor has been unconfirmed.
(2)		The parameter for WaitTime which is a scenario library need to be multiples of 100. This library can be worked even if the value for the parameter is less than 100, but the waiting time may be insatiable.
(3)		Don’ t transmit OCNS, in condition “W-CDMA+GSM” is selected at “BaseBand and RF Unit Connection” on parameter setup window of Control Software.
(4)	TE	In built-in PPP Server, the following protocols are out of correspondence. *CCP(compression control protocols:RFC1962)
(5)	PDCP	The following functionalities and worings are not supported. - Header compression function for IP data (RFC2507) - Lossless SRNS Relocation function - Buffering and retransmission of PDCP SDUs in case that more SDUs than 256 are not acknowledged.
(6)	RLC	There are the following restrictions on RLC functions. *15bit Length Indicators is not supported. *Timer Discard, Timer EPC, Timer MRW are not supported. *Maximum Window Size (Tx Window Size/Rx Window Size)is 2048. *SDU discard function is non-correspondence.
(7)		When AM RLC is already configured and you want to clear and configure RLC AM again, execute CrlcConfig(CRLC_AM_RELEASE) before new configuration. The working of MD8480 is not guaranteed except to using RLC AM Reconfiguration Function when CrlcConfig(CRLC_AM_ESTABLISH) or CrlcConfig(CRLC_AM_ESTABLISH_WITH_CIPHER) is executed for configured AM RLC without CrlcConfig(CRLC_AM_RELEASE) Please refer 3.7.25 about RLC AM Reconfiguration Function which can reconfigure RLC AM parameter without SN and Length.
(8)	MAC	There are the following restrictions on MAC functions. 1) With V3.3x, Data length of MAC SDU from RLC is not referred for downlink TFCI selection. (It's judged only by the number of Transport Block.) With versions after V5.10, this limitation can be cancelled. Please refer section 3.7.8 Variable rate AMR in “easy-to-understand” about detail. 2) Priority for each Logical Channel are not available.
(9)	PHY	Reception for PRACH in SHO test has been confirmed only on BTS#1. (Reception on BTS#2 and #3 has not been confirmed.)

<table>
<tr><td>(10)</td><td rowspan="4">PHY</td><td>When using a CphyRlSetup() function continuously for two BTS(s) in uplink, it will take approximately 40ms before the first processing of a CphyRlSetup() function is completed.
EX) CphyRlSetup (UNIT_BTS1,…....);
CphyRlSetup (UNIT_BTS2,…....);
When the next function was performed before processing of the first function is ended, processing of the next function becomes unavailable.</td></tr>
<tr><td>(11)</td><td>DPCH/PRACH simultaneous activation functionality has a structure that MD8480 alternatively waits to receive DPCH and PRACH in time division. So while it is waiting to receive one channel, it often fails to do the other channel. This is why MD8480 often can not synchronize pilot in uplink DPCH on this feature. (Success rate is about 80%.) Especially this phenomenon appears when it receives RRC Connection Setup Complete from UE.</td></tr>
<tr><td>(12)</td><td>MD8480 often can not receive DPCH whose symbol rate is 960ksps on DPCH/PRACH simultaneous activation functionality.(Success rate is about 80%.)</td></tr>
<tr><td>(13)</td><td>When transport channel of the following setting is two channels, MD8480B doesn't operate normally.
- Transport channels are mapped on downlink DPCH.
- TTI is 20ms and Coding type is Turbo Coding.</td></tr>
<tr><td>(14)</td><td>BMC</td><td>- Do not support ANSI-41.
- BMC message for both BTS1 and BTS2 cannot be transmitted simultaneously at HHO and SHO.
- Only one the logical channel (CTCH) to transmit the BMC message can be set.</td></tr>
</table>

2. BASIC OPERATION

2.1 Flow of Operation

Figure 2- 1shows the basic use procedure of the Signalling Tester. The right half of the figure shows the section in this manual that gives details of the corresponding procedure.

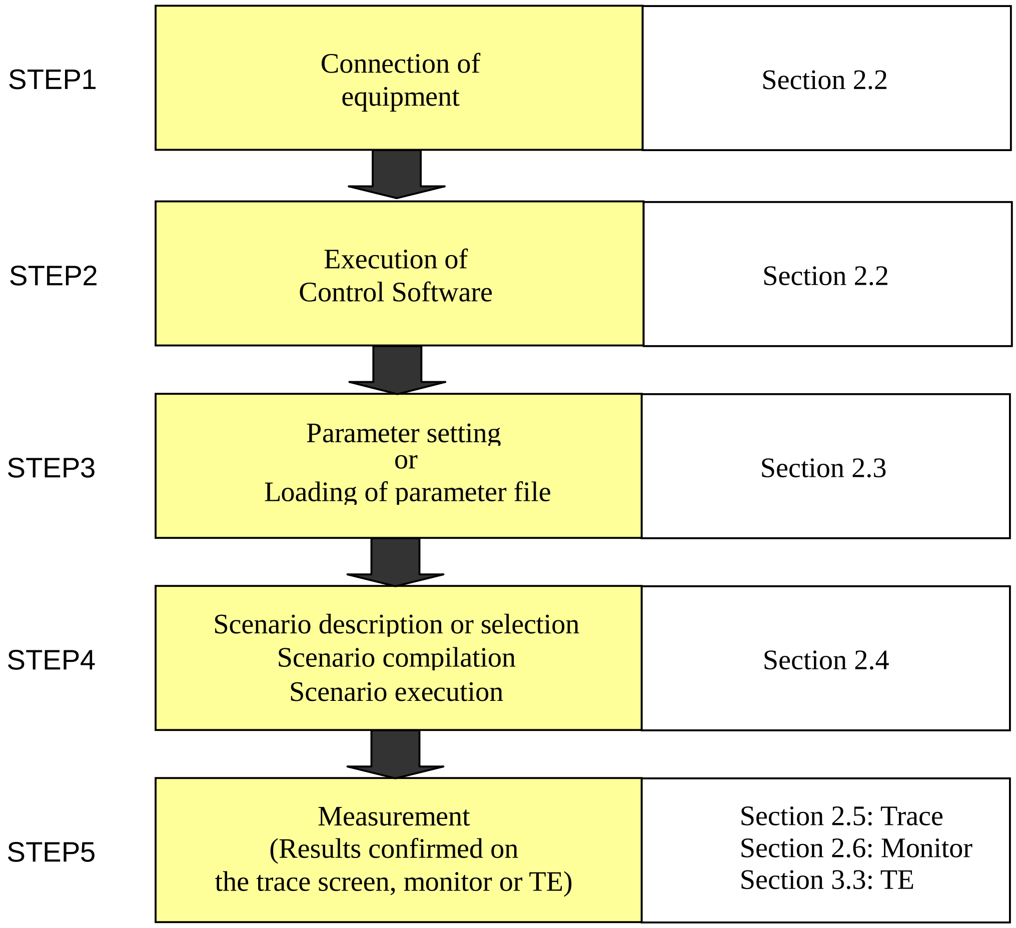


Figure 2- 1 Procedure for operation of Signalling Tester

2.2　Connection of Equipment

2.2.1　Connection of Equipment

Connect the equipment as shown in Figure 2-2.

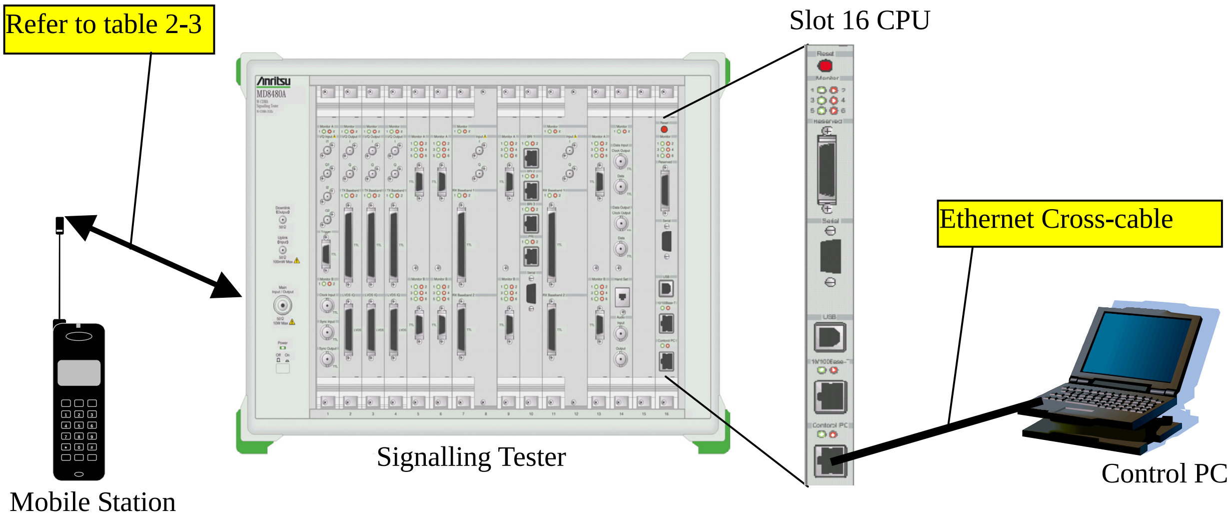


Figure 2-2　Connection diagram

2.2.2　Installation of "MX848000A Control Software"

(1) Preparation

When previous version of Control Software(MX848000A) is installed, delete it with the following procedures.

a) Delete Signalling Tester and scenario.

Windows 95/98 and using Visual C++ V6.0 : Signalling Tester and scenario

Windows NT/2000/XP and using Visual C++ V6.0 : Signalling Tester and scenario_NT

When Visual C++®.net™ 2002 is used : Signalling Tester and scenario_for.NET

When Visual C++®.net™ 2003 is used : Signalling Tester and scenario_for.NET2003

b) Delete folder MX848000 located on C:¥, or rename it.

2) Extract the MX848000A_XXX.zip. (XXX varies depending on software version.)

Normal extraction provides the following files.

Folder　　　　　MX848000

Shortcut　　　　Signalling Tester,

scenario,

scenario_NT

scenario_for.NET

scenario_for.NET2003

3) Move the folder MX848000 to C:¥.

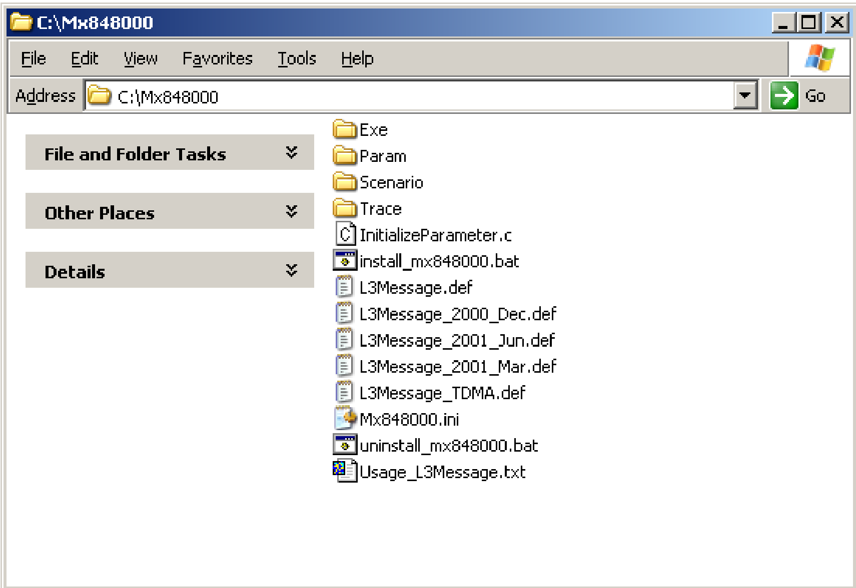


Figure 2-3　C:¥MX848000 Folder

4) Move the following shortcuts located in C:¥Temp to desktop.

For Windows95/98 and Visual C++ V6.0 users:

Shortcuts scenario and Signalling Tester

For WindowsNT/2000/XP and Visual C++ V6.0 users:

Shortcuts scenario_NT and Signalling Tester

When Visual C++®.net™2002 is used:

Shortcuts scenario_for.NET and Signalling Tester

When Visual C++®.net™2003 is used:

Shortcuts scenario_for.NET2003 and Signalling Tester

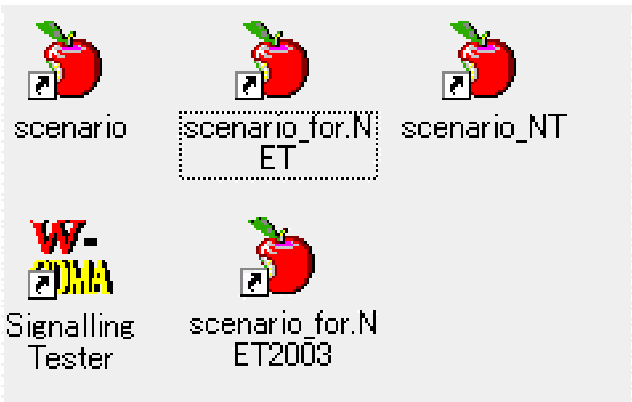


Figure 2-4 Shortcuts to be moved

5) Registry setting

Set registry if control PC is the following environment.

Windows2000 Service Pack 4 or greater.

Windows XP Service Pack 2 or greater.

(Note) The operation when the registry is set in the environments other than the above-mentioned is not guaranteed.

Double click the "C:¥MX848000¥install_mx848000.bat" to set the registry keys.

Click the OK button after reboot message shown in Figure 2- 5 is displayed, then re-boots the control PC.

Figure 2- 5 Reboot message

If the registry keys have already been set, the following message Figure 2- 6 will be displayed.

It is not necessary to reboot in this case. Click the OK button.

Figure 2- 6 Registry message

From here, messages displayed when registry was not able to be set normally are explained.

When the IP address of NIC of the control PC wasn’ t set to 1.1.0.x (x: 1 to 99), registry can’ t be set.

In this case, the message shown in Figure 2- 7 will be displayed.

Click the OK button, then after IP address is set, execute install_mx848000.bat again.

Figure 2- 7 Warning message (IP address issue)

When the OS version of the control PC isn’ t proper, registry can’ t be set.

It has to be Windows2000 SP4/ WindowsXP SP2 or later version.

In this case, the message shown in Figure 2- 8 will be displayed.

Click the OK button, then after Windows is updated, execute install_mx848000.bat again.

Figure 2- 8 Warning message (OS version issue)

When registry can' t be set by other issues, Error messages shown in Figure 2- 9 or Figure 2- 10 will be displayed.

Click the OK button, then please contact our support and let them know which message is displayed.

Figure 2- 9 Error message (1)

Figure 2- 10 Error message (2)

Installation of Control Software (MX848000A) is complete with the procedures above.

2.2.3 Confirmation of connection

Execute Control Software

Double click the Shortcuts Signalling Tester to execute Control Software (MX848000A).

Main Window (Figure 2- 11) is displayed.

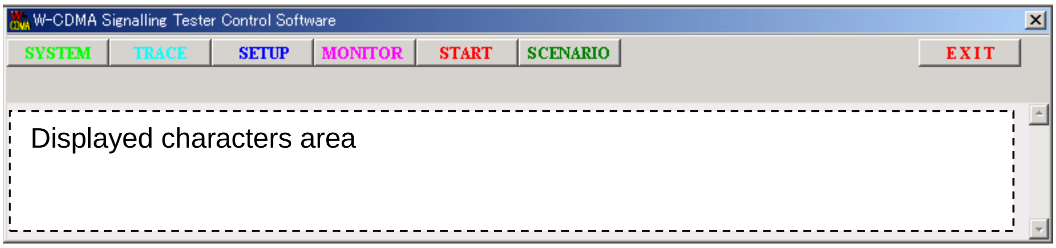


Figure 2- 11 Main Window

1　SYSTEM
System window opens, and enables to input serial number and refer to unit configuration or option information.

2　TRACE
Trace window opens. Refer to section 2.5.

3　SETUP
Setup window opens. Refer to section 2.3.2.

4　MONITOR
Monitor window opens. Refer to section 2.6.1.

5　START
Starts measurement in order to execute scenario.

6　SCENARIO
Scenario selection window opens. Select scenario to be executed.

7　EXIT
Control Software exits.

8　Displayed characters area
Displays characters by SequenceDisp().

Confirmation of connection

When the installation is complete, turn on the Signalling Tester. (The power switches are at two locations, on the front and rear of the tester.) Then click on SYSTEM of Control Software. Enter the serial number of the Signalling Tester (described on the rear of the Signalling Tester) on the system screen and press the OK button.

On the system screen, click on "Information Read." This allows you to check the versions of the firmware, FPGA and ISDN/PPP. In case you cannot check the versions, download of the firmware, FPGA and/or ISDN/PPP may have failed. See the Download Manual and try download again.

If wishing to update downloaded firmware, FPGA and/or ISDN/PPP here, see the Download Manual to perform updates. About option, the information of available option for downloaded firmware is shown by clicking on “Information Read”.

If the version later than v5.40 is used, the information (Control PC, FW, FPGA, ISDN, TDMA and software options) can be outputted as a text file (Information.txt) after clicking on “Information Read”.

The file: Information.txt is generated in C:¥Mx848000¥.

2.2.4 UnInstallation of Control Software

The following is uninstallation process of Control Software.

1) Clear registry setting

Double click the "C:¥MX848000¥uninstall_mx848000.bat" to clear the registry keys.

Click the OK button after reboot message is displayed, then reboot the control PC.

Figure 2- 12 Reboot message

When the registry keys has already been deleted, the following message Figure 2- 13 will be displayed.

It is not necessary to reboot in this case. Click the OK button.

Figure 2- 13 Registry message

From here, messages displayed when registry was not able to be cleared normally are explained.

When the IP address of NIC of the control PC wasn’ t set to 1.1.0.x (x: 1 to 99), registry can’ t be cleared.

In this case, the message shown in Figure 2- 14 will be displayed.

Click the OK button, then after IP address is set, execute uninstall_mx848000.bat again.

Figure 2- 14 Warning message (IP address issue)

When the OS version of the control PC isn’ t proper, registry can’ t be cleared.

It has to be Windows2000 SP4/ WindowsXP SP2 or later version.

In this case, the message shown in Figure 2- 15 will be displayed.

Click the OK button, then after Windows is updated, execute uninstall_mx848000.bat again.

Figure 2- 15 Warning message (OS version issue)

When registry can’ t be cleared by other issues, Error messages shown in Figure 2- 16 or Figure 2- 17 will be displayed.

Click the OK button, then please contact our support and let them know which message is displayed.

Figure 2- 16 Error message (1)

Figure 2- 17 Error message (2)

2) Delete folder

Delete the folder "C:¥MX848000" and shortcuts installed into the control PC.

Uninstallation of Control Software (MX848000A) is complete with the procedures above.

2.3 Setting

In Signalling Tester, Control Software(GUI) on Control PC sets the hardware and layer1.Scenario can also. If Control Software and Scenario set the same item or parameters, the setting of Scenario will be valid.

2.3.1 Setting List

Table 2- 1 shows a list of setting items.

Table 2- 1 Setting list

Major classification	Minor classification	Setting item	Setting method
Hardware of Signalling Tester	–	Clock selection (internal/ external)	Described in section 2.3.5
		I/O connector (Main, Sub, or Analog IQ)	Described in section 2.3.2
Layer1	RF	Downlink Level, Frequency	Described in section 2.3.2 or 2.4.*)
		Uplink Level, Frequency	Described in section 2.3.2 or 2.4.*)
	Baseband	Slot Format, Code, and Level of each channel	Described in section 2.3.2 or 2.4.*)
		TFCS of each channel	Described in section 2.4
		Frame Timing	Described in section 2.3.5
		Downlink TPC bit pattern	Described in section 2.4
		Uplink power control with uplink TPC bits	Described in section 2.4
Layer2(MAC,RLC)	–	Described in section 2.4	Described in section 2.4
Layer3(RRC)	–	Described in section 2.4	Described in section 2.4
Other	AWGN test	AWGN level	Described in section 2.3.3
	TE connection test (ISDN, Voice, ppp, AV, MS-to-MS, multi call)	Described in section 3.3	Described in section 3.3
	TX Diversity test	Described in section 3.4	Described in section 3.4
	Handover test	Described in section 3.5	Described in section 3.5
	Compressed Mode test	Described in section 3.6	Described in section 3.6
	GSM Handover test	Described in Appendix E	Described in Appendix E

*) For details of level setting, see the "Operation Manual Appendix G."

2.3.2　Parameter Setup window

Executing Control Software and Clicking on the SETUP button in the MAIN window opens the Parameter Setup window shown in Figure 2- 18.

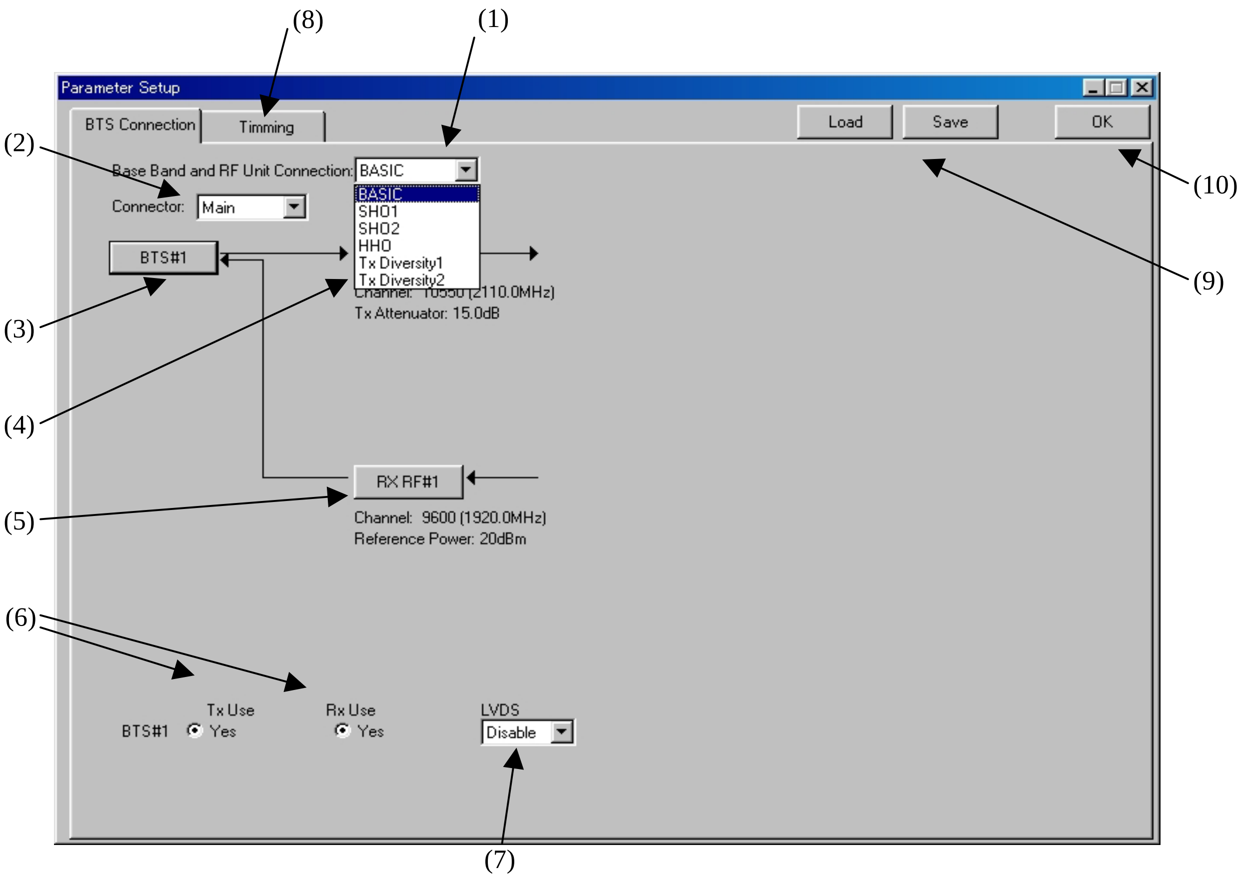


Figure 2- 18　Parameter Setup window

(1)　Baseband and RF Unit Connection

Set the hardware to be used.

Relationship between the selected setting and hardware is shown in Table 2-2.

Table 2-2 Hardware setting

Setting	Description
BASIC	Set to use the Signalling Tester in the standard configuration.
SHO1	Set when Soft Handover test using single antenna is to be conducted.
SHO2	Set when Soft Handover test using 2 antennas is to be conducted. (Additional RF unit is needed.)
HHO	Set when Hard Handover test is to be conducted. (Additional RF unit is needed.)
TX Diversity1	Set when TX Diversity test using single antenna is to be conducted.
TX Diversity2	Set when TX Diversity test using 2 antennas is to be conducted. (Additional RF unit is needed.)

(2) Connector

Set the connector used to connect to a mobile station.
Relationship between the selected setting and used connector is shown in Table 2-3.

Table 2-3 Selection of connector used to connect to a mobile station

Setting	Description
Main	Outputs as the RF signal from the Main connector on the front of the main unit. Connectors and level difference are shown in Figure 2- 19.
Sub	Inputs/Outputs as the RF signal from the Sub connector on the front of the main unit. Connectors and level difference are shown in Figure 2- 19.
Analog I,Q	Inputs/Outputs as the Analog Baseband signal by using IQ connectors of TX Baseband and RX Baseband.
Digital I,Q	Inputs/Outputs as the Digital Baseband signal by using the half pitch connectors of TX Baseband and RX Baseband. Note that the signal must be processed as a signal sampled at five times the Chip rate. Pin arrangements are shown in Table 2-11 and Table 2-12.

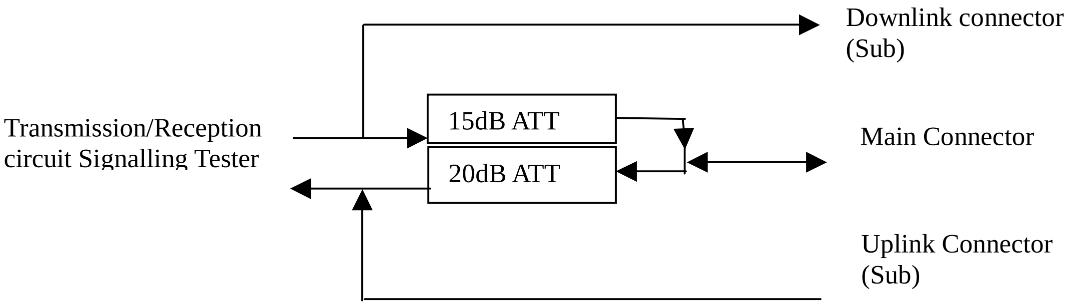


Figure 2- 19 Main and Sub connector

(3) BTX #X

Opens the window for setting the layer 1 parameters of BTS (base station feature) (see section 2.3.3).

(4) TX RF #X

Opens the window for setting TX RF (see section 2.3.4).

(5) RX RF #X

Opens the window for setting RX RF (see section 2.3.4).

(6) Tx Use, Rx Use

Allows to set the use of sending feature and receiving feature for each BTS.

(7) LVDS

Sets whether to enable LVDS connector. The LDVS connector is the one on the TX Baseband board. A fading simulator (baseband: PropSIM DBB of Electrobit) can be connected to this connector.

(8) Timing

Opens the timing panel (see section 2.3.5).

(9) LOAD, SAVE

Opens the file load/save screen. You can save set parameters in a file (*.pml) on the file save screen. You can read the file (*.pml) on the file load screen to reproduce the saved setting.

(10) OK.

Enables setting changes.

For details of level setting, see "Operation Manual Appendix G."

2.3.3 BTS setup window

(1) TX Setup

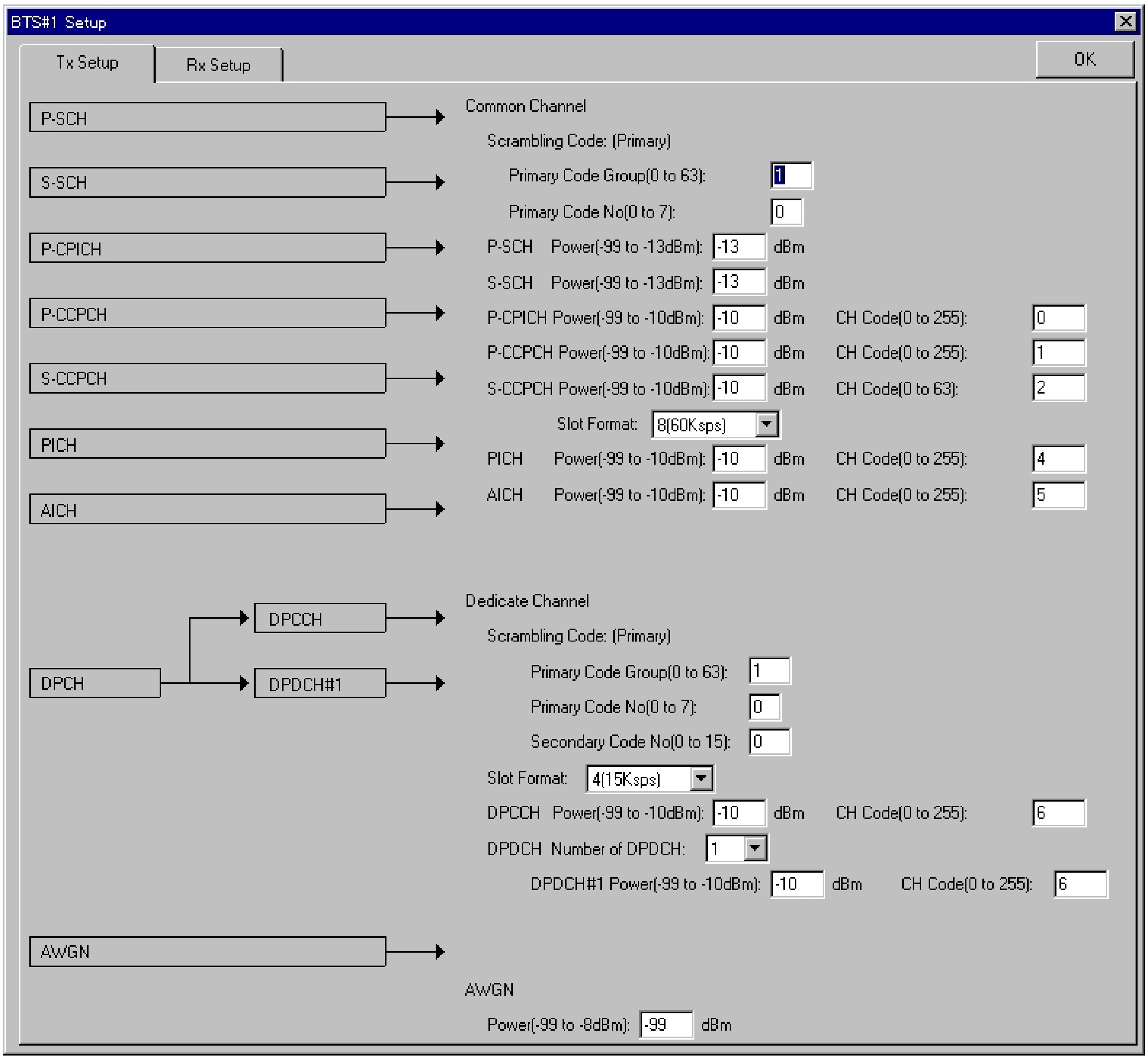


Figure 2- 20 BTS Setup window (TX Setup)

You can set the power, code and slot format of each channel in this window. You can change the setting in a scenario (CphyRlSetup() function) also. For details, see section 2.4 and "Appendix A SCENARIO LIBRARIES."

(2)　RX setup window

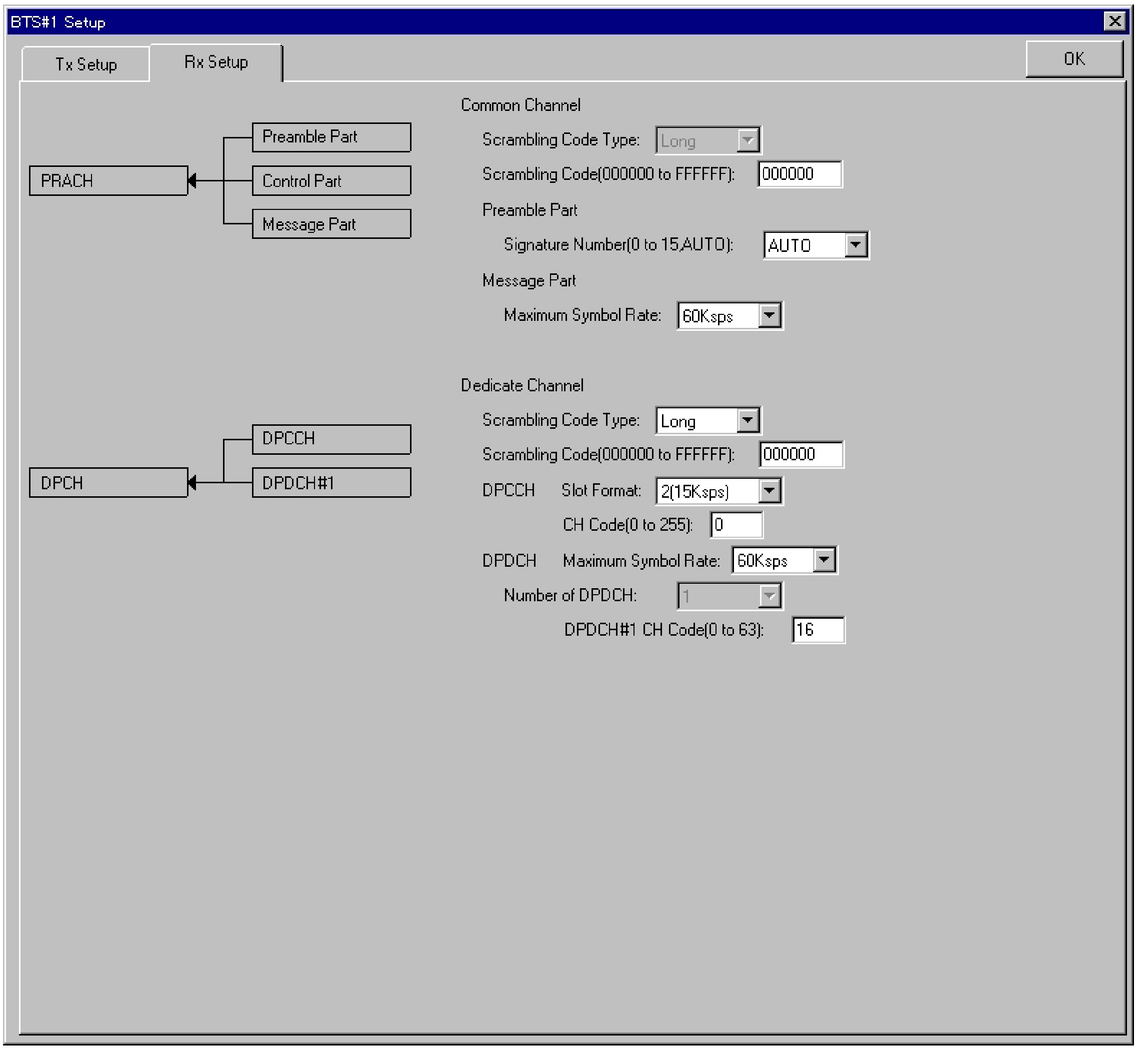


Figure 2- 21 BTS Setup window (RX Setup window)

You can set the code, slot format and rate of the receive (uplink) signal.　You can change the setting in a scenario (CphyRlSetup() function) also.　For details, see section 2.4 and "Appendix A　SCENARIO LIBRARIES."

2.3.4 TX RF/RX RF setup window

(1) TX RF setup window

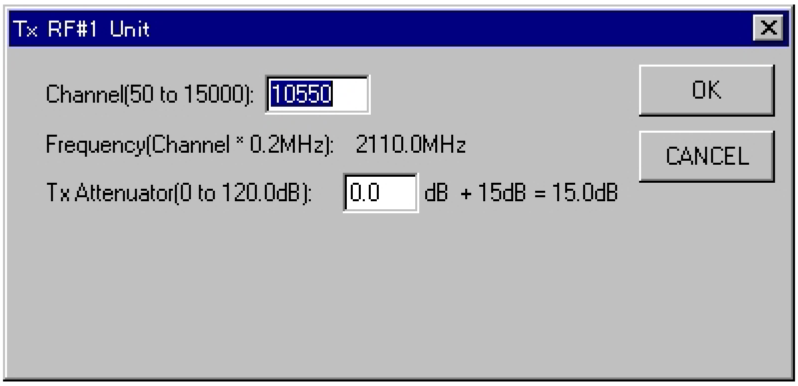


Figure 2- 22 TX RF setup window

You can set the values of the frequency of the transmit (downlink) RF signal for the Signalling Tester and the Attenuator. You can change the setting in a scenario (BtsFrequency() and BtsAttenuator() function) also. For details, see section 2.4 and "Appendix A SCENARIO LIBRARIES."

(2) RX RF setup window

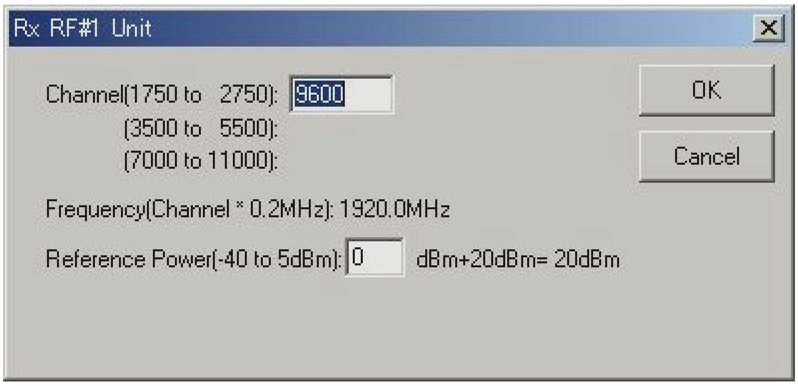


Figure 2-23 RX RF setup window

You can set the frequency and level of the receive (uplink) RF signal for the Signalling Tester. Set the approximate value of the total power of the uplink signal in Reference Power. You can change the setting in a scenario (BtsFrequency() and BtsAttenuator() function) also. For details, see section 2.4 and "Appendix A SCENARIO LIBRARIES." When you use PC:V5.30a or later versions, uplink frequency range is extended and you can specify 1750-2750 channel(350MHz-550MHz) and 3500-5500 channel (700-1100MHz) in addition to 7000-11000 channel(1400MHz-2200MHz).

2.3.5　Timing setup window

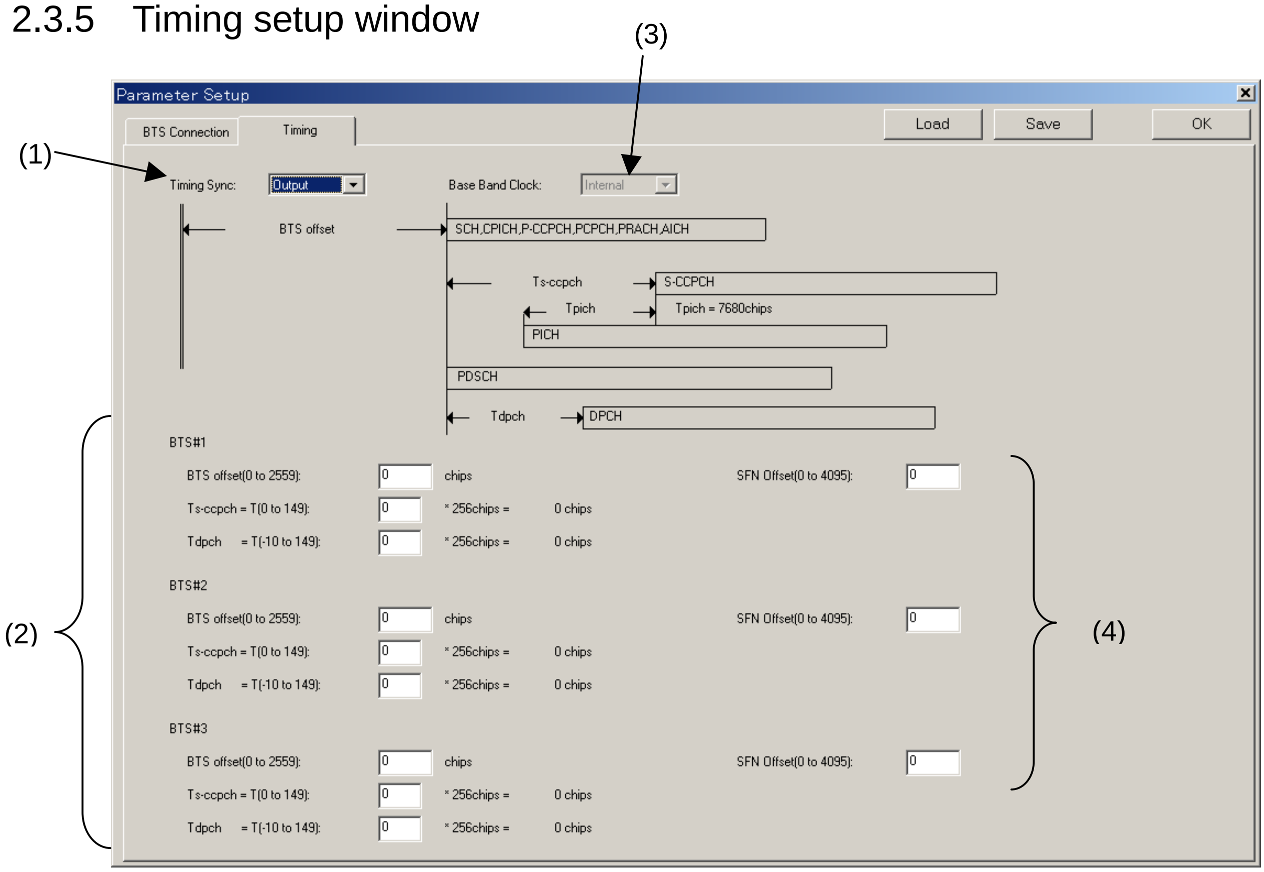


Figure 2-24　Parameter Setup window

(1)　Timing Sync

Output: Signalling Tester operates on an internal clock.　At the start of operation, a trigger pulse is output once from the SYNC OUT connector on the Timing Generator board.

Input: It is necessary to input a trigger pulse (rise pulse) in the SYNC INPUT on the Timing Generator board.　Clock can be set by (3) Baseband Clock.

(2)　BTS offset, Ts-ccpch, Tdpch

Set the timing offset of each channel and each BTS.
When setting timing offset on Timing setup window, this offset will be reflected to both of S-CCPCH and PICH.

(3)　Base Band Timing

Set the baseband timing of signalling tester. This setting is enabled when Input is selected in (1)Timing Sync.

External: Signalling Tester is operated using external clock.(This clock must be 5 times chip rate)

Internal: Signalling Tester is operated using an internal clock.

(4) SFN offset

SFN offset: Specify the SFN offset between BTSs.

Table 2-4　Example of timing setup

BTS / parameter	BTS1	BTS2	BTS3
BTS offset	0	0	0
Tdpch	0	0	0
SFN offset	2	4	8

In case of Table 2-4, the relation CFN and SFN between BTSs are as follows.

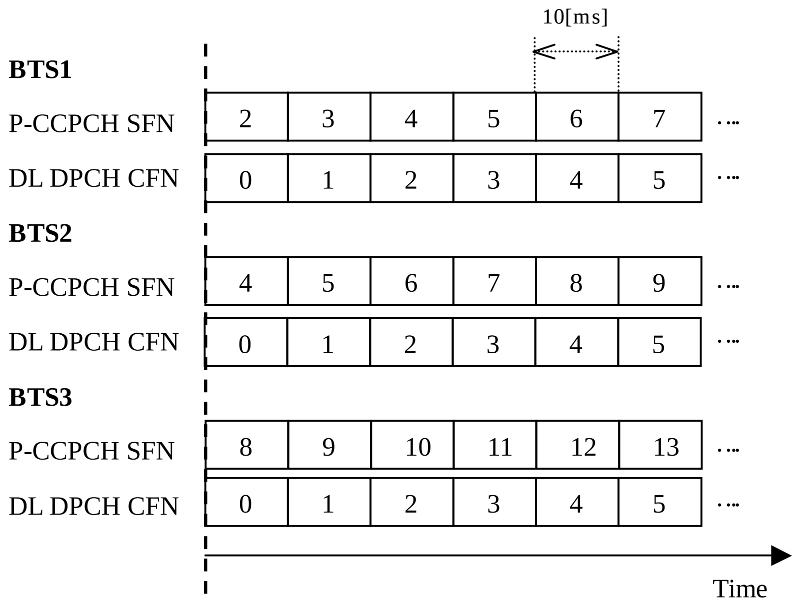


Figure 2-25　Relations of SFN and CFN

2.4 Scenario

2.4.1 About scenario

The Signalling Tester performs measurement by inter-layer communications called primitive.　The primitive can be divided into two categories.

1) Control of each layer (PHY, MAC, RLC, TE layer)

2) Transmission/reception of signalling messages and data

These primitives can be issued from the layer 3 scenario(RRC) of the Signalling Tester and executed in the order described in the scenario.　The Signalling Tester has ready for you library mathematical functions corresponding to respective primitives so that you can use the C language to describe a scenario.

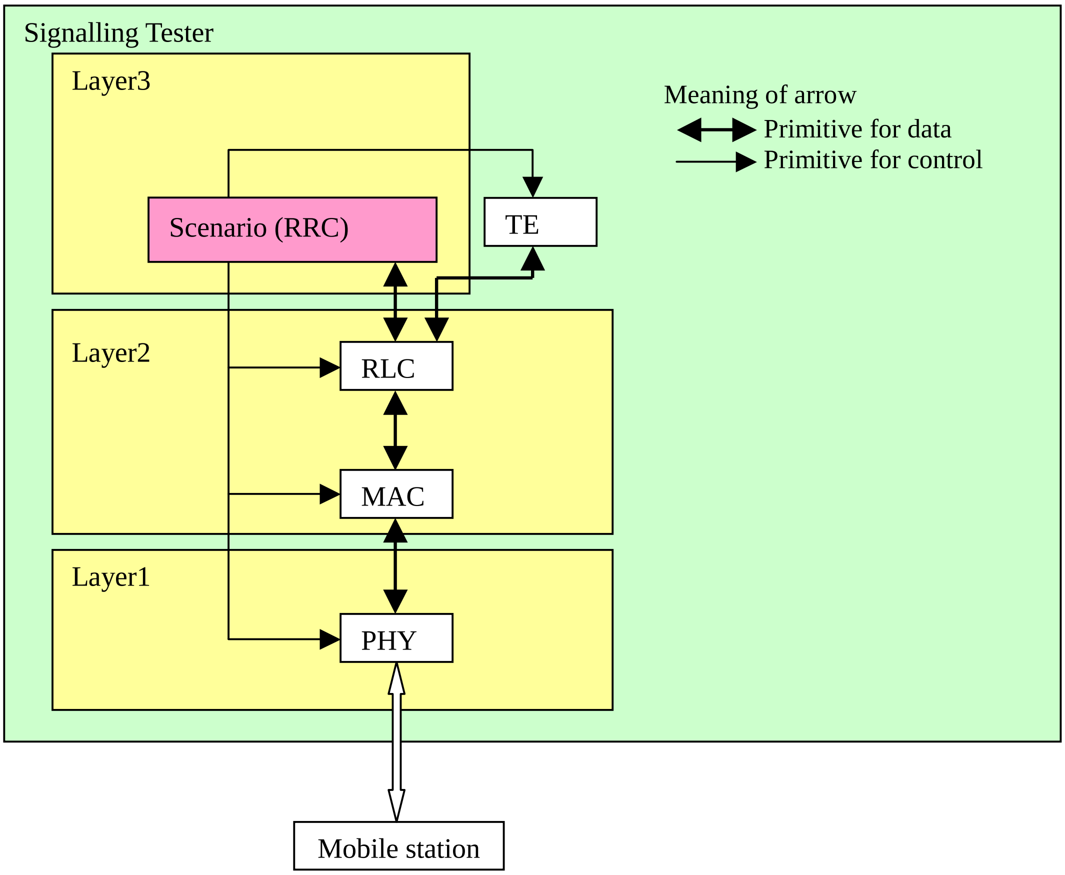


Figure2-26　Control of Signalling Tester by scenario

2.4.2 Description method

You can efficiently create a scenario by seeing a sample scenario attached to the Control Software. The sample scenario for the Harikiri (physical layer) and call processing test is in "C:MX848000/Scenario/Src."

The following shows six basic rules for describing a scenario:

Basic rule 1

When Control Software executes a scenario, it calls the ScenarioMain() function. Thus, processing to execute should be described in the ScenarioMain() function and in the ones called from the ScenarioMain() function.

Basic rule 2

At the beginning of each source file, describe:

```
#include        <windows.h>
#include        "wcdma.h"
#include        "primitive.h"
#include        "parameter.h"
#include        "scenario.h"
#include        "stdio.h"
DLLEXPORT INT ScenarioMain(LPVOID);
```

Basic rule 3

Since the SimulatorStart() function initializes the Signalling Tester, describe the SimulatorStart() function only in the scenario to be executed first when the START button of Control Software is pressed.

Basic rule 4

Before transmitting/receiving messages via a logical channel by using the SndMessage() function and RcvMessage() function, use the CphyRlSetup function, CphyTrchConfig() function, CmacConfig() function and CrlcConfig() function to set related transport channels and physical channels and provide a route (TFC setting, etc.) in advance.

Basic rule 5

When activating each channel, describe CalcRMPalameter() or CalcRMPalameterCM() before CphyRlSetup() and CphyTrchConfig().
(CalcRMParameter() calculates the parameter to need for CphyTrchConfig() action.)

Basic rule 6

Describe CmacConfig() for each CCtrCH(3GPP specification says it is for each Logical Channel.).

Precautions

Difference of primitives between Signalling Tester and 3GPP

Primitives used in scenarios of Signalling Tester do not completely match the scenarios specified in 3GPP.

Priority of setup in a scenario and in the Setup window

Some parameters such as Channelization Code can be set both in a scenario and in the Setup window of the Control Software. In case setup is made in both, setup in the scenario is given priority.

InitializeParameter.c

A channel parameter used in a scenario has an initial value set by firmware. A copy of the portion where the initial value is set is- "InitialParameter.c" file. Since the file "InitialParameter.c" is a copy and modifying the contents of this file cannot modify the initial value. In case it is necessary to modify the initial value in your scenario, describe a parameter setting portion in the scenario seeing "InitialParameter.c."

For the parameter modification method, see "Appendix D EXPLANATION OF SCENARIOS."

Reception of RACH and DPCH

Since Signalling Tester has only one Matched Filter, it cannot receive RACH and DPCH simultaneously. If RACH and DPCH are activated, channel activated later is available.

2.4.3 Scenario library

The Signalling Tester is not equipped with the library functions for scenarios shown inTable 2-5. For details of each function, see Appendix A SCENARIO LIBRARIES.

Table 2-5 Library functions for scenarios

Function name	Feature
SimulatorStart()	Execution processing
WaitTime()	Specified time wait
SequenceBtn()	Button display and press wait
SequenceStr()	Character string input
SequenceDisp()	Character string display
SndMessage()	Message transmission
RcvMessage()	Message reception
CrlcConfig ()	RLC setup
CmacConfig ()	MAC setup
CphyTrchConfig()	PHY setup
CphyTrchRelease()	PHY release
CphyRlSetup()	PHY setup
CphyRlRelease()	PHY release
CphyRlModify()	PHY modification
CphyCfnInd()	Request PHY to notify a scenario of a specified timing at the CFN
BtsPower()	Modifies transmission power of each BTS.
BtsPowerActTime()	Modifies transmission power of each BTS.
BtsAttnuator()	Sets transmission/reception attenuator value of each BTS.
BtsReadCFN()	Reads current CFN value.
BtsDownTPCBit()	Sets downlink TPC Bit.
BtsMeasure()	BLER, BTFD measurement (count)
BtsFrequency()	Modifies transmission/reception frequency.
CalcRMParameter()	Calculates Rate Matching Parameter in Normal Mode.
CalcRMParameterCM()	Calculates Rate Matching Parameter in Normal Mode and Compressed Mode.
CteConfig ()	Setting of logical channels and service type
CteRelease()	Release of logical channels and service type
CteConnect()	Connects a call to TE.
CteDisconnect()	Disconnects a call to TE.
ReplaceIE()	Overwrites a data in bit.
ExtractIE()	Fetches a data in bit.
Int2MsbIE()	Moves INT type (32bit) data to MSB.
Short2MsbIE()	Moves SHORT type data (16bit) to MSB.
Msb2IntIE()	Transforms MSB packed data to INT type (32bit).
Msb2ShortIE()	Transforms MSB packed data to SHORT type (16bit).
BtsOcnsActivate()	Starts the OCNS transmission
BtsOcnsDeactivate()	Halts the CNS transmission.
BtsOcnsPower()	Starts/ halts OCNS Power Control.
BtsOcnsPowerActTime()	Starts/ halts OCNS Power Control.
CpdcpConfig()	Configures PDCP
CpdcpRelease()	Releases PDCP
CpdcpReloc()	Acquires an uplink reception SN / downlink transmission SN

CpdcpSeqNum()	Transmits an uplink reception SN / downlink transmission SN
WcdmaRcvControl()	Receive control primitives in W-CDMA
CbmcConfig()	Setting of BMC
CbmcRelease()	Release of BMC
SndBMCMessage()	Setting of BMC message and the transmission of BMC message

2.4.4 Scenario compilation

Dragging and dropping a scenario source file (*.c) to the apple icon on the desktop executes compilation (this processing requires installation of [Visual C++ version 6.0, .net 2002, or .net 2003] in Drive C). When compilation is complete, an execution format file (*.dll) is created in the specified folder (C:/Mx848000/Scenario/).

2.4.5 Scenario execution

Clicking on the Scenario button in the Control Software brings up the scenario selection screen. On this screen, select a scenario to be used first. Then, clicking on the START button executes the selected scenario.
When execution of the scenario is complete, the scenario selection screen appears again. Specifying another scenario on the screen here starts execution of the specified scenario from the state where the previous scenario has been terminated.

2.5　Trace

2.5.1　Description of Trace Screen

This section explains how to see the Trace Control screen.

(1)　Trace screen

On the Trace screen, the results of executing a sequence described in a scenario by Signalling Tester are recorded.

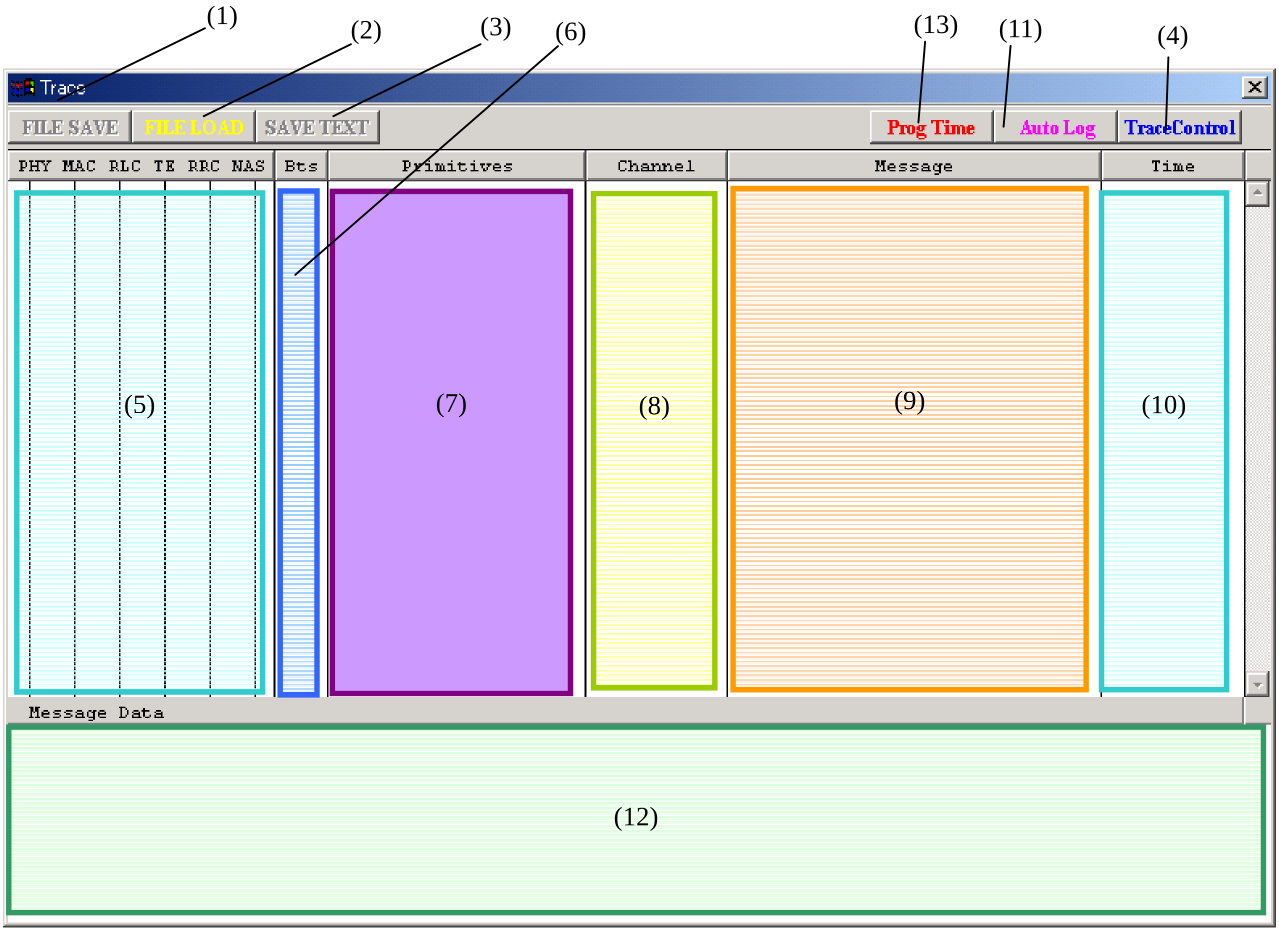


Figure 2-27　Trace screen

(1)　FILE SAVE

Stores the execution results of a scenario in a specific format (*.log).

(2)　FILE LOAD

Reads and displays a saved file (*.log).

(3)　SAVE TEXT

Saves the execution results of a scenario in the text format (*.txt).

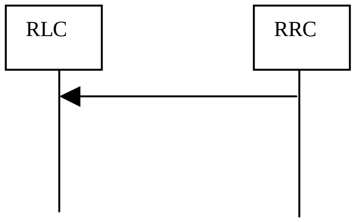
(4)　TraceControl

The Trace Control screen appears.　See section 2.5.1 (2).

(5) Arrow display

An arrow is displayed from a sender to a receiver.

Example: when the sender is RRC and the receiver is RLC



(6) Bts display

Displays the Bts no.

(7) Primitive display

Displays a primitive name.

The following shows contents displayed by primitives:

Table 2-6 contents displayed as primitives

CRLC_START_REQ CRLC_START_CNF	Used for start of RLC
CRLC_CONFIG_REQ CRLC_CONFIG_CNF	Used for setup of RLC
CRLC_BTS_MEAS_REQ	Used for BLER (RE-TRANSMISSION) measurement.
CRLC_BTS_MEAS_IND	Used for BLER (RE-TRANSMISSION) measurement (measurement result display).
RLC_AM_DATA_REQ	Used when AM mode data is sent from TE to RLC.
RLC_AM_DATA_IND	Used when AM mode data is sent from RLC to RRC.
RLC_AM_DATA_CNF	Used when AM data is sent from RLC to RRC.
RLC_UM_DATA_REQ	Used when UM mode data is sent from TE to RLC.
RLC_UM_DATA_IND	Used when UM mode data is sent from RLC to RRC.
RLC_TR_DATA_REQ	Used when TR mode data is sent from TE to RLC.
RLC_TR_DATA_IND	Used when TR data is sent from RLC to RRC.
CMAC_START_REQ CMAC_START_CNF	Used for start of MAC.
CMAC_CONFIG_REQ CMAC_CONFIG_CNF	Used for setup of MAC.
MAC_DATA_REQ	Used when data is sent from RLC to MAC.
MAC_DATA_IND	Used when data is sent from MAC RLC.
CPHY_START_REQ CPHY_START_CNF	Used for start of PHY.
CPHY_TRCH_CONFIG_REQ CPHY_TRCH_CONFIG_CNF	Used for setup of Transport Channel and TFS.
CPHY_TRCH_RELEASE_REQ CPHY_TRCH_RELEASE_CNF	Used for release of Transport Channel and TFS.
CPHY_RL_SETUP_REQ CPHY_RL_SETUP_CNF	Used for setup of Radio Link.
CPHY_RL_RELEASE_REQ CPHY_RL_RELEASE_CNF	Used for release of Radio Link.
CPHY_RL_MODIFY_REQ CPHY_RL_MODIFY_CNF	Used for modification of Radio Link.

CPHY_SYNC_IND	Used when synchronization is acquired
CPHY_OUTSYNC_IND	Used when synchronization is lost
CPHY_BTS_CONFIG_REQ CPHY_BTS_CONFIG_CNF	Used for setup of BTS
CPHY_BTS_POWER_REQ CPHY_BTS_POWER_CNF	Makes BTS-based power setup.
CPHY_BTS_ATT_REQ CPHY_BTS_ATT_CNF	Used for setup of TX Attenuator and RX Reference Power (PHY).
CPHY_BTS_MON_REQ CPHY_BTS_MON_IND CPHY_BTS_MON_CNF	Used for BTS monitor execution (PHY).
CPHY_BTS_TXTPC_REQ CPHY_BTS_TXTPC_CNF	Makes TPC setup (downlink).
CPHY_BTS_RXTPC_REQ CPHY_BTS_RXTPC_CNF	Makes TPC setup (uplink).
CPHY_BTS_MEAS_REQ CPHY_BTS_MEAS_IND	Used for measurement (count) of BLER or BTFD.
CPHY_BTS_FREQ_REQ CPHY_BTS_FREQ_CNF	Used for modification of uplink/downlink RF frequency.
PHY_DATA_REQ	Used when a data transmission request is sent from MAC to PHY.
PHY_DATA_IND	Used when data is sent from PHY to MAC.
CTE_START_REQ CTE_START_CNF	Used for activation start processing of the entire TE. (the entire TE is being initialized)
CTE_CONFIG_REQ CTE_CONFIG_CNF	Used for setup of Config parameter of TE.
CTE_RELEASE_REQ CTE_RELEASE_CNF	Used for reset of Config parameter of TE.
CTE_CONN_REQ CTE_CONN_IND CTE_CONN_CNF CTE_CONN_RESP	Used for connection of TE.
CTE_DISC_REQ CTE_DISC_IND CTE_DISC_CNF CTE_DISC_RESP	Used for disconnection of TE.
TE_Q931_REQ	Used for transmitting Q.931 message to TE (only when TE is ISDN).
TE_Q931_IND	Used for receiving Q.931 message from TE (only when TE is ISDN).
TE_DATA_REQ	Used when downlink data is sent from TE.
TE_DATA_IND	Used when uplink data is sent to TE.
CPDCP_START_REQ CPDCP_START_CNF	Used for start of PDCP.
CPDCP_CONFIG_REQ CPDCP_CONFIG_CNF	Used for setup of Config parameter of PDCP.
CPDCP_RELEASE_REQ CPDCP_RELEASE_CNF	Used to Release for setup of Config parameter of PDCP.
CPDCP_RELOC_REQ	Used for temporary stop of PDCP and demand of Sequence Number.
CPDCP_RELOC_CNF	Used for transmission of PDCP current Sequence Number to RRC.
CPDCP_SN_REQ CPDCP_SN_CNF	Used when Acquired Sequence Number is transmitted from UE with RRC Message to PDCP. Also, used when Data Re-transmission is started from PDCP.
PDCP_DATA_REQ	Used when data is sent from TE to PDCP.
CPDCP_FWD_REQ CPDCP_FWD_CNF	Used for the demand when data is sent from PDCP to SNDCP.
CPDCP_RCVCMP_IND	Used for notification to RRC that PDCP received all data, when data is sent from PDCP to SNDCP.

CPDCP_SNDCMP_IND	Used for notification to RRC that all data is transmitted, when data is sent again from PDCP.
PDCP_FWDPDU_REQ	Used for the data when data is sent from PDCP to SNDCP.
PHY_ERROR_IND	Used when CRC error data is received.(It is discarded at MAC layer)
CBMC_START_REQ CBMC_START_CNF	Used for start of BMC.
CBMC_CONFIG_REQ CBMC_CONFIG_CNF	Used for the setting of Config parameter of BMC.
CBMC_RELEASE_REQ CBMC_RELEASE_CNF	Used for stop of BMC and initializing of Config parameter of BMC.
BMC_DATA_REQ	Used for the setting and transmission of BMC messages
BMC_DATA_CNF	Used for indication from BMC to RRC, when transmission of CBS message is expired.

(8) Channel display

Displays Uplink/Downlink, Channel name (Transport Channel, Logical Channel) and Channel number.

Example: "U_DCCH 1" represents "Uplink, Channel name=DCCH, Channel number=1."

(9) Message display

Displays the type of message.

*)L3Message Usage

L3Message.def is a definition file to indicate the Layer3 message (RRC,CC,MM,GMM,SM) name in the message area of TRACE screen .

Usage:

1) Exit MX848000A or MX848040A
2) Rename L3Message.def to L3Message_Null.def.(to backup L3Message.def)
3) Rename L3Message_2001_Jun.def to L3Message.def,if you want to display the Layer3 message of 3GPP 2001 June version.

L3Message_2000_Dec.def #for 3GPP 2000 Dec version
L3Message_2001_Mar.def #for 3GPP 2001 March version
L3Message_2001_Jun.def #for 3GPP 2001 June version

Note:

Do not edit the contents of L3Message_2000_Dec.def, L3Message_2001_Mar.def, L3Message_2001_Jun.def, L3Message.def, L3Message_TDMA.def.

(10) Time display

Displays the elapsed or real time from START in units of 10 ms.

(11) Auto Log display

You can select whether to automatically save trace logs.
By this feature, trace data is stored in a file specified in the text format during operation.
The number of trace lines (current lines: 1000) is not restricted (*1) so that voluminous trace logs can be collected.

(*1) Note that you must specify the maximum file size. Once the file size exceeds the value specified here, the subsequent data cannot be written.

(12) Length display

Displays the length of the Message section in Octets.

(13) Change time display

This function changes method to display time stamp on MX848000A TRACE window. See section 3.7.19 for further information.

The MX848000A software version that is Ver. 5.21c or later is necessary for the function.

(2) Trace Control screen

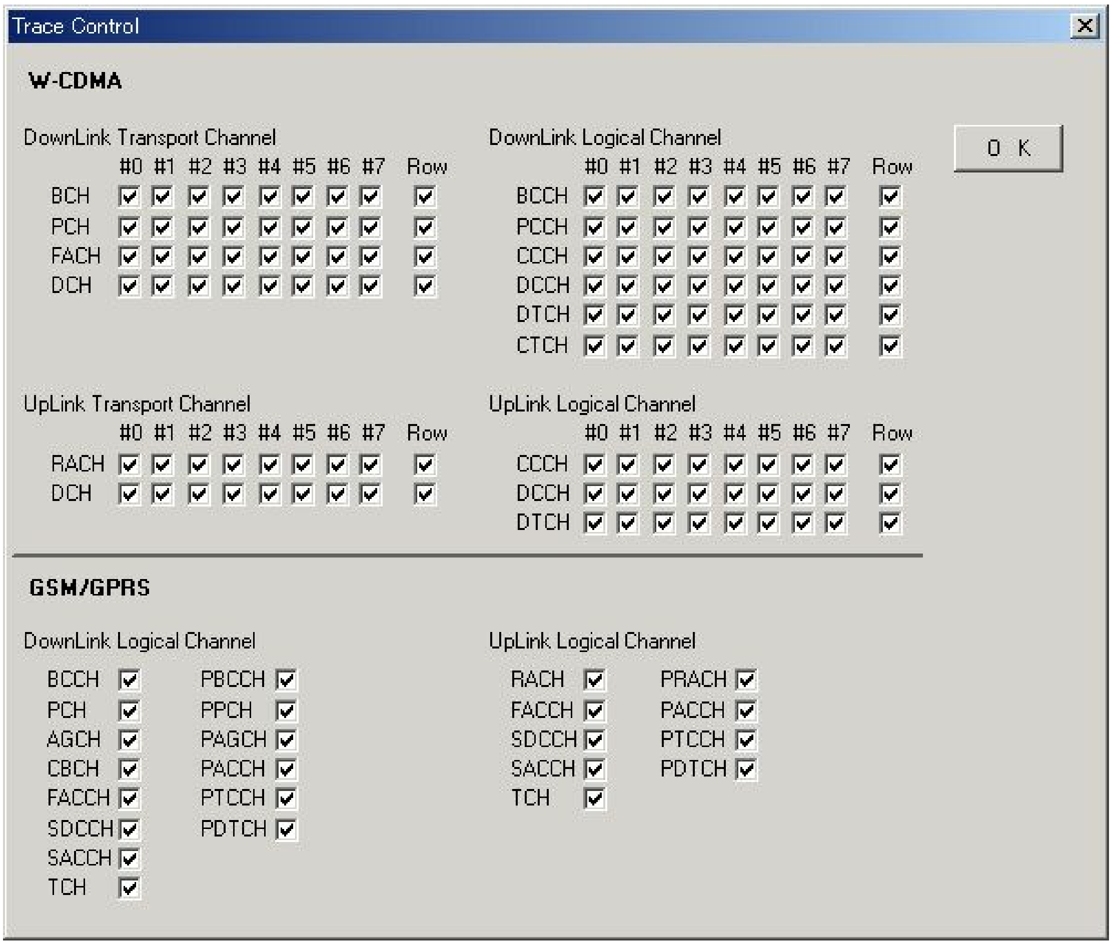


Figure 2-28 Trace Control screen

You can select ON/OFF for each item. #0 through #7 represent channel numbers set by scenarios. You can select ON/OFF for all channel numbers(#0 through #7) by checking "Row" boxes when you use PC:V5.30a or later versions. Care should be taken because setting details are saved even after Control Software is terminated.

2.5.2 Execution example

Figure 2-29 shows an example of screen obtained when a scenario is actually executed.

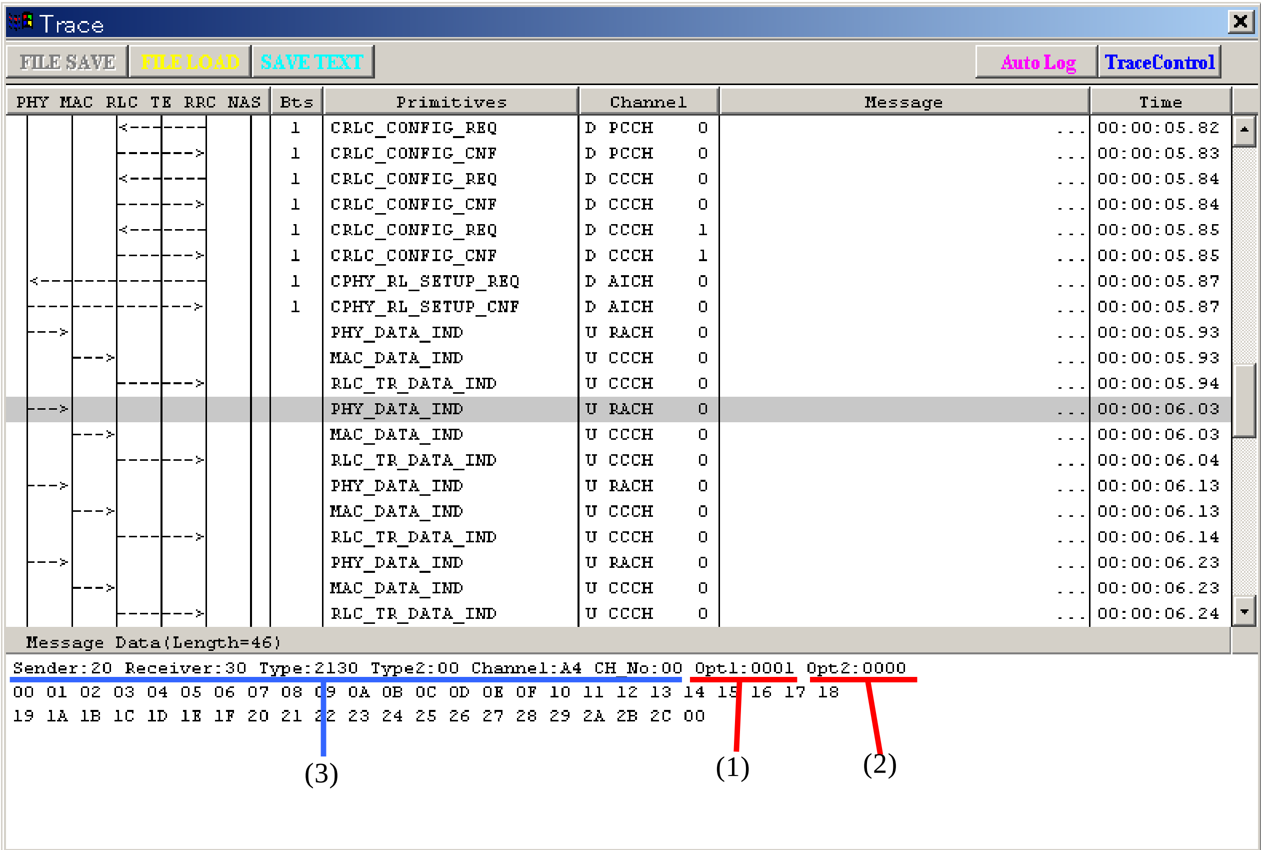


Figure 2-29 Example of execution of Trace screen

(1) TFI value is displayed (only PHY_DATA_IND is meaningful).

(2) CRC operation results are displayed in hexadecimals and the values have the following meanings (only PHY_DATA_IND is meaningful):

0000: CRC OK

Other than 0000: CRC NG

When CRC=NG, information shown in Table 2-7 can be obtained by converting Opt 2 to a binary number.

Table 2-7 Information obtained when CRC is NG

Bit15(MSB)	Bit14	• • •	Bit2	Bit1	Bit0(LSB)
0:All CRC OK 1:CRC NG exists	CRC result for TB#14	• • •	CRC result for TB#2	CRC result for TB#1	CRC result for TB#0

(3) Sender, Receiver, Type, Type 2, Unit, Channel, CH_No.: Numerical representation of arrow information on the Trace screen. (Our debug feature)

Note:
Although Two arrows of CPHY_TRCH_CONFIG_REQ in the direction from RRC to PHY will be displayed on the trace screen when CphyTrchConfig( ) to activate Uplink Channel (Refer to 2.4.3 Scenario libraries.) is executed, there would be any operational problems.

2.6 Monitor

This section describes the monitor feature attached to the Signalling Tester. The Signalling Tester can obtain monitor information shown in Table 2-8 during measurement. Such monitor information can be used to check operation of the Signalling Tester and development of a mobile station.

Table 2-8 Monitor item list

Major classification	Minor classification	Monitor information	Monitor method
Hardware/Software of Signalling Tester	–	Is software installation successful? No hardware fault?	LED
		Version information	SYSTEM screen (Information Read)
Layer1	Downlink	Transmission channel power	MONITOR screen
		Frame transmit timing	TX Baseband connector
		Channelization Code	TX Baseband connector
		Downlink TPC bit during uplink automatic power control	TX Baseband connector
	Uplink	Received total power	Rear connector, MONITOR screen
		Received channel power	MONITOR screen
		Received timing error	MONITOR screen
		Received frequency error	MONITOR screen
		Decoding result of received TFCI	LED, RX Baseband connector
		Pilot reception status	LED, RX Baseband connector
		Preamble reception result	LED, RX Baseband connector
		TPC bit reception result	RX Baseband connector
		FBI reception result	RX Baseband connector
		Channelization code used for reception	RX Baseband connector
Layer2(MAC,RLC)	–	Primitive message	Trace screen
Layer3(RRC)	–	Primitive message	Trace screen

2.6.1 Monitor screen

Information obtained on the Monitor screen is shown in Figure 2-30. Accuracy of display is as shown below:

Timing Error**:**	±0.6Chip(DPCH),± 1.0Chip(RACH) **This value is a post-calibration value.** The value is a measurement accuracy measured based on the measurement method in the "Operation Manual Appendix J."

Other: Not guaranteed. Use as a guideline.

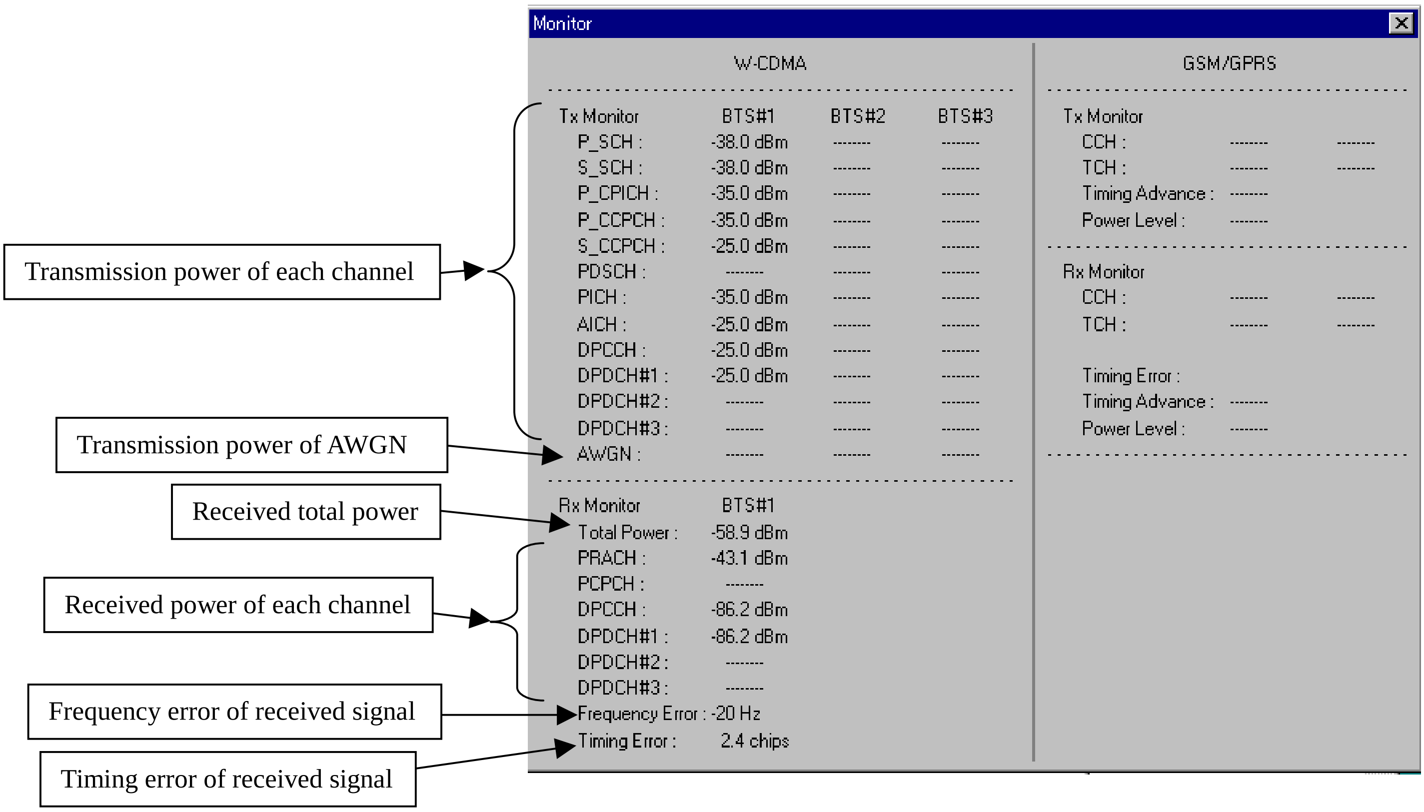


Figure 2-30 Monitor screen

2.6.2　Main unit front LEDs

Table 2-9 shows the initial state of the main unit front LEDs after power ON and conditions for lighting and blinking of each LED.

Table 2-9　Definition of each LED (diagonals mean undefined)

Board Name	LED name	Number	After power ON	Meaning of lighting or blinking
Timing Generator /2	Monitor A	1	Off	
		2	On(Red)	Power
	Monitor B	1	Off	
		2	On(Green)	
TX Baseband	Monitor A	1	Off	On with P-CCPCH ACT
		2	On(Red)	Power
	TX Baseband	1	Off	On with S-CCPCH ACT
		2	Off	On with DPCH ACT
Frame Coder	Monitor A	1	On(Red)	Power
		2	Off	On while Transport channel #1 data is being sent.
		3	Off	On while Transport channel #2 data is being sent.
		4	Off	On while Transport channel #3 data is being sent.
		5	Off	On while Transport channel #4 data is being sent.
		6	Off	On while Transport channel #5, #6 data is being sent.
	Monitor B	1	On(Red)	Power
		2	Off	On while BCH is being sent.
		3	Off	On while PCH is being sent.
		4	Off	On while FACH (CCCH) is being sent.
		5	Off	On while FACH (DCCH) is being sent.
		6	Off	
ISDN	BRI	1	Off	On when synchronization of ISDN physical layer is established.
		2	Off	On when ISDN data link is established.
	PRI	1,2	Blinking (500 ms interval)	
Voice Codec	Monitor	1	On(Red)	Power
		2	Off	On when transmission/reception of data using the User-data connector or voice data is started.
Frame Decoder	MonitorA	1	On(Red)	Power
		2	Off	On when CRC of Transport Channel #1 is OK.
		3	Off	On when CRC of Transport Channel #2 is OK.
		4	Off	On when CRC of Transport Channel #3 is OK.
		5	Off	On when CRC of Transport Channel #4 is OK.
		6	Off	On when CRC of Transport Channel #5, #6 is OK.
	Monitor B	1	On(Red)	Power
		2-5	Off	
RX Baseband	Monitor	1	Off	DPCH Pilot: On when Pilot is OK for eight slots running.

		2	On(Red)	Power
	RX Baseband1	1	Off	Preamble reception
		2	Off	RACH Pilot: On when Pilot is OK for all the slots.
CPU	Monitor	1	Off	
		2	On(Red)	Power
		3	Off	Blinks in operation.
		4	On(Green)	Blinks in operation.
		5	Off	
		6	Off	
	10/100BaseT	1	Off	On with LINK.
		2	Off	On with ACTIVITY.
	Control PC	1	Off	On with LINK.
		2	Off	On with ACTIVITY.
TDMA	CPU Monitor	1	Off	
		2	On(Red)	Power
		3	Off	
		4	Off	Blinks while operating
		5	Off	
		6	Off	Blinks while operating
	Tx Monitor	1	Off	Blinks when CCH is transmitted.
		2	Off	Blinks when TCH is transmitted.
		3	Off	Blinks when PTCH is transmitted.
		4	Off	Blinks when BCCH Frame is transmitted.
		5	Off	Goes on when CCH is activated.
		6	Off	Goes on when TCH is activated.
	Rx Monitor	1	Off	Blinks when CCH is normal received.
		2	Off	Blinks when TCH is normal received.
		3	Off	Blinks when PTCH is normal received.
		4	Off	
		5	Off	Blinks when RACH is detected.
		6	Off	Blinks when TSC is detected.

2.6.3　Information obtained from Monitor connector

Table 2-10 shows the information obtained from the monitor connector of the Signalling Tester main unit.

Table 2-10　Monitor Connector of the Signalling Tester main unit

Connector name	Description	I/O	Remarks
Detector Output	Receive total power can be monitored. When a value the same as Reference Power is received, the result is 2.335 V which varies 35 mV per dB.　Note that accuracy is not guaranteed.　Use this data as a guideline.	O	BNC connector

Table 2-11 shows information obtained from the connector on the TX Baseband board.

Table 2-11　TX Baseband connector (half-pitch 80-pin)

Pin No.	Description	I/O
1	Ground	O
2	5-time chip timing (19.2 MHz)	O
3	Ground	O
4	Chip clock (3.84 MHz)	O
5	Ground	O
6	P-CCPCH frame timing (10 msec)	O
7	CPICH frame timing (10 msec)	O
8	PICH frame timing(10 msec)	O
9	AICH frame timing (10 msec)	O
10	S-CCPCH frame timing (10 msec)	O
11	DPCH frame timing (10 msec)	O
12	Frame timing (80 ms) Output 7680 chips before the boundary of 80 ms TTI	O
13	Ground	O
14	Scrambling Code1 I	O
15	Scrambling Code1 Q	O
16	Scrambling Code2 I	O
17	Scrambling Code2 Q	O
18	Synchronization Code 1 (P-SCH)	O
19	Synchronization Code 2 (S-SCH)	O
20	P-CCPCH Channelization Code	O
21	CPICH Channelization Code	O
22	PICH Channelization Code	O
23	AICH Channelization Code	O
24	S-CCPCH Channelization Code	O
25	DPCH Channelization Code1	O
26	DPCH Channelization Code2	O
27	DPCH Channelization Code3	O
28	Ground	O
29	Reserved	O
30	Reserved	O

31	Reserved	O
32	Reserved	O
33	Ground	O
34	Non-Connection	
35	Non-Connection	
36	Ground	O
37	Ground	O
38	Ground	O
39	Digital baseband signal : Downlink I(0)	O
40	Digital baseband signal : Downlink I(1)	O
41	Digital baseband signal : Downlink I(2)	O
42	Ground	O
43	Digital baseband signal : Downlink I(3)	O
44	Digital baseband signal : Downlink I(4)	O
45	Digital baseband signal : Downlink I(5)	O
46	Digital baseband signal : Downlink I(6)	O
47	Digital baseband signal : Downlink I(7)	O
48	Digital baseband signal : Downlink I(8)	O
49	Digital baseband signal : Downlink I(9)	O
50	Digital baseband signal : Downlink I(10)	O
51	Digital baseband signal : Downlink I(11)	O
52	Digital baseband signal : Downlink I(12)	O
53	Digital baseband signal : Downlink I(13)	O
54	Digital baseband signal : Downlink I(14)	O
55	Digital baseband signal : Downlink I(15)	O
56	Non-Connection	
57	Non-Connection	
58	Ground	O
59	Ground	O
60	Digital baseband signal : Downlink Q(0)	O
61	Digital baseband signal : Downlink Q(1)	O
62	Ground	O
63	Digital baseband signal : Downlink Q(2)	O
64	Digital baseband signal : Downlink Q(3)	O
65	Digital baseband signal : Downlink Q(4)	O
66	Digital baseband signal : Downlink Q(5)	O
67	Digital baseband signal : Downlink Q(6)	O
68	Digital baseband signal : Downlink Q(7)	O
69	Digital baseband signal : Downlink Q(8)	O
70	Digital baseband signal : Downlink Q(9)	O
71	Digital baseband signal : Downlink Q(10)	O
72	Digital baseband signal : Downlink Q(11)	O
73	Digital baseband signal : Downlink Q(12)	O
74	Digital baseband signal : Downlink Q(13)	O
75	Digital baseband signal : Downlink Q(14)	O
76	Digital baseband signal : Downlink Q(15)	O
77	Ground	O
78	Non-Connection	
79	Non-Connection	
80	Non-Connection	

For details of timing of Frame Timing, Scrambling Code and Channelization Code output from the TX baseband connector, see Appendix C of the Operation Manual.

Table 2-12 shows information obtained from the connector on the RX Baseband board.

Table 2-12　RX Baseband connector (half-pitch 80-pin)

Pin No.	Description	I/O
1	Ground	O
2	5-time chip timing (19.2 MHz)	O
3	Ground	O
4	Chip clock (3.84 MHz)	O
5	Ground	O
6	Access Slot timing (1.33 msec)	O
7	DPCH frame timing (10 msec)	O
8	Reserved	O
9	Ground	O
10	RACH/CPCH Scrambling Code I	O
11	RACH/CPCH Scrambling Code Q	O
12	DPCH Scrambling Code I	O
13	DPCH Scrambling Code Q	O
14	Preamble Scrambling Code	O
15	RACH/CPCH Control Part Channelization Code	O
16	RACH/CPCH Data Part Channelization Code	O
17	DPCCH Channelization Code	O
18	DPDCH#1 Channelization Code	O
19	DPDCH#2 Channelization Code	O
20	DPDCH#3 Channelization Code	O
21	Reserved	O
22	Outputs a pulse when the timing of de-spreading is shifted, synchronized with the delay of transmission timing of MS.	O
23	Outputs a pulse when the timing of de-spreading is shifted, synchronized with the gain of transmission timing of MS.	O
24	Downlink TPC generated based on the uplink DPCCH power	O
25	Outputs "H" when DPCH Pilot is OK and "L" when NG.	O
26	TFCI of DPCH (serial data)	O
27	TPC and FBI of DPCH (serial data)	O
28	Outputs Signature No. when a preamble is received.	O
29	Outputs "H" when RACH Pilot is OK and "L" when NG.	O
30	TFCI of RACH (serial data)	O
31	Clock common to serial data	O
32	Reserved	O
33	Ground	O
34	Ground	O
35	Non-Connection	
36	Non-Connection	
37	Non-Connection	
38	Non-Connection	
39	Non-Connection	
40	Digital baseband signal : Uplink I(0)	I
41	Digital baseband signal : Uplink I(1)	I
42	Ground	O
43	Digital baseband signal : Uplink I(2)	I
44	Digital baseband signal : Uplink I(3)	I
45	Digital baseband signal : Uplink I(4)	I
46	Digital baseband signal : Uplink I(5)	I
47	Digital baseband signal : Uplink I(6)	I
48	Digital baseband signal : Uplink I(7)	I
49	Digital baseband signal : Uplink I(8)	I
50	Digital baseband signal : Uplink I(9)	I
51	Digital baseband signal : Uplink I(10)	I
52	Digital baseband signal : Uplink I(11)	I
53	Non-Connection	
54	Non-Connection	
55	Non-Connection	
56	Non-Connection	

57	Digital baseband signal : Uplink Q(0)	I
58	Digital baseband signal : Uplink Q(1)	I
59	Digital baseband signal : Uplink Q(2)	I
60	Digital baseband signal : Uplink Q(3)	I
61	Digital baseband signal : Uplink Q(4)	I
62	Ground	O
63	Digital baseband signal : Uplink Q(5)	I
64	Digital baseband signal : Uplink Q(6)	I
65	Digital baseband signal : Uplink Q(7)	I
66	Digital baseband signal : Uplink Q(8)	I
67	Digital baseband signal : Uplink Q(9)	I
68	Digital baseband signal : Uplink Q(10)	I
69	Digital baseband signal : Uplink Q(11)	I
70	Non-Connection	
71	Non-Connection	
72	Ground	O
73	Ground	O
74	Non-Connection	
75	Non-Connection	
76	Non-Connection	
77	Non-Connection	
78	Non-Connection	
79	Non-Connection	
80	Non-Connection	

For details of timing of Frame Timing, Scrambling Code and Channelization Code output from the RX baseband connector, see Appendix D of the Operation Manual.

For details of Pins 24 to 31 of the RX Baseband connector, see Appendix B 5.3 of the Easy-to-understand the Signalling Tester.

3. APPLICATION TESTS

3.1 Harikiri (Physical Layer) Test

This section outlines the method for the Harikiri (physical layer) test. For details, see Appendix B.

3.1.1 About Harikiri (physical layer) test

The Harikiri (physical layer) test is a modulation/demodulation feature test of W-CDMA mobile station that uses only layer 1. To develop a mobile station, establish a connection and conduct a Harikiri (physical layer) test as shown in Figure 3-1. Figure 3-2 is a conceptional diagram of the Harikiri (physical layer) test.

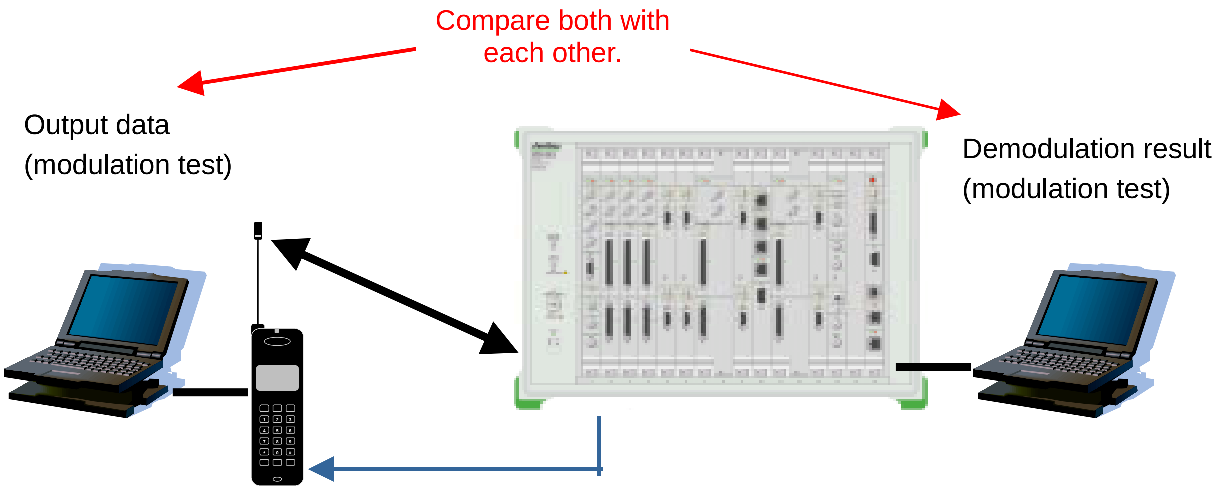


Figure 3-1 Configuration of Harikiri (physical layer) test

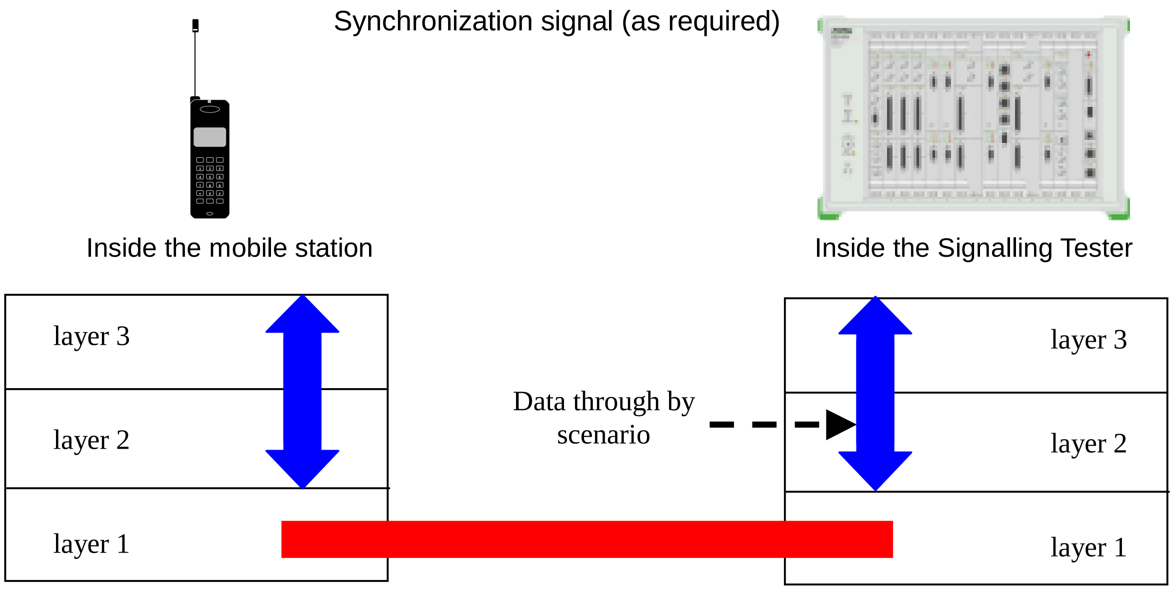


Figure 3-2 Concept of Harikiri (physical layer) test

3.1.2 Scenario to be Used

To conduct a Harikiri (physical layer) test, use Start_xxx.dll and xxx_Data(xxx).dll (C:¥MX848000¥Scenario¥Src) attached to the Control Software. For details of the scenario for the Harikiri (physical layer) test, see Appendix B.

3.1.3 Method for Harikiri (physical layer) Test

Modulation Test

The modulation test is enabled by outputting fixed data and PN9 data from the modulated part of a W-CDMA mobile station and comparing the output with the demodulation results obtained on the Trace screen of the Signalling Tester. In this test, Measurement of BLER or BER (BER test requires an external BER counter) is also available. There is a check method for investigating what part of demodulation of Signalling Tester does not work.(Refer to Section B.4)

Demodulation Test

The demodulation test is enabled by outputting fixed data and PN9 data from the Signalling Tester and comparing the output with the demodulation results on the mobile station side.

3.2 Call Processing Protocol Test

This section describes the call processing protocol test to be conducted using the Signalling Tester.

3.2.1 Method for call processing protocol test

Call processing protocol procedure for 3GPP is described according to the rule of scenario creation described in section 2.4. Using sample scenarios(C:¥Mx848000¥Scenario¥SRC¥Call_Sample) as a guideline allows efficient creation of scenarios.

3.2.2 Sample Scenarios

Sample scenarios are provided for basic call processing procedures including origination, termination and call disconnection. The following restrictions apply to the sample scenarios. Figure 3-3 shows a sample scenarios for the call processing protocol test.

Information element

Since the information element is vacant in the original scenarios, it must be embedded therein by yourself. Even when information is embedded, operation is not guaranteed because connection to a mobile station is not yet checked in Anritsu Corp.

Scheduling the system information

Scheduling the system information is made in Idle.c.

Unlike ordinary data transmission, set SIB_POS and SIB_REP before the SndMessage function. Besides, division necessary in case SEG_COUNTb is more than 2 (see System Information Block type 2 in Idle.c) has not been provided. Describe divided data. Use SIB_OFF in order to calculate SIB_POS. (Parameter SIB_OFF is not installed.)

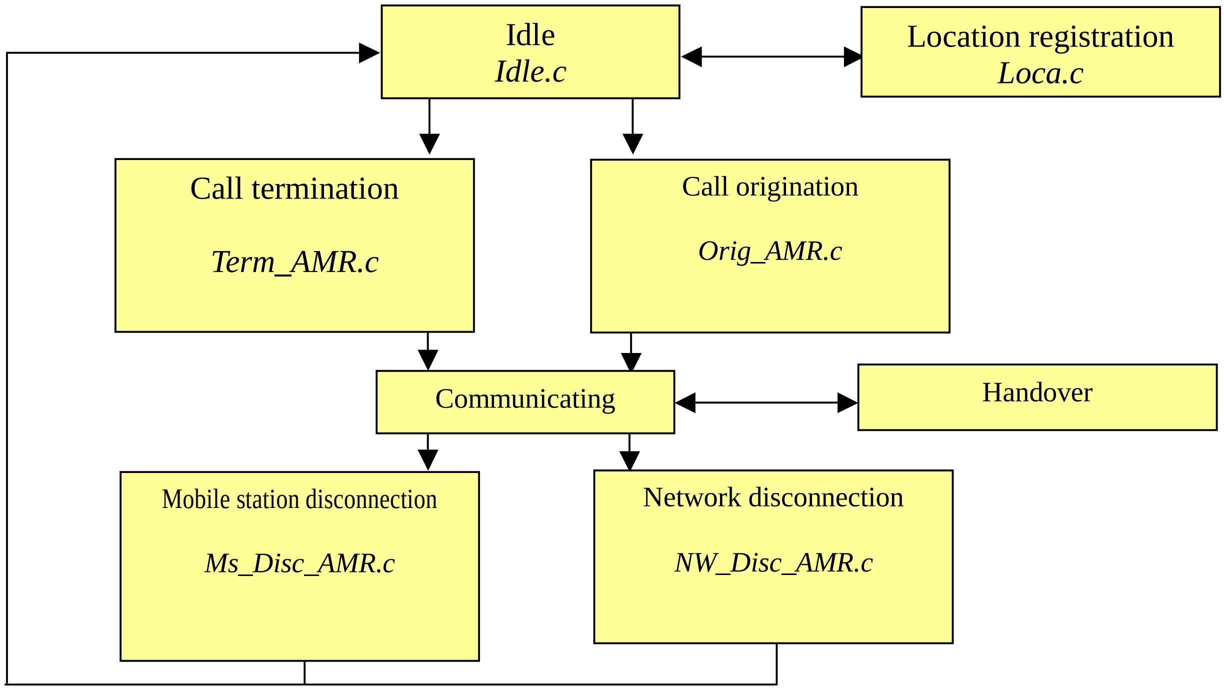


Figure 3-3 Call processing sample scenarios

3.3 TE Connection Test

This section summarizes the TE connection tests.　For details of the test method, see Appendix C.　Tests introduced in this section are as follows:

(1) ISDN test

(2) AMR Voice test

(3) Userdata test

(4) IP Packet test

(5) PPP (Serial connection)Test

(6) PPP (built-in server test)

(7) MS-to-MS test

3.3.1 ISDN test

Image/voice test can be conducted between a mobile station and the Signalling Tester by connecting the Signalling Tester to a video telephone, etc. For details of the test method, see Appendix C.

Caution:

ISDN options are necessary.
(For details of the ISDN options, see the separate Operation Manual.)

3.3.2 AMR Voice test

Voice communication test can be conducted between a mobile station and the Signalling Tester by connecting the Signalling Tester to a handset(An accessory of Signalling Tester). For details of the test method, see Appendix C DETAILS OF TE CONNECTION TEST.

3.3.3 Userdata test

You can insert arbitrary data into DTCH to be sent or fetch data from the demodulated DTCH. This test is available for measurement of error rate, etc.
For details of the test method, see Appendix C DETAILS OF TE CONNECTION TEST.

3.3.4 IP Packet test

IP protocol data communication test can be conducted by connecting the Signalling Tester to a PC (Ethernet cross cable connection).
For details of the test method, see Appendix C DETAILS OF TE CONNECTION TEST.
Figure 3-4 shows a conceptional diagram of the IP Packet test.

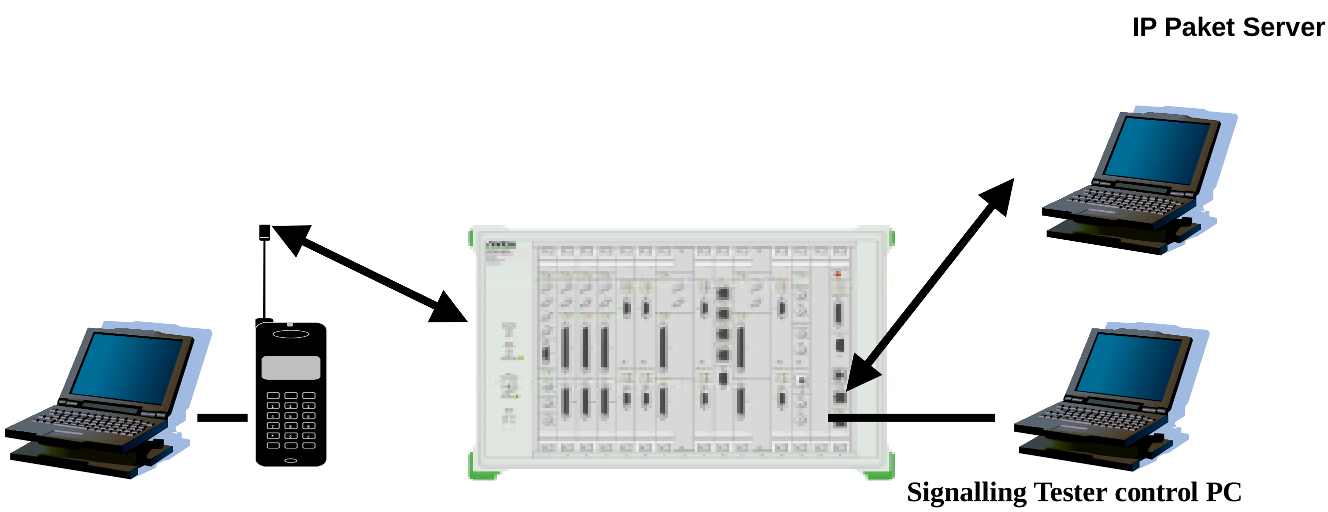


Figure 3-4 Conceptional diagram of IP Packet test

3.3.5 PPP(Serial connection) test

PPP protocol data communication test can be conducted by connecting the Signalling Tester to a PC (RS-232C serial connection—ISDN board option).
PPP is a protocol frequently used in dial-up connection to the internet.
For details of the test method, see Appendix C.

Caution:
Since the test configuration uses serial connection, communications upper 64 kbps is not allowed.
ISDN options are necessary. (For details of the ISDN options, see the separate Operation Manual.)

3.3.6 PPP (Built-in server) test

Differing from the PPP(Serial connection),the PPP protocol stack is installed in the Signalling Tester itself.
This test features termination of PPP itself in the Signalling Tester. IP level communications are made with a WindowsNT-installed PC by the Ethernet.
Since the test configuration uses the Ethernet, high-speed communications of 384 kbps is allowed.
For details of the test method, see Appendix C.

3.3.7 MS-to-MS test

MS-to-MS test using two mobile stations can be conducted by connecting two Signalling Testers by Ethernet(cross cable) connection.
For details of the test method, see Appendix C.
Caution: Two Signalling Testers are necessary.
Figure 3-5 shows the conceptional diagram of the MS-to-MS test.

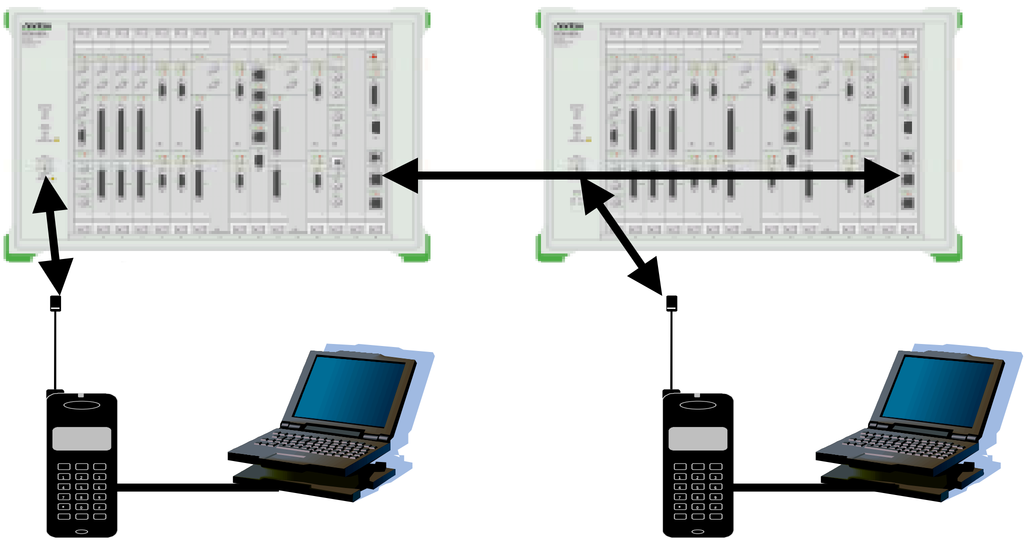


Figure 3-5 Conceptional diagram of MS-to-MS test

3.4 Tx Diversity

Radio waves used in mobile communications suffers from influence of obstacles and reflective objects such as buildings, trees, and topographical ups and downs, thus causing reflection, diffraction or scattering. As a result, a number of radio waves that followed various paths interfere with each other and the intensity of the radio wave varies to a great extent. This phenomenon is called fading.

The diversity technology is available to improve the situation where the level of the received radio wave varies due to fading, thereby assuring high-quality and high-reliability transmissions.

Tx Diversity is a transmission method to heighten receiving sensitivity on the receiving side by transmitting signals from two antennas placed spatially apart from each other.

For details of the Tx Diversity, see TS 25.211-330.

Figure 3-6 shows the conceptional diagram of Tx Diversity

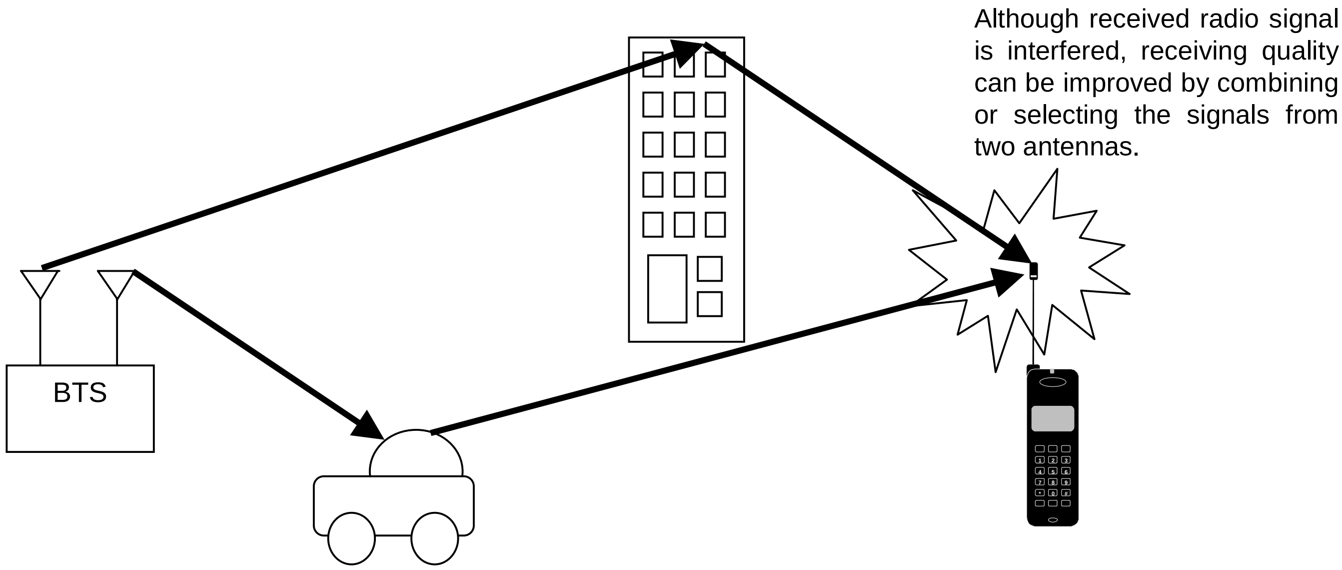


Figure 3-6 Conceptional diagram of Tx Diversity

3.4.1 Setup for Tx Diversity

The configuration necessary for implementing the TX Diversity feature is described below. This configuration comprises necessary software and hardware as well as the standard configuration.

- **Tx Diversity (1RF output) feature**

Tx Diversity is employed in a base station equipped with two antennas placed spatially apart from each other. The problem is that the Signalling Tester use cable connection and has only one RF output. Thus the Signalling Tester cannot have two antennas placed spatially apart. This problem is solved, by shifting the phases of antenna1 and antenna2 from each other in order to simulate that each of the two antennas are placed at spatially different positions.

MU848057A (Frame Coder)×2
MU848058A(TX Baseband)×2
MX848001A-01(W-CDMA Signalling Tester Tx Diversity)

- **Tx Diversity (2RF output) feature**

An additional RF output connector can be introduced by installing an optional RF expansion unit. In this case, two antenna outputs can be provided in separate systems. For the additional RF unit, see the "Operation Manual Appendix O."

MU848053A(RX Baseband) ×2
MU848057A (Frame Coder) ×2
MU848058A(TX Baseband) ×2
MD8480A-02 or MD8480B-02 (Additional RF unit)
MX848001A-01(W-CDMA Signalling Tester Tx Diversity)

- **Option Key**

Use the software for TX Diversity with writing the optional key code.For details of writing method, see “Download Manual”.

3.4.2　Parameter setup

Use the SETUP screen of the Control Software to load "default.pm1" then set the following:

Table 3-1　Parameter setup items

	Setup item	Description
(1)	"Base Band and RF Unit Connection" TX Diversity1=1 RF output. TX Diversity2=2 RF output (Refer to section 4.3.)	"TX Diversity 1" or "TX Diversity 2"
(2)	(On BTS#1 Tx Setup screen) Scrambling code of Common Channel and Dedicated Channel	Arbitrary value within the specification range
(3)	(On BTS#1 Tx Setup screen) Channelization Code and Slot Format of each PhCH of antenna1	Arbitrary value within the specification range
(4)	(On BTS#1 Tx Setup screen) Power of each PhCH of antenna1	Arbitrary value within the specification range
(5)	(On BTS#2 Tx Setup screen) Scrambling code of Common Channel and Dedicated Channel	Same value as (2)
(6)	(On BTS#2 Tx Setup screen) Channelization Code and Slot Format of each PhCH of antenna2	Same value as (3)
(7)	(On BTS#2 Tx Setup screen) Power of each PhCH of antenna2	Arbitrary value within the specification range

Note:

Set all parameters of antenna1 using the BTS#1 Tx Setup screen.
Set all parameters of antenna2 using the BTS#2 Tx Setup screen.

✳ Difference between TX Diversity1 and TX Diversity2

(1) Tx Diversity1
Please select “TX Diversity1” when you use MD8480A/B which has not been installed the Additional RF unit(MD8480A-02 or MD8480B-02). TX Diversity function will be realized with 1 antenna.

(2) Tx Diversity2
Please select “TX Diversity2” when you use MD8480A/B which has been installed the Additional RF unit(MD8480A-02 or MD8480B-02). TX Diversity function will be realized with 2 antennas.

3.4.3　Scenario setup

<Selecting scenario>

Select a scenario to use according to the transmission channel.
Scenarios for Tx Diversity is stored in the following directory after decompression of MX84800A_xxx.EXE:

MX848000¥Scenario¥src¥TX_Diversity¥

Channels used for transmission for each scenario are listed below (○: Channel that allows TX Diversity transmission for the scenario.　For transmission details of each channel, see <Post-"START" operation of antenna2> under 3.4.4.　Various operations.)

Table 3-2　List of transmittable channels

Transmission channel / Scenario names	P-SCH	S-SCH	CPICH✳1	P-CCPCH	S-CCPCH	AICH	PICH	DPCH	Compressed Mode Method	Gap pattern
Start_SCCPCH (PCH+FACH) _TxDiv.c	○	○	○	○	○ (PCH +FACH)	○	○	–	–	–
Start_P128K_TxDiv.c	○	○	○	○	–	–	–	○ ✳2	SF/2	Pattern 1

✳1　Symbol for Tx Diversity will be transmitted from anntena2 in Tx Diversity transmission.
✳2　Transmittable DPCH is Packet 128[kbps].

<Scenario setup method>

Since the Signalling Tester simulates transmission from two antennas placed spatially apart from each other, two signals with phases shifted are sent from antenna1 and antenna2 respectively.　Shift of phase between two signals is to be specified in the scenario .

Set the value of "#define TXDIV_DELTAPHASE" in the Start scenario file to a target shift angle (-180 to +180 degrees) and compile then link the value. (Initial setting is 150 degrees. For Firmware v3.20 and later versions, there is no need to describe BtsConfigParBTS3.DeltaPhase. It will be ignored even described.

3.4.4 Various operations

<Post-"START" operation of antenna2>

(1) Pressing the START button, Signalling Tester automatically transmits P/S-SCH and P-CCPCH of Tx Diversity (Open loop mode) and CPICH symbol for Tx Diversity. From antenna2, signals are sent with the phase shifted as specified in the scenario.

(2) Select "ACTIVATE" in the (service contents/channel)ACTIVATE?." At this point in time, S-CCPCH and PICH/AICH are sent by TX Diversity transmission (STTD).

(3) For DPCH, select TX Diversity mode in "DPCH Diversity mode?"

"OL STTD" Open loop
"CL mode1" Closed loop mode1
"CL mode2" Closed loop mode2

Transmission details of each physical channel is as follows:

Table 3-3 List of transmission details

<table>
<tr><th>channel</th><th>antenna1</th><th colspan="3">antenna2</th></tr>
<tr><td>P-SCH</td><td>TSTD</td><td colspan="3">TSTD</td></tr>
<tr><td>S-SCH</td><td>TSTD</td><td colspan="3">TSTD</td></tr>
<tr><td>CPICH</td><td>Ordinary transmission</td><td colspan="3">(Symbol for TX Diversity)</td></tr>
<tr><td>P-CCPCH</td><td>Ordinary transmission</td><td colspan="3">STTD</td></tr>
<tr><td>S-CCPCH *1</td><td>Ordinary transmission</td><td colspan="3">STTD</td></tr>
<tr><td>PICH</td><td>Ordinary transmission</td><td colspan="3">STTD</td></tr>
<tr><td>AICH</td><td>Ordinary transmission</td><td colspan="3">STTD</td></tr>
<tr><td rowspan="2">DPCH *2</td><td rowspan="2">Ordinary transmission</td><td>"OL STTD"</td><td>"CL mode1"</td><td>"CL mode1"</td></tr>
<tr><td>STTD</td><td>Closed loop mode1</td><td>Closed loop mode2</td></tr>
</table>

*1 Common to all of (PCH), (FACH) and (PCH+FACH)
*2 Common to all of (Stand-Alone DCCH), (UDI+DCCH) and (AMR+DCCH)

<Each Parameter in Gap patterns>

The following gap patterns are specified in Start_P128K_CM1_TxDiv.c. It allows it to transmit or receive data including Transmission Gap according to gap patterns.

Parameter	Pattern 1
TGSN	11
TGL1	7
TGL2	-
TGD	0
TGPL1	2
TGPL2	-
TGPRC	∞
TGCFN	0
UL method	SF/2
DL method	SF/2
Scrambling code change	No
Frame Type	A

<Operation common to Tx Diversity>

A signal with its phase shifted by "TXDIV_DELTAPHASE (Unit: Degrees) from antenna1 is sent from antenna2.

* "TXDIV_DELTAPHASE" is what the gap of phase rotation by propagation environment is simulated. This value is different from a shift in the phase of antenna2 in closed loop (initial value: 180 degrees).

The default value of the phase difference is shown in table 3-4.

Channel	mode	Phase difference(Unit: degree)
Except DPCH		Phase of Each PhCHantenna1 + TXDIV_DELTAPHASE
DPCH	Open Loop(STTD)	Phase of Each DPCHantenna1 + TXDIV_DELTAPHASE
	Closed Loop(mode1)	Phase of Each DPCHantenna1 + TXDIV_DELTAPHASE + 45 ***1**
	Closed Loop(mode2)	Phase of Each DPCHantenna1 + TXDIV_DELTAPHASE + 180 ***2**

***1** It is based on TS25.214 section 7.2.

***1,*2** The Default phase difference between DPCH antenna1 and antenna2 (TXDIV_DELTAPHASE+45deg and TXDIV_DELTAPHASE+180deg) of Closed Loop mode1,2 are initialized every time at opening DPCH

***1,*2** During the operation of Closed Loop mode1,2, "Phase of Each DPCH antenna1 +TXDIV_DELTAPHASE" is added to the phase shift decided by the uplink FBI from Mobile Station, and this total phase becomes the phase rotation amount of antenna2

<Open loop operation>

For details, see 3GPP Standard TS 25.211-330.

<Closed loop mode1 operation>

TS25214-340 section 7.1 "The downlink slot in which the adjustment is done" is supported for 1) and 2).

<Closed loop mode2 operation>

The same as mode1, TS25214-340 section 7.1 "The downlink slot in which the adjustment is done" is supported for 1) and 2).

3.4.5 Transmission power setting

< Transmission power setting when TX Diversity operates >

The transmission power for each antenna when TX Diversity operates shall be set by BTS#1, #2 of the TX Setup screen.
For instance, when the transmission power total from both antenna1 and antenna2 is made "-20dBm", transmission power antenna1 and antenna2 respectively is set to "-23dBm". (Power becomes half when lowering by 3dB.)

However, please note that the method of the power setting is different only for the case of Closed Loop Mode2 of TX Diversity. If the signalling tester should work in Closed Loop Mode2, it is necessary to make it to a set value that is 3dB higher compared with a set value of other modes. For instance, when the transmission power total from both antenna1 and antenna2 is made "-20dBm", transmission power antenna1 and antenna2 respectively is set to "-20dBm".

When OCNS is transmitted, OCNS power of each antenna and DPCH power at that time are set according to scenario function BtsOcnsActivate().

Table 3-3 shows the above-mentioned content.
A set value of antenna1/antenna2 becomes transmission power for P-SCH,S-SCH to which TSTD is applied.

Table 3-3 Transmission power setting at TX Diversity

	antenna1				antenna2			
	CCH	DPCH	OCNS		CCH	DPCH	OCNS	
	CCH power	DPCH power	OCNS power	DPCCH power	CCH power	DPCH power	OCNS power	DPCCH power
Excluding CL Mode2	Z-3 [dBm]	X-3 [dBm]	Y-3 [dBm]	X-3 [dBm]	Z-3 [dBm]	X-3 [dBm]	Y-3 [dBm]	X-3 [dBm]
CL Mode2		X [dBm]	Y-3 [dBm]	X-3 [dBm]		X [dBm]	Y-3 [dBm]	X-3 [dBm]

*This table indicates the transmission power setting value.
X[dBm]: DPCH transmission power to synthesize antenna1 to antenna2.
Y[dBm]: OCNS transmission power to synthesize antenna1 to antenna2.
Z[dBm]: CCH transmission power to synthesize antenna1 to antenna2.

*CCH shows either of "P-CCPCH,P-CPICH,S-CCPCH,AICH,PICH". Because Closed Loop Mode2 is not defined for these channels, it makes it to the slash in this table.

* DPCCH Power under the column of OCNS is a value that should be set as a parameter of BtsOcnsActivate(). Moreover, it is decided whether the column of CL Mode2 should be referred in this table for OCNS depending on the mode of DPCH.

* the setting range to CCH power of each antenna is "-99dBm≤CCH powe≤-10dBm".

*The setting range to DPCH power of each antenna is "-99dBm≤DPCH power≤-10dBm", except for CL Mode2.

* The setting range to DPCH power of each antenna is "-96dBm≤DPCH power≤-13dBm" for CL Mode2.

< Range of DPCH power control>

When the power control by TPC of the DPCH power is executed, it is necessary to set the range of the power control. A range of the DPCH power control each antenna1, antenna2 should be set ahead of the CphyRlSetup() function, and set like the following example. (Please refer to A.2.2.5 Structure for CphyRlSetup().)

ex)

```
CphyRlSetupD_DPCH.MaxDLPower = -10;
/*DL DPCH MAX Power -10dBm */
CphyRlSetupD_DPCH.MinDLPower = -73;
/*DL DPCH MIN Power -73dBm */
```

Moreover, when OCNS starts, the range of the DPCH power control is set as a parameter of the BtsOcnsActivate() function.

When the range of the power control of the DPCH transmission power of each antenna is assumed to be "Pmax [dBm] to Pmin [dBm]", a set value is indicated in Table 3-4.

Table 3-4. DPCH power control range setting at TX Diversity

	Range of DPCH power control (setting by structure for CphyRlSetup())		Range of DPCH power control set when OCNS starts (value set as parameter of BtsOcnsActivate() function)	
	MinDLPower	MaxDLPower	MinDLPower	MaxDLPower
Excluding CL Mode2	Pmin [dBm]	Pmax [dBm]	Pmin [dBm]	Pmax [dBm]
CL Mode2	Pmin+3 [dBm]	Pmax+3 [dBm]	Pmin [dBm]	Pmax [dBm]

* This table indicates a set value when the range of the power control of the DPCH transmission power of each antenna is assumed to be "Pmax [dBm] to Pmin [dBm]".

* The setting range to "MinDLPower" and "MaxDLPower" is "-99dBm≤MinDL-Power≤-10dBm" "-99dBm≤MaxDLPower≤-10dBm", except for CL Mode2.

* The setting range to "MinDLPower" and "MaxDLPower" is "-96dBm≤MinDLPower≤-13dBm" "-96dBm≤MaxDLPower≤-13dBm" for CL Mode2.

3.5 Handover Test

In case a mobile station moves from the coverage of a base station to the coverage of another, the mobile station seeks for a next base station while communicating with the first base station. The mobile station compares radio wave from the first base station with that from the next one, the latter becoming stronger. When the mobile station determines that it ought to shift to the next base station, it ceases communications with the first base station and transmits data to the next one. The mobile unit repeats this operation as it roams. This operation is called "handover."

The handover is classified into a soft handover (SHO) and a hard handover (HHO). The SHO is a method whereby switching of base stations is made gradually and the HHO is a method whereby switching is made instantaneously.

Figure 3-7 shows the conceptional diagram of the SHO. Figure 3-8 shows the conceptional diagram of the HHO.

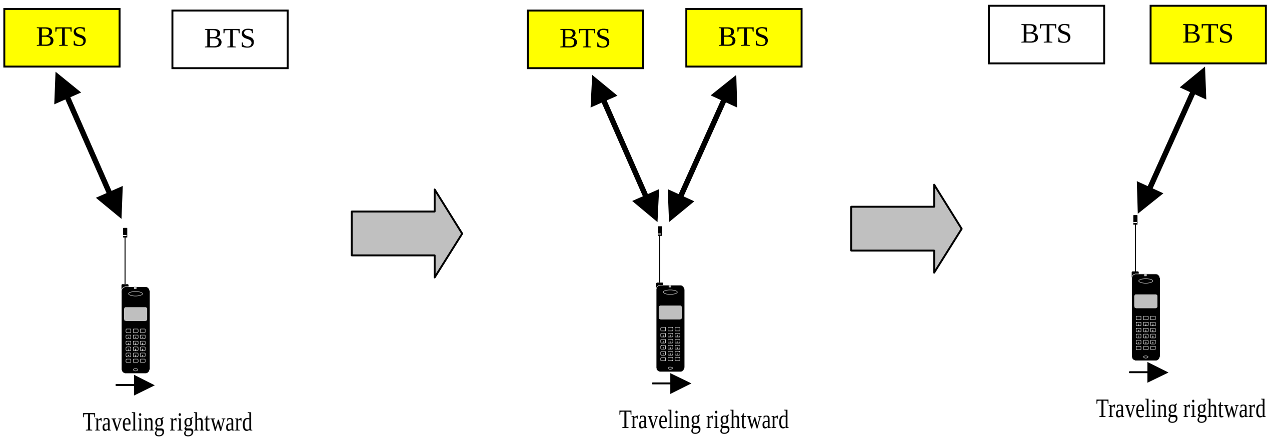


Figure 3-7 Conceptional diagram of SHO

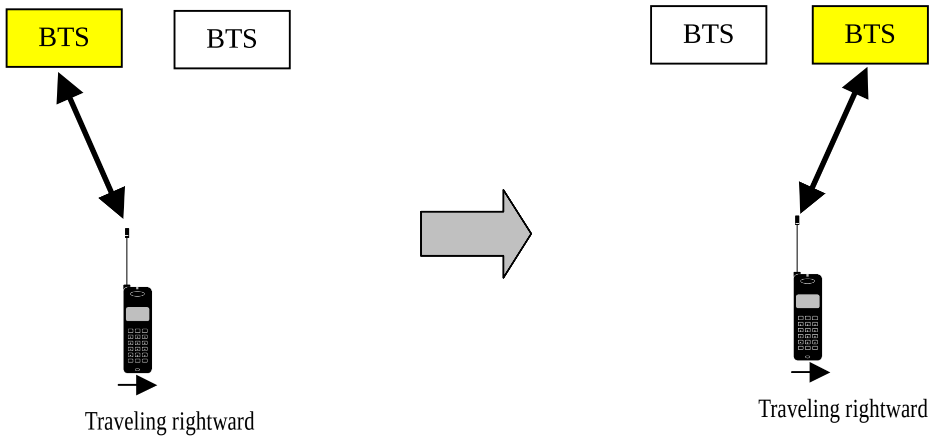


Figure 3-8 Conceptional diagram of HHO

3.5.1 Setup for handover

3.5.2 SHO test procedure

The SHO test is conducted in the following procedure:

3.5.2.1 Setup of BTS connection

1) 2BTSs

This configuration operates only in the configuration where options 2nd Frame Coder and 2nd TX Baseband are added.

Note)

In case 2nd Frame Coder and 2nd TX Baseband are not added, the configuration will not operate.

For details of the configuration, see the Operation Manual.

To perform handover, see the Download Manual to download the firmware (MD8480A.dl). (Since the handover software is included in the firmware, it is not necessary to download the firmware anew in case the firmware has been downloaded.)

Set the Parameter Setup screen of the Control Software as shown in Table 3-4.

Table 3-4 Setting on Parameter Setup screen (2BTSs)

	Base Band and RF Unit Connection		SHO1
Parameter Setup screen	BTS#1	Tx Use	ON
		Rx Use	ON
	BTS#2	Tx Use	ON
		Rx Use	OFF
	BTS#3	Tx Use	OFF
		Rx Use	OFF

2) 3BTSs

This configuration operates only in the configuration where options 2nd Frame Coder, 2nd TX Baseband and 3rd TX Baseband are added.

Note)

In case 2nd Frame Coder, 2nd TX Baseband and 3rd TX Baseband are not added, the configuration will not operate.

For details of the configuration, see the Operation Manual.

To perform handover, see the Download Manual to download the firmware (MD8480A.dl). (Since the handover software is included in the firmware, it is not necessary to download the firmware anew in case the firmware has been downloaded.)

Set the Parameter Setup screen of the Control Software as shown in Table 3-5.

Table 3-5 Setting on Parameter Setup screen (3BTSs)

Parameter Setup screen	Base Band and RF Unit Connection		SHO1
	BTS#1	Tx Use	ON
		Rx Use	ON
	BTS#2	Tx Use	ON
		Rx Use	OFF
	BTS#3	Tx Use	ON
		Rx Use	OFF

3.5.2.2 Channel activation/de-activation procedure

Use an appropriate scenario to activate or de-activate channels, BTS#1, #2, and #3. The scenarios necessary for handover are stored in C:¥MX848000¥Scenario¥Src¥SHO_Sample.

1) Start_BTS1_ACT(AMRxP384K_TTI20).dll
To activate UDI, execute Start_BTS1_ACT(UDI).c first.
To activate AMR, execute Start_BTS1_ACT(AMR).c first.
This results in:
BTS#1 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#2 transmit channel: None
BTS#3 transmit channel: None

2) DCH_Data(DCCH).dll, DCH_Data(AMR).dll, DCH_Data(Packet).dll
DCH_Data(DCCH).dll, DCH_Data(AMR).dll, DCH_Data(Packet).dll, to successively output data in the DPCH activated under (1).

3) BTS2_Act.c
This scenario activates P_SCH, S_SCH, P_CPICH, and P_CCPCH of BTS#2.
This results in:
BTS#1 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#2 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH
BTS#3 transmit channel: None

4) DPCH_Act_SHO.c
This scenario activates DPCH of BTS#1, #2, and #3. Select BTS#2 here. DPCH of BTS#2 is activated and the configuration of transmit channels will be:
BTS#1 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#2 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#3 transmit channel: None
DPCH transmit data of BTS#2 is the same as data specified under 2).

5) BTS3_Act.c
This scenario activates P_SCH,S_SCH,P_CPICH, and P_CCPCH of BTS#3. The configuration of transmit channels will be:
BTS#1 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#2 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#3 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH

6) DPCH_Act_SHO.c
This scenario activates DPCH of BTS#1, #2, and #3. Select BTS#3 here. DPCH of BTS#3 is activated and the configuration of transmit channels will be:
BTS#1 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#2 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#3 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH

Transmit data of BTS#3 is the same as data specified under 2).

7) DPCH_Dea_SHO.c
This scenario halts DPCH of BTS#1, #2, and #3. For example, when halting DPCH#2 in the state of 6), the configuration of transmit channels will be:
BTS#1 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH
BTS#2 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH
BTS#3 transmit channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH

3.5.2.3 Control of transmission power per BTS

Use BTS_Power.c. There are settings to specify BTS#1, #2, and #3and Up/Down. Power of the specified BTS. They will increase or decrease the power. Value of increasing/decreasing the power of each BTS is defined by BTS_POWER in BTS_Power.c.
In the sample scenarios, the BTSPower() function is used.
For details, see Appendix A SCENARIO LIBRARIES.

3.5.3 HHO test procedure

The HHO is classified into "channel change test," "Intra-frequency Hard Handover," and "Inter-frequency Hard Handover."

3.5.3.1 Channel change test (Channel change from 4B to 4B+Packet128[kbps])

Used scenario: Start_4B.dll,ACT_4BxP128K.dll

1) Parameter setup
Set Base Band and RF Unit Connection on the BTS Connection screen to "BASIC."
On the BTS#1 Setup screen, set P_SCH, S_SCH, P_CPICH, P_CCPCH and DPCH to 4B.
Set DPCH settings as 4B parameters.

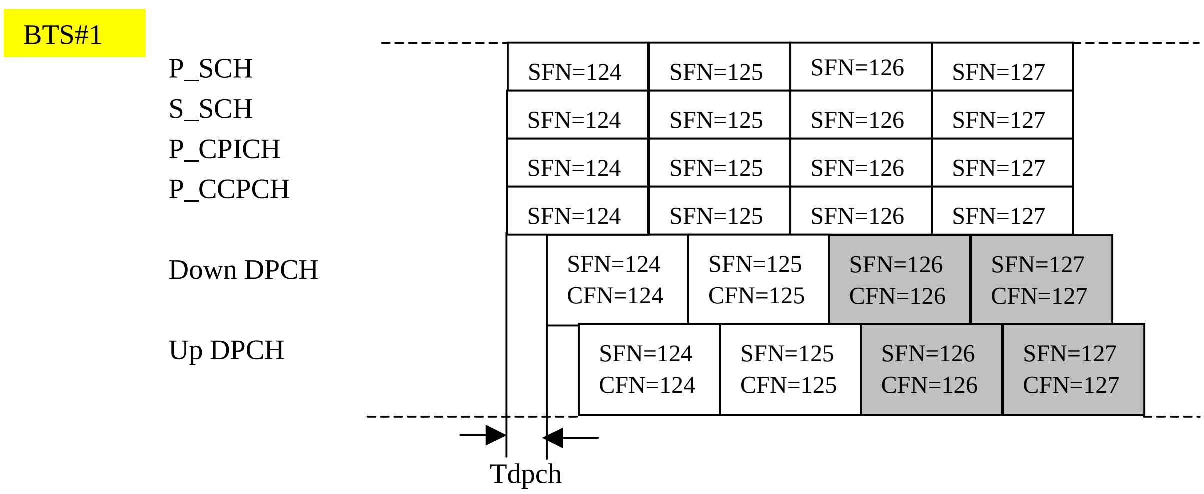
2) Start_4B.dll and press the START button.
When Start_4B.dll is terminated, the activation channel will be:
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(4B)

3) Use Dch_Data(1B-6B).c, Dch_Data(DCCH).c to transmit 4B data to DPCH.

4) Select and execute ACT_4BxP128K.dll
Switching is made as follows after the current CFN+64 frames.
BTS#1 activation channel:
P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(4B+P128[kbps])
Up to the current CFN+63 frames, transmission/reception state is as shown under 2).

5) Use Dch_Data(Packet).c to transmit packet data.

Radio transmit/receive signal (when Activation Time: CFN=126)



Up to CFN=125(SFN=125), the format of DPCH is AMR, and for CFN=126(SFN=126) and after, the format is 4B+P128[kbps].

3.5.3.2 Switching from BTS#1 to BTS#2 (Intra-frequency handover--without 2nd Rx Baseband (slot7, 8))

Used scenario:

Start_5B(HHO).dll,BTS1toBTS2(5BxP64K).dll

1) Parameter setup
Set Base Band and RF Unit Connection on the BTS Connection screen to "SHO1."
On the BTS#1 Setup screen, set P_SCH,S_SCH,P_CPICH,P_CCPCH and DPCH to 5B.
Set DPCH as a 5B parameter.
On the BTS#2 Setup screen, set P_SCH,S_SCH,P_CPICH,P_CCPCH and DPCH to 5BxP64[kbps].
Set DPCH as an 5BxP64[kbps]parameter.
On the Timing screen, set BTS Offset and Tdpch of BTS#1 and BTS Offset and Tdpch of BTS#2.
Here, arrange so that (BTS Offset of BTS#1+ Tdpch of BTS#1)=(BTS Offset of BTS#2+Tdpch of BTS#2) and match the transmission/reception timing of DPCH.

2) Start_5B(HHO).dll and press the START button.
Start_5B(HHO).dll is terminated, the activation channels will be:
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(AMR)
BTS#2 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH

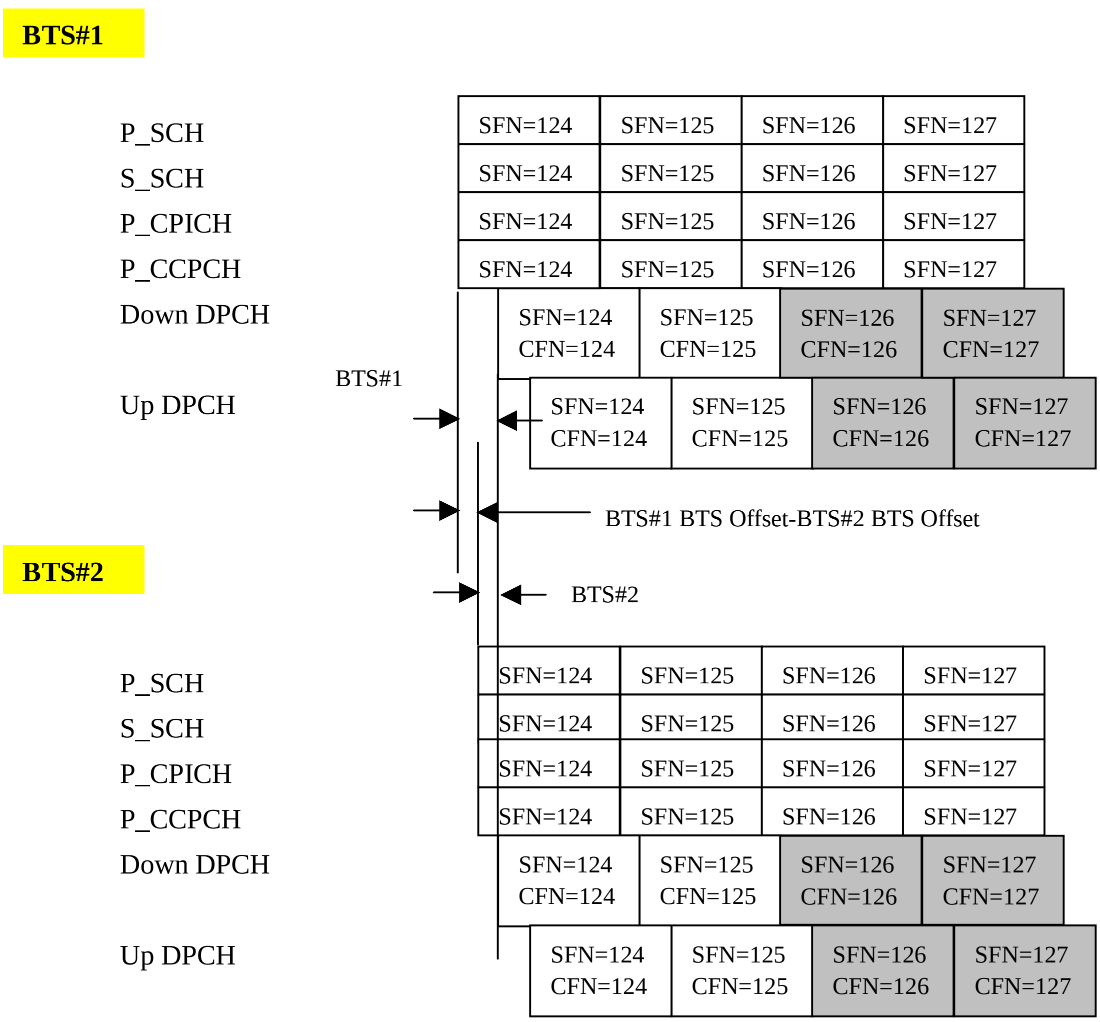
3) Use Dch_Data(1B-6B).c, Dch_Data(DCCH).c to transmit 5B data to DPCH.

4) Select and execute BTS1toBTS2(5BxP64[kbps]).c.
Switching is made as follows after the current CFN+64 frames.
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH
BTS#2 activation channel:
P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(5BxP64[kbps])
Up to the current CFN+63 frames, transmission/reception state is as shown under 2).

Caution:

In this configuration, the BTS#1 hardware is used to transmit/receive DPCH of BTS#2.
Thus, the BTS column on the Trace screen remains 1 even after switchover to BTS#2 (after execution of (4)).

Radio transmit/receive signal (when Activation Time: CFN=126)



Up to CFN=125(SFN=125), the parameters of DPCH are those of BTS#1, and for CFN=126(SFN=126) and after, the parameters are those of BTS#2.

3.5.3.3 Switching from BTS#1 to BTS#2 (Intra-frequency handover--with 2nd Rx Baseband (slot7, 8))

Used scenario:
Start_5B(HHO).dll,ACT_5BxP64K.dll

1) Parameter setup
Set Base Band and RF Unit Connection on the BTS Connection screen to "HHO."
Set the frequencies of TX RF#1 and TX RF#2 and the frequencies of RX RF#1 and RX RF#2, at the same value between each of these pairs.
On the BTS#1 Setup screen, set P_SCH,S_SCH,P_CPICH,P_CCPCH and DPCH to 5B.
Set DPCH as a 5B parameter.
On the BTS#2 Setup screen, set P_SCH,S_SCH,P_CPICH,P_CCPCH and DPCH to 5BxP64[kbps].
Set DPCH as an 5BxP64[kbps]parameter.
On the Timing screen, set BTS Offset and Tdpch of BTS#1, and BTS Offset and Tdpch of BTS#2.
Here, arrange so that (BTS Offset of BTS#1+ Tdpch of BTS#1)=(BTS Offset of BTS#2+Tdpch of BTS#2) and match the transmission/reception timing of DPCH.

2) Start_5B(HHO).dll and press the START button.
When Start_5B(HHO).dll.dll is terminated, the activation channels will be:
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(AMR)
BTS#2 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH

3) Use Dch_Data(1B-6B).c, Dch_Data(DCCH).c to transmit 5B data to DPCH.

4) Select and execute ACT_5BxP64K.dll
Switching is made as follows after the current CFN+64 frames.
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH
BTS#2 activation channel:
P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(5BxP64[kbps])
Up to the current CFN+63 frames, transmission/reception state is as shown under 2).
After the completion of switchover, the column of BTS on the Trace screen changes to 2.

The radio transmit/receive signal is the same as that under 2. Switching from BTS#1 to BTS#2 (Intra-frequency handover--without 2nd Rx Baseband (slot7, 8)).

3.5.3.4 Switching from BTS#1 to BTS#2 (Inter-frequency handover)

To execute this procedure, 2nd Rx Baseband (Slot7,8) is required.

Used scenario:
Start_5B(HHO).dll,ACT_5BxP64K.dll

1) Parameter setup
Set Base Band and RF Unit Connection on the BTS Connection screen to "HHO."
Set the frequencies of TX RF#1 and TX RF#2 and the frequencies of RX RF#1 and RX RF#2, at the different values for each of these pairs.
On the BTS#1 Setup screen, set P_SCH,S_SCH,P_CPICH,P_CCPCH and DPCH to 5B.
Set DPCH as an 5B parameter.
On the BTS#2 Setup screen, set P_SCH,S_SCH,P_CPICH,P_CCPCH and DPCH to 5BxP64[kbps].
Set DPCH as an 5BxP64[kbps] parameter.
On the Timing screen, set BTS Offset and Tdpch of BTS#1 and BTS Offset and Tdpch of BTS#2.
Here, arrange so that (BTS Offset of BTS#1+ Tdpch of BTS#1)=(BTS Offset of BTS#2+ Tdpch of BTS#2) and match the transmission/reception timing of DPCH.

2) Select Start_5B(HHO).dll and press the START button.
When Start_5B(HHO).dll is terminated, the activation channels will be:
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(5B)
BTS#2 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH

3) Use Dch_Data(1B-6B).c, Dch_Data(DCCH).c to transmit 5B data to DPCH.

4) Select and execute ACT_5BxP64K.dll
Switching is made as follows after the current CFN+64 frames.
BTS#1 activation channel: P_SCH,S_SCH,P_CPICH,P_CCPCH
BTS#2 activation channel:
P_SCH,S_SCH,P_CPICH,P_CCPCH,DPCH(5BxP64[kbps])
Up to the current CFN+63 frames, transmission/reception state is as shown under 2).

The radio transmit/receive signal is the same as that under 2. Switching from BTS#1 to BTS#2 (Intra-frequency handover--with 2nd Rx Baseband (slot7, 8)), except that the frequency differs between BTS#1 and BTS#2.

3.5.4 GSM Handover test

For details of test , see " Appendix E GSM/GPRS Option "

3.6 Compressed Mode Test

3.6.1 What is Compressed Mode

The Compressed Mode is used to create the time to monitor cells on other frequencies while communications are in progress. Monitoring is made while communications are under way, so that the ordinary communications data is compressed by the Compressed Mode and radio frames undergo time-division so that this communication data is distinguished for use by ordinary communications and monitoring. (For details, see 3GPP Standard TS25.211-215.)

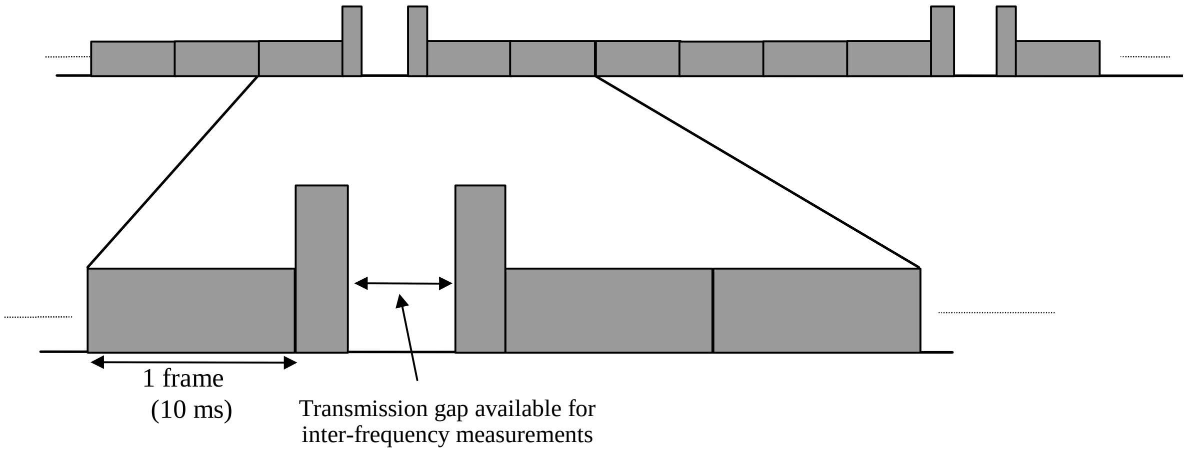


Figure 3-9 Compressed mode transmission

3.6.2 Compressed Mode of MD8480A/B

3.6.2.1 Corresponded Standards

An algorithm is pursuant to 3GPP Standards R99 TS25.212-350.

3.6.2.2 Supported Compressed Mode Parameter range

Compressed Mode parameters are pursuant to 3GPP Standards R99 TS25.215-370 6.1.1.2. The range of each parameter is pursuant to 3GPP Standards R99 TS25.331-370 10.3.6.33.

The concrete range of each parameter is as follows:

Parameter	Range
TGCFN	0..255, invalid
TGPRC	1..511, Infinity
TGSN	0..14
TGL1	1..14
TGL2	1..14, 0
TGD	15..269, 0
TGPL1	1..144
TGPL2	1..144, 0
Downlink compressed mode method	SF/2, puncturing, higher layer scheduling
Uplink compressed mode method	SF/2, higher layer scheduling
Downlink frame type	A, B
DeltaSIR1	0..3 by step of 0.1
DeltaSIRafter1	0..3 by step of 0.1
DeltaSIR2	0..3 by step of 0.1
DeltaSIRafter2	0..3 by step of 0.1
TGPS reconfiguration CFN	0..255, invalid
TGPSI	1..MaxTGPS(6)
Scrambling Code Change	left alternative, right alternative, no change
UL/DL compressed mode selection	DL only, UL only, UL&DL
RPP	no support
ITP	no support

3.6.2.3　MultiTGP and SingleTGP

MD8480A/B has 2 ways to set up gap patterns, they are MultiTGP and SingleTGP. There are differences below between MutliTGP and SingleTGP. And expression of “MultiTGP” and “SingleTGP” are used in following description.

	Scenario library functions to be used	Number of TGPS to be set up by single execution	Bearer modification when using multiple TGPS
MultiTGP	CphyRlSetupMtgp() CalcRMParameterMtgp()	6	Available
SingleTGP	CphyRlSetup() CalcRMParameterCM()	1	Not available

Both of Single TGP and MultiTGP are used in one scenario with MD8480A/B software V5.20 or later.

Difference Structures are used in MultiTGP and SingleTGP.

	Radio Link set up	Coding set up	MAC set up
MultiTGP	CPHY_RL_SETUP_MTGP_PAR	CPHY_TRCH_CONFIG_PAR (common)	CMAC_CONFIG_PAR (common)
SingleTGP	CPHY_RL_SETUP_PAR		

3.6.2.4　Restrictions

-There is no plant to apply to items below :

Uplink power control when compressed mode (3GPP specification R99 TS25214-370 5.1.2.3)

-This document describes operation using software V5.20. Some operation is different depending on software version. Please refer section 3.6.7.3.

-MutliTGP of V5.20 only applies to single BTS, and SF/2 of Method. It is confirmed that operation using 0 to 255 of Activation Time at configuration.

-When using multiple TGPS, MD8480A/B is not able to change bearer combination with SingleTGP.

-Gap pattern keeping function is not available when multiple TGPS are used with SingleTGP.

-When executing scenario library, it takes 255 frames at maximum from current CFN to specified Activation Time. And TGCFN becomes effective after Activation Time is effective. So it takes 510 frames at maximum from execution of scenario library function on scenario to timing that TGCFN is effective.

3.6.3 Operation and how to use Compressed Mode

3.6.3.1 MultiTGP

3.6.3.1.1 general rule of how to use

-CalcRMParameterMtgp(), CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig() should be executed when setting MultiTGP.
-Value of 0 to 255 should be set when configuring the value other than NONE to Method of Gap Pattern
-Value of 0 to 255 should be set to Activation Time. Please do not use ACTIVATE_NOW.

3.6.3.1.2 Contents of each parameters

Please refer section A.2.2.6 Structure for CPHY_RL_SETUP_MTGP_PAR.

3.6.3.1.3 Procedure and operation of gap pattern activation

(procedure)
-Specify values below to parameter of CPHY_RL_SETUP_MTGP_PAR, that is scenario structure. This procedure is needed only for TGPS to activate. Each TGPS parameters are set independently. n=0 to 5 corresponds to TGPS#1 to TGPS#6.

Scenario structure parameter (They are included in CPHY_RL_SETUP_MTGP_PAR.)	value
GapInfoFlag [n]	GAP_KEEP_OFF
GapInfo[n].Method	One of CM_MODE_SFR, CM_MODE_PUNC, CM_MODE_HLS
GapInfo[n].TGCFN	One of 0 to 255
Other parameters of GapInfo[n]	Arbitrary value within the specification range

-Specify same Activation Time(0 to 255) to CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig().
-Execute CalcRMParameterMtgp(), CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig() with this sequence.
-(Caution) Specify GAP_KEEP_OFF to GapFlagInfo, and CM_MODE_NONE to Method for TGPS which is not compressed mode.
-(Caution)Please refer section 3.6.3.1.6 for TGPS which doesn’ t change gap pattern.

(Operation)
-It activates compressed mode at TGCFN frame which comes after timing that Activation Time is effective. When Activation Time and TGCFN are same, they are effective at the same frame.

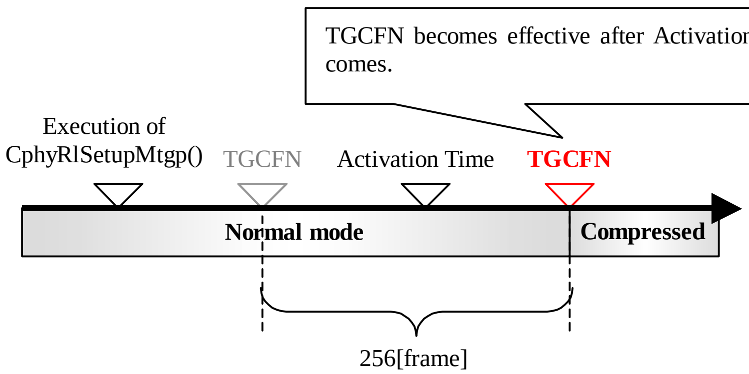


Figure 3-10 The timing that TGCFN becomes effective

3.6.3.1.4 Procedure and operation of gap pattern suspension

MD8480A/B has 2 way to suspend gap pattern with MultiTGP. One way uses Activation Time and the other uses TGPRC.

(1) Using Activation Time

(Procedure)

-When using MulitTGP, Activation Time should be applied to suspend gap pattern of MD8480A/B, even if TGPS reconfiguration CFN is applied to suspend gap pattern of UE.

-Specify values below to parameter of CPHY_RL_SETUP_MTGP_PAR, that is scenario structure. This procedure is needed only for TGPS to suspend. Each TGPS parameters are set independently. n=0 to 5 corresponds to TGPS#1 to TGPS#6.

Scenario structure parameter (They are included in CPHY_RL_SETUP_MTGP_PAR.)	value
GapInfoFlag [n]	GAP_KEEP_OFF
GapInfo[n].Method	CM_MODE_NONE
Other parameters of GapInfo[n]	(omitted)

-Specify same Activation Time(0 to 255) to CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig(). This Activation Time is used to suspend gap patterns.

-Execute CalcRMParameterMtgp(), CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig() with this sequence.

-(Caution) Specify GAP_KEEP_OFF to GapFlagInfo, and CM_MODE_NONE to Method for TGPS which is not compressed mode.

(Operation)

-It suspends compressed mode at specified Activation time frame.

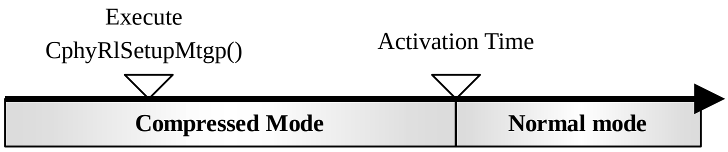


Figure3-11 The timing that TGCFN suspends

(2) using TGPRC

(Operation)

It repeats gap pattern for times specified in Scenario parameter, TGPRC. After that, it suspends gap pattern.

3.6.3.1.5 Procedure and operation of gap pattern change

With MultiTGP, multiple TGPSs can be changed with single execution.
(Procedure)
-Specify values below to parameter of CPHY_RL_SETUP_MTGP_PAR, that is scenario structure. This procedure is needed only for TGPS to change. Each TGPS parameters are set independently. n=0 to 5 corresponds to TGPS#1 to TGPS#6.

Scenario structure parameter (They are included in CPHY_RL_SETUP_MTGP_PAR.)	value
GapInfoFlag [n]	GAP_KEEP_OFF
GapInfo[n].Method	one of CM_MODE_SFR, CM_MODE_PUNC, CM_MODE_HLS
GapInfo[n].TGCFN	one of 0 to 255
Other parameters of GapInfo[n]	Arbitrary value within the specification range

-Specify same Activation Time(0 to 255) to CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig(). This Activation Time is used to suspend gap patterns before change.
-Execute CalcRMParameterMtgp(), CphyRlSetupMtgp(), CphyTrchConfig(), Cmac-Config() with this sequence.
-(Caution) Specify value other than NONE when setting value of 0 to 255 to TGCFN.
-(Caution) (Caution) Specify GAP_KEEP_OFF to GapFlagInfo, and CM_MODE_NONE to Method for TGPS which is not compressed mode.
-(Caution) (Caution)Please refer section 3.6.3.1.6 for TGPS which doesn' t change gap pattern.

(Operation)
-It suspends compressed mode at Activation Time timing on TGPS on which GAP_KEEP_OFF is specified to GapInfoFlag[n].
-Compressed mode is activated at TGCFN frame which comes after Activation Time is effective. If Activation Time and TGCFN are same value, they are effective at the same time.
-Other TGPSs continue doesn' t change their operation.

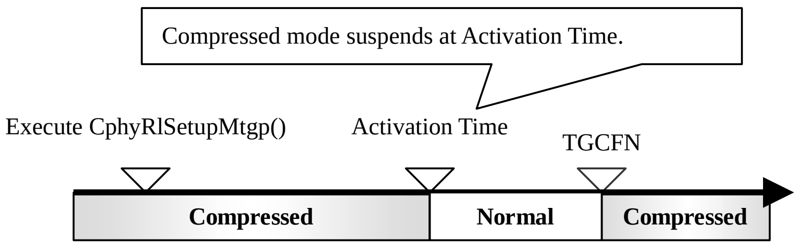


Figure 3-12 Timing to change gap pattern with MultiTGP

3.6.3.1.6 Procedure and operation of gap pattern keeping

-It is possible to keep gap patterns(to continue without initialization) when bearer combination in compressed mode is reconfigured.

-When releasing DPCH, and after that activating DPCH of same bearer combination using this functionality, gap pattern that is same as before releasing are continued.

(Procedure)
-Specify values below to parameter of CPHY_RL_SETUP_MTGP_PAR, that is scenario structure. This procedure is needed only for TGPS to continue gap pattern. Each TGPS parameters are set independently. n=0 to 5 corresponds to TGPS#1 to TGPS#6.

Scenario structure parameter (They are included in CPHY_RL_SETUP_MTGP_PAR.)	value
GapInfoFlag [n]	GAP_KEEP_ON
GapInfo[n].Method	Same as change of bearer combination
GapInfo[n].TGCFN, GapInfo[n].TGPRC	(omitted)
GapInfo[n]. Method, TGL1, TGL2, TGPL1, TGPL2, TGSN, TGPSI, TGD, FrameType	Same as change of bearer combination
Other parameters of GapInfo[n]	Arbitrary value within the specification range

-Specify same Activation Time(0 to 255) to CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig().
-Execute CalcRMParameterMtgp(), CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig() with this sequence.
(Operation)
-Bearer combination is changed at Activation Time, but gap pattern keeps the pattern before bearer combination change.
(Caution)
-MD8480A/B continues to count TGPRC after changing bearer combination. (For example, TGPRC specified for 1st bearer combination is 100. 2nd CphyRlSetup() is executed after 30times of repetition. At this time, remaining 70 times of repetition is executed after 2nd CphyRlSetup() with keeping gap pattern.)

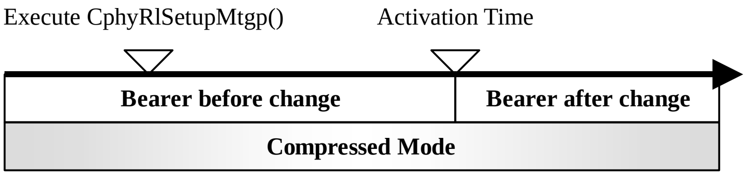


Figure 3-13 Gap pattern keeping of MultiTGP

3.6.3.2 SingleTGP

3.6.3.2.1 general rule of how to use

-CalcRMParameter(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() should be executed when setting SingleTGP.
-Value of 0 to 255 should be set when configuring the value other than NONE to Method of Gap Pattern
-Basically, value of 0 to 255 should be set to Activation Time. Please do not use ACTIVATE_NOW.

3.6.3.2.2 Contents of each parameters

Please refer section A.2.2.4 Structure for CphyRlSetup() (CPHY_RL_SETUP_PAR structure).

3.6.3.2.3 Procedure and operation of gap pattern activation

(Procedure)
-Specify values below to parameter of CPHY_RL_SETUP_PAR, that is scenario structure.

Scenario structure parameter (They are included in CPHY_RL_SETUP_PAR.)	value
GapInfoFlag	GAP_KEEP_OFF
GapInfo.TGPSI	Number of TGPS to activate compressed mode : TGPSI_1 to TGPSI_6
GapInfo.Method	One of CM_MODE_SFR, CM_MODE_PUNC, CM_MODE_HLS
GapInfo.TGCFN	0 to 255
GapInfo.TGPS_Reconfig_CFN	INVALID_PARAM
Other parameters of GapInfo	Arbitrary value within the specification range

-Specify same Activation Time(0 to 255) to CphyRlSetup(), CphyTrchConfig(), CmacConfig().
-Execute CalcRMParameter(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() with this sequence.
(Operation)
-On specified TGPS, it activates compressed mode at TGCFN frame which comes after timing that Activation Time is effective. When Activation Time and TGCFN are same, they are effective at the same frame.
-Other TGPS doesn’ t change operation.

3.6.3.2.4 Procedure and operation of gap pattern suspension

Procedure and operation of gap pattern suspension with SingleTGP differs depending on software version of MD8480A/B. Please refer section 3.6.3.7 for detail.
Here describes how to suspend gap pattern using SingleTGP. It can suspend one TPGS gap pattern with single execution. MD8480A/B has 3 way to suspend gap pattern with SingleTGP. First one uses Activation Time, 2nd one uses TGPS reconfiguration CFN, and the other uses TGPRC.

(1) Using Activation Time
(Procedure)
-Specify values below to parameter of CPHY_RL_SETUP_PAR, that is scenario structure.

Scenario structure parameter (They are included in CPHY_RL_SETUP_PAR.)	value
GapInfoFlag	GAP_KEEP_OFF
GapInfo.TGPSI	Number of TGPS to suspend compressed mode : TGPSI_1 to TGPSI_6
GapInfo.Method	CM_MODE_NONE
GapInfo.TGPS_Reconfig_CFN	INVALID_PARAM
Other parameters of GapInfo	(omitted)

-Specify same Activation Time(0 to 255) to CphyRlSetup(), CphyTrchConfig(), CmacConfig(). This is used to suspend gap pattern.
-Execute CalcRMParameter(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() with this sequence.
(Operation)
-It suspends compressed mode at specified Activation time frame on specified TGPS.
-Other TGPS doesn' t change.

(2) Using TGPS reconfiguration CFN
(Procedure)
-MD8480A/B uses TGPS_Reconfig_CFN which is included in structure CPHY_RL_SETUP_MTGP_PAR.
-Specify values below to parameter of CPHY_RL_SETUP_PAR, that is scenario structure.

Scenario structure parameter (They are included in CPHY_RL_SETUP_PAR.)	value
GapInfoFlag	GAP_KEEP_OFF
GapInfo.TGPSI	Number of TGPS to suspend compressed mode : TGPSI_1 to TGPSI_6
GapInfo.Method	CM_MODE_NONE
GapInfo.TGPS_Reconfig_CFN	one of 0 to 255
Other parameters of GapInfo	(omitted)

-Specify same value as TGPS_Reconfig_CFN to Activation Time of CphyRlSetup(), CphyTrchConfig(), CmacConfig()
-Execute CalcRMParameter(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() with this sequence.
(Operation)
-It suspends compressed mode at specified TGPS_Reconfig_CFN frame on specified TGPS.
-Other TGPS doesn' t change their operation.

(3) Using TGPRC
(Operation)

-It repeats gap pattern for times specified in Scenario parameter, TGPRC. After that, it suspends gap pattern.

3.6.3.2.5 Procedure and operation of gap pattern change

Procedure and operation of gap pattern change with SingleTGP differs depending on software version of MD8480A/B. Please refer section 3.6.3.7 for detail.

Here describes how to change gap pattern using SingleTGP. It can change one TPGS gap pattern with single execution.

(Procedure)

-Specify values below to parameter of CPHY_RL_SETUP_PAR, that is scenario structure.

Scenario structure parameter (They are included in CPHY_RL_SETUP_PAR.)	value
GapInfoFlag	GAP_KEEP_OFF
GapInfo.TGPSI	Number of TGPS to suspend compressed mode : TGPSI_1 to TGPSI_6:TGPSI_1 - TGPSI_6
GapInfo.Method	One of CM_MODE_SFR, CM_MODE_PUNC, CM_MODE_HLS
GapInfo.TGCFN	One of 0 to 255
GapInfo.TGPS_Reconfig_CFN	INVALID_PARAM or one of 0 to 255
Other parameters of GapInfo	Arbitrary value within the specification range

-With V5.20, specify CFN to suspend gap pattern before change gap pattern for Activation Time of CphyRlSetup(), CphyTrchConfig(), CmacConfig().

-Execute CalcRMParameter(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() with this sequence.

(Operation)

-At Activation Time frame, compressed mode suspends on specified TGPS.

-At TGCFN after Activation Time is effective, compressed mode activates on specified TGPS. If Activation Time and TGCFN are same, they are effective at same frame.

-If same bearer combination is set, other TGPSs don’ t change operation. If difference bearer combination is set, it can change gap pattern only when single TGPS is activated.

3.6.3.2.6 Procedure and operation of gap pattern keeping

-It is possible to keep gap patterns(to continue without initialization) when bearer combination in compressed mode is reconfigured.

-When releasing DPCH, and after that activating DPCH of same bearer combination using this functionality, gap pattern that is same as before releasing are continued.

(Procedure)

-Specify values below to parameter of CPHY_RL_SETUP_PAR, that is scenario structure.

Scenario structure parameter (They are included in CPHY_RL_SETUP_PAR.)	value
GapInfoFlag	GAP_KEEP_ON
GapInfo.TGPSI	Number of TGPS to suspend compressed mode : TGPSI_1 to TGPSI_6:TGPSI_1 - TGPSI_6
GapInfo.Method	Same as bearer combination change
GapInfo.TGCFN, GapInfo.TGPRC	(Omitted)
GapInfo.TGPS_Reconfig_CFN	INVALID_PARAM
GapInfo. Method, TGL1, TGL2, TGPL1, TGPL2, TGSN, TGPSI, TGD, FrameType	Same as bearer combination change
GapInfo	Arbitrary value within the specification range

-Execute CalcRMParameter(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() with this sequence.

(Operation)

-It changes bearer combination, but gap pattern on specified TGPS doesn’ t change its operation. (This functionality can be use only when single TGPS is activated.)

(Caution)

-MD8480A/B continues to count TGPRC after changing bearer combination. (For example, TGPRC specified for 1st bearer combination is 100. 2nd CphyRlSetup() is executed after 30times of repetition. At this time, remaining 70 times of repetition is executed after 2nd CphyRlSetup() with keeping gap pattern.)

3.6.3.3 Configuration of Alternative Scrambling Code

(Procedure) Common for MultiTGP and SingleTGP
-When configuring right alternative scrambling code, write scenario as below. (“CphyRlSetupD_DPCH” can be changed to “CphyRlSetupD_DPCH_BTS1” and others.)

```
SCR_CODE_TYPE_SET(CphyRlSetupD_DPCH.ScrCode,RIGHT_ALTER_SCR_CODE_CHG);
```

-When configuring left alternative scrambling code, write scenario as below. (“CphyRlSetupD_DPCH” can be changed to “CphyRlSetupD_DPCH_BTS1” and others.)

```
SCR_CODE_TYPE_SET(CphyRlSetupD_DPCH.ScrCode,LEFT_ALTER_SCR_CODE_CHG);
```

-When using MultiTGP, execute CalcRMParameterMTgp(), CphyRlSetupMtgp(), CphyTrchConfig(), CmacConfig() with this sequence.
-When using SingleTGP, execute CalcRMParameterCM(), CphyRlSetup(), CphyTrchConfig(), CmacConfig() with this sequence.

(Operation) Common for MultiTGP and SingleTGP
-Alternative Scrambling Code becomes effective on BTS specified in CphyRlSetupMtgp() (or CphyRlSetup()).

3.6.3.4 CphyRlRelease()

CphyRlRelease() doesn’ t give any influence to gap pattern.

3.6.3.5 Continuous configuration of Compressed Mode

Here uses expression below.

Word	Meaning
Configure Compressed Mode	Execute CalcRMParameterXXX(), CphyRlSetupXXX(), CphyTrchConfig(), CmacConfig() on MX848000A.
Available Compressed Mode setting	TGCFN becomes effective and compressed mode has activated.

Basically, if 2nd compressed mode is configured before 1st configured compressed mode is available, 2nd configuration is available. In following figure, 1st configured compressed mode is green, and 2nd one is red.

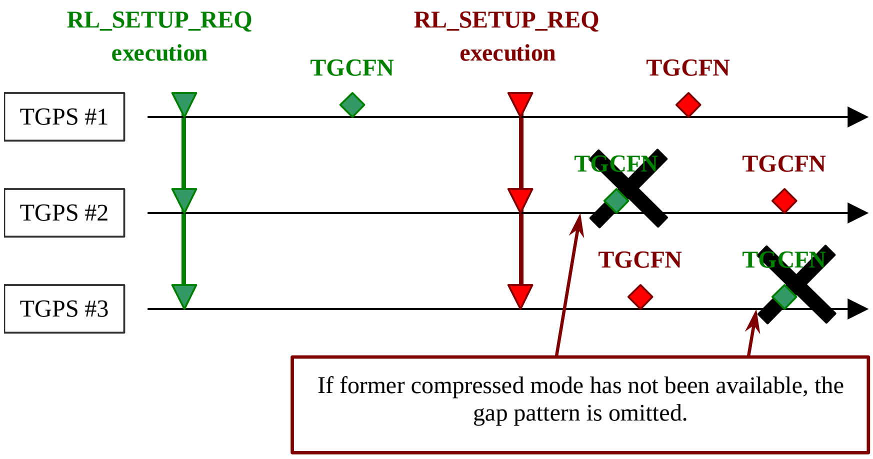


Figure 3-14 Continuous configuration of compressed mode

TGCFN is effective after Activation Time is effective, or at the same frame.

It takes 255 frames at maximum from current CFN to specified Activation Time. And TGCFN becomes effective after Activation Time is effective. So it takes 510 frames at maximum from execution of scenario library function on scenario to timing that TGCFN is effective. When executing continuous configuration of compressed mode, about 6 sec time is needed between these configurations. At this time, please use scenario library function, WaitTime().

3.6.3.6 Compressed Mode when TxDiversity

Here describes operation with version from firmware V5.11a to V5.20.

When configuring 2nd compressed mode before 1st TGCFN has come, 2nd configuration becomes available. But when configuring one of "DIVERSITY_OPEN_STTD", "DIVERSITY_OPEN_TSTD", "DIVERSITY_CLOSE_MODE1", "DIVERSITY_CLOSE_MODE2" to "TxDiversity" in structure CPHY_RL_SETUP_PAR, 1st configuration is available. This case is an exception.

3.6.3.7 Difference of SingleTGP operation depending on versions

3.6.3.7.1 Procedure and operation of gap pattern suspension

SigleTGP operation to suspend gap pattern is defined using 3 parameters of TGPS_Reconfig_CFN, TGCFN, Activation Time. They are configured in CphyRl-Setup(). The operation is different depending on MD8480A/B software version. When using scenarios which were made with software before V5.11, please take this into consideration. Please refer table below for operation differences. Version number tells control PC software(MX848000A) version. Please refer version combination table when selecting firmware.

TGPS_Reconfig_CFN	TGCFN	Timing to suspend gap pattern with version before or equal to V5.11	Timing to suspend gap pattern with version V5.11a, V5.11b, V5.20
INVALID_PARAM	0...255	CFN specified in TGCFN	CFN specified in Activation Time
	INVALID_PARAM	Not defined	
0...255	0...255	CFN specified in TGPS_Reconfig_CFN	
	INVALID_PARAM		

3.6.3.7.2 Procedure and operation of gap pattern change

SingleTGP operation to change gap pattern is defined using 3 parameters of TGPS_Reconfig_CFN, TGCFN, Activation Time. They are configured in CphyRl-Setup(). The operation is different depending on MD8480A/B software version. When using scenarios which were made with software before V5.11, please take this into consideration. Please refer table below for operation differences. Version number tells control PC software(MX848000A) version. Please refer version combination table when selecting firmware.

TGPS_Reconfig_CFN	TGCFN	Timing to change gap pattern with version before or equal to V5.11	Timing to change gap pattern with version V5.11a, V5.11b, V5.20
INVALID_PARAM	0...255	Change gap pattern at TGCFN	Suspend gap pattern at Activation Time. After that, activate gap pattern at TGCFN
	INVALID_PARAM	Not defined	Not defined
0...255	0...255	Suspend gap pattern at TGPS_Reconfig_CFN, and after that, activate gap pattern at TGCFN	Suspend gap pattern at Activation Time. After that, activate gap pattern at TGCFN.
	INVALID_PARAM	Suspend gap pattern at TGPS_Reconfig_CFN. After that, no gap pattern is activated.	Suspend gap pattern at Activation Time. After that, no gap pattern is activated.

3.6.3.7.3 Configuration of Alternative Scrambling Code

When configuring Alternative Scrambling Code using SingleTGP, the operation is different depending on MD8480A/B software version.When using scenarios which were made with software before V5.11, please take this into consideration. Please refer table below for operation differences. Version number tells control PC software(MX848000A) version. Please refer version combination table when selecting firmware.

term : "Alter.code" means "Alternative Scrambling Code".

Scenario description example(extract)	Operation after 2) with version before or equal V5.11	Operation after 2) with version V5.11a, V5.11b, V5.20
1) Configure Right Alter.code to TGPS#1 of BTS#1 2) Configure Left Alter.code to TGPS#2 of BTS#1	Right Alter.code is used on frame in which TGPS#1 is available. Left Alter.code is used in which TGPS#2 is available.	Latter configuration is available. Left Alter.code is used in both frame in which TGPSI#1 or TGPS#2 is available.
1) Configure Right Alter.code to TGPS#1 of BTS#1 2) Configure Left Alter.code to TGPS#1 of BTS#2	Common Alter.code is used for both of BTS#1 and BTS#2. Right Alter.code is used on frame in whichTGPS#1 is available. Left Alter.code is used on frame in which TGPS#2 is available.	BTS#1 and BTS#2 uses different configuration.Both BTSs uses same gap pattern, but BTS#1 uses Right Alter.code, and BTS#2 uses Left Alter.code.

3.6.3.7.4 gap parameters on multiple BTSs

When activating multiple BTSs on downlink, the compressed mode operation using SingleTGP is different depending on software version. When using scenarios which were made with software before V5.11, please take this into consideration. Please refer table below for operation differences. Version number tells control PC software(MX848000A) version. Please refer version combination table when selecting firmware.

Scenario description example(extract)	Operation after 2) with version before or equal V5.11	Operation after 2) with version V5.11a, V5.11b, V5.20
1) Configure gap pattern #1 to TGPS#1 of BTS#1 2) Configure gap pattern#2(that is different from gap pattern#1) toTGPS#2 of BTS#2	Only gap pattern #2 is available on both of BTS#1 and BTS#2.	Both of gap pattern#1 of TGPS#1 and gap pattern #2 of TGPS#2 on both of BTS#1 and BTS#2.

3.6.4 Compressed Mode Test sample scenario

Here describes attached compressed mode sample scenario.

term:

CM : Compressed Mode

NM : Normal Mode

TGCFN#n(n=1...6) : TGCFN specified on TGPS#n

3.6.4.1 MultiTGP

3.6.4.1.1 gap pattern activation scenario

Each sample scenario activates multiple TGPSs at single execution, and transmit or receive signal which has transmission gap.

(1) Stand-Alone DCCH

Use Start_SDCCH_CM14_MTGP.dll.

Refer "B.3.6 Stand-Alone setting" for transmission and reception of Stand-Alone DCCH.

Each Parameter for Gap Pattern

Parameter \ Pattern	Pattern14					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	7	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	0	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	3	4	5
UL method	SFR	SFR	SFR	SFR	SFR	SFR
DL method	SFR	SFR	SFR	SFR	SFR	SFR
Scrambling code change	No	No	No	No	No	No
Frame Type	A	A	A	A	A	A

Operation at scenario execution

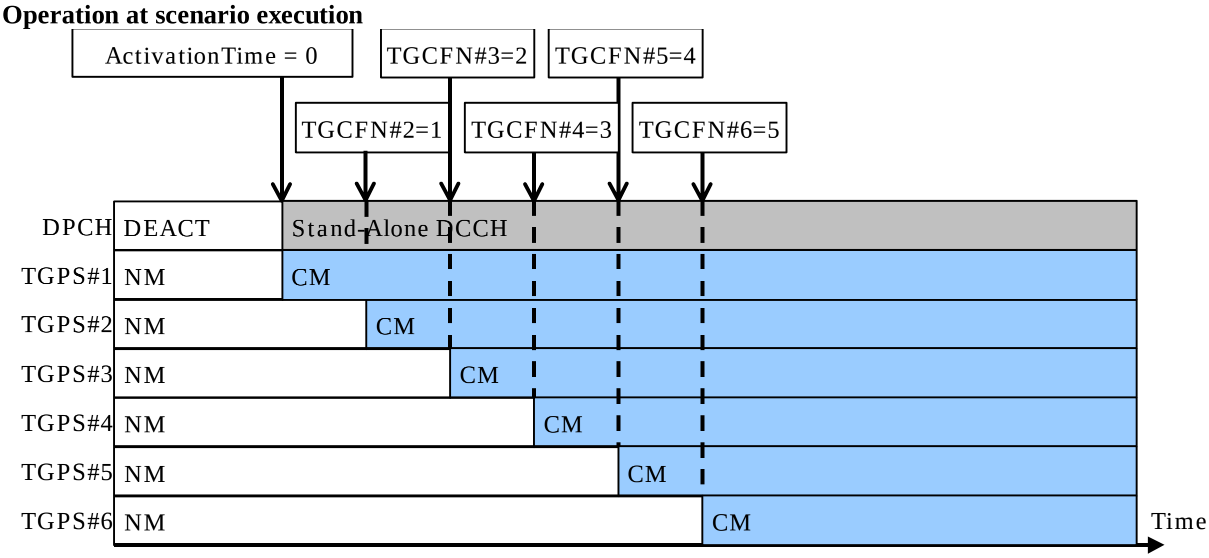


Figure3-15 Operation at execution of Start_SDCCH_CM14_MTGP.dll

(2) AMR 12.2[kbps]

Use Start_AMR_CM11_MTGP.dll.
Refer "B.3.9 AMR Voice(12.2[kbps])+DCCH Setting “ for transmission and reception of AMR.

Each Parameter for Gap Pattern

Parameter \ Pattern	Pattern11	
TGPSI	#1	#2
TGSN	11	11
TGL1	7	7
TGL2	-	-
TGD	4	4
TGPL1	0	0
TGPL2	-	-
TGPRC	∞	∞
TGCFN	0	2
UL method	SF/2	SF/2
DL method	SF/2	SF/2
Scrambling code change	No	No
Frame Type	A	A

Operation at scenario execution

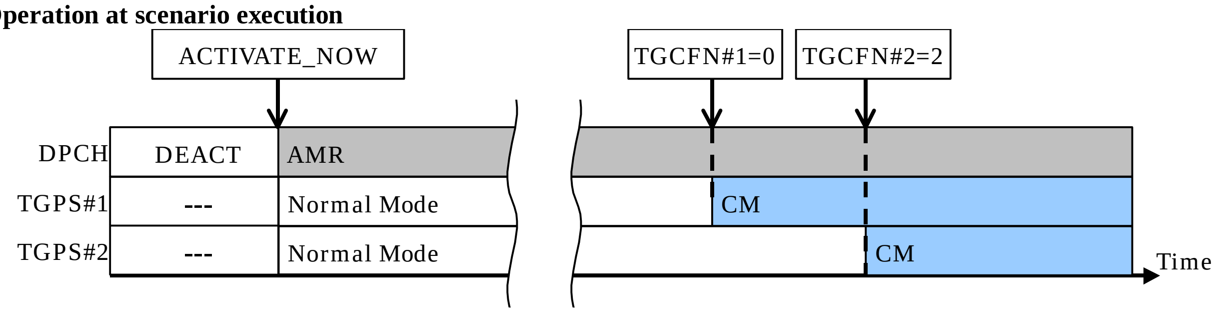


Figure 3-16 Operation at execution of Start_AMR_CM11_MTGP.dll

3.6.4.1.2 gap pattern suspend scenario

Here describes about scenario to suspend compressed mode using Activation Time when Stand-Alone DCCH is activated on DPCH.

Use sample scenario DEACT_AllTGPS_SDCCH.dll.This scenario suspends gap pattern of TGPS#1 and TGPS#2 that are activated with Stand-Alone DCCH.

Execute sample scenario Start_SDCCH_CM14_MTGP.dll(Please refer section 3.6.4.1.1 gap pattern activation scenario about transmission and reception). After that, execute DEACT_AllTGPS_SDCCH.dll. In this case, the operation is like below.

Operation at scenario execution

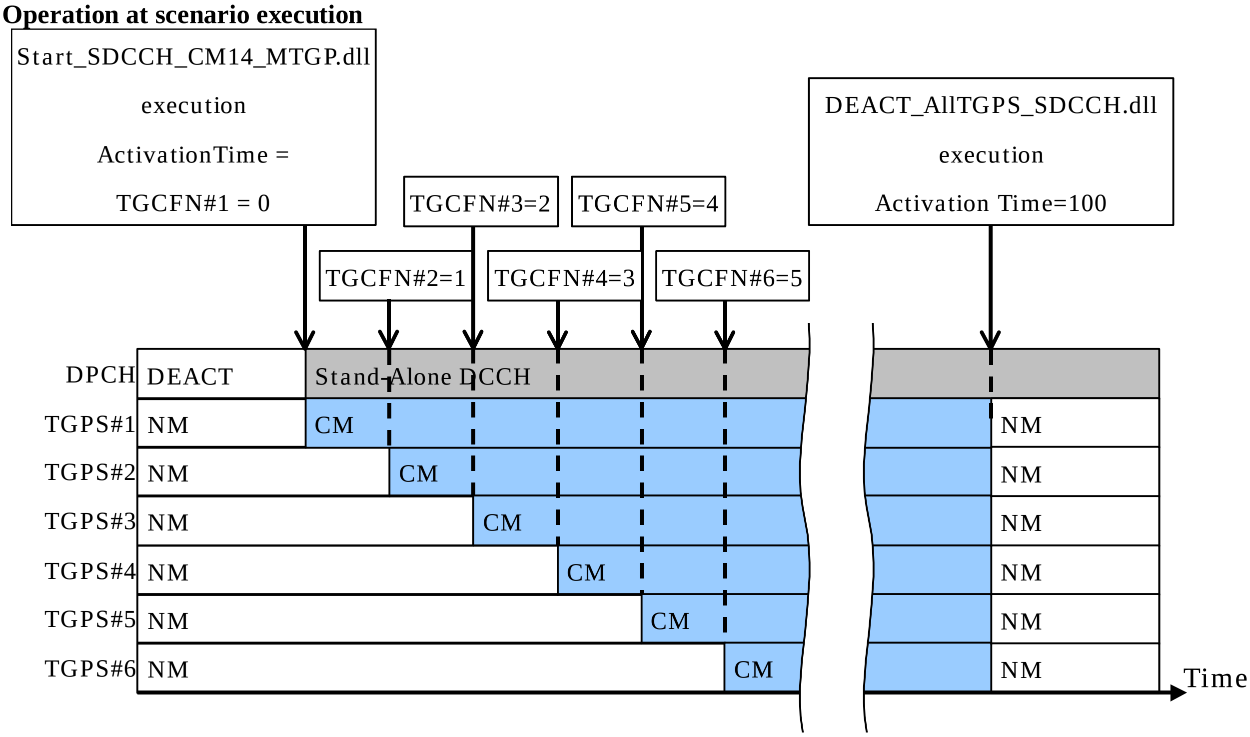


Figure3-17 Operation at execution of DEACT_AllTGPS_SDCCH.dll

3.6.4.1.3　gap pattern change scenario

Here describes scenario to change multiple TPGSs when AMR is activated.

1) Execute Start_AMR_CM11_MTGP.dll referring section 3.6.4.1.1 (2)AMR 12.2[kbps], and transmit or receive DPCH.
2) After 1), execute ACT_AMR_CM13_MTGP.dll and gap pattern of each TGPS changes.

Operation at scenario ACT_AMR_CM13_MTGP.dll execution

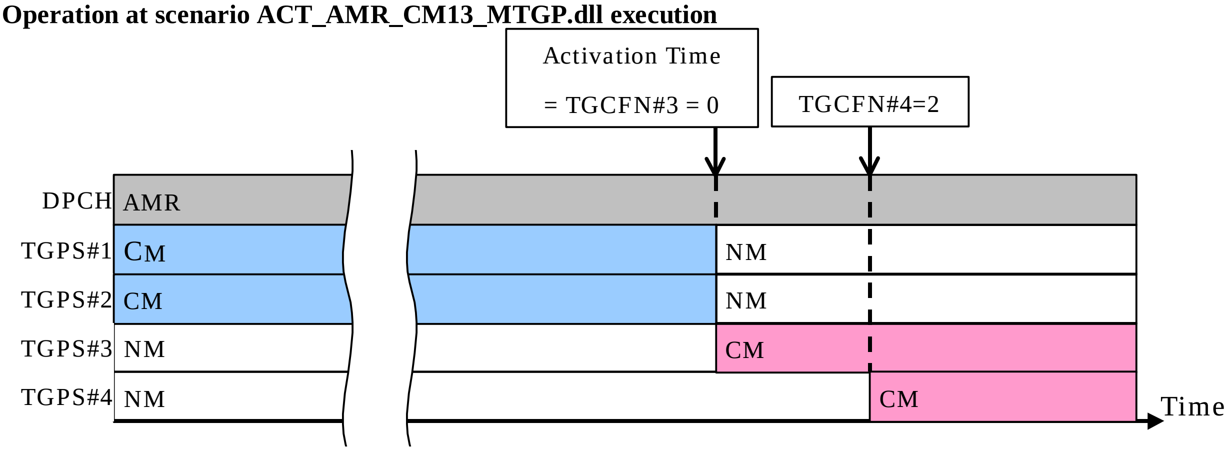


Figure3-18　Operation at scenario ACT_AMR_CM13_MTGP.dl execution

3.6.4.1.4 gap pattern keeping scenario

Here describes about scenario to keep(continue) gap pattern of part of TGPSs when changing bearer combination from AMR+P32K to AMR When keeping gap pattern, GapInfoFlag in structure CPHY_RL_SETUP_MTGP_PAR.
Use sample scenarios below.
Start_AMRxP32K_CM14_MTGP.dll
ACT_AMR_CM15_MTGP.dll

Please refer section B.3.18 AMR+Packet 32[kbps](DownlinkTTI=10[ms])+DCCH Setting for AMR+Packet32[kbps]+DCCH transmission and reception, and section B.3.9 AMR Voice(12.2[kbps])+DCCH Setting for AMR.

Each Parameter for Gap Pattern

Pattern / Parameter	Pattern14					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	7	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	0	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	3	4	5
UL method	SFR	SFR	SFR	SFR	SFR	SFR
DL method	SFR	SFR	SFR	SFR	SFR	SFR
Scrambling code change	No	No	No	No	No	No
Frame Type	A	A	A	A	A	A

Pattern / Parameter	Pattern15					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	4	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	45	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	4	5	6
UL method	SF/2	SF/2	SF/2	SF/2	SF/2	SF/2
DL method	SF/2	SF/2	SF/2	SF/2	SF/2	SF/2
Scrambling code change	No	No	No	No	No	No
Frame Type	A	A	A	A	A	A

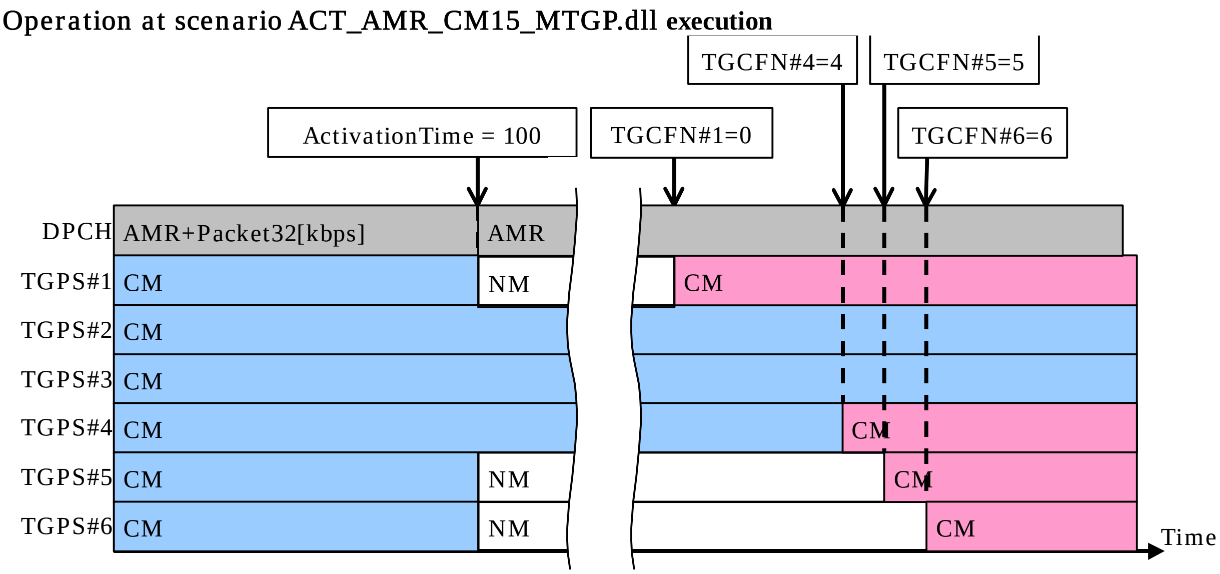


Figure 3-19 Operation at scenario ACT_AMR_CM15_MTGP.dll execution

3.6.4.2 SingleTGP

3.6.4.2.1 gap pattern activation scenario

Each sample scenario activates one TGPS one by one, and transmit or receive signal which has transmission gap. They activate RMC12.2K at Activation Time=100, and activate compressed mode at specified TGCFN.

(1) Reference Measurement Channel(12.2[kbps]), single TGPS

There are 3 sample scenario . Each has different gap pattern.

Pattern7...Use Start_RMC12_2K_CM7.dll

Pattern 8...Use Start_RMC12_2K_CM8.dll

Pattern 10...Use Start_RMC12_2K_CM10.dll

Please refer section B.3.22 Reference Measurement Channel(12.2[kbps]) setting for transmission and reception of RMC12.2[kbps].

Each Parameter for Gap Pattern

Pattern / Parameter	Pattern7	Pattern8	Pattern 10
TGPSI	**#1**	**#1**	**#1**
TGSN	8	10	4
TGL1	14	10	7
TGL2	7	5	-
TGD	22	20	0
TGPL1	5	3	1
TGPL2	3	5	-
TGPRC	∞	∞	∞
TGCFN	3	1	0
UL method	SF/2	SF/2	SF/2
DL method	SF/2	Punc	SF/2
Scrambling code change	Yes	No	No
Frame Type	A	B	A

Operation at scenario execution

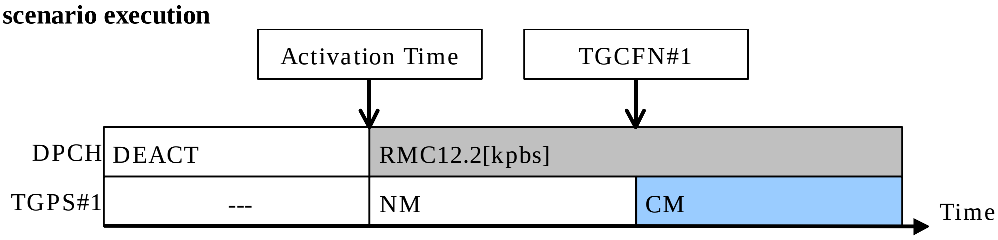


Figure 3-20 Operation at activation of one TGPS using SingleTGP

(2) Reference Measurement Channel(12.2[kbps]), multiple TGPS

There are 2 sample scenarios. They have difference gap pattern.

Pattern11...Use Start_RMC12_2K_CM11.dll

Pattern 12...Use Start_RMC12_2K_CM12.dll

Please refer section B.3.22 Reference Measurement Channel(12.2[kbps]) setting about transmission and reception of RMC12.2[kbps].

Each Parameter for Gap Pattern

Pattern / Parameter	Pattern11		Pattern12	
TGPSI	#1	#2	#1	#2
TGSN	11	11	4	4
TGL1	7	7	7	7
TGL2	-	-	-	-
TGD	4	4	0	0
TGPL1	0	0	8	8
TGPL2	-	-	-	-
TGPRC	∞	∞	∞	∞
TGCFN	0	2	0	4
UL method	SF/2	SF/2	SF/2	SF/2
DL method	SF/2	SF/2	Punc	Punc
Scrambling code change	No	No	No	No
Frame Type	A	A	A	A

Operation at scenario execution

SingleTGP needs to be executed twice when configuring 2 TGPSs.

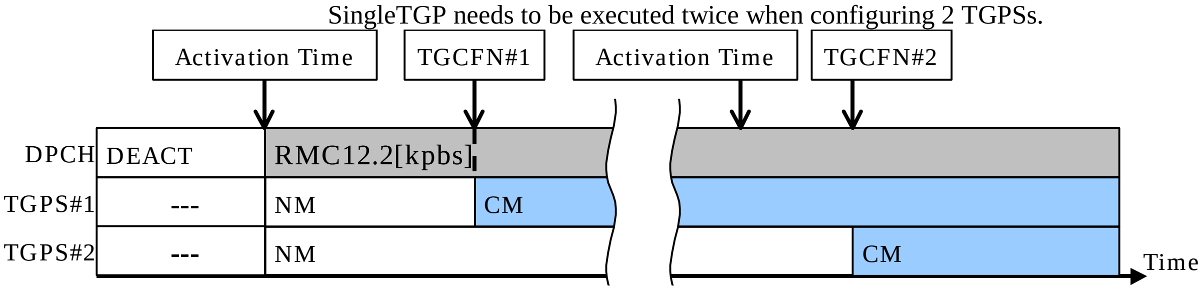


Figure 3-21　Operation when activating multiple TGPS using SingleTGP

(3) Packet 64[kbps](downlink TTI=10[ms]) +DCCH, one TGPS

There are 2 sample scenarios. They have different gap pattern.

Pattern5...Use Start_P64K(TTI10m)_CM5.dll

Pattern 6...Use Start_P64K(TTI10m)_CM6.dll

Please refer section B.3.13 Packet 64[kbps](downlink TTI=10[ms]) +DCCH setting about transmission and reception of Packet64[kbps](TTI=10[ms])

Each Parameter for Gap Pattern

Pattern / Parameter	Pattern5	Pattern6
TGPSI	#1	#1
TGSN	12	12
TGL1	5	5
TGL2	-	3
TGD	0	21
TGPL1	2	4
TGPL2	-	12
TGPRC	∞	∞
TGCFN	0	0
UL method	HLS	HLS
DL method	HLS	HLS
Scrambling code change	No	No
Frame Type	A	A

Operation at scenario execution

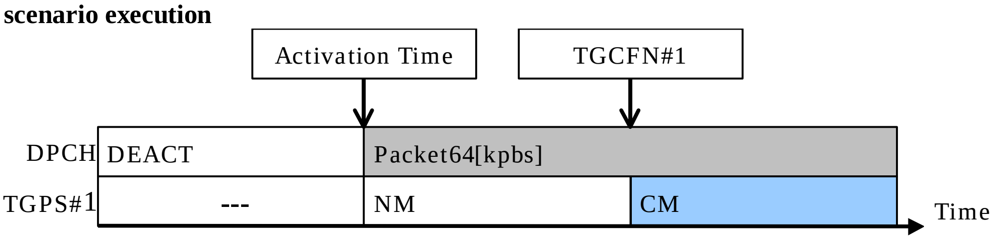


Figure3-22 Operation at execution of Start_P64K(TTI10m)_CMx.dll

(4) Packet 64[kbps](downlink TTI=10[ms]) +DCCH, multiple TGPS

Use Start_P64K(TTI10m)_CM13.dll.

Please refer section B.3.13 Packet 64[kbps](downlink TTI=10[ms]) +DCCH setting about transmission and reception of Packet64[kbps](TTI=10[ms])

Each Parameter for Gap Pattern

Parameter \ Pattern	Pattern13	
TGPSI	#1	#2
TGSN	4	11
TGL1	5	7
TGL2	-	-
TGD	0	0
TGPL1	8	4
TGPL2	-	-
TGPRC	∞	∞
TGCFN	0	2
UL method	HLS	HLS
DL method	HLS	HLS
Scrambling code change	No	No
Frame Type	A	A

Operation at scenario execution

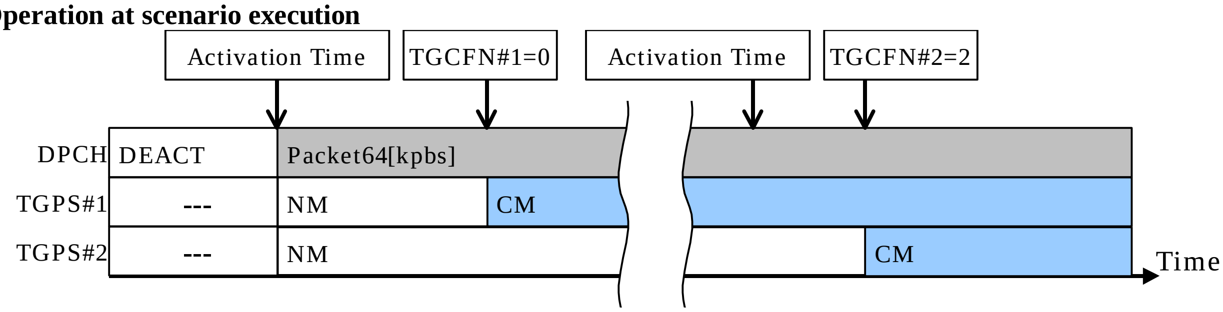


Figure 3-23　Operation at execution of Start_P64K(TTI10m)_CM13.dll

(5) Reference Measurement Channel(144[kbps]), single TGPS

There are 2 sample scenarios. They have different gap pattern.

Pattern1...Use Start_RMC144K_CM1.dll

Pattern2...Use Start_RMC144K_CM2.dll

Please refer section B.3.24 Reference Measurement Channel(144[kbps]) setting about transmission and reception of Reference Measurement Channel(144[kbps]).

Each Parameter for Gap Pattern

Pattern Parameter	Pattern1	Pattern2
TGPSI	#1	#1
TGSN	11	11
TGL1	7	7
TGL2	-	-
TGD	0	0
TGPL1	2	4
TGPL2	-	-
TGPRC	∞	∞
TGCFN	0	0
UL method	SF/2	SF/2
DL method	SF/2	Punc
Scrambling code change	No	No
Frame Type	A	A

Operation at scenario execution

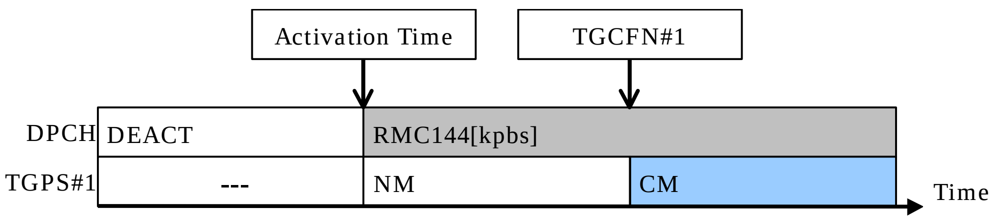


Figure 3-24　Operation at execution of Start_RMC144K_CMx.dll

(6) Reference Measurement Channel(144[kbps]), multiple TGPS

There are 2 sample scenarios and they have different gap pattern.

Pattern14...Use Start_RMC144K_CM14.dll

Pattern15...Use Start_RMC144K_CM15.dll

Please refer section B.3.24 Reference Measurement Channel(144[kbps]) setting about transmission and reception of Reference Measurement Channel(144[kbps])

Each Parameter for Gap Pattern

Pattern / Parameter	Pattern14					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	7	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	0	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	3	4	5
UL method	SF/2	SF/2	SF/2	SF/2	SF/2	SF/2
DL method	SF/2	SF/2	SF/2	SF/2	SF/2	SF/2
Scrambling code change	No	No	No	No	No	No
Frame Type	A	A	A	A	A	A

Pattern / Parameter	Pattern15					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	4	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	45	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	4	5	6
UL method	SF/2	SF/2	SF/2	SF/2	SF/2	SF/2
DL method	Punc	SF/2	SF/2	SF/2	SF/2	SF/2
Scrambling code change	No	No	No	No	No	No
Frame Type	A	A	A	A	A	A

Operation at scenario execution

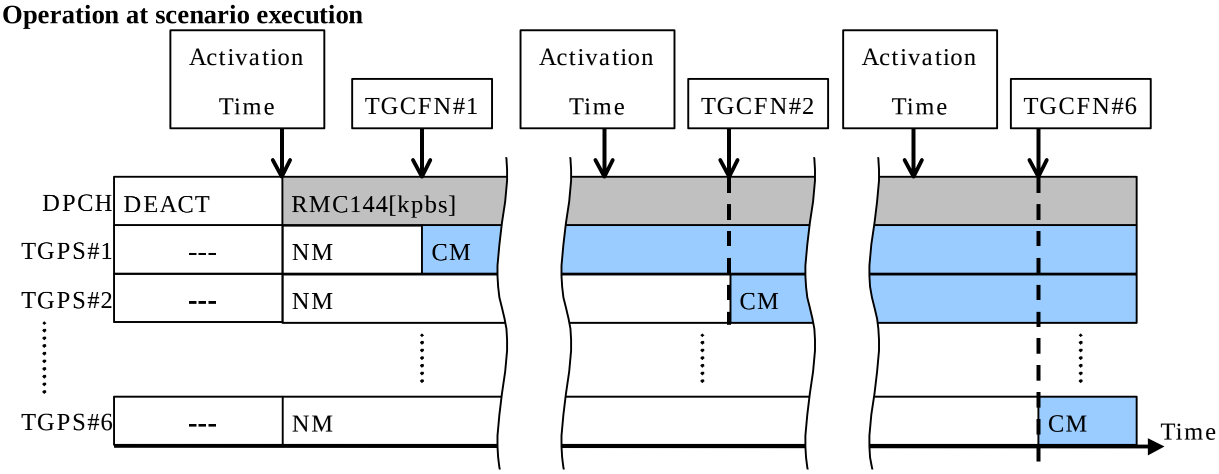


Figure 3-25 Operation at execution of Start_RMC144K_CM1x.dll

(7) Packet 128[kbps] +DCCH, one TGPS

Use Start_P128K_CM5.dll.

Please refer B.3.14 Packet 128[kbps] (downlinkTTI=10[ms]) +DCCH setting about transmission and reception of Packet 128[kbps] +DCCH.

Each Parameter for Gap Pattern

Parameter \ Pattern	Pattern5a
TGPSI	#1
TGSN	12
TGL1	5
TGL2	-
TGD	0
TGPL1	2
TGPL2	-
TGPRC	∞
TGCFN	0
UL method	HLS
DL method	HLS
Scrambling code change	No
Frame Type	A

Operation at scenario execution

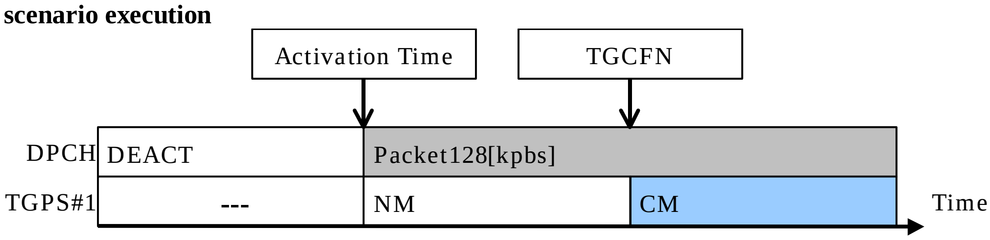


Figure3-26 Operation at execution of Start_P128K_CM5.dll

(8) Packet 128[kbps] +DCCH, one TGPS

There are 2 sample scenarios and they have different gap patterns.

Pattern14...Use Start_P128K_CM14.dll

Pattern15...Use Start_P128K_CM15.dll

Each Parameter for Gap Pattern

Pattern / Parameter	Pattern14					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	7	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	0	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	3	4	5
UL method	HLS	HLS	HLS	HLS	HLS	HLS
DL method	HLS	HLS	HLS	HLS	HLS	HLS
Scrambling code change	No	No	No	No	No	No
Frame Type	A	A	A	A	A	A

Pattern / Parameter	Pattern15					
TGPSI	#1	#2	#3	#4	#5	#6
TGSN	4	4	4	4	4	4
TGL1	4	7	7	7	7	7
TGL2	-	-	-	-	-	-
TGD	45	0	0	0	0	0
TGPL1	8	8	8	8	8	8
TGPL2	-	-	-	-	-	-
TGPRC	∞	∞	∞	∞	∞	∞
TGCFN	0	1	2	4	5	6
UL method	HLS	SF/2	SF/2	SF/2	SF/2	SF/2
DL method	HLS	SF/2	SF/2	SF/2	SF/2	SF/2
Scrambling code change	No	Code Change (Right)	No	No	No	No
Frame Type	A	A	A	A	A	A

Operation at scenario execution

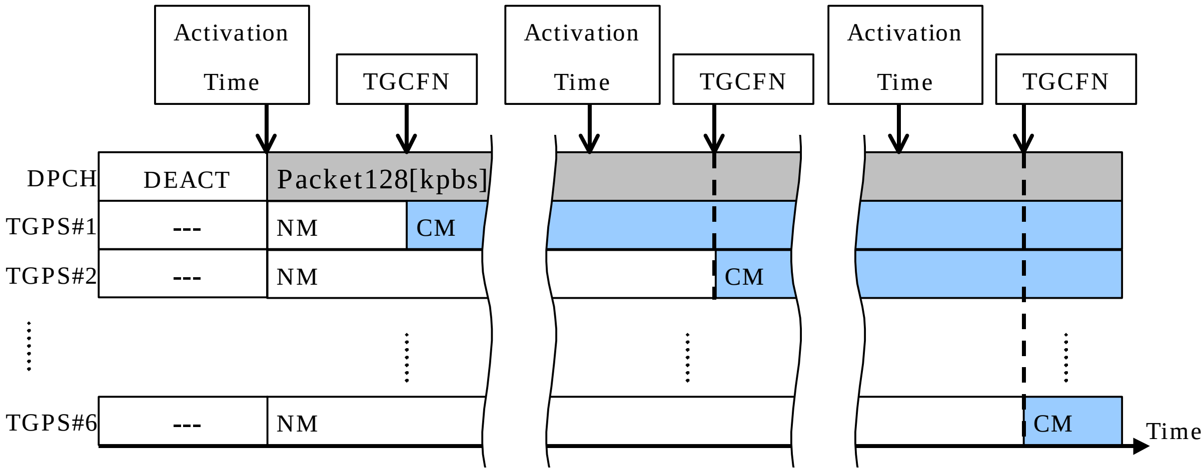


Figure 3-27 Operation at execution of Start_P128K_CM1x.dll

3.6.4.2.2　gap pattern suspend scenario

Here describes about scenario to suspend compressed mode using Activation Time when running UDI on DPCH.

Use sample scenario Act_UDI_CM1.dll and Act_UDI_NM.dll . Act_UDI_NM.dll

suspends gap pattern of TGPS#1 on UDI+DCCH.
Please refer section.6.4.2.1(5) pattern 1 about gap pattern.
The operation after execution of Start_UDI.dll and Act_UDI_CM1.dll , Act_UDI_NM .dll (with this sequence) is below.

Operation at scenario Act_UDI_NM.dll execution

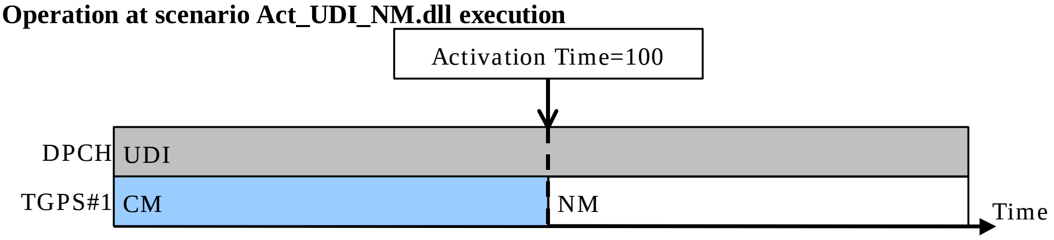


Figure 3-28　Operation to suspend gap pattern using SingleTGP

3.6.4.2.3　gap pattern suspend scenario

Here describes about scenario to change gap pattern using Activation Time and TGCFN when RMC12.2[kbps] running on DPCH.

Use sample scenario Start_RMC12_2K_CM10.dll and ACT_RMC12_2K_CM4.dll.

Please refer section 3.6.4.2.1 gap pattern activation scenario.

1) Execute Start_RMC12_2K_CM10.dll

2) Execute ACT_RMC12_2K_CM4.dll

After that, operation is as follows.

Please refer section 3.6.4.2.1 (1) pattern 1 about gap pattern before change.

Each Parameter for Gap Pattern after change

Parameter \ Pattern	Pattern 5
TGPSI	#1
TGSN	4
TGL1	7
TGL2	-
TGD	0
TGPL1	4
TGPL2	-
TGPRC	∞
TGCFN	100
UL method	SF/2
DL method	SF/2
Scrambling code change	No
Frame Type	A

Operation at scenario ACT_RMC12_2K_CM4.dll execution

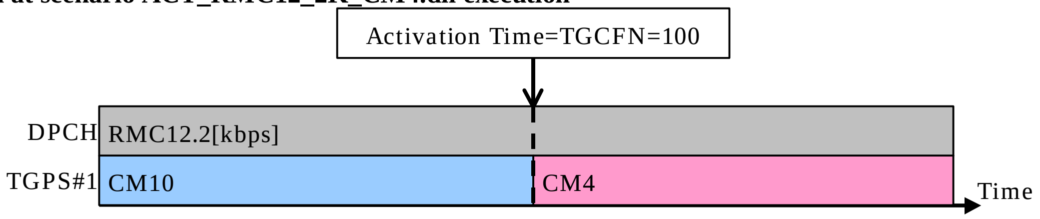


Figure 3-29　Operation to change gap pattern using SingleTGP

3.6.4.2.4 gap pattern keeping scenario

Here describes about scenario to keep(continue) gap pattern when changing bearer combination from Packet32K to Packet64K. When keeping gap pattern, use GapInfoFlag in structure CPHY_RL_SETUP_PAR. This functionality is effective when only one TGPS is used when using SingleTGP.

Use sample scenarios below.

Start_P32K_CM9.dll

ACT_P64K_CM9_KEEP.dll

Please refer section B.3.12 Packet 32[kbps](downlink TTI=20[ms]) +DCCH setting and section B.3.13 Packet 64[kbps](DownlinkTTI=10[ms]) +DCCH Setting about transmission and reception of Packet32[kbps]+DCCH and Packet64[kbps]+DCCH.

1) Execute Start_P32K_CM9.dll and transmit and/or receive DPCH.
2) After 1), execute ACT_P64K_CM9_KEEP.dll. And bearer combination changes from Packet32[kbps]+DCCH to Packet64[kbps]+DCCH at TGCFN=0. But gap pattern doesn' t change.

Each Parameter for Gap Pattern

Parameter \ Pattern	Pattern9
TGPSI	#1
TGSN	9
TGL1	10
TGL2	5
TGD	3
TGPL1	5
TGPL2	21
TGPRC	∞
TGCFN	0
UL method	HLS
DL method	HLS
Scrambling code change	No
Frame Type	B

Operation at scenario execution

This figure shows the operation when ACT_P64K_CM9_KEEP.dll is executed. DPCH changes from Packet32[kbps] to Packet64[kbps].

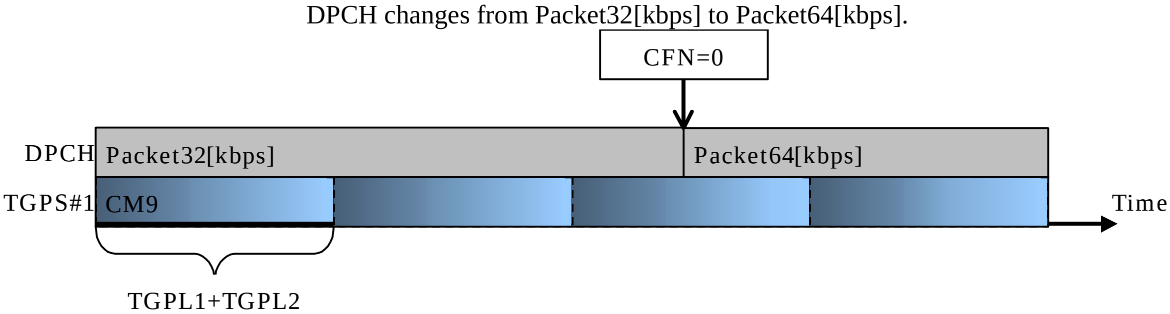


Figure 3-30 Operation to keep gap pattern using SingleTGP

3.6.4.3 The way to use corresponds to TS25.331 message

Here describes about an example to realize Information elements about compressed mode in 3GPP specification TS25.331 with MD8480A/B. MultiTGP is used.
Table below describes correspondence between main Information Element/Group name and MD8480A/B parameters.
It uses CPHY_RL_SETUP_MTGP_PARCphyRlSetupMtgpD_DPCH which is defined with structure for MultiTGP.

3.6.4.3.1 DPCH Compressed Mode Info

1) TGPSI

Value to set to message	Value to set to MD8480A/B
n(1...6)	Configure TGCFN and others using CphyRlSet-upMtgpD_DPCH.GapInfo[n-1]
* For TGPS#m which is not set to TGPSI of 3GPP message	CphyRlSetupMtgpD_DPCH.GapInfoFlag[m-1] = GAP_KEEP_ON;

2) TGPS Status Flag

When setting "n" to TGPSI of 3GPP message :

Value to set to message	Value to set to MD8480A/B
activate	CphyRlSetupMtgpD_DPCH.GapInfoFlag[n-1] = GAP_KEEP_OFF; CphyRlSetupMtgpD_DPCH.GapInfo[n-1].Method = CM_MODE_SFR; (or CM_MODE_PUNC, CM_MODE_PUNC)
deactivate	CphyRlSetupMtgpD_DPCH.GapInfoFlag[n-1] = GAP_KEEP_OFF; CphyRlSetupMtgpD_DPCH.GapInfo[n-1].Method = CM_MODE_NONE;

3) TGCFN

When setting "n" to TGPSI of 3GPP message :

Value to set to message	Value to set to MD8480A/B
x(0...255)	CphyRlSetupMtgpD_DPCH.GapInfo[n-1].TGCFN = x;

4) Caution

-TGSN and other parameters are needed to be set.

-INVALID_PARAM should be set to CphyRlSet-upMtgpD_DPCH.TGPS_Reconfig_CFN.

3.6.4.3.2 MEASUREMENT CONTROL

1) TGPS reconfiguration CFN

Value to set to message	Value to set to MD8480A/B
y(0-255)	CphyRlSetupMtgpD_DPCH.TGPS_Reconfig_CFN = y; Set "y" to Activation Time of CphyRlSetupMtgp()

2) TGPSI

Value to set to message	Value to set to MD8480A/B
n(1...6)	Set TGCFN and others to CphyRlSetupMtgpD_DPCH.GapInfo[n-1]
* For TGPS#m which is not set to TGPSI of 3GPP message	CphyRlSetupMtgpD_DPCH.GapInfoFlag[m-1] = GAP_KEEP_ON;

3) TGPS Status Flag

When setting "n" to TGPSI of 3GPP message :

Value to set to message	Value to set to MD8480A/B
activate	CphyRlSetupMtgpD_DPCH.GapInfoFlag[n-1] = GAP_KEEP_OFF; CphyRlSetupMtgpD_DPCH.GapInfo[n-1].Method = CM_MODE_SFR; (or CM_MODE_PUNC, CM_MODE_PUNC)
deactivate	CphyRlSetupMtgpD_DPCH.GapInfoFlag[n-1] = GAP_KEEP_OFF; CphyRlSetupMtgpD_DPCH.GapInfo[n-1].Method = CM_MODE_NONE;

4) TGCFN

When setting "n" to TGPSI of 3GPP message :

Value to set to message	Value to set to MD8480A/B
x(0...255)	CphyRlSetupMtgpD_DPCH.GapInfo[n-1].TGCFN = x;

5) Caution

3GPP "MEASUREMET CONTROL" doesn't require TGSN and others, but gap pattern parameters should be set up whenever CphyRlSetupMtgp() is executed.

3.7　Other Tests

3.7.1　BLER and BTFD Measurement feature

The BtsMeasure()(refer to section A) performs BLER and BTFD measurements. BtsMeasure() performs the measurement specified under 3GPP TS34.109 and TS34.121. BLER measurement has two ways. One is using CRC, the other is using Re-Transmission.

3.7.1.1　BLER (CRC) measurement feature

This feature performs measurement of the 3GPP TS34.109 A.3 Measurement of receiver performance (BLER) using UE test loop mode 2.
The Signalling Tester determines OK or NG of CRC and returns a value corresponding to OK or NG.　Determination of CRC is made as follows:

1) CRC determination is made per Transport Block Set of Transport Channel.　Even in case a single TTI has plural transport blocks, the single TTI is counted as one.
2) In case a pilot is not received successfully (synchronization failure), CRC determination is not made.
3) When the CRC determination count has exceeded the value specified in Count or a time-out has taken place, the BtsMeasure() is terminated.

In the sample scenario MeasBLER(CRC).dll, it is measured in one thousand of times of TTIs. The number of the measurement (Total Count), the number of CRC=NG (CRC NG), and BLER (BLER) are displayed.

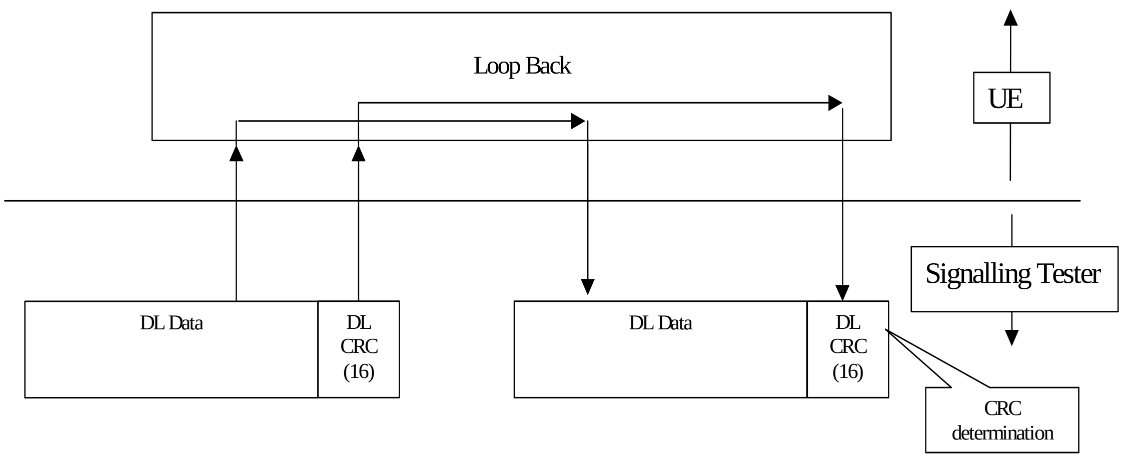


Figure 3-31 BLER (CRC) measurement

3.7.1.2 BLER (RE-TRANSMISSION) measurement feature

This feature performs measurement of the 3GPP TS34.109 A.2 Measurement of receiver performance (BLER) using UE test loop mode 1 and RLC AM.

The MD8480A/B counts the number of PUs in the STATUS PDUs sent from a mobile station for which retransmission requests have been issued.

When the number of transmitted PUs has exceeded the value specified in Count or a time-out has taken place, this function is terminated.

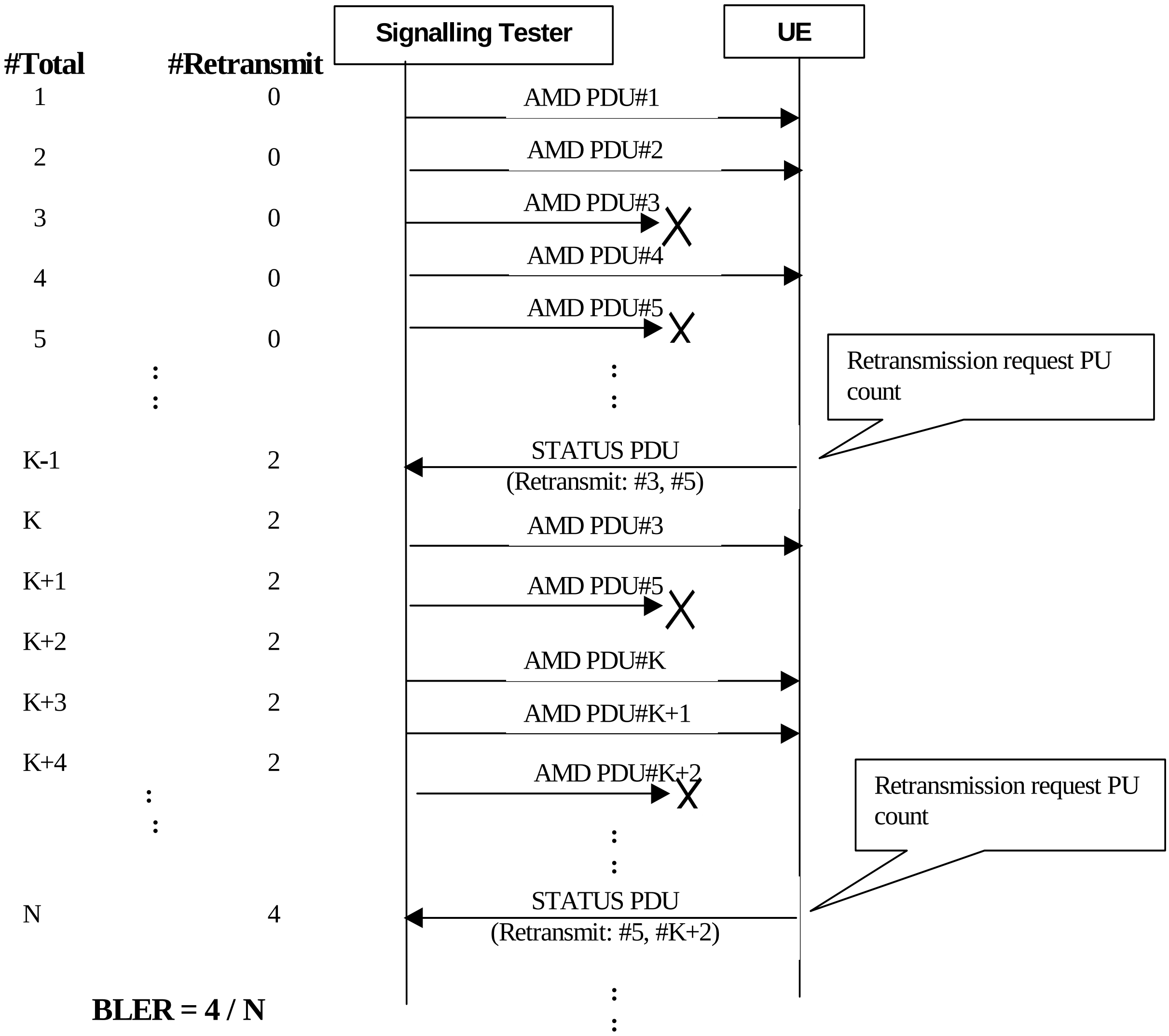


Figure 3-32 BLER (Re-transmission) measurement

In the sample scenario MeasBLER(RETRANS.dll, it is measured in one thousand of pieces of PUs. The number of the measurement (Total Count), the number of PU required to re-transmit (ReTrans Request), and BLER (BLER) are displayed

3.7.1.3 BTFD measurement feature

The Signalling Tester determines OK/NG of CRC on the uplink data and accord/discord of TFCI, then returns corresponding values. The determination is made in the following way:

1) Determination of CRC and TFCI is made per Transport Block Set of Transport Channel.
   Even in case a single TTI has plural transport blocks, the single TTI is counted as one.

2) In case a pilot is not received successfully (synchronization failure), CRC determination is not made.

3) When the CRC determination count has exceeded the value specified in Count or a time-out has taken place, this function is terminated.

The measurement results are returned using the four values,

CRC=OK,TFCI=OK
CRC=OK,TFCI=NG
CRC=NG,TFCI=OK
CRC=NG,TFCI=NG

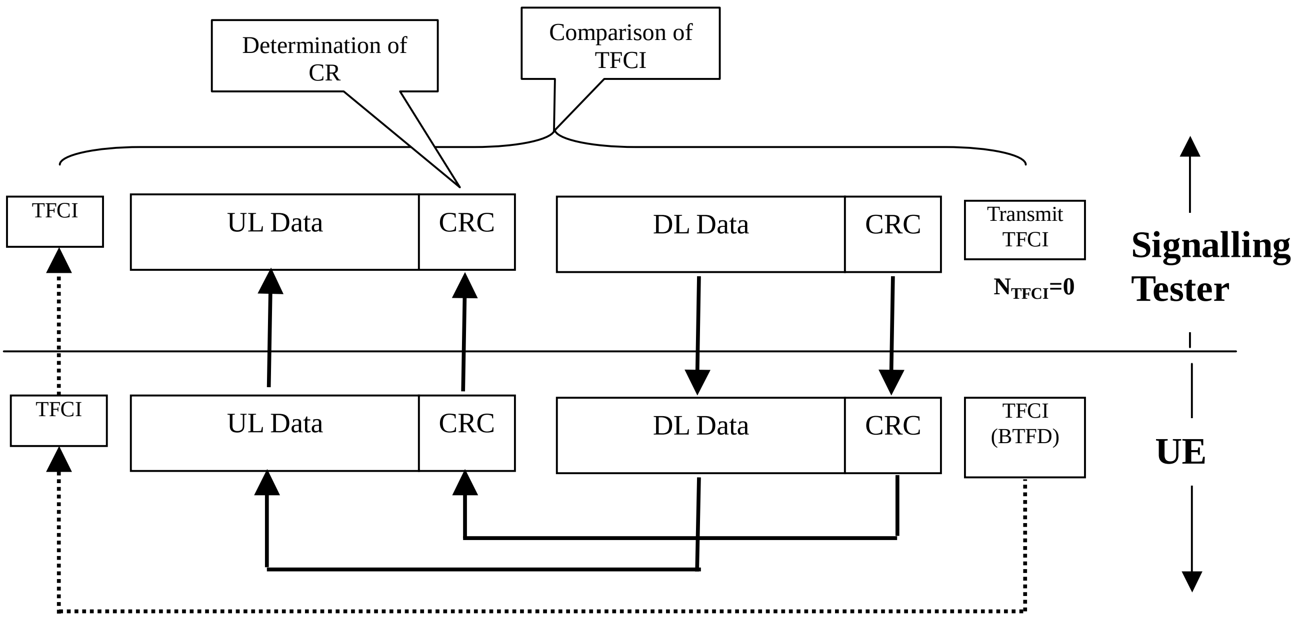


Figure 3-33 BTFD measurement

In the sample scenario MeasBTFD.dll, it is measured in one thousand of times of TTIs waiting for TFCI = 3, and the following items are displayed.

Items		Contents
Total Count		The number of the measurement
FDR		The ratio of “ Not TFCI=3” and “ CRC=OK” .
BLER		The ratio of “ CRC=NG” regardless of TFCI.
CRC OK	TFCI OK	The ratio of “ TFCI=3” and “ CRC=OK” .
	TFCI NG	The ratio of “ Not TFCI=3” and “ CRC=OK” .
CRC NG	TFCI OK	The ratio of “ TFCI=3” and “ CRC=NG” .
	TFCI NG	The ratio of “ Not TFCI=3” and “ CRC=NG” .

3.7.2　AWGN

3.7.2.1　Test Method

The following shows the transmission method for AWGN.

1. Execute Start_*.dll (for example Start_UDI.dll).
2. Execute Act_AWGN .dll(C¥: Mx848000 ¥ Scenario ¥ Src ¥Act_AWGN.c).

The above procedure activates AWGN.

3.7.2.2　Output level

The AWGN power is specified as a power in the 3.84 MHz bandwidth.
The AWGN power on the RF signal output (Main/Downlink) is determined by the setting of two items on the Parameter Setup screen, that is, AWGN Power[-99 to -8 dBm] and Tx Attenuator values on the Tx Setup screen. These set values provides the following AWGN output power:

(AWGN output power) = [-99 to -8 dBm] - TxAttenuator [dBm/3.84 MHz]

Here, a set value of [-99 to -8 dBm] determines the Baseband signal level.
Setting of TxAttenuator determines the ATT (Attenuator) value in the RF section.

Restriction:

The AWGN that is generated by the MD8480A/B does not meet the 3GPP specification with regard to the performance. According to the 3GPP specification, the AWGN is required to fulfill the following specification. "The flatness across this minimum bandwidth (e.g. 5.76 MHz for a chip rate of 3.84 Mcps) shall be less than +/-0.5 dB and the peak to average ratio at a probability of 0.001 % shall exceed 10 dB." But the flatness across the 5.76MHz of the AWGN from the MD8480A/B

is about 2dB. And the peak to average ratio of the AWGN from MD8480A/B has not been estimated.

3.7.3 OCNS Test

MD8480B can output OCNS(Orthogonal Channel Noise Simulator) signal that has been spread and multiplexed by different Channelization Code of 16 channels in Down Link output(Refer to 25.101 for the specification of OCNS in 3GPP.)
Possible OCNS to be output by MD8480B is as follows,

• OCNS Output Level

It is transmittable when the sum of 16 of multiplexed signal is between 0 [dBm] at maximum and –99 [dBm] at minimum.

• The number of OCNS output

Possible OCNS to be outputs are two.

In order to make OCNS for BTS#2 output, “ 2nd OCNS(MU848061B)” is required.

• Channelization Code and relative level
It transmits the following that is provided in the 3GPP Specification [TS25.101 Table C.6].

Channelization Code	Relative Level
2	-1dB
11	-3 dB
17	-3 dB
23	-5 dB
31	-2 dB
38	-4 dB
47	-8 dB
55	-7 dB
62	-4 dB
69	-6 dB
78	-5 dB
85	-9 dB
94	-10 dB
125	-8 dB
113	-6 dB
119	-0 dB

• PN9 is used for the symbol pattern of 16 channels that comprise the OCNS. PN9 default values for each 16 channel can be set.

• Power Control operation

You can change the OCNS output level according to the specification of TPC Bit of uplink DPCH*1. Variable steps of downlink DPCH(DPCCH) output level are 0.5, 1.0, 1.5 and 2.0 [dB]. STEP width of OCNS is not any fixed values such as " ~dB" .As the power of sum of downlink DPCH(DPCCH) and OCNS will be fixed, OCNS will increases as much as downlink DPCH (DPCCH)power decreases and will decreases as less as it increases. Also you can control the OCNS power to make the Total Power fixed after changing downlink DPCH(DPCCH) power as much as you specified from the scenario.

*1 OCNS Power Control will synchronize with downlink DPCH(DPCCH) output signal of BTS#1 or BTS#2 and operates as the sum of output of downlink DPCH(DPCCH) and OCNS will be fixed. For Downlink DPCH(DPDCH)Power of BTS#1 , set the same setting as that of downlink DPCH(DPCCH)Power of BTS#1, and for Downlink DPCH(DPDCH)Power of BTS#2, set the same setting as that of Downlink DPCH(DPCCH)Power of BTS#2. The operation of OCNS Power Control synchronized with BTS#3 has not been supported.

3.7.3.1 Scenario library specifications

Items to notice:
Configure the settings of scenarios, when using the OCNS scenario library, as follows:

Setting of scenario
• To make the Total Power for DPCH(DPCCH)Power and OCNS Power fixed, set DPCH(DPCCH)Max/Min Power referring to the example below before the function CphyRlSetup() which sets Down Link DPCH(DPCCH).

example)

```
CphyRlSetupD_DPCH.MaxDLPower = -10;
                /* DL DPCH MAX Power -10dBm */
CphyRlSetupD_DPCH.MinDLPower = -73;
                /* DL DPCH MIN Power -73dBm */
```

3.7.3.2 How to use sample scenarios

OCNS can be transmitted with the following procedures. This describes the procedures up to activating OCNS using an example of AMR.

• Activation of OCNS and DPCH

1) On Parameter Setup window, set DPCH to the same as that of AMR.

2) Activate DPCH (AMR) using Start_AMR.dll.

3) Select OCNS activation scenario (OCNS_ACT.dll) and execute it. Executing OCNS_Act.dll allows OCNS signal to be output in approx. 3 seconds.

• Starting Power Control

1) Select OCNS_Power.dll and execute it.
2) Select “ Start” when “ TPC Power Control?" is displayed.

With the procedures above, Power Control for DPCH and OCNS are activated and operates to fix the sum of the signal of DPCH and OCNS.

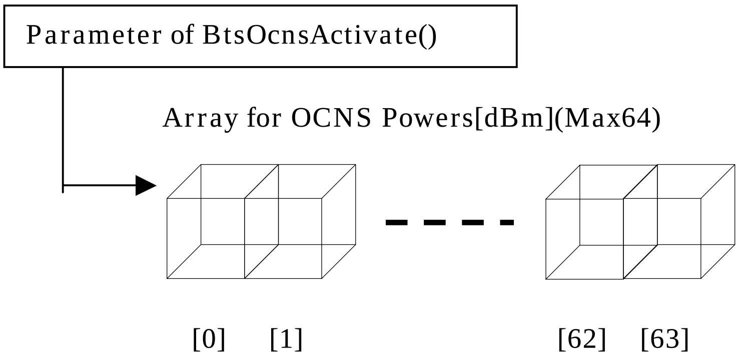
3.7.3.3 Items to notice

Power Control does not operate properly unless the relations of power between OCNS and DPCH meet the following formula. Please use it within the range to meet this relation.

$$1 - 10^{(PowerDPCH\ -\ Power\ OCNS)/10} \times (10^{PowerSTEP/10} - 1) > 0$$

PowerDPCH: default value of DPCH power [dBm]
PowerOCNS: default value of OCNS power [dBm]
PowerSTEP: changed amount from the default value of DPCH power [dB]

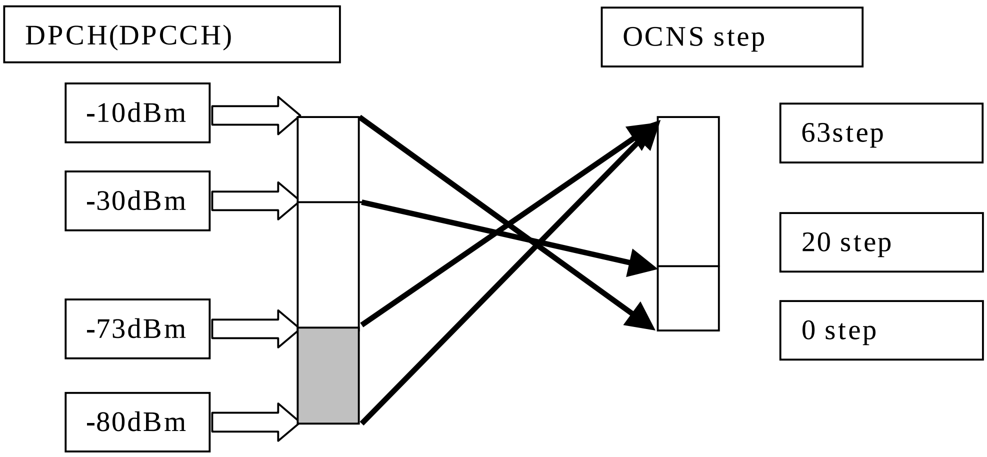
The power control for OCNS is limited to 64 steps, which is because Power control is conducted by storing OCNS Power to 64 or less arrangements due to MD8480B' s structure.



Consequently, some caution is necessary in setting a DPCH(DPCCH) Max/Min Power when you control the Power maintaining Total Power for DPCH(DPCCH)Power and OCNS Power fixed.
For example, if you set a parameter of CPHY_OCNS_ACT_PAR as follows:

MaxDLPower	-10dBm
DpchDpcchPower	-30dBm
MinDLPower	-80dBm
DpchPowerStep	1.0dB

There should be 71 steps from lower limit to upper limit of DPCH(DPCCH). However, the maximum number of step for OCNS is 64, and consequently the OCNS will reach to the upper limit when DPCH(DPCCH) reduces to -73dBm or less. And Total Power does not remain fixed. The followings are the images.



OCNS signal is output approximately 3 seconds after a completion of the execution of function BtsOcnsAcrivate(). Do not use any of function BtsOcnsActivate(), BtsOcnsDeactivate(), BtsOcnsPower(), or BtsOcnsPowerActTime() before OCNS is output. Which may cause the Signalling Tester to operate unstably.

Activate and leave Uplink/Downlink DPCH ON before executing function BtsOcnsPower() or BtsOcnsPowerActTime().

Execute function BtsOcnsPower() or BtsOcnsPowerActTime() after executing function BtsOcnsActivate().

To precisely conduct Power Control, do not change DPCH(DPCCH) Power using the function BtsPower() and so on, after the run of the function BtsOcnsActivate().

Do not use function BtsOcnsActivate() while output level is being changed by using function BtsOcnsPower() or BtsOcnsPowerActTime() following the direction of TPC Bit of Uplink DPCH. Which causes the Total Power of DPCH(DPCCH) and OCNS to become unfixed.

When OCNS Power is controlled to make Total power fixed by changing the power of Downlink DPCH(DPCCH) as much as specified from scenario while OCNS output level is being changed by using function BtsOcnsPower() or BtsOcnsPowerActTime() following the direction of TPC Bit of Uplink DPCH, the change of OCNS output level halts once and then OCNS power starts changing to make the Total Power fixed at the timing of the next frame. After this, Power Control halts. Consequently, the setting of the Power Control of OCNS is reflected at the timing of 1 frame late against DPCH(DPCCH).

When Compressed Mode test and OCNS test are simultaneously performed,
the Total Power of the Downlink DPCH(DPCCH)Power and the OCNS Power are to become unfixed.

3.7.4 FACH Measurement Occasion Function

Specifying the timing of FACH Measurement Occasion can temporally abort transmitting the FACH data. The FACH data of the part where the transmission is aborted is transmitted in the section to transmit next.

For details, see 3GPP TS 25.331 8.5.11 " FACH measurement occasion calculation" .

Note: FMO is an abbreviation for FACH Measurement Occasion.

3.7.4.1 How to use the function

When you Specify the timing of FACH Measurement Occasion, refer to the following formulas described in 3GPP TS 25.331.

SFN div N = C_RNTI mod M_REP + N * M_REP
M_REP = 2^k
(k is the FACH Measurement occasion cycle length coefficient.)

In MD8480A/B, specifying the coefficient " k" described above can decide the timing of aborting FACH data transmission.

In MD8480A/B, the coefficient, " k" , is set as the common variable: FMOCycleLength.

Value range configurable to FMOCycleLength : 0 to 12
To make this function effective, specify the value 1 to 12.
If 0 is set as FMOCycleLength, the function will be invalid.
The initial value of MD8480A/B is 0.

[Example of the setup]
When FACH data is transmitted at 10ms cycle(TTI=1),
if FMOCycleLength = 3; is set,
FACH data transmission is aborted at 80ms cycle.
The data that has aborted is transmitted in the next cycle(after 10ms).

To make the function effective, it is required to set up the following procedure (1) through (3) in the scenario.

(1) Set to the common variable: FMOCycleLength.
(2) Set UEIDType of LoCH mapped in FACH to UEID_C_RNTI and set UEID.
(3) Execute CmacConfig() to S-CCPCH containing the FACH setting of (2).

After executing the scenario of which the above setting is contained, execute the scenario to transmit FACH data.

[Example of scenario setting]
How to set the scenario which makes this function effective is described with a sample scenario: Start_SCCPCH(PCH_FACH).c) below.
This example omits unrelated parts with this function from the description of the sample scenario.
For details of the setup contents in the scenario, refer to B.3.4 S_CCPCH(PCH+FACH) setting in this manual.

Example of FMO setting with use of a sample scenario: Start_SCCPCH(PCH_FACH).c

```
/***** INCLUDE *****/
#include       <windows.h>
#include       "wcdma.h"
#include       "primitive.h"
#include       "parameter.h"
#include       "scenario.h"
#include       "stdio.h"

/***** FUNCTION *****/
DLLEXPORT INT ScenarioMain(LPVOID);

INT ReInitializeParameter();
INT InitializeParameter_S_CCPCH_PCHxFACH();
```

a) A function to set (2) is pronounced.

```
  :(Omitted)
INT ScenarioMain(LPVOID dmy)
{
  :(Omitted)

  SimulatorStart(0,NO_TIMEOUT );
  ReInitializeParameter();
  InitializeParameter_S_CCPCH_PCHxFACH();
```

b) The function to set (2) is executed.

```
  :(Omitted)
  FMOCycleLength = 1;
```

c) FMOCycleLength is set.
Please set it before CmacConfig() to use the parameter of InitializeParameter_S_CCPCH_PCHxFACH().

```
  CmacConfig(UNIT_BTS1,S_CCPCH,0,&CmacConfigS_CCPCH_PCHxFACH,ACTIVATE_NOW,NO_TIMEOUT);
  :(Omitted)
return(0);
}
  :(Omitted)

/*************************************************************
      InitializeParameter_S_CCPCH_PCHxFACH()
*************************************************************/
INT InitializeParameter_S_CCPCH_PCHxFACH()
```

d) The setting of (2) used in **a)** and **b)** is performed.
An optional function name is available to use. When setting with an optional function name, use it also for the setting **a)** and **b)**.
IF InitializeParameter_S_CCPCH_PCHxFACH() is named, the contents of the initial value will be " Initialize-Parameter.c" .

```
{
   CPHY_TRCH_CONFIG_PAR *CphyTrchConfigPar;
   CMAC_CONFIG_PAR          *CmacConfigPar;

   /***** for SCCPCH ******/
   CphyTrchConfigPar = &CphyTrchConfigS_CCPCH_PCHxFACH;
  :(Omitted)

   /***** for DPCH(DownLink S_CCPCH_PCHxFACH) ******/
   CmacConfigPar = &CmacConfigS_CCPCH_PCHxFACH;
  :(Omitted)

   /* TrCH #0 (logicl CH) */
  :(Omitted)

   /* TrCH #1 (logicl CH) */
  :(Omitted)
   CmacConfigPar->LochInfo[1][1].UEIDType     = UEID_C_RNTI;
   CmacConfigPar->LochInfo[1][1].UEID          = 0x0001;
  :(Omitted)

   return(0);
};
```

Set UEIDType of LoCH mapped in FACH to UEID_C_RNTI and set UEID.

3.7.5 DPCH /PRACH simultaneous boot

3.7.5.1 Outline

This manual explains the DPCH /PRACH simultaneous boot and multiple PRACH simultaneous boot function of MD8480A/B W-CDMA Signalling Tester.

In this manual, boards used with MD8480 are abbreviated as follows.

RxBB : RxBaseband
TxBB : TxBaseband

3.7.5.1.1 Function Outline

The following are presented with this function.

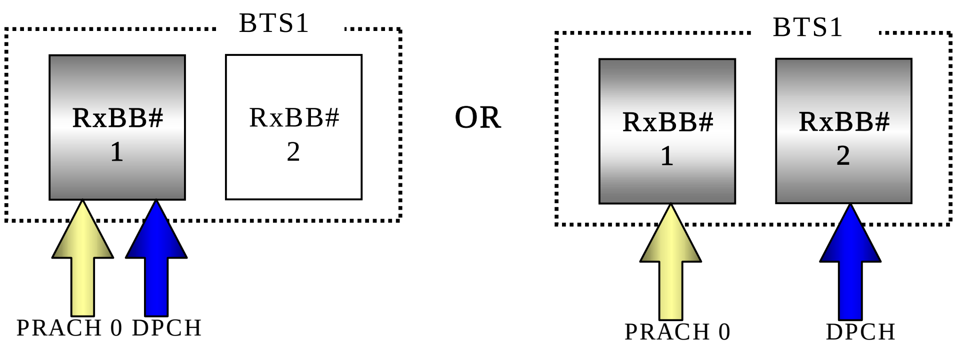
(1) PRACH and DPCH simultaneous boot at 1BTS
(2) Two PRACH boot at 1BTS
(3) PRACH boot at 2BTS
(4) PRACH and DPCH simultaneous boot at 2BTS

3.7.5.1.2 Function Details

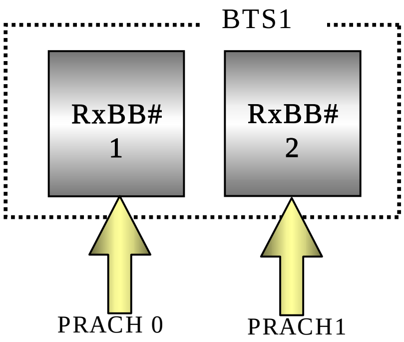
This section describes the details of the function corresponding to (1) to (4) of Section 3.7.5.1.1.

(1) There are two ways of setting to RxBB as shown below. When two RxBB are used, the frequency of PRACH and DPCH can be set with a different cycle.

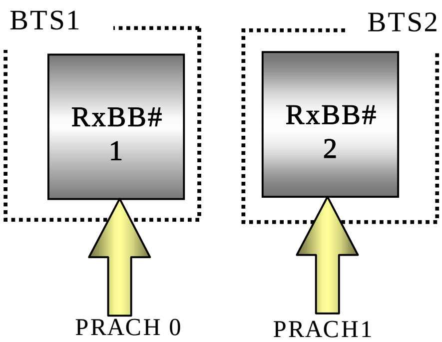
(Note) Use all Offset with the same value.



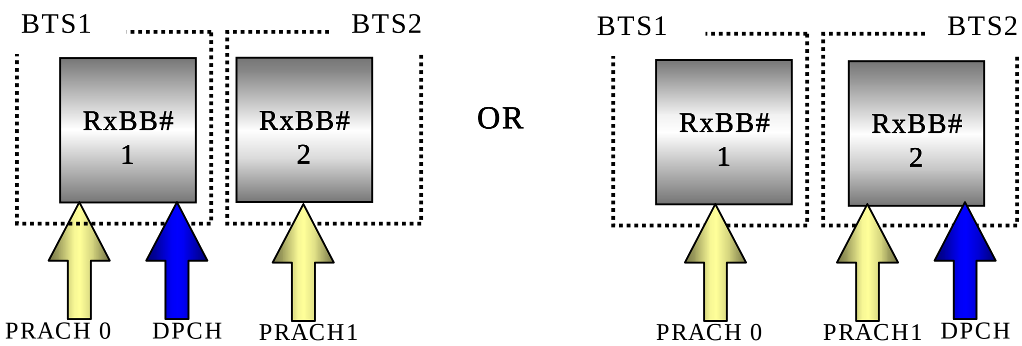
(2) The setting to RxBB is made as follows. A different Scrambling Code, Signature, and frequency can be set to PRACH.
(Note) Use all offset with the same value.



(3) The setting to RxBB is made as follows. A different Scrambling Code, Signature, and frequency can be set to PRACH.



(4) There are two ways of setting to RxBB as follows. A different Scrambling Code, Signature, and frequency and be set to PRACH.
(Note) Two DPCH can not boot simultaneously.

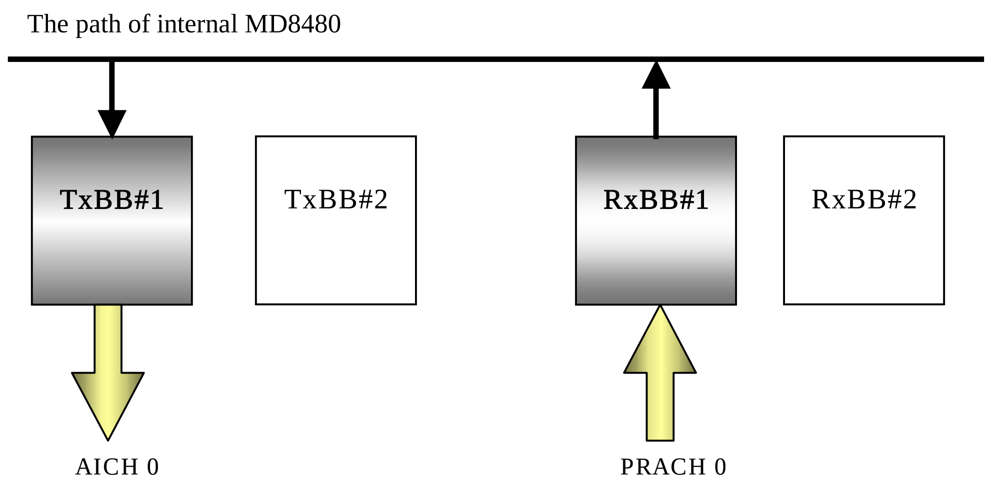


3.7.5.1.3 AICH Transmission

This section describes the AICH transmission in case of (1) through (4) of Section 3.7.5.1.1.

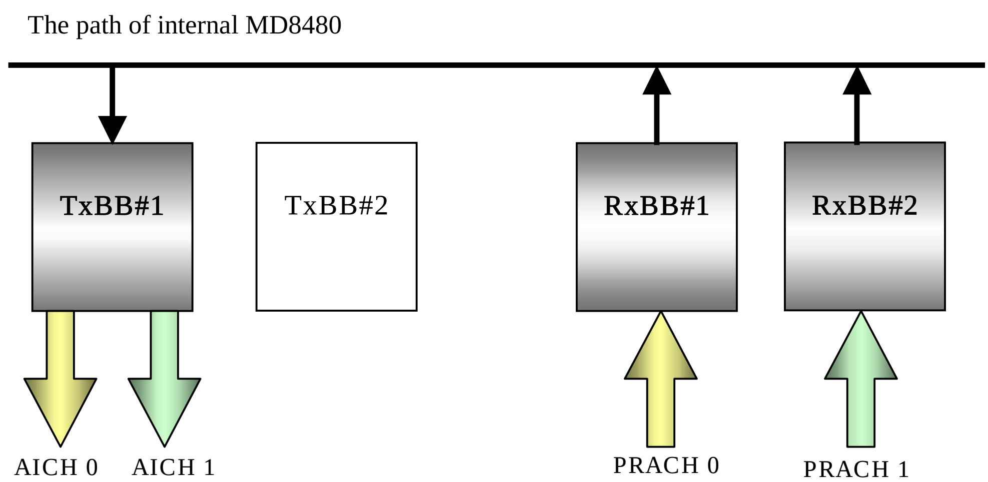
(1) Case of PRACH and DPCH simultaneously boot at 1BTS

AICH0 corresponding to PRACH0 received in RxBB#1 is transmitted from TxBB#1.



(2) Case of two PRACH boot at 1BTS

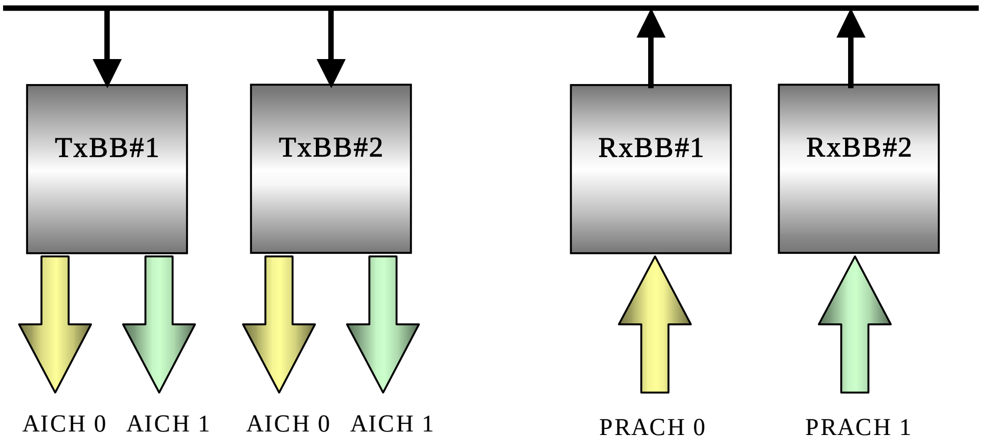
AICH0 corresponding to PRACH0 received in RxBB#1 and AICH1 corresponding to PRACH1 received in RxBB#2 are transmitted from TxBB#1.



(3) Case of PRACH boot at 2BTS

AICH0 corresponding to PRACH0 received at RxBB#1 and AICH1 corresponding to PRACH1 received at RxBB#2 are transmitted from both TxBB#1 and TxBB#2.
(note)When PRACH0 is received, AICH0 is outputted from both TxBB#1 and TxBB#2.

The path of internal MD8480



(4) Case of PRACH and DPCH simultaneously boot at 2BTS

It is the same as **(**3).

3.7.5.1.4 Reception Power Display

The reception power of BTS1-PRACH0 is displayed in BTS1 on the Monitor screen.
The reception power of BTS1-PRACH1 is displayed in **BTS2** on the Monitor screen.
The reception power of BTS2-PRACH0 is displayed in BTS2 on the Monitor screen.
The reception power of BTS2-PRACH1 is displayed in BTS2 on the Monitor screen.

3.7.5.2 How to use

3.7.5.2.1 Hardware Configuration Required

2BTS/3BTS HHO configuration (additional RF unit configuration) is required.

3.7.5.2.2 Setting of RxRF#2 (the second RxBaseBand)

This function is required to be set with BaseBand and RF Unit Connection:" SHO1" and " SHO2" .
It is not available to set RxRF#2 on the screen. RxRF#2 setting is performed by adding the following library function before SimulatorStart() on the scenario.

SetDpchPrachParm (INT UpFreq,INT RefPower)

(Note) It can be set the same cycle cell with SHO1 and a different cycle cell with SHO2.

3.7.5.3 Function Details

<table>
<tr><td>Function Name</td><td colspan="4">INT SetDpchPrachParm (INT UpFreq,INT RefPower)</td></tr>
<tr><td>Function Summary</td><td colspan="4">Setting of DPCH/PRACH simultaneous boot</td></tr>
<tr><td></td><td>Form</td><td>Argument Name</td><td>Description</td><td>Input/Output</td></tr>
<tr><td>Argument</td><td>INT
INT</td><td>UpFreq
RxRefPower</td><td>Uplink frequency
Reference Power of Rx</td><td>Input
Input</td></tr>
<tr><td></td><td>Form</td><td>Function Value</td><td colspan="2">Description</td></tr>
<tr><td>Function Value</td><td>INT</td><td>=0</td><td colspan="2">Normal end</td></tr>
<tr><td colspan="5">Function Details</td></tr>
</table>

With SetDpchPrachParm (), a parameter to RxRF#2 at the time of DPCH/PRACH simultaneous boot is set.
Be sure to carry out the following procedure only one time before calling SimulatorStart().

1) UpFreq
   The uplink frequency of RxRF#2 is specified and set by 0.1MHz.
   Ex.) When UpFreq=2110000000, it is set at 2110 MHz.
2) RxRefPower
   The uplink Reference Power of RxRF#2 is set.
   Ex.) RefPower=-300, it is set at –30dBm.

Example of use:

Add a function here.

```
INT ScenarioMain(LPVOID dmy)
{
 SetDpchPrachParm(2110000000, -300);

 SimulatorStart(0,NO_TIMEOUT);
```

Remarks	

3.7.5.4 An Example of Scenario Description

This section describes an example of Scenario Description in case of (1) through (4) of Section 3.7.5.1.1. In this case, the simultaneous boot of PRACH with TTI(20ms) and Stand-alone DCCH.

a) The simultaneous boot of PRACH and DPCH at 1BTS.

[1] Case of DPCH boot at RxBB#1

```
/* PRACH activation */
CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

/* AICH activation */
CphyRlSetup(UNIT_BTS1,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);

/* DPCH uplink activation */
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH,&CphyTrchConfigU_DPCH_SDCCH);
CphyRlSetup (UNIT_BTS1,U_DPCH,0,&CphyRlSetupU_DPCH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_DPCH,0,&CphyTrchConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig (UNIT_BTS1,U_DPCH,0,&CmacConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
```

[2] Case of DPCH boot at RxBB#2

```
/* PRACH activation */
CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

/* AICH activation */
CphyRlSetup(UNIT_BTS1,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);

/* DPCH uplink activation */
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH_BTS2,&CphyTrchConfigU_DPCH_SDCCH);
CphyRlSetup(UNIT_BTS2,U_DPCH,0,&CphyRlSetupU_DPCH_BTS2,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS2,U_DPCH,0,&CphyTrchConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS2,U_DPCH,0,&CmacConfigU_DPCH_SDCCH, ACTIVATE_NOW,NO_TIMEOUT);
```

b) The simultaneous boot of two PRACH at 1BTS

```
/* PRACH activation */
CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

CphyRlSetup(UNIT_BTS1,U_PRACH,1,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,1,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

CmacConfig(UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

/* AICH activation */
CphyRlSetup(UNIT_BTS1,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);
```

c) The simultaneous boot of PRACH at 2BTS

```
/* PRACH activation */
CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

CalcRMParameter(U_PRACH,&CphyRlSetupPRACH_BTS2,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS2,U_PRACH,0,&CphyRlSetupPRACH_BTS2,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS2,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

/* AICH activation */
CphyRlSetup(UNIT_BTS1,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);
CphyRlSetup(UNIT_BTS2,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);
```

d) The simultaneous boot of PRACH and DPCH at 2BTS

[1] Case of DPCH boot at BTS1

```
/* PRACH activation */
CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

CalcRMParameter(U_PRACH,&CphyRlSetupPRACH_BTS2,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS2,U_PRACH,0,&CphyRlSetupPRACH_BTS2, ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS2,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

/* AICH activation */
CphyRlSetup(UNIT_BTS1,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);
CphyRlSetup(UNIT_BTS2,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);

/* DPCH uplink activation */
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH,&CphyTrchConfigU_DPCH_SDCCH);
CphyRlSetup(UNIT_BTS1,U_DPCH,0,&CphyRlSetupU_DPCH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_DPCH,0,&CphyTrchConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS1,U_DPCH,0,&CmacConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
```

[2] Case of DPCH boot at BTS2

```
/* PRACH activation */
CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

CalcRMParameter(U_PRACH,&CphyRlSetupPRACH_BTS2,&CphyTrchConfigPRACH20M);
CphyRlSetup(UNIT_BTS2,U_PRACH,0,&CphyRlSetupPRACH_BTS2,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS2,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);

/* AICH activation */
CphyRlSetup(UNIT_BTS1,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);
CphyRlSetup(UNIT_BTS2,D_AICH,0,&CphyRlSetupAICH,ACTIVATE_NOW,NO_TIMEOUT);

/* DPCH uplink activation */
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH_BTS2,&CphyTrchConfigU_DPCH_SDCCH);
CphyRlSetup(UNIT_BTS2,U_DPCH,0,&CphyRlSetupU_DPCH_BTS2,ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS2,U_DPCH,0,&CphyTrchConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig(UNIT_BTS2,U_DPCH,0,&CmacConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
```

3.7.5.5 Restriction

This function has the following restriction matters.

- In use of the multiple PRACH Simultaneous Boot, any of the following combinations allow the function to boot.

(BTS1-PRACH1, BTS2-PRACH0, and BTS2-PRACH1 can not boot at the same time.)

(1) The simultaneous boot of BTS1-PRACH0 and BTS1-PRACH1
(2) The simultaneous boot of BTS1-PRACH0 and BTS2-PRACH0
(3) The simultaneous boot of BTS1-PRACH0 and BTS2-PRACH1

- When PRACH has never received, DPCH cannot be received.
- UL DPCH cannot boot two channels at the same time. (If one channel is booted at Rx Baseband, UL DPCH of another Rx Baseband will be stopped.) Be sure to set any of the frequency, Scrambling Code, and Signature of two PRACH to differ. (We will not ensure the performance when two RxBB are received simultaneously.)

3.7.6 SSDT Function

3.7.6.1 Overview

SSDT is a power control method applied at the time of Soft Handover. Using this function, you need use this function after writing Option Key for Tx Diversity. Its main roles of SSDT are to decrease the interference of multiplex transmission and to perform high-speed cell selection without exceeding the load of the network. A mobile station selects one cell for “primary” and the others for “non-primary” from cells under processing handover. To select the cell for “primary”, each cell is given a temporary ID and the mobile station transmits primary cell ID using Uplink FBI field. Cells selected as “non-primary” turn DPDCH transmission off. SSDT cannot use together with the other diversity.
For details of SSDT, please refer to 3GPP TS25.214.

3.7.6.2 How to use

3.7.6.2.1 Setting of the SETUP screen

Choose SHO1 or SHO2 at ”Base Band and RF Unit Connection” on the SETUP screen.

3.7.6.2.2 How to set a scenario

ID assignment of ID is performed at Scenario in Signalling Tester. To make SSDT effective, set the constant: DIVERSITY_SSDT to the member variable: TxDiversity of the common variable: CphyRlSetupD_DPCHXXX (See Table 3.6 as shown below.), store any of constants described in Table 3.8 in TmpCellID, and execute CphyRlSetup().
Use a proper constant according to the ID codes setting.
Furthermore, setting the constant: DIVERSITY_OFF to TxDiversity and executing　CphyRlSetup() make SSDT invalid.

Table 3.6: Common variables to be used

DL
CphyRlSetupD_DPCH; CphyRlSetupD_DPCH_BTS1; CphyRlSetupD_DPCH_BTS2; CphyRlSetupD_DPCH_BTS3;

Table 3.7: Member variables of CphyRlSetupD_DPCHXXX

Name	3.7.2.　Contents	Setup Value	Meaning of Value
TxDiversity	Starting/Stopping of SSDT *It is used for starting /stopping TxDiversity.	“DIVERSITY_OFF”	Stopping TxDiversity and SSDT performance
		“DIVERSITY_SSDT”	Starting SSDT performance
TmpCellID	CellID of the BTS	See Table 3.8.	See Table 3.8.

(Example in case of making SSDT effective)
```
/* Downlink BTS#1 */
CphyRlSetupD_DPCH_BTS1.TxDiversity   = DIVERSITY_SSDT;
CphyRlSetupD_DPCH_BTS1.TmpCellID   = SSDT_LABEL_A_SHORT_1BIT;   /* label a, short, FBI 1 bit */
CphyRlSetup(UNIT_BTS1, D_DPCH, 0, &CphyRlSetupD_DPCH_BTS1 ,ACTIVATE_NOW,NO_TIMEOUT);

/* Downlink BTS#2 */
CphyRlSetupD_DPCH_BTS2.TxDiversity = DIVERSITY_SSDT;
CphyRlSetupD_DPCH_BTS2.TmpCellID = SSDT_LABEL_C_SHORT_1BIT;   /* label c, short, FBI 1 bit */
CphyRlSetup(UNIT_BTS2, D_DPCH, 0, &CphyRlSetupD_DPCH_BTS2 ,ACTIVATE_NOW,NO_TIMEOUT);
```

(Example in case of making SSDT invalid)
```
/* Downlink BTS#1 */
CphyRlSetupD_DPCH_BTS1.TxDiversity = DIVERSITY_OFF;
CphyRlSetup(UNIT_BTS1, D_DPCH, 0, &CphyRlSetupD_DPCH_BTS1 ,ACTIVATE_NOW,NO_TIMEOUT);

/* Downlink BTS#2 */
CphyRlSetupD_DPCH_BTS2.TxDiversity = DIVERSITY_OFF;
CphyRlSetup(UNIT_BTS2, D_DPCH, 0, &CphyRlSetupD_DPCH_BTS2 ,ACTIVATE_NOW,NO_TIMEOUT);
```

*Be sure to set Temporary Call ID to TmpCellID when ” DIVERSITY_SSDT” is set in TxDiversity.
If the ID is not set, it performs as PrimaryCell.
*When SSDT is performed, execute CphyRlSetup() to both of CphyRlSetupD_DPCHXXX and CphyRlSetupU_DPCHXXX.
*Make FBI length which is set in ID and FBI bit length which is set in CphyRlSetup(U_DPCH) with the same.
*Make the Cell ID Length(Long/Medium/Short) of the ID which is set to each BTS with the same.
*Set a different Cell ID (a through h) to a different BTS.

Table 3.8: Constant to set for ” TmpCellID”

Constant	ID label	ID code length	FBI bits
SSDT_LABEL_A_SHORT_1BIT	A	short	1 bit
SSDT_LABEL_A_SHORT_2BIT	A	short	2 bit
SSDT_LABEL_A_MEDIUM_1BIT	A	medium	1 bit
SSDT_LABEL_A_MEDIUM_2BIT	A	medium	2 bit
SSDT_LABEL_A_LONG_1BIT	A	long	1 bit
SSDT_LABEL_A_LONG_2BIT	A	long	2 bit
SSDT_LABEL_B_SHORT_1BIT	B	short	1 bit
SSDT_LABEL_B_SHORT_2BIT	B	short	2 bit
SSDT_LABEL_B_MEDIUM_1BIT	B	medium	1 bit
SSDT_LABEL_B_MEDIUM_2BIT	B	medium	2 bit
SSDT_LABEL_B_LONG_1BIT	B	long	1 bit
SSDT_LABEL_B_LONG_2BIT	B	long	2 bit
SSDT_LABEL_C_SHORT_1BIT	C	short	1 bit
SSDT_LABEL_C_SHORT_2BIT	C	short	2 bit
SSDT_LABEL_C_MEDIUM_1BIT	c	medium	1 bit
SSDT_LABEL_C_MEDIUM_2BIT	c	medium	2 bit
SSDT_LABEL_C_LONG_1BIT	c	long	1 bit
SSDT_LABEL_C_LONG_2BIT	c	long	2 bit
SSDT_LABEL_D_SHORT_1BIT	d	short	1 bit
SSDT_LABEL_D_SHORT_2BIT	d	short	2 bit
SSDT_LABEL_D_MEDIUM_1BIT	d	medium	1 bit
SSDT_LABEL_D_MEDIUM_2BIT	d	medium	2 bit
SSDT_LABEL_D_LONG_1BIT	d	long	1 bit
SSDT_LABEL_D_LONG_2BIT	d	long	2 bit
SSDT_LABEL_E_SHORT_1BIT	e	short	1 bit
SSDT_LABEL_E_SHORT_2BIT	e	short	2 bit
SSDT_LABEL_E_MEDIUM_1BIT	e	medium	1 bit
SSDT_LABEL_E_MEDIUM_2BIT	e	medium	2 bit
SSDT_LABEL_E_LONG_1BIT	e	long	1 bit
SSDT_LABEL_E_LONG_2BIT	e	long	2 bit
SSDT_LABEL_F_SHORT_1BIT	f	short	1 bit
SSDT_LABEL_F_SHORT_2BIT	f	short	2 bit
SSDT_LABEL_F_MEDIUM_1BIT	f	medium	1 bit
SSDT_LABEL_F_MEDIUM_2BIT	f	medium	2 bit
SSDT_LABEL_F_LONG_1BIT	f	long	1 bit
SSDT_LABEL_F_LONG_2BIT	f	long	2 bit
SSDT_LABEL_G_SHORT_1BIT	g	short	1 bit
SSDT_LABEL_G_SHORT_2BIT	g	short	2 bit
SSDT_LABEL_G_MEDIUM_1BIT	g	medium	1 bit
SSDT_LABEL_G_MEDIUM_2BIT	g	medium	2 bit
SSDT_LABEL_G_LONG_1BIT	g	long	1 bit
SSDT_LABEL_G_LONG_2BIT	g	long	2 bit
SSDT_LABEL_H_SHORT_1BIT	h	short	1 bit
SSDT_LABEL_H_SHORT_2BIT	h	short	2 bit
SSDT_LABEL_H_MEDIUM_1BIT	h	medium	1 bit
SSDT_LABEL_H_MEDIUM_2BIT	h	medium	2 bit
SSDT_LABEL_H_LONG_1BIT	h	long	1 bit
SSDT_LABEL_H_LONG_2BIT	h	long	2 bit

3.7.6.2.3 Requirement for accordance of ID

In case the received primary ID code does not match with the own ID code, it considers as “ non-primary” . Otherwise it considers as “ primary” .

3.7.6.2.4 Performance of Signalling Tester when ID accords

On Signalling Tester, A ID can be set by every BTS according to a scenario and decided whether “ primary” or “ non-primary” by every BTS.
When it is decided as “ primary” by BTS, both DPCCH and DPDCH perform normally.
When it is decided as “ non-primary” by BTS, DPCCH normally performs but the transmission of the DPDCH will be stopped.

3.7.6.3 Restriction

(1) SSDT cannot use Closed Loop mode of TxDiversity together.
(2) When downlink DPCH is MultiCode, set its transmission power as the same as the total transmission power of DPDCH.
(3) SSDT cannot use OCNS together. The describing method of Scenario will be changed when OCNS is supported.
(4) SSDT cannot use Compressed Mode together.

3.7.7 RLC UM Reconfiguration Function

3.7.7.1 Outline

This function is to change the PLC PDU size without initializing SN on UM of RLC in MD 8480A/B.

3.7.7.1.1 Sequence Image

The following Figure 3.34 shows the image of performance sequence in case of changing the RLC PDU size without initializing SN.

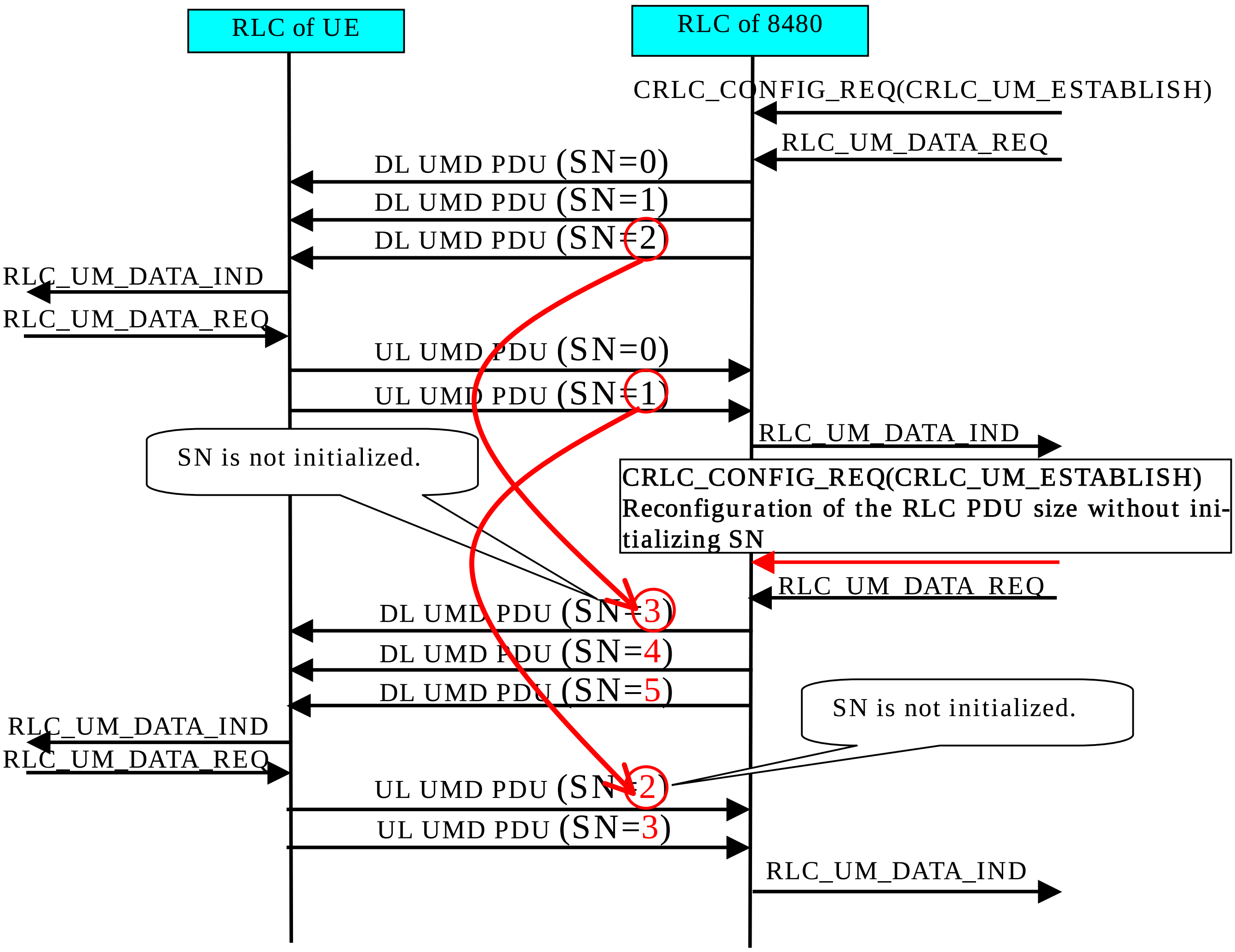


Figure 3.34 Sequence without initializing SN

3.7.7.2 Setting Function

The performance when CrlcConfig(CRLC_UM_ESTABLISH) is executed is set by the value of Global Variable: RlcUSKeep under scenario execution. The specific setting values are shown in Table 3.9.

Table 3.9 Setting values of RlcUSKeep

Setting Value	SN after CrlcConfig
RLC_US_INIT(initial value)	Initialized to 0
RLC_US_KEEP	Continuation

SN is initialized and becomes 0 when nothing or RLC_US_INIT is set to RlcUSKeep, and it is not initialized when RLC_US_KEEP is set to RlcUSKeep.
Since RlcUSKeep is a global variable, it is not automatically changed once a value has set unless the variable value is consciously changed. Execute Function: CrlcConfig() once by RLC_US_KEEP and set RLC_US_INIT in advance if SN is initialized on and after second execution of Function: CrlcConfig().
Also, this variable will be ignored when Argument: RlcMode of Function: CrlcConfig() is except CRLC_UM_ESTABLISH.
Moreover, since the variable TlcSkeep is already pronounced at the time of staring on ControlSoftware start-up, it is unnecessary to pronounce it in the scenario again.

3.7.7.3 An Example of Scenario Description

The following is an example of describing a part of RLC Config.

```
CrlcConfigPar.PU_LengthUM = 136;
CrlcConfig(UNIT_BTS1,CRLC_UM_ESTABLISH,DCCH,0,&CrlcConfigPar,RRC,NO_TIMEOUT);

                              ……..

RlcUSKeep = RLC_US_KEEP;
CrlcConfigPar.PU_LengthUM = 120;
CrlcConfig(UNIT_BTS1,CRLC_UM_ESTABLISH,DCCH,0,&CrlcConfigPar,RRC,NO_TIMEOUT);
RlcUSKeep = RLC_US_INIT;
```

3.7.8 Variable rate AMR

3.7.8.1 Outline

This function enables MD8480 to transfer and receive the service specified under 3GPP TS34.108 section 6.10.2.4.1.4a, section 6.10.2.4.1.5a and section 6.10.2.4.1.7a.

3.7.8.2 How to use

In order to use this function, you need set as below before carrying SimulatorStart()
CrlcStartPar.Flags = ENABLE_TM_BITMODE;

3.7.8.3 Settings

This is description about the service specified under 3GPP TS34.108 section 6.10.2.4.1.4a, section 6.10.2.4.1.5a and section 6.10.2.4.1.7a.

3.7.8.3.1 " 6.10.2.4.1.4a" Setting

3.7.8.3.1.1 Receiving " 6.10.2.4.1.4a"

3.7.8.3.1.1.1 Uplink Setting

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 60ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	0.84

3.7.8.3.1.1.2 Receivable TFCS etc.

Transport Format Set						
DPCH	TrCH type		DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x81	0x103	0x60	0x148
		TF1,bits	1x39	1x53	1x60	1x148
		TF2,bits	1x42	1x63	-	-
		TF3,bits	1x55	1x84		
		TF4,bits	1x75	1x103		
		TF5,bits	1x81	-		
	TTI,ms		20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Convolutional Coding1/3
	CRC, bit		12	-	-	16
	RM attribute		200	190	235	160

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0
1	TF1	TF0	TF0	TF0
2	TF2	TF1	TF0	TF0
3	TF3	TF2	TF0	TF0
4	TF4	TF3	TF0	TF0
5	TF5	TF4	TF1	TF0
6	TF0	TF0	TF0	TF1
7	TF1	TF0	TF0	TF1

8	TF2	TF1	TF0	TF1

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
9	TF3	TF2	TF0	TF1
10	TF4	TF3	TF0	TF1
11	TF5	TF4	TF1	TF1

3.7.8.3.1.2 Transmitting “ 6.10.2.4.1.4a”

3.7.8.3.1.2.1 Downlink Setting

Slot Format(Symbol Rate)	Be sure to set to 8(30ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

3.7.8.3.1.2.2 Transmittable TFCS etc.

Transport Format Set						
DPCH	TrCH type		DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	0x103	0x60	0x148
		TF1,bits	1x39	1x53	1x60	1x148
		TF2,bits	1x42	1x63	-	-
		TF3,bits	1x55	1x84		
		TF4,bits	1x75	1x103		
		TF5,bits	1x81	-		
	TTI,ms		20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Convolutional Coding1/3
	CRC, bit		12	-	-	16
	RM attribute		200	190	235	160

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0
1	TF1	TF0	TF0	TF0
2	TF2	TF1	TF0	TF0
3	TF3	TF2	TF0	TF0
4	TF4	TF3	TF0	TF0
5	TF5	TF4	TF1	TF0
6	TF0	TF0	TF0	TF1
7	TF1	TF0	TF0	TF1
8	TF2	TF1	TF0	TF1
9	TF3	TF2	TF0	TF1
10	TF4	TF3	TF0	TF1
11	TF5	TF4	TF1	TF1

3.7.8.3.2 " 6.10.2.4.1.5a" Setting

3.7.8.3.2.1 Receiving " 6.10.2.4.1.5a"

3.7.8.3.2.1.1 Uplink Setting

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 60ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	0.96

3.7.8.3.2.1.2 Receivable TFCS etc.

Transport Format Set						
DPCH	TrCH type		DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x65	0x99	0x40	0x148
		TF1,bits	1x39	1x53	1x40	1x148
		TF2,bits	1x42	1x63	-	-
		TF3,bits	1x55	1x76		
		TF4,bits	1x58	1x99		
		TF5,bits	1x65	-		
	TTI,ms		20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Convolutional Coding1/3
	CRC, bit		12	-	-	16
	RM attribute		200	190	235	160

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0
1	TF1	TF0	TF0	TF0
2	TF2	TF1	TF0	TF0
3	TF3	TF2	TF0	TF0
4	TF4	TF3	TF0	TF0
5	TF5	TF4	TF1	TF0
6	TF0	TF0	TF0	TF1
7	TF1	TF0	TF0	TF1
8	TF2	TF1	TF0	TF1
9	TF3	TF2	TF0	TF1
10	TF4	TF3	TF0	TF1
11	TF5	TF4	TF1	TF1

3.7.8.3.2.2 Transmitting “ 6.10.2.4.1.5a”

3.7.8.3.2.2.1 Downlink Setting

Slot Format(Symbol Rate)	Be sure to set to 8(30ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

3.7.8.3.2.2.2 Transmittable TFCS etc.

Transport Format Set						
DPCH	TrCH type		DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	0x99	0x40	0x148
		TF1,bits	1x39	1x53	1x40	1x148
		TF2,bits	1x42	1x63	-	-
		TF3,bits	1x55	1x76		
		TF4,bits	1x58	1x99		
		TF5,bits	1x65	-		
	TTI,ms		20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Convolutional Coding1/3
	CRC, bit		12	-	-	16
	RM attribute		200	190	235	160

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0
1	TF1	TF0	TF0	TF0
2	TF2	TF1	TF0	TF0
3	TF3	TF2	TF0	TF0
4	TF4	TF3	TF0	TF0
5	TF5	TF4	TF1	TF0
6	TF0	TF0	TF0	TF1
7	TF1	TF0	TF0	TF1
8	TF2	TF1	TF0	TF1
9	TF3	TF2	TF0	TF1
10	TF4	TF3	TF0	TF1
11	TF5	TF4	TF1	TF1

3.7.8.3.3 " 6.10.2.4.1.7a" Setting

3.7.8.3.3.1 Receiving " 6.10.2.4.1.7a"

3.7.8.3.3.1.1 Uplink Setting

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 60ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	0.96

3.7.8.3.3.1.2 Receivable TFCS etc.

Transport Format Set					
DPCH	TrCH type		DCH	DCH	DCH
	TFS	TF0,bits	0x61	0x87	0x148
		TF1,bits	1x39	1x53	1x148
		TF2,bits	1x42	1x63	-
		TF3,bits	1x55	1x76	
		TF4,bits	1x58	1x87	
		TF5,bits	1x61	-	
	TTI,ms		20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		12	-	16
	RM attribute		200	190	160

Transport Format Combination Set			
TFCI	DCH	DCH	DCH
0	TF0	TF0	TF0
1	TF1	TF0	TF0
2	TF2	TF1	TF0
3	TF3	TF2	TF0
4	TF4	TF3	TF0
5	TF5	TF4	TF0
6	TF0	TF0	TF1
7	TF1	TF0	TF1
8	TF2	TF1	TF1
9	TF3	TF2	TF1
10	TF4	TF3	TF1
11	TF5	TF4	TF1

3.7.8.3.3.2 Transmitting " 6.10.2.4.1.7a"

3.7.8.3.3.2.1 Downlink Setting

Slot Format(Symbol Rate)	Be sure to set to 8(30ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

3.7.8.3.3.2.2 Transmittable TFCS etc.

Transport Format Set					
DPCH	TrCH type		DCH	DCH	DCH
	TFS	TF0,bits	1x0	0x87	0x148
		TF1,bits	1x39	1x53	1x148
		TF2,bits	1x42	1x63	-
		TF3,bits	1x55	1x76	
		TF4,bits	1x58	1x87	
		TF5,bits	1x61	-	
	TTI,ms		20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		12	-	16
	RM attribute		200	190	160

Transport Format Combination Set			
TFCI	DCH	DCH	DCH
0	TF0	TF0	TF0
1	TF1	TF0	TF0
2	TF2	TF1	TF0
3	TF3	TF2	TF0
4	TF4	TF3	TF0
5	TF5	TF4	TF0
6	TF0	TF0	TF1
7	TF1	TF0	TF1
8	TF2	TF1	TF1
9	TF3	TF2	TF1
10	TF4	TF3	TF1
11	TF5	TF4	TF1

3.7.9 PDCP function

3.7.9.1 Outline

Packet Data Convergence Protocol (PDCP) is described in 3GPP TS 25.323. PDCP is U-plane protocol that realizes efficient packet communication between UE and RNC and has the following features.

1. Providing a function to transfer user data
2. Managing PDCP Sequence Number

3.7.9.1.1 Function to transfer user data

As shown in Figure 3.35, PDCP saves user data. TE overlaps with PDCP on Trace of Control Software as shown in Figure 3.36.

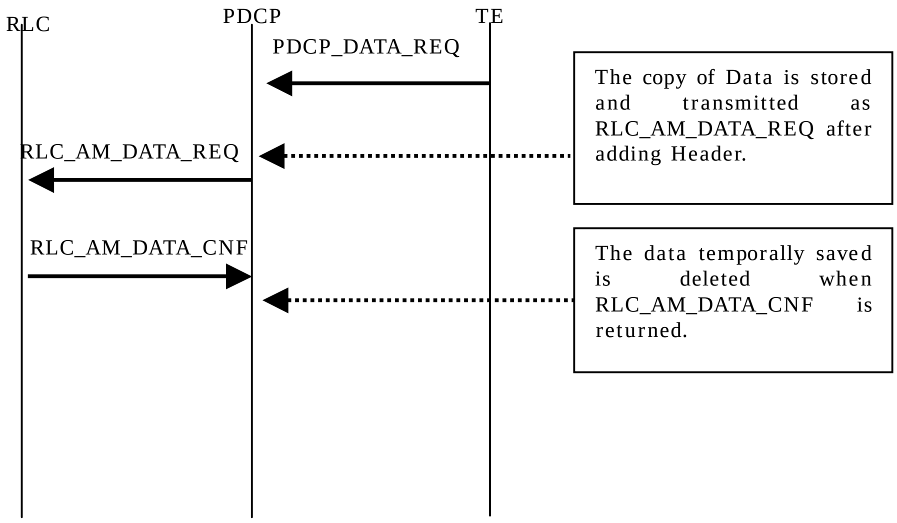
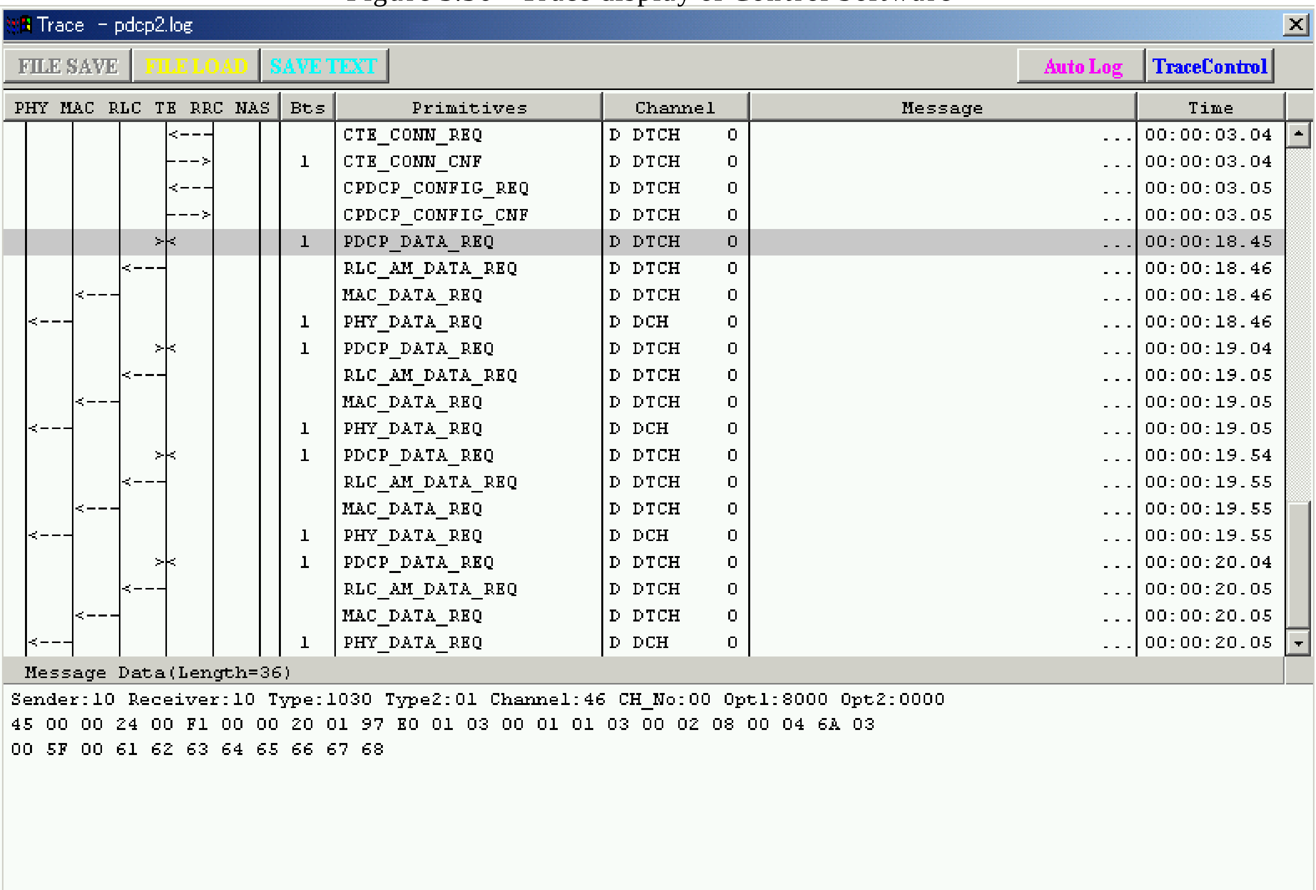


Figure 3.35 Transferring of user data

Figure 3.36　Trace display of Control Software



3.7.9.1.2 Management of PDCP Sequence Number

PDCP Sequence Number(hereafter SN) is transmitted with being added to the header of RLC_AM_DATA_REQ when retransmitting due to a lack of transfer data. Figure 3.37 shows an example of the sequence when SN is re-transmitted. The following are the explanation of Figure 3.37.

(1) Put it that the data SN 0 through 10 is transmitted.
(2) When the data was transmitted up to SN 9, CNF up to SN 6 was re-turned but CNF after SN7 was not returned.
(3) The base station checks to what number of SN is transmitted by CPDCP_RELOC_CNF. (Here, it is to 9.)
(4) The mobile station informs to what number of SN is received by PDCP_SN_REQ. (Here, it is to 6.)
(5) Retransmission is started if (3) is larger than (4) in comparison be-tween (3) and (4).
(6) After completing the retransmission, normal transmission is started from SN10.

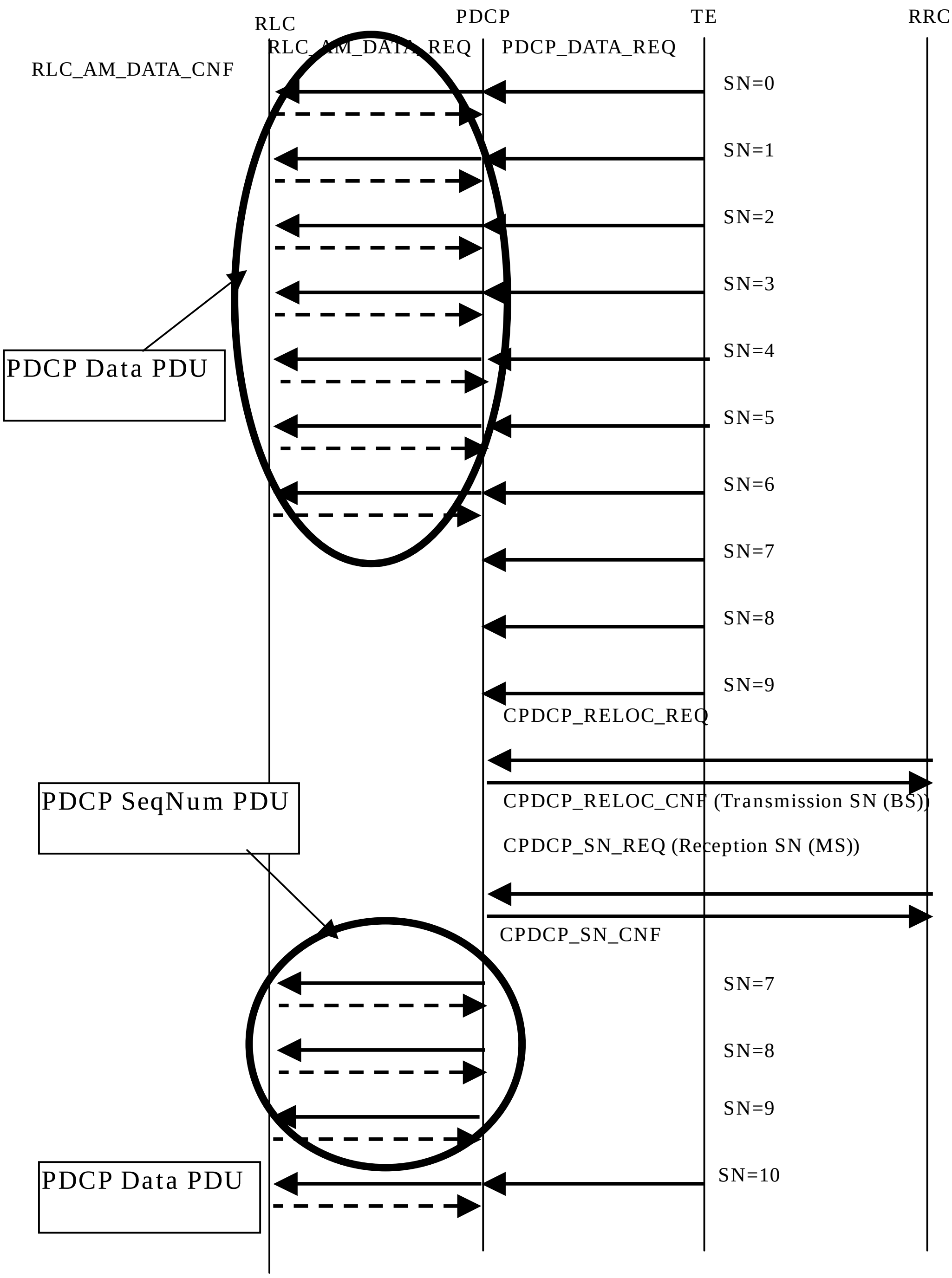


Figure 3.37 An example of PDCP sequence

Also, there are three kinds of PDU and they differ in their header forms. (See Figure 3.38.)

1. PDCP No Header PDU ---- PDU with no Header of PDCP
2. PDCP Data PDU ---- PDU in case of normal transmission in Figure 3.37
3. PDCP SeqNum PDU ---- PDU in case of retransmission in Figure 3.37.

In Figure 3.38,Type is 0 in case of (2) and 1 in case of (3).

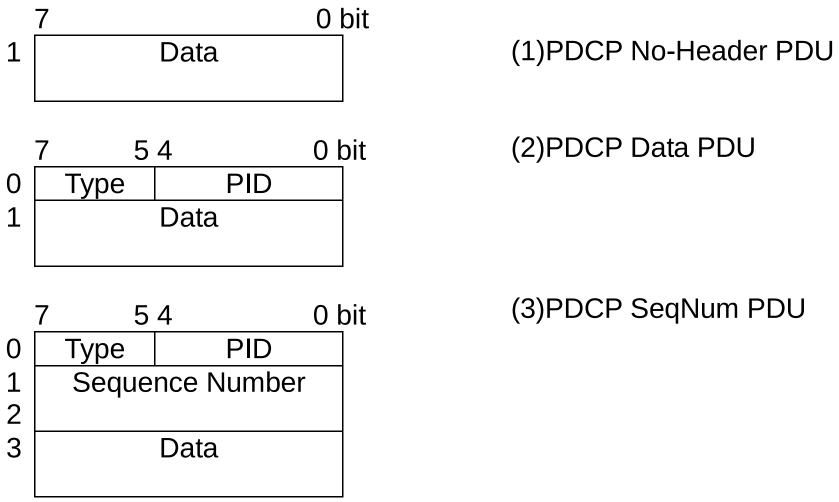


Figure 3.38 Classification of PDCP PDU

3.7.9.2 How to set

Set PDCP as follows when it is used.

3.7.9.2.1 Setting of CTE_CONFIG_PAR

When PDCP is used in Downlink, execute the CteConfig() after making the following setup.

```
CTE_CONFIG_PAR    CteConfigIPPacket;    /* Config Parameter */

        CteConfigIPPacket.Rate         = 0;

        CteConfigIPPacket.TTI          = 1;
        CteConfigIPPacket.NumOfTB      = 1;
        CteConfigIPPacket.TBS          = 312; /* Bit Size */
        CteConfigIPPacket.Frame        = PDCP_DATA_REQ;
        CteConfigIPPacket.Layer        = L2SUB_PDCP;

        CteConfig(Loch, LochNo, &CteConfigIPPacket, NO_TIMEOUT);
```

3.7.9.2.2. Setting of CRLC_CONFIG_PAR

When PDCP is used in Uplink, execute the CrlcConfig() after making the following setting of L2sublayer.

```
CrlcConfig(UNIT_BTS1,CRLC_AM_ESTABLISH,DTCH,0,
    &CrlcConfigDTCH, L2SUB_PDCP ,NO_TIMEOUT);
```

3.7.9.2.3 Configuration/ Release for PDCP

It is performed using the CpdcpConfig() and CpdcpRelease().

3.7.9.2.4 How to transmit/receive sequence numbers

In Downlink, the value of transmitted SN is acquired by using the CpdcpReloc().
The reception the mobile station side received by RRC Message is informed to the PDCP layer of the base station side, by using the CpdcpSeqNum(). When the reception SN of the mobile station side is smaller than the transmitted SN of the base station side, it is considered as a communication error and started to retransmit.
In Uplink, the reception SN of the base station side is acquired with the CpdcpReloc(). Then it is informed to the mobile station side by using RRC Message.

3.7.9.3 Restriction Matters

PDCP has the following function except 1 and 2 of 3.7.9.1.
But it is not supported.

- Header compression function for IP data (RFC2507)

The function of transfer user data is described in the chapter 3.7.9.1.1.
PDCP can buffer 256 SDUs or less without the acknowledgement in this function. If more PDCP SDUs than 256 are not acknowledged, the SDUs with older SN will be discarded in turn.

3.7.9.4 Scenario

In using PDCP, there is the necessity of setting up RLC apart from an appending scenario. For details, please look at "a setup of 3.7.9.2.2.CRLC_CONFIG_PAR."

3.7.9.4.1 Attached Scenario

The IP Packet scenarios for PDCP are shown below. Please refer the table below with C.4 IP Packet Test.

Table 3.10 Scenario list

Scenario Name	Description
AirOrig_IPPacket_PDCP.c	Connects a call of packet data on 10/100Base-T interface of Signalling Tester.
AirDisc_IPPacket_PDCP.c	Clears a call of packet data on 10/100Base-T interface of Signalling Tester.
PDCPSndRcvSeqNum.c	Transmits/Receives a sequence number.

3.7.9.4.2 Scenario Configuration

AirOrig_IPPacket_PDCP.c

By using the CteConfig(), a TE type is corresponded to a logical channel number and the configuration parameter of PDCP is set. And after the transmission/reception of Packet data by CteConnect( ), the configuration parameter of PDCP is set by the CpdcpConfig().

On the CteConfig() and CteConnect(), see C.4 IP Packet test.
But,
CteConfigIPPacket.Layer = L2SUB_PDCP;
is set for using PDCP on the CteConfig().

Table 3.11 CpdcpConfig() argument list

Argument	Description
1st argument	Specifies a logical channel
2nd argument	Specifies a logical channel number
3rd argument	Specifies options
4th argument	Specifies time for time-out

On the CpdcpConfig(), a logical channel is corresponded to a logical channel number. In this scenario, the logical channel number is 0.
The optional CPDCP_CONFIG_PAR structure member is set as follows.
The initial value of transmission SN: 0 to 65535 can be set as Start_DL_Send_SN.
The initial value of reception SN: 0 to 65535 can be set as Start_UL_Receive_SN.
As a frame, RLC_AM_DATA_REQ, RLC_UM_DATA_REQ, and RLC_TR_DATA_REQ can be selected.
RLC is specified as Lower Layer.
TE is specified as Upper Layer.

AirDisc_IPPacket_PDCP.c

To release PDCP, execute the CpdcpRelease().
Execute the CteDisconnect() to clear a call and reset a logical channel, a logical channel number, and a configuration parameter by the CteRelease().

Table 3.12 CpdcpRelease() argument list

Argument	Description
1st argument	Specifies a logical channel
2nd argument	Specifies a logical channel number
3rd argument	Specifies time for time-out

The CpdcpRelease() releases PDCP entity which corresponds a specified logical channel and logical channel number. On the CteDisconnect() and CteRelease(), see C.4 IP Packet Test.

PDCPSndRcvSeqNum.c

Execute the CpdcpReloc() to acquire the reception SN and transmitted SN of the base station side corresponding to a specified logical channel and logical channel number.

Execute the SndMessage() to allow the RRC of the mobile station side to inform on the SN of the mobile station side. (Since this scenario data: SndData is a dummy, the actual data will differ from it.)

Execute the RcvMessage() to acquire the SN of a mobile station side from its RRC.

Execute the CpdcpSeqNum() to inform transmitted and reception SN of the mobile station side corresponding to the specified logical channel and logical channel number.

Table 3.13 CpdcpReloc () argument list

Argument	Description
1st argument	Specifies a logical channel
2nd argument	Specifies a logical channel number
3rd argument	Specifies options
4th argument	Specifies time for time-out

Table 3.14 CpdcpSeqNum () argument list

Argument	Description
1st argument	Specifies a logical channel
2nd argument	Specifies a logical channel number
3rd argument	Specifies options
4th argument	Specifies time for time-out

3.7.10 Cmac Restriction function

This function enables data to transmit with using another TFC even if one TFC is set to 0 to be out of use, on the CMAC_CONFIG_PAR structure of Initialize Parameter.

The following describes the function showing an example of Harikiri Test of AMR, Packet, and DCCH with the attached scenario, Start_AMRxP384K(TTI20m).c.

In case of the following data transmission(hereafter (1,0,0,4,1)) on InitializeParameter

```
                  /* TFC = 27 */
CmacConfigPar->TFCS.TFC[27][0] = 1;     /* DCH#0 */
CmacConfigPar->TFCS.TFC[27][1] = 0;     /* DCH#1 */
CmacConfigPar->TFCS.TFC[27][2] = 0;     /* DCH#2 */
CmacConfigPar->TFCS.TFC[27][3] = 4;     /* DCH#3 */
CmacConfigPar->TFCS.TFC[27][4] = 1;     /* DCH#4 */
```

Data is usually transmitted in the form of (1,0,0,4,1)TFC = 27 from MAC. Here, (1,0,0,4,1) is rewritten to (0,0,0,0,0) and described as follows.

```
                  /* TFC = 27 */
CmacConfigPar->TFCS.TFC[27][0] = 0;     /* DCH#0 */
CmacConfigPar->TFCS.TFC[27][1] = 0;     /* DCH#1 */
CmacConfigPar->TFCS.TFC[27][2] = 0;     /* DCH#2 */
CmacConfigPar->TFCS.TFC[27][3] = 0;     /* DCH#3 */
CmacConfigPar->TFCS.TFC[27][4] = 0;     /* DCH#4 */
```

When the data of (1,0,0,3,1) is received from RLC, MAC cannot transmit it in the form of (1,0,0,4,1). First, the data of (1,0,0,3,1) TFC= 21 is transmitted and (0,0,0,1,0) is remained within MAC. Next, data is transmitted as (1,0,0,5,1) TFC = 33 when the data of (1,0,0,4,1) is received together with (0,0,0,1,0) (mentioned previously). Data is transmitted repeating TFC = 21, 33 after it.

But, we do not accept liability for the performance when there is no substitute TFC if one TFC is restricted.

(Ex.)In case that (1,0,0,8,1) is restricted to (0,0,0,0,0)

Since there is no setting of (1,0,0,9,1), data accumulates in the internal 8480 and cannot be outputted after being transmitted by (1,0,0,7,1). And 8480 will stopped.

3.7.11 SS_Interface Function

3.7.11.1 Outline

Function which is defined by this functional specification is Interface function which was defined to operate MX848000A(ControlPC) by via C-API. When using this function, please use it after creating application software with taking notice of definition item and restriction item which is described in this document. And when using this function, we recommend that it' s used by OS:WindowsNT/2000/XP, CPU:more than 500MHz and Memory:more than 128MB. (In Windows95/98, operation is unstable sometimes. Windows XP can be used PC/Firmware V5.11 or later.)

3.7.11.2 Creation Method

Please include in application software which creates the following four files in Mx848000¥Scenario¥include of Mx848000A.

1.ContorolApi.h
2.Primitive.h
3.Wcdma.h
4.DllFunc.h

Use the following file in Mx848000¥Senario¥Compiler of Mx848000A at build application software.

ControlAPI.lib

When implementing the created application software, the following six .dll in Mx848000¥Exe are needed.

1.ControlAPI.dll
2.Layer.dll
3.Layer3.dll
4.Scenalib.dll
5.Shared.dll
6.Sock.dll

3.7.11.3 Function List

The following functions are made arrangement in SS_Interface.

Name of function	Outline explanation
void SS_Run(void)	Starts up MX848000
void SS_Exit(void)	Ends MX848000
int SS_Start(char*)	Load of parameter file(.pm1 file) Establishes communication with MD8480A/B.
int SS_Stop(void)	Stops scenario during execution.
Int SS_ScenarioExec(char*,void*,int)	Executes scenario.
int SS_GetStatus(int*)	Get scenario execution status.

3.7.11.3.1 List of defined value

define	value	Explanation
TRUE	1	True (Finish normally)
FALSE	0	False (Finish abnormally)
FILE_NOT_EXIST	-1	Error at reading file.

3.7.11.3.2 Example of using function

Pattern 1

SS_Run()->SS_Start()->SS_ScenarioExec()
×10times->SS_Stop()->SS_Exit()

Pattern 2

SS_Run()->SS_Start()->SS_Status()->SS_ScenarioExec()->SS_Status()->SS
_Stop() ->SS_Status()-> SS_Exit()

Pattern 3

SS_Run()->SS_Start()->SS_ScenarioExec()->SS_Stop()->SS_Exit()
×100
times

Pattern 4

SS_Run()->SS_Start()->SS_ScenarioExec()->SS_Stop()->Sleep(100)
->SS_Exit()
×100 times

Pattern 5

SS_Run()->SS_Start()->SS_ScenarioExec()->SS_Stop() from another thread
->SS_Exit()
Loop in scenario

Notes) Please use it after creating another thread to end the scenario during execution. Do not stop from Stop button of MX848000A. (It will be freeze.).

3.7.11.4 Functional details

Note) When using this function, since there is the following restriction item, please be pay attention to use. In here, it describes about the whole restriction items. And since there is a restriction item in each function, please refer to remarks.

When using SS_Interface, do not start up another application software. (Especially, real time monitor). Operation will be unstable.

When starting up Mx848000A from SS_Interface, do not operate MX848000A by manual. (START button, Information Read button etc.). It will be freeze.

In case of executing test of pattern 4(example of using function), please insert Sleep() more than 100[msec] (At Win2000 CPU500MHZ/Memory128MB) after SS_Stop() definitely. This value is changed by performance of PC in use. And when executing continuous test of pattern 4 more than 100 times in Win95/98, it will stop in the middle of the test.

When executing SS_Start() if the status that EtherCable is off or SerialNumber isn’ t input correctly, it will be freeze. (Please pay attention when using Win95/98.)

<table>
<tr><td>Functional name</td><td colspan="4">INT SS_Run(VOID)</td></tr>
<tr><td>Functional summary</td><td colspan="4">Starts up Mx848000A. (The same as executing Mx848000.EXE.)</td></tr>
<tr><td></td><td>Type</td><td>Name of argument</td><td>Description</td><td>Input/Output</td></tr>
<tr><td>Argument</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>Type</td><td>Function value</td><td colspan="2">Description</td></tr>
<tr><td>Function value</td><td>INT</td><td>TRUE
FALSE</td><td colspan="2">Finish normally
Finish abnormally</td></tr>
<tr><td colspan="5">Functional details</td></tr>
<tr><td colspan="5">• Starts up Mx848000A. MX848000 starts up without screen display. (Minimization status)
• This function doesn’ t finish until starting process of Mx848000A is completed.
• When executing while Mx848000A has been already started by via C-API, function returns to FALSE. When executing while Mx848000A is started by manual, function returns to TRUE.</td></tr>
<tr><td>Remarks</td><td colspan="4">Please execute this function only one time before executing another SS-function.</td></tr>
</table>

Functional name	INT SS_Exit (VOID)			
Functional summary	Exits Mx848000A. (The same as pushing EXIT button.)			
	Type	Name of argument	Description	Input/Output
Argument				
	Type	Function value	Description	
Function value	INT	TRUE FALSE	Finish normally Finish abnormally	
Functional details				
• Exits Mx848000A. • This function doesn’ t finish until termination process of Mx848000A is completed. • When executing by status that Mx848000A isn’ t started up, function returns to FALSE.				
Remarks	Use this function after executing SS_Stop().			

Functional name	INT SS_Start (CHAR*Filname)			
Functional summary	Load parameter file (*.pm1).			
	Type	Name of argument	Description	Input/Output
Argument	CHAR *	Filename	Name of parameter file with path. (255 bites)	Input
	Type	Function value	Description	
Function value	INT	TRUE FALSE FILE_NOT_EXIST	Finish normally Finish abnormally Error at reading file.	
Functional details				
• This function reads parameter file and establish communication with MD8480. (Open socket.) • This function doesn’ t finish until communication with MD8480 is established. • When executing by status that Mx848000A isn’ t started up and SS_Start() has been already executed, function returns to FALSE. • When specifying NULL to Filename, road of File isn’ t implemented. (Stay as default).				
Remarks	Before executing this function, please confirm whether EhterCable is connected definitely and serial number has been already input. Especially, please pay attention in case of using Win95/98. (Processing can’ t be re-turned.)			

Functional name	INT SS_Stop (VOID)			
Functional summary	Stops scenario during execution. (The same as STOP button.)			
	Type	Name of argument	Description	Input/Output
Argument				
	Type	Function value	Description	
Function value	INT	TRUE FALSE	Finish normally Finish abnormally	
Functional details				
• Stop scenario during execution. • This function doesn’ t finish until scenario is stopped during execution. • In case of executing when scenario doesn’ t under execution, function returns to FALSE.				
Remarks	Executes this function after executing SS_ScenarioExec(). Right after executing this function, when executing SS_Start(), insert more than Sleep=more than 100msec. (It’ s changed by Spec of PC).			

Functional name	INT SS_ScenarioExec (CHAR *Filename,VOID *Param, INT ParamLength)			
Functional summary	Starts operation of simulation.			
	Type	Name of argument	Description	Input/Output
Argument	CHAR * VOID * INT	Filename Param ParamLength	Name of file with path.(127 bytes) Parameter Argument(Structure pointer) Length of Param(Structure)(Maximum 2048 bytes)	Input Input/Output Input
	Type	Function value	Description	
Function value	INT	TRUE FALSE FILE_NOT_EXIST	Finish normally Finish abnormally Error at reading file.	
Functional details				

- This function reads the specified scenario by Filename and executes the scenario.
- This function is able to execute continuously.
- Before executing this function, please execute SS_Start(). After executing SS_Stop(), when executing before SS_Start(), function returns to FALSE.
- At executing, transmit Argument (*Param) to ScenarioMain or receive (*Param) from ScenarioMain.

EX 1)

```
 CPHY_BTS_MEAS_PAR                    measPar;

 memset( &measPar, 0, sizeof(CPHY_BTS_MEAS_PAR) );
 double                          BLER;
 INT                             CRC_NG;
 INT                             Total;
 measPar.Mode    = 2;
 measPar.Count = 1234;

 SS_ScenarioExec( "FileName", &measPar, sizeof(CPHY_BTS_MEAS_PAR) ) );
 Total = measPar.Result.BLER_Crc.Total;
 CRC_NG = measPar.Result.BLER_Crc.CrcNG;
 BLER = (double )CRC_NG / (double )Total;
}                                                  ⇑ Scenario of application software
-------------------------------------------------------------------------------------------------------------------
                                                    ⇓Scenario of Mx848000A
INT ScenarioMain(LPVOID Param)
{
 CPHY_BTS_MEAS_PAR       *measParam;

 measParam = Param;

 BtsMeasure( UNIT_BTS1, U_DCH, 0, measParam, NO_TIMEOUT );
 }
=========================================================================
```

EX 2)

```
 INT ScenarioMain(LPVOID Param){
                         •
                         •
                         •
 strcpy(Param, "StartSimulate");
 return(0);
 }                                                                          ⇑ Scenario of Mx848000A
-------------------------------------------------------------------------------------------------------------------
 CHAR                          prmBuf[2048];                      ⇓Scenario of application software
 SS_ScenarioExec( " FileName" , prmBuf, 14 );
```

Remarks	If not to use Param, please specify Argument as follows definitely. SS_ScenarioExec(" FileName" , NULL, 0);

Functional name	INT SS_GetStatus (INT *status)			
Functional summary	Return scenario operation status.			
	Type	Name of argument	Description	Input/Output
Argument	INT *	Status	Return scenario operation status.	Output
	Type	Function value	Description	
Function value	INT	TRUE FALSE	Finish normally Finish abnormally	
Functional details				

• Sets status of MX848000 (or SS) to Argumentstatus and returns. Function value returns TRUE.
• If executing when not to execute SS_Run(), function returns to FALSE.

Bit31Bit3:Reserved (0 fixed)
Bit2:Scenario interruption flag
Interruption (1)
It is referred to as 1 when a scenario is interrupted for SS_Stop() of another thread. It’ s cleared by next SS_ScenarioExec() call.
* Processing of SS_ScenarioExec() finished and even if MD8480 executes SS_Stop() during operation, Bit2 doesn’ t become 1.

Bit1:Scenario execution flag ()
Stops(0)/Operation(1)
Operation:Processing of SS_ScenarioExec() by executing scenario is executing.
Stops:Although MD8480 is operating by executing scenario, the status that processing of SS_ScenarioExec() is finished.

Bit0:MD8480A operation flag
Stops(0)/Operation(1)
Operation:MD8480 is operated by executing scenario.
Stops:The status that SS_Stop() was executed and scenario isn’ t implemented even once.

EX)
• SS_Run->SS_Start->SS_Status(0) ->SS_ScenarioExec->SS_Status(1) ->SS_Stop->SS_Status(0) -> SS_Exit
• SS_Run->SS_Start->SS_ScenarioExec(Under execution)->SS_Stop(Another thread) ->SS_Status(4) -> SS_Start->SS_ScenarioExec->SS_Status(Another thread (3)) -> SS_Exit-> SS_Status=FALSE

Remarks	

3.7.12 TM Segmentation Function

3.7.12.1 Outline

TM Segmentation Function is a function which segments RLC SDU into each RLC PDU size (Downlink) or makes RLC SDU to reassemble RLC PDU (Uplink), when the RLC entity is configured as transparent mode.

3.7.12.2 The restrictions on mounting

In 3GPP specifications, segmentation and reassembly are performed in RLC. But in MD8480 they are performed in MAC.

3.7.12.3 Functional details

3.7.12.3.1 In the case of Downlink

When TM Segmentation Function is used, it is assumed that 1280 bit RLC_TR_DATA_REQ is transmitted by TE as shown in Fig. 3.39. In this case, a setup of Transport Block number=2 and Transport Block Size = 640 is assumed to be applied in MAC and physical layer. It becomes 1280 bit MAC_DATA_REQ by RLC, and transmits to MAC. The data is segmented to 640 bits in MAC. and PHY_DATA_REQ multiplexed in 2x640 bit is transmitted to PHY.

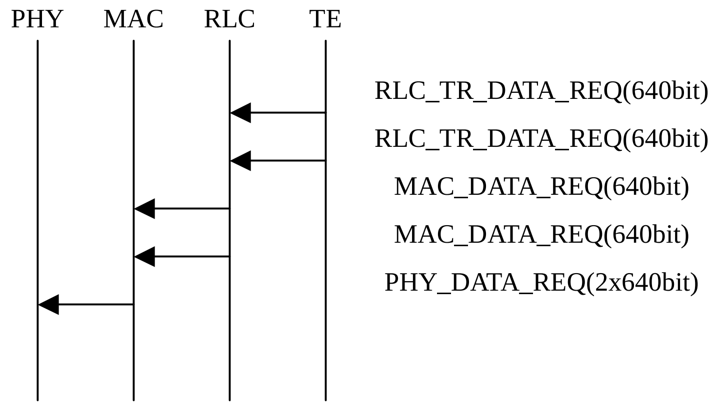


Fig. 3.39 Sequence when using TM Segmentation (Downlink)

*When not using a TM Segmentation function, since 1280 bit data cannot be segmented, as shown in Fig. 3.40, it is necessary to transmit two 640 bit RLC_TR_DATA_REQ, RLC transmits these 2 primitives to MAC, and transmits PHY_DATA_REQ multiplexed in 2x640 bit by MAC to PHY.

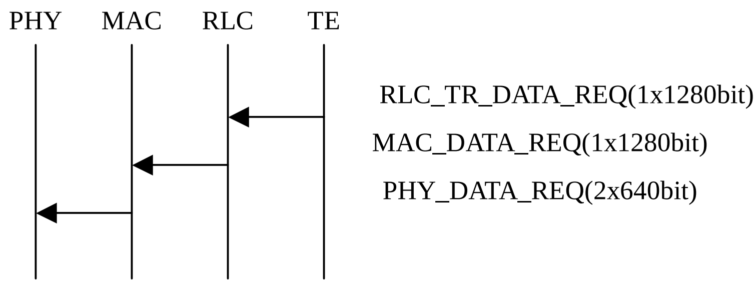


Fig. 3.40 Sequence when not using TM Segmentation (Downlink)

3.7.12.3.2 In the case of Uplink

When MAC receives 2x640 bit data from PHY,

it reassembles the data to 1280 bit MAC_DATA_IND, and the primitive is provided to RLC. RLC transfers the data to TE as 1280 bit RLC_TR_DATA_IND.

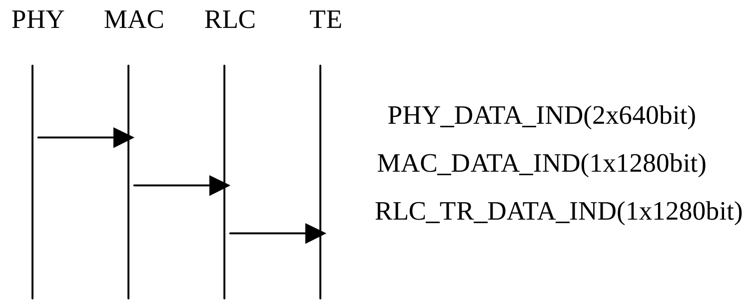


Fig. 3.41 Sequence when using TM Segmentation (Uplink)

*When TM Segmentation function is not used, 2x640bit PHY_DATA_IND as shown in Fig. 3.42, is not reassembled in MAC.
MAC provides two MAC_DATA_INDs to RLC. In this case, each data of the primitive has 640 bit length. RLC transfers two RLC_TR_DATA_INDs to TE.

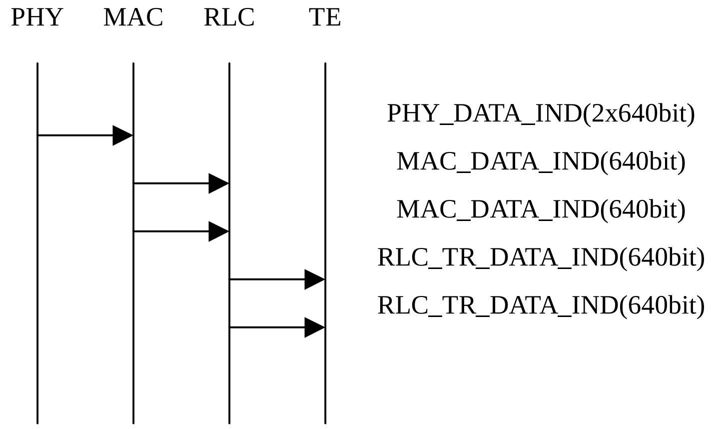


Fig. 3.42 Sequence figure when not using TM Segmentation (Uplink)

3.7.12.4 How to set

Please follows the setup below before SimulatorStart(0 NO_TIMEOUT);.

```
DispChannelState();
CrlcStartPar.Flags = ENABLE_TM_BITMODE; /* TM_BITMODE ON */
SimulatorStart(0,NO_TIMEOUT );
ReInitializeParameter();
```

PDU size after segmentation is set to PU_LengthTM variable in CRLC_CONFIG_PAR structure. In addition, a PU_LengthTM variable is 16 bit.

bit15: TM Segmentation(1: use 0:not use)
bit0-11: PDU size(1-3840)

For example, in case that PDU size is equal to 640bit (0x280), the setting value of PU_LengthTM is 0x8000 | 0x0280.

```
CrlcConfigDTCH.PU_LengthTM = (0x8000 | 0x0280);
CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,DTCH,0,&CrlcConfigDTCH,
TE ,NO_TIMEOUT);
```

*TE configuration in the case of using the above-mentioned setup

NumOfTB and TBS variables in CTE_CONFIG_PAR structure need changing. For example, when two 640 bit data are transmitted, the data corresponds to on 1280 bit size.Therefore it need to be set to NumOfTB=1 and TBS=1280. The example of description in a scenario is shown below.

```
CteConfigDtch.TeType   = TE_TYPE_NONE;
CteConfigDtch.Rate     = 0;
CteConfigDtch.TTI      = 2;
CteConfigDtch.NumOfTB = 1;
CteConfigDtch.TBS      = 1280;
CteConfigDtch.Frame    = RLC_TR_DATA_REQ;
CteConfigDtch.Layer    = RLC;
CteConfig(DTCH, 0, &CteConfigDtch, NO_TIMEOUT);
```

3.7.13 AM-RLC Asymmetric payload size setting functionality

3.7.13.1 Outline

This function enables MD8480 to send and receive the data specified in 3GPP TS34.108 section 6.10.2.4.1.58.

3.7.13.2 Outline of this function

According to 3GPP TS34.108 clause 6.10.2.4.1.58, logical channel parameters of RLC for Streaming / Unknown / UL: 16 DL: 64 kbps / PS RAB are shown below.

For UL 16kbps

RLC	Logical channel type	DTCH
	RLC mode	AM
	Payload sizes, bit	320
	Max data rate, bps	16000
	AMD PDU header, bit	16

For DL 64kbps

RLC	Logical channel type	DTCH
	RLC mode	AM
	Payload sizes, bit	640
	Max data rate, bps	64000
	AMD PDU header, bit	16

To achieve above parameters, MD8480A/B provides some functions shown below.

a) On AM RLC (DTCH only), payload sizes for downlink and uplink are able to be set individually.

b) On AM RLC (DTCH only), a maximum payload size is extended to 640bit (It was 624bit before).

c) On AM RLC (DTCH only), a maximum payload size is extended to 640bit when ciphering is performed (It was 320bit before).

3.7.13.3 Setting method of this function

Three variables are added in CRLC_CONFIG_PAR structure which is used as a parameter for CrlcConfig ().

```
typedef struct {
    USHORT     NumOfPUs;              /* Number of PUs               */
    USHORT     PU_LengthTM;           /* RLC PU Length(TM)           */
    USHORT     PU_LengthUM;           /* RLC PU Length(UM)           */
    USHORT     PU_LengthAM;           /* RLC PU Length(AM)           */
    USHORT     TxWindowSize;          /* AM Tx Window Size           */
    USHORT     RxWindowSize;          /* AM Rx Window Size           */
    USHORT     PollWindow;            /* AM Poll Window              */
    USHORT     PollPU;                /* AM Poll PU                  */
    USHORT     PollSDU;               /* AM Poll ADU                 */
    USHORT     SDUSize;               /* SDU Size                    */
    USHORT     MaxDat;                /* MAX Dat                     */
    USHORT     MaxRst;                /* MAX Rst                     */
    USHORT     MaxMRW;                /* MAX MRW                     */
    USHORT     TimerPoll;             /* Timer Poll                  */
    USHORT     TimerPollProhibit;     /* Timer Poll Prohibit         */
    USHORT     TimerEPC;              /* Timer EPC                   */
    USHORT     TimerDiscard;          /* Timer Discard               */
    USHORT     TimerPollPeriodic;     /* Timer Poll Periodic         */
    USHORT     TimerStatusProhibit;   /* Timer Status Prohibit       */
    USHORT     TimerStatusPeriodic;   /* Timer Staus Periodic        */
    USHORT     TimerRst;              /* Timer Rst                   */
    USHORT     TimerMRW;              /* Timer MRW                   */
    USHORT     PollingTriggers;       /* Polling Trigger             */
    USHORT     StatusTriggers;        /* Status Trigger              */
    USHORT     SDUDiscardMode;        /* SDU Discard Mode            */
    USHORT     UseAsymPUSize;         /* PDU_Length Asymmetry Mode */
    USHORT     PU_LengthDL;           /* Downlink PU Length          */
    USHORT     PU_LengthUL;           /* Uplink PU Length            */
    USHORT     Reserve [4];           /* for future use              */
} CRLC_CONFIG_PAR;
```

To set payload size individually downlink and uplink for an AM RLC, set a UseAsymPUSize value to 1 and set PU_LengthDL and PU_LengthUL values for payload sizes of downlink and uplink. If the UseAsymPUSize value is equal to 1, the PU_LengthAM value, which is conventional value in payload size, is not referred. If the UseAsymPUSize value is equal to 0, neither PU_LengthDL nor the PU_LengthUL value will be referred and PU_LengthAM value will be used to downlink and uplink. Since UseAsymPUSize variable is not initialized automatically like other members of CRLC_CONFIG_PAR structure, when same payload size in downlink and uplink is set with conventional method after setting UseAsymPUSize = 1, UseAsymPUSize needs to be initialized to 0. Maximum value of RLC AM payload size is 640 regardless of UseAsymPUSize value and ciphering operation.

3.7.13.4 An example of Scenario Description

```
CrlcConfigDTCH.UseAsymPUSize = 1;
CrlcConfigDTCH.PU_LengthUL = 320;
CrlcConfigDTCH.PU_LengthDL = 640;
CrlcConfig(UNIT_BTS1,CRLC_AM_ESTABLISH,DTCH,0,&CrlcConfigDTCH,TE ,NO_TIMEOUT);
```

On the above example, DTCH #0 is established with 640bit DL payload size and 320bit UL payload size.

```
CrlcConfigDTCH.UseAsymPUSize = 1;
CrlcConfigDTCH.PU_LengthUL = 320;
CrlcConfigDTCH.PU_LengthDL = 640;
CrlcConfigDTCH.PU_LengthAM = 624;
CrlcConfig(UNIT_BTS1,CRLC_AM_ESTABLISH,DTCH,0,&CrlcConfigDTCH,TE ,NO_TIMEOUT);
```

On the above example, even if PU_LengthAM is set, it will be ignored and DTCH #0 will be established with 640bit DL payload size and 320bit UL payload size. Because UseAsymPUSize is equal to 1.

3.7.13.5 Restriction items

This function is valid on RLC AM only. When UseAsymPUSize value is set to 1 on RLC TM or UM, the behavior is not warranted. This function is valid on DTCH. The behavior is not guaranteed when UseAsymPUSize is set to a value other than 0 on other bearers, such as DCCH.

3.7.14 Specifying feature for a start timing of TE transmission/the number of transmission

3.7.14.1 Outline

This is a feature to specify a start timing of TE data transmission in downlink and the number of transmission.

3.7.14.2 Outline of this feather

You can send a fixed or PN9 pattern data from TE in downlink with calling CteConnect( ) even in the existing version. Using this feature together allows you to specify a start timing of downlink transmission from TE and the number of transmission.
Execute the following procedure before calling CteConnect( ).

```
TeActTime = XXX;
TeTtiCounter = YYY;
CteConnect( );
```

By the procedure above, TE on MD8480 starts sending downlink data at the CFN timing specified by TeActTime, and the transmission is repeated the number of times specified in TeTtiCounter. The followings are the details for the variables.

Table 3.15 Details of the variables

Variable name	Contents	Initial value	Range
TeActTime	A variable to specify a start timing of downlink transmission from TE. Set your intended CFN value to this.	ACTIVATE_NOW	0--255, ACTIVATE_NOW
TeTtiCounter	A variable to specify the number of transmission from TE. The unit is TTI. Duration time of transmission and the number of transmitted transport block provided by this feature are as follows. Duration time = TTI * TeTtiCounter The number of transport block = TTI * NumOfTB specified in CteConfig( ) Transmission will stop when the number of transmission expires.	0 (=infinite transmission)	0--65535

You can use each variable separately. E.g. when you start sending data infinitely at 200 of CFN, please describe your scenario as follows.

```
TeActTime = 200;
TeTtiCounter = 0; /* = Specify infinite transmission */
CteConnect( );
```

You don’ t need to set 0 to TeTtiCounter, if you never change TeTtiCounter value after you click START button on Mx848000.

Note) MD8480 will behave same as the existing version, if you call CteConnect( ) without changing the values on the variable from initial ones.

3.7.14.3 An example of Scenario Description

An example of scenario description on DCCH is as follows.

Ex) In case of sending data twice 2 seconds later

Query a CFN value to MD8480 and add 200 frames offset to the value

```
TeActTime = ((BtsReadCFN(UNIT_BTS1,NO_TIMEOUT) + 200) % 256) & 0xFFFC;
TeTtiCounter = 2;
```

Specify the number of transmission from TE in downlink

```
CteConfigDcch.TeType   = TE_TYPE_NONE;
CteConfigDcch.Rate     = 0;
CteConfigDcch.TTI      = 4;
CteConfigDcch.NumOfTB = 1;
CteConfigDcch.TBS      = 148;
CteConfigDcch.Frame    = RLC_TR_DATA_REQ;
CteConfigDcch.Layer    = RLC;
memcpy(CteConfigDcch.Data,DCCH_DATA,sizeof(DCCH_DATA));

CteConfig (DCCH, 0, &CteConfigDcch, NO_TIMEOUT);
CteConnect(DCCH, 0, TE_PORT_FIXDATA, TE_PORT_USERDATA, CALL_FROM_AIR, (CHAR *)0, NO_TIMEOUT);
```

Specify the fixed pattern transmission in downlink

3.7.14.4 Image of PC trace

An example image of PC trace on this feature is as follows.

			<---		CTE_CONFIG_REQ	D DCCH	0	...	00:00:03.31
			--->		CTE_CONFIG_CNF	D DCCH	0	...	00:00:03.31
			<---		CTE_CONN_REQ	D DCCH	0	...	00:00:03.32
			--->	1	CTE_CONN_CNF	D DCCH	0	...	00:00:03.34
		<---			RLC_TR_DATA_REQ	D DCCH	0	...	00:00:05.30
	<---				MAC_DATA_REQ	D DCCH	0	...	00:00:05.31
<---				1	PHY_DATA_REQ	D DCH	0	...	00:00:05.31
		<---			RLC_TR_DATA_REQ	D DCCH	0	...	00:00:05.34
	<---				MAC_DATA_REQ	D DCCH	0	...	00:00:05.35
<---				1	PHY_DATA_REQ	D DCH	0	...	00:00:05.35

```
Message Data(Length=67)
Sender:50 Receiver:10 Type:9052 Type2:00 Channel:45 CH_No:00 Opt1:0002 Opt2:001C
01 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 3.43 Image of PC trace on this feature

3.7.14.5 Caution

1) This feature is available in PC:V5.11b, FW:V5.11b or later versions. These versions can be used for customers who make a contract for the support service of MD8480.

2) TeActTime and TeTtiCounter are global variables. You don’ t need to declare them in your scenario.

3) If you don’ t change the values explicitly for TeActTime or TeTtiCounter, the values remain in the variables as they are.

4) When you use this feature, set either TE_PORT_FIXDATA or TE_PORT_PN9 to the third parameter on CteConnect( )(DownLinkPort variable). Please set the same value as the third parameter or TE_PORT_USERDATA to the fourth parameter(UpLinkPort variable). The action is not guaranteed when the other values are specified.

5) Please set a value to TeActTime so that it may be multiples of TTI specified in CteConfig( ).

6) This feature can be applied to logical channels only in W-CDMA.

7) When the number of transmission expires, please never fail to call CteDisconnect( ) and CteRelease( ) to the logical channel.

3.7.15 CFN Timing Indication feature

3.7.15.1 Outline

This is a feature to notify a timing to a scenario when MD8480 reaches the specific CFN value.

3.7.15.2 Outline of this feature

If you call a scenario library of CphyCfnInd( ) with a specific CFN value, CPHY_CFN_IND is sent from PHY layer to RRC layer at the CFN timing. Using this library with WcdmaRcvControl( ) allows you to start other processing in your scenario when CFN reaches the timing.

Sample procedure of this feature is as follows.

1) Call CphyCfnInd( ) with a specific CFN value
2) Wait to receive CPHY_CFN_IND using a scenario library of WcdmaRcvControl( )
3) Terminate waiting procedure for the reception when CPHY_CFN_IND is received, and execute the following processing.

3.7.15.3 Details of scenario libraries

<table>
<tr><td>Function name</td><td colspan="4">INT CphyCfnInd(INT BtsNo, INT IndCFN, INT SkpCfnCycle, INT Timeout)</td></tr>
<tr><td>Feature summary</td><td colspan="4">Request PHY to notify a scenario of a specified timing at the CFN</td></tr>
<tr><td></td><td>Type</td><td>Argument name</td><td>Description</td><td>I/O</td></tr>
<tr><td>Argument</td><td>INT
INT
INT
INT</td><td>BtsNo
IndCFN
SkpCfnCycle
Timeout</td><td>BTS No
Indicated CFN
Skipped CFN cycle
Timeout value</td><td>Input
Input
Input
Input</td></tr>
<tr><td></td><td>Type</td><td>Function value</td><td colspan="2">Description</td></tr>
<tr><td>Function value</td><td>INT</td><td>= 0
<> 0</td><td colspan="2">Normal ending
Abnormal ending (Timeout)</td></tr>
<tr><td colspan="5">Feature details</td></tr>
<tr><td colspan="5">1) Set UNIT_BTS1 to BtsNo.
2) Specify a CFN value in IndCFN, that is a timing which should be notified to your scenario.
The range of this parameter : 0-255 (A value over 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
3) Specify a skipped CFN cycle value in SkpCfnCycle until CFN reaches to a value of IndCFN.
The range of this parameter : 0-255 (A value over 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
4) Timeout
Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.</td></tr>
<tr><td>Remarks</td><td colspan="4"></td></tr>
</table>

Function name	INT WcdmaRcvControl(INT *BtsNo, INT *Frame, INT *Loch, INT *LochNo,CHAR *Message, INT Timeout)			
Feature summary	Receive control primitives in W-CDMA			
	Type	Argument name	Description	I/O
Argument	INT *	BtsNo	BTS No	Output
	INT *	Frame	Received primitive	Output
	INT *	Loch	Logical Channel	Output
	INT *	LochNo	Logical Channel Number	Output
	CHAR *	Message	Received data	Output
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	>= 0	Normal ending (Received data length)	
		< 0	Abnormal ending (Timeout)	

Feature details

1) BtsNo is not used at the moment.
2) The following is returned in Frame.
  CPHY_CFN_IND : notifying a CFN timing specified by CphyCfnInd( )
3) 0 is returned in Loch.
4) 0 is returned in LochNo.
5) Null pointer is stored in Message.
6)Timeout
  Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

3.7.15.4 An example of Scenario Description

1) In case of sending a layer 3 message at 100 of CFN after 1 CFN cycle passes

```
CphyCfnInd( UNIT_BTS1, 100, 1, NO_TIMEOUT );
```

Wait for the notification of the CFN(=100) with CPHY_CFN_IND

Specify a CFN to be notified and a skipped CFN cycle

```
for(;;){
   WcdmaRcvControl( &BtsNo, &Frame, &Loch, &LochNo, RcvData,
   NO_TIMEOUT );
   if(Frame == CPHY_CFN_IND){
      break;
   }
}
```

Send this message at the timing later than CFN(=100)

```
{
   UCHAR    SndData[] = {0x00,0x01,0x02,0x03};
   SndMessage(UNIT_BTS1,RLC_UM_DATA_REQ,D_CCCH,0,SndData,sizeof(SndData));
}
```

2) In case of sending a layer 3 message 2 seconds later

Query a CFN value to MD8480 and add 200 frames offset to the value

```
CFN = ((BtsReadCFN(UNIT_BTS1,NO_TIMEOUT) + 200) % 256) & 0xFFFC;
CphyCfnInd( UNIT_BTS1, CFN, 0, NO_TIMEOUT );
```

Wait for a CFN timing to be notified with CPHY_CFN_IND

Specify a CFN value, which is expected to be notified

```
for(;;){
   WcdmaRcvControl( &BtsNo, &Frame, &Loch, &LochNo, RcvData, NO_TIMEOUT );
   if(Frame == CPHY_CFN_IND){
      break;
   }
}

{
   UCHAR   SndData[] = {0x00,0x01,0x02,0x03};
   SndMessage(UNIT_BTS1,RLC_UM_DATA_REQ,D_CCCH,0,SndData,sizeof(SndData));
}
```

Send this message at the timing when a period corresponding to 200 frames

3.7.15.5 Image of PC trace

An example image of PC trace on this feature is as follows.

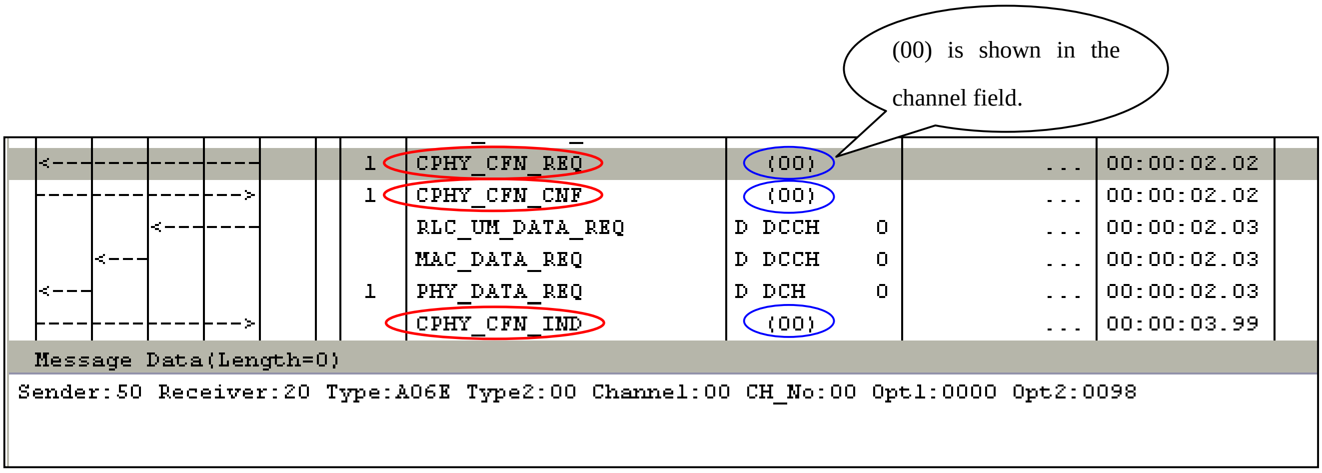


Figure 3.44 Image of PC trace on this feather

3.7.15.6 Caution

1) This feature is available in PC: V5.11b, FW: V5.11b or later versions. These versions can be used for customers who make a contract for the support service of MD8480.

2) If a second CphyCfnInd( ) is called before CPHY_CFN_IND corresponding to the first CphyCfnInd( ) has occurred then CPHY_CFN_IND corresponding to first CphyCfnInd( ) will not occur.(The Processing of the first CphyCfnInd( ) is overwritten by the second processing.)

3.7.16 RLC Stop, RLC Continue function

3.7.16.1 Outline

This feature supports “RLC Stop and RLC Continue function” specified in 3GPP TS25.322 clause 9.7.6.

3.7.16.2 Outline of this function

The following is 3GPP TS25.322 clause 9.7.6 “RLC Stop, RLC Continue function for acknowledged and unacknowledged mode”.

The upper layer may stop an RLC entity.
When an RLC entity is stopped, the RLC timers are not affected.
When a RLC entity is stopped by upper layers, the RLC entity shall:

- not submit any RLC PDUs to lower layer or receive any RLC PDUs;
- delay triggered Polling functions or status transmissions until the RLC entity is continued.

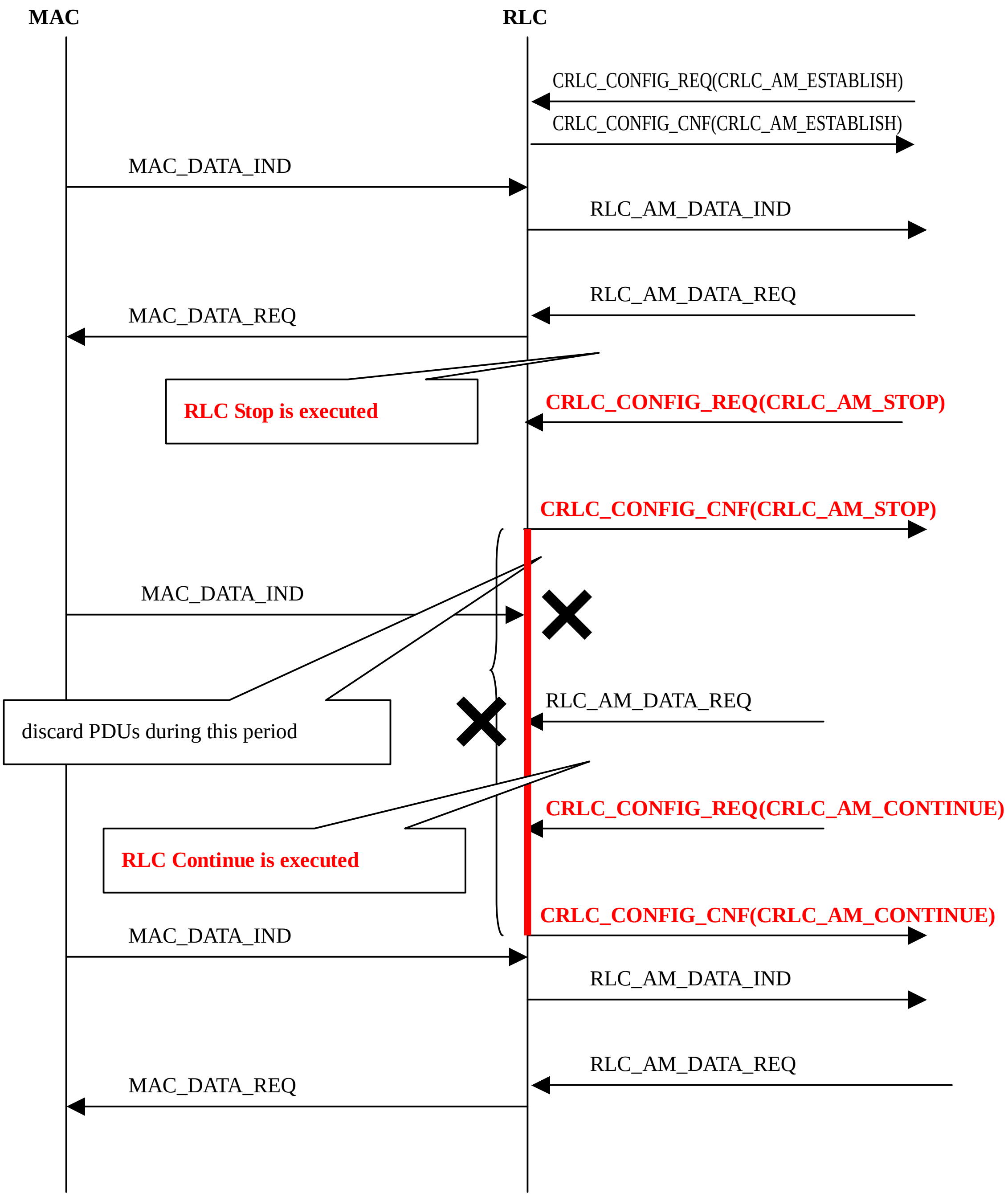
When a RLC entity is continued by upper layers, the RLC entity shall:

- if the RLC entity is stopped:
  - continue the data transmission and reception;
  - process the triggered Polling functions and status transmissions.
- otherwise, if the RLC is not stopped:
  - take no action.

I.E, RLC Stop function is the feature to stop submitting and receiving PDUs on some RLC entity, and RLC Continue function is the feature to restarts the RLC entity, which already stopped.

3.7.16.3 Sequence image

The following is sequence image of RLC Stop and RLC Continue.



3.7.16.4 Setting method of this function

In order to execute this function, one of parameters shown below is used to second argument of CrlcConfig().

Table 3.16 parameters

Parameter name	Description
CRLC_AM_STOP	Stop AM RLC
CRLC_UM_STOP	Stop UM RLC
CRLC_AM_CONTINUE	Continue AM RLC
CRLC_UM_CONTINUE	Continue UM RLC

When execute RLC Stop or RLC Continue, please use above parameter same as establishment of RLC. In this case, please select CRLC_AM_STOP or CRLC_AM_CONTINUE for RLC AM entity, and CRLC_UM_STOP or CRLC_UM_CONTINUE for RLC UM entity. For RLC TM entity, neither RLC Stop nor RLC Continue can execute.

3.7.16.5 An example of Scenario Description

To execute RLC stop to AM DTCH#0, please describe following sentence into a scenario.

```
CrlcConfig(UNIT_BTS1,CRLC_AM_STOP,DTCH,0,&CrlcConfigDTCH,TE ,NO_TIMEOUT);
```

To execute RLC continue to AM DTCH#1, please describe following sentence into a scenario.

```
CrlcConfig(UNIT_BTS1,CRLC_UM_CONTINUE,DTCH,1,&CrlcConfigDTCH,TE ,NO_TIMEOUT);
```

3.7.16.6 Restriction items

1) It takes about 10ms until RLC Stop or RLC Continue is effected from executing CrlcConfig() with RLC Stop or RLC Continue.

2) If the function CrlcSuspend(), CrlcConfig(RLC_**_STOP), CrlcResume() is executed in order, for some bearer, the bearer will transmit down link data which had already received before RLC Stop, in spite of the bearer is stopped.

3.7.17 W-CDMA CSD

This chapter describes the test method of W-CDMA CSD.
In order to perform a W-CDMA CSD examination, not only ISDN Option but also a support service contract and W-CDMA CSD Option are required. You can find if ISDN Option is configured to check the 10th card from the left. Refer to the follow-ing how to confirm that support service contract and the W-CDMA CSD option are installed in your Signalling Tester.

How to confirm support service contract,

Confirm that the version you are using is include in the option names that are shown in the System Window in the Control Software when Informa-tion Read button is clicked.

How to confirm W-CDMA CSD Option,

Confirm that the string “ W-CDMA CSD” is appeared in the System Win-dow in the Control Software when Information Read button is clicked.

(Note) Please contact MD8480A-G-support@zy.anritsu.co.jp if you need the detailed information regarding W-CDMA CSD option installation.

3.7.17.1 Connection of equipment

Please carry out like “ E.5.3.1 Connection of equipment” .

3.7.17.2 Corresponding mode

In the examination of W-CDMA CSD, the asynchronous data transmission in non-transparent mode is supported.

3.7.17.3 User rate

The user rates of 14.4 kbit/s, 28.8kbit/s, and 57.6kbit/s are supported in the examination of W-CDMA CSD at asynchronous and non-transparent mode.

3.7.17.4 Corresponding higher protocol

Only PPP specified in RFC1662 format is supported. As the higher layers than PPP are “ Don't care” , the data transmission by some application proto-cols using IP Packet (ping, FTP, and HTTP) is possible.

3.7.17.5 Serial interface

"Serial cable PC-PC connection" equipped in WindowsNT4.0 and "the communications cable between two computers" of Windows2000 are used.
* Please refer to "E. the 5.3.2 Windows2000 Server Setting Procedure" with regard to the Windows2000 server setting procedure.

3.7.17.6 Execution order of the sample scenario

This chapter explains a concrete operation procedure including the execution order of a sample scenario.

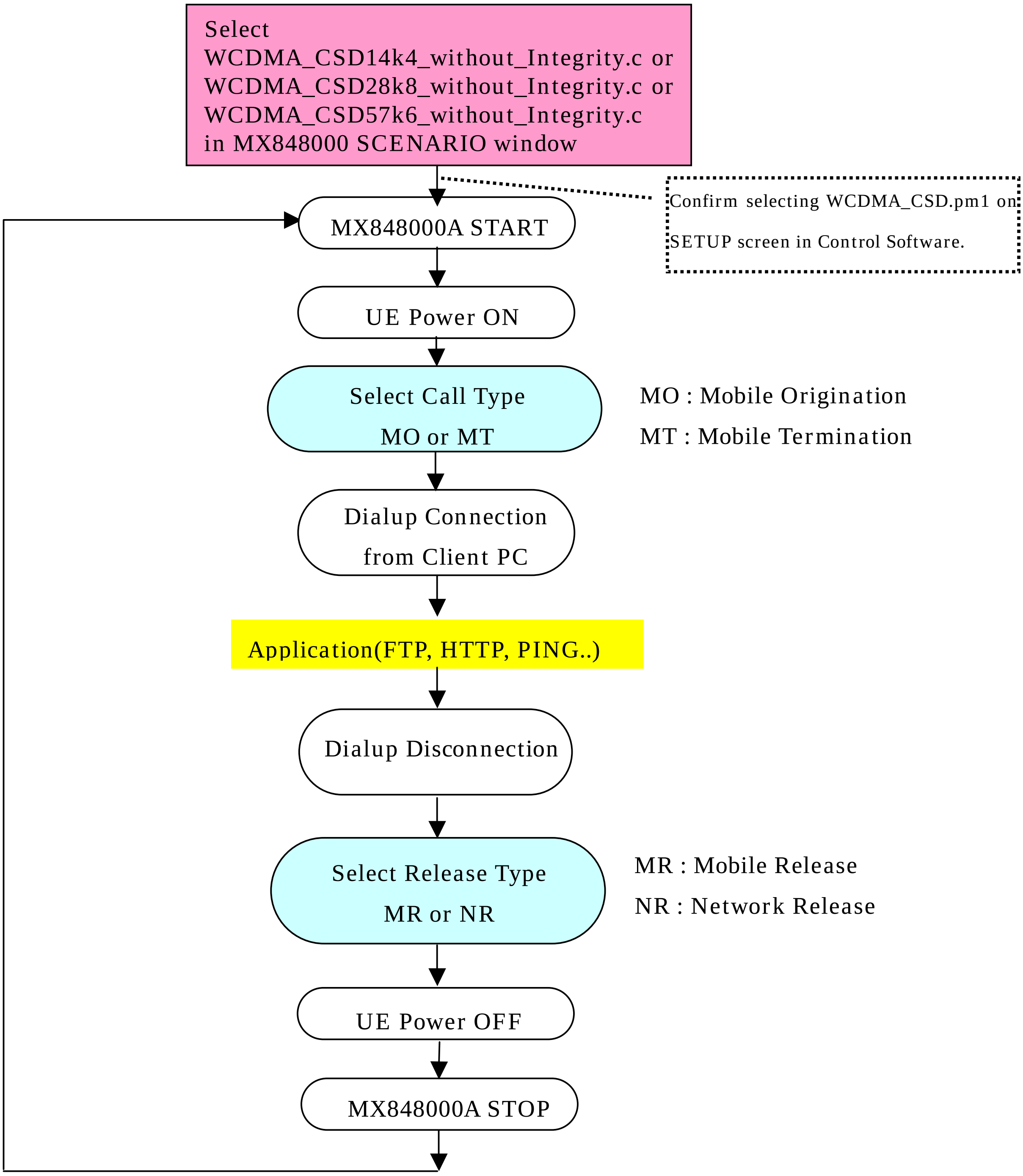
The following setups shall be completed before the test.

* "Serial cable PC-PC connection" or "Communications cable between two computers" have been set up for the COM port on Server PC.
* Modem driver has been set up for the COM port on Client PC.
* Network settings have been configured in Server and Client PC.
* RAS (Remote Access Server) has been set up on Server PC.
* Settings for dial up connection have been set up on Client PC.

Call origination/ Termination Procedure
(1) Start up Control Software on Control PC.
(2) Click [Load] in [SETUP] window and select [Open]->[OK] after clicking " WCDMA_CSD.pm1" . Set up TX Attenuator and RX Reference Level if needed (refer to E 2.3.4).
(3) Click the [SCENARIO] button and select
" WCDMA_CSD14k4_without_Integrity.dll"
or
" WCDMA_CSD28k8_without_Integrity.dll"
or
" WCDMA_CSD57k6_without_Integrity.dll"
in Scenario select window.
(4) Click the [START] button, then Signalling Tester will be initialized .
(5) Turn on the power of the UE.
(6) Confirm that Antenna mark which shows level OK is displayed on the UE Screen.
(7) Execute " MO" (dial up from the Client on UE side) or " MT" (dial up from the Client on NW side) by selecting in the dialog.
(8) Dialup from Client PC and login to RAS on the server PC.

note) WCDMA_CSD14k4_without_Integrity.c,
WCDMA_CSD28k8_without_Integrity.c,
WCDMA_CSD57k6_without_Integrity.c attached in Control software is a just sample scenario to explain how to write a scenario for per-forming W-CDMA CSD with UE, and does not guarantee the connection to the customer's UE. Please change the scenario according to the UE specification.

3.7.17.6.1 Flow of execution order of scenario



3.7.18 SFN Timing Indication function

3.7.18.1 Outline

This is a function to notify timing to a scenario when MD8480 reaches the specific SFN value.

3.7.18.2 Outline of this function

When you call a scenario library of CphySfnInd( ) with a specific SFN value, CPHY_SFN_IND is sent from PHY layer to RRC layer at the SFN timing. Using this function with WcdmaRcvControl( ) enables you to start other processing in your scenario when SFN reaches the timing.

Sample procedure of this function is as follows.

1) Call CphySfnInd( ) with a specific SFN value
2) Wait to receive CPHY_SFN_IND using WcdmaRcvControl( )
3) Terminate the procedure for waiting reception when CPHY_SFN_IND is received, and execute the following processing.

3.7.18.3 Details of scenario libraries

Function name	INT CphySfnInd(INT BtsNo, INT IndSFN, INT SkpSfnCycle, INT Timeout)			
Feature summary	Request PHY to notify a scenario of a specified timing at the SFN			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	IndSFN	Indicated SFN	Input
	INT	SkpSfnCycle	Skipped SFN cycle	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	= 0	Normal ending	
		<> 0	Abnormal ending (Timeout)	
Feature details				
1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo. SFN of the BTS that you specified is the object of this timing indication function. 2) Specify a SFN value in IndSFN, that is a timing which should be notified to your scenario. The range of this parameter : 0-4095 (A value over 4095 can be specified; in this case, the remainder value of division by 4096 is effective.) 3) Specify a skipped SFN cycle value in SkpSfnCycle until SFN reaches to a value of IndSFN. The range of this parameter : 0-15 (A value over 15 can be specified; in this case, the remainder value of division by 16 is effective.) 4) Timeout Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.				
Remarks				

Function name	INT WcdmaRcvControl(INT *BtsNo, INT *Frame, INT *Loch, INT *LochNo,CHAR *Message, INT Timeout)			
Feature summary	Receive control primitives in W-CDMA			
	Type	Argument name	Description	I/O
Argument	INT *	BtsNo	BTS No	Output
	INT *	Frame	Received primitive	Output
	INT *	Loch	Logical Channel	Output
	INT *	LochNo	Logical Channel Number	Output
	CHAR *	Message	Received data	Output
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	>= 0	Normal ending (Received data length)	
		< 0	Abnormal ending (Timeout)	
Feature details				
1) The following is returned in BtsNo. 0 : when 2) Frame is CPHY_CFN_IND BtsNo(UNIT_BTS1, UNIT_BTS2 or UNIT_BTS3) which specified in CphySfnInd : when 2) frame is CPHY_SFN_IND 2) The following is returned in Frame. CPHY_CFN_IND : notifying a CFN timing specified by CphyCfnInd( ) CPHY_SFN_IND : notifying a SFN timing specified by CphySfnInd( ) 3) 0 is returned in Loch. 4) 0 is returned in LochNo. 5) Null pointer is stored in Message. 6)Timeout Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.				

Remarks	

3.7.18.4 Example of scenario description

In case of sending a layer 3 message after SFN reaches 100.

```
CphySfnInd( UNIT_BTS1, 100, 0, NO_TIMEOUT );
for(;;){
    WcdmaRcvControl( &BtsNo, &Frame, &Loch, &LochNo, RcvData, NO_TIMEOUT );
    if(Frame == CPHY_SFN_IND){
        break;
    }
}
{
    UCHAR   SndData[] = {
        0x00,0x01,0x02,0x03
    };
    SndMessage(UNIT_BTS1,RLC_UM_DATA_REQ,D_CCCH,0,SndData,sizeof(SndData));
}
```

Specify a SFN to be notified and a skipped SFN cycle

Wait for the notification of the SFN(=100) with CPHY_SFN_IND

Send this message at the timing later than SFN(=100)

3.7.18.5 Image of PC trace

An example image of PC trace on this function is as follows.

(00) is shown in the channel field.

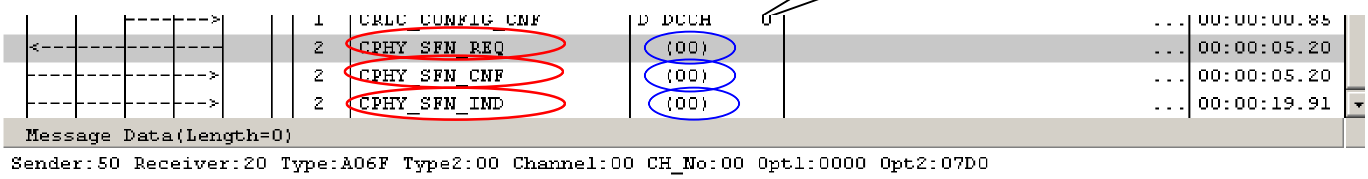


Figure 3-45) Image of PC trace on this function

3.7.18.6 Restriction:

If a second CphySfnInd( ) is called before CPHY_SFN_IND corresponding to the first CphySfnInd( ) is sent then CPHY_SFN_IND corresponding to first CphySfnInd( ) will not be sent.(The Processing of the first CphySfnInd( ) is overwritten by the second processing.)

3.7.19 Real time display on TRACE window

3.7.19.1 Outline

This function changes method to display time stamp on MX848000A TRACE window. Current function is “ progress time display” , and “ real time display” has been added. They can be changed “ Real time display” uses time of PC, which is used as control PC.

These words are used in following sections.

Word	Meaning
New Control PC	New MX848000A which applies to real time display.
New Log file	Log file, which is saved with New Control PC.
Old Control PC	Old MX848000A, which doesn’ t apply to real time display.
Old Log file	Log file, which is saved with Old Control PC.
progress time display	TRACE display method which start time is always 0.
real time display	TRACE display method, which start time is clock time of control PC.

3.7.19.2 Way to change display

<Way to change>

-There is a button to change on left side of “ Auto Log” button of TRACE window. Clicking it, and display method changes.

-Display method changes every time the button is clicked.

-The button shows current display method.

-When executing MX848000A, it displays in same way as when exiting before.

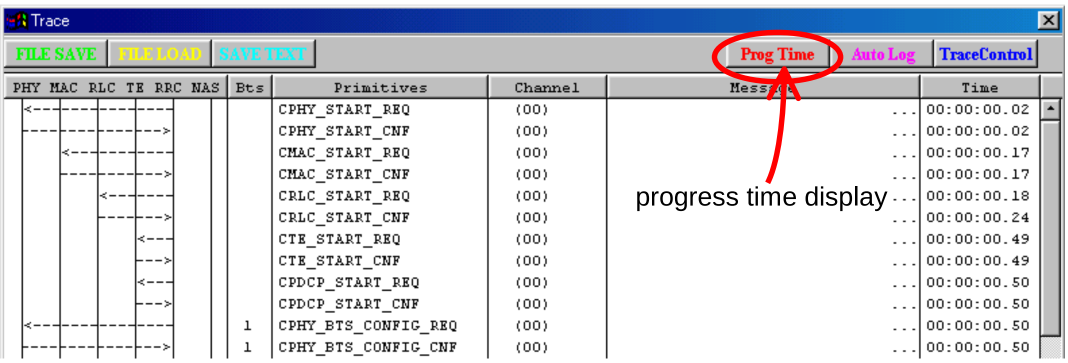


Figure 3-46) Image of progress time display

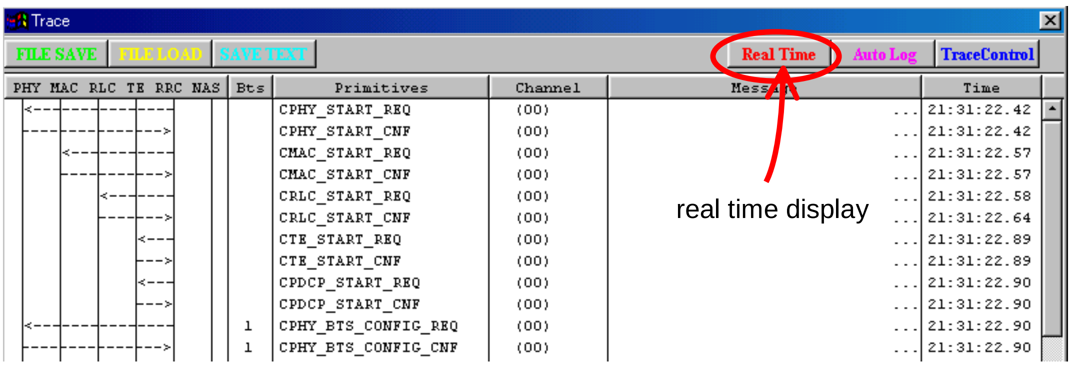


Figure 3-47) Image of real time display

3.7.19.3 FILE SAVE

-When New Control PC loads New Log file, it can change display method.

3.7.19.4 FILE LOAD

-Real time display can be used, when New Control PC load New Log file.

-New Control PC can load Old Log file. In this case, display method is progress time display.

-Old Control PC can load New Log file. In this case, display method is progress time display.

3.7.19.5 SAVE TEXT and Auto Log

-New Control PC saves text log file using a method, which is selected when clicking SATE TEXT button.

-New Control PC saves Auto log file using a method, which is selected when clicking START button.

3.7.19.6 Attentions

-The MX848000A software version that is Ver. 5.21c or later is necessary for the function.

3.7.20 PipeLine Delay Reduction Function

3.7.20.1 Outline

PipeLine Delay (PLD) means the process time that takes from starting transmission of data at RF of UE to receiving the response at UE from MD8480A/B.

This function is to reduce a process time in MD8480A/B. This function consists of three functions, PLD1, PLD2, and PLD3. PLD1 shortens 40[ms] at Uplink process time. PLD2 shortens 50[ms] at Downlink DPCH process time. PLD3 shortens 20[ms] at S-CCPCH process time. To use the function effectively, combine PLD1 and PLD2 or PLD1 and PLD3. Which combination should be used depends on the purpose. Please refer to Table 3-17 for details.

Table 3.17 Combination of usable PLD reduction function

Target	Target TrCH	Combination	Reduction time	Detail
Reduction for data channel	DCH	PLD1 PLD2	80-120[ms]	Sec.3.7.20.2
Reduction for control channel	PCH, FACH, RACH	PLD1 PLD3	60[ms]	Sec.3.7.20.3

3.7.20.1.1 Test Example

The example of using this function is shown in the following.

When RLC Poll Timer (Timer_poll) of RLC_AM_SRB (Signalling Radio Bearer) is set to 200[ms], because the time from the transmission of UE of Uplink data to the reception of Downlink data is more than 200[ms], retransmission occurs. When this function is applied to RLC AM bearer, the retransmission doesn’ t occur even if Poll Timer is set to 200[ms].

Figure 3-48 shows the example with retransmission. And, figure 3-49 shows the example without retransmission.

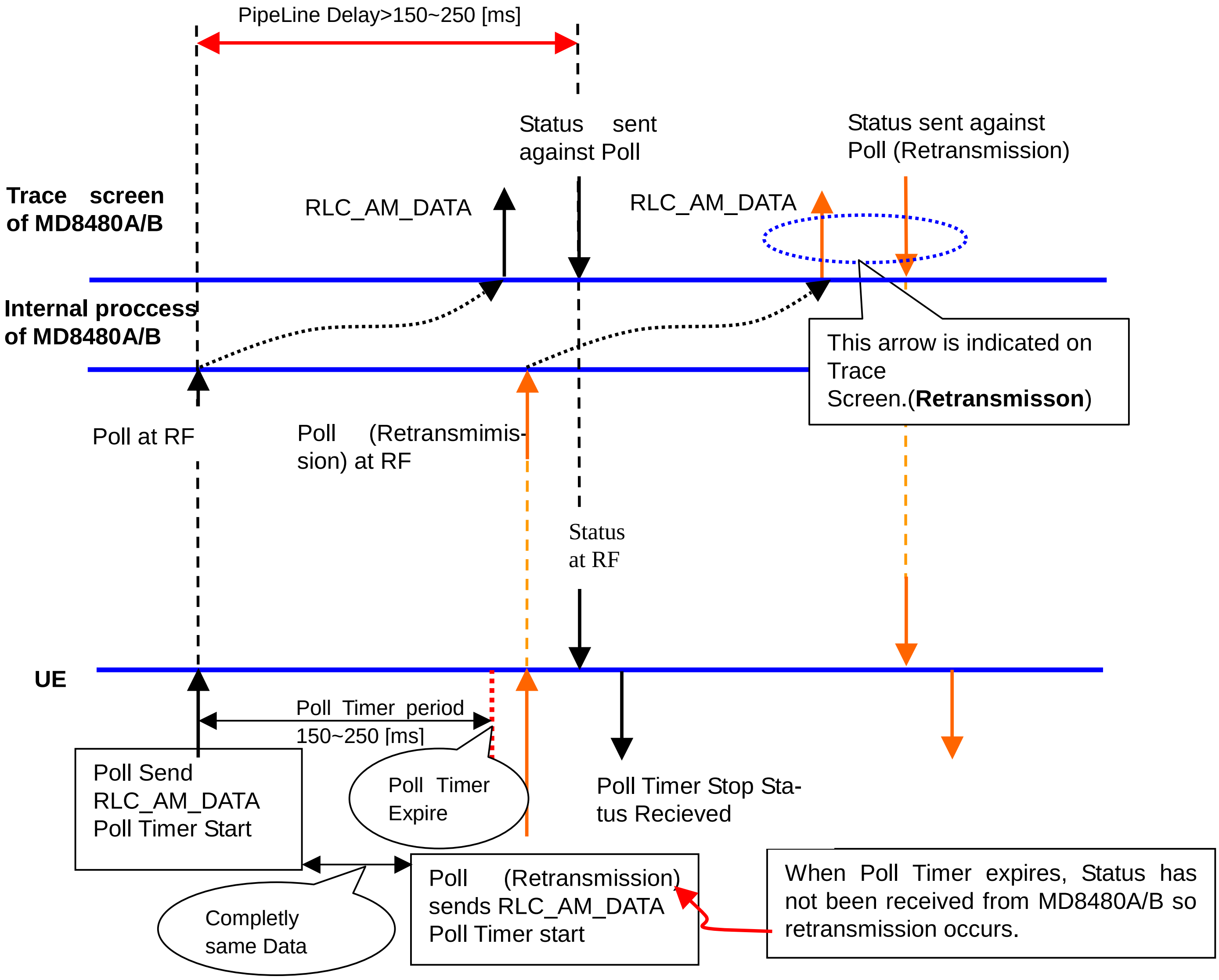


Figure 3-48) Example of Retransmission with setting Poll Timer

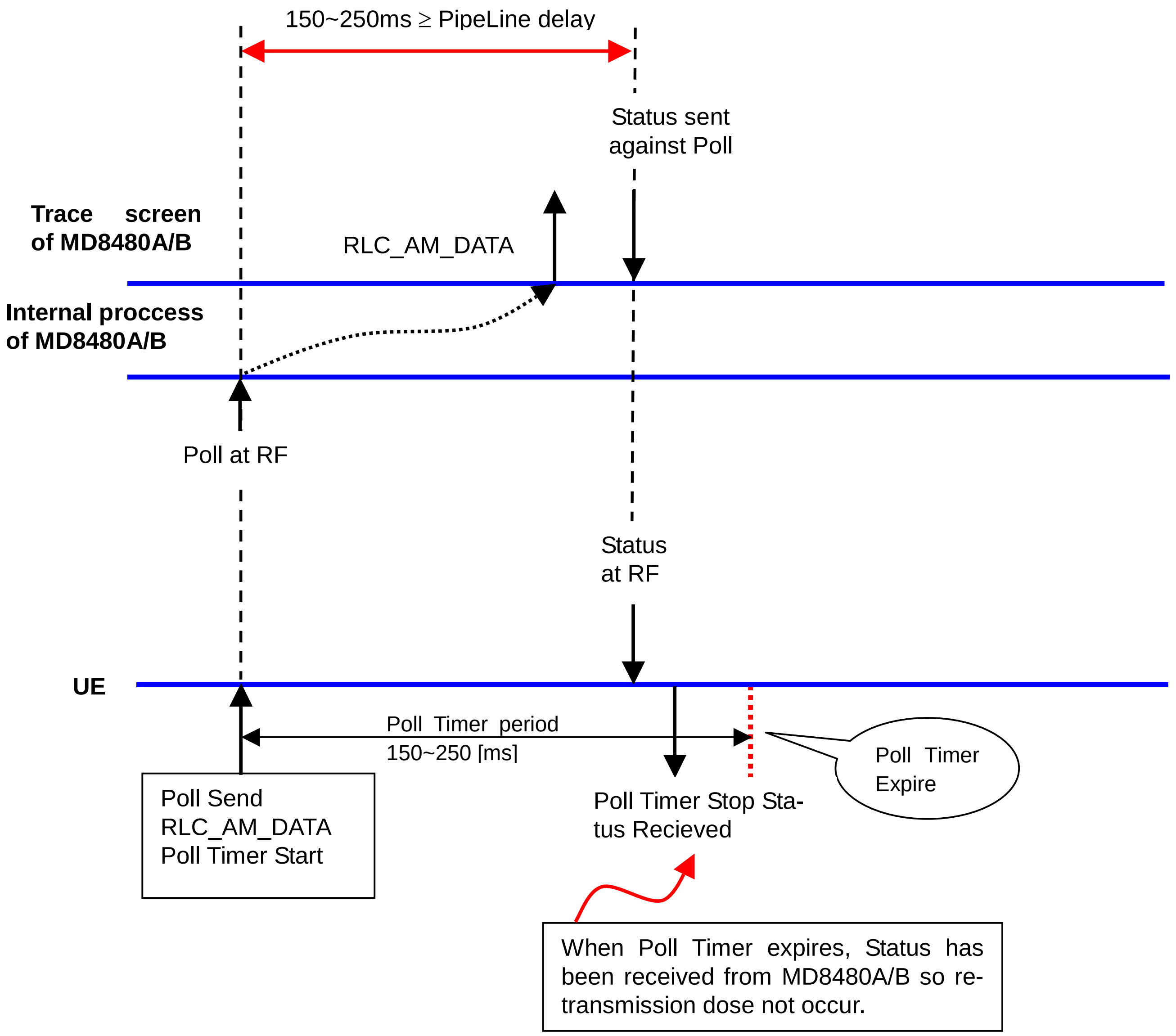


Figure 3-49) Retransmission does not occur with setting PLD function

3.7.20.2 Reduction for data channel

3.7.20.2.1 Function

The function is to reduce the processing time (from UE sending the data to receiving the STATUS of the data) by 80-120[ms]. The target TrCH for both Uplink and Downlink is DCH .

3.7.20.2.2 How to set

To reduce for data channel, set PLD1 and PLD2 on scenario as follows.

<Method>

1. Set PLD1 and 0x0100 for Uplink and PLD2 and 0x0100 for Downlink to TrchInfo[0].Static.RedFunc, which is member of CPHY_TRCH_CONFIG_PAR

(Example of describing scenario)

```
CphyTrchConfigU_DPCH_User0.TrchInfo[0].Static.RedFunc = (0x0100 | PLD1);
CphyTrchConfigD_DPCH_User0.TrchInfo[0].Static.RedFunc = (0x0100 | PLD2);
```

2. Set PLD1 and 0x0100 for Uplink and PLD2 and 0x0100 for Downlink to TrchInfo[0].Static.RedFunc, which is member of CPHY_TRCH_CONFIG_PAR. And set 50[ms] for Downlink Savetime, which is member of CMAC_CONFIG_PAR

(Example of describing scenario)

```
CmacConfig U_DPCH_User0.TrchInfo[0].Static.RedFunc     = (0x0100 | PLD1);
CmacConfig D_DPCH_User0.TrchInfo[0].Static.RedFunc     = (0x0100 | PLD2);
CmacConfig D_DPCH_User0.SaveTime                         =   50;
```

3 Execute 1, 2 as follows for U-DPCH and D-DPCH.

(Example of describing scenario)

```
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH,& CphyTrchConfigU_DPCH_User0);
CphyRlSetup      (UNIT_BTS1,U_DPCH,0,&CphyRlSetupU_DPCH ,CFN,NO_TIMEOUT);
CphyTrchConfig   (UNIT_BTS1,U_DPCH,0,& CphyTrchConfigU_DPCH_User0, CFN,NO_TIMEOUT);
CmacConfig        (UNIT_BTS1,U_DPCH,0,& CmacConfigU_DPCH_User0, CFN,NO_TIMEOUT);

CalcRMParameter(D_DPCH,&CphyRlSetupD_DPCH,& CphyTrchConfigD_DPCH_User0);
CphyRlSetup      (UNIT_BTS1,D_DPCH,0,&CphyRlSetupD_DPCH ,CFN,NO_TIMEOUT);
CphyTrchConfig   (UNIT_BTS1,D_DPCH,0,& CphyTrchConfigD_DPCH_User0, CFN,NO_TIMEOUT);
CmacConfig        (UNIT_BTS1,D_DPCH,0,& CmacConfigD_DPCH_User0, CFN,NO_TIMEOUT);
```

3.7.20.2.3 How to stop

To stop this function, execute CphyTrchConfig() and CmacConfig(). Then, the setting of RedFunc and SaveTime need not be described.

Note) After CphyTrchConfig() and CmacConfig() are executed, the Default value(=“ 0”) is overwritten in RedFunc and SaveTime.

3.7.20.2.4 Restriction

Following items aren’ t supported or corresponded.

- Channel more than Packet16 [kbps]
- In case of using certain function in MX848041A for TM bearer
- A channel of TTI=80[ms]
- HLS (Higher Layer Scheduling)
- In case each function is executed with ActivateNow, the performance before or after setting
- In case of UM (Unacknowlege Mode), switching PLD ON and OFF.
- When DCCH of TTI=10[ms] is specified as a priority channel, combination with other bearer
- AMR specified as a priority channel

In case of following combinations, it cannot be performed even if the specified priority channel is less than Packet16 [kbps].

Table 3.18 Not supported Bearer Combination for UpLink

UpLink
6B(UpLink)
5B+P64k(UpLink)
AMR+P384k(UpLink)

Note: There are some possibilities that the specified priority channel does not work even outside of the above combinations in case of combining high rate bearers. In case of specifying SRB as a priority channel, the following table shows bearer combinations that the operation has been confirmed in.

Table 3.19 Supported Bearer Combination for UpLink

UpLink
SDCCH 3.4K(TTI=40ms)
SDCCH 13.6K(TTI=10ms)
AV64K(TTI=20ms) CS RAB+ SRB 3.4k(TTI=40ms)
AV32K(TTI=20ms) CS RAB+ SRB 3.4k(TTI=40ms)
AV32K(TTI=40ms) CS RAB+ SRB 3.4k(TTI=40ms)
Packet8K(TTI=40ms)+SRB 3.4k(TTI=40ms)
Packet16K(TTI=40ms) + SRB 3.4k(TTI=40ms)
Packet32K(TTI=10ms) + SRB 3.4k(TTI=40ms)
Packet32K(TTI=20ms) + SRB3.4k(TTI=40ms)
Packet32K(TTI=40ms) + SRB 3.4k(TTI=40ms)
Packet64K(TTI=10ms) + SRB 3.4k(TTI=40ms)
Packet64K(TTI=20ms) + SRB3.4k(TTI=40ms)
Packet144kRAB (TTI=20ms) + SRB 3.4k(TTI=40ms)
Packet384K(TTI=10ms) Multicode+SRB3.4k(TTI=40ms)
Packet384K(TTI=20ms) Multicode+SRB3.4k(TTI=40ms)
AMR12.2k+SRB 3.4k(TTI=40ms) +SRB3.4k(TTI=40ms)
AMR12.2k+Packet0K(TTI=20ms)+SRB 3.4k(TTI=40ms)
AMR12.2K+Packet32K(TTI=10m) +SRB3.4k(TTI=40ms)
AMR12.2K+Packet32K(TTI=20ms)+ SRB 3.4k(TTI=40ms)
AMR12.2K+Packet64K(TTI=10m) +SRB3.4k(TTI=40ms)
AMR12.2K+Packet64K(TTI=20ms) + SRB 3.4k(TTI=40ms)
AMR12.2K+UDI(1B)(TTI=40ms) CS RAB+ SRB 3.4k(TTI=40ms)
UDI(1B)(TTI=40ms) +Packet8K(TTI=40m) + SRB 3.4k
UDI(1B)(TTI=40ms)+Packet64K(TTI=20ms) + SRB 3.4k
4B UL/DL(TC,TTI=40ms)+Packet128K(TTI=20m) +SRB3.4k(TTI=40ms)
RMC_DCCH(TTI=40ms)
RMC12.2K(TTI=20ms) + SRB(TTI=40ms)
RMC64K(TTI=20ms) + SRB(TTI=40ms)
RMC144K(TTI=20ms) + SRB(TTI=40ms)
RMC_BTFD(TTI=20ms) + SRB(TTI=40ms)
RMC384K(TTI=10ms)+SRB(TTI=40ms)

Table 3.20 Supported Bearer Combination for DownLink

DownLink
SDCCH 3.4K(TTI=40ms)
SDCCH 13.6K(TTI=10ms)
AV64K(TTI=20ms) CS RAB+ SRB 3.4k(TTI=40ms)
AV32K(TTI=20ms) CS RAB+ SRB 3.4k(TTI=40ms)
AV32K(TTI=40ms) CS RAB+ SRB 3.4k(TTI=40ms)
Packet8K(TC,TTI=40ms) + SRB 3.4k(TTI=40ms)
Packet16k (TC,TTI=40ms) + SRB 3.4k(TTI=40ms)
Packet32K(TTI=10m) + SRB3.4k(TTI=40ms)
Packet32K(TTI=20ms)+SRB 3.4k(TTI=40ms)
Packet32K(TTI=40ms) + SRB 3.4k(TTI=40ms)
Packet64K(TTI=10m) + SRB3.4k(TTI=40ms)
Packet128K(TTI=20ms)+SRB 3.4k(TTI=40ms)
Packet144k(TC,TTI=20ms) + SRB 3.4k(TTI=40ms)
Packet256K(TC,TTI=20ms) + SRB 3.4k(TTI=40ms)
Packet384K(TTI=10m) Multicode+SRB3.4k(TTI=40)
Packet384K(TTI=20m) Multicode+SRB3.4k(TTI=40)
AMR12.2K +SRB 3.4k(TTI=40ms)
AMR12.2K +Packet0K(TTI=20ms)+SRB 3.4k(TTI=40ms)
AMR12.2K +Packet 8K(TC,TTI=40ms)+ SRB 3.4k(TTI=40ms)
AMR12.2K +Packet32K(TTI=10ms) +SRB3k4
AMR12.2K +Packet64K(TTI=10ms) +SRB3k4
AMR12.2K +Packet128k(TTI=20ms)+SRB3k4
AMR12.2K +Packet256K(TC,TTI=20ms) + SRB 3.4k(TTI=40ms)
AMR12.2K +Packet384K(TTI=10ms) + SRB 3.4k(TTI=40ms)
AMR12.2K+UDI(1B)(TC,TTI=40ms) CS RAB+ SRB 3.4k(TTI=40ms)
UDI(1B)(TC,TTI=40ms)CS RAB+Packet64K(TC,TTI=20ms) + SRB 3.4k
UDI(1B)64K(TTI=40ms) +Packet8K(TTI=40m) + SRB 3.4k(TTI=40ms)
4B UL/DL(TC,TTI=40ms)+Packet128K(TTI=20m) +SRB3k4
RMC_DCCH(TTI=40ms)
RMC12.2K(TTI=20ms) + SRB(TTI=40ms)
RMC64K(TTI=20ms) + SRB(TTI=40ms)
RMC144K(TTI=20ms) + SRB(TTI=40ms)
RMC_BTFD(TTI=20ms) + SRB(TTI=40ms)
RMC384K(TTI=10ms)+SRB(TTI=40ms)

3.7.20.3 Reduction for control channel

3.7.20.3.1 Function

The function is to reduce the processing time (from UE sending the data to receiving the STATUS of the data) by 60[ms]. The target TrCH for Uplink is RACH, for Downlink are PCH and FACH.

3.7.20.3.2 How to set

To reduce for control channel, set PLD1 and PLD3 on scenario as follow. The setting in case of reducing S_CCPCH mapped PCH and FACH, is described as follow.

<Method>

1. Set PLD1 and 0x0100 for Uplink to TrchInfo[0].Static.RedFunc, which is member of CPHY_TRCH_CONFIG_PAR. And set PLD3 and 0x0100 for Downlink to TrchInfo[n].Static.RedFunc, which is member of CPHY_TRCH_CONFIG_PAR. (n=0,1,2)

(Example of describing scenario)

```
CphyTrchConfigU_PRACH.TrchInfo[0].Static.RedFunc = (0x0100 | PLD1);
CphyTrchConfigD_S_CCPCH_PCHxFACH.TrchInfo[0].Static.RedFunc = (0x0100 | PLD3);
CphyTrchConfigD_S_CCPCH_PCHxFACH.TrchInfo[1].Static.RedFunc = (0x0100 | PLD3);
CphyTrchConfigD_S_CCPCH_PCHxFACH.TrchInfo[2].Static.RedFunc = (0x0100 | PLD3);
```

2. Set PLD1 and 0x0100 for Uplink to TrchInfo[0].Static.RedFunc, which is member of CPHY_TRCH_CONFIG_PAR. And set PLD3 and 0x0100 for Downlink to TrchInfo[n].Static.RedFunc, which is member of CPHY_TRCH_CONFIG_PAR. And set 20[ms] for Downlink Savetime, which is member of CMAC_CONFIG_PAR. (n=0,1,2)

(Example of describing scenario)

```
CmacConfigU_PRACH.TrchInfo[0].Static.RedFunc         = (0x0100 | PLD1);
CmacConfigD_S_CCPCH_PCHxFACH.TrchInfo[0].Static.RedFunc    = (0x0100 | PLD3);
CmacConfigD_S_CCPCH_PCHxFACH.TrchInfo[1].Static.RedFunc    = (0x0100 | PLD3);
CmacConfigD_S_CCPCH_PCHxFACH.TrchInfo[2].Static.RedFunc    = (0x0100 | PLD3);
CmacConfigD_S_CCPCH_PCHxFACH.SaveTime                      =   20;
```

3　Execute 1, 2 as follows for U-PRACH and S_CCPCH.

(Example of describing scenario)

```
CalcRMParameter(U_PRACH,&CphyRlSetupU_PRACH,& CphyTrchConfigU_PRACH);
CphyRlSetup      (UNIT_BTS1,U_PRACH,0,&CphyRlSetupU_PRACH ,CFN,NO_TIMEOUT);
CphyTrchConfig   (UNIT_BTS1,U_PRACH,0,& CphyTrchConfigU_PRACH, CFN,NO_TIMEOUT);
CmacConfig       (UNIT_BTS1,U_PRACH,0,& CmacConfigU_PRACH, CFN,NO_TIMEOUT);

CalcRMParameter(S_CCPCH,&CphyRlSetupS_CCPCH,& CphyTrchConfigS_CCPCH_ PCHxFACH);
CphyRlSetup      (UNIT_BTS1,S_CCPCH,0,&CphyRlSetupS_CCPCH ,CFN,NO_TIMEOUT);
CphyTrchConfig   (UNIT_BTS1,S_CCPCH,0,& CphyTrchConfigS_CCPCH_ PCHxFACH, CFN,NO_TIMEOUT);
CmacConfig       (UNIT_BTS1,S_CCPCH,0,& CmacConfigS_CCPCH_ PCHxFACH, CFN,NO_TIMEOUT);
```

3.7.20.2.3 How to stop

To stop this function, execute CphyTrchConfig() and CmacConfig(). Then, the setting of RedFunc and SaveTime need not be described.

Note) After CphyTrchConfig() and CmacConfig() are executed, the Default value(=“ 0”) is overwritten in RedFunc and SaveTime.

3.7.20.2.4 Restriction

· At the first time to activate each channel, set the PLD reduction function to every target channel.

· The setting the PLD reduction function to S-CCPCH or PRACH should be set to all TrCH, which are mapped on these PhCH. In case of PCH and FACH are mapped, only PCH or only FACH setting is prohibited. When functions are set only to either channel, the setting is invalid.

· In case each function is executed with ActivateNow, the performance is not ensured before or after execution.

· When SaveTime is set excluding 20, the performance is not ensured.

3.7.21 DPCCH indication function

3.7.21.1 Outline

This function is to indicate DPCCH MD8480 receives and the result of synchronous judgment of each slot on Trace Screen.

3.7.21.2 Usage

Set MEAS_REPORT_ON(1) to DispFlag, which is the member of CPHY_RL_SETUP_PAR structure. And execute CphyRlSetup() for Channel : " U_DPCH" and ChNo. : 0.

(Example of describing scenario)

```
INT    CFN = 100;
CPHY_RL_SETUP_PAR       CphyRlSetupU_DPCH;
CphyRlSetupU_DPCH.DispFlag = MEAS_REPORT_ON;    /*ON(1)*/
CphyRlSetup(UNIT_BTS1,U_DPCH,0,&CphyRlSetupU_DPCH, CFN,NO_TIMEOUT);
```

After enabled CphyRlSetup() setting at ActivationTime, in case of receiving DPCCH, PHY_REPORT_IND is displayed on trace screen. PHY_REPORT_IND includes 8frames of DPCCH information.

(Note) CphyRlSetupMtgp() or CphyRlModify() can also be used for DPCCH indication function same as CphyRlsetup().

3.7.21.3 How to analyze Trace screen

When a DPCCH indication function became effective, Trace screen example is shown for Fig. 3-50.

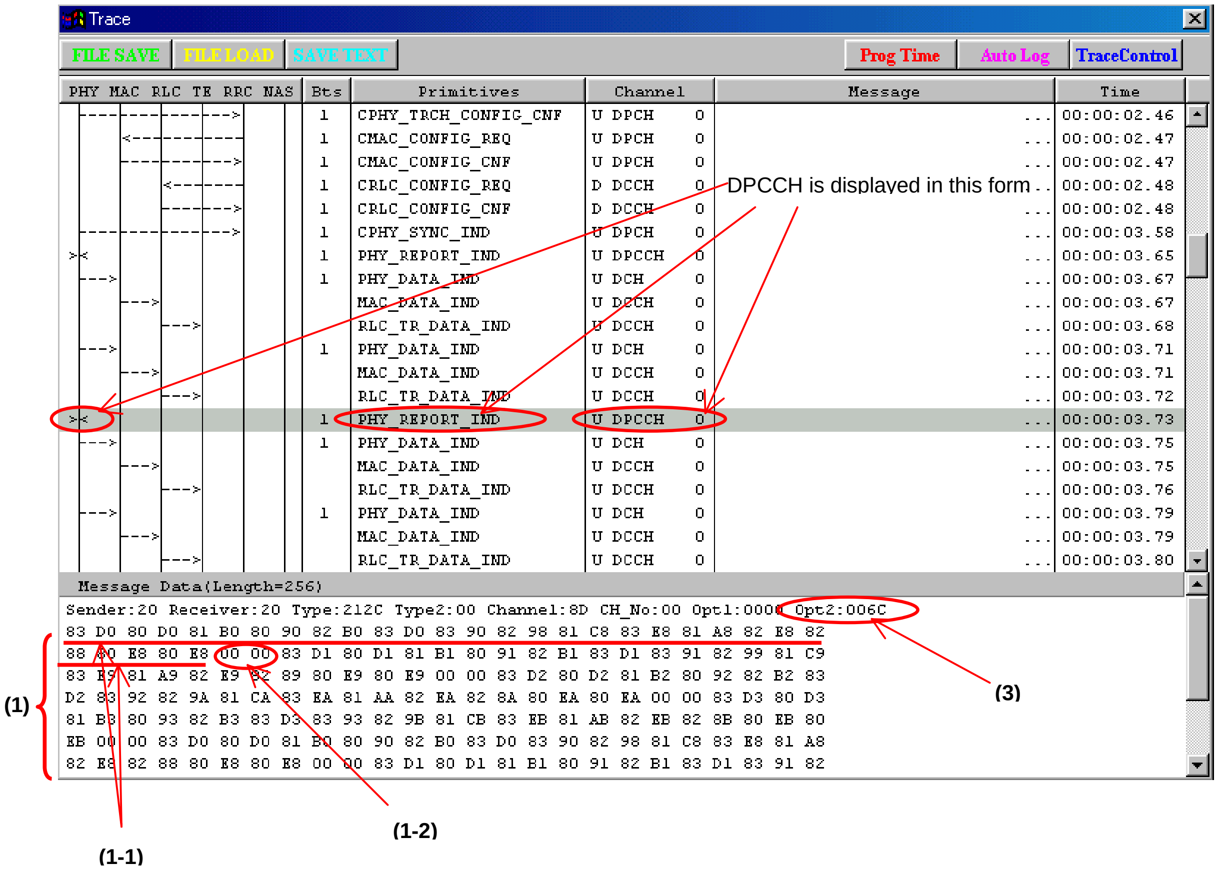


Fig.3-50 Trace screen example

(1) 8 frames of DPCCH information is indicated as the MessageData of primitive “ PHY_REPORT_IND” on trace screen.

(1-1) Indicate DPCCH information for 1 frame(15 slots).

Indicate DPCCH information for 1 slot by 2 bytes format.

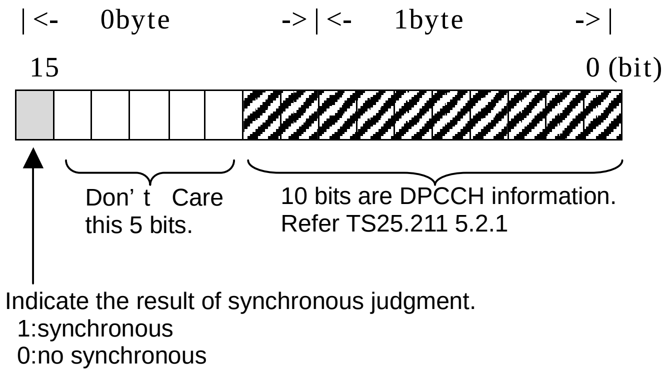


Fig 3-51 The format of DPCCH information for 1 slot.

The message of (1-1) in Fig. 3-50 indicates DPCCH information as follows

83 D0 :Slot#0 DPCCH(1 00000 11 1101 0000)
80 D0 : Slot#1 DPCCH(1 00000 00 1101 0000)
81 B0 : Slot#2 DPCCH(1 00000 01 1011 0000)
: : : : :
82 88 : Slot#12 DPCCH(1 00000 10 1000 1000)
80 E8 : Slot#13 DPCCH(1 00000 00 1110 1000)
80 E8 : Slot#14 DPCCH(1 00000 00 1110 1000)

When top bit is 1, the data is synchronous.

(When top bit is 0, DPCCH information is indicated “0x xx ..” on trace screen, so it’s easy to identify the state is no synchronous.)

(1-2) The value indicates the period for 1 frame.

“00 00” is indicated on trace screen.

(2) Indicate CFN.

The CFN of top frame in 8 frames is indicated on Opt2 in “ PHY_REPROT_IND” .

After enabled CphyRlSetup() setting, PHY_REPORT_IND is displayed on trace screen by 80[ms].

(Note)If activate PRACH after activate DPCH without DPCH/PRACH simultaneous boot function, then DPCH cannot be received, so DPCCH cannot be received. In case “ FF FF FF FF FF FF…” is indicated as DPCCH continuously, the information shows it is the state of not receiving DPCCH since the frame.
In this case, if it becomes enabled to receive DPCCH, DPCCH start to be indicated by 80[ms] again.

3.7.21.4.How to stop

There are two methods to stop this function.
(1) Use CphyRlSetup() to stop.
Set MEAS_REPORT_OFF(0) to DispFlag, which is the member of CPHY_RL_SETUP_PAR structure, execute CphyRlSetup() for Channel: "U_DPCH" ,ChNo.:0.
(Note) Using CphyRlSetupMtgp() or CphyRlModify() can also stop DPCCH indication function same as CphyRlSetup().

(2) Use CphyRlRelease() to stop.
Execute CphyRlRelease() for Channel: "U_DPCH" ,ChNo.:0.

3.7.21.5.Attentions and Restrictions

(1) DPCCH information is only indicated on trace screen.
You cannot get DPCCH information with scenario(RRC).
Check the data on trace screen or TraceLog.

(2) In case much information is indicated, "Discontinuous" may be indicated on trace screen. The DPCCH information in "Discontinuous" indication cannot be assured.
If you want to get DPCCH information on trace screen, reduce the information on trace screen.
(You can can change indicated channel with "TraceControl")

(3) Even if you use "TraceControl", you cannot configure to indicate DPCCH or not.

(4) When you test CompressedMode, in following condition DPCCH indication function cannot be enabled.
   a) The case that RLSetupFlag (member of CPHY_RL_SETUP_MTGP_PAR) is set to SETUP_EXCL_POWER.
   b) The case that RLSetupFlag (member of CPHY_RL_SETUP_MTGP_PAR) is set to SETUP_EXCL_POWER.

3.7.22 Preamble acquisition function

3.7.22.1 Outline

This function is to acquire Signature and Access Slot Number of Preamble MD8480 receives. Acquired Preamble information with RcvMessage() can be treated on scenario. The preamble information is indicated on trace screen.

3.7.22.2 Usage

In state of set MEAS_REPORT_ON(1) to DispFlag, which is the member of CPHY_RL_SETUP_PAR structure, execute CphyRlSetup() for Channel : " U_PRACH" and ChNo. : 0.

(Example of describing scenario)

```
CPHY_RL_SETUP_PAR       CphyRlSetupU_PRACH;
CphyRlSetupU_PRACH.DispFlag = MEAS_REPORT_ON;    /*ON*/
CphyRlSetup(UNIT_BTS1,U_PRACH,0,&CphyRlSetupU_PRACH, ACTIVATE_NOW,NO_TIMEOUT);
```

After executing CphyRlSetup(), in case of receiving Preamble, PHY_REPORT_IND is displayed on trace screen.

(Note 1) When use Preamble acquisition function, describe ACTIVATE_NOW in CphyRlSetup().

(Note 2) CphyRlModify() can also be used for Preamble acquisition function same as CphyRlSetup().

3.7.22.3 How to analyze Trace screen

When a Preamble acquisition function became effective, Trace screen example is shown for figure 3-52.

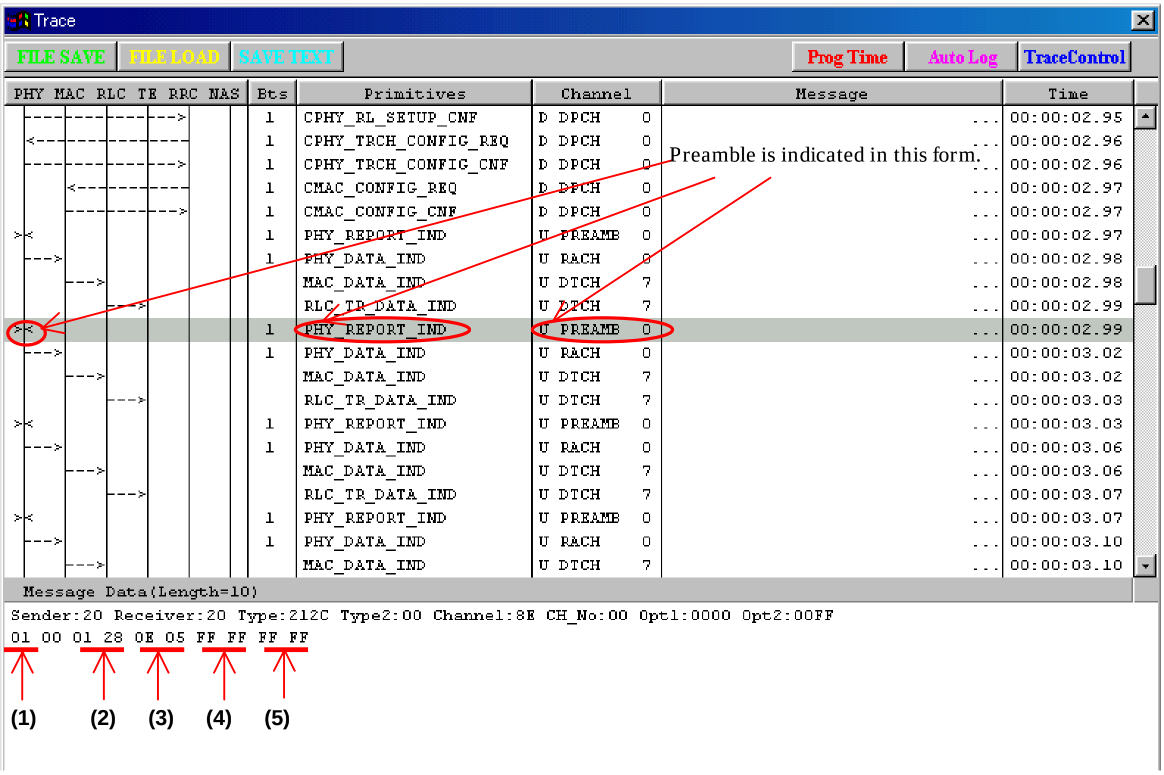


Fig.3-52 Trace screen example

Preamble information (Signature and AccessSlotNumber) of PRACH and SFN in this frame are indicated as the primitive " PHY_REPORT_IND" (U_PREAMB) on trace screen.

(1) Indicate the number of preamble acquired in 1 frame.
(The number of Preamble acquired in 1 frame is three at the maximum.)

(2) Indicate SFN
Indicate SFN(0 to 4095) by the hexadecimal number.(0 to 0x0FFF)
(Note) The SFN in the primitive is not included the SFN offset value that specified in a scenario or parameter files. If the SFN offset value for the specified BTS do not zero, add the SFN offset to the SFN value as needed.

In Fig.3-52 example, indication is the following information.

01 28 : SFN = 296

(3) Indicate Preamble information acquired first in the frame.
Indicate the information in order of AccessSlotNumber and Signature.
Indicate the following information by the hexadecimal number.

Access Slot Number	0x00 to 0x0E, 0xFF(not acquire Preamble)
Signature Number	0x00 to 0x0F, 0xFF(not acquire Preamble)

In Fig.3-52 example, indication is the following information.

0E : AccessSlot #14

05 : Signature Number #05

(4) In case of acquiring two Preambles or more, indicate Preamble information acquired second.
Indicate the information in order of AccessSlotNumber and Signature.
In case of acquiring only one or less Preamble, indicate “ 0xFF” .

(5) In case of acquiring three Preambles or more, indicate Preamble information acquired third.
Indicate the information in order of AccessSlotNumber and Signature.
In case of acquiring only two or less Preamble, indicate “ 0xFF” .

3.7.22.4 How to get the Preamble information by using RcvMessage()

Using RcvMessage() of an existing function enables to get the Preamble information, and to treat the information on scenario.

When executing RcvMessage(),and getting Preamble information, the following values return as arguments of RcvMessage().

Argument	Return value
Frame	PHY_REPORT_IND (0x212C)
Loch	PREAMBLE (0x8E)

By the following example, the Preamble information (Signature number = 1) can be indicated on Main Window in Control Software.

(Example of describing scenario)

```
INT                 BtsNo = 0;
 INT                Frame;
 INT                Phch = 0;
 INT                PhchNo = 0;
 CPHY_REPORT_PAR RlDispPar;
 INT                i, j;
 CHAR                 str[256];
 INT                SigNum;
 USHORT             Sfn;
 INT                SignatureNum;
 INT                ASnum;

 SigNum = 1;    /*Signature Number */

 for( i=0; i<1; ){

    RcvMessage( &BtsNo, &Frame, &Phch, &PhchNo, (CHAR *)&RlDispPar, NO_TIMEOUT );
    if( Phch!=U_PREAMBLE ){
       continue;
    }
    for( j=0; j<RlDispPar.GetPreamble[0].NumOfPreamble; j++ ){
       if( RlDispPar.GetPreamble[0].Preamble[j].SIGNumber!=SigNum ){
          continue;

       }
       SignatureNum = RlDispPar.GetPreamble[0].Preamble[j].SIGNumber;
       ASnum = RlDispPar.GetPreamble[0].Preamble[j].ASNumber;
       Sfn = Change_SFN(RlDispPar.GetPreamble[0].SFN);

       i++;
       sprintf(str,"SignatureNo=%d, AS=%d, SFN=%d", SignatureNum,ASnum, Sfn);
       SequenceDisp(str);
       if(i>=10){
          break;
       }
       break;
    }
```

(Note)
About definition of Change_SFN and structure CPHY_REPORT_PAR, refer to primitive.h (C:¥Mx848000¥Scenario¥include)

Esecute RcvMessage()

Distinguish whether the data is Preamble information.

Worth of the number of acquired Preamble information is acquired.

Acquire Preamble information.

Please use Change_SFN when you acquire SFN.

Preamble information is indicated on Main Window.

3.7.22.5 How to stop

There are two methods to stop this function.

(1) Use CphyRlSetup() to stop.

Set MEAS_REPORT_OFF(0) to DispFlag, which is the member of CPHY_RL_SETUP_PAR structure, execute CphyRlSetup() for Channel : "U_PRACH" ,ChNo. : 0.

(Note) Using CphyRlModify() can also stop Preamble acquisition function same as CphyRlSetup().

(2) Use CphyRlRelease() to stop.

Execute CphyRlRelease() for Channel: "U_PRACH" ,ChNo.:0.

3.7.23 Lossless InterRAT Cell Change Function

3.7.23.1 Outline

This function is to do InterRAT Cell Change between W-CDMA(SRNC) and GPRS(SGSN) without the loss of data. The basic sequence about this function is described in 3GPP TS23.060.

3.7.23.2 Sequence

The sequence to enable Lossless InterRAT Cell Change function with the Signalling Tester is shown.

3.7.23.2.1 W-CDMA to GPRS

The example of the Lossless InterRAT Cell Change sequence from W-CDMA to GPRS is shown in figure 3.53.

3.7.23.2.2 GPRS to W-CDMA

The example of the Lossless InterRAT Cell Change sequence from GPRS to W-CDMA is shown in figure 3.54.

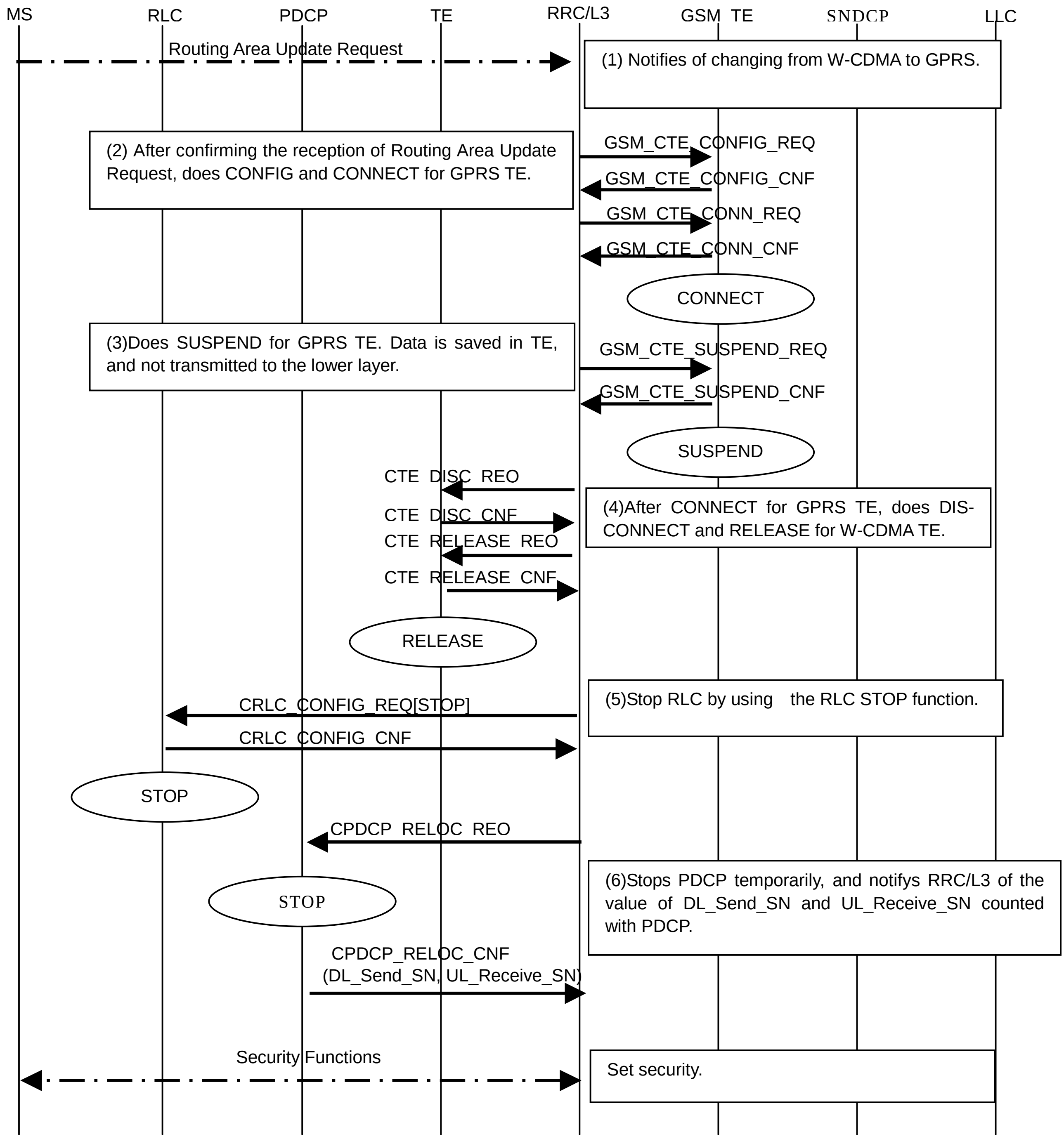


Figure 3.53 InterRAT Cell Change sequence from W-CDMA to GPRS(1/3)

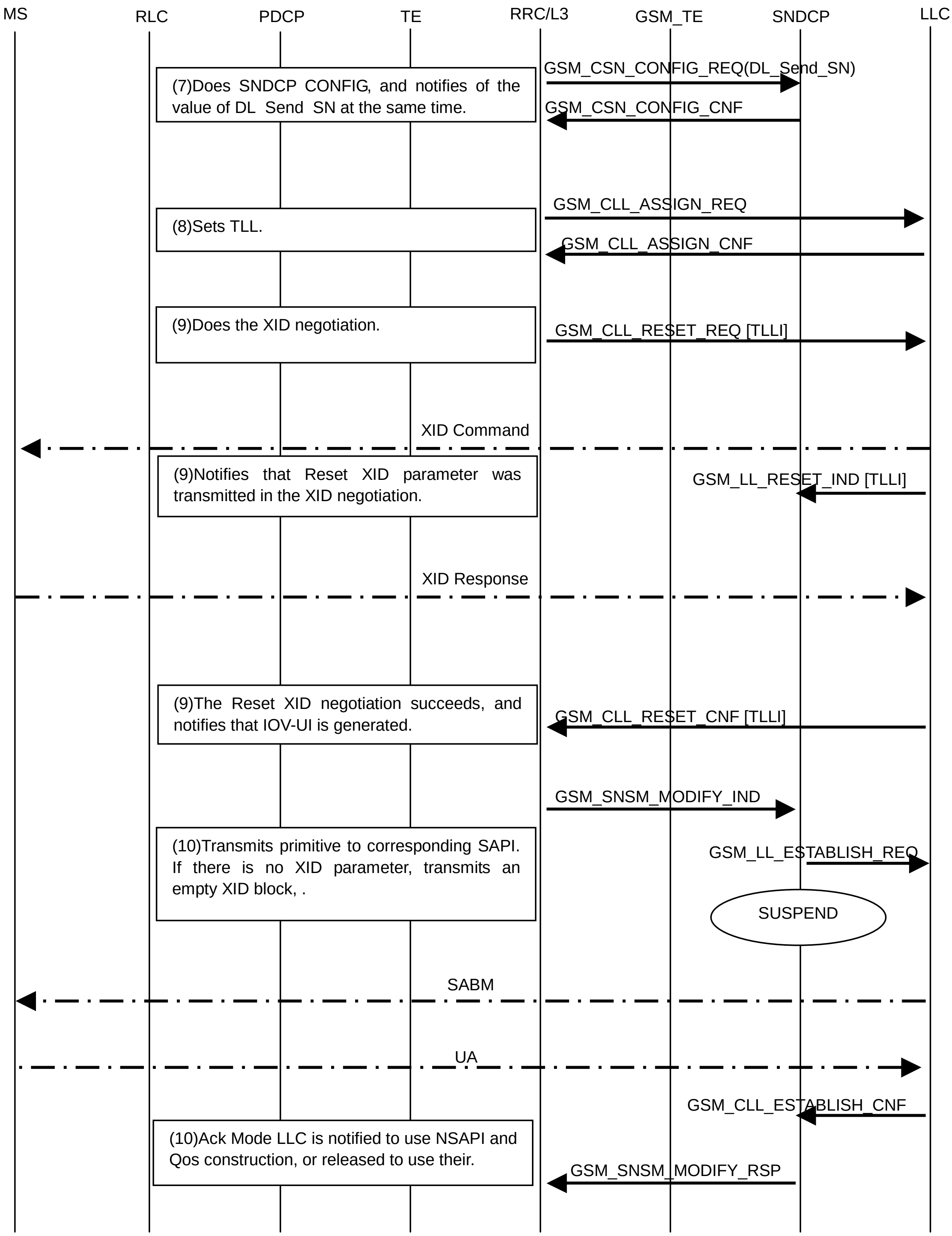


Figure 3.53 InterRAT Cell Change sequence from W-CDMA to GPRS(2/3)

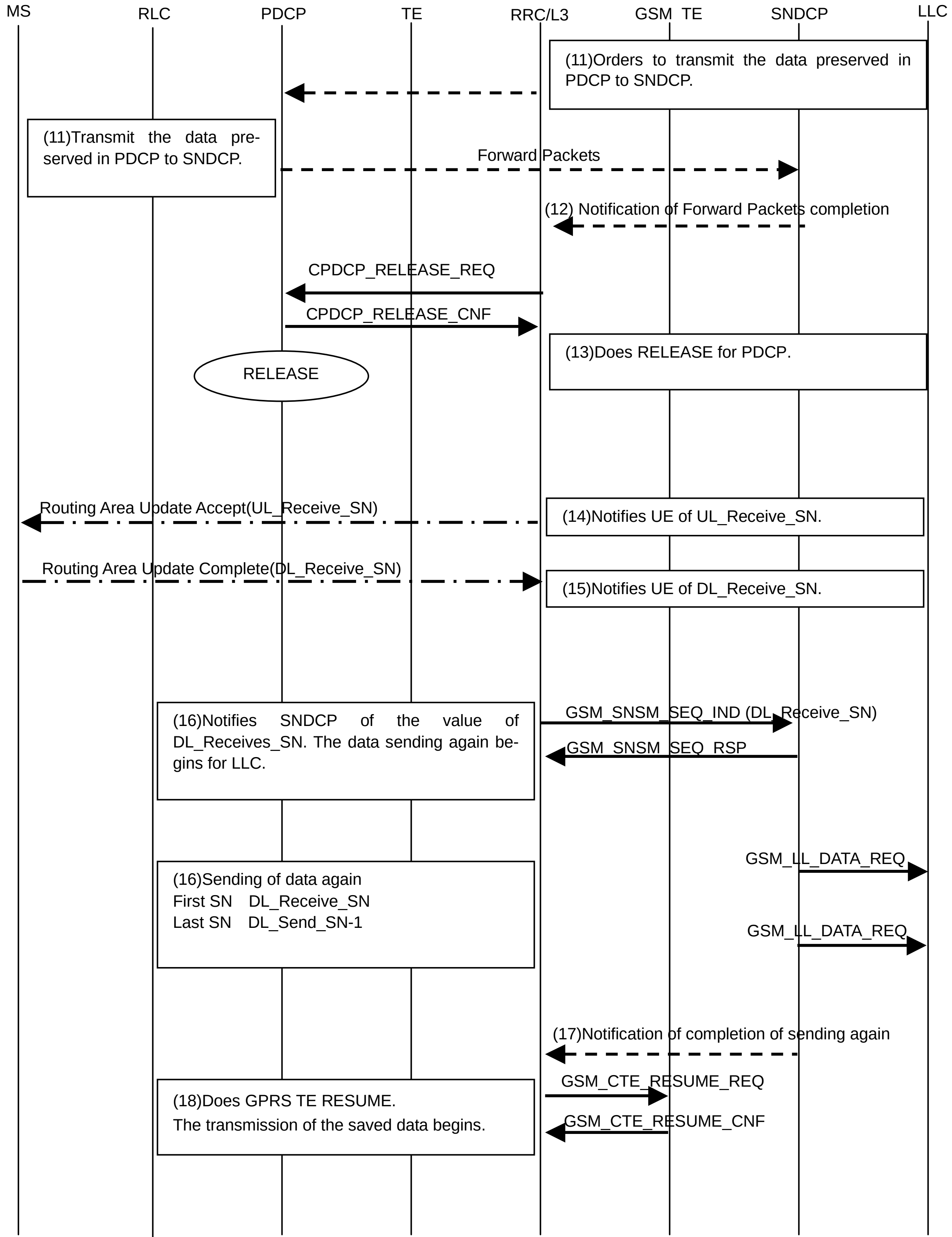


Figure 3.53 InterRAT Cell Change sequence from W-CDMA to GPRS(3/3)

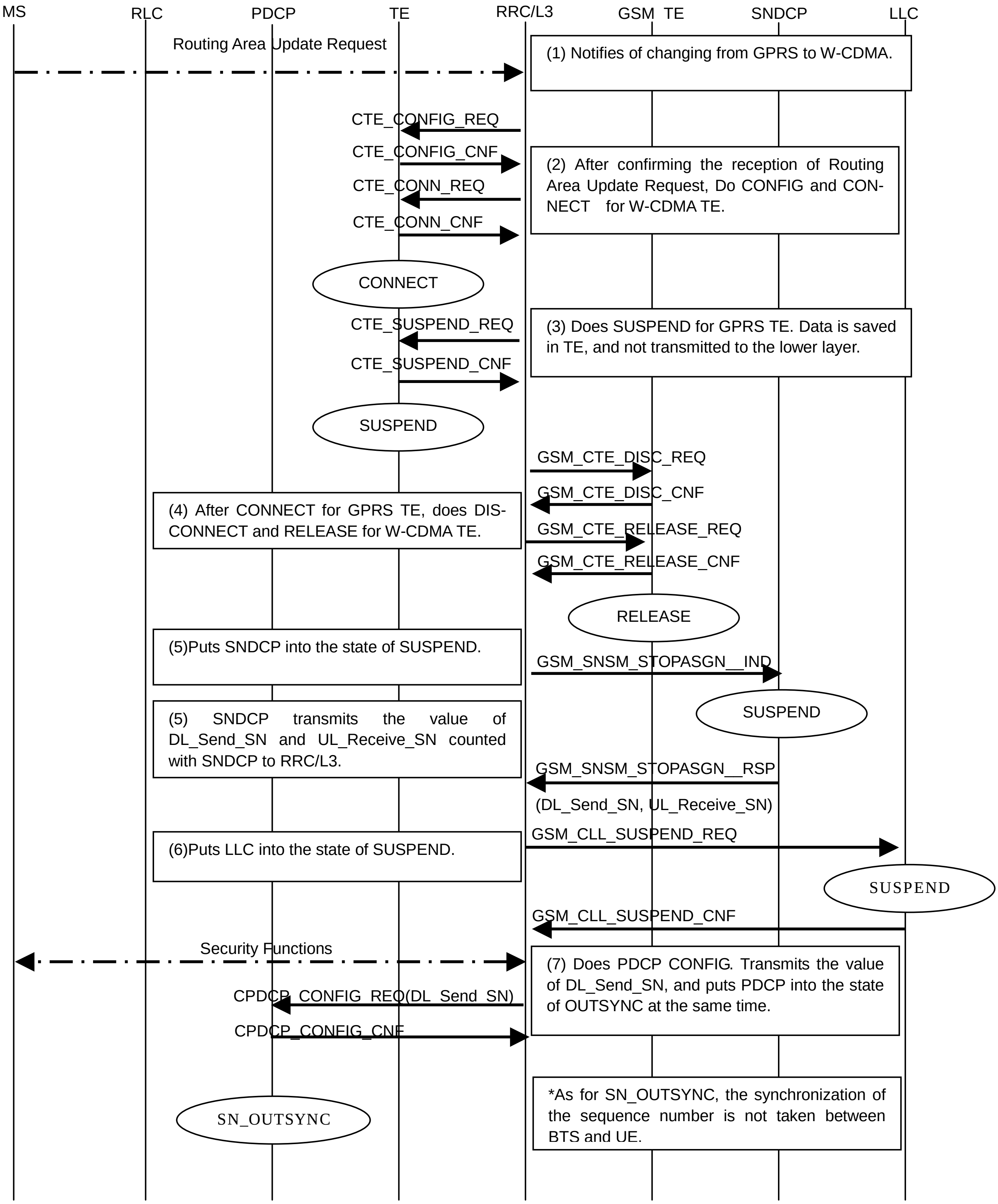


Figure 3.54 InterRAT Cell Change sequence from GPRS to W-CDMA(1/2)

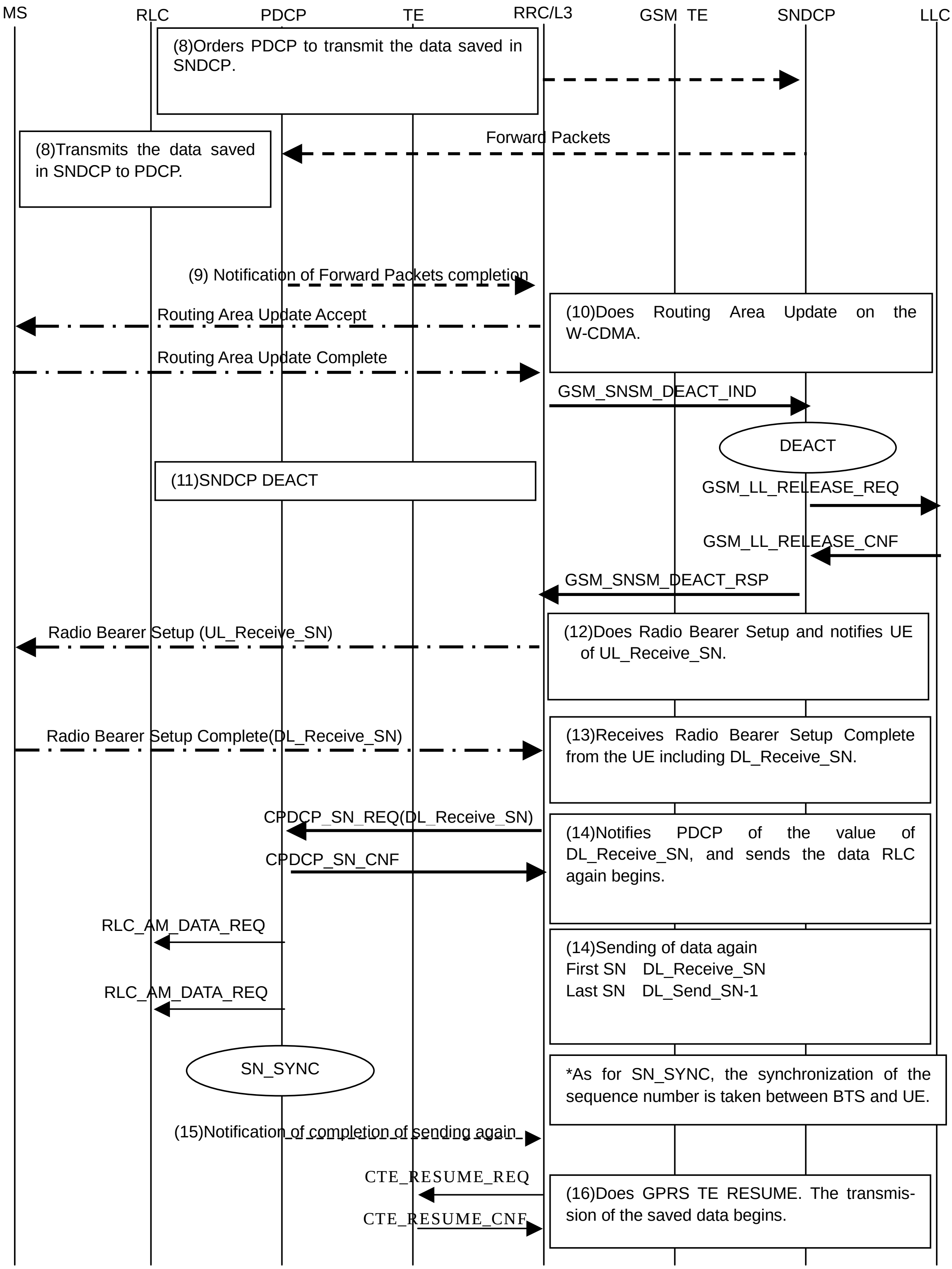


Figure 3.54 InterRAT Cell Change sequence from GPRS to W-CDMA(2/2)

3.7.23.3 Scenario

Please describe it referring to the following sample scenario when you use the Lossless InterRAT Cell Change function.

3.7.23.3.1 Sample Scenario

An attached Lossless InterRAT Cell Change scenario is shown below.

There are scenarios to "C:¥Mx848000¥Scenario¥SRC¥Lossless_InterRAT"

Table 3.21 Scenario list

Scenario Name	Description
LosslessInterRAT_ fromWtoG.c	Lossless InterRAT Cell Change is done from W-CDMA to GPRS.
LosslessInterRAT_ fromGtoW.c	Lossless InterRAT Cell Change is done from GPRS to W-CDMA.

*As for the Message content of the data, it is not the one that actually operates though RRC(L3) Message is sent and received by using the SndMessage() function and the RcvMessage() function in this scenario between the BTS and UE.

3.7.23.3.2 Scenario Configuration

LosslessInterRAT_fromWtoG.c
(The number in the inside of ( ) corresponds to figure 3-53.)

(1)Receives Routing Area Update Request from UE by using the RcvMessage() function.

(2)Does CONFIG and CONNECT for GPRS TE by using the CteConfig() function and the CteConnect() function.

(3)The CteSuspend() function stops the transmission of the data of the specified channel temporarily, and saves the received data.

(4)Does DISCONNECT and RELEASE for W-CDMA TE by using the CteDisconnect() function and the CteRelesase() function.

(5)Stops RLC by using CrlcConfig() function that sets CRLC_AM_STOP. (As for the RLC STOP function, it is refer to 3.7.16 RLC STOP,RLC Continue function.)

(6)Stops PDCP temporarily by the CpdcpReloc() function, and notifies RRC/L3 of DL_Send_SN and UL_Receive_SN counted with PDCP.

(7)Sets SNDCP by using the GsmSndcpConfig() function.

(8)Sets the TLLI parameter to LLC by using the GsmLlcAssign() function.

(9)Does the negotiation of the Reset XID parameter by using the GsmLlcReset() function.

(10)Changes the link of NSAPI and SAPI has already been constructed by using the GsmSndcpModify() function.

(11)Transmits the data preserved in PDCP to SNDCP by using the CpdcpFwd() function. The primitive when this transmitting uses primitive GSM_CSN_FWDPDU_REQ of the Signalling Tester originality definition (It is not defined in 3GPP standard).

(12) Confirms that the data transfer from W-CDMA to GPRS is completed by using the GsmSndcpRcvControl() function. This function comes off when the sequence number that has been transmitted from W-CDMA is corresponding to the sequence number of the data that came from W-CDMA to GPRS. It is possible to specify the time-out value. Uses primitive GSM_CSN_RCVCMP_IND of the Signalling Tester originality definition.

(13)Releases PDCP by using the CpdcpRelease() function.

(14)Transmits GPRS Routing Area Update Accept by using the SndMessage() function. Notifies UE of UL_Receive_SN.

(15)Receives GPRS Routing Area Update Complete by using the RcvMessage() function. Recives DL_Receive_SN from theUE.

(16)Notifies SNDCP of the sequence number in which the transmission of the data of descending is restarted by using the GsmSndcpSequence() function, and the data transmission is restarted. Substitutes DL_Receive_SN that receives for the third argument of the GsmSndcpSequence() function.

(17) Confirms that the data sending again is completed from SNDCP by using the GsmSndcpRcvControl() function. This function comes off when GSM_LL_DATA_REQ with the last N-PDU Number sent again from SNDCP is transmitted. It is possible to specify the time-out value. Uses primitive GSM_CSN_SNDCMP_IND of the Gignalling Tester originality definition.

(18) Restarts the transmission of the data of the specified channel by using the CteResume() function.

LosslessInterRAT_fromGtoW.c

(The number in the inside of ( ) corresponds to figure 3-54.)

(1)Receives Routing Area Update Request from UE by using the RcvMessage() function.

(2)Does CONFIG and CONNECT for W-CDMA TE by using the CteConfig() function and the CteConnect() function.

(3)The CteSuspend() function stops the transmission of the data of the specified channel temporarily, and preserves the received data.

(4)Does DISCONNECT and RELEASE of GPRS TE by using the CteDisconnect() function and the CteRelesase() function.

(5) Demands the descending data transmission stop and the DL Send N-PDU Number/UL Receive N-PDU Number notification by the GsmSndcpStopAssign() function.

(6) Puts LLC into the state of SUSPEND by the GsmLlcSuspend() function, and stops the transmission of the data of the direction of descending.

(7) When uses the Lossless InterRAT Cell Change function, SN_Sync is made PDCP_SN_SYNC_ENA. Moreover, DLSndNpduNo acquired in the GsmSndcpStopAssign() function is put in Start_DL_Send_SN.

(8)Transmits the data preserved in SNDCP to PDCP by the GsmSndcpFwd() function.

(9) Confirms that the data transfer from GPRS to W-CDMA is completed is confirmed by using the PdcpRcvControl() function. This function comes off when the sequence number that has been transmitted from GPRS is corresponding to the sequence number of the data that came from GPRS to W-CDMA. It is possible to specify the time-out value. Uses primitive CPDCP_RCVCMP_IND of the Signalling Tester originality definition.

(10) Sends and receives routing Area Update with W-CDMA between the BTS and UE.

(11) Notifies the sequence number in which the transmission of the descending data of SNDCP is restarted by the GsmSndcpDeact() function.

(12) Transmits radio Bearer Setup to the UE by using the SndMessage() function. Notifies UL_Receive_SN

(13) Receives Radio Bearer Setup Complete using the RcvMessage() function. Receives DL_Receive_SN from the UE.

(14) Notifies the sequence number in which the transmission of the data of descending is restarted to PDCP by using the CpdcpSeqNum() function, and restarts the data transmission. Substitutes DL_Receive_SN that receives for CpdcpSnPar.DL_Receive_SN.

(15) Confirms the thing that the data sending again is completed from PDCP by using the PdcpRcvControl() function. This function comes off when RLC_AM_DATA_REQ with the last sequence number sent again from PDCP is transmitted. It is possible to specify the time-out value. Uses primitive PDCP_SNDCMP_IND of the Cigna ring tester originality definition.

(16)Restarts the transmission of the data of the specified channels by using the CteResume() function.

3.7.24 W-CDMA CBS Function

3.7.24.1 Outline

CBS(Cell Broadcast Service) is service which enables an Information Provider to submit short messages for broadcasting to a specified area within the PLMN.

Support service contract of v5.40 or later is necessary to use W-CDMA CBS.

CBS is controlled to use the MD8480A/B inside of the BMC layer.

The function of the BMC layer supported by the MD8480A/B is as follows.

· CBS level 1 scheduling

This function can set the transmission timing of FACH that transmits the BMC message (CBS message and scheduling message).

(The detail about the CBS level1 scheduling see 3GPP TR25.925.)

· CBS level 2 scheduling

[Manual mode]

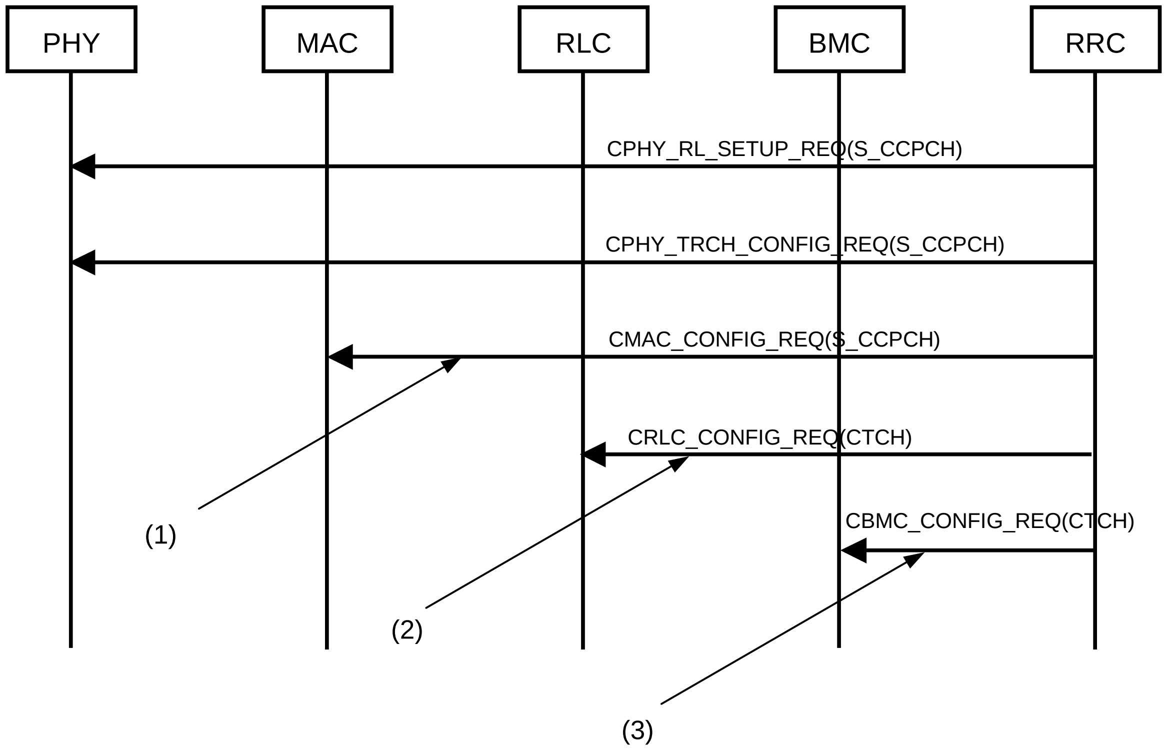
When the method of CBS level2 scheduling is set in the scenario, MD8480A/B transmits BMC message according to the setting.

· Correspondence to Pipeline Delay Reduction function (FACH)

When Pipeline Delay Reduction function (FACH) supported in v5.30 or later is used, W-CDMA CBS function can be used.

3.7.24.2　Sequence image

3.7.24.2.1　Sequence image to the setting of BMC layer



Setting of (1)

- CTCH allocation period and CBS frame offset specified with SIB5 and SIB6 are set.

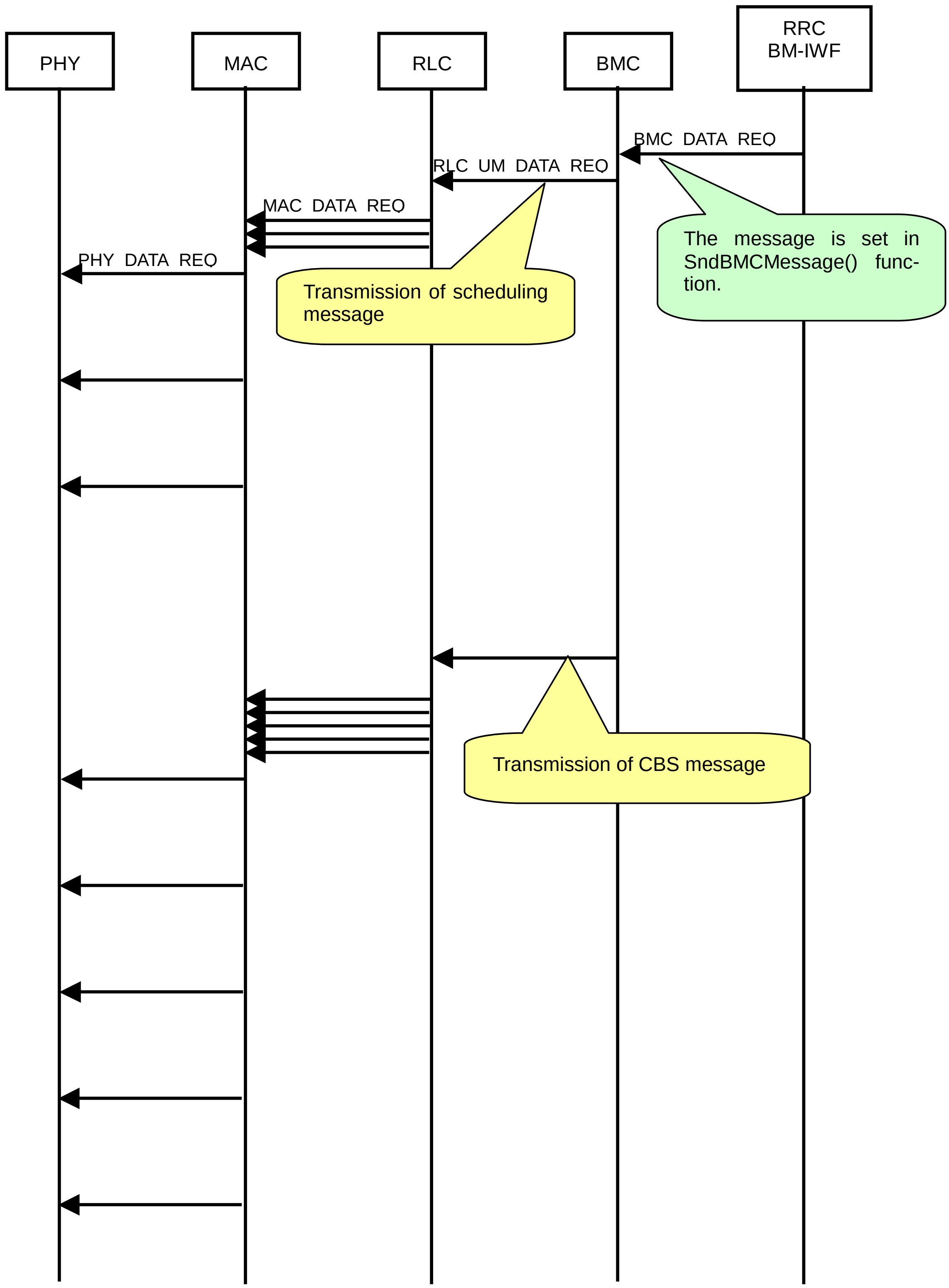
Setting of (2)

- CTCH is set.

Setting of (3)

- BMC layer is set and started.

3.7.24.2.2 Sequence image at BMC message transmission



3.7.24.3 How to set

3.7.24.3.1 Activation of BMC layer

The activation of BMC layer is executed by the CbmcConfig() function.

The BMC layer that can be activated is BTS1, BTS2 and BTS3.

[Note] BMC message can be transmitted by only one BTS which activated S_CCPCH for CBS using CmacConfig() function.

3.7.24.3.1.1 Explanation of CBMC_CONFIG_PAR structure

This structure has a parameter of " MaxTFI_FACH" .

This parameter is MD8480 original parameter.

This parameter is set to the maximum value of TFI of FACH to use CTCH.

For example, when the parameter like the table below is specified and FACH used for CTCH is TrCH#1, MaxTFI_FACH is 3.

```
CmacConfigPar->TFCS.NumOfTFC    = 9;        /* Number of TFC */
                /* TFC = 0 */
CmacConfigPar->TFCS.TFC[0][0] = 0;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[0][1] = 0;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[0][2] = 0;        /* TrCH#2 */
                /* TFC = 1 */
CmacConfigPar->TFCS.TFC[1][0] = 1;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[1][1] = 0;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[1][2] = 0;        /* TrCH#2 */
                /* TFC = 2 */
CmacConfigPar->TFCS.TFC[2][0] = 0;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[2][1] = 1;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[2][2] = 0;        /* TrCH#2 */
                /* TFC = 3 */
CmacConfigPar->TFCS.TFC[3][0] = 1;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[3][1] = 1;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[3][2] = 0;        /* TrCH#2 */
                /* TFC = 4 */
CmacConfigPar->TFCS.TFC[4][0] = 0;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[4][1] = 2;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[4][2] = 0;        /* TrCH#2 */
                /* TFC = 5 */
CmacConfigPar->TFCS.TFC[5][0] = 1;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[5][1] = 2;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[5][2] = 0;        /* TrCH#2 */
                /* TFC = 6 */
CmacConfigPar->TFCS.TFC[6][0] = 0;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[6][1] = 3;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[6][2] = 0;        /* TrCH#2 */
                /* TFC = 7 */
CmacConfigPar->TFCS.TFC[7][0] = 0;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[7][1] = 0;        /* TrCH#1 */
```

```
CmacConfigPar->TFCS.TFC[7][2] = 1;        /* TrCH#2 */
            /* TFC = 8 */
CmacConfigPar->TFCS.TFC[8][0] = 0;        /* TrCH#0 */
CmacConfigPar->TFCS.TFC[8][1] = 1;        /* TrCH#1 */
CmacConfigPar->TFCS.TFC[8][2] = 1;        /* TrCH#2 */
```

See section A.2 for explanation of CBMC_CONFIG_PAR structure other than the above-mentioned.

3.7.24.3.2 CBS level1 scheduling

CBS Level1 Scheduling is specified by both CmacConfig() function and CbmcConfig() function.

[Setting of CmacConfig() function]
Specify CbsInfo.CBS_Flag=1 in case of operating CBS Level1 Scheduling.
The condition that CBS_Flag becomes effective is as follows.

- Argument Phch is S_CCPCH.
- Setting of the logical channel CTCH exists.
- Setting of the transport channel FACH exists.
- CTCH is mapped on FACH and CCPCH.

See 3GPP TR25.925 or TS25.324 about the parameter of CbsInfo.Level1Info.
See explanation of the CMAC_CONFIG_PAR structure (in section A.2) for parameters other than the above-mentioned.

[Setting of CbmcConfig() function]
See explanation of the CBMC_CONFIG_PAR structure (in section A.2) about the setting contents of a parameter. Set Level1Info of both CMAC_CONFIG_PAR and CBMC_CONFIG_PAR to same values.

3.7.24.3.3 CBS level2 scheduling

CBS level2 scheduling is set by SndBMCMessage() function.
SndBMCMessage() function can be set the BMC message of BTS1, BTS2 and BTS3.
[Note] BMC message can be transmitted by only one BTS which activated S_CCPCH for CBS using CmacConfig() function.

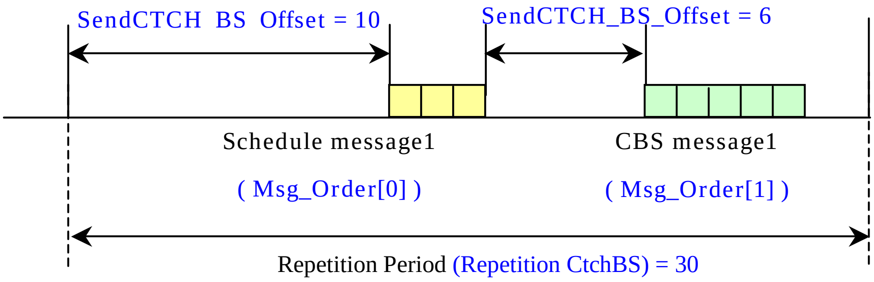
This section explains about the argument BMC_DATA_PAR structure and argument Start_period of the SndBMCMessage() function.

3.7.24.3.3.1 An example of setting for BMC_DATA_PAR structure

This structure is used for setting CBS message, Scheduling message and a transmission method of messages.

See section A.2 for explanation of BMC_DATA_PAR structure.

In case of transmitting messages as follows, make a scenario like the "scenario descriptions example".




```
/* Scenario description example */

   BMC_DATA_PAR          BMC_Message;
   UCHAR                 CBS_Message[90];             /* CBS message */
   UCHAR                 Schedule_Message[40];        /* Scheduling message */

   /* Abbreviation */

   /* Setting of CBS message */
   BMC_Message. CBS_Msg.Flag = 1;                     /* 1:change or add */
   BMC_Message. CBS_Msg.Message_number = CBSDATA_1;
   BMC_Message. CBS_Msg.Size = 90;
   BMC_Message. CBS_Msg.NumOfBroadcast = 0;           /* 0:infinity */
   memcpy( &BMC_Message. CBS_Msg.Data[0], &CBS_Message[0], 90 );

   /* Setting of Schedule message */
   BMC_Message. Schedule_Msg.Flag = 1;                /* 1:change or add */
   BMC_Message. Schedule_Msg.Message_number = CBS_SCHEDULE_1;
   BMC_Message. Schedule_Msg.Size = 40;
   memcpy( &BMC_Message. Schedule_Msg.Data[0], &Schedule_Message[0], 40 );

   /* Setting way of broadcasting BMC messages */
   BMC_Message. Level2Info.Flag = CHANGE_ORDER;
   BMC_Message. Level2Info.NumOfMessage = 2;
   BMC_Message. Level2Info.Repetition_CtchBS = 30;
   BMC_Message. Level2Info.Msg_Order[0].Message_number = CBS_SCHEDULE_1;
   BMC_Message. Level2Info.Msg_Order[0].SendCTCH_BS_Offset = 10;
   BMC_Message. Level2Info.Msg_Order[1].Message_number = CBS_DATA_1;
   BMC_Message. Level2Info.Msg_Order[1].SendCTCH_BS_Offset = 6;

   SndBMCMessage(D_CTCH, 0, &BMC_Message, 1);
```

3.7.24.3.3.2　An example of setting for Start_period

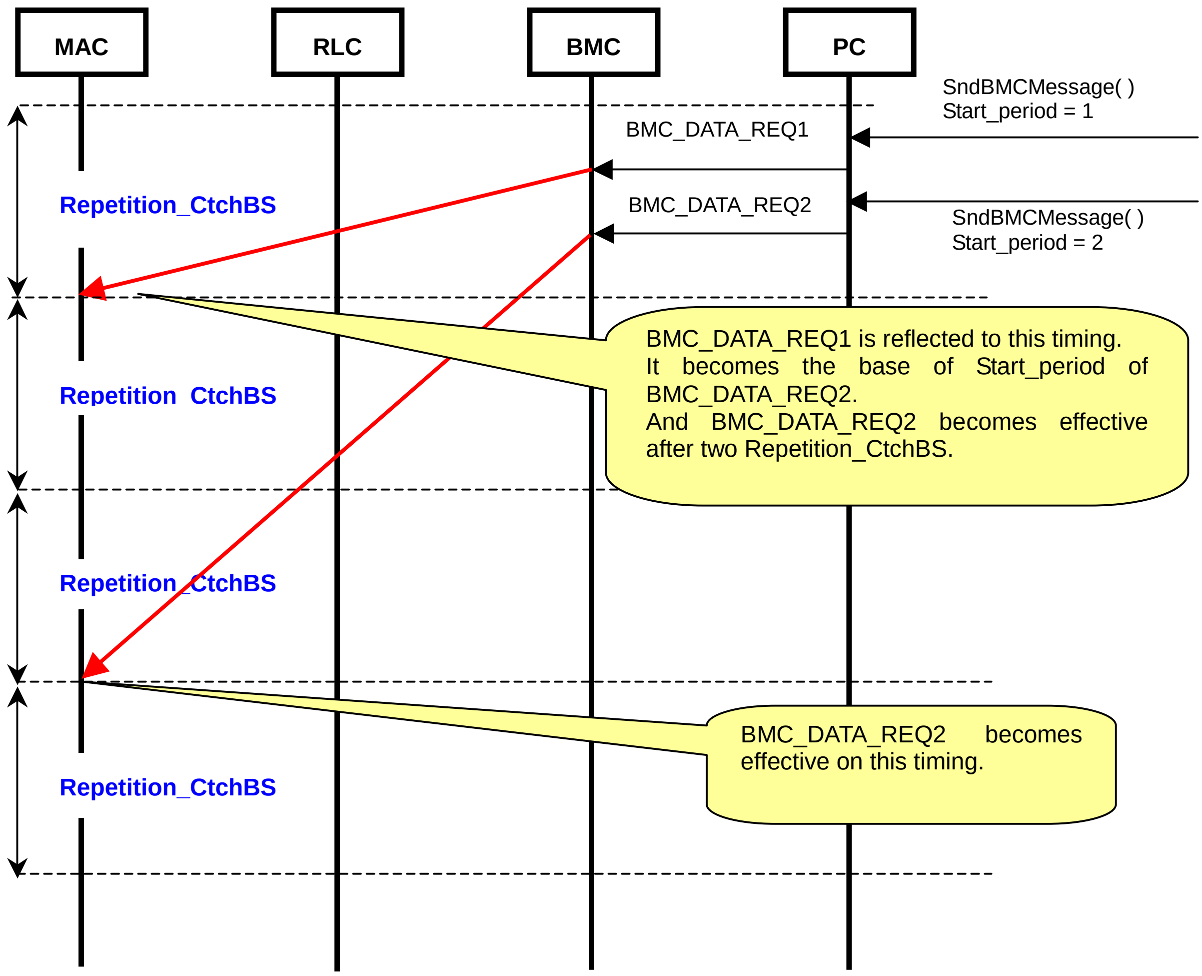
Argument Start_period of SndBMCMessage() function is a parameter that specifies timing in which this function becomes effective.

The unit is the number of times of Repetition_CtchBS.

When Start_period=2 is specified, this function becomes effective after two Repetition_CtchBS.

However, BMC layer can accumulate ten SndBMCMessage() function.

Therefore, if SndBMCMessage() function is continuously executed, Start_period sets as the basis for the beginning of the previous SndBMCMessage() function.



3.7.24.4 Sample Scenario

The attached Harikiri sample scenarios for W-CDMA CBS are as follows.

Table 3-22 Scenario view for W-CDMA CBS

Scenario name	Description
Wcdma_CBS_Manual(PCH_FACH).c	PCCH×1+(CCCH×1+DCCH×4+BCCH×1+CTCH×1) +DTCH×1
Wcdma_CBS_Manual(FACH).c	CCCH×1+CTCH×1

3.7.24.5 Restriction

The restriction in relation to using Cell Broadcast Service is described below.

- Does not support ANSI-41.
- Cannot transmit BMC message from both BTS1 and BTS2 in SHO and HHO.
- Logical channel (CTCH) to transmit BMC message can set only one.
- If S_CCPCH is used for FACH mapping CTCH but is not used for PCH, CTCH must be mapped as follows.

```
CmacConfigPar->LochInfo[0][0].Loch = D_CTCH;
```

See “ InitializeParameter_S_CCPCH_FACH_BMC()” that exists in “ Initialize-Parameter.c” .

- If S_CCPCH is used for FACH mapping CTCH and for PCH, CTCH must be mapped as follows.

```
CmacConfigPar->LochInfo[0][0].Loch        = D_PCCH;
CmacConfigPar->LochInfo[1][0].Loch        = D_CTCH;
```

See “ InitializeParameter_S_CCPCH_PCHxFACH_BMC()” that exists in “ Ini-tializeParameter.c” .

- If "FACH for CTCH" and "PCH" are mapped on same S_CCPCH, set parameter so that a timing of both does not compete.
  If they compete, PCH is preceded.

3.7.25 RLC AM Reconfiguration Function

3.7.25.1 Outline

The RLC AM Reconfiguration function is to change RLC parameters except for RLC PDU size without initializing SN for downlink, uplink or both directions.

3.7.25.2 How to set this function

Setting the global variable RlcAMReconf for CrlcConfig(CRLC_AM_ESTABLISH) in a scenario enables to perform this function. Setting the global variable RlcDirection specifies direction (downlink, uplink or both).

A concrete set value is indicated as follows.

Variable name	Setting Value	Contents
RlcAMReconf	RLC_AM_RECONFIG	Re-configures RLC AM. (Keep on SN.)
RlcDirection	RLC_DIRECTION_BOTH (Initial value)	Sets both directions (uplink and downlink).
	RLC_DIRECTION_SND	Sets downlink only.
	RLC_DIRECTION_RCV	Sets uplink only.

- RLC AM Reconfiguration

RLC AM is re-configured when executing CrlcConfig() with RLC_AM_CONFIG.

The following parameters are set in this case when RlcDirection is set to RLC_DIRECTION_SND.

- Setting of Tx_Window_Size
- Setting of Trigger
- Setting of Timer
- Updating of VT(MS)

The following parameters are set when RlcDirection is set to RLC_DIRECTION_RCV as well.

- Setting of Rx_Window_Size
- Updating of VR(MR)
- Deleting of received PDUs exceeding Rx_Window_Size

Moreover, when using initial value of RlcDirection or RlcDirection is set to RLC_DIRECTION_BOTH, all the settings enumerated in the above-mentioned are reflected. (Refer to section 9.7.9 of 3GPP TS 25.322 about re-configure.)

3.7.25.3 Restrictions

1) Since the variable RlcAMReconf and RlcDirection is already defined at

the time of staring on Control Software start-up, it is unnecessary to define them in the scenario again.

2) Since RlcAMReconf and RlcDirection are global variables, they are not automatically changed once values have set unless the variable values are consciously changed.
3) These variables will be ignored when Argument: RlcMode of Function: CrlcConfig() is except CRLC_AM_ESTABLISH or CRLC_AM_ESTABLISH_ WITH_CIPHERING.
4) These variables will be ignored when RLC isn' t activated. So it is impossible to activate whether uplink or downlink when activating at first.
5) When RlcAMReconfig is not RLC_AM_RECONFIG, the working of MD8480 is not guaranteed when CrlcConfig(CRLC_AM_ESTABLISH) or CrlcConfig(CRLC_AM_ESTABLISH_WITH_CIPHER) is executed for configured AM RLC. (Please refer 1.5 Restriction items (7))

3.7.26 RLC UM Special Length Indicator

3.7.26.1 Outline

This function is to do transmission of Uplink UM Data PDU that supported Release5. In the past (Release99), to use Special LI was sent by higher layer when sending UM Data. However, there were not Special LI use/unused, to be a concept that Release5 got to use always Special LI. (For the details of Special LI, refer to "3GPP TS 25.322 9.2.2.8 Length Indicator(LI)")

3.7.26.2 How to setup of the function

SpecVersion member of CrlcStartPar Structure into SimulatorStart() using that is set Release5 or Release99 of specification execute.
Do the following setting before execution of SimulatorStart( ).

```
CrlcStartPar.SpecVersion = SPEC_RELEASE_5;
SimulatorStart ( );
```

By this setting, it transmits RLC to UM Data PDU using Special LI.
The setting value of CrlcStartPar.SpecVersion is as follows.

Member	Setting value	
CrlcStartPar.SpecVersion	SPEC_RELEASE_99	Performs at Release99 (default value)
	SPEC_RELEASE_5	Performs at Release 5

A. SCENARIO LIBRARIES

This section summarizes the functions used for scenarios of the Signalling Tester. Table A-1 shows a list of functions. Section A.1 details each function. Section A.2 summarizes variables used in the functions. Section A.3 shows the initial value of each variable.

Table A-1 Scenario Library Function list

Function name	Feature
SimulatorStart()	Activation processing
WaitTime()	Specified wait time
SequenceBtn()	Button display and press wait
SequenceStr()	Character string input
SequenceDisp()	Character string display
SndMessage()	Message transmission
RcvMessage()	Message reception
CrlcConfig ()	Setting of RLC
CmacConfig ()	Setting of MAC
CphyTrchConfig()	Setting of PHY
CphyTrchRelease()	Release of PHY
CphyRlSetup()	Setting of PHY
CphyRlSetupMtgp()	Setting of PHY for MultiTGP
CphyRlRelease()	Release of PHY
CphyRlModify()	Modification to PHY
CphyCfnInd()	Request PHY to notify a scenario of a specified timing at the CFN
BtsPower()	Modification to transmission power of each BTS
BtsPowerActTime()	Modification to transmission power of each BTS
BtsAttnuator()	Setting of transmission/reception attenuator of each BTS
BtsReadCFN()	Reads the current CFN value.
BtsReadSFN()	Reads the current SFN value.
BtsDownTPCBit()	Sets downlink TPC Bit.
BtsMeasure()	Measurement (Counting) of BLER, BTFD
BtsFrequency()	Modification to transmit/receive frequency
CalcRMParameter()	Calculates Rate Matching Parameter in Normal Mode.
CalcRMParameterCM()	Calculates Rate Matching Parameter in Normal Mode and Compressed Mode.
CalcRMParameterMtgp()	Calculates Rate Matching Parameter in Normal Mode and Compressed Mode for MultiTGP.
CteConfig ()	Setting of logical channel and service type
CteRelease()	Release of logical channel and service type
CteConnect()	Connects a call to TE.
CteDisconnect()	Disconnects a call to TE.
ReplaceIE()	Overwrites a data in bit.
ExtractIE()	Fetches a data in bit.
Int2MsbIE()	Moves INT type (32bit) data to MSB.
Short2MsbIE()	Moves SHORT type data (16bit) to MSB.
Msb2IntIE()	Transforms MSB packed data to INT type (32bit).
Msb2ShortIE()	Transforms MSB packed data to SHORT type(16bit).
BtsOcnsActivate()	Starts the OCNS transmission
BtsOcnsDeactivate()	Halts the CNS transmission.
BtsOcnsPower()	Starts/ halts OCNS Power Control.
BtsOcnsPowerActTime()	Starts/ halts OCNS Power Control.
CpdcpConfig()	Configures PDCP
CpdcpRelease()	Releases PDCP

CpdcpReloc()	Acquires an uplink reception SN / downlink transmission SN
CpdcpSeqNum()	Transmits an uplink reception SN / downlink transmission SN
WcdmaRcvControl()	Receive control primitives in W-CDMA
CpdcpFwd()	The data preserved in the specified PDCP entity is transmitted to SNDCP.
PdcpRcvControl()	Receive control primitives in PDCP.
CteSuspend()	Stops data transmission and reception for the assigned TE Type temporarily and received data is preserved.
CteResume()	Restarts data transmission and reception for the assigned TE Type.
CbmcConfig()	Setting of BMC
CbmcRelease()	Release of BMC
SndBMCMessage()	Setting of BMC message and the transmission of BMC message

A.1 DETAILS OF EACH FUNCTION

Function name	INT SimulatorStart(INT Mode,INT Timeout)			
Feature summary	Activates Signalling Tester			
	Type	Argument name	Description	I/O
Argument	INT INT	Mode Timeout	Operation mode (Set to 0.) Timeout value	Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	
Feature details				

SimulationStar() activates Signalling Tester
Execute this function once before using other scenario libraries.

1) timeout
   Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT WaitTime (INT Time)			
Feature summary	Waits for specified time			
	Type	Argument name	Description	I/O
Argument	INT	Time	Wait time (in msec, in 100 msec steps)	Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	
Feature details				

1) timeout

Specify the wait time in msec and in 100 msec steps. If the wait time is set in less than 100 msec, it will perform, however the wait time sometimes may become unstable.

Remarks	

Function name	INT SequenceBtn (CHAR* Title, CHAR *Btn1, CHAR *Btn2, CHAR *Btn3,INT Timeout)			
Feature summary	Displays buttons and waits for button input.			
	Type	Argument name	Description	I/O
Argument	CHAR *	Title	Title (character string of up to 12 characters)	Input
	CHAR *	Btn1	Display of button 1 (character string of up to 8 characters)	Input
	CHAR *	Btn2	Display of button 2 (character string of up to 8 characters)	Input
	CHAR *	Btn3	Display of button 3 (character string of up to 8 characters)	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	>0 <0	Pressed button no. Timeout, abnormal ending	
Feature details				

1) timeout

Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Note; Buttons to be displayed are always three. If the selective contents are 2,a button without character strings is displayed and it cannot be used.

Remarks	

Function name	INT　SequenceStr (CHAR *Title,CHAR *str,INT Timeout);			
Feature summary	Inputs character strings.			
	Type	Argument name	Description	I/O
Argument	CHAR *	Title	Title (character string of up to 12 characters)	Input
	CHAR *	Str	Input character string (max. 255 characters)	Output
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	=0	Pressed button number	
		≠0	Timeout, abnormal ending	
Feature details				
1) timeout Specify the wait time in msec and in 100 msec steps.　If verification cannot be made within this period, the processing ends abnormally.　Specifying NO_TIMEOUT inhibits timeout.				
Remarks				

Function name	INT SequenceDisp (CHAR* String)			
Feature summary	Displays character strings.			
	Type	Argument name	Description	I/O
Argument	CHAR *	String	Displayed character string (up to 100 characters)	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending	
Feature details				
Remarks	It can extract and display only to 50 lines.			

Function name	INT SndMessage(INT BtsNo, INT Frame, INT Loch, INT LochNo,CHAR* message, INT Length)			
Feature summary	Transmits messages.			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	Frame	Transmit primitive	Input
	INT	Loch	Logical Channel	Input
	INT	LochNo	Logical Channel Number	Input
	CHAR *	Message	Transmit data (up to 1600 byte)	Input
	INT	Length	Transmit data length	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending	

Feature details

1) UNIT_BTS1,UNIT_BTS2,UNIT_BTS3
   (UNIT_BTS2 and UNIT_BTS3 are available when Logical Channel is set to BCCH).
2) You can specify the following in Frame:

RLC_AM_DATA_REQ	Transmission to RLC AM-SAP
RLC_UM_DATA_REQ	Transmission to RLC UM-SAP
RLC_TR_DATA_REQ	Transmission to RLC TR-SAP
TE_Q931_REQ	Transmission of Q.931 messages to TE (only when TE is ISDN.)

3) You can specify the following in Loch:
   · For RLC_AM_DATA_REQ, RLC_UM_DATA_REQ, RLC_TR_DATA_REQ:
   Logical Channel: D_BCCH, D_PCCH, D_CCCH, D_DCCH, D_DTCH
   ·TE_Q931_REQ
   Logical Channel: DTCH
4) You can specify 0 to 7 in LochNo for each logical channel.
5) Specify transmit data up to 1600 byte (octet).
6) Use the following global variables in order to specify detailed parameters.　Set these global variables before executing SndMessage().
   · When Loch is BCCH or BCH:

SIB_REP	SIB_REP value[2-4096, in units of 10 msec]
SIB_POS	SIB_POS value[0-4094, in units of 10 msec]

   · When Loch is PCCH or PCH:

PageIndicator	Page Indicator value[0-144]
PageNp	Number of Page indicators within a frame[18,36,72,144]
DRXCycleLength;	DRX Cycle Length[1-4096]
PagingBlock;	Paging Block[0-4095]

   · When Frame is RLC_AM_DATA_REQ:

RlcMUI	MUI value[0-32767]
RlcCNF	1 when using transmission verification with MUI, otherwise 0.

   · When Frame is RLC_UM_DATA_REQ:

RlcSpecialLI	1 when using Special LI, otherwise 0. (This variable is not referred when SPEC_RELEASE_5 is set on 3.7.26 RLC UM Special Length Indicator Function. In that case, Special LI is always used.)

Remarks

Function name	INT RcvMessage (INT *BtsNo, INT *Frame, INT *Loch, INT LochNo,CHAR* message, INT Timeout)			
Feature summary	Receives messages.			
	Type	Argument name	Description	I/O
Argument	INT *	BtsNo	BTS No	Output
	INT *	Frame	Received primitive	Output
	INT *	Loch	Logical Channel or Symbolize the Preamble	Output
	INT *	LochNo	Logical Channel Number	Output
	CHAR *	Message	Received data	Output
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	>0	Normal ending (Receive data length)	
		<0	Abnormal ending (Timeout)	

Feature details

1) In case Frame is PHY_REPORT_IND in BtsNo.
In others, BtsNo is not used.

2) The following are returned in Frame:

RLC_AM_DATA_IND	Data reception from RLC AM-SAP
RLC_AM_DATA_CNF	MUI verification from RLC AM-SAP
RLC_UM_DATA_IND	Data reception from RLC UM-SAP
RLC_TR_DATA_IND	Data reception from RLC TR-SAP
TE_Q931_IND	Q.931 message reception from TE (only when TE is ISDN.)
PHY_REPORT_IND	Preamble reception (only when Preamble report function is available.)

3) The following are returned in Loch:
- a)For RLC_AM_DATA_IND, RLC_AM_DATA_CNF, RLC_UM_DATA_IND, RLC_TR_DATA_IND:
  Logical Channel: U_BCCH, U_PCCH, U_CCCH, U_DCCH, U_DTCH
- b)TE_Q931_IND
  Logical Channel : DTCH
- c)For PHY_REPORT_IND
  Symbolize the Preamble: U_PREAMBLE

4) Numerals 0-7 are returned in LochNo for each Logical Channel.
(Note) For PHY_REPORT_IND 0 is returned in LochNo.

5) Detailed parameters of a received message are specified in the following global variables:
- In case Frame is RLC_AM_DATA_CNF:
  RlcMUI Verified MUI value[0-32767]

6) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
Specifying NO_TIMEOUT inhibits timeout.

Remarks

Function name	INT CrlcConfig(INT BtsNo, INT RlcMode, INT Loch, INT LochNo, CRLC_CONFIG_PAR *RlcParam, INT UpperLayer, INT Timeout)			
Feature summary	Instructs RLC to activate or halt AM, UM, or TR mode.			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No (fixed to UNIT_BTS1)	Input
	INT	RlcMode	Specifies RLC operation mode.	Input
	INT	Loch	Logical Channel	Input
	INT	LochNo	Logical Channel No	Input
	CRLC_CONFIG_PAR*	RlcParam	RLC operation parameter	Input
	INT	UpperLayer	Uplink data transmission layer	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending (Timeout)	

Feature details

1) BtsNo is not in use now.　(Specify UNIT_BTS1.)
2) You can specify the following in RlcMode:
   CRLC_AM_ESTABLISH : Activation of Acknowledge Mode
   CRLC_AM_RELEASE : Release of Acknowledge Mode
   CRLC_UM_ESTABLISH : Activation of Un-acknowledge Mode
   CRLC_UM_RELEASE : Release of Un-acknowledge Mode
   CRLC_TR_ESTABLISH : Activation of Transparent Mode
   CRLC_TR_RELEASE : Release of Transparent Mode
   CRLC_AM_STOP : Stop the data of Acknowledge Mode RLC
   CRLC_UM_STOP : Stop the data of Un-acknowledge Mode RLC
   CRLC_AM_CONTINUE : Continue the data of Acknowledge Mode RLC
   CRLC_UM_CONTINUE : Continue the data of Un-acknowledge Mode RLC

   When execute RLC Stop or RLC Continue, please use above parameter same as establishment of RLC. In this case, please select CRLC_AM_STOP or CRLC_AM_CONTINUE for RLC AM entity, and CRLC_UM_STOP or CRLC_UM_CONTINUE for RLC UM entity. For RLC TM entity, neither RLC Stop nor RLC Continue can execute.

3) You can specify the following in Logical Channel (Note: there is no distinction between DownLink and UpLink.)
   BCCH, PCCH, CCCH, DCCH, DTCH, CTCH
4) You can specify 0 to 7 in Logical Channel No for each logical channel.
5) For RlcParam, specify the pointer to the variable for CrlcConfig() described under A.2 VARIABLE USED IN EACH FUNCTION.　You can specify (CRLC_CONFIG_PAR *) 0 only in the Transparent Mode.
6) Specify a layer to which received data is passed in UpperLayer.
   Typically specify TE when Logical Channel is U_DTCH or D_DTCH; otherwise RRC.
7) TIMEOUT
   Specify the wait time in msec and in 100 msec steps.　If verification cannot be made within this period, the processing ends abnormally.
   Specifying NO_TIMEOUT inhibits timeout.

Remarks	Refer to 3.7.25 Chapter about RLC AM reconfiguration function which reconfigures AM RLC under operation. Refer to 3.7.13 Chapter about AM-RLC Asymmetric payload size setting function.

Function name	INT CmacConfig(INT BtsNo, INT Phch, INT PhchNo, CMAC_CONFIG_PAR *MacParam,INT ActTime, INT Timeout)			
Feature summary	Makes setting (CONFIG) per CCtrCH for MAC layer.			
	Type	Argument name	Description	I/O
Argument	INT INT INT CMAC_CONFIG_PAR* INT INT	BtsNo Phch PhchNo MacParam ActTime Timeout	BTS No Physical Channel Physical Channel No MAC Operation Parameter Activation Time Timeout value	Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending (Timeout)	

Feature details

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
   Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
   BASIC: BtsNo that can be specified for DownLink: UNIT_BTS1
   BtsNo that can be specified for UpLink: UNIT_BTS1
   SHO1, SHO2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
   BtsNo that can be specified for UpLink: UNIT_BTS1
   HHO: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
   BtsNo that can be specified for UpLink: UNIT_BTS1, UNIT_BTS2
   TxDiversity1, TxDiversity2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
   BtsNo that can be specified for UpLink UNIT_BTS1
2) You can specify the following values in Physical Channel:
   DownLink : P_CCPCH, S_CCPCH, D_DPCH
   UpLink : U_PRACH, U_DPCH
   Note) CmacConfig() is set per CCtrCH. Since CCtrCH has no general identification symbols, it uses physical channels.
3) You can specify 1 or 0 in Physical Channel No only when Physical Channel is S_CCPCH. For other Physical Channels, specify 0.
4) For MacParam, specify the pointer to the variable for CmacConfig() described under A.2 VARIABLE USED IN EACH FUNCTION. To halt operation of MAC, specify (CMAC_CONFIG_PAR *)0.
5) Specify a CFN value that enables setting in Activation Time. You can set the following values:
   When not specifying CFN: ACTIVATE_NOW
   When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
   *1 You can read the current CFN by using BtsReadCFN(). Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time. (This value may vary depending on the processing speed of your PC.)
   *2 Specify a CFN so that it may be the least common multiple of TTIs of all the Transport Channels used for the specified Physical Channel.
6) TIMEOUT
   Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
   Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT CphyTrchConfig(INT BtsNo, INT Phch, INT PhchNo, CPHY_TRCH_CONFIG_PAR *PhyParam, INT ActTime,INT Timeout)			
Feature summary	Makes setting (CONFIG) per CCtrCH for PHY layer.			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	Phch	Physical Channel	Input
	INT	PhchNo	Physical Channel No	Input
	PHY_TRCH_CONFIG_PAR*	PhyParam	PHY Transport Channel Parameter	Input
	INT	ActTime	Activation Time	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending (Timeout)	

Feature details

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
BASIC: BtsNo that can be specified for DownLink: UNIT_BTS1
BtsNo that can be specified for UpLink: UNIT_BTS1
SHO1, SHO2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
BtsNo that can be specified for UpLink: UNIT_BTS1
HHO: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink: UNIT_BTS1, UNIT_BTS2
TxDiversity1, TxDiversity2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink UNIT_BTS1
2) You can specify UNIT_BTS1 and UNIT_BTS2 when BaseBand and RF Connection is set to "SHO1".
You can specify the following values in Physical Channel:
DownLink : P_CCPCH, S_CCPCH, D_DPCH
UpLink : U_PRACH, U_DPCH
Note) CphyTrchConfig() is set per CCtrCH. Since CCtrCH has no general identification symbols, it uses physical channels.
3) You can specify 1 or 0 in Physical Channel No only when Physical Channel is S_CCPCH. For other Physical Channels, specify 0.
4) For PhyParam, specify the pointer to the variable for CphyTrchConfig() described under A.2 VARIABLE USED IN EACH FUNCTION. Before executing CphyTrchConfig(), execute CalcRMParameter() and calculate Rate Matching Parameter.
5) Specify a CFN value that enables setting in Activation Time. You can set the following values:
When not specifying CFN: ACTIVATE_NOW
When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
*1 You can read the current CFN by using BtsReadCFN(). Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time. (This value may vary depending on the processing speed of your PC.)
*2 Specify a CFN so that it may be the least common multiple of TTIs of all the Transport Channels used for the specified Physical Channel.
6) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT CphyTrchRelease (INT BtsNo, INT Phch, INT PhchNo, INT ActTime INT Timeout)			
Feature summary	Releases (Halts) Transport Channel per CCTrCH in the PHY layer.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT INT	BtsNo Phch PhchNo ActTime Timeout	BTS No Physical Channel Physical Channel No Activation Time Timeout value	Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending (Timeout)	
Feature details				

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
   Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
   BASIC: BtsNo that can be specified for DownLink: UNIT_BTS1
   BtsNo that can be specified for UpLink: UNIT_BTS1
   SHO1, SHO2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
   BtsNo that can be specified for UpLink: UNIT_BTS1
   HHO: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
   BtsNo that can be specified for UpLink: UNIT_BTS1, UNIT_BTS2
   TxDiversity1, TxDiversity2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
   BtsNo that can be specified for UpLink UNIT_BTS1
2) You can specify the following values in Physical Channel:
   DownLink : P_CCPCH, S_CCPCH, D_DPCH
   UpLink : U_PRACH, U_DPCH
   Note) CphyTrchRelease() is set per CCtrCH. Since CCtrCH has no general identification symbols, it uses physical channels.
3) You can specify 1 or 0 in Physical Channel No only when Physical Channel is S_CCPCH. For other Physical Channels, specify 0.
4) Specify a CFN value that enables setting in Activation Time. You can set the following values:
   When not specifying CFN: ACTIVATE_NOW
   When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
   *1 You can read the current CFN by using BtsReadCFN(). Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time. (This value may vary depending on the processing speed of your PC.)
   *2 Specify a CFN so that it may be the least common multiple of TTIs of all the Transport Channels used for the specified Physical Channel.
5) timeout
   Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
   Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT CphyRlSetup(INT BtsNo, INT Phch, INT PhchNo,CPHY_RL_SETUP_PAR *PhyParam, INT ActTime,INT Timeout)			
Feature summary	Activates Radio Link (SETUP) for PHY.			
	Type	Argument name	Description	I/O
Argument	INT INT INT PHY_RL_SETUP_PAR* INT INT	BtsNo Phch PhchNo PhyParam ActTime Timeout	BTS No Physical Channel Physical Channel Number PHY Radio Link Parameter Activation Time Timeout value	Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending (Timeout)	
Feature details				

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
BASIC: BtsNo that can be specified for DownLink: UNIT_BTS1
BtsNo that can be specified for UpLink: UNIT_BTS1
SHO1, SHO2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
BtsNo that can be specified for UpLink: UNIT_BTS1
HHO: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink: UNIT_BTS1, UNIT_BTS2
TxDiversity1, TxDiversity2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink UNIT_BTS1
2) You can specify the following values in Physical Channel:
DownLink : P_SCH,S_SCH,P_CPICH,S_CPICH,P_CCPCH,S_CCPCH, PICH,AICH,D_DPCH,AWGN
UpLink : U_PRACH,U_DPCH
Note1) CphyRlSetup() is set per CCtrCH. Since CCtrCH has no general identification symbols, it uses physical channels.
Note2) S_CPICH should be used exclusively with PICH.(S_CPICH and PICH can not be used at the same time.)
3) You can specify 1 or 0 in Physical Channel No only when Physical Channel is S_CCPCH. For other Physical Channels, specify 0.
Note3) When Physical Channel=S_CCPCH and Physical Channel No=1 are specified, DPCH must be used exclusively.
4) For PhyParam, specify the pointer to the variable for CphyRlsetup() described under A.2 VARIABLE USED IN EACH FUNCTION.
5) Specify a CFN value that enables setting in Activation Time. You can set the following values:
When not specifying CFN: ACTIVATE_NOW
When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
*1 You can read the current CFN by using BtsReadCFN(). Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time. (This value may vary depending on the processing speed of your PC.)
*2 Specify a CFN so that it may be the least common multiple of TTIs of all the Transport Channels used for the specified Physical Channel.
6) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.
7) To specify detailed parameters, use the following global variables. Set these global variables before executing CphyRlSetup().
· When channel is PICH:
PageNp Np[18, 36, 72, 144]

Remarks	In case CphyRlSetup () function is used continuously for Uplink of BTS1 and BTS2, the first function takes 40ms for finishing the task. ex) CphyRlSetup(UNIT_BTS1,….) CphyRlSetup(UNIT_BTS2,….) When the following function performed before the first function finish the task, the following function becomes ineffective.

Function name	INT CphyRlSetupMtgp(INT BtsNo, INT Phch, INT PhchNo, CPHY_RL_SETUP_MTGP_PAR *PhyParam, INT ActTime, INT Timeout)			
Function summary	Activates Radio Link (SETUP) for PHY using structure CPHY_RL_SETUP_MTGP_PAR.			
	Type	Argument name	Description	I/O
Argument	INT INT INT PHY_RL_SETUP_MTGP_PAR* INT INT	BtsNo Phch PhchNo PhyParam ActTime Timeout	BTS No Physical Channel Physical Channel Number PHY Radio Link Parameter Activation Time Timeout value	Input Input Input Input Input Input
	Type	Argument value	Description	
Function value	INT	= 0 ≠ 0	Normal ending Abnormal ending (Timeout)	

Feature details

1) You can specify UNIT_BTS1,UNIT_BTS2,or UNIT_BTS3 in BtsNo.

Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.

BASIC: BtsNo that can be specified for DownLink: UNIT_BTS1
BtsNo that can be specified for UpLink: UNIT_BTS1

SHO1, SHO2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
BtsNo that can be specified for UpLink: UNIT_BTS1

HHO: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink: UNIT_BTS1, UNIT_BTS2

TxDiversity1, TxDiversity2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink UNIT_BTS1

2) You can specify the following values in Physical Channel:

DownLink :D_DPCH
UpLink :U_DPCH

Note1) CphyRlSetup() is set per CCTrCH. Since CCtrCH has no general identification symbols, it uses physical channels.

3) For PhyParam, specify the pointer to the variable for CphyRlsetupMtgp() described under A.2 VARIABLE USED IN EACH FUNCTION.

4) Specify a CFN value that enables setting in Activation Time. You can set the following values:

When not specifying CFN: ACTIVATE_NOW

When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.

*1 You can read the current CFN by using BtsReadCFN(). Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time. (This value may vary depending on the processing speed of your PC.)

*2 Specify a CFN so that it may be the least common multiple of TTIs of all the Transport Channels used for the specified Physical Channel.

5)timeout

Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.Specifying NO_TIMEOUT inhibits timeout.

Remarks	Use CalcRMParameterMtgp() for calculation of Rate Matching parameter.

Function name	INT CphyRlRelease (INT BtsNo, INT Phch,INT PhchNo, INT ActTime,INT timeout)			
Feature summary	Halts (Releases) Radio Link for PHY.			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	Phch	Physical Channel	Input
	INT	PhchNo	Physical Channel Number	Input
	INT	ActTime	Activation Time	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending (Timeout)	

Feature details

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
BASIC: BtsNo that can be specified for DownLink: UNIT_BTS1
BtsNo that can be specified for UpLink: UNIT_BTS1
SHO1, SHO2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
BtsNo that can be specified for UpLink: UNIT_BTS1
HHO: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink: UNIT_BTS1, UNIT_BTS2
TxDiversity1, TxDiversity2: BtsNo that can be specified for DownLink: UNIT_BTS1, UNIT_BTS2
BtsNo that can be specified for UpLink UNIT_BTS1
2) You can specify the following values in Physical Channel:
DownLink : P_SCH,S_SCH,P_CPICH,S_CPICHP_CCPCH,S_CCPCH, PCIH,AICH,D_DPCH,AWGN
UpLink : U_PRACH, U_DPCH
Note) CphyRlRelease () is set per CCtrCH. Since CCtrCH has no general identification symbols, it uses physical channels.
3) You can specify 1 or 0 in Physical Channel No only when Physical Channel is S_CCPCH. For other Physical Channels, specify 0.
4) Specify a CFN value that enables setting in Activation Time. You can set the following values: When not specifying CFN: ACTIVATE_NOW When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
*1 You can read the current CFN by using BtsReadCFN(). Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time. (This value may vary depending on the processing speed of your PC.)
*2 Specify a CFN so that it is be the least common multiple of TTIs of all the Transport Channels used for the specified Physical Channel.
5) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks

Function name	INT CphyRlModify (INT BtsNo, INT Phch,INT PhchNo, PHY_RL_PARAM *PhyParam, INT ActTime,INT Timeout)			
Feature summary	Modifies the setting of Radio Link for PHY.			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	Phch	Physical Channel	Input
	INT	PhchNo	Physical Channel Number	Input
	PHY_RL_SETUP_PAR*	PhyParam	PHY Radio Link Parameter	Input
	INT	ActTime	Activation Time	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending (Timeout)	
Feature details				
Re-activates Radio Link. This feature is the same as CphyRlSetup(). For details, see CphyRlSetup().				
Remarks				

Function name	INT CphyCfnInd(INT BtsNo, INT IndCFN, INT SkpCfnCycle, INT Timeout)			
Feature summary	Request PHY to notify a scenario of a specified timing at the CFN			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	IndCFN	Indicated CFN	Input
	INT	SkpCfnCycle	Skipped CFN cycle	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	= 0	Normal ending	
		<> 0	Abnormal ending (Timeout)	
Feature details				
1) Set UNIT_BTS1 to BtsNo. 2) Specify a CFN value in IndCFN, that is a timing which should be notified to your scenario. The range of this parameter : 0-255 (A value over 255 can be specified; in this case, the remainder of division of the value by 256 is effective.) 3) Specify a skipped CFN cycle value in SkpCfnCycle until CFN reaches to a value of IndCFN. The range of this parameter : 0-255 (A value over 255 can be specified; in this case, the remainder of division of the value by 256 is effective.) 4) Timeout Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.				
Remarks				

Function name	INT BtsAttenuator(INT BtsNo, INT TxAtt, INT RefPower, INT Timeout)			
Feature summary	Sets TX Attenuator and RX Reference Power.			
	Type	**Argument name**	**Description**	**I/O**
Argument	INT	BtsNo	BTS No	Input
	INT	TxAtt	Tx Attenuator	Input
	INT	RxRefPower	Rx reference Power	Input
	INT	Timeout	Timeout value	Input

	Type	Function value	Description
Function value	INT	=0	Normal ending
		≠0	Abnormal ending

Feature details

1) You can specify UNIT_BTS1, UNIT_BTS2 in BtsNo.
   But "BtsNo" means RF_No.
   Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
   BASIC: BtsNo that can be specified is UNIT_BTS1.
   If you specify UNIT_BTS1, Tx Attenuator and Rx reference Power of BTS1 will change.
   SHO1 : BtsNo that can be specified is UNIT_BTS1.
   If you specify UNIT_BTS1, Tx Attenuator of BTS1,BTS2 and BTS3 will change and Rx reference Power of BTS1 will change.
   SHO2: BtsNo that can be specified is UNIT_BTS1 and BTS2.
   If you specify UNIT_BTS1, Tx Attenuator and Rx reference Power of BTS1 will change.
   If you specify UNIT_BTS2, Tx Attenuator of BTS2 and BTS3 will change.
   HHO: BtsNo that can be specified is UNIT_BTS1 and UNIT_BTS2.
   If you specify UNIT_BTS1, Tx Attenuator and Rx reference Power of BTS1 will change.
   If you specify UNIT_BTS2, Tx Attenuator and Rx reference Power of BTS2 will change.
   TxDiversity1: BtsNo that can be specified is UNIT_BTS1.
   If you specify UNIT_BTS1, Tx Attenuator of BTS1 and BTS2 will change and Rx Reference Power of BTS1 will change.
   TxDiversity2: BtsNo that can be specified is UNIT_BTS1.
   If you specify UNIT_BTS1, Tx Attenuator and Rx reference Power of BTS1 will change.
   If you specify UNIT_BTS2, Tx Attenuator of BTS2 will change.
2) TxAtt specification method:
   0(0dB)-1200(120.0dB)　 in 0.1 dB, in 0.1 dB steps

* When Main is specified for Connector on the SETUP screen, a value that 15 dB is added to the one set on the scenario should be set as a value of Tx Attenuator.

3) RxRefPower specification method:
   +50(5dB)- -400(-40.0dB)　in 0.1 dB, in 0.1 dB steps

* When Main is specified for Connector on the SETUP screen, a value that 20dB is added to the one set on the scenario should be set as a value of Reference Power

4) timeout
   Specify the wait time in msec and in 100 msec steps.　If verification cannot be made within this period, the processing ends abnormally.
   Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT BtsPower(INT BtsNo, INT Ph_CH, INT UpDown, INT Timeout)			
Feature summary	Increments/Decrements power per channel.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT	BtsNo Ph_Ch UpDown Timeout	BTS No Physical Channel Tx Power Up Down Timeout value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	

Feature details

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
BASIC: BtsNo that can be specified: UNIT_BTS1
SHO1,SHO2: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
HHO: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2
TxDiversity1,TxDiversity2: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3 (Both UNIT_BTS2 and UNIT_BTS3 are for setting of Antenna2.)

2) You can specify the following 3 kinds by specifying Physical Channel and UpDown:
Incremental/Decremental value per BTS:
Physical Channel: 0
UpDown: Incremental/Decremental value of Power in 1dB,-99~99
All Powers for each channel of specified BTS increments/decrements. The minimum value is -99dBm; the maximum value is -10dBm.(P_SCH, S_SCH: -13dBm; AWGN: -8dBm). Modifying the power beyond this range sets the minimum or maximum values.
Increment/decrement of Power of each channel:
Physical Channel: P_SCH, S_SCH, P_CPICH, S_CPICH, P_CCPCH, S_CCPCH, PICH, AICH, D_DPCH, AWGN
UpDown: Incremental/Decremental value of Power in 1dB,-99~99
The only power for each channel of specified BTS increments/decrements. The minimum value is -99dBm; the maximum value is -10dBm.(P_SCH, S_SCH: -13dBm; AWGN: -8dBm). Modifying the power beyond this range sets the minimum or maximum values.
Automatic control of downlink DPCH Power:
Physical Channel: D_DPCH
UpDown: POWER_AUTO_TPC_05 automatically controls it in 0.5dBSteps
POWER_AUTO_TPC_1 automatically controls it in 1dBSteps
POWER_AUTO_TPC_15 automatically controls it in 1.5dBSteps
POWER_AUTO_TPC_2 automatically controls it in 2dBSteps
0 releases automatic control
They control Downlink DPCH power automatically with adjusting Uplink TPC Bit value.
Specify 0 in UpDown to release automatic control for DPCH power.
Right after the activation of Downlink DPCH, the automatic control is not performed. (It is fixed to the power specified in the activation.)

3) TIMEOUT
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
Specifying NO_TIMEOUT inhibits timeout.

Remarks

Function name	INT BtsPowerActTime(INT BtsNo, INT Ph_CH, INT UpDown, INT ActTime, INT Timeout)			
Feature summary	Increments/decrements channel power.			
	Type	Argument name	Description	I/O
Argument	INT	BtsNo	BTS No	Input
	INT	Ph_Ch	Physical Channel	Input
	INT	UpDown	Tx Power Up Down	Input
	INT	ActTime	Activation Time	Input
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending	
Feature details				

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
BASIC: BtsNo that can be specified: UNIT_BTS1
SHO1,SHO2: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
HHO: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2
TxDiversity1,TxDiversity2:BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3 (Both UNIT_BTS2 and UNIT_BTS3 are for the setting of Antenna2.)

2) You can specify the following 3 kinds by specifying Physical Channel and UpDown:
Incremental/Decremental value per BTS::
Physical Channel: 0
UpDown: Incremental/Decremental value of Power in 1dB,-99~99
All Powers for each channel of specified BTS increments/decrements. The minimum value is -99dBm; the maximum value is -10dBm.(P_SCH, S_SCH: -13dBm; AWGN: -8dBm). Modifying the power beyond this range sets the minimum or maximum values.

Increment/decrement of Power of each channel
Physical Channel: P_SCH, S_SCH, P_CPICH, S_CPICH, P_CCPCH, S_CCPCH, PICH, AICH, D_DPCH, AWGN
UpDown: Incremental/Decremental value of Power in 1dB,-99~99
The only power for each channel of specified BTS increments/decrements. The minimum value is -99dBm; the maximum value is -10dBm.(P_SCH, S_SCH: -13dBm; AWGN: -8dBm). Modifying the power beyond this range sets the minimum or maximum values.

Automatic control of downlink DPCH Power:
Physical Channel: D_DPCH
UpDown: POWER_AUTO_TPC_05 automatically controls it in 0.5dBSteps
POWER_AUTO_TPC_1 automatically controls it in 1dBSteps
POWER_AUTO_TPC_15 automatically controls it in 1.5dBSteps
POWER_AUTO_TPC_2 automatically controls it in 2dBSteps
0 releases automatic control
They control Downlink DPCH power automatically with adjusting Uplink TPC Bit value.
Specify 0 in UpDown to release automatic control for DPCH power.
Right after the activation of Downlink DPCH, the automatic control is not performed. (It is fixed to the power specified in the activation.)

3)Timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
Specifying NO_TIMEOUT inhibits timeout.
4) Enter CFN value that enables the setup for Activation Time. The followings are available:
To specify CFN: ACTIVATE_NOW(In this case, the operation will be completely the same as one of BtsPower().)
To not specify CFN: 0~255 (Although 255 or above of values can be specified, the remainder of what divided by 256 will be valid.)
*1 Current CFN can be read by BtsReadCFN(). For CFN, specify +20 or more than the value read by BtsReadCFN() in consideration of the delay time of the Software. (This value may vary depending on processing rate of PC that is used.)

Remarks	

Function name	INT BtsReadCFN(INT BtsNo, INT Timeout)			
Feature summary	Reads the current CFN value.			
	Type	Argument name	Description	I/O
Argument	INT INT	BtsNo Timeout	BTS No (fixed to UNIT_BTS1) Timeout value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	< 0 ≠0	FN value Abnormal ending	
Feature details				

1) BtsNo is not in use now. (Specify UNIT_BTS1.)
2) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.
3) Reads the current CFN value; a delay time occurs due to software processing. (The delay time is not constant.)

Remarks	

Function name	INT BtsReadSFN(INT BtsNo, INT Timeout)			
Feature summary	Read the current SFN value.			
	Type	Argument name	Description	I/O
Argument	INT INT	BtsNo Timeout	BTS No Timeout value	Input Input
	Type	Function value	Description	
Function value	INT	0 - 4095 < 0	SFN value Abnormal ending (Timeout)	
Feature details				

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
2) Timeout
Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.
3) Reads the current CFN value; a delay time occurs due to software processing. (The delay time is not constant.)

Remarks	The SFN value is already calculated SFN offset value that specified in a scenario or parameter files. For using this function, you need Ver.5.30 or more.

Function name	INT BtsDownTPCBit(INT BtsNo, INT Mode,INT FrameLength,USHORT * TPC_Pattern,INT Timeout)			
Feature summary	Sets TCP bit of downlink DPCH.			
	Type	Argument name	Description	I/O
Argument	INT INT INT USHORT* INT	BtsNo Mode FrameLength TPC_Pattern Timeout	BTS No TPC transmission mode TPC transmission pattern length TPC transmission pattern Timeout value	Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠0	Normal ending Abnormal ending	

Feature details

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
   Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
   BASIC: BtsNo that can be specified: UNIT_BTS1
   SHO1,SHO2: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
   HHO: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2
   TxDiversity1,TxDiversity2: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3 (Both UNIT_BTS2 and UNIT_BTS3 are for the setting of Antenna2.)
2) You can specify the following TPC modes in Mode:
   TPC_AUTO Automatic control using the uplink DPCCH receiving power. (The target value is the set value in Reference Power.)
   TPC_SINGLE TPC transmit pattern specified in TPC_Pattern is transmitted once, followed by automatic control.
   TPC_CONTINUOUS TPC transmit pattern specified in TPC_Pattern is transmitted repeatedly.
3) Specify the valid TPC_Pattern frame count (1-10) in FrameLength.
4) Specify transmit TCP Bit for each frame unit (15 bits) in TPC_Pattern. When setting TPC bit values, Slot 0 corresponds to Bit 14, Slot1 to Bit 13,…,Slot 15 to Bit0.
   Example: /* Slot 0 1 2 3 4 5 6 7 8 910 11121314 */
   0x2AAA, /* 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 */
5) timeout

Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks

Function name	INT BtsMeasure(INT BtsNo, INT Channel, INT ChNo, CPHY_BTS_MEAS_PAR *MeasPar,INT Timeout)			
Feature summary	Measures (Counts) BLER or BTFD.			
	Type	Argument name	Description	I/O
Argument	INT INT INT CPHY_BTS_MEAS_PAR* INT	BtsNo Channel ChNo MeasPar Timeout	BTS No Tranport Channel/Logical Channel Tranport Channel/Logical Channel No Measurement parameter Timeout value	Input Input Input Input/Output Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	

Feature details

See section 3 for the operation.

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 in BtsNo.
   Values you can specify depend on the setting of BaseBand and RF Unit Connection on the Parameter Setup screen.
   BASIC: BtsNo that can be specified: UNIT_BTS1
   SHO1,SHO2: BtsNo that can be specified: UNIT_BTS1
   HHO: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2
   TxDiversity1,TxDiversity2: BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2
2) Specify Transport Channel or Logical Channel for Channel.
   For BLER (CRC), specify U_DCH.
   For BLER (RETRANS), specify DTCH or DCCH.
   For BTFD, specify U_DCH.
3) You can specify 0-7 per Transport Channel and per Logical Channel in Logical Channel No.
4) Specify the measurement parameters using the following structure:

```
typedef struct {
    UCHAR   Mode;       /*Specifies the measurement mode.  You can specify: */
            MEAS_BLER_CRC:      /*BLER (CRC) measurement */
            MEAS_BLER_RETRANS: /*BLER (RE-TRANSMISSION) measurement */
            MEAS_BTFD:          /*BTFD measurement */

    UCHAR   Tfci;           /*Specifies TFCI in BTFD measurement. */
    CHAR    Reserve1[2];
    UINT    Count;          /*Specifies the count of frames to be measured. */
    Union {
        struct {                /*Returns the result of BLER (CRC) measurement. */
            UINT    CrcOK;          /*CRC OK count */
            UINT    CrcNG;          /*CRC NG count */
            UINT    Reserve2[2];
            UINT    Total;          /*Number of actually measured frames */
        } BLER_Crc;
        struct {                /*Returns the result of BLER (RE-TRANSMISSION) measurement. */
            UINT    ReTransReq;     /*Number of PUs for retransmission requests from mobile station */
            UINT    Reserve2[3];
            UINT    Total;          /*Number of actually measured frames */
        } BLER_ReTrans;
        struct {                /*Returns the result of BTFD measurement. */
            UINT    CrcOK_TfciOK;   /*CRC=OK, TFCI=OK frame count */
            UINT    CrcOK_TfciNG;   /*CRC=OK, TFCI=NG frame count */
            UINT    CrcNG_TfciOK;   /*CRC=NG, TFCI=OK frame count */
            UINT    CrcNG_TfciNG;   /*CRC=NG, TFCI=NG frame count */
            UINT    Total;          /*Number of actually measured frames */
        } BTFD;
    } Result;
} CPHY_BTS_MEAS_PAR;
```

5) timeout
   Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.
   Caution) Whenever a value below Count*TTI is specified in Timeout, timeout will take place.

Remarks	

Function name	INT BtsFrequency (INT BtsNo,INT DownFreq,INT UpFreq,INT ActivationTime,INT Timeout)			
Feature summary	Modifies the uplink/downlink RF frequency.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT	BtsNo DownFreq UpFreq Timeout	BTS No Downlink frequency Uplink frequency Timeout value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	
Feature details				

1) You can specify UNIT_BTS1, UNIT_BTS2, or UNIT_BTS3 for BtsNo.
Values you can specify depend on the setting of BaseBand and RF Unit Connection　on the Parameter Setup screen.
BASIC:　BtsNo that can be specified: UNIT_BTS1
SHO1,SHO2:　BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2, UNIT_BTS3
HHO:　BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2
TxDiversity1,TxDiversity2:　BtsNo that can be specified: UNIT_BTS1, UNIT_BTS2

2) DownFreq
Specify the downlink frequency in Hz. Configurable values are 300(MHz)~3000(MHz).

3) UpFreq
Specify the uplink frequency in Hz.　Values that can be specified are those in higher digits than 1 MHz.　A value of 1 MHz or below is defaults to a value set in RF IF1 or RF IF2 on the Parameter Setup screen. Configurable values are 350(MHz)~550(MHz), 700(MHz)~1100(MHz), and 1400(MHz)~2200(MHz).
Example:
With the frequency set to 1920.2 MHz in RFIF on the Parameter Setup screen, executing
BtsFrequency (UNIT_BTS1,2137600000,1947600000,ACTIVATE_NOW,NO_TIMEOUT);
sets
a downlink frequency of 2137.6 MHz
and an uplink frequency of 1947.2 MHz, not 1947.6 MHz.

4) Specify a CFN value that enables setting in Activation Time.　You can set the following values:
When not specifying CFN:　ACTIVATE_NOW
When specifying CFN: 0-255 (A value above 255 can be specified; in this case, the remainder of division of the value by 256 is effective.)
*1 You can read the current CFN　by using BtsReadCFN().　Specify a value greater than the value read using BtsReadCFN() by at least 20 (200 msec or over) considering the software delay time.　(This value may vary depending on the processing speed of your PC.)
*2 Due to Hardware efficiency, it actually takes a few(ms) for the frequency to change even if Activation Time has been set.

5) timeout
Specify the wait time in msec and in 100 msec steps.　If verification cannot be made within this period, the processing ends abnormally.　Specifying NO_TIMEOUT inhibits timeout.

Caution:
Since this function switches over the frequencies while the system is in operation, one RF unit each for the uplink and for downlink can be used to conduct a simulated inter-frequency handover test.　This procedure, however, takes time for switchover of frequencies so that data in several frames before and after the specified Activation Time cannot be normally transmitted/received.

Remarks	

Function name	INT CalcRMParameter(INT Phch, CPHY_RL_SETUP_PAR* RlParam, CPHY_TRCH_CONFIG_PAR* PhyParam);			
Feature summary	Calculates Rate Matching Parameter in Normal Mode.			
	Type	Argument name	Description	I/O
Argument	INT CPHY_RL_SETUP_PAR* CPHY_TRCH_CONFIG_PAR*	Phch RlParam PhyParam	Physical Channel Radio Link parameter Transport parameter(Rate Matching parameter)	Input Input/output Input/output
	Type	Function value	Description	
Function value	INT	0 -1	Normal ending Maximum SymbolRate is too small. (Uplink)	
Feature details				

1) You can specify the following values in Physical Channel:
   DownLink: P_CCPCH, S_CCPCH, D_DPCH
   UpLink: U_PRACH,U_DPCH
2) For RlParam, specify the pointer to the variable for CphyRlseup() described under A.2 VARIABLE USED IN EACH FUNCTION.
3) For TrchParam, specify the pointer to the variable for CphyTrchConfig() described under A.2 VARIABLE USED IN EACH FUNCTION.
4) timeout
   Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
   Specifying NO_TIMEOUT inhibits timeout.

Remarks	RateMatching Parameter in Compressed Mode can not be calculated with this function. In Compressed Mode, use CalcRMParameterCM().

<table>
<tr><td>Function name</td><td colspan="4">INT CalcRMParameterCM(INT Phch, CPHY_RL_SETUP_PAR* RlParam, CPHY_TRCH_CONFIG_PAR* PhyParam, CMAC_CONFIG_PAR* MacParam)</td></tr>
<tr><td>Feature summary</td><td colspan="4">Calculates Rate Matching Parameter in Normal Mode and Compressed Mode.</td></tr>
<tr><td></td><td>Type</td><td>Argument name</td><td>Description</td><td>I/O</td></tr>
<tr><td>Argument</td><td>INT
CPHY_RL_SETUP_PAR*
CPHY_TRCH_CONFIG_PAR*
CMAC_CONFIG_PAR*</td><td>Phch
RlParam
PhyParam
MacParam</td><td>Physical Channel
Radio Link parameter
Transport parameter(Rate Matching parameter)
For Higher Layer Scheduling</td><td>Input
Input/Output
Input/Output
Input/Output</td></tr>
<tr><td></td><td>Type</td><td>Function value</td><td colspan="2">Description</td></tr>
<tr><td>Function value</td><td>INT</td><td>0
-1
-10

-11

-12

-13

-14

-15</td><td colspan="2">Normal ending
Maximum SymbolRate is too small. (Uplink)
Method=Puncturing specified at the time of Flexible Position in Compressed Mode.
Method=Higher Layer Scheduling specified at the time of Fixed Position in Compressed Mode
Method=Puncturing specified in Uplink Channel of Compressed Mode
The value out of range specified in the parameter of Compressed Mode
Transmission gap can not be created correctly from Compressed Mode Parameter
TGCFN is INVALID when Method≠NONE</td></tr>
<tr><td colspan="2">Feature details</td><td colspan="3"></td></tr>
<tr><td colspan="5">1) You can specify the following values in Physical Channel:
DownLink : P_CCPCH,S_CCPCH,D_DPCH
UpLink : U_PRACH,U_DPCH
2) For RlParam, specify the pointer to the variable for CphyRlseup() described in A.2 VARIABLE USED IN EACH FUNCTION.
3) For TrchParam, specify the pointer to the variable for CphyTrchConig() described in A.2 VARIABLE USED IN EACH FUNCTION.
4) For MacParam, specify the pointer for the variable for CmacConfig() described in A.2 VARIABLE USED IN EACH FUNCTION.
5) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally.
Specifying NO_TIMEOUT inhibits timeout.</td></tr>
<tr><td>Remarks</td><td colspan="4">Rate Matching Parameter in either Normal Mode and Compressed Mode can be calculated with this function. When you will describe scenario in the future, we recommend that you use this CalcRMParamterCM().</td></tr>
</table>

Function name	INT CalcRMParameterMtgp(INT Phch, CPHY_RL_SETUP_MTGP_PAR* RlParam, CPHY_TRCH_CONFIG_PAR* PhyParam, CMAC_CONFIG_PAR* MacParam)			
Function summary	Calculates Rate Matching Parameter when using structure CPHY_RL_SETUP_MTGP_PAR.			
	Type	ARgument name	Descriptions	I/O
Argument	INT CPHY_RL_SETUP_MTGP_PAR* CPHY_TRCH_CONFIG_PAR* CMAC_CONFIG_PAR*	Phch RlParam PhyParam MacParam	Physical Channel Radio Link parameter Transport parameter(Rate Matching parameter) For Higher Layer Scheduling	Input Input/Output Input/Output Input/Output
	Type	Function value	Description	
Function value	INT	0 -1 -10 -11 -12 -13 -14 -15	Normal ending Maximum SymbolRate is too small. (Uplink) Method=Puncturing specified at the time of Flexible Position in Compressed Mode. Method=Higher Layer Scheduling specified at the time of Fixed Position in Compressed Mode Method=Puncturing specified in Uplink Channel of Compressed Mode The value out of range specified in the parameter of Compressed Mode Transmission gap can not be created correctly from Compressed Mode paramter TGCFN is INVALID when Method≠NONE	
Feature details				
1) You can specify the following values in Physical Channel: Downlink : D_DPCH Uplink : U_DPCH 2) For RlParam, specify the pointer to the variable for CphyRlseup() described in A.2 VARIABLE USED IN EACH FUNCTION. 3) For TrchParam, specify the pointer to the variable for CphyTrchConig() described in A.2 VARIABLE USED IN EACH FUNCTION. 4) For MacParam, specify the pointer for the variable for CmacConfig() described in A.2 VARIABLE USED IN EACH FUNCTION.				
Remarks	This can calculate rate matching parameter for both of single TGPS and multiple TGPS, if structure CPHY_RL_SEUTP_MTGP_PAR.			

Function name	INT CteConfig(INT Loch, INT LochNo, CTE_CONFIG_PAR *CteParam,INT Timeout)			
Feature summary	Associates services with Logical Channels.			
	Type	Argument name	Description	I/O
Argument	INT INT CTE_CONFIG_PAR* INT	Loch LochNo CteParam Timeout	Logical Channel Logical Channel No TE Type and Parameter Timeout value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending (Timeout)	
Feature details				

1) You can specify the following values in Logical Channel:
   CCCH, DTCH, DCCH
2) You can specify 0-7 per Logical Channel in Logical Channel No.
3) Set TE Type and Parameter as structure variables. Make setting as follows:
   TeType: Set TE Type. You can specify the following values:

```
TE_TYPE_NONE            /* NO TE      */
TE_TYPE_VOICE_AMR_A /* AMR Class A */
TE_TYPE_VOICE_AMR_B /* AMR Class B */
TE_TYPE_VOICE_AMR_C /* AMR Class C */
TE_TYPE_ISDN_UDI        /* ISDN UDI     */
TE_TYPE_ISDN_Q931UDI /* ISDN UDI(Q931 DirectMode) */
TE_TYPE_ISDN_AV64       /* ISDN AV 64K */
TE_TYPE_ISDN_AV32       /* ISDN AV 32K */
TE_TYPE_IPPACKET        /* Packet Service   */
TE_TYPE_PPP             /* Packet Switched PPP */
TE_TYPE_PPPSERVER    /* PPP Server(Ether)   */
TE_TYPE_CSD             /* CSD   */
```

   Rate: Set a voice encoding rate in case TE Type is AMR Voice. You can set the following values.

```
VOICE_RATE_4_75   /* 4.75 kbps */
VOICE_RATE_5_15   /* 5.15 kbps */
VOICE_RATE_5_90   /* 5.90 kbps */
VOICE_RATE_6_70   /* 6.70 kbps */
VOICE_RATE_7_40   /* 7.40 kbps */
VOICE_RATE_7_95   /* 7.95 kbps */
VOICE_RATE_10_2   /* 10.2 kbps */
VOICE_RATE_12_2   /* 12.2 kbps */
```

   Set 0 when TeType is TE_TYPE_CSD.

   TTI: Set data transmission interval I in units of 10 ms. Specify 1-4 to represent 10 msec-40 msec.
   NumOfTB Set the number of Transport Blocks to be transmitted to TTI unit. You can specify 1-32.
   TBS Sets Transport Block Size in bits.
   Frame Sets a primitive to be transmitted to RLC. Specify RLC_TR_DATA_REQ or RCL_UM_DATA_REQ,RLC_AM_DATA_REQ in accordance with the RLC operation mode.
   Layer Sets the transmit destination layer for downlink data. Fix it to RLC.
   Data[1024] Sets a pattern used to transmit fixed data from TE.
   CsdConfPar The configuration is necessary in case of CSD. Set Initial Value in "Initial values of common values" when using the function of W-CDMA CSD.
4) timeout
   Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT CteRelease(INT Loch, INT LochNo,INT Timeout)			
Feature summary	Releases associations between TE Type and Logical Channel/Logical Channel No.			
	Type	Argument name	Description	I/O
Argument	INT INT INT	Loch LochNo Timeout	Logical Channel Logical Channel No Timeout value	Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending (Timeout)	
Feature details				
1) You can specify the following values in Logical Channel: CCCH, DTCH, DCCH 2) You can specify 0-7 per Logical Channel in Logical Channel No. 3) timeout Specify the wait time in msec and in 100 msec steps.　If verification cannot be made within this period, the processing ends abnormally.　Specifying NO_TIMEOUT inhibits timeout.				
Remarks				

Function name	INT CteConnect(INT Loch, INT LochNo, INT DownPort, INT UpPort,INT Direction, UCHAR *Opt, INT Timeout)			
Feature summary	Starts data transmission/reception for the assigned TE Type.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT INT UCHAR * INT	Loch LochNo DownPort UpPort Direction Opt Timeout	Logical Channel Logical Channel No Downlink port Uplink port Origination source Option Timeout value	Input Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending (Timeout)	

Feature details

1) You can specify the following values in Logical Channel:
CCCH, DTCH, DCCH
2) You can specify 0-7 per Logical Channel in Logical Channel No.
3) Parameters that can be specified as downlink ports are as follows:
TE_PORT_NORMAL: Transmits data from TE via downlink.
TE_PORT_FIXDATA: Transmits the fixed pattern specified in CteConfig() via downlink.
TE_PORT_LOOPBACK: Loops back uplink data to transmit it via downlink (Make same setting for Uplink port.).
TE_PORT_PN9: Transmits PN9 stage via downlink.
TE_PORT_USERDATA: Transmits the pattern input from the Userdata connector of the Voice Codec via downlink.
TE_PORT_AUDIO: Transmits the signal input from the Audio connector of the Voice Codec. as AMR data via downlink.
TE_PORT_MSTOMS: Used for the MS-to-MS test. Transmits data received from another D8480A/B Signalling Tester via downlink.

4) Parameters that can be specified as Uplink Ports are as follows:
TE_PORT_NORMAL: Transmits uplink data to TE.
TE_PORT_LOOPBACK: Loops back uplink data to transmit it via downlink (Make same setting for Downlink port.).
TE_PORT_USERDATA: Outputs uplink data from the Userdata connector of the Voice Codec.
TE_PORT_AUDIO: Outputs uplink AMR data from the Audio connector.
TE_PORT_MSTOMS: Used for the MS-to-MS test. Transmits uplink data to another MD8480A/B.

5) Parameters that can be specified in Direction are as follows:
CALL_FROM_AIR: Origination from the Air side
CALL_FROM_TE: Origination from the TE side

6) Opt is an option used for execution of CteConnect(). Specify a called number when originating a call from the Air side of ISDN by using CteConnect(). In this case use a character string such "1234567890". Other methods are undetermined for the time being.

7) timeout
Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks: TE on MD8480 starts sending downlink data at the CFN timing specified by TeActTime, and the transmission is repeated the number of times specified in TeTtiCounter. Refer to 3.7.14 Chapter about Specifying feature for a start timing of TE transmission/the number of transmission.

Function name	INT CteDisconnect(INT Loch, INT LochNo, INT Direction, INT Timeout)			
Feature summary	Halts data transmission/reception for the assigned TE Type.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT	Loch LochNo Direction Timeout	Logical channel Logical Channel No Disconnect source Timeout value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	>0 <0	Normal ending Abnormal ending (Timeout)	
Feature details				
1) You can specify the following values in Logical Channel: CCCH, DTCH, DCCH 2) You can specify 0-7 per Logical Channel in Logical Channel No. 3) Parameters that can be specified in Direction are as follows: CALL_FROM_AIR: Disconnection from the Air side CALL_FROM_TE: Disconnection from the TE side 4) timeout Specify the wait time in msec and in 100 msec steps. If verification cannot be made within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.				
Remarks				

Function name	VOID ReplaceIE(CHAR *Message ,CHAR *Data, INT StartBit , INT BitLen)			
Feature summary	Overwrites data in bits.			
	Type	Argument name	Description	I/O
Argument	CHAR * CHAR* INT INT	Message Data StartBit BitLen	Source data Overwritten data Starting location (in bits) Length of data overwriting original data (in bits)	Input/Output Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	
Feature details				
This function enables it to embed information element in Layer3 in hexadecimal format. Example of usage When CHAR Message []={ 0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x 07, 0x08, 0x09, 0x0A, 0x0B}; CHAR Data []={ 0xFF,0xFF,0xFF }; Executing ReplaceIE (Message,Data,53,21) makes Message {0x00, 0x01,0x02, 0x03, 0x04, 0x05, 0x07, 0xFF, 0xFF, 0xC9, 0x0A,0x0B}.				
Remarks				

Function name	VOID ExtractIE (CHAR *Message, CHAR *Data, INT StartBit , INT BitLen);			
Feature summary	Fetches data in bits.			
	Type	Argument name	Description	I/O
Argument	CHAR *	Message	Source data	Input/Output
	CHAR*	Data	Fetched data	Input
	INT	StartBit	Starting location (in bits)	Input
	INT	BitLen	Length of overwriting data (in bits)	Input
	Type	Function value	Description	
Function value	INT	=0	Normal ending	
		≠0	Abnormal ending	
Feature details				

This function enables it to fetch information element from Layer3 in hexadecimal format.
Example of usage
When
CHAR Message []={ 0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x 07, 0xFF, 0xFF, 0xC9, 0x0A, 0x0B},
Executing
ReplaceIE (Message,Data,53,21);
makes
Data {0xFF, 0xFF,0xF8• • • };.

Remarks	

Function name	VOID Int2MsbIE (UINT Value,CHAR *Data , INT BitLen);			
Feature summary	Transforms INT type(32 bit) data to MSB pack.			
	Type	Argument name	Description	I/O
Argument	UNT CHAR INT	Value Data BitLen	UINT type(32bit) data Data after transformation Valid bit length of data (1 through 32)	Input Output Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	
Feature details				
Example of usage When a value of variable Value in UINT type is 0x123456 and the value is represented in 25 bit, executing UNIT Value=0x123456; Int2MsbIE(Value, Data, 25); Makes Data {0x09, 0x1A, 0x2B, 0x00,• • • }.				
Remarks				

Function name	VOID Short2MsbIE(USHORT Value,CHAR *Data , INT BitLen);			
Feature summary	Transforms SHORT type(16 bit) data to MSB pack.			
	Type	Argument name	Description	I/O
Argument	USHORT CHAR* INT	Value Data BitLen	USHORT type(16 bit) data Data after transformation Valid Bit length of data (1 through 16)	Input Output Input
	Type	Function value	Description	
Function value	INT	=0 ≠0	Normal ending Abnormal ending	
Feature details				
Example of usage When a value of variables Value in USHORT type is 0x123 and the value is represented in 11 bit, executing UINT value=0x123; Short2MsbIE(Value, Data, 11); Makes Data {0x24, 0x60,• • • }.				
Remarks				

Function name	INT Msb2IntIE(CHAR *Data ,INT BitLen);		
Feature summary	Transforms MSB packed data to INT type (32bit).		

	Type	Argument name	Description	I/O
Argument	CHAR INT	Data BitLen	Source Data Valid bit length of data (1 through 32)	Input Input

	Type	Function value	Description
Function value	UINT		UINT type (32bit) data

Feature details

Example of usage
When data Data={ 0x09,0x1A,0x2B,0x00,• • • } is represented in 25 bit,
executing
CHAR Data={0x09,0x1A,0x2B,0x00};
Value=Msb2IntIE(Data, 25);
makes
Value=0x123456;.

Remarks

Function name	USHORT Msb2ShortIE(CHAR * Data,INT BitLen);
Feature summary	Transforms MSB packed data to USHORT type (16bit).

	Type	Argument name	Description	I/O
Argument	CHAR INT	Data BitLen	Source Data Valid bit length of data (1 through 16)	Input Input

	Type	Function value	Description
Function value	USHORT		USHORT (32bit) data

Feature details

Example of usage
When data Data={ 0x24, 0x60,• • • } is represented in 11 bit,
executing
CHAR Data={0x24, 0x60};
Value=Msb2ShortIE(Data, 11);
makes
Value=0x123;.

Remarks

Function name	INT BtsOcnsActivate (INT BtsNo, CPHY_OCNS_ACT_PAR *OcnsPar, INT Timeout)			
Feature summary	Starts the OCNS transmission			
	Type	Argument name	Explanations	I/O
Argument	INT CPHY_OCNS_PAR* INT	BtsNo OcnsPar Timeout	BTS No OCNS Parameter Timeout value	Input Input Input
	Type	Function value	explanations	
Function value	INT	= 0 ≠ 0	Normal ending Abnormal ending	
Feature details				

In case of tests on 2BTS configuration(using handover function or Tx Diversity function), [2nd OCNS(MU848061B)] is needed.

1) You can specify UNIT_BTS1 or UNIT_BTS2 in BtsNo.

2) OcnsPar

Specify OCNS to output with the following structures.

```
Typedef struct {
    USHORT   Power;              /* OCNS Power               */
    USHORT   BtsOffset;          /* BtsOffset                */
    ULONG    ScrCode;            /* Scrambling Code          */
    USHORT   DpchOffset;         /* DPCH Offset              */
    USHORT   DpchSlotFormat;     /* DPCH   Slot Format       */
    USHORT   DpchDpcchPower;     /* DPCCH Power              */
    USHORT   DpchPowerStep;      /* DPCH   Power Control Step */
    USHORT   MaxDLPower;         /* Maximun DL Power         */
    USHORT   MinDLPower;         /* Minimum DL Power         */
    USHORT   ChCode[16];         /* OCNS Pattern             */
} CPHY_OCNS_ACT_PAR;
```

Power: You can specify output power for OCNS in one dBs.
Configurable values are 0(0dBm)~-99(-99dBm).

BtsOffset: You can specify Offset for P_CPICH of OCNS.
Specify the same value as the one of BTS offset of the Parameter Setup window.

ScrCode: You can specify Scrambling Code of OCNS.
Specify the same value as the one in downlink DPCH.

DpchOffset: You can specify Offset to P_CPICH of OCNS.
Specify the same value as the one of Offset of downlink DPCH.

DpchSlotFormat: You can designate the location to conduct Power Control of OCNS by uplink TPC.
Specify the same value as the one of Slotformat in downlink DPCH.

DpchDpcchPower: You can specify DPCH(DPCCH)Power.
Specify the same value as the one of the downlink DPCH.

DpchPowerStep: You can specify the step of Power Control of OCNS by uplink TPC.
You can specify the following values.
POWER_AUTO_TPC_05: 0.5dB Step
POWER_AUTO_TPC_1: 1dB Step
POWER_AUTO_TPC_15: 1.5dB Step
POWER_AUTO_TPC_2: 2dB Step
To run the Power Control of uplink TPC Bit by function BtsOcnsPower(), step specified here will become the Power Control step for DPCH.

MaxDLPower: You can specify the value of upper limit for OCNS Power.
Specify the same value as the one of upper limit of the downlink DPCH(DPCCH)Power.

MinDLPower: You can specify the value of lower limit for OCNS Power.
Specify the same value as the one of lower limit of the downlink DPCH(DPCCH)Power.

ChCode[16]: You can specify PN9 default values for each 16 channel.
Specify them within the range of 0 through 511.

3) Timeout

Specify the wait time in msecs and in 100msec steps. If the confirmation cannot be made within this period of time, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT BtsOcnsDeactivate (INT BtsNo , INT Timeout)			
Feature summary	Halts the CNS transmission.			
	Type	Argument name	Explanation	I/O
argument	INT INT	BtsNo Timeout	BTS No Timeout value	Input Input
	Type	Function value	Explanation	
Function value	INT	= 0 ≠ 0	Normal ending Abnormal ending	
Feature details				

1) Specify UNIT_BTS1 or UNIT_BTS2 for BtsNo.
2) Timeout
You can specify wait time in msec and in 100msec steps. If the confirmation cannot be made within this period of time, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT BtsOcnsPower (INT BtsNo, INT Power, INT Timeout)			
Feature summary	Starts/ halts OCNS Power Control.			
	Type	Argument name	Explanation	I/O
argument	INT INT INT	BtsNo Power Timeout	BTS No OCNS power Timeout value	Input Input Input
	Type	Function name	Explanation	
Function value	INT	= 0 ≠ 0	Normal ending Abnormal ending	
Feature details				

1) Specify UNIT_BTS1 or UNIT_BTS2 for BtsNo.
2) power
You can specify the amount of change of OCNS power.
POWER_AUTO_TPC:starts Power Control by uplink TPC Bit. For this power control, what is specified in activating OCNS by function BtsOcnsActivate() is used for DPCH step width.
0: halts Power Control by uplink TPC Bit.
-99 to -1 or +1 to +99: changes DPCH channel power as much the same as specified here. The power of OCNS is controlled to maintain the Total Power fixed. However, to conduct this operation, the step width of DPCH must be specified as 1.0dB in activating OCNS by function BtsOcnsActivate().
3) Timeout
You can specify wait time in msec and in 100msec steps. If the confirmation cannot be made within this period of time, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function name	INT BtsOcnsPowerActTime (INT BtsNo, INT Power, INT ActTime, INT Timeout)			
Feature summary	Starts/ halts OCNS Power Control.			
	Type	Argument name	Explanation	I/O
argument	INT INT INT INT	BtsNo Power ActTime Timeout	BTS No OCNS power Activate Time Timeout value	Input Input Input Input
	Type	Function name	Explanation	
Function value	INT	= 0 ≠ 0	Normal ending Abnormal ending	

Feature details

1) Specify UNIT_BTS1 or UNIT_BTS2 for BtsNo.
2) power
You can specify the amount of change of OCNS power.
POWER_AUTO_TPC:starts Power Control by uplink TPC Bit. For this power control, what is specified in activating OCNS by function BtsOcnsActivate() is used for DPCH step width.
0: halts Power Control by uplink TPC Bit.
-99 to -1 or +1 to +99: changes DPCH channel power as much the same as specified here. The power of OCNS is controlled to maintain the Total Power fixed. However, to conduct this operation, the step width of DPCH must be specified as 1.0dB in activating OCNS by function BtsOcnsActivate().
3) Timeout
You can specify wait time in msec and in 100msec steps. If the confirmation cannot be made within this period of time, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.
4) Specify a CFN value in Actibation Time, that is a timing which a setting becomes effective.
The case that CFN is not set up: ACTIVATE_NOW (It is the same operation as BtsOcnsPower().)
The case that CFN is set up : 0 to 255 (If the value is over 255, an effective value is a remainder divided by 256.)
Note) You can read now CFN value with BtsReadCFN(). When you set up CFN value, you add to read CFN 20 or more than it, and set to it.(The added value changes at the processing speed of PC etc.)

Remarks

Function Name	INT CpdcpConfig(INT Loch, INT LochNo, CPDCP_CONFIG_PAR *CpdcpConfigPar, INT Timeout)			
Feature Summary	Setting for a specified PDCP entity			
	Type	Argument Name	Description	Input/Output
Argument	INT INT CPDCP_CONFIG_PAR * INT	Loch LochNo CpdcpConfigPar Timeout	Logical Channel Logical Channel No Stores the initializing information of PDCP layer Time-out value	Input Input Input Input
	Type	Function Value	Description	
Function Value	INT	= 0 ≠ 0	Normal End Abnormal End	
Feature Details				

1)Specify DTCH for Logical Channel.
2)Logical Channel No is available to specify 0 to 7 by each Logical Channel.
3)CPDCP_CONFIG_PAR is the following variable structure.
The following one can be specified. PDCP_SN_SYNC_DIS,PDCP_SN_SYNC_ENA,Please specify PDCP_SN_SYNC_ENA when you use the Lossless InterRAT Cell Change function.
Start_DL_Send_SN: The initial value of downlink transmission SN (0 to 65535 can be set.)
Start_UL_Receive_SN: The initial value of uplink reception SN (0 to 65535 can be set.)
Frame: The following are available to specify: RLC_AM_DATA_REQ, RLC_UM_DATA_REQ, and RLC_TR_DATA_REQ.
LowerLayer: Specify RLC to the lower layer.
UpperLayer: Specify TE to the upper layer.
4)Timeout: Waiting time is specified in unit of msec and 100msec step. Abnormal end occurs if confirmation cannot be made by this time. Time-out will not occur if the value is set to NO_TIMEOUT.

Remarks	If the following requirements are not filled, Anritsu will not accept liability for the performance. 1. On TE specified by CteConfig( ), TeType shall be any of TE_TYPE_IPPACKET and TE_TYPE_PPPSERVER. 2. On TE specified by CteConnect( ), both of UpLinkPort and DownLinkPort shall be TE_PORT_NORMAL.

Function Name	INT CpdcpRelease (INT Loch, INT LochNo, INT Timeout)			
Feature Summary	Releases for a specified PDCP entity			
	Type	Argument	Description	Input/Output
Argument	INT INT INT	Loch LochNo Timeout	Logical Channel Logical Channel No Time-out value	Input Input Input
	Type	Function Value	Description	
Function Value	INT	= 0 ≠ 0	Normal end Abnormal end	
Feature Details				

1) Specify DTCH to Logical Channel.
2) Logical Channel No is available to specify from 0 to 7 by each Logical Channel.
3) Timeout: Waiting time is specified in unit of msec and 100msec step. Abnormal end occurs if confirmation cannot be made by this time. Time-out will not occur if the value is set to NO_TIMEOUT.

Remarks	If the following requirements are not filled. we will not accept liability for the performance. 1. On TE specified by CteConfig( ), TeType shall be any of TE_TYPE_IPPACKET and TE_TYPE_PPPSERVER. 2. On TE specified by CteConnect( ), both of UpLinkPort and DownLinkPort shall be TE_PORT_NORMAL.

Function Name	INT CpdcpReloc (INT Loch, INT LochNo, CPDCP_RELOC_PAR **CpdcpRelocPar, INT Timeout)			
Feature Summary	Requests to return the values of uplink reception sequence number and downlink transmission sequence number to a specified PDCP entity.			
	Type	Argument	Description	In-put/Output
Argument	INT INT CPDCP_RELOC_PAR ** INT	Loch LochNo CpdcpRelocPar Timeout	Logical Channel Logical Channel No Stores uplink reception SN and downlink transmission SN. Time-out value	Input Input Output Input
	Type	Function Value	Description	
Function Value	INT	= 0 ≠ 0	Normal end Abnormal end	

Feature Details

1) Specify DTCH to Logical Channel.
2) Logical Channel No is available to specify 0 to 7 by each Logical Channel.
3) CPDCP_RELOC__PAR is a structure that has the following variables.
UL_Receive_SN: Uplink reception sequence number
DL_Send_SN: Downlink transmission sequence number
4)Timeout: Waiting time is specified in unit of msec and 100msec step. Abnormal end occurs if confirmation cannot be made by this time. Time-out will not occur if the value is set to NO_TIMEOUT.

Remarks	IF the following requirements are not filled, we will not accept liability for the performance. 1. On TE specified by CteConfig( ), TeType shall be any of TE_TYPE_IPPACKET and TE_TYPE_PPPSERVER. 2. On TE specified by CteConnect( ), both of UpLinkPort and DownLinkPort shall be TE_PORT_NORMAL.

Function Name	INT CpdcpSeqNum (INT Loch, INT LochNo,CPDCP_SN_PAR *CpdcpSnPar, INT Timeout)			
Feature Summary	Informs the uplink transmission sequence number and downlink reception sequence number of the mobile station side to a specified PDCP entity.			
	Type	Argument Name	Description	Input/Output
Argument	INT INT CPDCP_SN_PAR * INT	Loch LochNo CpdcpSnPar Timeout	Logical Channel Logical Channel No Stores uplink transmission SN and downlink reception SN. Time-out value	Input Input Input Input
	Type	Function Value	Description	
Function Value	INT	= 0 ≠ 0	Normal end Abnormal end	

Feature Details

1) Specify DTCH to Logical Channel.
2) Logical Channel No is available to specify from 0 to 7 by each Logical Channel.
3) CPDCP_SN_PAR is a structure that has the following variables. Sequence numbers received from a mobile station by TTC Message are stored in the following variables.
UL_Send_SN: Uplink transmission sequence number
DL_Receive_SN: Downlink reception sequence number
4) Timeout: Waiting time is specified in unit of msec and 100msec step. Abnormal end occurs if confirmation cannot be made by this time. Time-out will not occur if the value is set to NO_TIMEOUT.

Remarks	TE shall be used on prerequisite conditions to fill in the following requirements. If the function is used without filling these requirements, we will not accept liability for the performance. 1. On TE specified by CteConfig( ), TeType shall be any of TE_TYPE_IPPACKET and TE_TYPE_PPPSERVER. 2. On TE specified by CteConnect( ), both of UpLinkPort and DownLinkPort shall be TE_PORT_NORMAL.

Function name	INT WcdmaRcvControl(INT *BtsNo, INT *Frame, INT *Loch, INT *LochNo,CHAR *Message, INT Timeout)			
Feature summary	Receive control primitives in W-CDMA			
	Type	Argument name	Description	I/O
Argument	INT *	BtsNo	BTS No	Output
	INT *	Frame	Received primitive	Output
	INT *	Loch	Logical Channel	Output
	INT *	LochNo	Logical Channel Number	Output
	CHAR *	Message	Received data	Output
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	>= 0	Normal ending (Received data length)	
		< 0	Abnormal ending (Timeout)	

Feature details

1) The following is returned in BtsNo.
 0 : when 2) Frame is CPHY_CFN_IND
 BtsNo(UNIT_BTS1, UNIT_BTS2 or UNIT_BTS3) which specified in CphySfnInd : when 2) frame is CPHY_SFN_IND
2) The following is returned in Frame.
 CPHY_CFN_IND : notifying a CFN timing specified by CphyCfnInd( )
 CPHY_SFN_IND : notifying a SFN timing specified by CphySfnInd( )
3) 0 is returned in Loch.
4) 0 is returned in LochNo.
5) Null pointer is stored in Message.
6)Timeout
Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.

Remarks	

Function Name	INT CpdcpFwd(INT Loch, INT LochNo, INT Timeout)			
Feature Summary	The data preserved in the specified PDCP entity is transmitted to SNDCP.			
	Type	Argument Name	Description	Input/Output
Argument	INT	Loch	Logical Channel	Input
	INT	LochNo	Logical Channel No	Input
	INT	Timeout	Time-out value	Input
	Type	Function Value	Description	
Function Value	INT	= 0	Normal End	
		≠ 0	Abnormal End	

Feature Details

1) Specify DTCH for Logical Channel.
2) Logical Channel No is available to specify 0 to 7 by each Logical Channel.
3) Timeout: Waiting time is specified in unit of msec and 100msec step. Abnormal end occurs if confirmation cannot be made by this time. Time-out will not occur if the value is set to NO_TIMEOUT.

Remarks	If the following requirements are not filled, Anritsu will not accept liability for the performance. 1. On TE specified by CteConfig( ), TeType shall be any of TE_TYPE_IPPACKET and TE_TYPE_PPPSERVER. 2. On TE specified by CteConnect( ), both of UpLinkPort and DownLinkPort shall be TE_PORT_NORMAL.

Function name	INT PdcpRcvControl(INT *Loch, INT *LochNo, INT Timeout)			
Feature summary	Receive control primitives in PDCP.			
	Type	Argument name	Description	I/O
Argument	INT *	Loch	Logical Channel	Output
	INT *	LochNo	Logical Channel Number	Output
	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	>= 0	Normal ending (Received data length)	
		< 0	Abnormal ending (Timeout)	
Feature details				
1) DTCH is returned in Loch. 2) From 0 to 7 are returned in LochNo. 3)Timeout Specify the wait time in msec and in 100 msec steps. If the confirmation is not done within this period, the processing ends abnormally. Specifying NO_TIMEOUT inhibits timeout.				
Remarks				

Function name	INT CteSuspend (INT Loch, INT LochNo, INT Timeout)			
Feature summary	Stops data transmission and reception for the assigned TE Type temporarily and received data is preserved.			
	Type	Argument name	Description	I/O
Argument	INT	Loch	Logical Channel	Input
	INT	LochNo	Logical Channel No	Input
	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0	Normal termination	
		≠ 0	Abnormal termination (Timeout)	
Feature details				
1)Specify DTCH for Logical Channel. 2)Logical Channel No is available to specify 0 to 7 by each Logical Channel. 3) Timeout Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.				
Remarks:	If the following requirements are not filled, Anritsu will not accept liability for the performance. - On TE specified by CteConfig(), TeType shall be any of TE_TYPE_IPPACKET,TE_TYPE_PPPSERVER, and TE_TYPE_PPP.			

Function name	INT CteResume (INT Loch, INT LochNo, INT Timeout)			
Feature summary	Restarts data transmission and reception for the assigned TE Type.			
	Type	Argument name	Description	I/O
Argument	INT INT INT	Loch LochNo Timeout	Logical Channel Logical Channel No Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				
1) Specify DTCH for Logical Channel. 2) Logical Channel No is available to specify 0 to 7 by each Logical Channel. 3) Timeout Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.				
Remarks:	If the following requirements are not filled, Anritsu will not accept liability for the performance. - On TE specified by CteConfig(), TeType shall be any of TE_TYPE_IPPACKET,TE_TYPE_PPPSERVER, and TE_TYPE_PPP.			

Function name	INT CbmcConfig(INT BtsNo,INT Loch,INT LochNo,CBMC_CONFIG_PAR *BMCParam,INT Timeout)			
Feature summary	Set the parameters of BMC layer.			
	Type	Argument name	Description	I/O
Argument	INT INT INT BMC_CONFIG_PAR * INT	BtsNo Loch LochNo BMCParam Timeout	Bts No Logical Channel Logical Channel Number The parameter of BMC layer Time-out value	Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Specify UNIT_BTS1, UNIT_BTS2 and UNIT_BTS3 for BtsNo.
2) Specify the following values in Logical Channel.
   CTCH
3) Specify "0" in Logical Channel Number.
4) For BMCParam, specify the pointer to the variable for CbmcConfig() described under A.2 VARIABLE USED IN EACH FUNCTION.
5) Timeout
   Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks: The display of layer in Trace screen is as follows.
- CBMC_CONFIG_REQ: RRC -> TE
- CBMC_CONFIG_CNF: TE -> RRC

Function name	INT CbmcRelease(INT BtsNo, INT Loch, INT LochNo, INT Timeout)			
Feature summary	Release BMC layer and initialize the parameters of BMC layer.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT	BtsNo Loch LochNo Timeout	Bts No Logical Channel Logical Channel Number Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Specify UNIT_BTS1, UNIT_BTS2 and UNIT_BTS3 for BtsNo.
2) Specify the following values in Logical Channel.
   CTCH
3) Specify "0" in Logical Channel Number.
4) Timeout
   Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks: The display of layer in Trace screen is as follows.
- CBMC_RELEASE_REQ: RRC -> TE
- CBMC_RELEASE_CNF: TE -> RRC

Function name	INT SndBMCMessage(INT BtsNo, INT Loch, INT LochNo, BMC_DATA_PAR *BMCmessage, INT Start_period)			
Feature summary	Transmit BMC messages.			
	Type	Argument name	Description	I/O
Argument	INT INT INT BMC_DATA_PAR * INT	BtsNo Loch LochNo BMCmessage Start_period	Bts No Logical Channel Logical Channel Number The setting CBS message and Scheduling message Start_period	Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0	Normal termination	
Feature details				

1) Specify UNIT_BTS1, UNIT_BTS2 and UNIT_BTS3 for BtsNo.
2) Specify the following values in Logical Channel.
   CTCH
3) Specify "0" in Logical Channel Number.
4) For BMCmessage specify the pointer to the variable for SndBMCmessage() described under A.2 VARIABLE USED IN EACH FUNCTION.
5) Start_period
   Specify the beginning timing to transmit BMC message (the number of Repetition_CtchBS).

Remarks:	Note1: See section 3.7.24 about the explanation of Repetition_CtchBS and Start_period. Note2: The maximum number of SndBMCMessage() to be able to accumulate before Start_period is ten for each BTS. Note3: BMC message can be set for each BTS. BMC message can be transmitted by only one BTS activated S_CCPCH for CBS using CmacConfig() function. Note4: If BMC_Level2_Mode=CBS_MANUAL is set and CbmcConfig() function is executed, one CBS message can be set to one SndBMCMessage() function. However, BMC layer can be accumulated ten CBS messages for each BTS. If user tries to accumulate CBS message above ten, the message is not accumulated. Note5: If BMC_Level2_Mode=CBS_MANUAL is set and CbmcConfig() function is executed, one Scheduling message can be set to one SndBMCMessage() function. However, BMC layer can be accumulated two Scheduling messages for each BTS.

A.2 VARIABLE USED IN EACH FUNCTION

A.2.1 Basic types

Basic types of variables are shown in Table A-2.

Table A-2 Basic types of variables

Type	Definition
INT	signed 32bit
UINT	Unsigned 32bit
SHORT	signed 16bit
USHORT	Unsigned 16bit
CHAR	signed 8bit
UCHAR	Unsigned 8bit

A.2.2 Structure

A.2.2.1 Structure for CrlcConfig (CRLC_CONFIG_PAR structure)

A.2.2.1.1 Definition of CRLC_CONFIG_PAR structure

Parameters used for each RLC as mentioned below are based on 3GPP TS25.322.

/* CRLC_CONFIG_REQ */

```
typedef struct {
    USHORT  NumOfPUs;              /* Number of PUs         */
    USHORT  PU_LengthTM;           /* RLC PU Length(TM)*/
    USHORT  PU_LengthUM;           /* RLC PU Length(UM)*/
    USHORT  PU_LengthAM;           /* RLC PU Length(AM)*/
    USHORT  TxWindowSize;          /* AM Tx Window Size */
    USHORT  RxWindowSize;          /* AM Rx Window Size */
    USHORT  PollWindow;            /* AM Poll Window        */
    USHORT  PollPU;                /* AM Poll PU            */
    USHORT  PollSDU;               /* AM Poll ADU           */
    USHORT  SDUSize;               /* SDU Size              */
    USHORT  MaxDat;                /* MAX Dat               */
    USHORT  MaxRst;                /* MAX Rst               */
    USHORT  MaxMRW;                /* MAX MRW               */
    USHORT  TimePoll;              /* Timer Poll            */
    USHORT  TimePollProhibit;      /* Timer Poll Prohibit   */
    USHORT  TimerEPC;              /* Timer EPC             */
    USHORT  TimerDiscard;          /* Timer Discard         */
    USHORT  TimerPollPeriodic;     /* Timer Poll Periodic   */
    USHORT  TimerStatusProhibit;   /* Timer Status Prohibit */
    USHORT  TimerStatusPeriodic;   /* Timer Staus Periodic  */
    USHORT  TimerRst;              /* Timer Rst             */
    USHORT  TimerMRW;              /* Timer MRW             */
    USHORT  PollingTriggers;       /* Polling Trigger       */
    USHORT  StatusTriggers;        /* Status Trigger        */
    USHORT  SDUDiscardMode;        /* SDU Discard Mode      */
    USHORT  UseAsymPUSize;         /* PDU_Length Asymmetry Mode */
    USHORT  PU_LengthDL;           /* Down link PU Length */
    USHORT  PU_LengthUL;           /* Up link PU Length */
    USHORT  Reserve[4];            /* for future use */
} CRLC_CONFIG_PAR;
```

A.2.2.1.2　Description of CRLC_CONFIG_PAR structure

Member	Description
NumberOfPUs	Specify the number of PUs contained in a single PDU.　You cannot set a value other than 1. Value on scenario　Specification 1 -------------------- 1PDU = 1PU (Fixed)
PU_LengthTM	Only when you use TM Segmentation function or a certain function in MX848041A, you can set to this member. Refer to the section of each function for the detail. This member is not used in the UM/AM mode. Only this value is referenced in the TM mode.
PU_LengthUM	Length of one PU in the UM mode (in bits). You can specify a multiple value of 8 among 8 through 320. The value plus 8 bits equals the length of PDU. This member is not used in the TM/AM mode.　Only this value is referenced in the UM mode.
PU_LengthAM	Length of one PU in the AM mode (in bits). You can specify a multiple value of 8 among 8 through 624. The value plus 16 bits equals the length of PDU. This is not used in TM/UM mode.
TxWindowSize	Specify the size of WindowSize for DownLink.　You can specify 1-2048.
RxWindowSize	Specify the size of WindowSize for UpLink.　You can specify 1-2048.
PollWindow	Used when Poll_Window% of transmission window that belongs to Polling-Triggers (mentioned later) is enabled.　Polling is made in case the rate of current Window to TxWindowSize(mentioned earlier) has dropped below (100-Pollwindow)%. Value on scenario　Specification 1..100 (natural number) -------- 1% to 100%
PollPU	Specify the maximum value (natural number) of VT (PU).
PollSDU	Specify the maximum value (natural number) of VT (SDU).
SDUSize	Currently not in use.
MaxDat	Specify the maximum value (natural number) of VT (DAT).
MaxRst	Specify the maximum value (natural number) of VT (RST).
MaxMRW	Specify the maximum value (natural number) of VT(MRW). Currently not in use.
TimerPoll **TimerPollProhibit** **TimerEPC:(Unsupported)** **TimerDiscard:(Unsupported)** **TimerPollPeriodic** **TimerStatusProhibit** **TimerStatusPeriodic** **TimerRst** **TimerMRW(Unsupported)**	Value on scenario　Meaning 1-65535　10 ms-655350 ms (in 10 ms steps) 0　Timer not used ∗You must not specify nothing to the unsupported members.
PollingTriggers	Set the Trigger for sending Poll.　Set a value in bits as follows.　Setting 1 enables the trigger and 0 disables the trigger.　When you set up Poll Timer and Timer Based effectively, please do not set the value (a TimerPoll variable, TimerPollPeriodic variable) of corresponding Timer as 0. Bit15(0x8000) Last PU in buffer Bit14(0x4000) Last PU in retransmission buffer Bit13(0x2000) Poll timer Bit12(0x1000) Every Poll_PU PU Bit11(0x0800) Every Poll_SDU SDU Bit10(0x0400) Poll_Window% of transmissions window Bit9(0x0200)　Timer based

	(Setting example) CrlcConfigDCCH.PollingTriggers = 0x0C00; CrlcCon- fig(UNIT_BTS1,CRLC_AM_ESTABLISH,DCCH,0,&CrlcConfigDC CH,RRC,NO_TIMEOUT); The above setting enables Every Poll_SDU SDU and Poll_Window% of transmissions window.
StatusTriggers	Set the Trigger for sending STATUS.　Set a value in bits as follows.　Setting 1 enables the trigger and 0 disables the trigger.　In confirming Timer based STATUS transfer, please do not set the value of above-mentioned Timer-StatusPeriodic as 0. Bit15(0x8000) Detection of missing PU(s) Bit14(0x4000) Timer based STATUS transfer
SDUDiscardMode	Currently As it is not used, please set nothing up.
UseAsymPUSize	To set payload size individually downlink and uplink for an AM RLC, set a UseAsymPUSize value to 1 and set PU_LengthDL and PU_LengthUL values for payload sizes of downlink and uplink. If the UseAsymPUSize value is equal to 0, neither PU_LengthDL nor the PU_LengthUL value will be referred and PU_LengthAM value will be used to downlink and uplink.
PU_LengthDL	Set a UseAsymPUSize value to 1 and set PU_LengthDL value for payload size of downlink.
PU_LengthUL	Set a UseAsymPUSize value to 1 and set PU_LengthUL value for payload size of uplink.

A.2.2.2 Structure for CmacConfig() (CMAC_CONFIG_PAR structure)

A.2.2.2.1 Definition of CMAC_CONFIG_PAR structure

```
typedef struct {
    USHORT      CTCH_allocation_Period;     /* N:Period of CTCH allocation      */
    USHORT      CBS_Frame_Offset;           /* K:CBS frame offset               */
} LEVEL1_INFO;
```

```
typedef struct {
    UCHAR           CBS_Flag;               /* 0:not operate CBS  1:operate CBS */
    UCHAR           Loch;                   /* logical channel for CBS          */
    UCHAR           LochNo;                 /* logical channel number for CBS   */
    UCHAR           reserve;
    LEVEL1_INFO     Level1Info;             /* CBS DRX Level1 information       */
} CBS_INFO;
```

```
typedef struct {
    UCHAR     NumOfTFC;                     /* Number of TFC                    */
    UCHAR     Reserve[3];                   /* Reserve for future use           */
    UCHAR     TFC[MAX_NO_TRANSPORT_FORMAT_COMBINATION][MAX_NO_TRANSPORT_CHANNELS];
                                            /* TFCI Table                       */
} TFCS_PAR;
```

```
typedef struct {
    UCHAR     TTI;                          /* Transmission Time Interval       */
    UCHAR     EPType;                       /* Error Protection Type            */
    UCHAR     CodingRate;                   /* Error Protection Coding Rate     */
    UCHAR     CRC_Size;                     /* CRC Size                         */
    USHORT    RM_Attr;                      /* Rate Matching Attribute          */
    USHORT    RedFunc;                      /* RedFunc                          */
} TF_STATIC_PART;
```

```
typedef struct {
    USHORT    NumOfTB;                      /* Number of Transport Blocks       */
    USHORT    TBSize;                       /* Transport Block Size             */
} TF_DYNAMIC_PART;
```

```
typedef struct {
    UCHAR              Trch;                /* Transport Channel                */
    UCHAR              TrchNo;              /* Transport Channel Number         */
    UCHAR              InterLv1st;          /* First interleave                 */
    UCHAR              NumOfDynamic;        /* Number of Dynamic part           */
    TF_STATIC_PART     Static;              /* Semi-Static part                 */
    TF_DYNAMIC_PART Dynamic[MAX_NO_TRANSPORT_FORMAT_SET];
                                            /* Dynamic Part                     */
} TRCH_INFO;
```

```
typedef struct {
    UCHAR     Loch;                         /* Logical Channel                  */
    UCHAR     LochNo;                       /* Logical Channel Number           */
    UCHAR     Priority;                     /* Priority                         */
    UCHAR     CTLength;                     /* CT Length                        */
    UCHAR     CTValue;                      /* CT Value                         */
    UCHAR     TCTFLength;                   /* TCTF Length                      */
    USHORT    TCTFValue;                    /* TCTF Value                       */
    UCHAR     UEIDType;                     /* User Equipment ID type           */
    UCHAR     Reserve[3];                   /* Reserved to fit alignment        */
    UINT      UEID;                         /* User Equipment ID                */
} LOCH_INFO;
```

```
/* CMAC_CONFIG_REQ */
typedef struct {
      UCHAR          ActFlag;                    /* Active/Deactive                  */
      UCHAR          Reserve1;                   /* Reserved for future use          */
      UCHAR          NumOfTrch;                  /* Number of Transport Channel      */
      UCHAR          Reserve2;                   /* Reserved for future use          */
      TFCS_PAR       TFCS;                       /* Transport Format Combination Set */
      TRCH_INFO      TrchInfo[MAX_NO_TRANSPORT_CHANNELS];
                                                 /* Transport Format Set             */
      UCHAR          NumOfLoch[MAX_NO_TRANSPORT_CHANNELS];
                                                 /* Number of Logical Channel        */
      UCHAR          Reserve3;                   /* Reserve for future use           */
      LOCH_INFO      LochInfo[MAX_NO_TRANSPORT_CHANNELS][MAX_NO_LOGICAL_CHANNELS];
                                                 /* Transport Channel Information    */
      MAC_SORTED_TFCS SortedTFCS[MAX_NO_TRANSPORT_FORMAT_COMBINATION];
                                                 /* Sorted Tfcs(Internal Use Only)   */
} CMAC_CONFIG_PAR;
```

A.2.2.2.2 Description of CMAC_CONFIG_PAR structure

<table>
<tr><th>Member</th><th colspan="3">Description</th></tr>
<tr><td>ActFlag</td><td colspan="3">Specify whether MAC is active.
Value on scenario Meaning
MAC_DEACTIVE ------ MAC DeActive
MAC_ACTIVE ------ MAC Active</td></tr>
<tr><td>NumOfTrch</td><td colspan="3">Specify the number of Transport Channels in CCTrCH. You can specify 1-8.</td></tr>
<tr><td>TFCS</td><td colspan="3">Specify TFCS (Transport Format Combination Set).</td></tr>
<tr><td rowspan="12">TrchInfo</td><td colspan="3">Specify parameters of each Transport Channel in CCTrCH. Description of each parameter of TrchInfo is as follows:</td></tr>
<tr><td>Trch</td><td colspan="2">Transport Channel Type. Set as follows.
DownLink : D_BCH,D_PCH, D_FACH,D_DCH
UpLink : U_RACH,U_DCH</td></tr>
<tr><td>TrchNo</td><td colspan="2">Channel number of Transport Channel. You can specify 0-7.
Be sure to specify the same value as a number of TrchInfo.
Example: specify “n” in case of TrchInfo[n].
Exception: the case where Trch is two or more kinds like FACH+PCH. If n= 0 considers as Trch = D_PCH and n= 1,2 consider as Trch = D_FACH, n= 1 will be set to TrchNo = 0, and n= 2 will be set to TrchNo = 1.That is, TrchNo is specified from 0 by each Trch.</td></tr>
<tr><td>InterLv1st</td><td colspan="2">Specify whether to perform 1st Interleaving. This setting is invalid.</td></tr>
<tr><td rowspan="8">Static</td><td colspan="2">Specify parameters of Semi-Static Part of TFS (Transport Format Set). Description of each parameter of Static is as follows:</td></tr>
<tr><td>TTI</td><td>Specify TTI.
Value on scenario Meaning
1 -------- TTI = 10ms
2 -------- TTI = 20ms
4 -------- TTI = 40ms
8 -------- TTI = 80ms(only when NumOfTrch is 1)</td></tr>
<tr><td>EPType</td><td>Specify Error Protection Type. This setting is invalid.</td></tr>
<tr><td>CodingRate</td><td>Specify Coding Rate. This setting is invalid.</td></tr>
<tr><td>CRC_Size</td><td>Specify the number of CRC bits. This setting is invalid.</td></tr>
<tr><td>RM_Attr</td><td>Specify RateMatching Attribute. This setting is invalid.</td></tr>
<tr><td>RedFunc</td><td>Enable PipeLine Delay reduction function.
Value on scenario Meaning
PLD1 Enable PLD1 function
PLD2 Enable PLD2 function
PLD3 Enable PLD3 function</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>NumOfDynamic</td><td colspan="3">Specify the number of parameters of TFS(Transport Format Set) Dynamic Part. You can specify 1~64.</td></tr>
<tr><td rowspan="3">Dynamic</td><td colspan="3">Specify the parameter of Dynamic Part of TFS(Transport Format Set)
Descriptions for each parameter of Dynamic are as follows:</td></tr>
<tr><td>NumOfTB</td><td colspan="2">Specify the number of Transport Block.</td></tr>
<tr><td>TBSize</td><td colspan="2">Specify the number of bit for Transport Block.</td></tr>
<tr><td>NumOfLoch</td><td colspan="3">Specify the number of Logical Channels in TrCH. If Trch is D_FACH or U_RACH, you can specify 1-8.
If Trch is not D_FACH nor U_RACH, you can specify 1-6.</td></tr>
<tr><td>LochInfo</td><td colspan="3">Specify parameters of each Logical Channel in TrCH. Description of each parameter of LochInfo is as follows:</td></tr>
</table>

	Loch	Logical Channel Type. DownLink : You can specify D_BCCH, D_PCCH, D_CCCH, D_DCCH, or D_DTCH. UpLink : You can specify U_CCCH, U_DCCH, or U_DTCH.	
	LochNo	Channel Number of Logical Channel. You can specify 0-7.	
	Priority	Specify the priority of Logical Channel. Currently invalid.	
	CTLength	Length of CT Field in MAC Header. You can specify 0 or 4. (Specifying 0 renders CT Field unavailable.)	
	CTValue	Value of CT Field in MAC Header. You can specify 0-15.	
	TCTFLength	Length of TCTF (Target Channel Type Field). You can specify 0, 2 or 8. (Specifying 0 renders TCTF Field unavailable.)	
	TCTFValue	Value of TCTF (Target Channel Type Field). Currently invalid (automatically assigned).	
	UEIDType	Value of UEID Type field. Value on scenario Meaning UEID_NOTUSE--------- UEID Field not in use UEID_U_RNTI ---------- UEID is U_RNTI(32bit) UEID_C_RNTI----------- UEID is C_RNTI(16bit)	
	UEID	Value of UDIE Field In case UEIDType is UEID_NOTUSE, this setting is meaningless. In case UEIDType is UEID_U_RNTI, you can specify 0-0xFFFFFFFF. In case UEIDType is UEID_C_RNTI, you can specify 0-0xFFFF.	
SortedTFCS	You can not specify it while creating scenario.		
CbsInfo	Specify the setting of CBS level1 scheduling.		
	CBS_Flag	Flag for whether CBS is used. Value on scenario Meaning 0 CBS is not used. 1 CBS is used.	
	Loch	Specify Logical channel(CTCH) used for CBS.	
	LochNo	Specify Logical channel number(0) used for above-mentioned Loch.	
	reserve	Reservation	
	Level1Info	The parameter for CBS level1 scheduling	
		CTCH_allocation_Period	N: Period of CTCH allocation
		CBS_Frame_Offset	K: CBS frame offset

A.2.2.3 Structure for CphyTrchConfig() (CPHY_TRCH_CONFIG_PAR structure)

A.2.2.3.1 Definition of CPHY_TRCH_CONFIG_PAR structure

```
typedef struct {
UCHAR      NumOfTFC;        /* Number of TFC              */
UCHAR      Reserve[3];      /* Reserve for future use     */
UCHAR
TFC[MAX_NO_TRANSPORT_FORMAT_COMBINATION][MAX_NO_TRANSPORT_CHANNE
LS];
/* TFCI Table                      */
} TFCS_PAR;
```

```
typedef struct {
UCHAR     TTI;              /* Transmission Time Interval     */
UCHAR     EPType;           /* Error Protection Type          */
UCHAR     CodingRate;       /* Error Protection Coding Rate */
UCHAR     CRC_Size;         /* CRC Size                       */
USHORT    RM_Attr;          /* Rate Matching Attribute        */
USHORT    RedFunc;          /* RedFunc                        */
} TF_STATIC_PART;
```

```
typedef struct {
USHORT    NumOfTB;          /* Number of Transport Blocks     */
USHORT    TBSize;           /* Transport Block Size           */
} TF_DYNAMIC_PART;
```

```
typedef struct {
UCHAR              Trch;           /* Transport Channel           */
UCHAR              TrchNo;         /* Transport Channel Number */
UCHAR              InterLv1st;     /* First interleave            */
UCHAR              NumOfDynamic;   /* Number of Dynamic part      */
TF_STATIC_PART     Static;         /* Semi-Static part            */
TF_DYNAMIC_PART Dynamic[MAX_NO_TRANSPORT_FORMAT_SET];
/* Dynamic Part                  */
} TRCH_INFO;
```

```
/* CPHY_TRCH_CONFIG_REQ */
typedef struct {
UCHAR          DTXPosition;       /* Fix/Flexible Position             */
UCHAR          InterLv2nd;        /* 2nd Interleave ON/OFF             */
UCHAR          PuncLimit;         /* Puncturing Limit(for Up Link)     */
UCHAR          NumOfTrch;         /* Number of Transport Channel       */
TFCS_PAR       TFCS;              /* Transport Format Combination Set */
TRCH_INFO      TrchInfo[MAX_NO_TRANSPORT_CHANNELS];
/* Transport format Information        */
USHORT         CCTrCHLength;      /* Length of CCTrCH                  */
USHORT         RMMaxLength[MAX_NO_TRANSPORT_CHANNELS];
/* Max length after RateMatching       */
USHORT         RMParameterLength;    /* Length of RateMatching Parameters*/
UCHAR          RMParameter[4608];    /* RateMatching Parameters(Variable Length)*/
} CPHY_TRCH_CONFIG_PAR;
```

A.2.2.3.2　Description of CPHY_TRCH_CONFIG_PAR structure

Member			Description
DTXPosition			Specify the position of DTX after RateMatching for DownLink. Value on scenario　　Meaning DTX_FIXED_POSITION ------ Fixed Position DTX_FLEXIBLE_POSITION --- Flexible Position
InterLv2nd			Specify whether to perform 2nd Interleaving Value on scenario　　Meaning INTERLEAVE_OFF ---------- Interleaving OFF INTERLEAVE_ON ----------- Interleaving ON
PuncLimit			Specify Puncturing Limit used in RateMatching for UpLink. Value on scenario　　Meaning 50..100 (integer value) ------ 50% to 100%
NumOfTrch			Specify the number of Transport Channels in CCTrCH.　You can specify 1-8.
TFCS			Specify TFCS (Transport Format Combination Set).
TrchInfo			Specify parameters of each Transport Channel in CCTrCH.　Description of each parameter of TrchInfo is as follows:
	Trch		Transport Channel Type. DownLink: You can specify D_BCH, D_PCH, D_FACH, or D_DCH. UpLink: You can specify U_RACH or U_DCH.
	TrchNo		Channel number of Transport Channel.　You can specify 0-7. Be sure to specify the same value as a number of TrchInfo. Example: specify “n” in case of TrchInfo[n]. Exception: the case where Trch is two or more kinds like FACH+PCH. If n= 0 considers as Trch = D_PCH and n= 1,2 consider as Trch = D_FACH, n= 1 will be set to TrchNo = 0, and n= 2 will be set to TrchNo = 1.That is, TrchNo is specified from 0 by each Trch.
	InterLv1st		Specify whether to perform 1st Interleaving. Value on scenario　　Meaning INTERLEAVE_OFF ---------- Interleaving OFF INTERLEAVE_ON ----------- Interleaving ON
	Static		Specify parameters of Semi-Static Part of TFS (Transport Format Set).　Description of each parameter of Static is as follows:
		TTI	Specify TTI. Value on scenario　Meaning 1 ----------------------- TTI = 10ms 2 ----------------------- TTI = 20ms 4 ----------------------- TTI = 40ms 8 ----------------------- TTI = 80ms(only when NumOfTrch is 1)
		EPType	Specify Error Protection Type. Value on scenario　　Meaning CODING_NO -------------- no Coding CODING_CONV -- Convolutional Coding CODING_TURBO ---------- Turbo Coding
		CodingRate	Specify Coding Rate. Value on scenario　　Meaning CODINGRATE1_2 ------- Coding Rate 1/2 CODINGRATE1_3 ------- Coding Rate 1/3
		CRC_Size	Specify the number of CRC bits.　You can specify 0, 8, 12, 16 or 24. ＊
		RM_Attr	Specify RateMatching Attribute. Value on scenario　　Specification 1..256(integer value) ---------- 1..256
	NumOfDynamic		Specify the number of parameters of Dynamic Part of TFS (Transport Format Set).　You can specify 1-64.
	Dynamic		Specify parameters of Dynamic Part of TFS (Transport Format Set) Description of each parameter in Dynamic is as follows:
		NumOfTB	Specify the number of Transport Blocks. ＊
		TBSize	Specify the number of bits in Transport Block. ＊
	CCTrCHLength **RMMaxLength** **RMParameterLength**		Cannot be specified in creating a scenario.　Calculated from the set value of CPHY_RL_SETUP_PAR structure　and set value of CPHY_TRCH_CONFIG_PAR structure specified as arguments specified in CalcRMParameter().

❋ Input bit length of Turbo Coding in downlink are confined to kinds below. If it is necessary for you to change setting value in scenario, configure the value that input bit length to the Turbo Coding makes the number below.

40,	48,	64,	128,	144,	192,	256,	336,	352,	356,
370,	376,	384,	512,	576,	592,	656,	672,	704,	712,
740,	768,	1008,	1024,	1056,	1068,	1152,	1176,	1184,	1296,
1312,	1344,	1360,	1408,	1424,	1480,	1688,	1728,	1776,	2304,
2368,	2624,	2688,	2816,	2896,	2960,	3168,	3352,	3360,	3520,
3584,	3700,	3755,	3848,	3856,	4032,	4224,	4440,	4480,	4506,
4694,	4704,	4928,	5016,	5114					

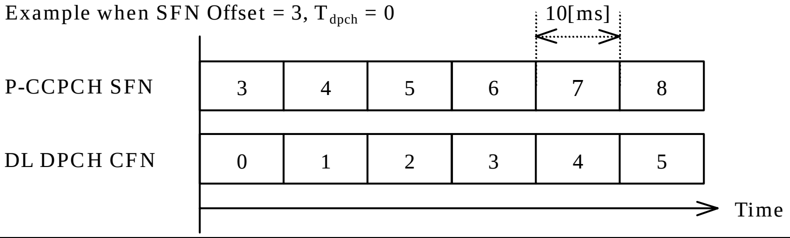
A.2.2.4　Structure for CphyRlSetup() (CPHY_RL_SETUP_PAR structure)

A.2.2.4.1　Definition of CPHY_RL_SETUP_PAR structure

```
/* GAP_INFO_PAR */
typedef struct {
UCHAR          TGPSI;
UCHAR          Method;
UCHAR          FrameType;
UCHAR          FrameCombiningIndex;
UCHAR          DeltaSIR1;
UCHAR          DeltaSIRafter1;
UCHAR          DeltaSIR2;
UCHAR          DeltaSIRafter2;
USHORT         TGCFN;
USHORT         TGPRC;
USHORT         TGSN;
USHORT         TGL1;
USHORT         TGL2;
USHORT         TGD;
USHORT         TGPL1;
USHORT         TGPL2;
USHORT         TGPS_Reconfig_CFN;
} GAP_INFO_PAR;
```

```
/* CPHY_RL_SETUP_REQ and CPHY_RL_MODIFY_REQ */
typedef struct {
USHORT    Power;      /* DPDCH Power                          */
UCHAR     ChCode;    /* DPDCH Channelization Code */
UCHAR     Reserve; /* Reserved to fit alignment */
} DPDCH_PAR;

typedef struct {
USHORT         Offset;          /* Channel Offset                    */
USHORT         SFNOffset;      /* SFN Offset(P-CCPCH Only) */
ULONG          ScrCode;        /* Scrambling Code                   */
USHORT         SlotFormat;    /* Slot Format                        */
UCHAR          SymbolRate;    /* Symbol Rate                        */
UCHAR          ChCode;          /* Channelization Code          */
USHORT         MaxDLPower;    /* Maximum DL Power                  */
USHORT         MinDLPower;    /* Minimum DL Power                  */
UCHAR          AdjustTime;    /* Adjust Time of TxDiversiy      */
UCHAR          GapInfoFlag    /* keep off : 0, keep on : 1 */
UCHAR          DispFlag         /* Indication Flag */
ULONG          TmpCellID;      /* Temporary Cell ID (SSDT) */
USHORT         Power;             /* Power                              */
UCHAR          NumOfDPDCH;    /* Number of DPDCH                    */
UCHAR          Reserve1;        /* Reserved to fit alignment*/
DPDCH_PAR       Dpdch[3];        /* DPDCH Power and CH Code    */
UCHAR          AICHTiming;    /* AICH Transmission Timing */
UCHAR          AICHPositive;/* AICH Positive/Negative      */
UCHAR          TxDiversity; /* Tx Diversity                       */
UCHAR          RLSetupFlag; /* RLSetup Flag                       */
GAP_INFO_PAR                     /* Compressed Mode Info          */
GapInfo;
SYMRATE_TBL_STRUCT             /* TFCI-SymbolRate Table        */
SymRateTbl;} CPHY_RL_SETUP_PAR;
```

A.2.2.4.2　Description of CPHY_RL_SETUP_PAR structure

Member	Description
Offset	Specify the offset of each channel in chip.　Since PICH timing is used as a reference value, specify the value plus 7680. 　　Example　When Tdpch is 12, 12x256+7680 = 10752 For uplink, add 1024 to the value for downlink. Allowable setting range is 0-46080 for downlink and 1024-47104 for uplink. The initial value is a set value in Timing on the Parameter Setup screen. If the Offset of S_CCPCH to which PCH is mapped is changed, please set a value similar as the Offset of PICH.
SFNOffset	Specify the offset between BTSs. The unit is chip. Specifiable range is 0-4095. Default value is 0. Relationship of SFN Offset, SFN and CFN is as below. CFN = {(SFN – SFN Offset) % 256 } Example when SFN Offset = 3, T_{dpch} = 0 
ScrCode	Specify the scrambling code. Specification method for downlink scrambling code 　bit 0 - bit3　Secondary Code No (0-15) 　bit 4 - bit6　Primary Code No (0-7) 　bit 7 - bit12 Primary Code Group(0-63) 　bit13 -bit31 reserve (Always set to 0) 　bit24 -bit31 Scrambling Code Change Type(Only SF/2 of Compressed Mode are effective.) (0:No change　1:Left alternative　2:Right alternative) 　(0:No change　1:Left alternative　2:Right alternative) Specification method for uplink scrambling code 　bit 0 -bit23　Scrambling Code No (000000h-FFFFFFh) 　bit 24 -bit31　Scrambling Code Type (0: Short　1: Long) In case the activated channels are P_SCH and S_SCH, this setting is invalid.
SlotFormat	Specify SlotFormat of CCTrCH. 　Value on scenario　　　　Meaning 　SLOT_FORMAT_0 ----------- #00 　SLOT_FORMAT_1 ----------- #01 　SLOT_FORMAT_2 ----------- #02 　SLOT_FORMAT_3 ----------- #03 　SLOT_FORMAT_4 ----------- #04 　SLOT_FORMAT_5 ----------- #05 　SLOT_FORMAT_6 ----------- #06 　SLOT_FORMAT_7 ----------- #07 　SLOT_FORMAT_8 ----------- #08 　SLOT_FORMAT_9 ----------- #09 　SLOT_FORMAT_10 ---------- #10 　SLOT_FORMAT_11 ---------- #11 　SLOT_FORMAT_12 ---------- #12 　SLOT_FORMAT_13 ---------- #13 　SLOT_FORMAT_14 ---------- #14 　SLOT_FORMAT_15 ---------- #15 　SLOT_FORMAT_16 ---------- #16 　SLOT_FORMAT_17 ---------- #17 　Note: For Slot Format, specify values within the following range: 　　Uplink Slot Format #0, #2, and #5 　　Downlink Slot Format #0 ~ #15 　　　(similar to for Parameter Setup window)
SymbolRate	Specify Symbol Rate of Physical Channel.

	Value on scenario Meaning SYMRATE15K -------------- 15ksps SYMRATE30K -------------- 30ksps SYMRATE60K -------------- 60ksps SYMRATE120K ------------- 120ksps SYMRATE240K ------------- 240ksps SYMRATE480K ------------- 480ksps SYMRATE960K ------------- 960ksps
ChCode	Specify Channelization Code of Physical Channel. For DPCH, specify Channelization Code of DPCCH. For S_SCH, specify the value of Primary Code Group of P_CCPCH and P_CPICH. For P_SCH, this setting is invalid.
MaxDLPower	Specify max Power of Down Link DPCH. Value on scenario Meaning -99..-10 ---------- -99dBm to -10dBm
MinDLPower	Specify min Power of Down Link DPCH Value on scenario Meaning -99..-10 ---------- -99dBm to -10dBm
AdjustTime	Specify AdjustTime of Tx Diversity Value on scenario Meaning 0-------------------- TS25.214 The downlink slot in which the adjustment is done 1) 1-------------------- TS25.214 The downlink slot in which the adjustment is done 2)
GapInfoFlag	When changing bearer combination with Compressed Mode, Gap Pattern will be 1: kept 0: overwritten by new gap pattern This setting is applied for TGPSI in GapInfo.
DispFlag	Enable DPCCH indication function or Preamble indication function. Value on scenario Meaning MEAS_REPORT_OFF ---invalidate DPCCH indication function or Preamble indication function(0) MEAS_REPORT_ON ----enable DPCCH indication function or Preamble indication function (1)
TmpCellID	Specify ID label,ID code length,FBI bits for SSDT

The value o the scenario	ID label	ID code length	FBI bits
SSDT_LABEL_A_SHORT_1BIT	A	short	1 bit
SSDT_LABEL_A_SHORT_2BIT	A	short	2 bit
SSDT_LABEL_A_MEDIUM_1BIT	A	medium	1 bit
SSDT_LABEL_A_MEDIUM_2BIT	A	medium	2 bit
SSDT_LABEL_A_LONG_1BIT	A	long	1 bit
SSDT_LABEL_A_LONG_2BIT	A	long	2 bit
SSDT_LABEL_B_SHORT_1BIT	B	short	1 bit
SSDT_LABEL_B_SHORT_2BIT	B	short	2 bit
SSDT_LABEL_B_MEDIUM_1BIT	B	medium	1 bit
SSDT_LABEL_B_MEDIUM_2BIT	B	medium	2 bit
SSDT_LABEL_B_LONG_1BIT	B	long	1 bit
SSDT_LABEL_B_LONG_2BIT	B	long	2 bit
SSDT_LABEL_C_SHORT_1BIT	C	short	1 bit
SSDT_LABEL_C_SHORT_2BIT	C	short	2 bit
SSDT_LABEL_C_MEDIUM_1BIT	c	medium	1 bit
SSDT_LABEL_C_MEDIUM_2BIT	c	medium	2 bit
SSDT_LABEL_C_LONG_1BIT	c	long	1 bit
SSDT_LABEL_C_LONG_2BIT	c	long	2 bit
SSDT_LABEL_D_SHORT_1BIT	d	short	1 bit
SSDT_LABEL_D_SHORT_2BIT	d	short	2 bit
SSDT_LABEL_D_MEDIUM_1BIT	d	medium	1 bit
SSDT_LABEL_D_MEDIUM_2BIT	d	medium	2 bit
SSDT_LABEL_D_LONG_1BIT	d	long	1 bit
SSDT_LABEL_D_LONG_2BIT	d	long	2 bit
SSDT_LABEL_E_SHORT_1BIT	e	short	1 bit
SSDT_LABEL_E_SHORT_2BIT	e	short	2 bit
SSDT_LABEL_E_MEDIUM_1BIT	e	medium	1 bit
SSDT_LABEL_E_MEDIUM_2BIT	e	medium	2 bit
SSDT_LABEL_E_LONG_1BIT	e	long	1 bit
SSDT_LABEL_E_LONG_2BIT	e	long	2 bit
SSDT_LABEL_F_SHORT_1BIT	f	short	1 bit
SSDT_LABEL_F_SHORT_2BIT	f	short	2 bit
SSDT_LABEL_F_MEDIUM_1BIT	f	medium	1 bit
SSDT_LABEL_F_MEDIUM_2BIT	f	medium	2 bit

SSDT_LABEL_F_LONG_1BIT	f	long	1 bit
SSDT_LABEL_F_LONG_2BIT	f	long	2 bit
SSDT_LABEL_G_SHORT_1BIT	g	short	1 bit
SSDT_LABEL_G_SHORT_2BIT	g	short	2 bit
SSDT_LABEL_G_MEDIUM_1BIT	g	medium	1 bit
SSDT_LABEL_G_MEDIUM_2BIT	g	medium	2 bit
SSDT_LABEL_G_LONG_1BIT	g	long	1 bit
SSDT_LABEL_G_LONG_2BIT	g	long	2 bit
SSDT_LABEL_H_SHORT_1BIT	h	short	1 bit
SSDT_LABEL_H_SHORT_2BIT	h	short	2 bit
SSDT_LABEL_H_MEDIUM_1BIT	h	medium	1 bit
SSDT_LABEL_H_MEDIUM_2BIT	h	medium	2 bit
SSDT_LABEL_H_LONG_1BIT	h	long	1 bit
SSDT_LABEL_H_LONG_2BIT	h	long	2 bit

Power	Specify Power of Downlink Physical Channel. This setting is meaningless for Uplink Physical Channel. For DPCH, specify Power of DPCCH. Value on scenario Meaning -99..-10(-13) ---------- -99dBm to -10dBm(-13dBm)
NumOfDPDCH	Specify the number of Physical Channels to which a single CCTrCH is mapped.
Dpdch	Specify parameters per DPDCH. **Power**: Specify Power of DPDCH. **ChCode**: Specify Channelization Code of DPDCH.
TxDiversity	Specify whether to perform Tx Diversity. Value on scenario Meaning DIVERSITY_OFF -------------------- Diversity OFF DIVERSITY_OPEN_STTD -------- Open Loop STTD DIVERSITY_OPEN_TSTD -------- Open Loop TSTD DIVERSITY_CLOSE_MODE1 --- Closed Loop mode1 DIVERSITY_CLOSE_MODE2 --- Closed Loop mode2
AICHTiming	Specify AccessSlot of AICH. Value on scenario Meaning AICH_3ACCESS_SLOT ------- 3 AccessSlot(Transmission Timing =0) AICH_4ACCESS_SLOT ------- 4 AccessSlot (Transmission Timing =1)
AICHPositive	Specify whether to set AICH transmission to Positive Ack or Negative Ack. Value on scenario Meaning AICH_POSITIVE ----------- Positive Ack AICH_NEGATIVE ----------- Negative Ack
RLSetupFlag	Specify whether to validate the settings including Power. Value on scenario Meaning ALL_RL_SETUP ------------validate all settings including Power SETUP_EXCL_POWER -------- validate all settings excluding Power.
GapInfo	Specify the parameters of Compressed Mode Patterns. Description for each parameter of GapInfo are as follows:

TGPSI	Specify TGPSI (Transmission Gap Pattern Sequence Identifier) Default value is TGPSI_1. Value on scenario Meaning TGPSI_1..TGPSI_6 -------- TGPSI#1..TGPSI#6
Method	Specify the Method of Compressed Mode. Value on scenario Meaning CM_MODE_NONE ------------ (Not activate Compressed Mode) CM_MODE_SFR ------------- Spreading Factor Reduction CM_MODE_PUNC ------------ Puncturing CM_MODE_HLS ------------- Higher Layer Scheduling
FrameType	Specify FrameStructure of Downlink Compressed Mode Frame. Value on scenario Meaning FRAME_TYPE_A ------------ Type A FRAME_TYPE_B ------------ Type B
FrameCombiningIndex	You can not specify when creating scenario.

	DeltaSIR1 **Delta-SIRafter1** **DeltaSIR2** **Delta-SIRafter2**	Specify ΔSIR1_coding or ΔSIR2_coding. Value on scenario Meaning 0..30 ------------------ 0.0..3.0(dB)
	TGCFN	Specify CFN to start Compressed Mode Pattern. Value on scenario Meaning 0..255 ------------------ 0..255
	TGPRC	Specify the number to repeat Compressed Mode Pattern. Value on scenario Meaning 1..511 ------------------ 1..511 REP_INFINITY ------------ infinity
	TGSN	Specify the Slot Number to start the Compressed Mode in the length of Slot. Value on scenario Meaning 0..14 ------------------- 0..14
	TGL1	Specify the length of Gap1 of Compressed Mode Pattern in the length of Slot. Value on scenario Meaning 1..14 ------------------- 1..14
	TGL2	Specify the length of Gap2 of Compressed Mode Pattern in the length of Slot. Value on scenario Meaning 0..14 ------------------- 0..14
	TGD	Specify the interval between the heads of Gap1 and Gap2 of Compressed Mode Pattern in the length of Slot. Value on scenario Meaning 15..269 ----------------- 15..269 0 ----------------------- In case Gap2 does not exist.
	TGPL1	Specify the length of pattern1 of Compressed Mode Pattern in the length of Frame. Value on scenario Meaning 1..144 ------------------ 1..144
	TGPL2	Specify the length of pattern2 of Compressed Mode Pattern in the length of Frame. Value on scenario Meaning 1..144 ------------------ 1..144
	TGPS_Reconfig_CFN	Specify the CFN to switch the settings depending on the combinations with values specified by TGCFN. Value on scenario Meaning 0..255 ------------------ 0..255 INVALID_PARAM ----------- No specification

A.2.2.5 Structure for CpdcpConfig() (CPDCP_CONFIG_PAR structure)

A.2.2.5.1 Definition of CPDCP_CONFIG_PAR structure

```
typedef struct {
    UCHAR        SRNS_Reloc;
    UCHAR        Reserve[11];
} PDCP_INFO;
```

```
typedef struct {
    UCHAR        SN_Sync;
    UCHAR        Reserve1;
    USHORT       Start_DL_Send_SN;
    USHORT       Start_UL_Receive_SN;
    USHORT       Frame;
    UCHAR        LowerLayer;
    UCHAR        UpperLayer;
    UCHAR        Reserve2[2];
    PDCP_INFO    PdcpInfo;
} CPDCP_CONFIG_PAR;
```

A.2.2.5.2 Description of CPDCP_CONFIG_PAR structure

Member	Description	
SN_Sync	Not in use	
Start_DL_Send_SN	The initial value of transmitted SN. 0 to 32767 can be set.	
Start_UL_Receive_SN	The initial value of received SN. 0 to 32767 can be set.	
Frame	The following can be set. RLC_AM_DATA_REQ RLC_UM_DATA_REQ RLC_TR_DATA_REQ	
LowerLayer	Lower layer is specified. RLC is specified.	
UpperLayer	Upper layer is specified. TE is specified.	
PdcpInfo	Not in use	
	SRNS_Reloc	Not in use

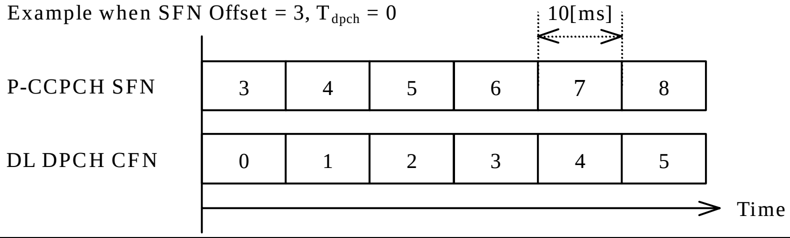
A.2.2.6 Structure for CphyRlSetupMtgp()(CPHY_RL_SETUP_MTGP_PAR structure)

A.2.2.6.1 Definition of CPHY_RL_SETUP_MTGP_PAR

```
/* Multiple TGP GAP_INFO_PAR */
typedef struct {
      UCHAR          Reserve1;
      UCHAR          Method;
      UCHAR          FrameType;
      UCHAR          FrameCombiningIndex;
      UCHAR          DeltaSIR1;
      UCHAR          DeltaSIRafter1;
      UCHAR          DeltaSIR2;
      UCHAR          DeltaSIRafter2;
      USHORT         TGCFN;
      USHORT         TGPRC;
      USHORT         TGSN;
      USHORT         TGL1;
      USHORT         TGL2;
      USHORT         TGD;
      USHORT         TGPL1;
      USHORT         TGPL2;
      USHORT         Reserve2;
} GAP_INFO_PAR_MTGP;
```

```
/* CPHY_RL_SETUP_REQ and CPHY_RL_MODIFY_REQ */
typedef struct {
USHORT   Power;    /* DPDCH Power                    */
UCHAR    ChCode;   /* DPDCH Channelization Code */
UCHAR    Reserve; /* Reserved to fit alignment */
} DPDCH_PAR;

/* For Multiple TGP */
typedef struct {
         USHORT            Offset;            /* Channel Offset              */
         USHORT            SFNOffset;         /* SFN Offset(P-CCPCH Only) */
         ULONG             ScrCode;           /* Scrambling Code             */
         USHORT            SlotFormat;        /* Slot Format                 */
         UCHAR             SymbolRate;        /* Symbol Rate                 */
         UCHAR             ChCode;            /* Channelization Code         */
         USHORT            MaxDLPower;        /* Maximum DL Power            */
         USHORT            MinDLPower;        /* Minimum DL Power            */
         UCHAR             NumOfPreamble;     /* Num of Preamble ignored     */
         UCHAR             AdjustTime;        /* Reserved                    */
         UCHAR             Reserve[2];        /* Reserved                    */
         ULONG             TmpCellID;         /* Temporary Cell ID (SSDT) */
         USHORT            Power;             /* Power                       */
         UCHAR             NumOfDPDCH;        /* Number of DPDCH             */
         UCHAR             AICHAck;           /* AICH Ack/Nack               */
         DPDCH_PAR          Dpdch[3];         /* DPDCH Power and CH Code     */
         UCHAR             AICHTiming;        /* AICH Transmission Timing */
         UCHAR             AICHPositive;      /* AICH Positive/Negative      */
         UCHAR             TxDiversity;       /* Tx Diversity                */
         UCHAR             RLSetupFlag;       /* RLSetup Flag                */
         UCHAR             GapInfoFlag[6];    /* 1:Keep On, 0:Keep Off       */
         UCHAR             DispFlag        /* Indication Flag */
         USHORT            TGPS_Reconfig_CFN; /* TGPS Reconfiguration CFN*/
         GAP_INFO_PAR_MTGP   GapInfo[6]; /* Compressed Mode Info         */
         SYMRATE_TBL_STRUCT                   /* TFCI-SymbolRate Table       */
                        SymRateTbl;
} CPHY_RL_SETUP_MTGP_PAR;
```

A.2.2.6.2 Description of CPHY_RL_SETUP_MTGP_PAR structure

Member	Description
Offset	Specify the offset of each channel in chip. Since PICH timing is used as a reference value, specify the value plus 7680. Example When Tdpch is 12, 12x256+7680 = 10752 For uplink, add 1024 to the value for downlink. Allowable setting range is 0-46080 for downlink and 1024-47104 for uplink. The initial value is a set value in Timing on the Parameter Setup screen.
SFNOffset	Specify the offset between BTSs. The unit is chip. Specifiable range is 0-4095 Default value is 0. Relationship of SFN Offset, SFN and CFN is as below. CFN = {(SFN – SFN Offset) % 256 } Example when SFN Offset = 3, T_{dpch} = 0 
ScrCode	Specify the scrambling code. Specification method for downlink scrambling code bit 0 - bit3 Secondary Code No (0-15) bit 4 - bit6 Primary Code No (0-7) bit 7 - bit12 Primary Code Group(0-63) bit13 -bit31 reserve (Always set to 0) bit24 -bit31 Scrambling Code Change Type(Only SF/2 of Compressed Mode are effective.) (0:No change 1:Left alternative 2:Right alternative) (0:No change 1:Left alternative 2:Right alternative) Specification method for uplink scrambling code bit 0 -bit23 Secondary Code No (000000h-FFFFFFh) bit 24 -bit31 Scrambling Code Type (0: Short 1: Long) In case the activated channels are P_SCH and S_SCH, this setting is invalid.
SlotFormat	Specify SlotFormat of CCTrCH. Value on scenario Meaning SLOT_FORMAT_0 ----------- #00 SLOT_FORMAT_1 ----------- #01 SLOT_FORMAT_2 ----------- #02 SLOT_FORMAT_3 ----------- #03 SLOT_FORMAT_4 ----------- #04 SLOT_FORMAT_5 ----------- #05 SLOT_FORMAT_6 ----------- #06 SLOT_FORMAT_7 ----------- #07 SLOT_FORMAT_8 ----------- #08 SLOT_FORMAT_9 ----------- #09 SLOT_FORMAT_10 ---------- #10 SLOT_FORMAT_11 ---------- #11 SLOT_FORMAT_12 ---------- #12 SLOT_FORMAT_13 ---------- #13 SLOT_FORMAT_14 ---------- #14 SLOT_FORMAT_15 ---------- #15 SLOT_FORMAT_16 ---------- #16 SLOT_FORMAT_17 ---------- #17 Note: For Slot Format, specify values within the following range: Uplink Slot Format #0, #2, and #5 Downlink Slot Format #0 ~ #15 (similar to for Parameter Setup window)
SymbolRate	Specify Symbol Rate of Physical Channel. Value in scenario meaning

<table>
<tr><th>Member</th><th colspan="4">Description</th></tr>
<tr><td></td><td colspan="4">SYMRATE15K -------------- 15ksps
SYMRATE30K -------------- 30ksps
SYMRATE60K -------------- 60ksps
SYSpecify Symbol Rate of Physical Channel.
Value on scenario　　　　Meaning
SYMRATE15K -------------- 15ksps
SYMRATE30K -------------- 30ksps
SYMRATE60K -------------- 60ksps
SYMRATE120K ------------- 120ksps
SYMRATE240K ------------- 240ksps
SYMRATE480K ------------- 480ksps
SYMRATE960K ------------- 960kspsMRATE120K ------------- 120ksps
SYMRATE240K ------------- 240ksps
SYMRATE480K ------------- 480ksps
SYMRATE960K ------------- 960ksps</td></tr>
<tr><td>ChCode</td><td colspan="4">Specify Channelization Code of Physical Channel.
For DPCH, specify Channelization Code of DPCCH.
For S_SCH, specify the value of Primary Code Group of P_CCPCH and P_CPICH.
For P_SCH, this setting is invalid.</td></tr>
<tr><td>MaxDLPower</td><td colspan="4">Specify max Power of Down Link DPCH.
Value on scenario　　　　Meaning
-99..-10　---------- -99dBm to -10dBm</td></tr>
<tr><td>MinDLPower</td><td colspan="4">Specify min Power of Down Link DPCH
Value on scenario　　　　Meaning
-99..-10　---------- -99dBm to -10dBm</td></tr>
<tr><td>AdjustTime</td><td colspan="4">Specify AdjustTime of Tx Diversity
Value on scenario　　　　Meaning
0-------------------- TS25.214 The downlink slot in which the adjustment is done 1)
1-------------------- TS25.214 The downlink slot in which the adjustment is done 2)</td></tr>
<tr><td rowspan="37">TmpCellID</td><td colspan="4">Specify ID label,ID code length,FBI bits for SSDT</td></tr>
<tr><td>The value o the scenario</td><td>ID label</td><td>ID code length</td><td>FBI bits</td></tr>
<tr><td>SSDT_LABEL_A_SHORT_1BIT</td><td>A</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_A_SHORT_2BIT</td><td>A</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_A_MEDIUM_1BIT</td><td>A</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_A_MEDIUM_2BIT</td><td>A</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_A_LONG_1BIT</td><td>A</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_A_LONG_2BIT</td><td>A</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_B_SHORT_1BIT</td><td>B</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_B_SHORT_2BIT</td><td>B</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_B_MEDIUM_1BIT</td><td>B</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_B_MEDIUM_2BIT</td><td>B</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_B_LONG_1BIT</td><td>B</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_B_LONG_2BIT</td><td>B</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_C_SHORT_1BIT</td><td>C</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_C_SHORT_2BIT</td><td>C</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_C_MEDIUM_1BIT</td><td>c</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_C_MEDIUM_2BIT</td><td>c</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_C_LONG_1BIT</td><td>c</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_C_LONG_2BIT</td><td>c</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_D_SHORT_1BIT</td><td>d</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_D_SHORT_2BIT</td><td>d</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_D_MEDIUM_1BIT</td><td>d</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_D_MEDIUM_2BIT</td><td>d</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_D_LONG_1BIT</td><td>d</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_D_LONG_2BIT</td><td>d</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_E_SHORT_1BIT</td><td>e</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_E_SHORT_2BIT</td><td>e</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_E_MEDIUM_1BIT</td><td>e</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_E_MEDIUM_2BIT</td><td>e</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_E_LONG_1BIT</td><td>e</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_E_LONG_2BIT</td><td>e</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_F_SHORT_1BIT</td><td>f</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_F_SHORT_2BIT</td><td>f</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_F_MEDIUM_1BIT</td><td>f</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_F_MEDIUM_2BIT</td><td>f</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_F_LONG_1BIT</td><td>f</td><td>long</td><td>1 bit</td></tr>
</table>

<table>
<tr><th>Member</th><th colspan="4">Description</th></tr>
<tr><td rowspan="13"></td><td>SSDT_LABEL_F_LONG_2BIT</td><td>f</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_G_SHORT_1BIT</td><td>g</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_G_SHORT_2BIT</td><td>g</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_G_MEDIUM_1BIT</td><td>g</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_G_MEDIUM_2BIT</td><td>g</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_G_LONG_1BIT</td><td>g</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_G_LONG_2BIT</td><td>g</td><td>long</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_H_SHORT_1BIT</td><td>h</td><td>short</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_H_SHORT_2BIT</td><td>h</td><td>short</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_H_MEDIUM_1BIT</td><td>h</td><td>medium</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_H_MEDIUM_2BIT</td><td>h</td><td>medium</td><td>2 bit</td></tr>
<tr><td>SSDT_LABEL_H_LONG_1BIT</td><td>h</td><td>long</td><td>1 bit</td></tr>
<tr><td>SSDT_LABEL_H_LONG_2BIT</td><td>h</td><td>long</td><td>2 bit</td></tr>
<tr><td>Power</td><td colspan="4">Specify Power of Downlink Physical Channel. This setting is meaningless for Uplink Physical Channel. For DPCH, specify Power of DPCCH.
Value on scenario Meaning
-99..-10(-13) ---------- -99dBm to -10dBm(-13dBm)</td></tr>
<tr><td>NumOfDPDCH</td><td colspan="4">Specify the number of Physical Channels to which a single CCTrCH is mapped.</td></tr>
<tr><td rowspan="3">Dpdch</td><td colspan="4">Specify parameters per DPDCH.</td></tr>
<tr><td>Power</td><td colspan="3">Specify Power of DPDCH.</td></tr>
<tr><td>ChCode</td><td colspan="3">Specify Channelization Code of DPDCH.</td></tr>
<tr><td>AICHTiming</td><td colspan="4">Specify AccessSlot of AICH.
Value on scenario Meaning
AICH_3ACCESS_SLOT ------- 3 AccessSlot(Transmission Timing =0)
AICH_4ACCESS_SLOT ------- 4 AccessSlot (Transmission Timing =1)</td></tr>
<tr><td>AICHPositive</td><td colspan="4">Specify whether to set AICH transmission to Positive Ack or Negative Ack.
Value on scenario Meaning
AICH_POSITIVE ----------- Positive Ack
AICH_NEGATIVE ----------- Negative Ack</td></tr>
<tr><td>TxDiversity</td><td colspan="4">Specify whether to perform Tx Diversity.
Value on scenario Meaning
DIVERSITY_OFF -------------------- Diversity OFF
DIVERSITY_OPEN_STTD -------- Open Loop STTD
DIVERSITY_OPEN_TSTD -------- Open Loop TSTD
DIVERSITY_CLOSE_MODE1 --- Closed Loop mode1
DIVERSITY_CLOSE_MODE2 --- Closed Loop mode2</td></tr>
<tr><td>RLSetupFlag</td><td colspan="4">Specify whether to validate the settings including Power.
Value on scenario Meaning
ALL_RL_SETUP ------------validate all settings including Power
SETUP_EXCL_POWER -------- validate all settings excluding Power.</td></tr>
<tr><td>GapInfoFlag</td><td colspan="4">When changing bearer combination with Compressed Mode, Gap Pattern will be
1: kept
0: overwritten by new gap pattern
GapInfoFlag[0 to 5] correspondes to TGPS#1 to 6.</td></tr>
<tr><td>DispFlag</td><td colspan="4">Enable DPCCH indication function or Preamble indication function.
Value on scenario Meaning
MEAS_REPORT_OFF ---invalidate DPCCH indication function or Preamble indication function(0)
MEAS_REPORT_ON ----enable DPCCH indication function or Preamble indication function (1)</td></tr>
<tr><td>TGPS_Reconfig_CFN</td><td colspan="4">Specify the CFN to switch the settings depending on the combinations with values specified by TGCFN.
Value on scenario Meaning
0..255 ------------------ 0..255
INVALID_PARAM ----------- No specification</td></tr>
<tr><td rowspan="2">GapInfo</td><td colspan="4">Specify the parameters of Compressed Mode Patterns for each TGPS. GapInfo[n] corresponds to TGPS#1 to #6. Description for each parameter of GapInfo are as follows:</td></tr>
<tr><td>Method</td><td colspan="3">Specify the Method of Compressed Mode.
Value on scenario Meaning
CM_MODE_NONE ------------ (Not activate Compressed Mode)
CM_MODE_SFR ------------- Spreading Factor Reduction
CM_MODE_PUNC ------------ Puncturing
CM_MODE_HLS ------------- Higher Layer Scheduling</td></tr>
</table>

Member	Description	
	FrameType	Specify FrameStructure of Downlink Compressed Mode Frame. Value on scenario Meaning FRAME_TYPE_A ------------ Type A FRAME_TYPE_B ------------ Type B
	FrameCombin-ingIndex	You can not specify when creating scenario.
	DeltaSIR1 **DeltaSIRafter1** **DeltaSIR2** **DeltaSIRafter2**	Specify ΔSIR1_coding or ΔSIR2_coding. Value on scenario Meaning 0..30 ------------------ 0.0..3.0(dB)
	TGCFN	Specify CFN to start Compressed Mode Pattern. Value on scenario Meaning 0..255 ------------------ 0..255
	TGPRC	Specify the number to repeat Compressed Mode Pattern. Value on scenario Meaning 1..511 ------------------ 1..511 REP_INFINITY ------------ infinity
	TGSN	Specify the Slot Number to start the Compressed Mode in the length of Slot. Value on scenario Meaning 0..14 ------------------- 0..14
	TGL1	Specify the length of Gap1 of Compressed Mode Pattern in the length of Slot. Value on scenario Meaning 1..14 ------------------- 1..14
	TGL2	Specify the length of Gap2 of Compressed Mode Pattern in the length of Slot. Value on scenario Meaning 0..14 ------------------- 0..14
	TGD	Specify the interval between the heads of Gap1 and Gap2 of Compressed Mode Pattern in the length of Slot. Value on scenario Meaning 15..269 ----------------- 15..269 0 ---------------------- In case Gap2 does not exist.
	TGPL1	Specify the length of pattern1 of Compressed Mode Pattern in the length of Frame. Value on scenario Meaning 1..144 ------------------ 1..144
	TGPL2	Specify the length of pattern2 of Compressed Mode Pattern in the length of Frame. Value on scenario Meaning 1..144 ------------------ 1..144

A.2.2.7 Structure for CteConfig()(CTE_CONFIG_PAR structure)

A.2.2.7.1 Definition of CTE_CONFIG_PAR

```
typedef struct{
    USHORT     TTI;                    /* TTI */
    USHORT     ProtoType;              /* Protocol Type */
    USHORT     Version;                /* Version */
} RLP_PAR;
```

```
typedef struct{
    RLP_PAR    RlpConfPar;
    UCHAR      ConElement;         /* Connection Element */
    UCHAR      SyncMode;           /* Synchronous Mode */
    UCHAR      InfoTransCap;       /* Information Transfer Capability */
    UCHAR      UserInfoL2P;        /* User Information Layer2 Protocol */
    USHORT     SerialRxBuff;       /* Serial Receive Buffer */
    UCHAR      Waiur;              /* Wanted Air Interface User Rate */
    UCHAR      AcptChCoding;       /* Acceptable Channel Coding */
    UCHAR      UpMaxTchNo;         /* Uplink Max TCH/F Number */
    UCHAR      DownMaxTchNo;       /* Downlink Max TCH/F Number */
    UCHAR      Uimi;               /* UIMI */
    UCHAR      Reserve1;           /* Reserve field */
} CSD_PAR;
```

```
typedef struct{
    UCHAR        TeType;
    UCHAR        Rate;
    UCHAR        TTI;
    UCHAR        NumOfTB;
    USHORT       TBS;
    USHORT       Frame;
    UCHAR        Layer;
    UCHAR        Reserve2;
    UCHAR        Data[1024];
    USHORT       Reserve3;
    CSD_PAR      CsdConfPar;    /* CSD Config Parameter */
} CTE_CONFIG_PAR;
```

A.2.2.7.2　Description of CTE_CONFIG_PAR structure

Member			Description
TeType			Refer to the description for CteConfig( ) function in "A.1 DETAILS OF EACH FUNCTION"
Rate			Same as above
TTI			Same as above
NumOfTB			Same as above
TBS			Same as above
Frame			Same as above
Layer			Same as above
Data[1024]			Same as above
CsdConfPar	**RlpConfPar**	**TTI**	Transmission time interval for downlink data
		ProtoType	Select a radio system. You need to set with any of the followings. RLP_PROTO_UMTS, RLP_PROTO_GSM
		Version	Select RLP version with any of the followings. RLP_VERSION_0, RLP_VERSION_1, RLP_VERSION_2
	ConElement		Set Connection Element. Please always set CSD_CE_NT to this variable.
	SyncMode		Set transfer mode. Please always set CSD_SYNCMD_ASYNC to this variable.
	InfoTransCap		Set Information transfer capability. Please always set CSD_ITC_AUDIO to this variable.
	UserInfoL2P		Set User information layer2 protocol. Please always set CSD_UIL2P_OUTBAND to this variable.
	SerialRxBuff		Set receiving interval at serial interface. Please always set 100 to this variable.
	Waiur		Select Wanted Air Interface User Rate. You need to set with any of the followings. CSD_WAIUR_9_6, CSD_WAIUR_14_4
	AcptChCoding		Select Acceptable Channel Coding. You need to set with any of the followings. CSD_ACC_9_6, CSD_ACC_14_4
	UpMaxTchNo		Select Uplink Maximum TCH number. You need to set with any of the followings. 1-- 4
	DownMaxTchNo		Select Downlink Maximum TCH number. You need to set with any of the followings. 1-- 4
	Uimi		Select UIMI. You need to set with any of the followings. 1-- 4

A.2.2.8　Structure for BtsOcnsActivate()(CPHY_OCNS_ACT_PAR structure)

A.2.2.8.1　Definition of CPHY_OCNS_ACT_PAR

```
typedef struct{
    USHORT   Power;             /* OCNS Power               */
    USHORT   BtsOffset;         /* BtsOffset                */
    ULONG    ScrCode;           /* Scrambling Code          */
    USHORT   DpchOffset;        /* DPCH Offset              */
    USHORT   DpchSlotFormat;    /* DPCH    Slot Format      */
    USHORT   DpchDpcchPower;    /* DPCCH Power              */
    USHORT   DpchPowerStep;     /* DPCH Power Control Step  */
    USHORT   MaxDLPower;        /* Maximum DL Power         */
    USHORT   MinDLPower;        /* Minimum DL Power         */
    USHORT   ChCode[ 16 ];      /* OCNS Pattern             */
} CPHY_OCNS_ACT_PAR;
```

A.2.2.8.2　Description of CPHY_OCNS_ACT_PAR structure

Member	Description
Power	Specify initial Power of OCNS. 0(0dBm) to.-99(-99dBm) in 1dBm step
BtsOffset	Specify same value as BTS#x BTS offset screen set value.
ScrCode	Scrambling Code of OCNS. Specify same value as downlink DPCH Scrambling Code.
DpchOffset	This variable sets Offset of downlink DPCH to OCNS. Specify same value as downlink DPCH Offset.
DpchSlotForamt	This variable sets Slot Format of downlink DPCH to OCNS. Specify same value as downlink DPCH Slotformat
DpchDpcchPower	This variable sets Power of downlink DPCH to OCNS. Specify DPCH Power at the point in activating OCNS.
DpchPowerStep	Specify DPCH power step value of automatic power control.. When using function BtsOcnsPower(), downlink DPCH and OCNS power are controlled by uplink TPC bit, this power step value is used. POWER_AUTO_TPC_05 : 0.5dB Step POWER_AUTO_TPC_1 : 1dB Step POWER_AUTO_TPC_15 : 1.5dB Step POWER_AUTO_TPC2 : 2dB Step
MaxDLPower	This variable sets upper limit of the downlink DPCH(DPCCH) Power to OCNS. Specify the same value as upper limit of the downlink DPCH(DPCCH) Power.
MinDLPower	This variable sets lower limit of the downlink DPCH(DPCCH) Power to OCNS. Specify the same value as lower limit of the downlink DPCH(DPCCH) Power.
ChCode[16]	Initial value of PN9 for Data of OCNS for 16 channels. Specify them within the range of 0 to 511. When this value is specified for 0, initial value of PN9 are set as the same value as each channelization code.

A.2.2.9　Structure for CbmcConfig()(CBMC_CONFIG_PAR structure)

A.2.2.9.1　Definition of CBMC_CONFIG_PAR

```
typedef struct {
    USHORT      CTCH_allocation_Period;     /* N:Period of CTCH allocation (SIB5 or SIB6) */
    USHORT      CBS_Frame_Offset;           /* K:CBS frame offset (SIB5 or SIB6) */
} LEVEL1_INFO;
```

```
typedef struct {
    UCHAR           BMC_Level2_Mode;    /* Level2 scheduling method */
    USHORT          RLCSpecialLI;       /* 1:use UMD special LI    0:not use */
    USHORT          RedFunc;            /* The setting of "PipeLine Delay Reduction" */
    USHORT          Frame;              /* Transmit primitive (specify RLC_UM_DATA_REQ) */
    USHORT          PU_Length;          /* RLC PU_Length */
    USHORT          MaxTFI_FACH;        /* Maximum value of FACH's TFI on which CTCH is mapped */
    USHORT          FACH_TTI;           /* TTI of FACH on which CTCH is mapped */
    LEVEL1_INFO     Level1Info;         /* CBS DRX Level1 information */
} CBMC_CONFIG_PAR;
```

A.2.2.9.2　Description of CBMC_CONFIG_PAR structure

<table>
<tr><th>Member</th><th colspan="2">Description</th></tr>
<tr><td>BMC_Level2_Mode</td><td colspan="2">Specify method of Level2 Scheduling. Specify CBS_MANUAL in v5.40.
Value on scenario　　Meaning
CBS_MANUAL　　Manual mode</td></tr>
<tr><td>RLCSpecialLI</td><td colspan="2">Specify whether UMD Special LI is used.
Value on scenario　　Meaning
0　　UMD Special LI is not used.
1　　UMD Special LI is used.</td></tr>
<tr><td>RedFunc</td><td colspan="2">Specify setting of Pipeline Delay Reduction.
Specify “value: 0x0100 | PLD3” when CTCH is mapped on FACH and Pipeline Delay Reduction function is used for FACH.</td></tr>
<tr><td>Frame</td><td colspan="2">Specify the transmission primitive. Set RLC_UM_DATA_REQ.</td></tr>
<tr><td>PU_Length</td><td colspan="2">Specify length of RLC_PDU (CTCH).
Set “PU_LengthUM” set by CrlcConfig(CTCH).</td></tr>
<tr><td>MaxTFI_FACH</td><td colspan="2">The maximum value of TFI of FACH used for CTCH is specified.</td></tr>
<tr><td>FACH_TTI</td><td colspan="2">Specify TTI of FACH used for CTCH.</td></tr>
<tr><td rowspan="3">Level1Info</td><td colspan="2">Specify a parameter for CBS level1 scheduling.
Set the same value as Level1Info specified by CMAC_CONFIG_PAR structure.</td></tr>
<tr><td>CTCH_allocation_Period</td><td>Specify cycle of CTCH allocation.</td></tr>
<tr><td>CBS_Frame_Offset</td><td>Specify CBS Frame offset.</td></tr>
</table>

A.2.2.10　Structure for SndBMCMessage()(BMC_DATA_PAR structure)

A.2.2.10.1　Definition of BMC_DATA_PAR

```
typedef struct {
    USHORT      Message_number;           /* Message number */
    USHORT      SendCTCH_BS_Offset;       /* Number of CTCH_BS from last BMC message */
} MESSAGE_ORDER_INFO;
```

```
typedef struct {
    UCHAR                 Flag;           /* 0:No change    1:Change BMC message */
    UCHAR                 NumOfMessage;
                            /* The number of messages that is broadcast in Repetition_CtchBS */
    USHORT                Repetition_CtchBS;  /* Repetition period (Unit:the number of CTCH_BS) */
    MESSAGE_ORDER_INFO Msg_Order[10];     /* The order and allocation of BMC message */
} LEVEL2_INFO;
```

```
typedef struct {
    USHORT      Flag;                     /* 0:No change    1:Change scheduling message */
    USHORT      Message_number;           /* Message number for scheduling message */
    USHORT      Size;                     /* The size of scheduling message (Unit:byte) */
    USHORT      Reserve[1];
    UCHAR       Data[1280];               /* The content of scheduling message */
} SHCEDULE_INFO;
```

```
typedef struct {
    USHORT      Flag;               /* 0:No change   1:Change CBS message   2:Delete CBS message */
    USHORT      Message_number;     /* Message number for CBS message */
    USHORT      NumOfBroadcast;     /* Number of times that CBS message is broadcast */
    USHORT      Size;               /* The size of CBS message (Unit:byte) */
    UCHAR       Data[1280];         /* The content of CBS message */
} CBS_DATA_INFO;
```

```
typedef struct {
    CBS_DATA_INFO      CBS_Msg;          /* the parameter of message changed or deleted */
    SCHEDULE_INFO      Schedule_Msg;     /* Scheduling message */
    LEVEL2_INFO        Level2Info;       /* Level2 Scheduling information */
} BMC_DATA_PAR
```

A.2.2.10.2 Definition of BMC_DATA_PAR structure

<table>
<tr><th>Member</th><th colspan="3">Description</th></tr>
<tr><td rowspan="6">CBS_Msg</td><td colspan="3">Specify CBS message for Change/Addition and Deletion. The explanation of each parameter of CBS_DATA_INFO structure is as follows.</td></tr>
<tr><td>Flag</td><td colspan="2">Specify the status of CBS message.
Value on scenario Meaning
0 No change.
1 Change/Addition of a message
2 Deletion of a message</td></tr>
<tr><td>Message_number</td><td colspan="2">Specify a message number. The value that can be specified is as follows.
CBSDATA_1 to CBSDATA10</td></tr>
<tr><td>NumOfBroadcast</td><td colspan="2">Specify number of times transmitting message. The value that can be specified is as follows.
1 to 0xFFFF 0:infinity</td></tr>
<tr><td>Size</td><td colspan="2">Specify size of CBS message.
1 to 1280byte</td></tr>
<tr><td>Data[1280]</td><td colspan="2">Specify a message that is changed/added.</td></tr>
<tr><td rowspan="6">Schedule_Msg</td><td colspan="3">Specify Scheduling message for Change/Addition. The explanation of each parameter of SCHEDULE_INFO structure is as follows.</td></tr>
<tr><td>Flag</td><td colspan="2">Specify the status of Scheduling message.
Value on scenario Meaning
0 No change
1 Change/Addition of a message</td></tr>
<tr><td>Message_number</td><td colspan="2">Specify a message number. The value that can be specified is as follows.
CBS_SCHEDULE_1 to CBS_SCHEDULE_2</td></tr>
<tr><td>Size</td><td colspan="2">Specify size of Scheduling message.
1 to 1280byte</td></tr>
<tr><td>Reserve[1]</td><td colspan="2">Reservation</td></tr>
<tr><td>Data[1280]</td><td colspan="2">Specify a message that is changed/added.</td></tr>
<tr><td rowspan="6">Level2Info</td><td colspan="3">Specify transmission method of BMC message. The explanation of each parameter of LEVEL2_INFO structure is as follows.</td></tr>
<tr><td>Flag</td><td colspan="2">Specify the status of the transmission method of a message.
Value on scenario Meaning
0 No change
1 Changed/Added of the transmission method of message</td></tr>
<tr><td>NumOfMessage</td><td colspan="2">Specify the number of messages transmitted in Repetition-Period.
0 to 10
If NumOfMessage is 2, messages of Mesg_Order[0] and Msg_Order[1] are transmitted.</td></tr>
<tr><td>Repetition_CtchBS</td><td colspan="2">Specify Repetition-Period. (unit: number of CTCH BS)</td></tr>
<tr><td rowspan="2">Msg_Order[ ]</td><td colspan="2">Specify transmission method of message. The setting of parameter of MESSAGE_ORDER_INFO structure is as follows.</td></tr>
<tr><td>Message_number</td><td>Specify a message number.
CBSDATA_1 to CBSDATA_10
CBS_SCHEDULE_1, CBS_SCHEDULE_2</td></tr>
</table>

		SendCTCH_BS_Offset	Case of MsgOrder[0]: - Specify the number of CTCH_BS from the beginning of Repetition-Period to transmission of this message. Case except for MsgOrder[0]: - Specify the number of CTCH_BS from the previous message to the transmission of this message.

A.2.3　Common variables

The following common variables are provided to use scenario libraries.　These variables use the common memory area and may be shared among scenarios.　Section A.2.3.2 shows the initial set values.

A.2.3.1　Types of common variables

A.2.3.1.1　Variables for CrlcConfig()

These variables contain parameters to specify the operation of the RLC layer. Specify the pointer to this variable in accordance with the Logical Channel to be specified when CrlcConfig() is used.

```
CRLC_CONFIG_PAR    CrlcConfigBCCH;          /* BCCH    */
CRLC_CONFIG_PAR    CrlcConfigPCCH;          /* PCCH    */
CRLC_CONFIG_PAR    CrlcConfigCCCH;          /* CCCH    */
CRLC_CONFIG_PAR    CrlcConfigDCCH;          /* DCCH(DPCH),DCCH(FACH CRNTI) */
CRLC_CONFIG_PAR    CrlcConfigDCCH_URNTI;    /* DCCH(FACH URNTI)            */
CRLC_CONFIG_PAR    CrlcConfigDTCH;          /* DTCH    */
CRLC_CONFIG_PAR    CrlcConfigCTCH;          /* CTCH    */
```

A.2.3.1.2 Variables for CmacConfig()

These variables contain parameters to specify the operation of the MAC layer. Specify the pointer to one of these variables in accordance with CCtrCH and the format of the activated channel to be specified when CmacConfig() is used.

```
CMAC_CONFIG_PAR    CmacConfigP_CCPCH;              /* P-CCPCH(BCCH) */
CMAC_CONFIG_PAR    CmacConfigS_CCPCH_PCH;          /* S-CCPCH(PCH)    */
CMAC_CONFIG_PAR    CmacConfigS_CCPCH_FACH;         /* S-CCPCH(FACH) */
CMAC_CONFIG_PAR    CmacConfigS_CCPCH_PCHxFACH;/* S-CCPCH(PCH+FACH) */
CMAC_CONFIG_PAR    CmacConfigPRACH10M;             /* PRACH(TTI=10msec) */
CMAC_CONFIG_PAR    CmacConfigPRACH20M;             /* PRACH(TTI=20msec) */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_SDCCH;         /* DPCH(Down Link Stand Alone DCCH) */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_SDCCH;         /* DPCH(Up Link Stand Alone DCCH)      */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AMR;           /* DPCH(Down Link AMR Speech)            */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AMR;           /* DPCH(Up Link AMR Speech)              */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_UDI;           /* DPCH(Down Link ISDN 1B UDI)           */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_UDI;           /* DPCH(Up Link ISDN 1B UDI)             */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AV64K;         /* DPCH(Down Link AV64K)                 */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AV64K;         /* DPCH(Up Link AV64K)                   */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AV32K;         /* DPCH(Down Link AV32K)                 */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AV32K;         /* DPCH(Up Link AV32K)                   */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_P32K;          /* DPCH(Down Link Packet32K)             */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_P32K;          /* DPCH(Up Link Packet32K)               */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_P64K;          /* DPCH(Down Link Packet64K)             */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_P64K;          /* DPCH(Up Link Packet64K)               */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_P128K;         /* DPCH(Down Link Packet128K)            */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_P128K;         /* DPCH(Up Link Packet128K)              */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_P384K;         /* DPCH(Down Link Packet384K)            */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_P384K;         /* DPCH(Up Link Packet384K)              */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AMRxP32K; /* DPCH(Down Link Packet AMR Speech + Packet32K)*/
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AMRxP32K; /* DPCH(Up Link Packet AMR Speech + Packet32K)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AMRxP64K; /* DPCH(Down Link Packet AMR Speech + Packet64K)*/
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AMRxP64K;      /* DPCH(Up Link Packet AMR Speech + Packet64K) */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AMRxP128K;/*DPCH(Down Link Packet AMR Speeech+Packet 28K)*/
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AMRxP128K;    /* DPCH(Up Link Packet AMR Speech + Packet 28K)*/
CMAC_CONFIG_PAR    CmacConfigD_DPCH_AMRxP384K;/*DPCH(Down Link Packet AMR Speech Packet384K)*/
CMAC_CONFIG_PAR    CmacConfigU_DPCH_AMRxP384K; /* DPCH(Up Link Packet AMR Speech + Packet384K)*/
CMAC_CONFIG_PAR    CmacConfigD_RMCDCCH;            /* DPCH(Down Link RMC DCCH) */
CMAC_CONFIG_PAR    CmacConfigU_RMCDCCH;            /* DPCH(Up Link RMC DCCH)      */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_RMC12_2K;      /* DPCH(Down Link RMC12.2K) */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_RMC12_2K;      /* DPCH(Up Link RMC12.2K)      */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_RMC64K;        /* DPCH(Down Link RMC64K)      */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_RMC64K;        /* DPCH(Up Link RMC64K)        */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_RMC144K;       /* DPCH(Down Link RMC144K)     */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_RMC144K;       /* DPCH(Up Link RMC144K)       */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_RMC384K;       /* DPCH(Down Link RMC384K)     */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_RMC384K;       /* DPCH(Up Link RMC384K)       */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_RMCBTFD;       /* DPCH(Down Link RMC BTFD) */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_RMCBTFD;       /* DPCH(Up Link RMC BTFD)      */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_CS14k4;        /* DPCH(Down Link CS14.4K) */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_CS14k4;        /* DPCH(Up Link CS14.4K)      */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_CS28k8;        /* DPCH(Down Link CS28.8K) */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_CS28k8;        /* DPCH(Up Link CS28.8K)      */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_CS57k6;        /* DPCH(Down Link CS57.6K) */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_CS57k6;        /* DPCH(Up Link CS57.6K)      */
```

```
CMAC_CONFIG_PAR    CmacConfigS_CCPCH_FACH_BMC;        /* S-CCPCH(FACH) for BMC    */
CMAC_CONFIG_PAR    CmacConfigS_CCPCH_PCHxFACH_BMC;    /* S-CCPCH(PCH+FACH) for BMC    */
```

And there are the following variables that users can configure voluntarily.

```
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User0;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User1;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User2;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User3;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User4;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User5;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User6;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User7;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User8;    /* DPCH User Define(DownLink)    */
CMAC_CONFIG_PAR    CmacConfigD_DPCH_User9;    /* DPCH User Define(DownLink)    */

CMAC_CONFIG_PAR    CmacConfigU_DPCH_User0;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User1;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User2;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User3;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User4;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User5;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User6;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User7;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User8;    /* DPCH User Define(UpLink)    */
CMAC_CONFIG_PAR    CmacConfigU_DPCH_User9;    /* DPCH User Define(UpLink)    */
```

A.2.3.1.3 Variables for CphyTrchConfig()

These variables contain parameters (mainly CodingParameter) to specify the operation of the PHY layer. Specify the pointer to one of these variables in accordance with CCtrCH and the format of the activated channel to be specified when CphyTrchConfig() and CalcRMParameter() are used.

```
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigP_CCPCH;              /* P-CCPCH(BCCH) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigS_CCPCH_PCH;          /* S-CCPCH(PCH)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigS_CCPCH_FACH;         /* S-CCPCH(FACH) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigS_CCPCH_PCHxFACH;     /* S-CCPCH(PCH+FACH) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigPRACH10M;             /* PRACH(TTI=10msec) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigPRACH20M;             /* PRACH(TTI=20msec) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_SDCCH;         /* DPCH(Down Link Stand Alone DCCH) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_SDCCH;         /* DPCH(Up Link Stand Alone DCCH)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_AMR;           /* DPCH(Down Link AMR Speech)            */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AMR;           /* DPCH(Up Link AMR Speech)              */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_UDI;           /* DPCH(Down Link ISDN 1B UDI)           */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_UDI;           /* DPCH(Up Link ISDN 1B UDI)             */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_AV64K;         /* DPCH(Down Link AV64K)                 */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AV64K;         /* DPCH(Up Link AV64K)                   */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_AV32K;         /* DPCH(Down Link AV32K)                 */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AV32K;         /* DPCH(Up Link AV32K)                   */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_P32K;          /* DPCH(Down Link Packet32K)             */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_P32K;          /* DPCH(Up Link Packet32K)               */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_P64K;          /* DPCH(Down Link Packet64K)             */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_P64K;          /* DPCH(Up Link Packet64K)               */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_P128K;         /* DPCH(Down Link Packet128K)            */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_P128K;         /* DPCH(Up Link Packet128K)              */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_P384K;         /* DPCH(Down Link Packet384K)            */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_P384K;         /* DPCH(Up Link Packet384K)              */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_AMRxP32K;      /* DPCH(Down Link Packet AMR Speech + Packet32K)*/
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AMRxP32K;      /* DPCH(Up Link Packet AMR Speech + Packet32K)   */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_AMRxP64K;      /* DPCH(Down Link Packet AMR Speech + Packet64K)*/
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AMRxP64K;      /* DPCH(Up Link Packet AMR Speech + Packet64K) */
CPHY_TRCH_CONFIG_PAR    CphyTrchonfigD_DPCH_AMRxP128K;      /* DPCH(Down Link Packet AMR Speech + Packet 28K)*/
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AMRxP128K;     /* DPCH(Up Link Packet AMR Speech + Packet 28K)*/
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_AMRxP384K;     /* DPCH(Down Link Packet AMR Speech + Packet384K)*/
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_AMRxP384K;     /* DPCH(Up Link Packet AMR Speech + Packet384K)*/
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_RMCDCCH;            /* DPCH(Down Link RMC DCCH) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_RMCDCCH;            /* DPCH(Up Link RMC DCCH)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_RMC12_2K;      /* DPCH(Down Link RMC12.2K) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_RMC12_2K;      /* DPCH(Up Link RMC12.2K)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_RMC64K;        /* DPCH(Down Link RMC64K)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_RMC64K;        /* DPCH(Up Link RMC64K)        */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_RMC144K;       /* DPCH(Down Link RMC144K)     */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_RMC144K;       /* DPCH(Up Link RMC144K)       */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_RMC384K;       /* DPCH(Down Link RMC384K)     */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_RMC384K;       /* DPCH(Up Link RMC384K)       */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_RMCBTFD;       /* DPCH(Down Link RMC BTFD) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_RMCBTFD;       /* DPCH(Up Link RMC BTFD)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_CS14k4;        /* DPCH(Down Link CS14.4K) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_CS14k4;        /* DPCH(Up Link CS14.4K)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_CS28k8;        /* DPCH(Down LinkCS28.8K) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_CS28k8;        /* DPCH(Up LinkCS28.8K)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_CS57k6;        /* DPCH(Down LinkCS57.6K) */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_CS57k6;        /* DPCH(Up LinkCS57.6K)      */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigS_CCPCH_FACH_BMC;         /* S-CCPCH(FACH)    for    BMC */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigS_CCPCH_PCHxFACH_BMC;     /* S-CCPCH(PCH+FACH)    for    BMC */
```

And there are the following variables that users can configure voluntarily.

```
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User0;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User1;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User2;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User3;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User4;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User5;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User6;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User7;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User8;    /* userdefined DPCH(DownLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigD_DPCH_User9;    /* userdefined DPCH(DownLink)    */

CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User0;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User1;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User2;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User3;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User4;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User5;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User6;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User7;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User8;    /* userdefined DPCH(UpLink)    */
CPHY_TRCH_CONFIG_PAR    CphyTrchConfigU_DPCH_User9;    /* userdefined DPCH(UpLink)    */
```

A.2.3.1.4 Variables for CphyRlSetup()

These variables contain parameters (mainly Radio Link Parameter) to specify the operation of the PHY layer. Specify the pointer to one of these variables in accordance with CCtrCH and the format of the activated channel to be specified when CphyTrchConfig() and CalcRMParameter() are used.

```
CPHY_RL_SETUP_PAR CphyRlSetupP_SCH;       /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_SCH;       /* S_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CPICH;     /* P_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CPICH;     /* S_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CCPCH;     /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CCPCH;     /* P_CCPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPICH;        /* PICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupAICH;        /* AICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPRACH;       /* PRACH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupD_DPCH;      /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupU_DPCH;      /* UpLink DPCH Radio Frame Parameter      */
CPHY_RL_SETUP_PAR CphyRlSetupAWGN;        /* AWGN Parameter                          */
USHORT            PageNp;                 /* PICH Np                                 */
*The above variables contains the same values as variables for BTS1 in START.
```

```
Variables for setting BTS1
CPHY_RL_SETUP_PAR CphyRlSetupP_SCH_BTS1;     /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_SCH_BTS1;     /* S_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CPICH_BTS1;   /* P_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CPICH_BTS1;   /* S_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CCPCH_BTS1;   /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CCPCH_BTS1;   /* P_CCPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPICH_BTS1;      /* PICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupAICH_BTS1;      /* AICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPRACH_BTS1;     /* PRACH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupD_DPCH_BTS1;    /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupU_DPCH_BTS1;    /* UpLink DPCH Radio Frame Parameter      */
CPHY_RL_SETUP_PAR CphyRlSetupAWGN_BTS1;      /* AWGN Parameter                          */
```

```
Variables for setting BTS2
CPHY_RL_SETUP_PAR CphyRlSetupP_SCH_BTS2;     /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_SCH_BTS2;     /* S_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CPICH_BTS2;   /* P_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CPICH_BTS2;   /* S_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CCPCH_BTS2;   /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CCPCH_BTS2;   /* P_CCPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPICH_BTS2;      /* PICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupAICH_BTS2;      /* AICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPRACH_BTS2;     /* PRACH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupD_DPCH_BTS2;    /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupU_DPCH_BTS2;    /* UpLink DPCH Radio Frame Parameter      */
CPHY_RL_SETUP_PAR CphyRlSetupAWGN_BTS2;      /* AWGN Parameter                          */
```

```
Variables for setting BTS3
CPHY_RL_SETUP_PAR CphyRlSetupP_SCH_BTS3;     /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_SCH_BTS3;     /* S_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CPICH_BTS3;   /* P_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CPICH_BTS3;   /* S_CPICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupP_CCPCH_BTS3;   /* P_SCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupS_CCPCH_BTS3;   /* P_CCPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPICH_BTS3;      /* PICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupAICH_BTS3;      /* AICH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupPRACH_BTS3;     /* PRACH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupD_DPCH_BTS3;    /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_PAR CphyRlSetupU_DPCH_BTS3;    /* UpLink DPCH Radio Frame Parameter      */
CPHY_RL_SETUP_PAR CphyRlSetupAWGN_BTS3;      /* AWGN Parameter                          */
```

A.2.3.1.5 Variables for CphyAttenuator()

These variables are the current setting value of TX Attenuator and RX Reference Power. Executing CphyAttenuator() provides the latest values.

```
INT CurrentTxAttenuator;          /* Current Tx Attenuator Value       */
INT CurrentRXRefPower;            /* Current Rx Reference Power Value */
INT CurrentTxAttenuatorBTS1    /* Current Tx Attenuator Value          */
INT CurrentRxRefPowerBS1;         /* Current Rx Reference Power Value BTS1 */
INT CurrentTxAttenuatorBTS2;   /*Current Tx Attenuator Value BTS1          */
INT CurrentRxRefPowerBTS2;        /* Current Rx Reference Power Value    BTS2*/
INT CurrentTxAttenuatorBTS3;   /* Current Tx Attenuator Value    BTS3        */
INT CurrentRxRefPowerBTS3;        /* Current Rx Reference Power Value BTS3    */
```

A.2.3.1.6 Variables for SndMessage() and RcvMessage()

```
USHORT SIB_POS;               /* SIB_POS(For transmission of BCCH)              */
USHORT SIB_REP;               /* SIB_REP(For transmission of BCCH)              */
UCHAR    PageIndicator;       /* PI Value(For transmission of PCCH)             */
USHORT PageNp;                /* Np (For transmission of PCCH)                  */
USHORT DRXCycleLength;        /* DRX Cycle Length(For transmission of PCCH) */
USHORT PagingBlock;           /* Paging Block(For transmission of PCCH)         */
USHORT RlcMUI;                /* MUI(For RLC_AM_DATA_XXX)                       */
USHORT RlcCNF;                /* CNF(For RLC_AM_DATA_XXX)                       */
USHORT RlcSpecialLI;          /* use UMD SpecialLI                              */
```

A.2.3.1.7 Variables for CteConnect()

Refer to 3.7.14 Chapter.

```
INT TeActTime;            /* CFN Value to start transmission in down link */
INT TeTtiCounter;         /* transmission counter in down link                */
```

A.2.3.1.8 Variables for CphyRlSetupMtgp( )

These variables contain parameters (mainly Radio Link Parameter) to specify the operation of the PHY layer. It can activate and suspend multiple TGPS with single execution.Specify the pointer to one of these variables in accordance with CCtrCH and the format of the activated channel to be specified when CphyTrchConfig() and CalcRMParameterMtgp() are used.

```
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpD_DPCH;          /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpU_DPCH;          /* UpLink DPCH Radio Frame Parameter      */
*The above variables contains the same values as variables for BTS1 in START.
```

```
Variables for setting BTS1
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpD_DPCH_BTS1;     /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpU_DPCH_BTS1;     /* UpLink DPCH Radio Frame Parameter      */
```

```
Variables for setting BTS2
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpD_DPCH_BTS2;     /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpU_DPCH_BTS2;     /* UpLink DPCH Radio Frame Parameter      */
```

```
Variables for setting BTS3
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpD_DPCH_BTS3;     /* DownLink DPCH Radio Frame Parameter */
CPHY_RL_SETUP_MTGP_PAR    CphyRlSetupMtgpU_DPCH_BTS3;     /* UpLink DPCH Radio Frame Parameter      */
```

A.2.3.1.9 Variables for CteConfig( )

Some parameters are included in this variable to set up TE layer. Please declare a variable that has the same structure locally in your scenario, when you run the scenario for other purpose than CSD. Please pass the pointer of this variable to CteConfig( ) function with a logical channel parameter when the function is called.

```
CTE_CONFIG_PAR              CteConfigCSD; /* for CSD */
```

A.2.3.1.10 Variables for BtsOcnsActivate( )

```
CPHY_OCNS_ACT_PAR        OcnsConfigPar;
CPHY_OCNS_ACT_PAR        OcnsConfigBTS1;     /* Variable for setting BTS1 */
CPHY_OCNS_ACT_PAR        OcnsConfigBTS2;     /* Variable for setting BTS2 */
CPHY_OCNS_ACT_PAR        OcnsConfigBTS3;     /* Valiable for setting BTS3 (Use prohibition) */
```

A.2.3.1.11 Variables for CbmcConfig()

These variables contain parameters to specify the operation of the BMC layer. Specify the pointer to one of these variables in accordance with CCtrCH and the format of the activated channel to be specified when CbmcConfig() is used.

```
CBMC_CONFIG_PAR     CbmcConfigCBS;              /* BMC */
CBMC_CONFIG_PAR     CbmcConfigCBS_BTS1;         /* BMC for BTS1 */
CBMC_CONFIG_PAR     CbmcConfigCBS_BTS2;         /* BMC for BTS2 */
CBMC_CONFIG_PAR     CbmcConfigCBS_BTS3;         /* BMC for BTS3 */
```

A.2.3.1.12　Other variables

The following variables can be used for handing over data between scenarios.

```
INT    UserVar0;
INT    UserVar1;
INT    UserVar2;
INT    UserVar3;
INT    UserVar4;
INT    UserVar5;
INT    UserVar6;
INT    UserVar7;
INT    UserVar8;
INT    UserVar9;

CHAR UserMem0[255];
CHAR UserMem1[255];
CHAR UserMem2[255];
CHAR UserMem3[255];
CHAR UserMem4[255];
CHAR UserMem5[255];
CHAR UserMem6[255];
CHAR UserMem7[255];
CHAR UserMem8[255];
CHAR UserMem9[255];
```

A.2.3.2 Initial set values of common variables

A.2.3.2.1 Variables for CrlcConfig()

For initial set values, refer to InitializeParameter.c that comes with MX848000 (in the folder MX848000).

Note)
"InitializeParameter.c" is a copy of initial values set in the firmware of Signalling Tester. Modifying the values in this file does not affect the initial values. In case it is necessary for you to modify initial set values in your scenario, describe a portion to set parameters in your scenario referring "InitializeParameter.c".

A.2.3.2.2 Variables for CmacCofig()

For initial set values, refer to InitializeParameterMACxPHY() in the attached InitializeParameter.c.

Input bit length for Turbo Coding in downlink is confined. If it is necessary for you to change set values, configure the value referring to A.2.2.3.2.

A.2.3.2.3 Variables for CphyTrchConfig()

For initial set values, refer to InitializeParameterMACxPHY() in the attached InitializeParameter.c.

Input bit length for Turbo Coding in downlink is confined. If it is necessary for you to change set values, configure the value referring to A.2.2.3.2.

A.2.3.2.4 Variables for CphyRlSetup()

1) Basic parameter

CphyRlSetupP_SCH /* for CPHY_RL_SETUP(P_SCH) */

Parameter	Initial set value	Meaning
Offset	7680	Offset=0 chip
ScrCode	BTS#1 screen set value	
SlotFormat	-	
SymbolRate	SYMRATE15K	Symbol Rate=15ksps
ChCode	-	
Power	BTS#1 screen set value	

CphyRlSetupS_SCH /* for CPHY_RL_SETUP(S_SCH) */

Parameter	Initial set value	Meaning
Offset	7680	Offset=0 chip
ScrCode	BTS#1 screen set value	
SlotFormat	-	
SymbolRate	SYMRATE15K	Symbol Rate=15ksps
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	

CphyRlSetupP_CPICH /* for CPHY_RL_SETUP(P_CPICH) */

Parameter	Initial set value	Meaning
Offset	7680	Offset=0 chip
ScrCode	BTS#1 screen set value	
SlotFormat	-	
SymbolRate	SYMRATE15K	Symbol Rate=15ksps
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	

CphyRlSetupP_CCPCH /* for CPHY_RL_SETUP(P_CCPCH) */

Parameter	Initial set value	Meaning
Offset	7680	Offset=0 chip
ScrCode	BTS#1 screen set value	
SlotFormat	-	
SymbolRate	SYMRATE15K	Symbol Rate=15ksps
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	

CphyRlSetupS_CCPCH /* for CPHY_RL_SETUP(S_CCPCH) */

Parameter	Initial set value	Meaning
Offset	BTS#1 screen set value +7680	Offset= BTS#1 screen set value
ScrCode	BTS#1 screen set value	
SlotFormat	BTS#1 screen set value	
SymbolRate	Value from Slot Format	
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	

CphyRlSetupPICH /* for CPHY_RL_SETUP(PICH) */

Parameter	Initial set value	Meaning
Offset	0	Offset=-7680 chip
ScrCode	BTS#1 screen set value	
SlotFormat	BTS#1 screen set value	
SymbolRate	SlotFormat	
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	

CphyRlSetupAICH /* for CPHY_RL_SETUP(AICH) */

Parameter	Initial set value	Meaning
Offset	7680	Offset=0 chip
ScrCode	BTS#1 screen set value	
SlotFormat	-	
SymbolRate	SYMRATE15K	Symbol Rate=15ksps
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	
AICHPositive	AICH_POSITIVE	AICH Positive
AICHAck	AICH_ACK	AICH Ack

CphyRlSetupPRACH /* for CPHY_RL_SETUP(PRACH) */

Parameter	Initial set value	Meaning
Offset	7680+2560	Offset=0 chip
ScrCode	BTS#1 screen set value	
SlotFormat	SLOT_FORMAT_0	Slot Format #0
SymbolRate	BTS#1 screen set value	
ChCode	BTS#1 screen set value	
Power	BTS#1 screen set value	
AICHTiming	AICH_3ACCESS_SLOT	Transmission Timing=0

CphyRlSetupD_DPCH /* for CPHY_RL_SETUP(D_DPCH) */

Parameter	Initial set value	Meaning
Offset	BTS#1 screen set value +7680	Offset= screen set value
ScrCode	BTS#1 screen set value	
SlotFormat	BTS#1 screen set value	
SymbolRate	Value from Slot Format	
ChCode	BTS#1 screen set value (DPCCH)	
Power	BTS#1 screen set value (DPCCH)	
NumOfDPDCH	BTS#1 screen set value	
Dpdch[0].Power	BTS#1 screen set value (DPCCH#1)	
Dpdch[0].ChCode	BTS#1 screen set value (DPCCH#1)	
Dpdch[1].Power	BTS#1 screen set value (DPCCH#2)	
Dpdch[1].ChCode	BTS#1 screen set value (DPCCH#2)	
Dpdch[2].Power	BTS#1 screen set value (DPCCH#3)	
Dpdch[2].ChCode	BTS#1 screen set value (DPCCH#3)	

CphyRlSetupU_DPCH　　　/* for CPHY_RL_SETUP(D_DPCH) */

Parameter	Initial set value	Meaning
Offset	BTS#1 screen set value +7680+1024	BTS#1 screen set value +1024
ScrCode	BTS#1 screen set value	
SlotFormat	BTS#1 screen set value	
SymbolRate	BTS#1 screen set value	
ChCode	BTS#1 screen set value (DPCCH)	
Power	-	
NumOfDPDCH	BTS#1 screen set value (Fixed to 1)	
Dpdch[0].Power	BTS#1 screen set value (DPCCH#1)	
Dpdch[0].ChCode	BTS#1 screen set value (DPCCH#1)	

CphyRlSetupAWGN　　　/* for CPHY_RL_SETUP(AWGN) */

Parameter	Initial set value	Meaning
Power	BTS#1 screen set value	

2) Parameters for BTS#1

The following initial set values of parameters for BTS1 are the same as those of the basic parameters .

```
CphyRlSetupP_SCH_BTS1;
CphyRlSetupS_SCH_BTS1;
CphyRlSetupP_CPICH_BTS1;
CphyRlSetupP_CCPCH_BTS1;
CphyRlSetupS_CCPCH_BTS1;
CphyRlSetupPICH_BTS1;
CphyRlSetupAICH_BTS1;
CphyRlSetupPRACH_BTS1;
CphyRlSetupD_DPCH_BTS1;
CphyRlSetupU_DPCH_BTS1;
CphyRlSetupAWGN_BTS1;
```

3) Parameters for BTS#2

Replace the following initial set values of the parameters of BTS2 with those of the parameters for BTS#1, to obtain the screen set values of the parameters of BTS#2:

```
CphyRlSetupP_SCH_BTS2;
CphyRlSetupS_SCH_BTS2;
CphyRlSetupP_CPICH_BTS2;
CphyRlSetupP_CCPCH_BTS2;
CphyRlSetupS_CCPCH_BTS2;
CphyRlSetupPICH_BTS2;
CphyRlSetupAICH_BTS2;
CphyRlSetupPRACH_BTS2;
CphyRlSetupD_DPCH_BTS2;
CphyRlSetupU_DPCH_BTS2;
CphyRlSetupAWGN_BTS2;
```

4) Parameters for BTS#3

Replace the following initial set values of the parameters of BTS3 with those of the parameters for BTS#1, to obtain the screen set values of the parameters of BTS#3:

```
CphyRlSetupP_SCH_BTS3;
CphyRlSetupS_SCH_BTS3;
CphyRlSetupP_CPICH_BTS3;
CphyRlSetupP_CCPCH_BTS3;
CphyRlSetupS_CCPCH_BTS3;
CphyRlSetupPICH_BTS3;
CphyRlSetupAICH_BTS3;
CphyRlSetupPRACH_BTS3;
CphyRlSetupD_DPCH_BTS3;
CphyRlSetupU_DPCH_BTS3;
CphyRlSetupAWGN_BTS3;
```

A.2.3.2.5 Variables for CphyAttenuator()

Values right after executing SimulatorStart() are as follows:

```
CurrentTxAttenuator;          Setting values on BTS#1 Tx Attenuator screen
CurrentRxRefPower;            Setting value on BTS#1 Rx Reference screen
CurrentTxAttenuatorBTS1       Setting value on BTS#1 Tx Attenuator screen
                              BTS#1 ( same value as CurrentTxattenuator)
CurrentRxRefPowerBS1;         Setting value on BTS#1 Tx Attenuator screen
                              BTS#1(same value as CurrentRxRefPowerBS1)
CurrentTxAttenuatorBTS2;      Setting value on BTS#2 Tx Attenuator screen
CurrentRxRefPowerBTS2;        Setting value on BTS#2 Tx Attenuator screen
CurrentTxAttenuatorBTS3;      Setting value on BTS#3 Tx Attenuator screen
CurrentRxRefPowerBTS3;        Setting value on BTS#3 Tx Attenuator screen
```

A.2.3.2.6 Variables for SndMessage() and RcvMessage()

Values right after executing SimulatorStart() are as follows:

```
SIB_PCS =0;
SIB_REP =0;
PageIndicator =0;
PageNp =0;
DRXCycleLength =0;
PagingBlock =0;
RlcMUI =0;
RlcCNF =0;
```

A.2.3.2.7 Variables for CteConnect()

Values right after executing SimulatorStart() are as follows:

```
TeActTime =ACTIVATE_NOW ( 0xFFFF );
TeTtiCounter =0;
```

A.2.3.2.8 Variables for CphyRlSetupMtgp( )

1) Basic parameter CphyRlSetupMtgpD_DPCH /* for CPHY_RL_SETUP_MTGP(D_DPCH) */

Parameter	Initial set value	Meaning
Offset	BTS#1 screen set value +7680	Offset= screen set value
ScrCode	BTS#1 screen set value	
SlotFormat	BTS#1 screen set value	
SymbolRate	Value from Slot Format	
ChCode	BTS#1 screen set value (DPCCH)	
Power	BTS#1 screen set value (DPCCH)	
NumOfDPDCH	BTS#1 screen set value	
Dpdch[0].Power	BTS#1 screen set value (DPCCH#1)	
Dpdch[0].ChCode	BTS#1 screen set value (DPCCH#1)	
Dpdch[1].Power	BTS#1 screen set value (DPCCH#2)	
Dpdch[1].ChCode	BTS#1 screen set value (DPCCH#2)	
Dpdch[2].Power	BTS#1 screen set value (DPCCH#3)	
Dpdch[2].ChCode	BTS#1 screen set value (DPCCH#3)	

CphyRlSetupMtgpU_DPCH /* for CPHY_RL_SETUP_MTGP(U_DPCH) */

Parameter	Initial set value	Meaning
Offset	BTS#1 screen set value +7680+1024	BTS#1 screen set value +1024
ScrCode	BTS#1 screen set value	
SlotFormat	BTS#1 screen set value	
SymbolRate	BTS#1 screen set value	
ChCode	BTS#1 screen set value (DPCCH)	
Power	-	
NumOfDPDCH	BTS#1 screen set value (Fixed to 1)	
Dpdch[0].Power	BTS#1 screen set value (DPCCH#1)	
Dpdch[0].ChCode	BTS#1 screen set value (DPCCH#1)	

2) Parameters for BTS#1

The following initial set values of parameters for BTS1 are the same as those of the basic parameters .

```
CphyRlSetupMtgpD_DPCH_BTS1;
CphyRlSetupMtgpU_DPCH_BTS1;
```

3) Parameters for BTS#2

Replace the following initial set values of the parameters of BTS2 with those of the parameters for BTS#1, to obtain the screen set values of the parameters of BTS#2:

```
CphyRlSetupMtgpD_DPCH_BTS2;
CphyRlSetupMtgpU_DPCH_BTS2;
```

4) Parameters for BTS#3

Replace the following initial set values of the parameters of BTS3 with those of the parameters for BTS#1, to obtain the screen set values of the parameters of BTS#3:

```
CphyRlSetupMtgpD_DPCH_BTS3;
CphyRlSetupMtgpU_DPCH_BTS3;
```

A.2.3.2.9 Variables for CteConfig( )

CteConfigCSD

Parameter	Initial set value	Meaning
TeType	TE_TYPE_CSD	TE Interface Type
Rate	0	-
TTI	4	TTI
NumOfTB	1	Number of Transport Block
TBS	576	Transport Block Size
Frame	RLC_TR_DATA_REQ	Primitive to submit to the lower layer
Layer	RLC	Lower layer for TE
Data[1024]	all 0	-

(CteConfigCSD.)CsdConfPar

Parameter	Initial set value	Meaning
ConElement	CSD_CE_NT	Non transparent
SyncMode	CSD_SYNCMD_ASYNC	Asynchronous
InfoTransCap	CSD_ITC_AUDIO	3.1KHz Audio
UserInfoL2P	CSD_UIL2P_OUTBAND	
SerialRxBuff	100	
Waiur	CSD_WAIUR_14_4	Wanted Air Interface User Rate
AcptChCoding	CSD_ACC_14_4	Acceptable Channel Coding
UpMaxTchNo	1	Uplink Max. TCH number
DownMaxTchNo	1	Downlink Max. TCH number
Uimi	1	UIMI

Note)The same value is needed in UpMaxTchNo and DownMaxTchNo.

(CteConfigCSD.CsdConfPar.)RlpConfPar

Parameter	Initial set value	Meaning
TTI	4	-
ProtoType	RLP_PROTO_UMTS	Radio System Selection
Version	RLP_VERSION_2	RLP Version

Note)
The values above have already set in WCDMA_CSD14k4_without_Integrity.c
that is a scenario attached in control software.
When you use W-CDMA CSD functionality, please set the same values as :
1)above,
2)WCDMA_CSD28k8_without_Integrity.c or
3)WCDMA_CSD57k6_without_Integrity.c
in RLP_PAR, CSD_PAR and CTE_CONFIG_PAR.
MD8480 behaviour is not guaranteed if other values are used.

A.2.3.2.10 Variables for CbmcConfig()

For initial set values, refer to “InitParam_CbmcConfig()” in the attached “IntializeParameter.c”.
.

A.3 Range of Layer 1 Parameter

This chapter contains a range of Layer1 Parameter that can be operated on MD8480A/B Signalling Tester. All of Layer1 Parameter Set/Item numbers described in this chapter are based on 3GPP TS34.108.

Symbols in "Confirmation State" indicated in Correspondence Tables are intended to stand for as follows:

A	The operation has been confirmed. Operation for Rate Matching Attribute has been confirmed on a certain combination. Contact us in the event that it does not operate properly.
B	The operation has not been confirmed by us. Contact us in the event that it does not operate properly.
C	The operation has not been confirmed. It is not scheduled to be confirmed in the future.

A.3.1 DPCH

A.3.1.1 Correspondence to ” 3GPP TS34.108”

Refer to 6.10.2.4.1 Chapter X for the details of parameter. “Xs” of Chapter number are correspondent to “X” in the Table below.

X	Radio Interface Combination	Confirmation State	Remarks
1	Stand-alone UL:1.7 DL:1.7 kbps SRBs for DCCH	B	
2	Stand-alone UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
3	Stand-alone UL:13.6 DL:13.6 kbps SRBs for DCCH	B	
4	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
4a	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL:(12.2 7.95 5.9 4.75) kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	For support service
5	Conversational / speech / UL:10.2 DL:10.2 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
5a	Conversational / speech / UL(10.2 6.7 5.9 4.75) DL:(10.2 6.7 5.9 4.75) kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	For support service
6	Conversational / speech / UL:7.95 DL:7.95 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
7	Conversational / speech / UL:7.4 DL:7.4 kbps / CS RAB+ UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
7a	Conversational / speech / UL(7.4 6.7 5.9 4.75) DL:(7.4 6.7 5.9 4.75) kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	For support service
8	Conversational / speech / UL:6.7 DL:6.7 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
9	Conversational / speech / UL:5.9 DL:5.9 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
10	Conversational / speech / UL:5.15 DL:5.15 kbps / CS RAB + UL:1.7 DL:1.7 kbps SRBs for DCCH	C	
11	Conversational / speech / UL:4.75 DL:4.75 kbps / CS RAB + UL:1.7 DL:1.7 kbps SRBs for DCCH	C	
12	Conversational / unknown / UL:28.8 DL:28.8 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
13	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
14	Conversational / unknown / UL:32 DL:32 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
15	Streaming / unknown / UL:14.4/DL:14.4 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
16	Streaming / unknown / UL:28.8/DL:28.8 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
17	Streaming / unknown / UL:57.6/DL:57.6 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
18	Streaming / unknown / UL:0 DL:64 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
19	Streaming / unknown / UL:64 DL:0 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
20	Streaming / unknown / UL:0 DL:128 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
21	Streaming / unknown / UL:128 DL:0 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	

22	Streaming / unknown / UL:0 DL:384 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	C	
23	Interactive or background / UL:32 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	UL TTI=10: B DL TC: B
23a	Interactive or background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
23b	Interactive or background / UL:16 DL:16 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
23c	Interactive or background / UL:32(40msTTI) DL:32(40msTTI) kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
23d	Interactive or background / UL:32(20msTTI) DL:32(20msTTI) kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
24	Interactive or background / UL:64 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	DL TC: B
25	Interactive or background / UL:32 DL: 64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	UL TTI=10: B
26	Interactive or background / UL:64 DL: 64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
27	Interactive or background / UL:64 DL:128 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
28	Interactive or background / UL:128 DL:128 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
29	Interactive or background / UL:64 DL:144 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
30	Interactive or background / UL:144 DL:144 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
31	Interactive or background / UL:64 DL:256 kbps / PS RAB + UL:3.4 DL: 3.4 kbps SRBs for DCCH	B	
32	Interactive or background / UL:64 DL:384 kbps / PS RAB + UL:3.4 DL: 3.4 kbps SRBs for DCCH	A	DL: combination of B • TTI=10 single code • TTI=20 multi code
33	Interactive or background / UL:128 DL:384 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	DL: combination of B • TTI=10 single code • TTI=20 multi code
34	Interactive or background / UL:384 DL:384 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	DL: combination of B • TTI=10 single code • TTI=20 multi code
35	Interactive or background / UL:64 DL:2048 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	C	
36	Interactive or background / UL:128 DL:2048 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	C	
37	Interactive or background / UL:384 DL:2048 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	C	
38	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:32 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	UL TTI=10: B DL TC: B
38a	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:0 DL:0 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	

38 b	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 c	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:32 DL:32 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	
38 d	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:64 DL:64 kbps / PS RAB + Interactive or background / UL:64 DL:64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 e	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL (12.2 7.95 5.9 4.75) kbps / CS RAB + Interactive or background / UL:0 DL:0 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 f	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL (12.2 7.95 5.9 4.75) kbps / CS RAB + Interactive or background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 g	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL (12.2 7.95 5.9 4.75) kbps / CS RAB + Interactive or background / UL:16 DL:16 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 h	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL (12.2 7.95 5.9 4.75) kbps / CS RAB + Interactive or background / UL:32 DL:32 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 i	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL (12.2 7.95 5.9 4.75) kbps / CS RAB + Interactive or background / UL:64 DL:64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 j	Conversational / speech / UL(12.2 7.95 5.9 4.75) DL (12.2 7.95 5.9 4.75) kbps / CS RAB + Interactive or background / UL:64 DL:128 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
38 k	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:32 DL:32 kbps / PS RAB + Interactive or background / UL:32 DL:32 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH (L1 multiplexing).	B	
39	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:32 DL:64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
40	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:64 DL:64 kbps / PS RAB + UL:3.4 DL: 3.4 kbps SRBs for DCCH	B	

41	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:64 DL:128 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
42	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:64 DL:256 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
43	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:64 DL:384 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	DL: combination of B • TTI=10 single code • TTI=20 multi code
44	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Interactive or background / UL:128 DL:2048 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	C	
45	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Streaming / unknown / UL:57.6 DL:57.6 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
46	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Streaming / unknown / UL:0 DL:64 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
47	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Streaming / unknown / UL:0 DL:128 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
48	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Streaming / unknown / UL:0 DL:384 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	C	
49	Conversational / speech / UL:12.2 DL:12.2 kbps / CS RAB + Conversational / unknown / UL:64 DL:64 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	UL/DL64K TTI=20: B
50	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + Conversational / unknown / UL:64 DL:64 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	A	UL/DL64K TTI=20: B
51	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + Interactive or background / UL:64 DL:64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
51a	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + Interactive or background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
51b	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + Interactive or background / UL:16 DL:64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
52	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + Interactive or background / UL:64 DL:128 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	

53	Conversational / unknown / UL:64 DL:64 kbps / CS RAB + Interactive or background / UL:128 DL:128 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
54	Interactive or /background / UL:64 kbps DL:128 kbps / PS RAB + Streaming / unknown / UL:0 DL:64 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
55	Interactive or /background / UL:64 kbps DL:128 kbps / PS RAB + Streaming / unknown / UL:0 DL:128 kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
56	Interactive or /background / UL:8 kbps DL:8 kbps / PS RAB + Interactive or /background / UL:8 kbps DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
57	Interactive or /background / UL:64 kbps DL:64 kbps / PS RAB + Interactive or /background / UL:64 kbps DL:64 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
58	Streaming / unknown / UL:16 DL:64 kbps / PS RAB + Interactive or /background / UL:8 kbps DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
58a	Streaming / unknown / UL:16 DL:128 kbps / PS RAB + Interactive or background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
59	Conversational / Speech / UL:42.8 DL:42.8 kbps / PS RAB + Interactive or background / UL:16 DL:16 kbps / PS RAB + Interactive or background / UL:16 DL:16 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH (REL-5).	B	
60	Conversational / Speech / UL:42.8 DL:42.8 kbps / PS RAB + Interactive or background / UL:16 DL:16 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH (REL-5).	B	
61	Conversational / unknown / UL:8 DL:8 kbps / PS RAB + Interactive or Background / UL:8 DL:8 kbps / PS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH	B	
62	Conversational / speech / UL:(12.65 8.85 6.6) DL:(12.65 8.85 6.6) kbps / CS RAB + UL:3.4 DL:3.4 kbps SRBs for DCCH + DL:0.15 kbps SRB#5 for DCCH (REL-5).	B	
63	Interactive or background / UL:64 DL:768 kbps / PS RAB + UL:3.4 DL: 3.4 kbps SRBs for DCCH (REL-5).	C	

A.3.1.2 Range of Layer1 Downlink correspondence Parameter

(1) The following items are pursuant to 3GPP Standard TS25306.

(*1) TDD, PDSCH are not correspondent.

(*2) Functions on Mobile Station

Reference combination of UE Radio Access capability parameters in DL		
Transport channel parameters		
Maximum sum of number of bits of all transport blocks being transmitted at an arbitrary time instant		8064
Maximum sum of number of bits of all convolutionally coded transport blocks being transmitted at an arbitrary time instant		640
Maximum sum of number of bits of all turbo coded transport blocks being transmitted at an arbitrary time instant		8064
Maximum number of simultaneous transport channels		8
Maximum number of simultaneous CCTrCH (FDD)		1
Maximum number of simultaneous CCTrCH (TDD)		(*1)
Maximum total number of transport blocks transmitted within TTIs that start at the same time		96
Maximum number of TFC in the TFCS		128
Maximum number of TF		64
Support for turbo encoding		Yes
Physical channel parameters (FDD)		
Maximum number of DPCH/PDSCH codes to be simultaneously	DPCH	3
	PDSCH	(*1)
Maximum number of physical channel bits transmitted in any 10 ms interval (DPCH, PDSCH, S-CCPCH).	DPCH	9600
	PDSCH	(*1)
	S-CCPCH	4800
Support for SF 512		No
Support of PDSCH		No
Maximum number of simultaneous S-CCPCH radio links		2
Support of dedicated pilots for channel estimation		(*2)
Physical channel parameters (TDD)		
Maximum number of timeslots per frame		(*1)
Maximum number of physical channels per frame		(*1)
Minimum SF		(*1)
Support of PDSCH		(*1)
Maximum number of physical channels per timeslot		(*1)

(2) Range of the other feasible parameter

Items	Feasible range
Input bit length for Turbo Coding	40, 48, 64, 128, 144, 192, 256, 336, 352, 356, 370, 376, 384, 512, 576, 592, 656, 672, 704, 712, 740, 768, 1008, 1024, 1056, 1068, 1152, 1176, 1184, 1296, 1312, 1344, 1360, 1408, 1424, 1480, 1688, 1728, 1776, 2304, 2368, 2624, 2688, 2816, 2896, 2960, 3168, 3352, 3360, 3520, 3584, 3700, 3755, 3848, 3856, 4032, 4224, 4440, 4480, 4506, 4694, 4704, 4928, 5016, 5114
Total maximum bit length of Transport Block for entire TrCH	15508[bit]
A condition to use TTI=80[ms]	Only when the number of TrCH is 1.

A.3.1.3 Range of Layer1 Uplink correspondence Parameter

(1) The following items are pursuant to 3GPP Standard TS25306.

(*1)　TDD and PCPCH are not corresponded.
(*2)　Functions on Mobile Station

Reference combination of UE Radio Access capability parameters in UL	
Transport channel parameters	
Maximum sum of number of bits of all transport blocks being received at an arbitrary time instant	8064
Maximum sum of number of bits of all convolutionally coded transport blocks being received at an arbitrary time instant	640
Maximum sum of number of bits of all turbo coded transport blocks being received at an arbitrary time instant	8064
Maximum number of simultaneous transport channels	8
Maximum number of simultaneous CCTrCH(TDD only)	(*1)
Maximum total number of transport blocks received within TTIs that end at the same time	32
Maximum number of TFC in the TFCS	128
Maximum number of TF	32
Support for turbo decoding	Yes
Physical channel parameters (FDD)	
Maximum number of DPDCH bits received per 10 ms	9600
Simultaneous reception of SCCPCH and DPCH	(*2)
Simultaneous transmission of SCCPCH, DPCH and PDSCH	(*1)
Support of PCPCH	No
Physical channel parameters (TDD)	
Maximum Number of timeslots per frame	(*1)
Maximum number of physical channels per timeslot	(*1)
Minimum SF	(*1)
Support of PUSCH	(*1)

(2) Range of the other feasible parameter

Items	Feasible range
Maximum bit length for Transport Block of all TrCHs	15508[bit]
A condition to use TTI=80[ms]	Only when the number of TrCH is 1.
Uplink Slot Format	Slot Format#0, #2, or　#5
Downlink Slot Format	Slot Format#0~#15

A.3.2 PDSCH

PDSCH is not corresponded.

A.3.3 S-CCPCH

Refer to 6.10.2.4.3.Chapter X for the details of parameter. “X” of Chapter number is correspondent to numbers in “X” in the Table below.

X	Radio Interface Combination	Confirmation state	Remarks
1	Stand-alone signalling RB for PCCH	A	
2	Interactive/Background 32 kbps PS RAB + SRBs for CCCH + SRB for DCCH + SRB for BCCH	A	
3	Interactive/Background 32 kbps RAB + SRB for PCCH + SRB for CCCH + SRB for DCCH + SRB for BCCH	A	

A.3.4 PRACH

Refer to 6.10.2.4.4.Chapter X for the details of parameter. “X” of Chapter number is correspondent to numbers in “X” in the Table below.

X	Radio Interface Combination	Confirmation state	Remarks
1	Interactive/Background 32 kbps PS RAB + SRB for CCCH + SRB for DCCH	A	

B. DETAILS OF HARIKIRI (PHYSICAL LAYER) TEST

This chapter describes the details of the Harikiri (physical layer) test.

B.1 Harikiri (Physical Layer) Test List

Tables B-1 and 2 show the list of functions that can be tested in the Harikiri (physical layer) test. For the test items other than those described in this material and the scenarios other than the attached sample scenarios, operations have not been verified.

Table B-1 List of transmit checks of layer 1 downlink signals

Physical Channel	Contents of services	Logical Channel	Symbol Rate	Compressed Mode Method
P_SCH		-	15ksps	-
S_SCH		-	15ksps	-
P_SCH		-	15ksps	-
P_CCPCH		BCCH×1	15ksps	-
S_CCPCH		PCCH×1+(CCCH×1+DCCH×4+BCCH×1) +DTCH×1	60ksps	-
PICH		-	15ksps	-
AICH	Positive/Negative, Transmission Timing specifiable	-	15ksps	-
DPCH	Stand-Alone DCCH	DCCH×1	15ksps	-
	Stand-Alone DCCH (TTI=80[ms])	DCCH×1	7.5ksps	-
	ISDN UDI(64kbps)	DTCH(UDI) ×1+DCCH×1	120ksps	-
	AMR Speech	DTCH(AMR) ×3+DCCH×1	30ksps	-
	DCH 8Channel	DTCH×8	240ksps	-
	Packet 8[kbps] (Convolutional Coding)	DTCH(Packet) ×1+DCCH×1	30ksps	-
	Packet 32[kbps](TTI=20[ms])	DTCH(Packet) ×1+DCCH×1	60ksps	-
	Packet 64[kbps](TTI=10[ms])	DTCH(Packet) ×1+DCCH×1	120ksps	Higher Layer Scheduling Higher Layer Scheduling×2
	Packet 128[kbps]	DTCH(Packet) ×1+ DCCH×1	240ksps	Higher Layer Scheduling Higher Layer Scheduling×6 Higher Layer Scheduling +SF/2×5
	Packet 384[kbps]	DTCH(Packet) ×1+DCCH×1	480ksps	-
	Audio Visual32[ksps]	DTCH(AV) ×1+DCCH×1	60ksps	-
	Audio Visual64[ksps]	DTCH(AV) ×1+DCCH×1	120ksps	-
	AMR Speech + Packet32[kbps] (TTI=10ms)	DTCH(AMR) ×3+DTCH(Packet) ×1 +DCCH×1	120ksps	-
	AMR Speech + Packet64[kbps] (TTI=10ms)	DTCH(AMR) ×3+DTCH(Packet) ×1 +DCCH×1	120ksps	-
	AMR Speech + Packet384[kbps] (TTI=20ms)	DTCH(AMR) ×3+DTCH(Packet) ×1 +DCCH×1	480ksps	-
	AMR Speech + ISDN UDI	DTCH(AMR) ×3+DTCH(UDI) ×1 +DCCH×1	120ksps	-
	RMC 12.2[kbps]	DTCH(RMC) ×1+DCCH×1	30ksps	- SF/2 Puncturing SF/2×2 Puncturing×2
	RMC 64[kbps]	DTCH(RMC) ×1+DCCH×1	120ksps	-
	RMC 144[kbps]	DTCH(RMC) ×1+DCCH×1	240ksps	SF/2 Puncturing SF/2×6 Puncturing +SF/2×5
	RMC 384[kbps]	DTCH(RMC) ×1+DCCH×1	480ksps	-

	RMC BTFD	DTCH(RMC) ×1+DCCH×1	30ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)	DTCH(UDI) ×4+DCCH×1	480ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)	DTCH(UDI)x5+DCCHx1	480ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)	DTCH(UDI)x6+DCCHx1	480ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 64[kbps]	DTCH(UDI)x5+DTCH(Packet)x1 +DCCHx1	480ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 128[kbps]	DTCH(UDI)x4+DTCH(Packet)x1 +DCCHx1	480ksps	-

Table B-2 List of receive checks of layer 1 uplink signals

Physical Channel	Contents of services	Logical Channel	Symbol Rate	Compressed Mode Method
PRACH	TTI=20msec		60ksps	-
DPCH	Stand-Alone DCCH	DCCH×1	15sps	-
	Stand-Alone DCCH (TTI=80[ms])	DCCH×1	15ksps	-
	ISDN UDI(64kbps)	DTCH(UDI) ×1+DCCH×1	240ksps	-
	AMR Speech	DTCH(AMR) ×3+DCCH×1	60ksps	-
	DCH 8Channel	DTCH×8	240ksps	-
	Packet 32kbps] (Convolutional Coding)	DTCH(Packet) ×1+DCCH×1	120ksps	- Higher Layer Scheduling×2
	Packet 32[kbps]	DTCH(Packet) (TTI=20[ms])×1+DCCH×1	120ksps	-
	Packet 64[kbps]	DTCH(Packet) (TTI=10[ms])×1+DCCH×1	240ksps	Higher Layer Scheduling Higher Layer Scheduling×2
	Packet 128[kbps]	DTCH(Packet) ×1+ DCCH×1	480ksps	Higher Layer Scheduling Higher Layer Scheduling×6 Higher Layer Scheduling +SF/2×5
	Audio Visual32[ksps]	DTCH(AV) ×1+DCCH×1	120ksps	-
	Audio Visual64[ksps]	DTCH(AV) ×1+DCCH×1	240ksps	-
	AMR Speech + Packet32[kbps] (TTI=10ms)	DTCH(AMR) ×3+DTCH(Packet) ×1 +DCCH×1	240ksps	-
	AMR Speech + Packet64[kbps] (TTI=10ms)	DTCH(AMR) ×3+DTCH(Packet) ×1 +DCCH×1	240ksps	-
	AMR Speech + Packet384[kbps] (TTI=20ms)	DTCH(AMR) ×3+DTCH(Packet) ×1 +DCCH×1	960ksps	-
	AMR Speech + ISDN UDI	DTCH(AMR) ×3+DTCH(UDI) ×1 +DCCH×1	240ksps	-
	RMC 12.2[kbps]	DTCH(RMC) ×1+DCCH×1	60ksps	-
	RMC 64[kbps]	DTCH(RMC) ×1+DCCH×1	240ksps	-
	RMC 144[kbps]	DTCH(RMC) ×1+DCCH×1	480ksps	-
	RMC 384[kbps]	DTCH(RMC) ×1+DCCH×1	960ksps	-
	RMC BTFD	DTCH(RMC) ×1+DCCH×1	60ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)	DTCH(UDI)x4+DCCHx1	960ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)	DTCH(UDI)x5+DCCHx1	960ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)	DTCH(UDI)x6+DCCHx1	960ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 64[kbps]	DTCH(UDI)x5+DTCH(Packet)x1 +DCCHx1	960ksps	-
	ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 128[kbps]	DTCH(UDI)x4+DTCH(Packet)x1 +DCCHx1	960ksps	-

B.2 OPERATION IN EACH SCENARIO

B.2.1 Start_xxx.dl

- Used to start the Signalling Tester
- P_SCH, S_SCH, P_CPICH, and P_CCPCH are transmitted automatically.
- By selecting the "ACTIVATE" button, data of individual physical channels can be received and transmitted.
- Since Start_xxx.dl is a scenario for testing layer 1, it transmits the data of which MAC header is removed.

B.2.2 xxx_Data(xxx).dll

- Used when transmitting the Transport Channel (referred to as TrCH, hereafter) data in downlink.
- Transport Block length, Transport Block and others for each TrCH can be selected by selection button.
- In "Sending Pattern?," select transmission pattern from the fixed data transmissions; "Fix Data," "PN9" or "Loop Back."
- Fix Data: Value specified in DCH_Data(DCCH).dll is set
- PN9: Pseudo random pattern of PN9 is transmitted
- Loop Back: Loops back DPCH of each Trch to Downlink

  When no description is mentioned, Transport Block is transmitted in each specified TTI.

B.3 LAYER 1 OPERATION PARAMETER WHEN THE ATTACHED SCENARIOA IS USED

The following should be noted when reading this chapter.

1. Symbol Rate is a parameter for setting the maximum symbol rate that can be received. Therefore, a value below the symbol rate set in the Signalling Tester is also supported.
   Example: When Symbol Rate is set to 60ksps, a value below 60ksps such as 15sps or 30ksps is also supported.
2. When "CRC, bit" is "_", it indicates that the bit length of CRC is 0, or "Transmits/Receives data without CRC.
3. Difference between 0×♣♣ and ♦×0

	There is no transport channel to be provided with CRC	There are transport channels to be provided with CRC
0×♣♣	CRC needs not to be provided.	CRC needs not to be provided.
♦×0	CRC needs not to be provided.	CRC needs to be provided.

4. The order of transport channels that can be transmitted and received with a sample scenario is the same as the order of them described in the tables in this section.

B.3.1 Procedure and Operation Common to Uplink

All uplink Harikiri (physical layer) tests should be conducted in the following procedure.

1) On the BTS Setup screen of Control Software, set the parameters of the PhCH to receive data. (See "Uplink settings" for each channel described later.)
2) Select Start_xxx.dll in the Scenario File Select window and click "OK."
3) Clicking the START button will cause the Signalling Tester to operate and send P_SCH, S_SCH, P_CPICH, and P_CCPCH automatically. In addition, the selection button with the title "XXX ACTIVATE?" appears. (XXX differs depending on the scenario used.)
4) Selecting "ACTIVATE" enables each TFCI to be received.
5) When uplink pilots can be received for eight or more slots continuously, demodulated or decoded DCH data is displayed on the trace screen. The number of permissible error bits in Pilots is one bit per pilot of each slot. The permissible uplink timing is ±200 chips.
6) CRC can be checked on the display of Opt2 for trace data for each TrCH. When Opt2 is 0, CRC=OK; when Opt2 is other than 0, CRC=NG.

Details of opt2 (16 [bits])

Bit15		bit14	……	bit2	bit1	bit0
0	No CRC error	CRC result for TB#14	(0:CRC=OK, 1:CRC=NG)	CRC result for TB#2	CRC result for TB#1	CRC result for TB#0
1	CRC error exists					

* Note: If TFCI is out of the receiving range, Opt2 is not displayed on the trace screen.
* If the number of transport blocks is 0 or the transport block length is 0, Opt2 is not displayed on the trace screen even if TFCI is within the receiving range.
* TFS indicates the Transport Block Size and the number of Transport Blocks. In the Signalling Tester, operation of TFCS is not verified for values other than those below.
  The Transport Block Size can be checked by clicking the arrow of the data to be checked on the Trace screen. (The data is displayed in "Message Data.")
* If Channelization Code is set to SF/4 (16 for AMR), data can be received even if TFCI is changed.

B.3.2 Procedure and Operation Common to Downlink

All downlink Harikiri (physical layer) tests should be conducted in the following procedure.

1) On the BTS Setup screen of the MX848000A, set the parameters of the PhCH to transmit data. (See "Downlink settings.")

2) Select Start_xxx.dll in the Scenario File Select window and click "OK."

3) Clicking the START button will cause the Signalling Tester to operate and send P_SCH, S_SCH, P_CPICH, P_CCPCH automatically. In addition, the selection button with the title "XXX ACTIVATE?" appears. (XXX differs depending on the scenario used.)

4) Selecting "ACTIVATE" causes S-CCCPCH or DPCH to be transmitted at TFCI=0 and terminates the execution of "Start_xxx.dll."

5) To transmit data to the Transport Block, select xxx_Data(xxx).dll in accordance with TFCI to be transmitted in the Scenario File Select window and set the transmitting data for each TrCH (See "Transmit procedure for each TFCI.")
   * In the sample scenario for the Harikiri (physical layer) test, the TX Diversity transmit is not supported.
   * TFS indicates the Transport Block Size and the number of Transport Blocks. In the Signalling Tester, operation of TFCS is not verified for values those below.
     The transport block size can be checked by clicking the arrow of the data to be checked on the Trace screen. (This data is displayed in "Message Data.")

B.3.3 P_SCH, S_SCH, P_CPICH, and P_CCPCH Settings

It applies all the scenarios of "Start_xxx.dll."

B.3.3.1 Settings

• P_SCH

SYMBOL RATE	15ksps
Scrambling Code	Value set on the GUI
Tx Power	Value set on the GUI

• S_SCH

SYMBOL RATE	15ksps
Scrambling Code	Value set on the GUI
Synchronization Code Group	Value set on the GUI
Tx Power	Value set on the GUI

• P_CPICH

SYMBOL RATE	15ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI

• P_CCPCH

SYMBOL RATE	15ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI

Transport Format Set			
PCCPCH	TrCH type		BCH
	TFS	TF0, bits	1×246
	TTI, ms		20
	Coding type		Convolutional Coding 1/2
	CRC, bit		16
	RM attribute		Invalid

transmit data: The pattern consisting of the following value coded with BCH is transmitted every 20m sec (TTI).

Transmit pattern 246 bits:SFN-Prime (11 bits) + 1010 (4 bits) + ALL0 (231b bits)

(SFN-Prime is transmitted every 20m sec in 2048 frame cycle from 0 to 2047.)

B.3.3.2 BCH data transmit procedure

After the completion of "Start_xxx.dll," select BCH_Data.dll on the Scenario File Select window and click "OK."

This changes the BCH transmit data. With this change, the value specified in BCH_Data.dll with SFN Prime overwritten on the initial 11 bits is transmitted repeatedly as transmit data.

B.3.4 S_CCPCH(PCH+FACH) Setting

Start_SCCPCH(PCH_FACH).dll is used.

B.3.4.1 Downlink settings

• S_CCPCH(PCH+FACH)

Slot Format(Symbol Rate)	Be sure to set to 8 (60ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

• PICH

Symbol Rate	15ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI

B.3.4.2 Transmittable TFCS, etc.

• S_CCPCH(PCH+FACH)

Transport Format Set					
SCCPCH	TrCH type		PCH	FACH1	FACH2
	TFS	TF0, bits	0×240	0×168	0×360
		TF1, bits	1×240	1×168	1×360
		TF2, bits	–	2×168	–
		TF3, bits	–	3×168	–
	TTI, ms		10	10	10
	Coding type		Convolutional Coding 1/2	Convolutional Coding 1/2	Turbo Coding
	CRC, bit		16	16	16
	RM attribute		230	220	130

Transport Format Combination Set			
TFCI	PCH	FACH1	FACH2
0	TF0	TF0	TF0
1	TF1	TF0	TF0
2	TF0	TF1	TF0
3	TF1	TF1	TF0
4	TF0	TF2	TF0
5	TF1	TF2	TF0
6	TF0	TF3	TF0
7	TF0	TF0	TF1
8	TF0	TF1	TF1

B.3.4.3 Transmittable TFCS, etc.

TFCI	PCH_Data(DTCH).dll	FACH_Data(CCCH).dll "FACH1:Num of Transport Block?"	FACH_Data(DTCH).dll
0	–	–	–
1	Run	–	–
2	–	Select "1"	–
3	Run	Select "1"	–
4	–	Select "2"	–
5	Run	Select "2"	–
6	–	Select "3"	–
7	–	–	Run
8	–	Select "1"	Run

When "Single" is selected for "Sending Mode?" when executing PCH_Data.dll, PCH is transmitted once in each execution. When "Continuous" is selected, PCH is transmitted every 10msec continuously.

Note:

Don't execute the combination without transmittable TFCI.
(Example: PCH_Data.dll and FACH_Data(DTCH).dll, etc.)

B.3.5 RACH receive and AICH transmit Settings

Start_SCCPCH(xxxx).dll is used as a scenario. When TTI=10msec, change RACH_TTI in Start_SCCPCH(xxx).c to 10. (The initial value is 20.)

B.3.5.1 PhCH setting

• PRACH

Symbol Rate (when TTI=20msec)	Be sure to set to 60ksps.
Symbol Rate (when TTI=10msec)	Be sure to set to 120ksps.
Slot Format	0
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A

B.3.5.2 TFCS, etc. that can be received by PRACH

Transport Format Set			
DPCH	TrCH type		RACH
	TFS	TF0,bits	1×168
		TF1,bits	1×360
	TTI,ms		10 or 20
	Coding type		Convolutional Coding1/2
	CRC, bit		16
	RM attribute		N/A

Transport Format Combination Set	
TFCI	RACH
0	TF0
1	TF1

ACIH is transmitted when the Preamble Part of PRACH is received correctly.

1) After the uplink signature is received correctly, the Control Part and the Message Part are received next. When receiving these two parts, demodulated or decoded RACH data is displayed on the trace screen if pilots can be received for all the slots. The number of permissible error bits in pilots is one bit per pilot of each slot. The permissible uplink timing is +-200 chips.

2) CRC can be checked on the display of Opt2 of trace data. When Opt2 is 0, CRC is OK; when Opt2 is other than 0, CRC is NG.

Note:

When TFCI is other than 0 and 1, Opt2 is not displayed on the trace screen.

B.3.6　Stand-Alone DCCH Setting

Start_SDCCH.dll is used.

B.3.6.1　Receiving Stand-Alone DCCH

B.3.6.1.1　Uplink settings

• DPCH(Stand Alone DCCH)

SLOT FORMAT	Be sure to set to 2 (15ksps).
Symbol Rate	Be sure to set to 15ksps.
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	1

B.3.6.1.2　Receivable TFCS, etc.

Transport Format Set			
DPCH	TrCH type		DCH
	TFS	TF0,bits	0×148
		TF1,bits	1×148
	TTI,ms		40
	Coding type		Convolutional Coding1/3
	CRC, bit		16
	RM attribute		N/A

Transport Format Combination Set	
TFCI	DCH
0	TF0
1	TF1

B.3.6.2 Transmitting Stand-Alone DCCH

B.3.6.2.1 Downlink settings

• DPCH(Stand-Alone DCCH)

SLOT FORMAT(SYMBOL RATE)	Be sure to set to 4 (15ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.6.2.2 Downlink settings

Transport Format Set			
DPCH	TrCH type		DCH
	TFS	TF0,bits	0×148
		TF1,bits	1×148
	TTI,ms		40
	Coding type		Convolutional Coding1/3
	CRC, bit		16
	RM attribute		N/A

Transport Format Combination Set	
TFCI	DCH
0	TF0
1	TF1

B.3.6.2.3 Transmittable TFCS, etc.

TFCI	DCH_Data(DCCH).dll
0	-
1	Select “40” for “TTI?”

B3.7　Stand-Alone DCCH(TTI=80m) Setting

Start_SDCCH(TTI80m).dll is used.

B.3.7.1 Receiving Stand-Alone DCCH

B.3.7.1.1Uplink setting

• DPCH(Stand-Alone DCCH(TTI=80m))

SLOT FORMAT	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 15ksps
Scrambling Code	Value set on GUI
Channelization Code	Value set on GUI
DTX position	N/A
Puncturing Limit	1

B.3.7.1.2 Receivable TFCS etc.

Transport Format Set			
DPCH	TrCH type		DCH
	TFS	TF0,bits	0x148
		TF1,bits	1x148
	TTI,ms		80
	Coding type		Convolutional Coding1/3
	CRC, bit		16
	RM attribute		N/A

Transport Format Combination Set	
TFCI	DCH
0	TF0
1	TF1

B.3.7.2 Transmitting Stand-Alone DCCH(TTI=80m)

B.3.7.2.1 Downlink setting

• DPCH(Stand-Alone DCCH(TTI=80m))

SLOT FORMAT(SYMBOL RATE)	Be sure to set to 0(7.5ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.7.2.2 Transmittable TFCS etc.

Transport Format Set			
DPCH	TrCH type		DCH
	TFS	TF0,bits	0x148
		TF1,bits	1x148
	TTI,ms		80
	Coding type		Convolutional Coding1/3
	CRC, bit		16
	RM attribute		N/A

Transport Format Combination Set	
TFCI	DCH
0	TF0
1	TF1

B.3.7.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(DCCH).dll
0	-
1	Select “80” for “TTI?”

B.3.8 UDI+DCCH Setting

Start_UDI.dll is used.

B.3.8.1 Receiving UDI+DCCH

B.3.8.1.1 Uplink setting

• DPCH(UDI+DCCH)

SLOT FORMAT	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 240ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	1

B.3.8.1.2 Receivable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x640	0x148
		TF1,bits	4x640	1x148
	TTI,ms		40	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		170	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.8.2 Transmitting UDI+DCCH

B.3.8.2.1 Downlink setting

• DPCH(UDI+DCCH)

SLOT FORMAT(SYMBOL RATE)	Be sure to set to 13(120ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.8.2.2 Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x640	0x148
		TF1,bits	4x640	1x148
	TTI,ms		40	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		170	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.8.2.3 Transmit procedures for each TFCI

TFCI	DCH_Data(UDI).dll	DCH_Data(DCCH).dll
0	–	–
1	–	Select “40” for “TTI?”
2	execution	–
3	execution	Select “40” for “TTI?”

B.3.9　AMR Voice(12.2[kbps])+DCCH Setting

Start_AMR.dll is used.

B.3.9.1　Receiving AMR+DCCH

B.3.9.1.1　Uplink setting

• DPCH(Voice)

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 60ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	0.88

B.3.9.1.2　Receivable TFCS etc.

Transport Format Set						
DPCH	TrCH type		DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	0x103	0x60	0x148
		TF1,bits	1x39	1x103	1x60	1x148
		TF2,bits	1x81	-	-	-
	TTI,ms		20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Convolutional Coding1/3
	CRC, bit		12	-	-	16
	RM attribute		200	190	235	160

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF1
2	TF1	TF0	TF0	TF0
3	TF1	TF0	TF0	TF1
4	TF2	TF1	TF1	TF0
5	TF2	TF1	TF1	TF1

B.3.9.2　Transmitting AMR+DCCH

B.3.9.2.1　Downlink setting

• DPCH(Voice)

Slot Format(Symbol Rate)	Be sure to set to 8(30ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

B.3.9.2.2　Transmittable TFCS etc.

Transport Format Set						
DPCH	TrCH type		DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	0x103	0x60	0x148
		TF1,BITS	1x39	1x103	1x60	1x148
		TF2,bits	1x81	-	-	-
	TTI,ms		20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Convolutional Coding1/3
	CRC, bit		12	-	-	16
	RM attribute		200	190	235	160

Transport Format Combination Set				
TFCI	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF1
2	TF1	TF0	TF0	TF0
3	TF1	TF0	TF0	TF1
4	TF2	TF1	TF1	TF0
5	TF2	TF1	TF1	TF1

B.3.9.2.3　Transmit procedure for each TFCI

TFCI	DCH_Data(AMR).dll		DCH_Data(DCCH).dll
0	–	–	–
1	–	–	Select “40” for “TTI?”
2	Select “39bit “ after selecting A for “AMR?”	–	–
3	Select “39bit “ after selecting Afor “AMR?”	–	Select “40” for “TTI?”
4	Select “81bit “ after selecting Afor “AMR?”	Select B for “AMR?”	–
5	Select “81bit “ after selecting A for “AMR?”	Select B for “AMR?”	Select “40” for “TTI?”

B.3.10 DCH 8Channel Setting

Start_DCH8CH_1.dll is used.

B3.10.1 Receiving (DCH 8Channel)

B3.10.1.1 Uplink setting

• DPCH(DCH 8Channel)

SLOT FORMAT	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 240ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	1

B3.10.1.2 Receivable TFCS etc.

Transport Format Set										
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	1x0	1x0	1x0	1x0	1x0	0x800	1x0
		TF1,bits	1x20	1x27	1x33	1x20	1x27	1x33	1x800	1x40
		TF2,bits	1x40	1x54	1x66	1x40	1x54	1x66	1x1600	1x80
		TF3,bits	1x80	1x81	1x99	1x80	1x81	1x99	-	1x160
	TTI,ms		20	20	20	40	20	20	40	40
	Coding type		Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Convolu-tional Coding 1/2	Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Turbo Coding	Convolu-tional Coding 1/3
	CRC, bit		24	24	24	24	24	24	24	24
	RM attribute		200	200	200	200	200	200	200	200

Transport Format Combination Set								
TFCI	DCH	DCH	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF1	TF0	TF0	TF0	TF1	TF0	TF0	TF0
2	TF2	TF1	TF0	TF0	TF2	TF1	TF0	TF0
3	TF3	TF1	TF0	TF0	TF3	TF1	TF0	TF0
4	TF0	TF2	TF1	TF0	TF0	TF2	TF1	TF0
5	TF1	TF2	TF1	TF0	TF1	TF2	TF1	TF0
6	TF2	TF3	TF1	TF0	TF2	TF3	TF1	TF0
7	TF3	TF3	TF1	TF0	TF3	TF3	TF1	TF0
8	TF0	TF0	TF2	TF1	TF0	TF0	TF2	TF1
9	TF1	TF0	TF2	TF1	TF1	TF0	TF2	TF1
10	TF2	TF1	TF2	TF1	TF2	TF1	TF2	TF1
11	TF3	TF1	TF2	TF1	TF3	TF1	TF2	TF1
12	TF0	TF2	TF0	TF2	TF0	TF2	TF0	TF2
13	TF1	TF2	TF0	TF2	TF1	TF2	TF0	TF2
14	TF2	TF3	TF0	TF2	TF2	TF3	TF0	TF2
15	TF3	TF3	TF0	TF2	TF3	TF3	TF0	TF2
16	TF0	TF0	TF1	TF2	TF0	TF0	TF1	TF2
17	TF1	TF0	TF1	TF2	TF1	TF0	TF1	TF2
18	TF2	TF1	TF1	TF2	TF2	TF1	TF1	TF2

19	TF3	TF1	TF1	TF2	TF3	TF1	TF1	TF2
20	TF0	TF2	TF2	TF3	TF0	TF2	TF2	TF3
21	TF1	TF2	TF2	TF3	TF1	TF2	TF2	TF3
22	TF2	TF3	TF2	TF3	TF2	TF3	TF2	TF3
23	TF3	TF3	TF2	TF3	TF3	TF3	TF2	TF3

B3.10.2 Transmitting (DCH 8Channel)

B3.10.2.1 Downlink setting

• DPCH(DCH 8Channel)

SLOT FORMAT(SYMBOL RATE)	Be sure to set to 14(240ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B3.10.2.2 Transmittable TFCS etc.

Transport Format Set										
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	1x0	1x0	1x0	1x0	1x0	1x0	1x0
		TF1,bits	1x12	1x16	1x20	1x12	1x16	1x20	1x1664	1x32
		TF2,bits	1x24	1x32	1x40	1x24	1x32	1x40	1x3328	1x64
		TF3,bits	1x36	1x48	1x60	1x36	1x48	1x60	-	1x96
		TF4,bits	1x48	1x64	1x80	1x48	1x64	1x80	-	1x128
		TF5,bits	1x60	1x80	1x100	1x60	1x80	1x100	-	1x160
		TF6,bits	2x30	2x40	2x50	2x30	2x40	2x50	-	2x80
		TF7,bits	4x15	4x20	4x25	4x15	4x20	4x25	-	4x40
	TTI,ms		20	20	20	20	20	20	40	40
	Coding type		Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Convolu-tional Coding 1/3	Turbo Coding	Convolu-tional Coding 1/3
	CRC, bit		24	24	24	24	24	24	24	24
	RM attribute		200	200	200	200	200	200	200	200

Transport Format Combination Set								
TFCI	DCH	DCH	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF1	TF0	TF0	TF1	TF0	TF0	TF1	TF0
2	TF2	TF1	TF0	TF2	TF1	TF0	TF2	TF0
3	TF0	TF4	TF2	TF0	TF4	TF2	TF0	TF0
4	TF1	TF4	TF2	TF1	TF4	TF2	TF1	TF0
5	TF2	TF5	TF2	TF2	TF5	TF2	TF2	TF0
6	TF0	TF0	TF4	TF0	TF0	TF4	TF0	TF1
7	TF1	TF0	TF4	TF1	TF0	TF4	TF1	TF1
8	TF2	TF1	TF4	TF2	TF1	TF4	TF2	TF1
9	TF0	TF4	TF6	TF0	TF4	TF6	TF0	TF2
10	TF1	TF4	TF6	TF1	TF4	TF6	TF1	TF2
11	TF2	TF5	TF6	TF2	TF5	TF6	TF2	TF2
12	TF0	TF0	TF0	TF0	TF0	TF0	TF0	TF2
13	TF1	TF0	TF0	TF1	TF0	TF0	TF1	TF2

14	TF2	TF1	TF0	TF2	TF1	TF0	TF2	TF2
15	TF0	TF4	TF2	TF0	TF4	TF2	TF0	TF3
16	TF1	TF4	TF2	TF1	TF4	TF2	TF1	TF3
17	TF2	TF5	TF2	TF2	TF5	TF2	TF2	TF3
18	TF0	TF0	TF4	TF0	TF0	TF4	TF0	TF4
19	TF1	TF0	TF4	TF1	TF0	TF4	TF1	TF4
20	TF2	TF1	TF4	TF2	TF1	TF4	TF2	TF4
21	TF0	TF4	TF6	TF0	TF4	TF6	TF0	TF4
22	TF1	TF4	TF6	TF1	TF4	TF6	TF1	TF4
23	TF2	TF5	TF6	TF2	TF5	TF6	TF2	TF4
24	TF0	TF0	TF0	TF0	TF0	TF0	TF0	TF5
25	TF1	TF0	TF0	TF1	TF0	TF0	TF1	TF5
26	TF2	TF1	TF0	TF2	TF1	TF0	TF2	TF5
27	TF0	TF4	TF2	TF0	TF4	TF2	TF0	TF6
28	TF1	TF4	TF2	TF1	TF4	TF2	TF1	TF6
29	TF2	TF5	TF2	TF2	TF5	TF2	TF2	TF6
30	TF0	TF0	TF4	TF0	TF0	TF4	TF0	TF6
31	TF1	TF0	TF4	TF1	TF0	TF4	TF1	TF6
32	TF2	TF1	TF4	TF2	TF1	TF4	TF2	TF6
33	TF0	TF4	TF6	TF0	TF4	TF6	TF0	TF7
34	TF1	TF4	TF6	TF1	TF4	TF6	TF1	TF7
35	TF2	TF5	TF6	TF2	TF5	TF6	TF2	TF7

B3.10.2.3 Transmit procedures for each TFCI

✳1 "DCH #X" (X should be DCH channel number, 0~7.)

✳2 "TrCH#X TFI Num?" (X should be DCH channel number, 0~7.)

TFCI		DCH_Data(8CH_1).dll							
		DCH #0	DCH #1	DCH #2	DCH #3	DCH #4	DCH #5	DCH #6	DCH #7
0	✳1	–	–	–	–	–	–	–	–
	✳2	–	–	–	–	–	–	–	–
1	✳1	Act	DeAct	DeAct	Act	DeAct	DeAct	Act	DeAct
	✳2	1	–	–	1	–	–	–	–
2	✳1	Act	Act	DeAct	Act	Act	DeAct	Act	DeAct
	✳2	2	1	–	2	1	–	2	–
3	✳1	DeAct	Act	Act	DeAct	Act	Act	DeAct	DeAct
	✳2	–	4	2	–	4	2	–	–
4	✳1	Act	Act	Act	Act	Act	Act	Act	DeAct
	✳2	1	4	2	1	4	2	1	–
5	✳1	Act	Act	Act	Act	Act	Act	Act	DeAct
	✳2	2	5	2	2	5	2	2	–
6	✳1	DeAct	DeAct	Act	DeAct	DeAct	Act	DeAct	DeAct
	✳2	–	–	4	–	–	4	–	1
7	✳1	Act	DeAct	Act	Act	DeAct	Act	Act	Act
	✳2	1	–	4	1	–	4	1	1
8	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	1	4	2	1	4	2	1
9	✳1	DeAct	Act	Act	DeAct	Act	Act	DeAct	Act
	✳2	–	4	6	–	4	6	–	2
10	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	1	4	6	1	4	6	1	2
11	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	5	6	2	5	6	2	2
12	✳1	DeAct	DeAct	DeAct	DeAct	DeAct	DeAct	DeAct	DeAct
	✳2	–	–	–	–	–	–	–	2
13	✳1	Act	DeAct	DeAct	Act	DeAct	DeAct	Act	Act
	✳2	1	–	–	1	–	–	1	2
14	✳1	Act	Act	DeAct	Act	Act	DeAct	Act	Act
	✳2	2	1	–	2	1	–	2	2
15	✳1	DeAct	Act	Act	DeAct	Act	Act	DeAct	Act
	✳2	–	4	2	–	4	2	–	3
16	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	1	4	2	1	4	2	1	3
17	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	5	2	2	5	2	2	3
18	✳1	DeAct	DeAct	Act	DeAct	DeAct	Act	DeAct	Act
	✳2	–	–	4	–	–	4	–	4
19	✳1	Act	DeAct	Act	Act	DeAct	Act	Act	Act
	✳2	1	–	4	1	–	4	1	4
20	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	1	4	2	1	4	2	4
21	✳1	DeAct	Act	Act	DeAct	Act	Act	DeAct	Act
	✳2	–	4	6	–	4	6	–	4
22	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	1	4	6	1	4	6	1	4
23	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	5	6	2	5	6	2	4

24	✳1	DeAct	DeAct	DeAct	DeAct	DeAct	DeAct	DeAct	Act
	✳2	–	–	–	–	–	–	–	5
25	✳1	Act	DeAct	DeAct	Act	DeAct	DeAct	Act	Act
	✳2	1	–	–	1	–	–	1	5
26	✳1	Act	Act	DeAct	Act	Act	DeAct	Act	Act
	✳2	2	1	–	2	1	–	2	5
27	✳1	DeAct	Act	Act	DeAct	Act	Act	DeAct	Act
	✳2	–	4	2	–	4	2	–	6
28	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	1	4	2	1	4	2	1	6
29	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	5	2	2	5	2	2	6
30	✳1	DeAct	DeAct	Act	DeAct	DeAct	Act	DeAct	Act
	✳2	–	–	4	–	–	4	–	6
31	✳1	Act	DeAct	Act	Act	DeAct	Act	Act	Act
	✳2	1	–	4	1	–	4	1	6
32	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	1	4	2	1	4	2	6
33	✳1	DeAct	Act	Act	DeAct	Act	Act	DeAct	Act
	✳2	–	4	6	–	4	6	–	7
34	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	1	4	6	1	4	6	1	7
35	✳1	Act	Act	Act	Act	Act	Act	Act	Act
	✳2	2	5	6	2	5	6	2	7

B.3.11　Packet Convolutional Coding(Downlink 8[kbps]/ Uplink 32[kbps]) +DCCH Setting

Uplink･･･Start_P32K(UL_Conv).dll is used.

Downlink･･･Start_P8K(DL_Conv).dll is used.

B.3.11.1 Receiving Packet Convolutional Coding 32[kbps]+DCCH

B3.11.1.1 Uplink setting

• DPCH(Packet 32k)

Slot Format	Be sure to 2(15ksps)
Symbol Rate	Be sure to set to 120ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	0.96

B3.11.1.2 Receivable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
	TTI,ms		10	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		150	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B3.11.2 Transmitting Packet Convolutional Coding 8[kbps] +DCCH

B3.11.2.1 Downlink setting

• DPCH(Packet 8kbps)

Slot Format(Symbol Rate)	Be sure to set to 9(30ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B3.11.2.2 Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
	TTI,ms		40	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		150	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B3.11.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(Packet).dll			DCH_Data(DCCH).dll
	"DCH Ch No.?"	“TTI?”	"Packet:Num of Transport Block?"	"TTI?"
0	–	–	–	–
1	–	–	–	”40”
2	“0”	"40"	"1"	–
3	“0”	"40"	"1"	”40”

B.3.12 Packet 32[kbps](downlink TTI=20[ms])+DCCH Setting

Start_P32K.dll or Start_P32K(TTI10m).dll is used.

B.3.12.1 Receiving Packet 32[kbps] (TTI=20[ms])+DCCH

B.3.12.1.1 Uplink settings

• DPCH(Packet 32k)

Slot Format		Be sure to set to 2 (15ksps).
Symbol Rate		Be sure to set to 120ksps.
Scrambling Code		Value set on the GUI
Channelization Code		Value set on the GUI
Tx Power		Value set on the GUI
DTX position		N/A
DPCCH	N_{TFCI}	2
	N_{Pilot}	5
	N_{TPC}	2
	N_{FBI}	1
DPDCH	N_{data}	1200
Puncturing Limit		0.96

B.3.12.1.2 Receivable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×336	0×148
		TF1,bits	1×336	1×148
		TF2,bits	2×336	–
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	16
	RM attribute		155	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1

B.3.12.2 Transmitting Packet 32[kbps](TTI=20[ms])+DCCH

B.3.12.2.1 Downlink settings

• DPCH(Packet 32k)

Slot Format	Be sure to set to 12 (60ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.12.2.2 Transmittable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×336	0×148
		TF1,bits	1×336	1×148
		TF2,bits	2×336	–
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	16
	RM attribute		155	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1

B.3.12.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(Packet).dll	DCH_Data(DCCH).dll
0	–	–
1	–	Run
2	–	–
3	After selecting 20 for "TTI?," select "1" for "Packet:Num of Transport Block?"	Run
4	–	–
5	After selecting 20 for "TTI?," select "2" for "Packet:Num of Transport Block?"	Run

B.3.13 Packet 64[kbps](DownlinkTTI=10[ms]) +DCCH Setting

The scenario is not attached.

B.3.13.1 Receiving Packet 64[kbps]+DCCH

B.3.13.1.1 Uplink setting

• DPCH(Packet 64k)

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 240ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	1

B.3.13.1.2 Receivable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
		TF2,bits	2x336	-
		TF3,bits	3x336	-
		TF4,bits	4x336	-
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		150	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1
6	TF3	TF0
7	TF3	TF1
8	TF4	TF0
9	TF4	TF1

B.3.13.2 Transmitting Packet 64[kbps](DownlinkTTI=10[ms]) +DCCH

B.3.13.2.1 Downlink setting

• DPCH(Packet 64kbps)

Slot Format(Symbol Rate)	Be sure to set to 13(120ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.13.2.2Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
		TF2,bits	2x336	-
		TF3,bits	3x336	-
		TF4,bits	4x336	-
	TTI,ms		10	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		145	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1

B.3.13.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(Packet).dll			DCH_Data(DCCH).dll
	"DCH Ch No.?"	"TTI?"	"Packet:Num of Transport Block?"	"TTI?"
0	–	–	–	–
1	–	–	–	"40"
2	"0"	"20"	"1"	–
3	"0"	"20"	"1"	"40"
4	"0"	"20"	"2"	–
5	"0"	"20"	"2"	"40"

B.3.14 Packet 128[kbps] (DownlinkTTI=10[ms]) + DCCH Setting

Start_P128K.dll is used.

B.3.14.1 Receiving Packet 128[kbps]+DCCH

B.3.14.1.1 Uplink setting

• DPCH(Packet 128k)

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 480ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	1

B.3.14.1.2 Receivable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
		TF2,bits	2x336	-
		TF3,bits	4x336	-
		TF4,bits	8x336	-
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		140	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1
6	TF3	TF0
7	TF3	TF1
8	TF4	TF0
9	TF4	TF1

B.3.14.2 Transmitting Packet 128[kbps] +DCCH

B.3.14.2.1 Downlink setting

• DPCH(Packet 128kbps)

Slot Format(Symbol Rate)	Be sure to set to 14(240ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.14.2.2 Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
		TF2,bits	2x336	-
		TF3,bits	4x336	-
		TF4,bits	8x336	-
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		140	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1
6	TF3	TF0
7	TF3	TF1
8	TF4	TF0
9	TF4	TF1

B.3.14.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(Packet).dll			DCH_Data(DCCH).dll
	"DCH Ch No.?"	"TTI?"	"Packet:Num of Transport Block?"	"TTI?"
0	–	–	–	–
1	–	–	–	Select "40" for "TTI?"
2	"0"	"20"	"1"	–
3	"0"	"20"	"1"	Select "40" for "TTI?"
4	"0"	"20"	"2"	–
5	"0"	"20"	"2"	Select "40" for "TTI?"
6	"0"	"20"	"4"	–
7	"0"	"20"	"4"	Select "40" for "TTI?"
8	"0"	"20"	"8"	–
9	"0"	"20"	"8"	Select "40" for "TTI?"

B.3.15　Packet 384[kbps](DownlinkTTI=10[ms])+DCCH Setting

Start_P384K.dll is used.

B.3.15.1 Transmitting Packet 384[kbps] +DCCH

B.3.15.1.1 Downlink setting

• DPCH(Packet 384kbps)

Slot Format(Symbol Rate)	Be sure to set to 15(480ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.15.1.2Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x336	0x148
		TF1,bits	1x336	1x148
		TF2,bits	2x336	-
		TF3,bits	4x336	-
		TF4,bits	8x336	-
		TF5,bits	12x336	-
	TTI,ms		10	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16
	RM attribute		145	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1
6	TF3	TF0
7	TF3	TF1
8	TF4	TF0
9	TF4	TF1
10	TF5	TF0
11	TF5	TF1

B.3.15.1.3 Transmit procedure for each TFCI

TFCI	DCH_Data(Packet).dll			DCH_Data(DCCH).dll
	"DCH Ch No.?"	"TTI?"	"Packet:Num of Transport Block?"	"TTI?"
0	–			–
1	–			Select "40" for "TTI?"
2	"0"	"10"	"1"	–
3	"0"	"10"	"1"	Select "40" for "TTI?"
4	"0"	"10"	"2"	–
5	"0"	"10"	"2"	Select "40" for "TTI?"
6	"0"	"10"	"4"	–
7	"0"	"10"	"4"	Select "40" for "TTI?"
8	"0"	"10"	"8"	–
9	"0"	"10"	"8"	Select "40" for "TTI?"
10	"0"	"10"	"12"	–
11	"0"	"10"	"12"	Select "40" for "TTI?"

B.3.16 AudioVisual(32[kbps])+DCCH Setting

Start_AV32K.dll is used.

B.3.16.1 Receiving AV 32[kbps]+DCCH

B.3.16.1.1 Uplink settings

• AudioVisual+DCCH(32Kbps)

Slot Format	Be sure to set to 2 (15ksps).
Symbol Rate	Be sure to set to 120ksps.
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.8

B.3.16.1.2 Receivable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×640	0×148
		TF1,bits	1×640	1×148
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	16
	RM attribute		185	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.16.2　Transmitting AV 32[kbps]+DCCH

B.3.16.2.1　Downlink settings

• AudioVisual+DCCH(32Kbps)

Slot Format(Symbol Rate)	Be sure to set to 12 (60ksps).
Scrambling Code	Be sure to set to 12 (60ksps).
Channelization Code	Be sure to set to 12 (60ksps).
Tx Power	Be sure to set to 12 (60ksps).
DTX position	Flexible

B.3.16.2.2　Transmittable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×640	0×148
		TF1,bits	1×640	1×148
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	16
	RM attribute		185	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.16.2.3　Transmit procedure for each TFCI

TFCI	DCH_Data(AV).dll	DCH_Data(DCCH).dll
0	–	–
1	–	Run
2	Select "1" for "AV:Num of Transport Block?"	–
3	Select "1" for "AV:Num of Transport Block?"	Run

B.3.17 AudioVisual(64[kbps])+DCCH Setting

Start_AV64K.dll is used.

B.3.17.1 Receiving AV 64[kbps]+DCCH

B.3.17.1.1 Uplink settings

• AudioVisual+DCCH(64Kbps)

Slot Format	Be sure to set to 2 (15ksps).
Symbol Rate	Be sure to set to 240ksps.
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.92

B.3.17.1.2 Receivable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×640	0×148
		TF1,bits	2×640	1×148
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	16
	RM attribute		170	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.17.2 Transmitting AV 64[kbps]+DCCH

B.3.17.2.1 Downlink settings

• AudioVisual+DCCH(64Kbps)

Slot Format(Symbol Rate)	Be sure to set to 13 (120ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.17.2.2 Transmittable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×640	0×148
		TF1,bits	2×640	1×148
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	16
	RM attribute		170	160

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.17.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(AV).dll	DCH_Data(DCCH).dll
0	–	–
1	–	Run
2	Select "2" for "AV:Num of Transport Block?"	–
3	Select "2" for "AV:Num of Transport Block?"	Run

B.3.18. AMR+Packet 32[kbps](DownlinkTTI=10[ms])+DCCH Setting

Start_AMRxP32K.dll is used.

B.3.18.1 Receiving AMR+Packet 32[kbps]+DCCH

B.3.18.1.1 Uplink settings

• DPCH(AMR + Packet 32k)

Slot Format	Be sure to set to 2 (15ksps).
Symbol Rate	Be sure to set to 240ksps.
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	1.00

B.3.18.1.2 Receivable TFCS, etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1×0	0×103	0×60	0×336	0×148
		TF1,BITS	1×39	1×103	1×60	1×336	1×148
		TF2,bits	1×81	–	–	2×336	–
	TTI,ms		20	20	20	20	40
	Coding type		Convolutional Coding 1/3	Convolution-al Coding 1/3	Convolutional Coding 1/2	Turbo Coding	Convolutional Coding 1/3
	CRC, bit		12	–	–	16	16
	RM attribute		200	190	235	155	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF2	TF1
6	TF1	TF0	TF0	TF0	TF0
7	TF1	TF0	TF0	TF0	TF1
8	TF1	TF0	TF0	TF1	TF0
9	TF1	TF0	TF0	TF1	TF1
10	TF1	TF0	TF0	TF2	TF0
11	TF1	TF0	TF0	TF2	TF1
12	TF2	TF1	TF1	TF0	TF0
13	TF2	TF1	TF1	TF0	TF1
14	TF2	TF1	TF1	TF1	TF0
15	TF2	TF1	TF1	TF1	TF1
16	TF2	TF1	TF1	TF2	TF0
17	TF2	TF1	TF1	TF2	TF1

B.3.18.2 Transmitting AMR+Packet 32[kbps]+DCCH

B.3.18.2.1 Downlink settings

- DPCH(AMR + Packet 32k)

Slot Format(Symbol Rate)	Be sure to set to 13 (120ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.18.2.2 Transmittable TFCS, etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1×0	0×103	0×60	0×336	0×148
		TF1,BITS	1×39	1×103	1×60	1×336	1×148
		TF2,bits	1×81	–	–	–	–
	TTI,ms		20	20	20	10	40
	Coding type		Convolutional Coding 1/3	Convolutional Coding 1/3	Convolutional Coding 1/2	Turbo Coding	Convolutional Coding 1/3
	CRC, bit		12	-	-	16	16
	RM attribute		200	190	235	155	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF1	TF0	TF0	TF0	TF0
5	TF1	TF0	TF0	TF0	TF1
6	TF1	TF0	TF0	TF1	TF0
7	TF1	TF0	TF0	TF1	TF1
8	TF2	TF1	TF1	TF0	TF0
9	TF2	TF1	TF1	TF0	TF1
10	TF2	TF1	TF1	TF1	TF0
11	TF2	TF1	TF1	TF1	TF1

B.3.18.2.3　transmit procedure for each TFCI

TFCI	DCH_Data(AMR).dll For "AMR?"		DCH_Data_Mc(Packet).dll	DCH_Data (DCCH).dll
0	–	–	–	–
1	–	–	–	Run
2	–	–	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
3	–	–	Ditto	Run
4	Select "39bit" after selecting A	–	–	–
5	Ditto	–	–	Run
6	Ditto	–	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
7	Ditto	–	Ditto	Run
8	Select "81bit" after selecting A	Select "B and C"	–	–
9	Ditto	Select "B and C"	–	Run
10	Ditto	Select "B and C"	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
11	Ditto	Select "B and C"	Ditto	Run

B.3.19　AMR+Packet 64[kbps](DownlinkTTI=10[ms])+DCCH Setting

Start_AMRxP64K.dll is used.

B.3.19.1　Receiving AMR+Packet 64[kbps]+DCCH

B.3.19.1.1　Uplink settings

• DPCH(AMR + Packet 64k)

Slot Format	Be sure to set to 2 (15ksps).
Symbol Rate	Be sure to set to 240ksps.
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.76

B.3.19.1.2　Receivable TFCS, etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1×0	0×103	0×60	0×336	0×148
		TF1,BITS	1×39	1×103	1×60	1×336	1×148
		TF2,bits	1×81	–	–	2×336	–
		TF3,bits	–	–	–	3×336	–
		TF4,bits	–	–	–	4×336	–
	TTI,ms		20	20	20	20	40
	Coding type		Convolutional Coding 1/3	Convolutional Coding 1/3	Convolutional Coding 1/2	Turbo Coding	Convolutional Coding 1/3
	CRC, bit		12	–	–	16	16
	RM attribute		200	190	235	150	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF2	TF1
6	TF0	TF0	TF0	TF3	TF0
7	TF0	TF0	TF0	TF3	TF1
8	TF0	TF0	TF0	TF4	TF0
9	TF0	TF0	TF0	TF4	TF1
10	TF1	TF0	TF0	TF0	TF0
11	TF1	TF0	TF0	TF0	TF1
12	TF1	TF0	TF0	TF1	TF0
13	TF1	TF0	TF0	TF1	TF1
14	TF1	TF0	TF0	TF2	TF0
15	TF1	TF0	TF0	TF2	TF1
16	TF1	TF0	TF0	TF3	TF0
17	TF1	TF0	TF0	TF3	TF1
18	TF1	TF0	TF0	TF4	TF0
19	TF1	TF0	TF0	TF4	TF1
20	TF2	TF1	TF1	TF0	TF0
21	TF2	TF1	TF1	TF0	TF1
22	TF2	TF1	TF1	TF1	TF0
23	TF2	TF1	TF1	TF1	TF1
24	TF2	TF1	TF1	TF2	TF0
25	TF2	TF1	TF1	TF2	TF1
26	TF2	TF1	TF1	TF3	TF0
27	TF2	TF1	TF1	TF3	TF1
28	TF2	TF1	TF1	TF4	TF0
29	TF2	TF1	TF1	TF4	TF1

B.3.19.2　Transmitting AMR+Packet 64[kbps] +DCCH

B.3.19.2.1　Downlink settings

• DPCH(AMR + Packet 64k)

Slot Format(Symbol Rate)	Be sure to set to 13 (120ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.19.2.2　Transmittable TFCS, etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1×0	0×103	0×60	0×336	0×148
		TF1,BITS	1×39	1×103	1×60	1×336	1×148
		TF2,bits	1×81	–	–	2×336	–
	TTI,ms		20	20	20	10	40
	Coding type		Convolutional Coding 1/3	Convolutional Coding 1/3	Convolutional Coding 1/2	Turbo Coding	Convolutional Coding 1/3
	CRC, bit		12	–	–	16	16
	RM attribute		200	190	235	150	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF2	TF1
6	TF1	TF0	TF0	TF0	TF0
7	TF1	TF0	TF0	TF0	TF1
8	TF1	TF0	TF0	TF1	TF0
9	TF1	TF0	TF0	TF1	TF1
10	TF1	TF0	TF0	TF2	TF0
11	TF1	TF0	TF0	TF2	TF1
12	TF2	TF1	TF1	TF0	TF0
13	TF2	TF1	TF1	TF0	TF1
14	TF2	TF1	TF1	TF1	TF0
15	TF2	TF1	TF1	TF1	TF1
16	TF2	TF1	TF1	TF2	TF0
17	TF2	TF1	TF1	TF2	TF1

B.3.19.2.3　Transmit procedure for each TFCI

TFCI	DCH_Data(AMR).dll For "AMR?"		DCH_Data_Mc(Packet).dll	DCH_Data (DCCH).dll
0	–	–	–	–
1	–	–	–	Run
2	–	–	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
3	–	–	Ditto	Run
4	–	–	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
5	–	–	Ditto	Run
6	Select "39bit" after selecting A	–	–	–
7	Ditto	–	–	Run
8	Ditto	–	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
9	Ditto	–	Ditto	Run
10	Ditto	–	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
11	Ditto	–	Ditto	Run
12	Select "81bit" after selecting A	Select "B and C"	–	–
13	Ditto	Select "B and C"	–	Run
14	Ditto	Select "B and C"	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
15	Ditto	Select "B and C"	Ditto	Run
16	Ditto	Select "B and C"	After selecting "10" for "TTI?," select "1" for "Packet:Num of Transport Block?"	–
17	Ditto	Select "B and C"	Ditto	Run

B.3.20 AMR+Packet384[kbps]+DCCH(TTI=20ms) Setting

Start_AMRxP384K(TTI20ms).dll is used.

B.3.20.1 Receiving AMR+Packet384[kbps]+DCCH

B.3.20.1.1 Uplink Setting

• DPCH(AMR + Packet 384[kbps](TTI=20ms))

Slot Format	Be sure to set to 2(15ksps)
Symbol Rate	Be sure to set to 960ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.68

B.3.20.1.2 Receivable TFCS etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1x0	0x103	0x60	0x336	0x148
		TF1,bits	1x39	1x103	1x60	1x336	1x148
		TF2,bits	1x81	-	-	2x336	-
		TF3,bits	-	-	-	4x336	-
		TF4,bits	-	-	-	8x336	-
		TF5,bits	-	-	-	12x336	-
		TF6,bits	-	-	-	16x336	-
		TF7,bits	-	-	-	20x336	-
		TF8,bits	-	-	-	24x336	-
	TTI,ms		20	20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Turbo Coding	Convolutional Coding1/3
	CRC, bit		12	-	-	16	16
	RM attribute		200	190	235	145	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF1	TF0	TF0	TF0	TF0
3	TF1	TF0	TF0	TF0	TF1
4	TF2	TF1	TF1	TF0	TF0
5	TF2	TF1	TF1	TF0	TF1
6	TF0	TF0	TF0	TF1	TF0
7	TF0	TF0	TF0	TF1	TF1
8	TF1	TF0	TF0	TF1	TF0
9	TF1	TF0	TF0	TF1	TF1
10	TF2	TF1	TF1	TF1	TF0
11	TF2	TF1	TF1	TF1	TF1
12	TF0	TF0	TF0	TF2	TF0
13	TF0	TF0	TF0	TF2	TF1

14	TF1	TF0	TF0	TF2	TF0
15	TF1	TF0	TF0	TF2	TF1
16	TF2	TF1	TF1	TF2	TF0
17	TF2	TF1	TF1	TF2	TF1
18	TF0	TF0	TF0	TF3	TF0
19	TF0	TF0	TF0	TF3	TF1
20	TF1	TF0	TF0	TF3	TF0
21	TF1	TF0	TF0	TF3	TF1
22	TF2	TF1	TF1	TF3	TF0
23	TF2	TF1	TF1	TF3	TF1
24	TF0	TF0	TF0	TF4	TF0
25	TF0	TF0	TF0	TF4	TF1
26	TF1	TF0	TF0	TF4	TF0
27	TF1	TF0	TF0	TF4	TF1
28	TF2	TF1	TF1	TF4	TF0
29	TF2	TF1	TF1	TF4	TF1
30	TF0	TF0	TF0	TF5	TF0
31	TF0	TF0	TF0	TF5	TF1
32	TF1	TF0	TF0	TF5	TF0
33	TF1	TF0	TF0	TF5	TF1
34	TF2	TF1	TF1	TF5	TF0
35	TF2	TF1	TF1	TF5	TF1
36	TF0	TF0	TF0	TF6	TF0
37	TF0	TF0	TF0	TF6	TF1
38	TF1	TF0	TF0	TF6	TF0
39	TF1	TF0	TF0	TF6	TF1
40	TF2	TF1	TF1	TF6	TF0
41	TF2	TF1	TF1	TF6	TF1
42	TF0	TF0	TF0	TF7	TF0
43	TF0	TF0	TF0	TF7	TF1
44	TF1	TF0	TF0	TF7	TF0
45	TF1	TF0	TF0	TF7	TF1
46	TF2	TF1	TF1	TF7	TF0
47	TF2	TF1	TF1	TF7	TF1
48	TF0	TF0	TF0	TF8	TF0
49	TF0	TF0	TF0	TF8	TF1
50	TF1	TF0	TF0	TF8	TF0
51	TF1	TF0	TF0	TF8	TF1
52	TF2	TF1	TF1	TF8	TF0
53	TF2	TF1	TF1	TF8	TF1

B.3.20.2　Transmitting AMR+Packet 384[kbps]+DCCH(TTI=20ms)

B.3.20.2.1　Downlink setting

• DPCH(AMR + Packet 384[kbps](TTI=20ms))

Slot Format(Symbol Rate)	Be sure set to 14(240ksps)
Scrambling Code	Each value set on the GUI
Channelization Code	Each value set on the GUI
Tx Power	Each value set on the GUI
DTX position	Flexible

B.3.20.2.2　Transmittable TFCS etc.

Transport Format Set							
D	TrCH type		DCH	DCH	DCH	DCH	DCH
P	T	TF0,bits	1x0	0x103	0x60	0x336	0x148
C	F	TF1,bits	1x39	1x103	1x60	1x336	1x148
H	S	TF2,bits	1x81	-	-	2x336	-
		TF3,bits	-	-	-	4x336	-
		TF4,bits	-	-	-	8x336	-
		TF5,bits	-	-	-	12x336	-
		TF6,bits	-	-	-	16x336	-
		TF7,bits	-	-	-	20x336	-
		TF8,bits	-	-	-	24x336	-
	TTI,ms		20	20	20	20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3	Convolutional Coding1/2	Turbo Coding	Convolutional Coding1/3
	CRC, bit		12	-	-	16	16
	RM attribute		200	190	235	145	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF1	TF0	TF0	TF0	TF0
3	TF1	TF0	TF0	TF0	TF1
4	TF2	TF1	TF1	TF0	TF0
5	TF2	TF1	TF1	TF0	TF1
6	TF0	TF0	TF0	TF1	TF0
7	TF0	TF0	TF0	TF1	TF1
8	TF1	TF0	TF0	TF1	TF0
9	TF1	TF0	TF0	TF1	TF1
10	TF2	TF1	TF1	TF1	TF0
11	TF2	TF1	TF1	TF1	TF1
12	TF0	TF0	TF0	TF2	TF0
13	TF0	TF0	TF0	TF2	TF1
14	TF1	TF0	TF0	TF2	TF0
15	TF1	TF0	TF0	TF2	TF1
16	TF2	TF1	TF1	TF2	TF0
17	TF2	TF1	TF1	TF2	TF1
18	TF0	TF0	TF0	TF3	TF0

19	TF0	TF0	TF0	TF3	TF1
20	TF1	TF0	TF0	TF3	TF0
21	TF1	TF0	TF0	TF3	TF1
22	TF2	TF1	TF1	TF3	TF0
23	TF2	TF1	TF1	TF3	TF1
24	TF0	TF0	TF0	TF4	TF0
25	TF0	TF0	TF0	TF4	TF1
26	TF1	TF0	TF0	TF4	TF0
27	TF1	TF0	TF0	TF4	TF1
28	TF2	TF1	TF1	TF4	TF0
29	TF2	TF1	TF1	TF4	TF1
30	TF0	TF0	TF0	TF5	TF0
31	TF0	TF0	TF0	TF5	TF1
32	TF1	TF0	TF0	TF5	TF0
33	TF1	TF0	TF0	TF5	TF1
34	TF2	TF1	TF1	TF5	TF0
35	TF2	TF1	TF1	TF5	TF1
36	TF0	TF0	TF0	TF6	TF0
37	TF0	TF0	TF0	TF6	TF1
38	TF1	TF0	TF0	TF6	TF0
39	TF1	TF0	TF0	TF6	TF1
40	TF2	TF1	TF1	TF6	TF0
41	TF2	TF1	TF1	TF6	TF1
42	TF0	TF0	TF0	TF7	TF0
43	TF0	TF0	TF0	TF7	TF1
44	TF1	TF0	TF0	TF7	TF0
45	TF1	TF0	TF0	TF7	TF1
46	TF2	TF1	TF1	TF7	TF0
47	TF2	TF1	TF1	TF7	TF1
48	TF0	TF0	TF0	TF8	TF0
49	TF0	TF0	TF0	TF8	TF1
50	TF1	TF0	TF0	TF8	TF0
51	TF1	TF0	TF0	TF8	TF1
52	TF2	TF1	TF1	TF8	TF0
53	TF2	TF1	TF1	TF8	TF1

B.3.20.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(AMR).dll			DCH_Data(Packet).dll			DCH_Data (DCCH).dll
	The 1st time		The 2nd time				
	"AMR?"	"TB Size?"	"AMR?"	"DCH Ch No.?"	"TTI?"	"Packet:Num of Transport Block?"	"TTI?"
0	–		–	–	–	–	–
1	–		–	–	–	–	“40”
2	”A”	"39bit"	–	–	–	–	–
3	”A”	"39bit"	–	–	–	–	“40”
4	”A”	"81bit"	"B and C"	–	–	–	–
5	”A”	"81bit"	"B and C"	–	–	–	“40”
6	–		–	“3”	“20”	“1”	–
7	–		–	“3”	“20”	“1”	“40”
8	”A”	"39bit"	–	“3”	“20”	“1”	–
9	”A”	"39bit"	–	“3”	“20”	“1”	“40”
10	”A”	"81bit"	"B and C"	“3”	“20”	“1”	–
11	”A”	"81bit"	"B and C"	“3”	“20”	“1”	“40”
12	–		–	“3”	“20”	“2”	–
13	–		–	“3”	“20”	“2”	“40”
14	”A”	"39bit"	–	“3”	“20”	“2”	–
15	”A”	"39bit"	–	“3”	“20”	“2”	“40”
16	”A”	"81bit"	"B and C"	“3”	“20”	“2”	–
17	”A”	"81bit"	"B and C"	“3”	“20”	“2”	“40”
18	–		–	“3”	“20”	“3”	–
19	–		–	“3”	“20”	“3”	“40”
20	”A”	"39bit"	–	“3”	“20”	“3”	–
21	”A”	"39bit"	–	“3”	“20”	“3”	“40”
22	”A”	"81bit"	"B and C"	“3”	“20”	“3”	–
23	”A”	"81bit"	"B and C"	“3”	“20”	“3”	“40”
24	–		–	“3”	“20”	“4”	–
25	–		–	“3”	“20”	“4”	“40”
26	”A”	"39bit"	–	“3”	“20”	“4”	–
27	”A”	"39bit"	–	“3”	“20”	“4”	“40”
28	”A”	"81bit"	"B and C"	“3”	“20”	“4”	–

29	”A”	"81bit"	"B and C"	“3”	“20”	“4”	“40”
30	–		–	“3”	“20”	“5”	–
31	–		–	“3”	“20”	“5”	“40”
32	”A”	"39bit"	–	“3”	“20”	“5”	–
33	”A”	"39bit"	–	“3”	“20”	“5”	“40”
34	”A”	"81bit"	"B and C"	“3”	“20”	“5”	–
35	”A”	"81bit"	"B and C"	“3”	“20”	“5”	“40”
36	–		–	“3”	“20”	“6”	–
37	–		–	“3”	“20”	“6”	“40”
38	”A”	"39bit"	–	“3”	“20”	“6”	–
39	”A”	"39bit"	–	“3”	“20”	“6”	“40”
40	”A”	"81bit"	"B and C"	“3”	“20”	“6”	–
41	”A”	"81bit"	"B and C"	“3”	“20”	“6”	“40”
42	–		–	“3”	“20”	“7”	–
43	–		–	“3”	“20”	“7”	“40”
44	”A”	"39bit"	–	“3”	“20”	“7”	–
45	”A”	"39bit"	–	“3”	“20”	“7”	“40”
46	”A”	"81bit"	"B and C"	“3”	“20”	“7”	–
47	”A”	"81bit"	"B and C"	“3”	“20”	“7”	“40”
48	–		–	“3”	“20”	“8”	–
49	–		–	“3”	“20”	“8”	“40”
50	”A”	"39bit"	–	“3”	“20”	“8”	–
51	”A”	"39bit"	–	“3”	“20”	“8”	“40”
52	”A”	"81bit"	"B and C"	“3”	“20”	“8”	–
53	”A”	"81bit"	"B and C"	“3”	“20”	“8”	“40”

B.3.21 AMR+UDI+DCCH Setting

Start_AMRxUDI.dll is used.

B.3.21.1 Receiving AMR+UDI+DCCH

B.3.21.1.1 Uplink settings

• DPCH(AMR + UDI)

Slot Format	Be sure to set to 2 (15ksps).
Symbol Rate	Be sure to set to 240ksps.
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.72

B.3.21.1.2 Receivable TFCS, etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1×0	0×103	0×60	0×640	0×148
		TF1,BITS	1×39	1×103	1×60	4×640	1×148
		TF2,bits	1×81	–	–	–	–
	TTI,ms		20	20	20	40	40
	Coding type		Convolutional Coding 1/3	Convolutional Coding 1/3	Convolutional Coding 1/2	Turbo Coding	Convolutional Coding 1/3
	CRC, bit		12	–	–	16	16
	RM attribute		200	190	235	170	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF1	TF0	TF0	TF0	TF0
5	TF1	TF0	TF0	TF0	TF1
6	TF1	TF0	TF0	TF1	TF0
7	TF1	TF0	TF0	TF1	TF1
8	TF2	TF1	TF1	TF0	TF0
9	TF2	TF1	TF1	TF0	TF1
10	TF2	TF1	TF1	TF1	TF0
11	TF2	TF1	TF1	TF1	TF1

B.3.21.2 Transmitting AMR+UDI+DCCH

B.3.21.2.1 Downlink settings

• DPCH(AMR + UDI)

Slot Format(Symbol Rate)	Be sure to set to 13 (120ksps).
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.21.2.2 Transmittable TFCS, etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	1×0	0×103	0×60	0×640	0×148
		TF1,BITS	1×39	1×103	1×60	4×640	1×148
		TF2,bits	1×81	–	–	–	–
	TTI,ms		20	20	20	40	40
	Coding type		Convolutional Coding 1/3	Convolutional Coding 1/3	Convolutional Coding 1/2	Turbo Coding	Convolutional Coding 1/3
	CRC, bit		12	–	–	16	16
	RM attribute		200	190	235	170	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF1	TF0	TF0	TF0	TF0
5	TF1	TF0	TF0	TF0	TF1
6	TF1	TF0	TF0	TF1	TF0
7	TF1	TF0	TF0	TF1	TF1
8	TF2	TF1	TF1	TF0	TF0
9	TF2	TF1	TF1	TF0	TF1
10	TF2	TF1	TF1	TF1	TF0
11	TF2	TF1	TF1	TF1	TF1

B.3.21.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(AMR).dll For "AMR?"		DCH_Data_Mc(UdI).dll	DCH_Data (DCCH).dll
0	–	–	–	–
1	–	–	–	Run
2	–	–	Run	–
3	–	–	Run	Run
4	Select "39bit" after selecting A	–	–	–
5	Ditto	–	–	Run
6	Ditto	–	Run	–
7	Ditto	–	Run	Run
8	Select "81bit" after selecting A	Select "B and C"	–	–
9	Ditto	Select "B and C"	–	Run
10	Ditto	Select "B and C"	Run	–
11	Ditto	Select "B and C"	Run	Run

B.3.22　Reference Measurement Channel(12.2[kbps]) Setting

Start_RMC12_2K.dll is used.

B.3.22.1　Receiving RMC(12.2[kbps])

B.3.22.1.1　Uplink setting

• Reference Measurement Channel (12.2K)

Slot Format	Be sure to set to 0(15ksps)
Symbol Rate	Be sure to set to 60ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	1

B.3.22.1.2　Receivable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x244	0x100
		TF1,bits	1x244	1x100
	TTI,ms		20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.22.2 Transmitting RMC(12.2[kbps])

B.3.22.2.1 Downlink setting

• RMC12.2K+DCCH

Slot Format(Symbol Rate)	Be sure to set to 11(30ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

B.3.22.2.2 Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×244	0×100
		TF1,bits	1×244	1×100
	TTI,ms		20	40
	Coding type		Convolutional Cod-ing1/3	Convolutional Coding1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.22.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(RMC).dll	DCH_Data(RMCDCCH).dll
0	–	–
1	–	Run
2	Select “12.2” for “RMC?” after the run	–
3	Select “12.2” for “RMC?” after the run	Run

B.3.23　Reference Measurement Channel (64[kbps]) Setting

Start_RMC64K.dll is used.

B.3.23.1　Receiving RMC (64[kbps])

B.3.23.1.1　Uplink settings

• Reference Measurement Channel (64K)

Slot Format	Be sure to set to 0 (15ksps).
Symbol Rate (#1)	Be sure to set to 240ksps.
Scrambling Code	Value set on the GUI
Channelization Code (#2)	Value set on the GUI
DTX position	N/A
Puncturing Limit	1

B.3.23.1.2　Receivable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×1280	0×100
		TF1,bits	1×1280	1×100
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.23.2 Transmitting RMC (64[kbps])

B.3.23.2.1 Downlink settings

• RMC64K+DCCH

Slot Format(Symbol Rate (#1))	Be sure to set to 13 (120ksps).
Scrambling Code	Value set on the GUI
Channelization Code (#2)	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

B.3.23.2.2 Transmittable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×1280	0×100
		TF1,bits	1×1280	1×100
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.23.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(RMC).dll	DCH_Data(RMCDCCH).dll
0	–	–
1	–	Run
2	Select "64" for "RMC?" after execution	–
3	Select "64" for "RMC?" after execution	Run

B.3.24　Reference Measurement Channel(144[kbps]) Setting

The scenario is not attached.

B.3.24.1　Receiving RMC(144[kbps])

B.3.24.1.1　Uplink setting

• Reference Measurement Channel (144K)

Slot Format	Be sure to set to 0(15ksps)
Symbol Rate	Be sure to set to 480ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	Flexible
Puncturing Limit	1

B.3.24.1.2　Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x2880	0x100
		TF1,bits	1x2880	1x100
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.24.2 Transmitting RMC(144[kbps])

B.3.24.2.1 Downlink setting

• RMC144K+DCCH

Slot Format(Symbol Rate)	Be sure to set to 14(240ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX POSITION	Fixed

B.3.24.2.2 Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0x2880	0x100
		TF1,bits	1x2880	1x100
	TTI,ms		20	40
	Coding type		Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.24.2.3 Transmit procedure for each TFCI

TFCI	DCH_Data(RMC).dll	DCH_Data(RMCDCCH).dll
0	–	–
1	–	Run
2	Select “144” for “RMC?” after the run	–
3	Select “144” for “RMC?” after the run	Run

B.3.25 Reference Measurement Channel (384[kbps]) Setting

Start_RMC384K.dll is used.

B.3.25.1 Receiving RMC (384[kbps])

B.3.25.1.1 Uplink settings

• Reference Measurement Channel (384K)

Slot Format	Be sure to set to 0 (15ksps).
Symbol Rate (#1)	Be sure to set to 960ksps.
Scrambling Code	Value set on the GUI
Channelization Code (#2)	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.82

B.3.25.1.2 Receivable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×3840	0×100
		TF1,bits	1×3840	1×100
	TTI,ms		10	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.25.2　Transmitting RMC (384[kbps])

B.3.25.2.1　Downlink settings

• RMC384K+DCCH

Slot Format(Symbol Rate (#1))	Be sure to set to 15 (480ksps).
Scrambling Code	Value set on the GUI
Channelization Code (#2)	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

B.3.25.2.2　Transmittable TFCS, etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	0×3840	0×100
		TF1,bits	1×3840	1×100
	TTI,ms		10	40
	Coding type		Turbo Coding	Convolutional Coding 1/3
	CRC, bit		16	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1

B.3.25.2.3　Transmit procedure for each TFCI

TFCI	DCH_Data(RMC).dll	DCH_Data(RMCDCCH).dll
0	–	–
1	–	Run
2	Select "384" for "RMC?" after execution	–
3	Select "384" for "RMC?" after execution	Run

B.3.26　Reference Measurement Channel(BTFD) Setting

Start_RMCBTFD.dll is used.

B.3.26.1　Receiving RMC(BTFD)

B.3.26.1.1　Uplink setting

• Reference Measurement Channel (BTFD)

Slot Format	Be sure to set to 0(15ksps)
Symbol Rate	Be sure to set to 60ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	1

B.3.26.1.2　Receivable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	1x51	0x100
		TF1,bits	1x107	1x100
		TF2,bits	1x115	–
		TF3,bits	1x130	–
		TF4,bits	1x146	–
		TF5,bits	1x160	–
		TF6,bits	1x171	–
		TF7,bits	1x216	–
		TF8,bits	1x256	–
	TTI,ms		20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		0	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1
6	TF3	TF0
7	TF3	TF1
8	TF4	TF0
9	TF4	TF1
10	TF5	TF0
11	TF5	TF1
12	TF6	TF0
13	TF6	TF1
14	TF7	TF0
15	TF7	TF1
16	TF8	TF0
17	TF8	TF1

B.3.26.2　Transmitting RMC(BTFD)

B.3.26.2.1　Downlink setting

• Reference Measurement Channel (BTFD)

Slot Format(Symbol Rate)	Be sure to set to 8(30ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Fixed

B.3.26.2.2　Transmittable TFCS etc.

Transport Format Set				
DPCH	TrCH type		DCH	DCH
	TFS	TF0,bits	1x39	0x100
		TF1,bits	1x95	1x100
		TF2,bits	1x103	–
		TF3,bits	1x118	–
		TF4,bits	1x134	–
		TF5,bits	1x148	–
		TF6,bits	1x159	–
		TF7,bits	1x204	–
		TF8,bits	1x244	–
	TTI,ms		20	40
	Coding type		Convolutional Coding1/3	Convolutional Coding1/3
	CRC, bit		12	12
	RM attribute		256	256

Transport Format Combination Set		
TFCI	DCH	DCH
0	TF0	TF0
1	TF0	TF1
2	TF1	TF0
3	TF1	TF1
4	TF2	TF0
5	TF2	TF1
6	TF3	TF0
7	TF3	TF1
8	TF4	TF0
9	TF4	TF1
10	TF5	TF0
11	TF5	TF1
12	TF6	TF0
13	TF6	TF1
14	TF7	TF0
15	TF7	TF1
16	TF8	TF0
17	TF8	TF1

B.3.26.2.3　Transmit procedure for each TFC

TFCI	DCH_Data(RMC).dll	DCH_Data(RMCDCCH).dll
0	–	–
1	–	Run
2	Select "1" for "BTFD TFI Num?" after the run	–
3	Select "1" for "BTFD TFI Num?" after the run	Run
4	Select "2" for "BTFD TFI Num?" after the run	–
5	Select "2" for "BTFD TFI Num?" after the run	Run
6	Select "3" for "BTFD TFI Num?" after the run	–
7	Select "3" for "BTFD TFI Num?" after the run	Run
8	Select "4" for "BTFD TFI Num?" after the run	–
9	Select "4" for "BTFD TFI Num?" after the run	Run
10	Select "5" for "BTFD TFI Num?" after the run	–
11	Select "5" for "BTFD TFI Num?" after the run	Run
12	Select "6" for "BTFD TFI Num?" after the run	–
13	Select "6" for "BTFD TFI Num?" after the run	Run
14	Select "7" for "BTFD TFI Num?" after the run	–
15	Select "7" for "BTFD TFI Num?" after the run	Run
16	Select "8" for "BTFD TFI Num?" after the run	–
17	Select "8" for "BTFD TFI Num?" after the run	Run

B.3.27 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH Setting

Start_4B.dll is used.

B.3.27.1 Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH

B.3.27.1.1 Uplink setting

• DTCH(UDI)x4+DCCHx1

Slot Format	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 960ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.96

B.3.27.1.2 Receivable TFCS etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x148
		TF1,bits	4x640	4x640	4x640	4x640	1x148
	TTI,ms		40	40	40	40	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16
	RM attribute		170	170	170	170	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF1	TF0	TF0
5	TF0	TF0	TF1	TF0	TF1
6	TF0	TF0	TF1	TF1	TF0
7	TF0	TF0	TF1	TF1	TF1
8	TF0	TF1	TF0	TF0	TF0
9	TF0	TF1	TF0	TF0	TF1
10	TF0	TF1	TF0	TF1	TF0
11	TF0	TF1	TF0	TF1	TF1
12	TF0	TF1	TF1	TF0	TF0
13	TF0	TF1	TF1	TF0	TF1
14	TF0	TF1	TF1	TF1	TF0
15	TF0	TF1	TF1	TF1	TF1
16	TF1	TF0	TF0	TF0	TF0
17	TF1	TF0	TF0	TF0	TF1
18	TF1	TF0	TF0	TF1	TF0

19	TF1	TF0	TF0	TF1	TF1
20	TF1	TF0	TF1	TF0	TF0
21	TF1	TF0	TF1	TF0	TF1
22	TF1	TF0	TF1	TF1	TF0
23	TF1	TF0	TF1	TF1	TF1
24	TF1	TF1	TF0	TF0	TF0
25	TF1	TF1	TF0	TF0	TF1
26	TF1	TF1	TF0	TF1	TF0
27	TF1	TF1	TF0	TF1	TF1
28	TF1	TF1	TF1	TF0	TF0
29	TF1	TF1	TF1	TF0	TF1
30	TF1	TF1	TF1	TF1	TF0
31	TF1	TF1	TF1	TF1	TF1

B.3.27.2 Transmitting ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH

B.3.27.2.1 Downlink setting

• DTCH(UDI)x4+DCCHx1

Slot Format(Symbol Rate)	Be sure to set to15(480ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.27.2.2 Transmittable TFCS etc.

Transport Format Set							
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x148
		TF1,bits	4x640	4x640	4x640	4x640	1x148
	TTI,ms		40	40	40	40	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16
	RM attribute		170	170	170	170	160

Transport Format Combination Set					
TFCI	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF1	TF0	TF0
5	TF0	TF0	TF1	TF0	TF1
6	TF0	TF0	TF1	TF1	TF0
7	TF0	TF0	TF1	TF1	TF1
8	TF0	TF1	TF0	TF0	TF0
9	TF0	TF1	TF0	TF0	TF1
10	TF0	TF1	TF0	TF1	TF0
11	TF0	TF1	TF0	TF1	TF1
12	TF0	TF1	TF1	TF0	TF0
13	TF0	TF1	TF1	TF0	TF1
14	TF0	TF1	TF1	TF1	TF0
15	TF0	TF1	TF1	TF1	TF1
16	TF1	TF0	TF0	TF0	TF0
17	TF1	TF0	TF0	TF0	TF1
18	TF1	TF0	TF0	TF1	TF0
19	TF1	TF0	TF0	TF1	TF1
20	TF1	TF0	TF1	TF0	TF0
21	TF1	TF0	TF1	TF0	TF1
22	TF1	TF0	TF1	TF1	TF0

23	TF1	TF0	TF1	TF1	TF1
24	TF1	TF1	TF0	TF0	TF0
25	TF1	TF1	TF0	TF0	TF1
26	TF1	TF1	TF0	TF1	TF0
27	TF1	TF1	TF0	TF1	TF1
28	TF1	TF1	TF1	TF0	TF0
29	TF1	TF1	TF1	TF0	TF1
30	TF1	TF1	TF1	TF1	TF0
31	TF1	TF1	TF1	TF1	TF1

B.3.27.2.3 Transmit procedure for each TFCI

TFCI	"DCH Ch No.?" in DCH_Data(1B-6B).dll				DCH_Data (DCCH).dll "TTI?"
	The 1st time	The 2nd time	The 3rd time	The 4th time	
0	–	–	–	–	–
1	–	–	–	–	"40"
2	"3"	–	–	–	–
3	"3"	–	–	–	"40"
4	–	"2"	–	–	–
5	–	"2"	–	–	"40"
6	"3"	"2"	–	–	–
7	"3"	"2"	–	–	"40"
8	–	–	"1"	–	–
9	–	–	"1"	–	"40"
10	"3"	–	"1"	–	–
11	"3"	–	"1"	–	"40"
12	–	"2"	"1"	–	–
13	–	"2"	"1"	–	"40"
14	"3"	"2"	"1"	–	–
15	"3"	"2"	"1"	–	"40"
16	–	–	–	"0"	–
17	–	–	–	"0"	"40"
18	"3"	–	–	"0"	–
19	"3"	–	–	"0"	"40"
20	–	"2"	–	"0"	–
21	–	"2"	–	"0"	"40"
22	"3"	"2"	–	"0"	–
23	"3"	"2"	–	"0"	"40"

24	–	–	“1”	“0”	–
25	–	–	“1”	“0”	“40”
26	“3”	–	“1”	“0”	–
27	“3”	–	“1”	“0”	“40”
28	–	“2”	“1”	“0”	–
29	–	“2”	“1”	“0”	“40”
30	“3”	“2”	“1”	“0”	–
31	“3”	“2”	“1”	“0”	“40”

B.3.28 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH Setting

Start_5B.dll is used.

B.3.28.1 Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) +ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH

B.3.28.1.1 Uplink setting

• DTCH(UDI)x5+DCCHx1

Slot Format	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 960ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.76

B.3.28.1.2 Receivable TFCS etc.

Transport Format Set								
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x640	0x148
		TF1,bits	4x640	4x640	4x640	4x640	4x640	1x148
	TTI,ms		40	40	40	40	40	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16
	RM attribute		170	170	170	170	170	160

Transport Format Combination Set						
TFCI	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF1	TF0	TF0
5	TF0	TF0	TF0	TF1	TF0	TF1
6	TF0	TF0	TF0	TF1	TF1	TF0
7	TF0	TF0	TF0	TF1	TF1	TF1
8	TF0	TF0	TF1	TF0	TF0	TF0
9	TF0	TF0	TF1	TF0	TF0	TF1
10	TF0	TF0	TF1	TF0	TF1	TF0
11	TF0	TF0	TF1	TF0	TF1	TF1
12	TF0	TF0	TF1	TF1	TF0	TF0
13	TF0	TF0	TF1	TF1	TF0	TF1
14	TF0	TF0	TF1	TF1	TF1	TF0
15	TF0	TF0	TF1	TF1	TF1	TF1
16	TF0	TF1	TF0	TF0	TF0	TF0
17	TF0	TF1	TF0	TF0	TF0	TF1

18	TF0	TF1	TF0	TF0	TF1	TF0
19	TF0	TF1	TF0	TF0	TF1	TF1
20	TF0	TF1	TF0	TF1	TF0	TF0
21	TF0	TF1	TF0	TF1	TF0	TF1
22	TF0	TF1	TF0	TF1	TF1	TF0
23	TF0	TF1	TF0	TF1	TF1	TF1
24	TF0	TF1	TF1	TF0	TF0	TF0
25	TF0	TF1	TF1	TF0	TF0	TF1
26	TF0	TF1	TF1	TF0	TF1	TF0
27	TF0	TF1	TF1	TF0	TF1	TF1
28	TF0	TF1	TF1	TF1	TF0	TF0
29	TF0	TF1	TF1	TF1	TF0	TF1
30	TF0	TF1	TF1	TF1	TF1	TF0
31	TF0	TF1	TF1	TF1	TF1	TF1
32	TF1	TF0	TF0	TF0	TF0	TF0
33	TF1	TF0	TF0	TF0	TF0	TF1
34	TF1	TF0	TF0	TF0	TF1	TF0
35	TF1	TF0	TF0	TF0	TF1	TF1
36	TF1	TF0	TF0	TF1	TF0	TF0
37	TF1	TF0	TF0	TF1	TF0	TF1
38	TF1	TF0	TF0	TF1	TF1	TF0
39	TF1	TF0	TF0	TF1	TF1	TF1
40	TF1	TF0	TF1	TF0	TF0	TF0
41	TF1	TF0	TF1	TF0	TF0	TF1
42	TF1	TF0	TF1	TF0	TF1	TF0
43	TF1	TF0	TF1	TF0	TF1	TF1
44	TF1	TF0	TF1	TF1	TF0	TF0
45	TF1	TF0	TF1	TF1	TF0	TF1
46	TF1	TF0	TF1	TF1	TF1	TF0
47	TF1	TF0	TF1	TF1	TF1	TF1
48	TF1	TF1	TF0	TF0	TF0	TF0
49	TF1	TF1	TF0	TF0	TF0	TF1
50	TF1	TF1	TF0	TF0	TF1	TF0
51	TF1	TF1	TF0	TF0	TF1	TF1
52	TF1	TF1	TF0	TF1	TF0	TF0
53	TF1	TF1	TF0	TF1	TF0	TF1
54	TF1	TF1	TF0	TF1	TF1	TF0
55	TF1	TF1	TF0	TF1	TF1	TF1
56	TF1	TF1	TF1	TF0	TF0	TF0
57	TF1	TF1	TF1	TF0	TF0	TF1
58	TF1	TF1	TF1	TF0	TF1	TF0
59	TF1	TF1	TF1	TF0	TF1	TF1
60	TF1	TF1	TF1	TF1	TF0	TF0
61	TF1	TF1	TF1	TF1	TF0	TF1
62	TF1	TF1	TF1	TF1	TF1	TF0
63	TF1	TF1	TF1	TF1	TF1	TF1

B.3.28.2 Transmitting ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) +DCCH

B.3.28.2.1Downlink setting

• DTCH(UDI)x5+DCCHx1

Slot Format(Symbol Rate)	Be sure to set to 15(480ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.28.2.2 Transmittable TFCS etc.

Transport Format Set								
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x640	0x148
		TF1,bits	4x640	4x640	4x640	4x640	4x640	1x148
	TTI,ms		40	40	40	40	40	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16
	RM attribute		170	170	170	170	170	160

Transport Format Combination Set						
TFCI	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF1	TF0	TF0
5	TF0	TF0	TF0	TF1	TF0	TF1
6	TF0	TF0	TF0	TF1	TF1	TF0
7	TF0	TF0	TF0	TF1	TF1	TF1
8	TF0	TF0	TF1	TF0	TF0	TF0
9	TF0	TF0	TF1	TF0	TF0	TF1
10	TF0	TF0	TF1	TF0	TF1	TF0
11	TF0	TF0	TF1	TF0	TF1	TF1
12	TF0	TF0	TF1	TF1	TF0	TF0
13	TF0	TF0	TF1	TF1	TF0	TF1
14	TF0	TF0	TF1	TF1	TF1	TF0
15	TF0	TF0	TF1	TF1	TF1	TF1
16	TF0	TF1	TF0	TF0	TF0	TF0
17	TF0	TF1	TF0	TF0	TF0	TF1
18	TF0	TF1	TF0	TF0	TF1	TF0
19	TF0	TF1	TF0	TF0	TF1	TF1
20	TF0	TF1	TF0	TF1	TF0	TF0
21	TF0	TF1	TF0	TF1	TF0	TF1
22	TF0	TF1	TF0	TF1	TF1	TF0
23	TF0	TF1	TF0	TF1	TF1	TF1

24	TF0	TF1	TF1	TF0	TF0	TF0
25	TF0	TF1	TF1	TF0	TF0	TF1
26	TF0	TF1	TF1	TF0	TF1	TF0
27	TF0	TF1	TF1	TF0	TF1	TF1
28	TF0	TF1	TF1	TF1	TF0	TF0
29	TF0	TF1	TF1	TF1	TF0	TF1
30	TF0	TF1	TF1	TF1	TF1	TF0
31	TF0	TF1	TF1	TF1	TF1	TF1
32	TF1	TF0	TF0	TF0	TF0	TF0
33	TF1	TF0	TF0	TF0	TF0	TF1
34	TF1	TF0	TF0	TF0	TF1	TF0
35	TF1	TF0	TF0	TF0	TF1	TF1
36	TF1	TF0	TF0	TF1	TF0	TF0
37	TF1	TF0	TF0	TF1	TF0	TF1
38	TF1	TF0	TF0	TF1	TF1	TF0
39	TF1	TF0	TF0	TF1	TF1	TF1
40	TF1	TF0	TF1	TF0	TF0	TF0
41	TF1	TF0	TF1	TF0	TF0	TF1
42	TF1	TF0	TF1	TF0	TF1	TF0
43	TF1	TF0	TF1	TF0	TF1	TF1
44	TF1	TF0	TF1	TF1	TF0	TF0
45	TF1	TF0	TF1	TF1	TF0	TF1
46	TF1	TF0	TF1	TF1	TF1	TF0
47	TF1	TF0	TF1	TF1	TF1	TF1
48	TF1	TF1	TF0	TF0	TF0	TF0
49	TF1	TF1	TF0	TF0	TF0	TF1
50	TF1	TF1	TF0	TF0	TF1	TF0
51	TF1	TF1	TF0	TF0	TF1	TF1
52	TF1	TF1	TF0	TF1	TF0	TF0
53	TF1	TF1	TF0	TF1	TF0	TF1
54	TF1	TF1	TF0	TF1	TF1	TF0
55	TF1	TF1	TF0	TF1	TF1	TF1
56	TF1	TF1	TF1	TF0	TF0	TF0
57	TF1	TF1	TF1	TF0	TF0	TF1
58	TF1	TF1	TF1	TF0	TF1	TF0
59	TF1	TF1	TF1	TF0	TF1	TF1
60	TF1	TF1	TF1	TF1	TF0	TF0
61	TF1	TF1	TF1	TF1	TF0	TF1
62	TF1	TF1	TF1	TF1	TF1	TF0
63	TF1	TF1	TF1	TF1	TF1	TF1

B.3.28.2.3 Transmit procedure for each TFCI

TFCI	"DCH Ch No.?" in DCH_Data(1B-6B).dll					DCH_Data (DCCH).dll "TTI?"
	The 1st time	The 2nd time	The 3rd time	The 4th time	The 5th time	
0	–	–	–	–	–	–
1	–	–	–	–	–	"40"
2	"4"	–	–	–	–	–
3	"4"	–	–	–	–	"40"
4	–	"3"	–	–	–	–
5	–	"3"	–	–	–	"40"
6	"4"	"3"	–	–	–	–
7	"4"	"3"	–	–	–	"40"
8	–	–	"2"	–	–	–
9	–	–	"2"	–	–	"40"
10	"4"	–	"2"	–	–	–
11	"4"	–	"2"	–	–	"40"
12	–	"3"	"2"	–	–	–
13	–	"3"	"2"	–	–	"40"
14	"4"	"3"	"2"	–	–	–
15	"4"	"3"	"2"	–	–	"40"
16	–	–	–	"1"	–	–
17	–	–	–	"1"	–	"40"
18	"4"	–	–	"1"	–	–
19	"4"	–	–	"1"	–	"40"
20	–	"3"	–	"1"	–	–
21	–	"3"	–	"1"	–	"40"
22	"4"	"3"	–	"1"	–	–
23	"4"	"3"	–	"1"	–	"40"
24	–	–	"2"	"1"	–	–
25	–	–	"2"	"1"	–	"40"
26	"4"	–	"2"	"1"	–	–
27	"4"	–	"2"	"1"	–	"40"
28	–	"3"	"2"	"1"	–	–
29	–	"3"	"2"	"1"	–	"40"
30	"4"	"3"	"2"	"1"	–	–

31	“4”	“3”	“2”	“1”	–	“40”
32	–	–	–	–	“0”	–
33	–	–	–	–	“0”	“40”
34	“4”	–	–	–	“0”	–
35	“4”	–	–	–	“0”	“40”
36	–	“3”	–	–	“0”	–
37	–	“3”	–	–	“0”	“40”
38	“4”	“3”	–	–	“0”	–
39	“4”	“3”	–	–	“0”	“40”
40	–	–	“2”	–	“0”	–
41	–	–	“2”	–	“0”	“40”
42	“4”	–	“2”	–	“0”	–
43	“4”	–	“2”	–	“0”	“40”
44	–	“3”	“2”	–	“0”	–
45	–	“3”	“2”	–	“0”	“40”
46	“4”	“3”	“2”	–	“0”	–
47	“4”	“3”	“2”	–	“0”	“40”
48	–	–	–	“1”	“0”	–
49	–	–	–	“1”	“0”	“40”
50	“4”	–	–	“1”	“0”	–
51	“4”	–	–	“1”	“0”	“40”
52	–	“3”	–	“1”	“0”	–
53	–	“3”	–	“1”	“0”	“40”
54	“4”	“3”	–	“1”	“0”	–
55	“4”	“3”	–	“1”	“0”	“40”
56	–	–	“2”	“1”	“0”	–
57	–	–	“2”	“1”	“0”	“40”
58	“4”	–	“2”	“1”	“0”	–
59	“4”	–	“2”	“1”	“0”	“40”
60	–	“3”	“2”	“1”	“0”	–
61	–	“3”	“2”	“1”	“0”	“40”
62	“4”	“3”	“2”	“1”	“0”	–
63	“4”	“3”	“2”	“1”	“0”	“40”

B.3.29 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH Setting

Start_6B.dll is used.

B.3.29.1 Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH

B.3.29.1.1 Uplink setting

• DTCH(UDI)x6+DCCHx1

Slot Format	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 960ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.64

B.3.29.1.2 Receivable TFCS etc.

Transport Format Set									
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x640	0x640	0x148
		TF1,bits	4x640	4x640	4x640	4x640	4x640	4x640	1x148
	TTI,ms		40	40	40	40	40	40	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16	16
	RM attribute		170	170	170	170	170	170	160

Transport Format Combination Set							
TFCI	DCH	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF0	TF1	TF0	TF0
5	TF0	TF0	TF0	TF0	TF1	TF0	TF1
6	TF0	TF0	TF0	TF0	TF1	TF1	TF0
7	TF0	TF0	TF0	TF0	TF1	TF1	TF1
8	TF0	TF0	TF0	TF1	TF0	TF0	TF0
9	TF0	TF0	TF0	TF1	TF0	TF0	TF1
10	TF0	TF0	TF0	TF1	TF0	TF1	TF0
11	TF0	TF0	TF0	TF1	TF0	TF1	TF1
12	TF0	TF0	TF0	TF1	TF1	TF0	TF0
13	TF0	TF0	TF0	TF1	TF1	TF0	TF1
14	TF0	TF0	TF0	TF1	TF1	TF1	TF0
15	TF0	TF0	TF0	TF1	TF1	TF1	TF1
16	TF0	TF0	TF1	TF0	TF0	TF0	TF0

17	TF0	TF0	TF1	TF0	TF0	TF0	TF1
18	TF0	TF0	TF1	TF0	TF0	TF1	TF0
19	TF0	TF0	TF1	TF0	TF0	TF1	TF1
20	TF0	TF0	TF1	TF0	TF1	TF0	TF0
21	TF0	TF0	TF1	TF0	TF1	TF0	TF1
22	TF0	TF0	TF1	TF0	TF1	TF1	TF0
23	TF0	TF0	TF1	TF0	TF1	TF1	TF1
24	TF0	TF0	TF1	TF1	TF0	TF0	TF0
25	TF0	TF0	TF1	TF1	TF0	TF0	TF1
26	TF0	TF0	TF1	TF1	TF0	TF1	TF0
27	TF0	TF0	TF1	TF1	TF0	TF1	TF1
28	TF0	TF0	TF1	TF1	TF1	TF0	TF0
29	TF0	TF0	TF1	TF1	TF1	TF0	TF1
30	TF0	TF0	TF1	TF1	TF1	TF1	TF0
31	TF0	TF0	TF1	TF1	TF1	TF1	TF1
32	TF0	TF1	TF0	TF0	TF0	TF0	TF0
33	TF0	TF1	TF0	TF0	TF0	TF0	TF1
34	TF0	TF1	TF0	TF0	TF0	TF1	TF0
35	TF0	TF1	TF0	TF0	TF0	TF1	TF1
36	TF0	TF1	TF0	TF0	TF1	TF0	TF0
37	TF0	TF1	TF0	TF0	TF1	TF0	TF1
38	TF0	TF1	TF0	TF0	TF1	TF1	TF0
39	TF0	TF1	TF0	TF0	TF1	TF1	TF1
40	TF0	TF1	TF0	TF1	TF0	TF0	TF0
41	TF0	TF1	TF0	TF1	TF0	TF0	TF1
42	TF0	TF1	TF0	TF1	TF0	TF1	TF0
43	TF0	TF1	TF0	TF1	TF0	TF1	TF1
44	TF0	TF1	TF0	TF1	TF1	TF0	TF0
45	TF0	TF1	TF0	TF1	TF1	TF0	TF1
46	TF0	TF1	TF0	TF1	TF1	TF1	TF0
47	TF0	TF1	TF0	TF1	TF1	TF1	TF1
48	TF0	TF1	TF1	TF0	TF0	TF0	TF0
49	TF0	TF1	TF1	TF0	TF0	TF0	TF1
50	TF0	TF1	TF1	TF0	TF0	TF1	TF0
51	TF0	TF1	TF1	TF0	TF0	TF1	TF1
52	TF0	TF1	TF1	TF0	TF1	TF0	TF0
53	TF0	TF1	TF1	TF0	TF1	TF0	TF1
54	TF0	TF1	TF1	TF0	TF1	TF1	TF0
55	TF0	TF1	TF1	TF0	TF1	TF1	TF1
56	TF0	TF1	TF1	TF1	TF0	TF0	TF0
57	TF0	TF1	TF1	TF1	TF0	TF0	TF1
58	TF0	TF1	TF1	TF1	TF0	TF1	TF0
59	TF0	TF1	TF1	TF1	TF0	TF1	TF1
60	TF0	TF1	TF1	TF1	TF1	TF0	TF0
61	TF0	TF1	TF1	TF1	TF1	TF0	TF1
62	TF0	TF1	TF1	TF1	TF1	TF1	TF0
63	TF0	TF1	TF1	TF1	TF1	TF1	TF1
64	TF1	TF0	TF0	TF0	TF0	TF0	TF0
65	TF1	TF0	TF0	TF0	TF0	TF0	TF1
66	TF1	TF0	TF0	TF0	TF0	TF1	TF0

67	TF1	TF0	TF0	TF0	TF0	TF1	TF1
68	TF1	TF0	TF0	TF0	TF1	TF0	TF0
69	TF1	TF0	TF0	TF0	TF1	TF0	TF1
70	TF1	TF0	TF0	TF0	TF1	TF1	TF0
71	TF1	TF0	TF0	TF0	TF1	TF1	TF1
72	TF1	TF0	TF0	TF1	TF0	TF0	TF0
73	TF1	TF0	TF0	TF1	TF0	TF0	TF1
74	TF1	TF0	TF0	TF1	TF0	TF1	TF0
75	TF1	TF0	TF0	TF1	TF0	TF1	TF1
76	TF1	TF0	TF0	TF1	TF1	TF0	TF0
77	TF1	TF0	TF0	TF1	TF1	TF0	TF1
78	TF1	TF0	TF0	TF1	TF1	TF1	TF0
79	TF1	TF0	TF0	TF1	TF1	TF1	TF1
80	TF1	TF0	TF1	TF0	TF0	TF0	TF0
81	TF1	TF0	TF1	TF0	TF0	TF0	TF1
82	TF1	TF0	TF1	TF0	TF0	TF1	TF0
83	TF1	TF0	TF1	TF0	TF0	TF1	TF1
84	TF1	TF0	TF1	TF0	TF1	TF0	TF0
85	TF1	TF0	TF1	TF0	TF1	TF0	TF1
86	TF1	TF0	TF1	TF0	TF1	TF1	TF0
87	TF1	TF0	TF1	TF0	TF1	TF1	TF1
88	TF1	TF0	TF1	TF1	TF0	TF0	TF0
89	TF1	TF0	TF1	TF1	TF0	TF0	TF1
90	TF1	TF0	TF1	TF1	TF0	TF1	TF0
91	TF1	TF0	TF1	TF1	TF0	TF1	TF1
92	TF1	TF0	TF1	TF1	TF1	TF0	TF0
93	TF1	TF0	TF1	TF1	TF1	TF0	TF1
94	TF1	TF0	TF1	TF1	TF1	TF1	TF0
95	TF1	TF0	TF1	TF1	TF1	TF1	TF1
96	TF1	TF1	TF0	TF0	TF0	TF0	TF0
97	TF1	TF1	TF0	TF0	TF0	TF0	TF1
98	TF1	TF1	TF0	TF0	TF0	TF1	TF0
99	TF1	TF1	TF0	TF0	TF0	TF1	TF1
100	TF1	TF1	TF0	TF0	TF1	TF0	TF0
101	TF1	TF1	TF0	TF0	TF1	TF0	TF1
102	TF1	TF1	TF0	TF0	TF1	TF1	TF0
103	TF1	TF1	TF0	TF0	TF1	TF1	TF1
104	TF1	TF1	TF0	TF1	TF0	TF0	TF0
105	TF1	TF1	TF0	TF1	TF0	TF0	TF1
106	TF1	TF1	TF0	TF1	TF0	TF1	TF0
107	TF1	TF1	TF0	TF1	TF0	TF1	TF1
108	TF1	TF1	TF0	TF1	TF1	TF0	TF0
109	TF1	TF1	TF0	TF1	TF1	TF0	TF1
110	TF1	TF1	TF0	TF1	TF1	TF1	TF0
111	TF1	TF1	TF0	TF1	TF1	TF1	TF1
112	TF1	TF1	TF1	TF0	TF0	TF0	TF0
113	TF1	TF1	TF1	TF0	TF0	TF0	TF1
114	TF1	TF1	TF1	TF0	TF0	TF1	TF0
115	TF1	TF1	TF1	TF0	TF0	TF1	TF1
116	TF1	TF1	TF1	TF0	TF1	TF0	TF0

117	TF1	TF1	TF1	TF0	TF1	TF0	TF1
118	TF1	TF1	TF1	TF0	TF1	TF1	TF0
119	TF1	TF1	TF1	TF0	TF1	TF1	TF1
120	TF1	TF1	TF1	TF1	TF0	TF0	TF0
121	TF1	TF1	TF1	TF1	TF0	TF0	TF1
122	TF1	TF1	TF1	TF1	TF0	TF1	TF0
123	TF1	TF1	TF1	TF1	TF0	TF1	TF1
124	TF1	TF1	TF1	TF1	TF1	TF0	TF0
125	TF1	TF1	TF1	TF1	TF1	TF0	TF1
126	TF1	TF1	TF1	TF1	TF1	TF1	TF0
127	TF1	TF1	TF1	TF1	TF1	TF1	TF1

B.3.29.2 Transmitting ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps)+ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + DCCH

B.3.29.2.1 Downlink setting

• DTCH(UDI)x6+DCCHx1

Slot Format(Symbol Rate)	Be sure to set to 15(480ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.29.2.2 Transmittable TFCS etc.

Transport Format Set									
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x640	0x640	0x148
		TF1,bits	4x640	4x640	4x640	4x640	4x640	4x640	1x148
	TTI,ms		40	40	40	40	40	40	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16	16
	RM attribute		170	170	170	170	170	170	160

Transport Format Combination Set							
TFCI	DCH	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF0	TF1	TF0	TF0
5	TF0	TF0	TF0	TF0	TF1	TF0	TF1
6	TF0	TF0	TF0	TF0	TF1	TF1	TF0
7	TF0	TF0	TF0	TF0	TF1	TF1	TF1
8	TF0	TF0	TF0	TF1	TF0	TF0	TF0
9	TF0	TF0	TF0	TF1	TF0	TF0	TF1
10	TF0	TF0	TF0	TF1	TF0	TF1	TF0
11	TF0	TF0	TF0	TF1	TF0	TF1	TF1
12	TF0	TF0	TF0	TF1	TF1	TF0	TF0
13	TF0	TF0	TF0	TF1	TF1	TF0	TF1
14	TF0	TF0	TF0	TF1	TF1	TF1	TF0
15	TF0	TF0	TF0	TF1	TF1	TF1	TF1
16	TF0	TF0	TF1	TF0	TF0	TF0	TF0
17	TF0	TF0	TF1	TF0	TF0	TF0	TF1
18	TF0	TF0	TF1	TF0	TF0	TF1	TF0
19	TF0	TF0	TF1	TF0	TF0	TF1	TF1
20	TF0	TF0	TF1	TF0	TF1	TF0	TF0
21	TF0	TF0	TF1	TF0	TF1	TF0	TF1
22	TF0	TF0	TF1	TF0	TF1	TF1	TF0

23	TF0	TF0	TF1	TF0	TF1	TF1	TF1
24	TF0	TF0	TF1	TF1	TF0	TF0	TF0
25	TF0	TF0	TF1	TF1	TF0	TF0	TF1
26	TF0	TF0	TF1	TF1	TF0	TF1	TF0
27	TF0	TF0	TF1	TF1	TF0	TF1	TF1
28	TF0	TF0	TF1	TF1	TF1	TF0	TF0
29	TF0	TF0	TF1	TF1	TF1	TF0	TF1
30	TF0	TF0	TF1	TF1	TF1	TF1	TF0
31	TF0	TF0	TF1	TF1	TF1	TF1	TF1
32	TF0	TF1	TF0	TF0	TF0	TF0	TF0
33	TF0	TF1	TF0	TF0	TF0	TF0	TF1
34	TF0	TF1	TF0	TF0	TF0	TF1	TF0
35	TF0	TF1	TF0	TF0	TF0	TF1	TF1
36	TF0	TF1	TF0	TF0	TF1	TF0	TF0
37	TF0	TF1	TF0	TF0	TF1	TF0	TF1
38	TF0	TF1	TF0	TF0	TF1	TF1	TF0
39	TF0	TF1	TF0	TF0	TF1	TF1	TF1
40	TF0	TF1	TF0	TF1	TF0	TF0	TF0
41	TF0	TF1	TF0	TF1	TF0	TF0	TF1
42	TF0	TF1	TF0	TF1	TF0	TF1	TF0
43	TF0	TF1	TF0	TF1	TF0	TF1	TF1
44	TF0	TF1	TF0	TF1	TF1	TF0	TF0
45	TF0	TF1	TF0	TF1	TF1	TF0	TF1
46	TF0	TF1	TF0	TF1	TF1	TF1	TF0
47	TF0	TF1	TF0	TF1	TF1	TF1	TF1
48	TF0	TF1	TF1	TF0	TF0	TF0	TF0
49	TF0	TF1	TF1	TF0	TF0	TF0	TF1
50	TF0	TF1	TF1	TF0	TF0	TF1	TF0
51	TF0	TF1	TF1	TF0	TF0	TF1	TF1
52	TF0	TF1	TF1	TF0	TF1	TF0	TF0
53	TF0	TF1	TF1	TF0	TF1	TF0	TF1
54	TF0	TF1	TF1	TF0	TF1	TF1	TF0
55	TF0	TF1	TF1	TF0	TF1	TF1	TF1
56	TF0	TF1	TF1	TF1	TF0	TF0	TF0
57	TF0	TF1	TF1	TF1	TF0	TF0	TF1
58	TF0	TF1	TF1	TF1	TF0	TF1	TF0
59	TF0	TF1	TF1	TF1	TF0	TF1	TF1
60	TF0	TF1	TF1	TF1	TF1	TF0	TF0
61	TF0	TF1	TF1	TF1	TF1	TF0	TF1
62	TF0	TF1	TF1	TF1	TF1	TF1	TF0
63	TF0	TF1	TF1	TF1	TF1	TF1	TF1
64	TF1	TF0	TF0	TF0	TF0	TF0	TF0
65	TF1	TF0	TF0	TF0	TF0	TF0	TF1
66	TF1	TF0	TF0	TF0	TF0	TF1	TF0
67	TF1	TF0	TF0	TF0	TF0	TF1	TF1
68	TF1	TF0	TF0	TF0	TF1	TF0	TF0
69	TF1	TF0	TF0	TF0	TF1	TF0	TF1
70	TF1	TF0	TF0	TF0	TF1	TF1	TF0
71	TF1	TF0	TF0	TF0	TF1	TF1	TF1
72	TF1	TF0	TF0	TF1	TF0	TF0	TF0

73	TF1	TF0	TF0	TF1	TF0	TF0	TF1
74	TF1	TF0	TF0	TF1	TF0	TF1	TF0
75	TF1	TF0	TF0	TF1	TF0	TF1	TF1
76	TF1	TF0	TF0	TF1	TF1	TF0	TF0
77	TF1	TF0	TF0	TF1	TF1	TF0	TF1
78	TF1	TF0	TF0	TF1	TF1	TF1	TF0
79	TF1	TF0	TF0	TF1	TF1	TF1	TF1
80	TF1	TF0	TF1	TF0	TF0	TF0	TF0
81	TF1	TF0	TF1	TF0	TF0	TF0	TF1
82	TF1	TF0	TF1	TF0	TF0	TF1	TF0
83	TF1	TF0	TF1	TF0	TF0	TF1	TF1
84	TF1	TF0	TF1	TF0	TF1	TF0	TF0
85	TF1	TF0	TF1	TF0	TF1	TF0	TF1
86	TF1	TF0	TF1	TF0	TF1	TF1	TF0
87	TF1	TF0	TF1	TF0	TF1	TF1	TF1
88	TF1	TF0	TF1	TF1	TF0	TF0	TF0
89	TF1	TF0	TF1	TF1	TF0	TF0	TF1
90	TF1	TF0	TF1	TF1	TF0	TF1	TF0
91	TF1	TF0	TF1	TF1	TF0	TF1	TF1
92	TF1	TF0	TF1	TF1	TF1	TF0	TF0
93	TF1	TF0	TF1	TF1	TF1	TF0	TF1
94	TF1	TF0	TF1	TF1	TF1	TF1	TF0
95	TF1	TF0	TF1	TF1	TF1	TF1	TF1
96	TF1	TF1	TF0	TF0	TF0	TF0	TF0
97	TF1	TF1	TF0	TF0	TF0	TF0	TF1
98	TF1	TF1	TF0	TF0	TF0	TF1	TF0
99	TF1	TF1	TF0	TF0	TF0	TF1	TF1
100	TF1	TF1	TF0	TF0	TF1	TF0	TF0
101	TF1	TF1	TF0	TF0	TF1	TF0	TF1
102	TF1	TF1	TF0	TF0	TF1	TF1	TF0
103	TF1	TF1	TF0	TF0	TF1	TF1	TF1
104	TF1	TF1	TF0	TF1	TF0	TF0	TF0
105	TF1	TF1	TF0	TF1	TF0	TF0	TF1
106	TF1	TF1	TF0	TF1	TF0	TF1	TF0
107	TF1	TF1	TF0	TF1	TF0	TF1	TF1
108	TF1	TF1	TF0	TF1	TF1	TF0	TF0
109	TF1	TF1	TF0	TF1	TF1	TF0	TF1
110	TF1	TF1	TF0	TF1	TF1	TF1	TF0
111	TF1	TF1	TF0	TF1	TF1	TF1	TF1
112	TF1	TF1	TF1	TF0	TF0	TF0	TF0
113	TF1	TF1	TF1	TF0	TF0	TF0	TF1
114	TF1	TF1	TF1	TF0	TF0	TF1	TF0
115	TF1	TF1	TF1	TF0	TF0	TF1	TF1
116	TF1	TF1	TF1	TF0	TF1	TF0	TF0
117	TF1	TF1	TF1	TF0	TF1	TF0	TF1
118	TF1	TF1	TF1	TF0	TF1	TF1	TF0
119	TF1	TF1	TF1	TF0	TF1	TF1	TF1
120	TF1	TF1	TF1	TF1	TF0	TF0	TF0
121	TF1	TF1	TF1	TF1	TF0	TF0	TF1
122	TF1	TF1	TF1	TF1	TF0	TF1	TF0

123	TF1	TF1	TF1	TF1	TF0	TF1	TF1
124	TF1	TF1	TF1	TF1	TF1	TF0	TF0
125	TF1	TF1	TF1	TF1	TF1	TF0	TF1
126	TF1	TF1	TF1	TF1	TF1	TF1	TF0
127	TF1	TF1	TF1	TF1	TF1	TF1	TF1

B.3.29.2.3 Transmit procedure for each TFCI

TFCI	"DCH Ch No.?" in DCH_Data(1B-6B).dll						DCH_Data (DCCH).dll "TTI?"
	The 1st time	The 2nd time	The 3rd time	The 4th time	The 5th time	The 6th time	
0	–	–	–	–	–	–	–
1	–	–	–	–	–	–	"40"
2	"5"	–	–	–	–	–	–
3	"5"	–	–	–	–	–	"40"
4	–	"4"	–	–	–	–	–
5	–	"4"	–	–	–	–	"40"
6	"5"	"4"	–	–	–	–	–
7	"5"	"4"	–	–	–	–	"40"
8	–	–	"3"	–	–	–	–
9	–	–	"3"	–	–	–	"40"
10	"5"	–	"3"	–	–	–	–
11	"5"	–	"3"	–	–	–	"40"
12	–	"4"	"3"	–	–	–	–
13	–	"4"	"3"	–	–	–	"40"
14	"5"	"4"	"3"	–	–	–	–
15	"5"	"4"	"3"	–	–	–	"40"
16	–	–	–	"2"	–	–	–
17	–	–	–	"2"	–	–	"40"
18	"5"	–	–	"2"	–	–	–
19	"5"	–	–	"2"	–	–	"40"
20	–	"4"	–	"2"	–	–	–
21	–	"4"	–	"2"	–	–	"40"
22	"5"	"4"	–	"2"	–	–	–
23	"5"	"4"	–	"2"	–	–	"40"
24	–	–	"3"	"2"	–	–	–
25	–	–	"3"	"2"	–	–	"40"
26	"5"	–	"3"	"2"	–	–	–

27	“5”	–	“3”	“2”	–	–	“40”
28	–	“4”	“3”	“2”	–	–	–
29	–	“4”	“3”	“2”	–	–	“40”
30	“5”	“4”	“3”	“2”	–	–	–
31	“5”	“4”	“3”	“2”	–	–	“40”
32	–	–	–	–	“1”	–	–
33	–	–	–	–	“1”	–	“40”
34	“5”	–	–	–	“1”	–	–
35	“5”	–	–	–	“1”	–	“40”
36	–	“4”	–	–	“1”	–	–
37	–	“4”	–	–	“1”	–	“40”
38	“5”	“4”	–	–	“1”	–	–
39	“5”	“4”	–	–	“1”	–	“40”
40	–	–	“3”	–	“1”	–	–
41	–	–	“3”	–	“1”	–	“40”
42	“5”	–	“3”	–	“1”	–	–
43	“5”	–	“3”	–	“1”	–	“40”
44	–	“4”	“3”	–	“1”	–	–
45	–	“4”	“3”	–	“1”	–	“40”
46	“5”	“4”	“3”	–	“1”	–	–
47	“5”	“4”	“3”	–	“1”	–	“40”
48	–	–	–	“2”	“1”	–	–
49	–	–	–	“2”	“1”	–	“40”
50	“5”	–	–	“2”	“1”	–	–
51	“5”	–	–	“2”	“1”	–	“40”
52	–	“4”	–	“2”	“1”	–	–
53	–	“4”	–	“2”	“1”	–	“40”
54	“5”	“4”	–	“2”	“1”	–	–
55	“5”	“4”	–	“2”	“1”	–	“40”
56	–	–	“3”	“2”	“1”	–	–
57	–	–	“3”	“2”	“1”	–	“40”
58	“5”	–	“3”	“2”	“1”	–	–
59	“5”	–	“3”	“2”	“1”	–	“40”
60	–	“4”	“3”	“2”	“1”	–	–
61	–	“4”	“3”	“2”	“1”	–	“40”
62	“5”	“4”	“3”	“2”	“1”	–	–

63	"5"	"4"	"3"	"2"	"1"	–	"40"
64	–	–	–	–	–	"0"	–
65	–	–	–	–	–	"0"	"40"
66	"5"	–	–	–	–	"0"	–
67	"5"	–	–	–	–	"0"	"40"
68	–	"4"	–	–	–	"0"	–
69	–	"4"	–	–	–	"0"	"40"
70	"5"	"4"	–	–	–	"0"	–
71	"5"	"4"	–	–	–	"0"	"40"
72	–	–	"3"	–	–	"0"	–
73	–	–	"3"	–	–	"0"	"40"
74	"5"	–	"3"	–	–	"0"	–
75	"5"	–	"3"	–	–	"0"	"40"
76	–	"4"	"3"	–	–	"0"	–
77	–	"4"	"3"	–	–	"0"	"40"
78	"5"	"4"	"3"	–	–	"0"	–
79	"5"	"4"	"3"	–	–	"0"	"40"
80	–	–	–	"2"	–	"0"	–
81	–	–	–	"2"	–	"0"	"40"
82	"5"	–	–	"2"	–	"0"	–
83	"5"	–	–	"2"	–	"0"	"40"
84	–	"4"	–	"2"	–	"0"	–
85	–	"4"	–	"2"	–	"0"	"40"
86	"5"	"4"	–	"2"	–	"0"	–
87	"5"	"4"	–	"2"	–	"0"	"40"
88	–	–	"3"	"2"	–	"0"	–
89	–	–	"3"	"2"	–	"0"	"40"
90	"5"	–	"3"	"2"	–	"0"	–
91	"5"	–	"3"	"2"	–	"0"	"40"
92	–	"4"	"3"	"2"	–	"0"	–
93	–	"4"	"3"	"2"	–	"0"	"40"
94	"5"	"4"	"3"	"2"	–	"0"	–
95	"5"	"4"	"3"	"2"	–	"0"	"40"
96	–	–	–	–	"1"	"0"	–
97	–	–	–	–	"1"	"0"	"40"
98	"5"	–	–	–	"1"	"0"	–

99	“5”	–	–	–	“1”	“0”	“40”
100	–	“4”	–	–	“1”	“0”	–
101	–	“4”	–	–	“1”	“0”	“40”
102	“5”	“4”	–	–	“1”	“0”	–
103	“5”	“4”	–	–	“1”	“0”	“40”
104	–	–	“3”	–	“1”	“0”	–
105	–	–	“3”	–	“1”	“0”	“40”
106	“5”	–	“3”	–	“1”	“0”	–
107	“5”	–	“3”	–	“1”	“0”	“40”
108	–	“4”	“3”	–	“1”	“0”	–
109	–	“4”	“3”	–	“1”	“0”	“40”
110	“5”	“4”	“3”	–	“1”	“0”	–
111	“5”	“4”	“3”	–	“1”	“0”	“40”
112	–	–	–	“2”	“1”	“0”	–
113	–	–	–	“2”	“1”	“0”	“40”
114	“5”	–	–	“2”	“1”	“0”	–
115	“5”	–	–	“2”	“1”	“0”	“40”
116	–	“4”	–	“2”	“1”	“0”	–
117	–	“4”	–	“2”	“1”	“0”	“40”
118	“5”	“4”	–	“2”	“1”	“0”	–
119	“5”	“4”	–	“2”	“1”	“0”	“40”
120	–	–	“3”	“2”	“1”	“0”	–
121	–	–	“3”	“2”	“1”	“0”	“40”
122	“5”	–	“3”	“2”	“1”	“0”	–
123	“5”	–	“3”	“2”	“1”	“0”	“40”
124	–	“4”	“3”	“2”	“1”	“0”	–
125	–	“4”	“3”	“2”	“1”	“0”	“40”
126	“5”	“4”	“3”	“2”	“1”	“0”	–
127	“5”	“4”	“3”	“2”	“1”	“0”	“40”

B.3.30 ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet [64kbps] + DCCH Setting

Start_5BxP64K.dll is used.

B.3.30.1 Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 64[kbps] + DCCH

B.3.30.1.1 Uplink setting

• DTCH(UDI)x5+DTCH(Packet)+DCCHx1

Slot Format	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 960ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.64

B.3.30.1.2 Receivable TFCS etc.

Transport Format Set									
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x640	0x336	0x148
		TF1,bits	4x640	4x640	4x640	4x640	4x640	1x336	1x148
		TF2,bits	-	-	-	-	-	2x336	-
		TF3,bits	-	-	-	-	-	3x336	-
		TF4,bits	-	-	-	-	-	4x336	-
	TTI,ms		40	40	40	40	40	20	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16	16
	RM attribute		170	170	170	170	170	140	160

Transport Format Combination Set							
TFCI	DCH	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF0	TF0	TF2	TF1
6	TF0	TF0	TF0	TF0	TF0	TF3	TF0
7	TF0	TF0	TF0	TF0	TF0	TF3	TF1
8	TF0	TF0	TF0	TF0	TF0	TF4	TF0
9	TF0	TF0	TF0	TF0	TF0	TF4	TF1
10	TF0	TF0	TF0	TF0	TF1	TF0	TF0
11	TF0	TF0	TF0	TF0	TF1	TF0	TF1
12	TF0	TF0	TF0	TF0	TF1	TF1	TF0

13	TF0	TF0	TF0	TF0	TF1	TF1	TF1
14	TF0	TF0	TF0	TF0	TF1	TF2	TF0
15	TF0	TF0	TF0	TF0	TF1	TF2	TF1
16	TF0	TF0	TF0	TF0	TF1	TF3	TF0
17	TF0	TF0	TF0	TF0	TF1	TF3	TF1
18	TF0	TF0	TF0	TF0	TF1	TF4	TF0
19	TF0	TF0	TF0	TF0	TF1	TF4	TF1
20	TF0	TF0	TF0	TF1	TF1	TF0	TF0
21	TF0	TF0	TF0	TF1	TF1	TF0	TF1
22	TF0	TF0	TF0	TF1	TF1	TF1	TF0
23	TF0	TF0	TF0	TF1	TF1	TF1	TF1
24	TF0	TF0	TF0	TF1	TF1	TF2	TF0
25	TF0	TF0	TF0	TF1	TF1	TF2	TF1
26	TF0	TF0	TF0	TF1	TF1	TF3	TF0
27	TF0	TF0	TF0	TF1	TF1	TF3	TF1
28	TF0	TF0	TF0	TF1	TF1	TF4	TF0
29	TF0	TF0	TF0	TF1	TF1	TF4	TF1
30	TF0	TF0	TF1	TF1	TF1	TF0	TF0
31	TF0	TF0	TF1	TF1	TF1	TF0	TF1
32	TF0	TF0	TF1	TF1	TF1	TF1	TF0
33	TF0	TF0	TF1	TF1	TF1	TF1	TF1
34	TF0	TF0	TF1	TF1	TF1	TF2	TF0
35	TF0	TF0	TF1	TF1	TF1	TF2	TF1
36	TF0	TF0	TF1	TF1	TF1	TF3	TF0
37	TF0	TF0	TF1	TF1	TF1	TF3	TF1
38	TF0	TF0	TF1	TF1	TF1	TF4	TF0
39	TF0	TF0	TF1	TF1	TF1	TF4	TF1
40	TF0	TF1	TF1	TF1	TF1	TF0	TF0
41	TF0	TF1	TF1	TF1	TF1	TF0	TF1
42	TF0	TF1	TF1	TF1	TF1	TF1	TF0
43	TF0	TF1	TF1	TF1	TF1	TF1	TF1
44	TF0	TF1	TF1	TF1	TF1	TF2	TF0
45	TF0	TF1	TF1	TF1	TF1	TF2	TF1
46	TF0	TF1	TF1	TF1	TF1	TF3	TF0
47	TF0	TF1	TF1	TF1	TF1	TF3	TF1
48	TF0	TF1	TF1	TF1	TF1	TF4	TF0
49	TF0	TF1	TF1	TF1	TF1	TF4	TF1
50	TF1	TF1	TF1	TF1	TF1	TF0	TF0
51	TF1	TF1	TF1	TF1	TF1	TF0	TF1
52	TF1	TF1	TF1	TF1	TF1	TF1	TF0
53	TF1	TF1	TF1	TF1	TF1	TF1	TF1
54	TF1	TF1	TF1	TF1	TF1	TF2	TF0
55	TF1	TF1	TF1	TF1	TF1	TF2	TF1
56	TF1	TF1	TF1	TF1	TF1	TF3	TF0
57	TF1	TF1	TF1	TF1	TF1	TF3	TF1
58	TF1	TF1	TF1	TF1	TF1	TF4	TF0
59	TF1	TF1	TF1	TF1	TF1	TF4	TF1

B.3.30.2 Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 64[kbps] + DCCH

B.3.30.2.1 Downlink setting

• DTCH(UDI)x5+DTCH(Packet)+DCCHx1

Slot Format(Symbol Rate)	Be sure to set to 15(480ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.30.2.2 Transmittable TFCS etc.

Transport Format Set									
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x640	0x336	0x148
		TF1,bits	4x640	4x640	4x640	4x640	4x640	1x336	1x148
		TF2,bits	-	-	-	-	-	2x336	-
		TF3,bits	-	-	-	-	-	3x336	-
		TF4,bits	-	-	-	-	-	4x336	-
	TTI,ms		40	40	40	40	40	20	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16	16
	RM attribute		170	170	170	170	170	140	160

Transport Format Combination Set							
TFCI	DCH	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF0	TF0	TF2	TF1
6	TF0	TF0	TF0	TF0	TF0	TF3	TF0
7	TF0	TF0	TF0	TF0	TF0	TF3	TF1
8	TF0	TF0	TF0	TF0	TF0	TF4	TF0
9	TF0	TF0	TF0	TF0	TF0	TF4	TF1
10	TF0	TF0	TF0	TF0	TF1	TF0	TF0
11	TF0	TF0	TF0	TF0	TF1	TF0	TF1
12	TF0	TF0	TF0	TF0	TF1	TF1	TF0
13	TF0	TF0	TF0	TF0	TF1	TF1	TF1
14	TF0	TF0	TF0	TF0	TF1	TF2	TF0
15	TF0	TF0	TF0	TF0	TF1	TF2	TF1
16	TF0	TF0	TF0	TF0	TF1	TF3	TF0
17	TF0	TF0	TF0	TF0	TF1	TF3	TF1
18	TF0	TF0	TF0	TF0	TF1	TF4	TF0
19	TF0	TF0	TF0	TF0	TF1	TF4	TF1
20	TF0	TF0	TF0	TF1	TF1	TF0	TF0

21	TF0	TF0	TF0	TF1	TF1	TF0	TF1
22	TF0	TF0	TF0	TF1	TF1	TF1	TF0
23	TF0	TF0	TF0	TF1	TF1	TF1	TF1
24	TF0	TF0	TF0	TF1	TF1	TF2	TF0
25	TF0	TF0	TF0	TF1	TF1	TF2	TF1
26	TF0	TF0	TF0	TF1	TF1	TF3	TF0
27	TF0	TF0	TF0	TF1	TF1	TF3	TF1
28	TF0	TF0	TF0	TF1	TF1	TF4	TF0
29	TF0	TF0	TF0	TF1	TF1	TF4	TF1
30	TF0	TF0	TF1	TF1	TF1	TF0	TF0
31	TF0	TF0	TF1	TF1	TF1	TF0	TF1
32	TF0	TF0	TF1	TF1	TF1	TF1	TF0
33	TF0	TF0	TF1	TF1	TF1	TF1	TF1
34	TF0	TF0	TF1	TF1	TF1	TF2	TF0
35	TF0	TF0	TF1	TF1	TF1	TF2	TF1
36	TF0	TF0	TF1	TF1	TF1	TF3	TF0
37	TF0	TF0	TF1	TF1	TF1	TF3	TF1
38	TF0	TF0	TF1	TF1	TF1	TF4	TF0
39	TF0	TF0	TF1	TF1	TF1	TF4	TF1
40	TF0	TF1	TF1	TF1	TF1	TF0	TF0
41	TF0	TF1	TF1	TF1	TF1	TF0	TF1
42	TF0	TF1	TF1	TF1	TF1	TF1	TF0
43	TF0	TF1	TF1	TF1	TF1	TF1	TF1
44	TF0	TF1	TF1	TF1	TF1	TF2	TF0
45	TF0	TF1	TF1	TF1	TF1	TF2	TF1
46	TF0	TF1	TF1	TF1	TF1	TF3	TF0
47	TF0	TF1	TF1	TF1	TF1	TF3	TF1
48	TF0	TF1	TF1	TF1	TF1	TF4	TF0
49	TF0	TF1	TF1	TF1	TF1	TF4	TF1
50	TF1	TF1	TF1	TF1	TF1	TF0	TF0
51	TF1	TF1	TF1	TF1	TF1	TF0	TF1
52	TF1	TF1	TF1	TF1	TF1	TF1	TF0
53	TF1	TF1	TF1	TF1	TF1	TF1	TF1
54	TF1	TF1	TF1	TF1	TF1	TF2	TF0
55	TF1	TF1	TF1	TF1	TF1	TF2	TF1
56	TF1	TF1	TF1	TF1	TF1	TF3	TF0
57	TF1	TF1	TF1	TF1	TF1	TF3	TF1
58	TF1	TF1	TF1	TF1	TF1	TF4	TF0
59	TF1	TF1	TF1	TF1	TF1	TF4	TF1

B.3.30.2.3 Transmit procedure for each TFCI

TFCI	“DCH Ch No.?” in DCH_Data(1B-6B).dll					DCH_Data(Packet).dll			DCH_Data (DCCH).dll
	The 1st time	The 2nd time	The 3rd time	The 4th time	The 5th time	”DCH Ch No.?”	"TTI?"	"Packet:Num of Transport Block?"	"TTI?"
0	–	–	–	–	–	–	–	–	–
1	–	–	–	–	–	–	–	–	“40”

2	–	–	–	–	–	“5”	“40”	“1”	–
3	–	–	–	–	–	“5”	“40”	“1”	“40”
4	–	–	–	–	–	“5”	“40”	“2”	–
5	–	–	–	–	–	“5”	“40”	“2”	“40”
6	–	–	–	–	–	“5”	“40”	“3”	–
7	–	–	–	–	–	“5”	“40”	“3”	“40”
8	–	–	–	–	–	“5”	“40”	“4”	–
9	–	–	–	–	–	“5”	“40”	“4”	“40”
10	“4”	–	–	–	–	–	–	–	–
11	“4”	–	–	–	–	–	–	–	“40”
12	“4”	–	–	–	–	“5”	“40”	“1”	–
13	“4”	–	–	–	–	“5”	“40”	“1”	“40”
14	“4”	–	–	–	–	“5”	“40”	“2”	–
15	“4”	–	–	–	–	“5”	“40”	“2”	“40”
16	“4”	–	–	–	–	“5”	“40”	“3”	–
17	“4”	–	–	–	–	“5”	“40”	“3”	“40”
18	“4”	–	–	–	–	“5”	“40”	“4”	–
19	“4”	–	–	–	–	“5”	“40”	“4”	“40”
20	“4”	“3”	–	–	–	–	–	–	–
21	“4”	“3”	–	–	–	–	–	–	“40”
22	“4”	“3”	–	–	–	“5”	“40”	“1”	–
23	“4”	“3”	–	–	–	“5”	“40”	“1”	“40”
24	“4”	“3”	–	–	–	“5”	“40”	“2”	–
25	“4”	“3”	–	–	–	“5”	“40”	“2”	“40”
26	“4”	“3”	–	–	–	“5”	“40”	“3”	–
27	“4”	“3”	–	–	–	“5”	“40”	“3”	“40”
28	“4”	“3”	–	–	–	“5”	“40”	“4”	–
29	“4”	“3”	–	–	–	“5”	“40”	“4”	“40”
30	“4”	“3”	“2”	–	–	–	–	–	–
31	“4”	“3”	“2”	–	–	–	–	–	“40”
32	“4”	“3”	“2”	–	–	“5”	“40”	“1”	–
33	“4”	“3”	“2”	–	–	“5”	“40”	“1”	“40”
34	“4”	“3”	“2”	–	–	“5”	“40”	“2”	–
35	“4”	“3”	“2”	–	–	“5”	“40”	“2”	“40”
36	“4”	“3”	“2”	–	–	“5”	“40”	“3”	–
37	“4”	“3”	“2”	–	–	“5”	“40”	“3”	“40”

38	"4"	"3"	"2"	–	–	"5"	"40"	"4"	–
39	"4"	"3"	"2"	–	–	"5"	"40"	"4"	"40"
40	"4"	"3"	"2"	"1"	–	–	–	–	–
41	"4"	"3"	"2"	"1"	–	–	–	–	"40"
42	"4"	"3"	"2"	"1"	–	"5"	"40"	"1"	–
43	"4"	"3"	"2"	"1"	–	"5"	"40"	"1"	"40"
44	"4"	"3"	"2"	"1"	–	"5"	"40"	"2"	–
45	"4"	"3"	"2"	"1"	–	"5"	"40"	"2"	"40"
46	"4"	"3"	"2"	"1"	–	"5"	"40"	"3"	–
47	"4"	"3"	"2"	"1"	–	"5"	"40"	"3"	"40"
48	"4"	"3"	"2"	"1"	–	"5"	"40"	"4"	–
49	"4"	"3"	"2"	"1"	–	"5"	"40"	"4"	"40"
50	"4"	"3"	"2"	"1"	"0"	–	–	–	–
51	"4"	"3"	"2"	"1"	"0"	–	–	–	"40"
52	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"1"	–
53	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"1"	"40"
54	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"2"	–
55	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"2"	"40"
56	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"3"	–
57	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"3"	"40"
58	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"4"	–
59	"4"	"3"	"2"	"1"	"0"	"5"	"40"	"4"	"40"

B.3.31　ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet 128[kbps] + DCCH Setting

Start_4BxP128K.dll is used.

B.3.31.1　Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet [128kbps] + DCCH

B.3.31.1.1 Uplink setting

• DTCH(UDI)x4+DTCH(Packet)+DCCHx1

Slot Format	Be sure to set to 2(15ksps)
Symbol rate	Be sure to set to 960ksps
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
DTX position	N/A
Puncturing Limit	0.64

B.3.31.1.2 Receivable TFCS etc.

Transport Format Set								
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x336	0x148
		TF1,bits	4x640	4x640	4x640	4x640	1x336	1x148
		TF2,bits	-	-	-	-	2x336	-
		TF3,bits	-	-	-	-	4x336	-
		TF4,bits	-	-	-	-	8x336	-
	TTI,ms		40	40	40	40	20	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16
	RM attribute		170	170	170	170	140	160

Transport Format Combination Set						
TFCI	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF0	TF2	TF1
6	TF0	TF0	TF0	TF0	TF3	TF0
7	TF0	TF0	TF0	TF0	TF3	TF1
8	TF0	TF0	TF0	TF0	TF4	TF0
9	TF0	TF0	TF0	TF0	TF4	TF1
10	TF0	TF0	TF0	TF1	TF0	TF0
11	TF0	TF0	TF0	TF1	TF0	TF1
12	TF0	TF0	TF0	TF1	TF1	TF0
13	TF0	TF0	TF0	TF1	TF1	TF1
14	TF0	TF0	TF0	TF1	TF2	TF0

15	TF0	TF0	TF0	TF1	TF2	TF1
16	TF0	TF0	TF0	TF1	TF3	TF0
17	TF0	TF0	TF0	TF1	TF3	TF1
18	TF0	TF0	TF0	TF1	TF4	TF0
19	TF0	TF0	TF0	TF1	TF4	TF1
20	TF0	TF0	TF1	TF1	TF0	TF0
21	TF0	TF0	TF1	TF1	TF0	TF1
22	TF0	TF0	TF1	TF1	TF1	TF0
23	TF0	TF0	TF1	TF1	TF1	TF1
24	TF0	TF0	TF1	TF1	TF2	TF0
25	TF0	TF0	TF1	TF1	TF2	TF1
26	TF0	TF0	TF1	TF1	TF3	TF0
27	TF0	TF0	TF1	TF1	TF3	TF1
28	TF0	TF0	TF1	TF1	TF4	TF0
29	TF0	TF0	TF1	TF1	TF4	TF1
30	TF0	TF1	TF1	TF1	TF0	TF0
31	TF0	TF1	TF1	TF1	TF0	TF1
32	TF0	TF1	TF1	TF1	TF1	TF0
33	TF0	TF1	TF1	TF1	TF1	TF1
34	TF0	TF1	TF1	TF1	TF2	TF0
35	TF0	TF1	TF1	TF1	TF2	TF1
36	TF0	TF1	TF1	TF1	TF3	TF0
37	TF0	TF1	TF1	TF1	TF3	TF1
38	TF0	TF1	TF1	TF1	TF4	TF0
39	TF0	TF1	TF1	TF1	TF4	TF1
40	TF1	TF1	TF1	TF1	TF0	TF0
41	TF1	TF1	TF1	TF1	TF0	TF1
42	TF1	TF1	TF1	TF1	TF1	TF0
43	TF1	TF1	TF1	TF1	TF1	TF1
44	TF1	TF1	TF1	TF1	TF2	TF0
45	TF1	TF1	TF1	TF1	TF2	TF1
46	TF1	TF1	TF1	TF1	TF3	TF0
47	TF1	TF1	TF1	TF1	TF3	TF1
48	TF1	TF1	TF1	TF1	TF4	TF0
49	TF1	TF1	TF1	TF1	TF4	TF1

B.3.31.2 Receiving ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + ISDN UDI(64kbps) + Packet [128kbps] + DCCH

B.3.31.2.1 Downlink setting

• DTCH(UDI)x4+DTCH(Packet)+DCCHx1

Slot Format(Symbol Rate)	Be sure to set to 15(480ksps)
Scrambling Code	Value set on the GUI
Channelization Code	Value set on the GUI
Tx Power	Value set on the GUI
DTX position	Flexible

B.3.31.2.2 Transmittable TFCS etc.

Transport Format Set								
DPCH	TrCH type		DCH	DCH	DCH	DCH	DCH	DCH
	TFS	TF0,bits	0x640	0x640	0x640	0x640	0x336	0x148
		TF1,bits	4x640	4x640	4x640	4x640	1x336	1x148
		TF2,bits	-	-	-	-	2x336	-
		TF3,bits	-	-	-	-	4x336	-
		TF4,bits	-	-	-	-	8x336	-
	TTI,ms		40	40	40	40	20	40
	Coding type		Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Turbo Coding	Convolutional Coding1/3
	CRC, bit		16	16	16	16	16	16
	RM attribute		170	170	170	170	140	160

Transport Format Combination Set						
TFCI	DCH	DCH	DCH	DCH	DCH	DCH
0	TF0	TF0	TF0	TF0	TF0	TF0
1	TF0	TF0	TF0	TF0	TF0	TF1
2	TF0	TF0	TF0	TF0	TF1	TF0
3	TF0	TF0	TF0	TF0	TF1	TF1
4	TF0	TF0	TF0	TF0	TF2	TF0
5	TF0	TF0	TF0	TF0	TF2	TF1
6	TF0	TF0	TF0	TF0	TF3	TF0
7	TF0	TF0	TF0	TF0	TF3	TF1
8	TF0	TF0	TF0	TF0	TF4	TF0
9	TF0	TF0	TF0	TF0	TF4	TF1
10	TF0	TF0	TF0	TF1	TF0	TF0
11	TF0	TF0	TF0	TF1	TF0	TF1
12	TF0	TF0	TF0	TF1	TF1	TF0
13	TF0	TF0	TF0	TF1	TF1	TF1
14	TF0	TF0	TF0	TF1	TF2	TF0
15	TF0	TF0	TF0	TF1	TF2	TF1
16	TF0	TF0	TF0	TF1	TF3	TF0
17	TF0	TF0	TF0	TF1	TF3	TF1
18	TF0	TF0	TF0	TF1	TF4	TF0
19	TF0	TF0	TF0	TF1	TF4	TF1
20	TF0	TF0	TF1	TF1	TF0	TF0
21	TF0	TF0	TF1	TF1	TF0	TF1
22	TF0	TF0	TF1	TF1	TF1	TF0

23	TF0	TF0	TF1	TF1	TF1	TF1
24	TF0	TF0	TF1	TF1	TF2	TF0
25	TF0	TF0	TF1	TF1	TF2	TF1
26	TF0	TF0	TF1	TF1	TF3	TF0
27	TF0	TF0	TF1	TF1	TF3	TF1
28	TF0	TF0	TF1	TF1	TF4	TF0
29	TF0	TF0	TF1	TF1	TF4	TF1
30	TF0	TF1	TF1	TF1	TF0	TF0
31	TF0	TF1	TF1	TF1	TF0	TF1
32	TF0	TF1	TF1	TF1	TF1	TF0
33	TF0	TF1	TF1	TF1	TF1	TF1
34	TF0	TF1	TF1	TF1	TF2	TF0
35	TF0	TF1	TF1	TF1	TF2	TF1
36	TF0	TF1	TF1	TF1	TF3	TF0
37	TF0	TF1	TF1	TF1	TF3	TF1
38	TF0	TF1	TF1	TF1	TF4	TF0
39	TF0	TF1	TF1	TF1	TF4	TF1
40	TF1	TF1	TF1	TF1	TF0	TF0
41	TF1	TF1	TF1	TF1	TF0	TF1
42	TF1	TF1	TF1	TF1	TF1	TF0
43	TF1	TF1	TF1	TF1	TF1	TF1
44	TF1	TF1	TF1	TF1	TF2	TF0
45	TF1	TF1	TF1	TF1	TF2	TF1
46	TF1	TF1	TF1	TF1	TF3	TF0
47	TF1	TF1	TF1	TF1	TF3	TF1
48	TF1	TF1	TF1	TF1	TF4	TF0
49	TF1	TF1	TF1	TF1	TF4	TF1

B.3.31.2.3 Transmit procedure for each TFCI

TFCI	“DCH Ch No.?” in DCH_Data(1B-6B).dll				DCH_Data(Packet).dll			DCH_Data (DCCH).dll
	The 1st time	The 2nd time	The 3rd time	The 4th time	"DCH Ch No.?"	“TTI?”	"Packet:Num of Transport Block?"	"TTI?"
0	–	–	–	–	–	–	–	–
1	–	–	–	–	–	–	–	“40”
2	–	–	–	–	“4”	“20”	“1”	–
3	–	–	–	–	“4”	“20”	“1”	“40”
4	–	–	–	–	“4”	“20”	“2”	–
5	–	–	–	–	“4”	“20”	“2”	“40”
6	–	–	–	–	“4”	“20”	“4”	–
7	–	–	–	–	“4”	“20”	“4”	“40”
8	–	–	–	–	“4”	“20”	“8”	–
9	–	–	–	–	“4”	“20”	“8”	“40”

10	“3”	–	–	–	–	–	–	–
11	“3”	–	–	–	–	–	–	“40”
12	“3”	–	–	–	“4”	“20”	“1”	–
13	“3”	–	–	–	“4”	“20”	“1”	“40”
14	“3”	–	–	–	“4”	“20”	“2”	–
15	“3”	–	–	–	“4”	“20”	“2”	“40”
16	“3”	–	–	–	“4”	“20”	“4”	–
17	“3”	–	–	–	“4”	“20”	“4”	“40”
18	“3”	–	–	–	“4”	“20”	“8”	–
19	“3”	–	–	–	“4”	“20”	“8”	“40”
20	“3”	“2”	–	–	–	–	–	–
21	“3”	“2”	–	–	–	–	–	“40”
22	“3”	“2”	–	–	“4”	“20”	“1”	–
23	“3”	“2”	–	–	“4”	“20”	“1”	“40”
24	“3”	“2”	–	–	“4”	“20”	“2”	–
25	“3”	“2”	–	–	“4”	“20”	“2”	“40”
26	“3”	“2”	–	–	“4”	“20”	“4”	–
27	“3”	“2”	–	–	“4”	“20”	“4”	“40”
28	“3”	“2”	–	–	“4”	“20”	“8”	–
29	“3”	“2”	–	–	“4”	“20”	“8”	“40”
30	“3”	“2”	“1”	–	–	–	–	–
31	“3”	“2”	“1”	–	–	–	–	“40”
32	“3”	“2”	“1”	–	“4”	“20”	“1”	–
33	“3”	“2”	“1”	–	“4”	“20”	“1”	“40”
34	“3”	“2”	“1”	–	“4”	“20”	“2”	–
35	“3”	“2”	“1”	–	“4”	“20”	“2”	“40”
36	“3”	“2”	“1”	–	“4”	“20”	“4”	–
37	“3”	“2”	“1”	–	“4”	“20”	“4”	“40”
38	“3”	“2”	“1”	–	“4”	“20”	“8”	–
39	“3”	“2”	“1”	–	“4”	“20”	“8”	“40”
40	“3”	“2”	“1”	“0”	–	–	–	–
41	“3”	“2”	“1”	“0”	–	–	–	“40”
42	“3”	“2”	“1”	“0”	“4”	“20”	“1”	–
43	“3”	“2”	“1”	“0”	“4”	“20”	“1”	“40”
44	“3”	“2”	“1”	“0”	“4”	“20”	“2”	–
45	“3”	“2”	“1”	“0”	“4”	“20”	“2”	“40”

46	“3”	“2”	“1”	“0”	“4”	“20”	“4”	–
47	“3”	“2”	“1”	“0”	“4”	“20”	“4”	“40”
48	“3”	“2”	“1”	“0”	“4”	“20”	“8”	–
49	“3”	“2”	“1”	“0”	“4”	“20”	“8”	“40”

B.4 HOW TO CHECK THE HARIKIRI (PHYSICAL LAYER) TEST UPLINK OPERATION

This section describes to know which level of process the demodulated DPCH and RACH signal comes in Signalling Tester when HARIKIRI uplink DPCH and DPCH HARIKIRI test is not worked.

B.4.1 DPCH Demodulation

Figure B-1 shows "Which level of process the demodulated DPCH signal comes." The following is the explanation of Figure B-1.Following is the detail description of Figure B-1.

1) **Checking that the RF power is entered into the receiving system of the Signalling Tester**
Check that the power equivalent to the power expected to be output from the mobile station is displayed on the Total Power in Monitor Screen(refer to section 2.6.1.)

2) **Checking that de-spreading is performed**
Check that Channel Power on the monitor screen is equivalent to the power expected to be output from the mobile station. If de-spreading can be performed in the correct timing, the timing error on the monitor screen indicates a value around zero.
If de-spreading is not performed, the spreading code of Signalling Tester and Mobile Station can be compared by monitoring RX Baseband connector(refer to 2.6.3).

3) **Checking that the pilot is OK.**
LED1(upper left) on RX Baseband board lights up when the pilot is OK for eight slots continuously (synchronous complementary state) and goes out when the pilot is NG for four slots continuously. (Judgement of pilot OK/NG is judged for each slot. The number of permissible errors in one slot is one bit.) The result of judgement for each slot is output as an electric signal from the RX Baseband connector(refer to section B.4.3).

4) **Checking that TFCI is OK**
Check that TFCI is OK by checking that the data is displayed on the Trace screen. TFI is displayed on Opt1 on the trace screen(refer to section 2.5).

5) **Checking that CRC is OK**
The result of CRC check is displayed at OPT2 on the trace screen(refer to section 2.5.2). If CRC is OK and expected data is not displayed,there is a method comparing the Coding Data using at Signalling Tester and Mobile Station. The Coding Data used by Signalling Tester is provided in the attached CDROM.Refer to “Readme.txt” inthe CDROM regarding the folder.

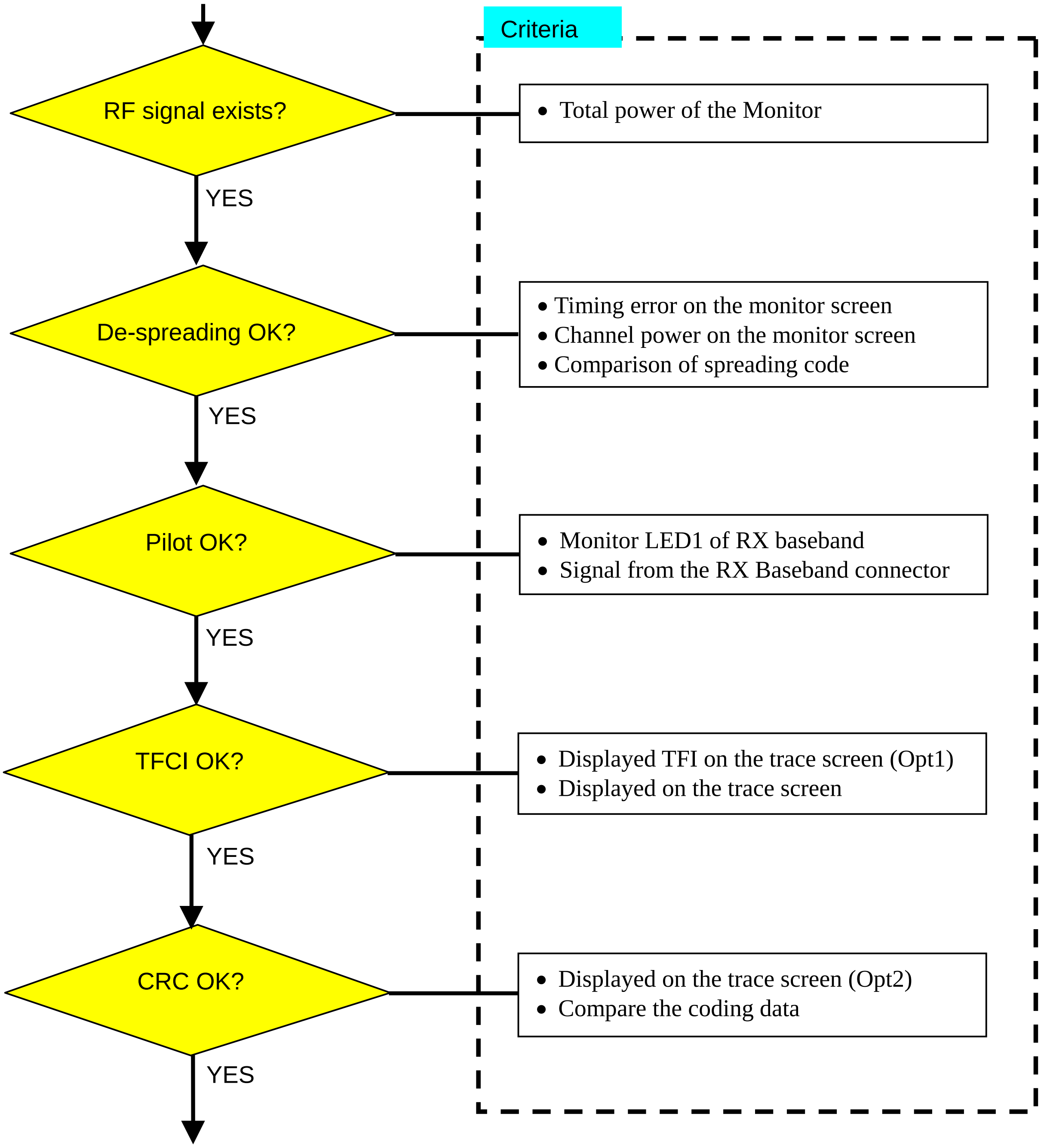


Figure B-1　Criteria of DPCH reception

B.4.2 RACH Demodulation

Figure B-2 shows " Which level of process the demodulated RACH signal comes." The following is the explanation of Figure B-2.

1) **Checking that RF power is entered into the receiving part of the Signalling Tester**
Check that the power equivalent to the power expected to be output from the mobile station is displayed on the Total Power in Monitor Screen(refer to section 2.6.1.).The waveform of the signal power output from the mobile station can be checked by connecting an oscilloscope to the back connector (Detector Output).

2) **Checking that the preamble is received successfully**
When a preamble is received, a signature number is output as an electric signal from the connector on the RX Baseband connector(see B.4.3). The signature number is output as serial data and a start bit is added to its head. Similarly, though an LED(RX Baseband1: Lower left) is lit, it may not be checked visually when a preamble is received singly, but you may not be able confirm the LED is lit ore not because single signature has so short time duration.
If the timing error on the monitor screen is around zero, it means that a preamble could be received at the correct timing.

3) **Checking that de-spreading is performed**
Check that Channel Power on the monitor screen is equivalent to the power expected to be output from the mobile station. If de-spreading can be performed in the correct timing, the timing error on the monitor screen indicates a value around zero. If de-spreading is not performed, the spreading code of Signalling Tester and Mobile Station can be compared by monitoring RX Baseband connector(refer to 2.6.3).

4) **Checking that the pilot of messages is OK.**
LED(RX Baseband2: Lower right) lights up when the pilot is OK for eight slots continuously (synchronous complementary state) and goes out when the pilot is NG for four slots continuously. (Judgement of pilot OK/NG is judged for each slot. The number of permissible errors in one slot is one bit.) The result of judgement for each slot is output as an electric signal from the RX Baseband1 connector(refer to section B.4.3).

5) **Checking that TFCI is OK**
Check that TFCI is OK by checking that the data is displayed on the Trace screen. TFI is displayed on Opt1 on the trace screen(refer to section 2.5).

6) **Checking that CRC is OK**
The result of CRC check is displayed at OPT2 on the trace screen(refer to section 2.5.2). If CRC is OK and expected data is not displayed,there is a method comparing the Coding Data using at Signalling Tester and Mobile Station. The Coding Data used by Signalling Tester is provided in the attached CDROM.Refer to “Readme.txt” inthe CDROM regarding the folder.

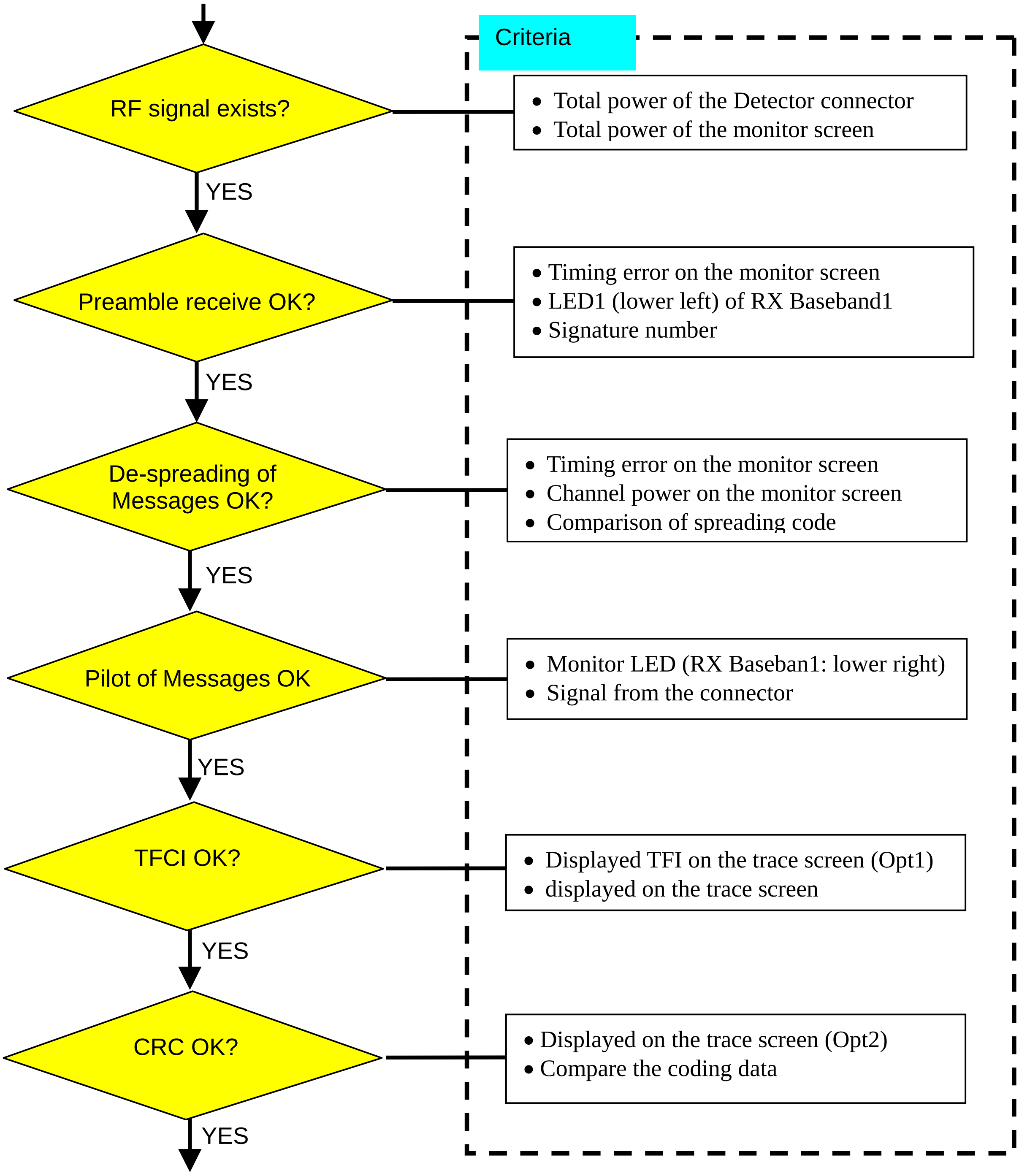


Figure B-2 Criteria of RACH reception

B.4.3 Signals from the RxBB Front Panel

Signals for monitoring the demodulation states of DPCH and RACH are output from the RX BASEBAND connector. Table B-3 shows the output signals. Data is output as serial data. The formats are shown in Figures B-3, B-4, and B-5. "L" level is always output when no signal is output. In addition, start and stop bits are added to recognize that data is output even if the transmit data is ALL0. Clock signals common to all the serial data are output to know the output interval of one bit of serial data.

Table B-3 Timing signals on the RX BASEBAND front panel (RX-Baseband1 connector)

Pin number	Description	Channel
25	DPCH pilot, "H" is output for OK and "L" is output for NG	DPCH
26	DPCH TFCI (serial data)	
27	DPCH TPC and FBI (serial data)	
28	Signature number is output when a preamble is received. (Serial data)	RACH
29	"H" is output when RACH pilot is OK while "L" is output when it is NG.	
30	RACH TFCI (serial data)	
31	Clock common to serial data	DPCH/ RACH

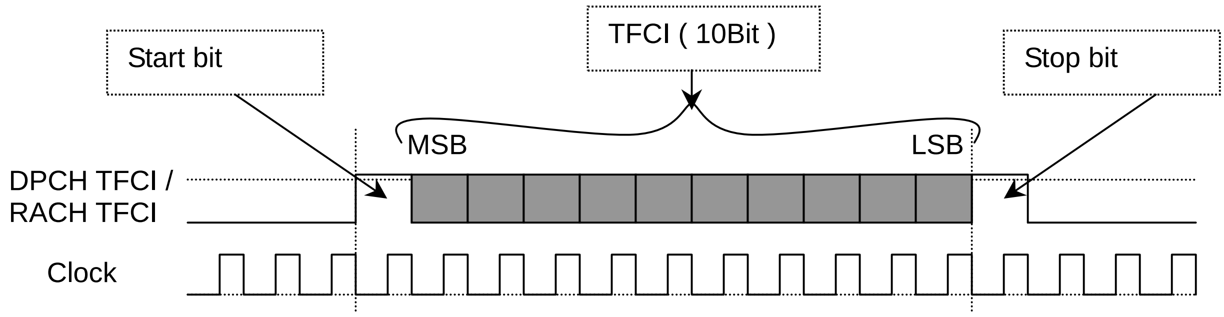


Figure B-3 Format of TFCI (10-bit) monitor signal

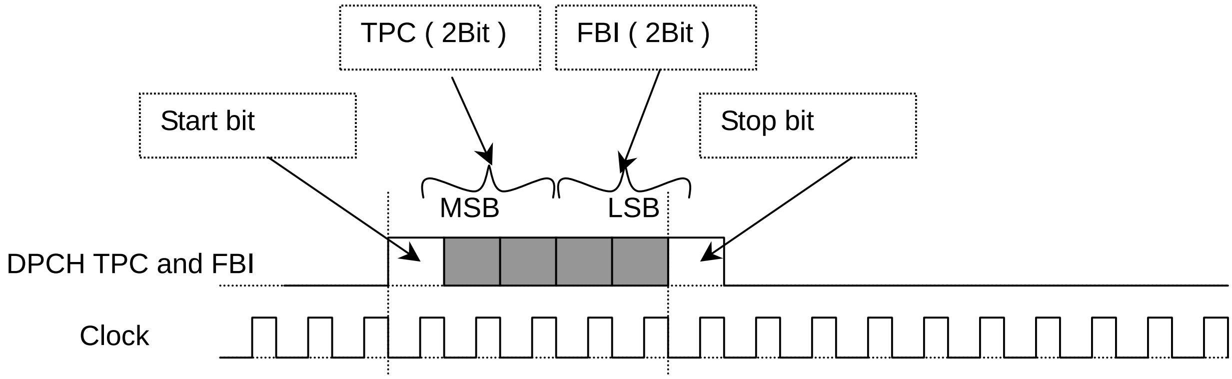


Figure B-4 Format of TPC/FBI monitor signal

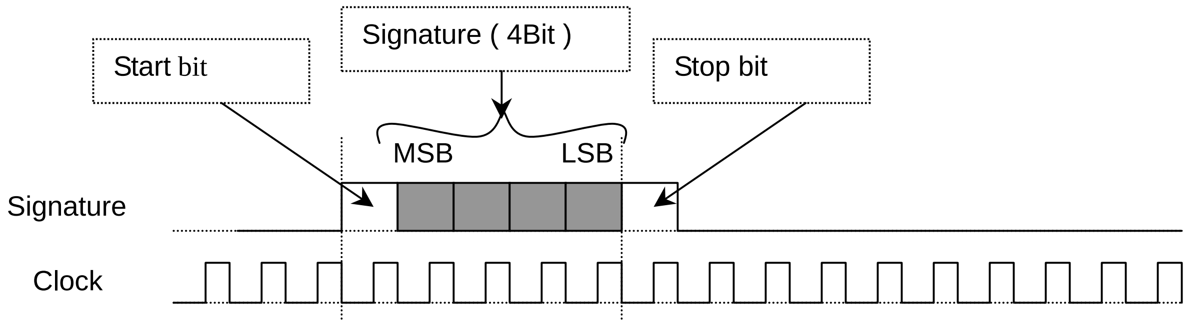


Figure B-5 Format of signature monitor signal

C. DETAILS OF TE CONNECTION TEST

Concept of TE port

Signalling Tester has the concept of "Port" in TE function. The port means the input/output destination of the U-Plane data. The types of the ports are shown in Table C-1 below. These types are specified with the library function CteConnect() in "Appendix A Scenario Libraries." (The "setting" in Table C-1 below refers to the setting used in CteConnect().)

Table C-1 Port List

Port type	Setting	Function	Remarks
NORMAL	TE_PORT_NORMAL	Provides ordinary service. (Connection to the TE side terminal) *1	Uplink and downlink, bi-directional
FIXDATA	TE_PORT_FIXDATA	Sends the fixed pattern.	Downlink only
LOOPBACK	TE_PORT_LOOPBACK	Uplink data loopback	The same value is set to uplink and downlink.
PN9	TE_PORT_PN9	Transmits the PN9 stage.	Downlink only
USERDATA	TE_PORT_USERDATA	Transmits and receives user-data.	Uplink and downlink, bi-directional
AUDIO	TE_PORT_AUDIO	Inputs and outputs the audio connector.	Uplink and downlink, bi-directional
MStoMS	TE_PORT_MSTOMS	MS to MS	Uplink and downlink, bi-directional
MStoMS2	TE_PORT_MSTOMS2	MS to MS (Transmits dummy data when there is no transmitted data.)	Uplink and downlink, bi-directional

*1 Ordinary service includes TV, TEL, handset, data terminal, etc.

The port configuration is shown in Figure C-1 below.

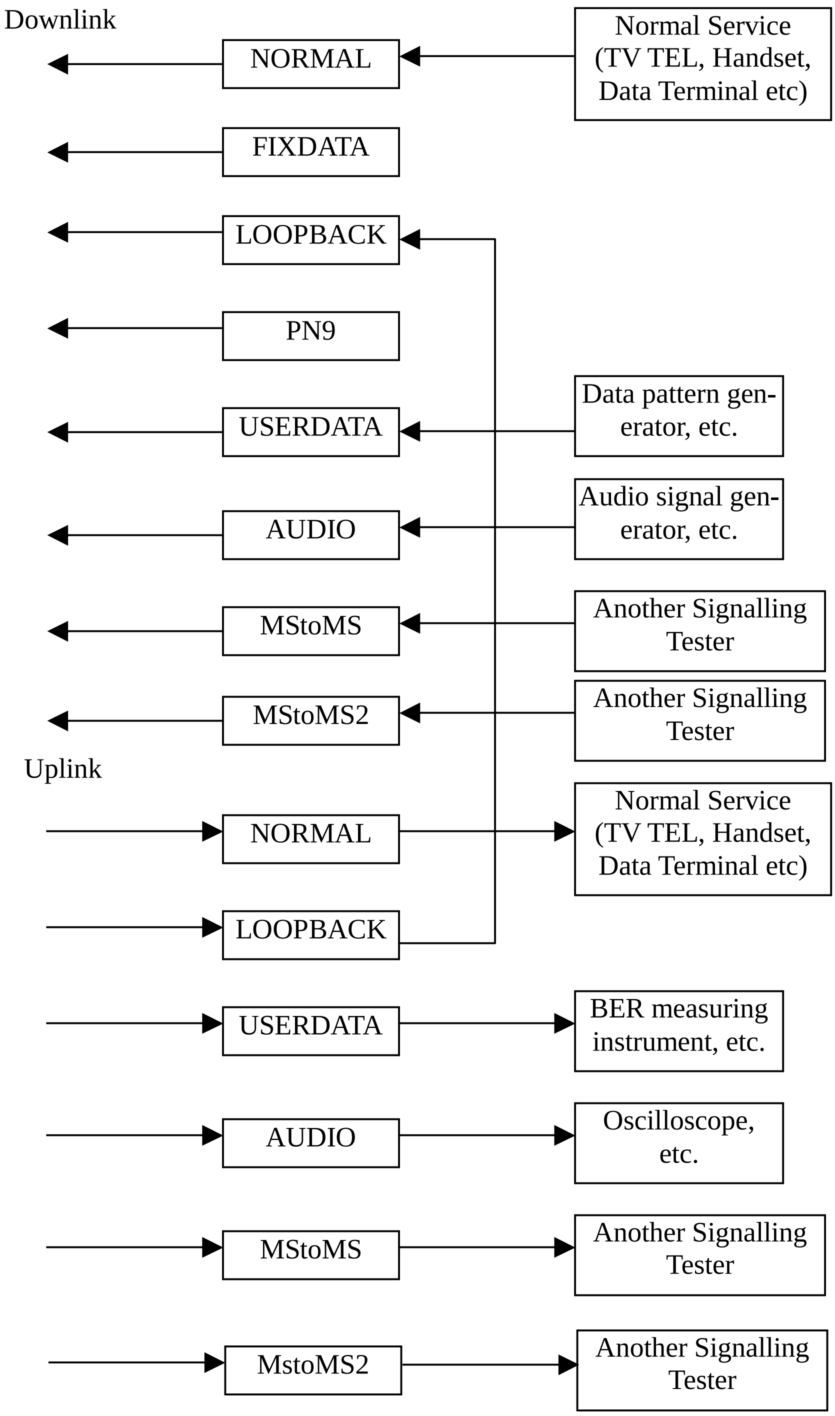


Figure C-1 Port configuration

Concept of Air side and TE side

Figure C-2 shows the conceptual diagram of Air side and TE side.

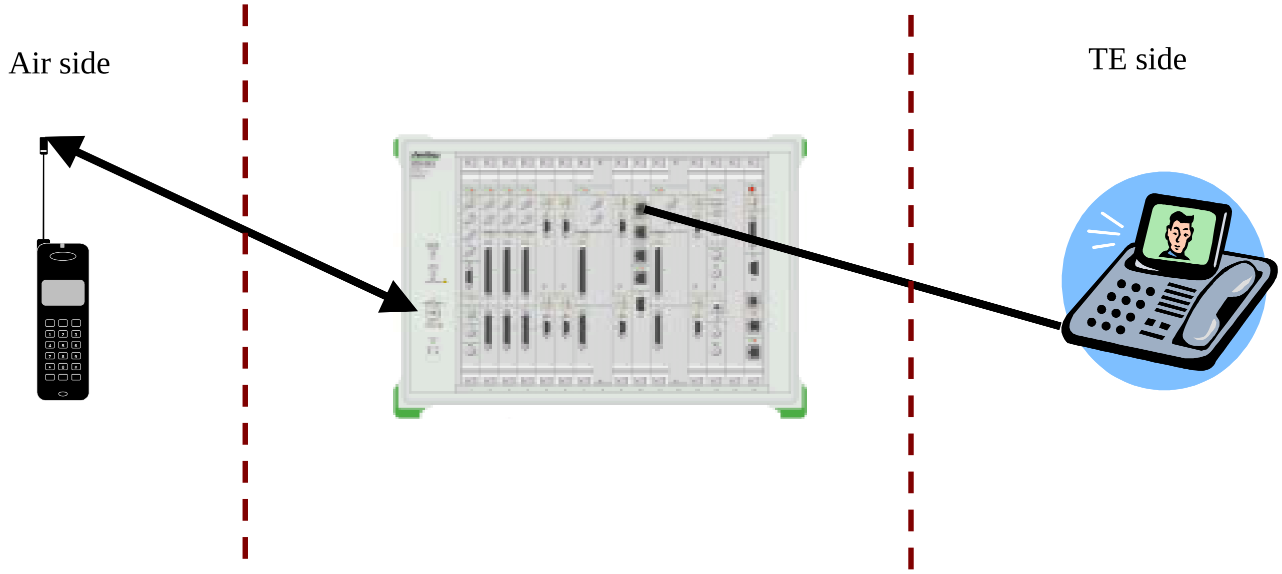


Figure C-2 Conceptual diagram of the ISDN test.

C.1 ISDN TEST

To conduct an ISDN test, the ISDN option is required.

C.1.1 Sample Scenarios

The sample ISDN scenarios are shown below.

The scenarios are located in C:¥Mx848000¥Scenario¥SRC¥TE Sample¥ISDN.

Table C-2 Scenario List

Scenario name	Description
AirOrig_Q931.c	Originates a call of 1B from Air to TE by handling Q.931 directly.
AirDisc_Q931.c	Disconnects a call of 1B from Air to TE by handling Q.931 directly.
TeOrig_Q931.c	Originates a call of 1B from TE to Air by handling Q.931 directly.
TeDisc_Q931.c	Disconnects a call of 1B from TE to Air by handling Q.931 directly.
AirOrig_UDI.c	Originates a call of 1B from Air to TE without handling Q.931 directly.
AirDisc_UDI.c	Disconnects a call of 1B from Air to TE without handling Q.931 directly.
TeOrig_UDI.c	Originates a call of 1B from TE to Air without handling Q.931 directly.
TeDisc_UDI.c	Disconnects a call of 1B from TE to Air without handling Q.931 directly.

Note that Signalling Tester sets the telephone number as shown below.

TE side
BRI1 connector:”1111111”
BRI2 connector:”2222222”
BRI3 connector:”3333333”

Air side :”4444444”

C.1.2 Scenario Configuration

AirOrig_Q931.c

This scenario executes the CteConfig( ) function to associate the TE type (Specify TE_TYPE_ ISDN Q931UDI because this scenario handles Q.931 directly) with the logical channel and the logical channel number (0 is given in this scenario as the logical channel number), and then execute the AirOrig_Q931( ) function.

Table C-3 AirOrig_Q931() argument list

Argument	Description
1st argument	Specifies the logical channel number (0 is given in this scenario)
2nd argument	Specifies the called party number ("11111111" is given in this scenario)

This function connects a call from Air to TE by exchanging Q.931 messages. The completion of Q.931 CONN ACK transmission starts the transmit/receive of B channel data and ends the scenario.

AirDisc_Q931.c

The AirDisc_Q931( ) function is executed.

Table C-4 AirDisc_Q931() argument list

Argument	Description
1st argument	Specifies the logical channel number (0 is given in this scenario)

This function disconnects a call from Air by exchanging Q.931 messages.
The completion of Q.931 REL COMP transmission ends the scenario.
Subsequently, execute the CteRelease( ) function to release the logical channel number. Note that the number 0 is given as the logical channel number that serves as an argument.

TeOrig_Q931.c

This scenario executes the CteConfig( ) function to associate the TE type (Specify TE_TYPE_ISDN_Q931UDI because this scenario handles Q.931 directly) with the logical channel and the logical channel number (0 is given in this scenario as the logical channel number), and then it executes the AirOrig_Q931( ) function.
The function TeOrig_Q931( ) has no argument.　It returns the logical channel number used for controlling a call from the TE, as a return value.　This function connects a call from the TE to Air by exchanging Q.931 messages.　The completion of Q.931 CONN ACK reception starts the transmit/receive of B channel data and ends the scenario.

TeDisc_Q931.c

TeDisc_Q931( ) is executed.　This function has no argument.　It returns the logical channel number used for controlling a call from the TE, as a return value.　This function disconnects a call from the TE by exchanging Q.931 messages.　The completion of Q.931 REL COMP reception ends the scenario.　Subsequently, it executes the CteRelease( ) function to release the logical channel number.　Note that the number 0 is given as the logical channel number that serves as an argument.

AirOrig_UDI.c

This scenario executes the CteConfig( ) function to associate the TE type (Specify TE_TYPE_ISDN_UDI because this scenario does not handle Q.931 directly) with the logical channel and the logical channel number (0 is given in this scenario as the logical channel number), and then it executes the CteConnect ( ) function.

Table C-5 CteConnect( ) argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Downlink port
4th argument	Specifies Uplink port
5th argument	Specifies Direction
6th argument	Specifies the option
7th argument	Specifies the timeout time

The CteConnect( ) provides the function for connecting a call with the TE side interface for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.　(Therefore, it is necessary to associate the TE type with the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.)　For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here.
For "Direction," specify either Air side call origination or TE side call origination. The parameters shown in Table C-6 can be specified.
In this scenario, CALL_FROM_AIR is specified.

Table C-6　Specified value for Direction at call origination

Type	Direction
Air side call origination	CALL_FROM_AIR
TE side call origination	CALL_FROM_TE

In this scenario, the called party number is specified as an option.　By specifying "1111111," the above function connects a call with the ISDN terminal of the BRI1 port.　The reception of CTE_CONN_CNF by RRC starts the transmit/receive of B channel data and ends the scenario.
By using this scenario, you can originates a call of ISDN without being conscious of Q.931.

AirDisc_UDI.c

This scenario executes the CteDisconnect( ) function to disconnect a call, and execute the CteRelease( ) function to reset the logical channel, the logical channel number, and the configuration parameters.

Table C-7 CteDisconnect( ) argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Direction
4th argument	Specifies the timeout time

The CteDisconnect( ) provides the function for disconnecting a call with the TE side interface for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.
For "Direction," specify either Air side call disconnection or TE side call disconnection.　The parameters shown in Table C-8 can be specified.

Table C-8　Specified value for Direction at call disconnection

Type	Direction
Air side call disconnection	CALL_FROM_AIR
TE side call disconnection	CALL_FROM_TE

By using this scenario, you can disconnect a call of ISDN without worrying about Q.931.

TeOrig_UDI.c

This scenario executes the CteConfig( ) function to associate the TE type (Specify TE_TYPE_ISDN_UDI because this scenario does not handle Q.931 directly) with the logical channel and the logical channel number (0 is given in this scenario as the logical channel number), and then it executes the CteConnect ( ) function.
The use of CteConnect( ) is different from AirOrig_UDI.c in the method of specifying the fifth argument "Direction" and the sixth argument "Option."　For "Direction" in this scenario, CALL_FROM_TE is specified.　Thereby, call origination processing from the TE is accepted.　"Option" is not used in this scenario.
The reception of CTE_CONN_RESP by the TE in the trace of the Control Software starts the transmit/receive of B channel data and ends the scenario.
By using this scenario, you can originates a call of ISDN without worrying about Q.931.

TeDisc_UDI.c

This scenario executes the CteDisconnect( ) function to disconnect a call, and execute the CteRelease( ) function to reset the logical channel, the logical channel number, and the configuration parameters.
The use of CteDisconnect( ) is different from AirDisc_UDI.c in the method of specifying the third argument "Direction." For "Direction" in this scenario, CALL_FROM_TE is specified. Thereby, disconnection processing from the TE is accepted.

By using this scenario, you can disconnect a call of ISDN without being conscious of Q.931.

C.1.3 Handling Calling Party Number

Normally, the calling party number length in the Q.931 message is one octet for BRI and two octets for PRI.
In these scenarios, however, the length "two octets" is used in either case.
For the control of calls, the method using a logical channel number is adopted.
Since the logical channel number is converted into a calling party number internally, recognize the logical channel number as a calling party number.

C.1.4　Executing Scenarios

Advance preparation

- Connect the ISDN terminal to the BRI1 connector of the ISDN board with an ISDN cable.
- The operation check of these scenarios has been made with NTT's Phoenix mini by Anritsu Corp.
- This section describes the procedure for conducting Harikiri (physical layer) tests with scenarios handling Q.931 directly as examples.

When conducting tests that do not handle Q.931, replace the former with the latter as follows in process of the test described below.

AirOrig_Q931.dll → AirOrig_UDI.c
AirDisc_Q931.dll → AirDisc_UDI.c
TeOrig_Q931.dll → TeOrig_UDI.c
TeDisc_Q931.dll → TeDisc_UDI.c

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_UDI.dll and click the START button.

(2) After the window for selecting the start channel appears, click the UDI button.

(3) After the window for selecting the scenario appears, execute AirOrig_Q931.dll or TeOrig_Q931.dll.

(4) [1] For AirOrig_Q931.dll, a ringing tone sounds as it stands.　Off hook the receiver.
[2] For TeOrig_Q931.dll, it is necessary to make a call from the ISDN terminal to the Air side.　For the time being, press "4444444" and Off hook the receiver.
#) The completion of Step [1] or [2] starts the transmit/receive of the B channel data and turns the system into the Harikiri (physical layer) state.

(5) [1] To disconnect from the Air side, execute AirDisc_Q931.d.　The connection is disconnected as it stands.
[2] To disconnect from the TE side, execute TeDisc_Q931.dll and on hook the receiver.

C.2 AMR VOICE TEST

C.2.1 Sample Scenarios

The AMR Voice sample scenarios are shown below.

Table C-9 Scenario list

Scenario name	Description
AirOrig_AMR.c	Originates a call of AMR Voice from Air to the Handset of the Signalling Tester.
AirDisc_AMR.c	Disconnects a call of AMR Voice from Air to the Handset of the Signalling Tester.
CteConn_Audio.c	Starts Input/Output of signals at the Audio connector of the Voice Codec.
CteDisc_Audio.c	Stops Input/Output of signals at the Audio connector of the Voice Codec.

C.2.2 Scenario Configuration

AirOrig_AMR.c

This scenario executes the CteConfig( ) function to associate the TE type with the logical channel and the logical channel number, and then it executes the CteConnect( ) function to start the transmit/receive of AMR Voice data for the handset.

In this scenario, the parameters to be set in CteConfig( ) are as shown in Table C-10.

Table C-10 Logical channel numbers and configuration parameters

Logical channel number	TE Type	Voice encoding rate	TBS*)
0	AMR Class A	12.2kbps	81bits
1	AMR Class B	12.2kbps	103bits
2	AMR Class C	12.2kbps	60bits

*) For the TBS value, set the same value as the transport size.

Table C-11 CteConnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Downlink port
4th argument	Specifies Uplink port
5th argument	Specifies Direction
6th argument	Specifies the option
7th argument	Specifies the timeout time

The CteConnect( ) provides the function for connecting a call with the TE side hand-set for the AMR Class corresponding to the logical channel and the logical channel number specified in the arguments. (Therefore, it is necessary to associate the AMR Class with the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.) For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here. For "Direction," specify either Air side call origination or TE side call origination. In this scenario only CALL_FROM_AIR can be specified.

Table C-12 Direction settings

Type	Direction
Air side call origination	CALL_FROM_AIR

CteConfig( ) and CteConnect( ) should be executed for Classes A, B, and C at each time of their necessity. That is, they need to be executed three times for each type.

AirDisc_AMR.c

This scenario executes the CteDisconnect( ) function to disconnect a call of AMR Voice, and execute the CteRelease( ) function to release the logical channel number and to reset the configuration parameters.
it executes the CteDisconnect( ) function to disconnect a call, and it executes the CteRelease( ) function to reset the logical channel, the logical channel number, and the configuration parameters.

Table C-13　CteDisconnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Direction
4th argument	Specifies the timeout time

The CteDisconnect( ) provides the function for disconnecting a call with the TE side handset for the AMR Class corresponding to the logical channel and the logical channel number specified in the arguments.
For "Direction," specify either Air side call disconnection or TE side call disconnection.　The parameters shown in Table C-14 can be specified.
only CALL_FROM_AIR can be specified.

Table C-14　Specified value for Direction

Type	Direction
Air side call origination	CALL_FROM_AIR

CteDisconnect( ) and CteRelease( ) should be executed for Classes A, B, and C at each time.　That is, they should be executed three times for each type.

CteConn_Audio.c

This scenario executes the CteConfig( ) function to associate the TE type with the logical channel and the logical channel number and to set the configuration parameters of AMR Voice, and then it executes the CteConnect( ) function to start the input and output of the signal at the Audio connector.　The parameters to be set in CteConfig( ) are the same as those for the scenario using the handset.

Table C-15　CteConnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Downlink port
4th argument	Specifies Uplink port
5th argument	Specifies Direction
6th argument	Specifies the option
7th argument	Specifies the timeout time

The CteConnect( ) provides the function for inputting and outputting the AMR class corresponding to the logical channel and the logical channel number specified in the arguments at the Audio connector.　(Therefore, it is necessary to associate the AMR Class with the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.)　For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here.　Though "Direction" is originally to specify either Air side call origination or TE side call origination, specify CALL_FROM_AIR in the case of the audio connector.

CteConfig( ) and CteConnect( ) should be executed for Classes A, B, and C at each time.　That is, they should be executed three times for each type.

CteDisc_Audio.c

This scenario executes the CteDisconnect( ) function to stop the input and output of signals at the audio connector, and then it executes the CteRelease( ) function to release the logical channel number and to reset the configuration parameters.

Table C-16　CteDisconnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Direction
4th argument	Specifies the timeout time

The CteDisconnect( ) provides the function for stopping the input and output of the AMR class corresponding to the logical channel and the logical channel number specified in the arguments at the Audio connector.　Though "Direction" is originally to specify either Air side call disconnection or TE side call disconnection, specify CALL_FROM_AIR in the case of the audio connector.

CteDisconnect( ) and CteRelease( ) should be executed for Classes A, B, and C at each time.　That is, they should be executed three times for each class.

Note) When you use Handset or Audio connector, you need to call CteConfig( ), CteConnect( ) for Class A,B,C(all classes). There are some voice rates that don't need class C. Please set 0 to TBS and NumOfTB on CTE_CONFIG_PAR structure for Class C at that rates, and call CteConfig( ) and CteConnect( ).

C.2.3 Executing Scenarios

<When using the handset>

Advance preparation

(1) Compile the sample scenario.

(2) Insert the modular jack of the handset into the Voice Codec.

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_AMR.dll and click the START button.

(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute AirOrig_AMR.dll.

#) The completion of Step (3) starts the transmit/receive of the AMR voice data and turns the system into the Harikiri (physical layer) state.

(4) Execute AirDisc_AMR.dll. The call is disconnected and the transmit/receive of the AMR Voice data is stopped.

<When using the audio connector>

Advance preparation

(1) Compile the sample scenario.

(2) Connect the measuring instrument for audio signals to the audio connector.

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_AMR.dll and click the START button.

(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute Cte-Conn_Audio.dll.

#) The completion of Step (3) starts the transmit/receive of the audio signal that is processed for the AMR Codec and turns the system into the Harikiri (physical layer) state.

(4) Execute CteDisc_Audio.dll. The input and output of audio signals are stopped.

C.3　USERDATA TEST

C.3.1　Sample Scenarios

The sample scenarios for userdata are shown below.

Table C-17　Scenario list

Scenario name	Description
CteConn_Userdata.c	Transmits the userdata fetched from the Userdata Input connector of the Voice Codec as downlink data, and outputs the uplink data from the Userdata Output connector.
CteDisc_Userdata.c	Stops the transmit/receive of the userdata.

C.3.2　Scenario Configuration

CteConn_Userdata.c

This scenario executes the CteConfig( ) function to set the logical channel, the logical channel number, and the configuration parameters, and then it executes the CteConnect( ) function to promotes specifying the port and starting the transmit/receive of userdata.　These scenarios specify the parameters to be set in CteConfig( ) as shown in Table C-18.　(UDI is taken as an example.)

Table C-18 Logical channel number and parameters

Logical channel number	TTI	NumOfTB	TBS*)
0	40ms	4	640 bits

*)　For the TBS value, set the same value as the transport size.

Table C-19 CteConnect( ) argument list

Argument	Description
1st argument	Logical channel
2nd argument	Logical channel number
3rd argument	Downlink port
4th argument	Uplink port
5th argument	Direction
6th argument	Option
7th argument	Timeout time

The CteConnect( ) provides the function for starting the transmit/receive of userdata for the data corresponding to the logical channel and the logical channel number specified in the arguments. (Therefore, it is necessary to set the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.) For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here. Though "Direction" is originally to specify either Air side call origination or TE side call origination, specify CALL_FROM_AIR in the case of userdata.

CteDisc_Userdata.c

This scenario executes the CteDisconnect( ) function to stop the transmit/receive of userdata, and then execute the CteRelease( ) function to release the logical channel and the logical channel number and to reset the configuration parameters.

For the CteDisconnect( ) function, specify the logical channel, logical channel number, Direction, and timeout time for the first, second, third, and fourth arguments, respectively. For the data corresponding to the logical channel and logical channel number specified in the arguments, the input and output at the userdata connector are stopped.

Though "Direction" is originally to specify either Air side call disconnection or TE side call disconnection, specify CALL_FROM_AIR in the case of userdata.

C.3.3 Executing Scenarios

Advance preparation

(1) Compile the sample scenario.

(2) Connect the external BER measuring instrument, etc. to the Userdata connector of the Voice Codec.

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_UDI.dll and click the START button.

(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute Cte-Conn_Userdata.dll.

#) The completion of Step (3) starts the transmit/receive of userdata and turns the system into the Harikiri (physical layer) state.

(4) Execute CteDisc_Userdata.dll. The transmit/receive of userdata is stopped.

C.4　IP PACKET TEST

C.4.1　Sample Scenarios

The sample scenarios for IP Packet are shown below.

Table C-20　Scenario list

Scenario name	Description
AirOrig_IPPacket.c	Connects a call of packet data in the 10/100Base-T interface of the Signalling Tester.
AirDisc_IPPacket.c	Disconnects a call of packet data in the 10/100Base-T interface of the Signalling Tester.

C.4.2　Scenario Configuration

AirOrig_IPPacket.c

Execute the CteConfig( ) function to associate the TE type with the logical channel number and to set the configuration parameters of the packet, and then execute the CteConnect( ) function to start the transmit/receive of packet data.

Table C-21　CteConnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Downlink port
4th argument	Specifies Uplink port
5th argument	Specifies Direction
6th argument	Specifies the option
7th argument	Specifies the timeout time

The CteConnect( ) provides the function for connecting a call with the TE side interface for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.　(Therefore, it is necessary to associate the TE type with the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.)　For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here.　For "Direction," specify either Air side call origination　or TE side call origination.　The parameters shown in Table C-22 can be specified.　Only CALL_FROM_AIR can be specified.
In this scenario, the option is not used.

Table C-22　Direction settings

Type	Direction
Air side call origination	CALL_FROM_AIR

For the TE type for which CteConfig( ) and CteConnect( ) are not specified, the transmit/receive of data is not performed.

AirDisc_IPPacket.c

This scenario executes the CteDisconnect( ) function to disconnect a call, and execute the CteRelease( ) function to reset the logical channel, the logical channel number, and the configuration parameters.

Table C-23 CteDisconnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Direction
4th argument	Specifies the timeout time

The CteDisconnect( ) provides the function for disconnecting a call with the TE side interface for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.
For "Direction," specify either Air side call disconnection or TE side call disconnection. The parameters shown in Table C-24 can be specified.
Only CALL_FROM_AIR can be specified.

Table C-24 Direction settings

Type	Direction
Air side call origination	CALL_FROM_AIR

C.4.3 Executing Scenarios

Advance preparation

(1) Compile the sample scenario.

(2) Connect the connector of the data terminal (PC) to the 10/100Base-T connector of the Signalling Tester with an Ethernet cross type cable.

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_P64K.dll or Start_P128K.dll and click the START button.

(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute AirOrig_IPPacket.dll.
#) The completion of Step (3) starts the transmit/receive of packet data and turns the system into the Harikiri (physical layer) state.

(4) Execute AirDisc_IPPacket.dll. The call is disconnected and the transmit/receive of packet data is stopped.

C.4.4 Setting IP address of Mobile Station side

Normal IP Packet test(Signalling Tester x 1,Server PC x 1)

When conducting IP Packet test, Signalling Tester return ARP Replay automatically. Therefore, use Ethernet cross cable one-to-one with the connection between Signalling Tester and Server PC. In this case , special setting is not required.

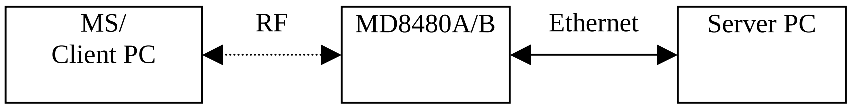


Figure C-3 Ethernet connection for normal IP Packet test

Multiple Signalling Testers and Server PCs test

Connections shown in Figure C-4 and C-5 cause a problem because of the automatic reply described above.

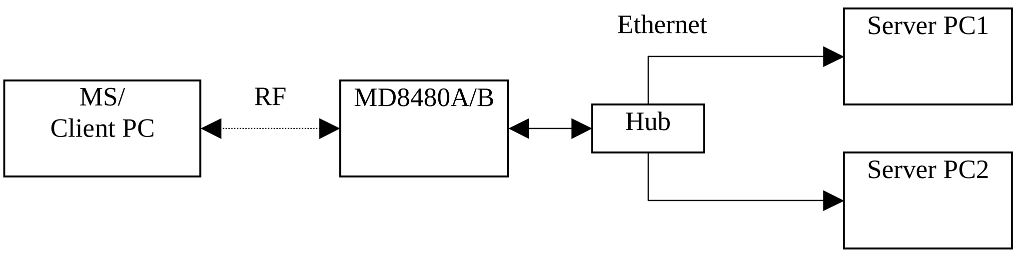


Figure C-4 Ethernet connection for multiple Servers

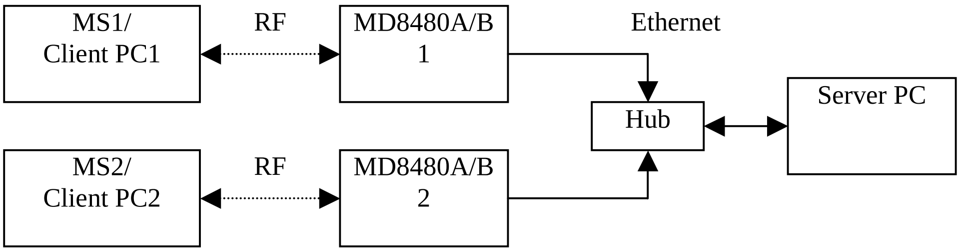


Figure C-5 Ethernet connection for multiple Signalling Testers

To avoid this problem, describe Mobile Station side IP address(Client PC) before SimulatorStart() in your scenario as follows.

Example)Client PC IP address = 1.1.0.2

CteStartPar.IppacketPar.MsIpv4Addr = 0x01010002;

Since Signalling Tester operates as TCP/IP stack of Client PC from the Server PC' s point of view, ARP Request which has the other destination address is not transmitted by specifying the Mobile Station side IP address.

(Note)

The above description had the default value = 0.In this case, the problem must be occurred.

C.4.5 ServerPC IP Address Setting Function

ApplicableTE Types

This function is available only for PPP(Built-in server) and IP Packet. Please note that it does not apply to PPP(serial connection.)

How to set

You can set up before the SimulatorStart( ) function of scenario. Below is an example to specify the address only for 1.1.0.1.And maximum IP address that can be set is 20.(The values 0 through 19 are active for Index.)

Ex.)

```
Index = 0;
CteStartPar.IppacketPar.BtsIpv4Addr[Index] = 0x01010001;
```

Matters to notice

1. Uplink Packet will stay in MD8480A/B inside for a certain period of time when a server PC that has a specified address is not connected.
2. The IP address which is set on a scenario is active for both IP Packet and PPP(built-in) server, and there is no need to specify PPP server address.(If you use PPP(built-in) server, this function would be the one to specify "IP address to communicate with other than PPP server address.)
3. If no setting in regard to this function are made, it will operate as usual.
4. This function works on PC:V1.41a and later, Firm:V1.41a and later.

C.5　PPP(Serial connection) TEST

C.5.1　Sample Scenarios

The sample scenarios for the PPP(serial connection) are shown below.

Table C-25　Scenario list

Scenario name	Description
AirOrig_PPP.c	Establishes a data communication channel in the serial interface of the Signalling Tester from the Air side.
AirDisc_PPP.c	Disconnects a data communication channel in the serial interface of the Signalling Tester from the Air side.
TeOrig_PPP.c	Establishes a data communication channel in the serial interface of the Signalling Tester from the TE side.
TeDisc_PPP.c	Disconnects a data communication channel in the serial interface of the Signalling Tester from the TE side.

* AirOrig_PPP.c　AirDisc_PPP.c operates only on Windows NT4.0.
* TeOrig_PPP.c　Operation of TeDisc_PPP.c has been checked on Windows NT4.0 and Windows2000.

C.5.2　Scenario Configuration

AirOrig_PPP.c

This scenario executes the CteConfig( ) function to associate the TE type with the logical channel and the logical channel number and to set the configuration parameters of the PPP, and then it executes the CteConnect( ) function to enable the transmit/receive of PPP data.
The settings in the serial interface of the Signalling Tester are shown in Table C-26.

For the Windows NT, make similar settings.

Table C-26　Serial Interface Setting

Parameter type	Function
Baud rate	115.2 kbps
Data bit	Data bit 8
Parity	No parity
Stop bit	Stop bit 1
Flow control	Hardware

Table C-27　CteConnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Downlink port
4th argument	Specifies Uplink port
5th argument	Specifies Direction
6th argument	Specifies the option
7th argument	Specifies the timeout time

The CteConnect( ) provides the function for establishing a communication channel with the TE side interface (Windows NT) for the TE type corresponding to the logical channel and the logical channel number specified in the arguments. (Therefore, it is necessary to associate the TE type with the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) before executing this function.) For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here. For "Direction," specify either Air side call origination or TE side call origination. The parameters shown in Table C-28 can be specified. In this scenario, CALL_FROM_AIR is specified.
In this scenario, the option is not used.

Table C-28　Direction settings

Type	Direction
Air side call origination	CALL_FROM_AIR
TE side call origination	CALL_FROM_TE

For the TE type for which CteConfig( ) and CteConnect( ) are not specified, no communication channel is established.

AirDisc_PPP.c

This scenario executes the CteDisconnect( ) function to disconnect a call, and then execute the CteRelease ( ) function to reset the logical channel, the logical channel number, and the configuration parameters.

Table C-29 CteDisconnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Direction
4th argument	Specifies the timeout time

The CteDisconnect( ) provides the function for disconnecting a communication channel with the TE side interface for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.
For "Direction," specify either Air side call disconnection or TE side call disconnection. The parameters shown in Table C-30 can be specified.
In this scenario, CALL_FROM_AIR is specified.

Table C-30 Specified value for Direction

Argument	Direction
Air side call origination	CALL_FROM_AIR
TE side call origination	CALL_FROM_TE

TeOrig_PPP.c

This scenario executes the CteConfig( ) function to associate the TE type with the logical channel and the logical channel number and to set the configuration parameters of the PPP, and then execute the CteConnect( ) function to enable the transmit/receive of PPP data.
The settings in the serial interface of the Signalling Tester and the Windows NT are the same as those of AirOrig_PPP.c. The specifications of the CteConnect( ) function are the same as those of AirOrig_PPP.c. In this scenario, CALL_FROM_TE indicating a call origination from the TE side is specified as the third argument "Direction." This specification provides the function for accepting the dialup connection from the Windows NT .
For the TE type for which CteConfig( ) and CteConnect( ) are not specified, no communication channel is established.

TeDisc_PPP.c

This scenario executes the CteDisconnect( ) function to disconnect a call, and then execute the CteRelease ( ) function to reset the logical channel, the logical channel number, and the configuration parameters.

The specifications of the CteDisconnect( ) function are the same as those of AirDisc_PPP.c. Note that in this scenario, CALL_FROM_TE indicating disconnection from the TE side is specified as the third argument "Direction." This specification provides the function for accepting disconnection from the Windows NT.

C.5.3 Executing Scenarios

Advance preparation

(1) Set the modem of the Windows NT to PC-PC connection.

(2) Connect the RS-232C connector of the Windows NT to the RS-232C connector of the ISDN board of the Signalling Tester with a serial straight cable.

(3) Compile the sample scenario.

Note 1: The serial cable supplied to the Signalling Tester as an accessory cannot be used for the PPP because it is a cross cable. We are sorry for giving you inconvenience, but would appreciate your kind understanding of this matter.

Note 2: A serial cable is used to connect the ISDN board and the Windows NT at this point, but it may be changed in the future.

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_P64K. and click the START button.

(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute AirOrig_PPP.dll or TeOrig_PPP.dll.

#) If AirOrig_PPP.dll is executed in Step (3), the communication channel between the Signalling Tester and the Windows NT has been established at this point. Establish the PPP session from the mobile station.

#) If TeOrig_PPP.dll is executed in Step (3), connection from the Windows NT is on standby at this point. Dial up from the Windows NT.

(4) Execute AirDisc_PPP.dll or TeDisc_PPP.dll.

#) If AirDisc_PPP.dll is executed in Step (4), the communication channel between the Signalling Tester and the Windows NT is disconnected at this point.

#) If TeDisc_PPP.dll is executed in Step (4), disconnection from the Windows NT is on standby at this point. Disconnect the dial-up connection from the Windows NT.

Notice to edit scenarios

If you edit sample scenarios, make sure that the setting of ISDNBOARD_PPP is defined before SimulatorStart() as follows:

```
CphyStartPar.IsdnPPP = ISDNBOARD_PPP;
SimulatorStart();
```

C.5.4 Restriction

To confirm the operation, we use windows NT as Server PC(Client PC) that is connected to the ISDN board through a serial cable. We have not confirmed the operation with windows 2000.

C.6　PPP (Built-in SERVER) TEST

For the PPP (built-in server) communication, the Signalling Tester is equipped with the PPP server function.　Therefore, the PPP of the TE on the mobile station is always a client and the negotiation of the PPP needs to be started from the TE on the mobile station.　Since Ethernet is used for connection, high-speed communication up to 384kbps can be conducted.

In addition, this function is supporting only the data communications of packet switching. It cannot use for a line switching.

* The operation on Windows2000 has not been checked.

C.6.1　Sample Scenarios

The sample scenarios are shown below.

Table C-31 Scenario list

Scenario name	Description
CteConn_PPPServer.c	Instructs the start of the transmit/receive of the PPP frame to the Signalling Tester.
CteDisc_PPPServer.c	Instructs the stop of the transmit/receive of the PPP frame to the Signalling Tester.

Support function and PPP option setting

This function supports the following protocols.

LCP
IPCP

Note:　NCPs other than IPCP are not supported.

The optional parameters specified in the PPP can be set by negotiation.　Table C-32 shows the parameters that can be negotiated.

Table C-32　PPP optional parameters

No.	Item	Parameter	Remarks
1	LCP option	MRU, PFC, ACFC, ACCM	The authentication protocol is PAP only.　Quality protocol is not available.
2	IPCP option	IP address, TCP/IP header compression	The IP addresses are shown below. PPP Server (Signalling Tester): 1.3.0.1 PPP Client (mobile station side): 1.3.0.2

C.6.2　Explanation of Scenarios

CteConn_PPPServer.c

This scenario executes the CteConfig( ) function to associate the TE type with the logical channel and the logical channel number and to set the configuration parameters of the channels.

Table C-33　CteConnect() argument list

Argument	Description
1st argument	Logical channel
2nd argument	Logical channel number
3rd argument	Downlink port
4th argument	Uplink port
5th argument	Direction
6th argument	Option
7th argument	Timeout time

The CteConnect( ) provides the function for starting the transmit/receive of frames for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.　(Therefore, it is necessary to associate the TE type with the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.)　For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_NORMAL here.　Though "Direction" is originally to specify either Air side call origination or TE side call origination, this selection does no make sense in this scenario.　Be sure to specify CALL_FROM_AIR uniquely.

For the TE type for which CteConfig( ) and CteConnect( ) are not specified, frames are not transmitted and received.

CteDisc_PPPServer.c

Table C-34　CteDisconnect() argument list

Argument	Description
1st argument	Logical channel
2nd argument	Logical channel number
3rd argument	Direction
4th argument	Timeout time

The CteDisconnect( ) provides the function for stopping the transmit/receive of frames for the TE type corresponding to the logical channel and the logical channel number specified in the arguments.
Though "Direction" is originally to specify either Air side call disconnection or TE side call disconnection, this selection does not make sense in this scenario.　Be sure to specify CALL_FROM_AIR uniquely.
This scenario execute the CteRelease ( ) function to reset the logical channel, the logical channel number, and the configuration parameters.

C.6.3 Executing Scenarios

Advance preparation

(1) Connect the 10/100BASE-T interface of the Signalling Tester to the PC server with an Ethernet cross type cable.

(2) Compile the sample scenario.

Procedure for conducting Harikiri (physical layer) tests

(1) Select Start_P384K.dll and click the START button.

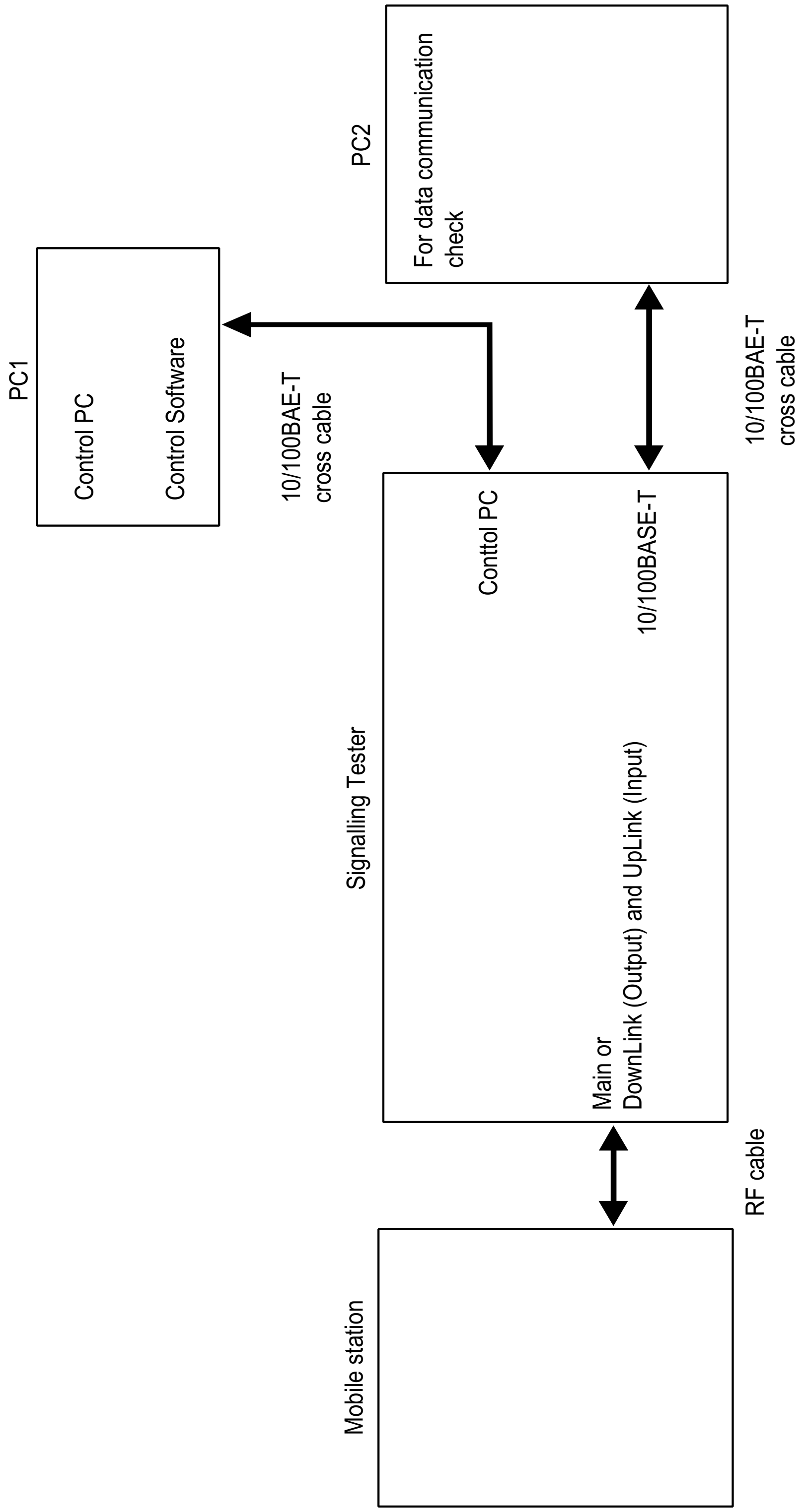
(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute CteConn_PPPServer.dll.
#) The completion of Step (3) enables the transmit/receive of PPP frames. Start the negotiation of the PPP from the mobile station side TE.

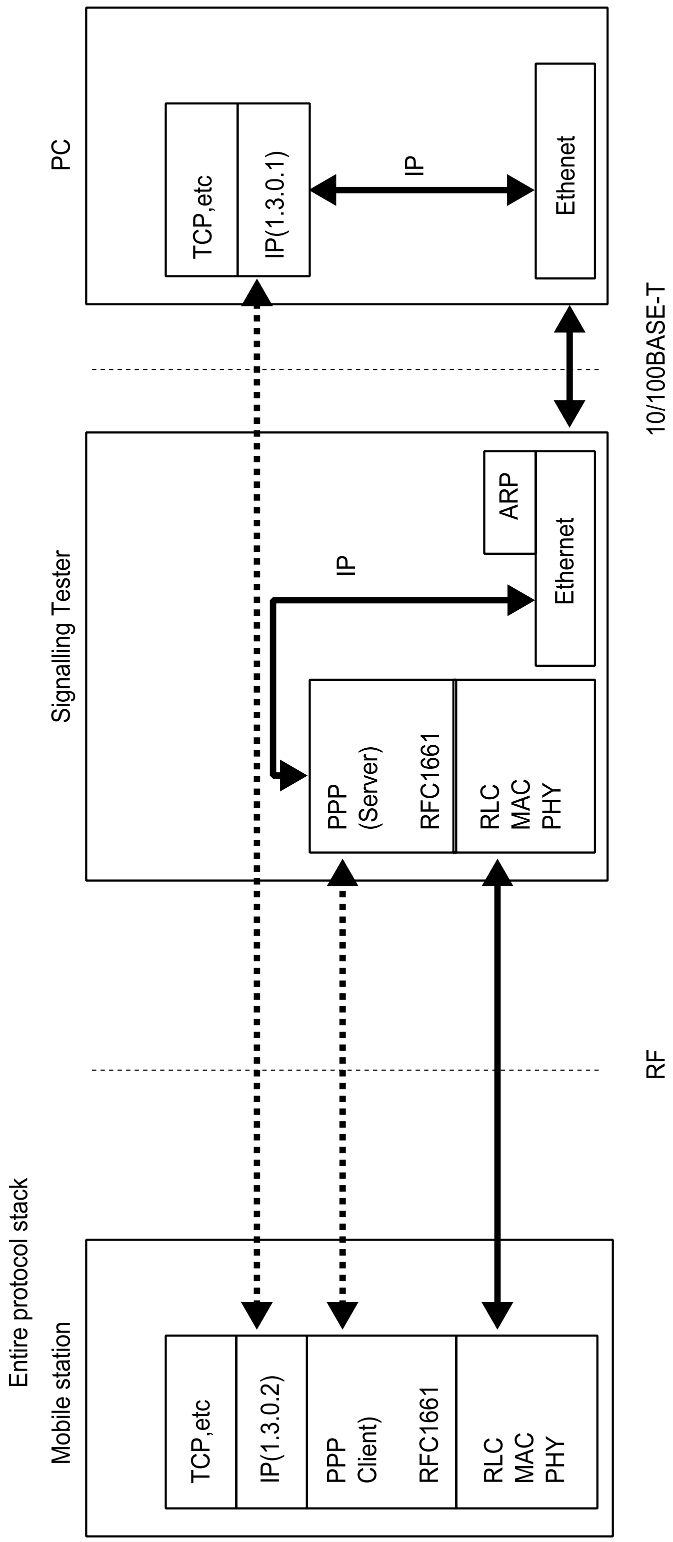
(4) Execute CteDisc_PPPServer.dll. The Signalling Tester stops the transmit/receive of PPP frames.

C.6.4 Connection

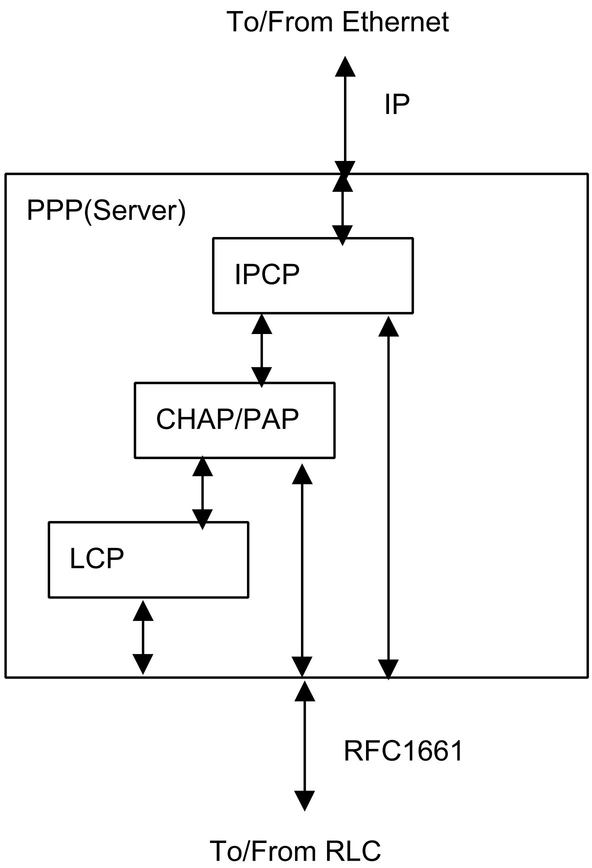
Connect in the same manner as the IP packet communication as shown below.



C.6.5　Protocol Stack



Protocol stack of the PPP (server) built in the Signalling Tester



The support and initial setting of each protocol is shown below.

LCP: Link Control Protocol

MRU:	Initial value: 1500oct (Determined by negotiation)
PFC, ACFC:	Initial value enable (Determined by negotiation)
ACCM:	Initial value 0x000A0000 (Determined by negotiation)
Quality protocol:	Not supported
Magic number:	Initial value enable

CHAP: Challenge Handshake Authentication Protocol
Not supported

PAP: Password Authentication Protocol
Supported

IPCP: Internet Protocol Control Protocol
Fixed to the following IP addresses.
PPP Server (Signalling Tester): 1.3.0.1
PPP Client (mobile station side): 1.3.0.2

Note: NCPs other than IPCP are not supported..
Note: MRU=MTU
For the PPP, MTU is not used but the term MRU indicating the Maximum Receive Unit is used instead.

C.6.6 PPP Option Setting

Option parameters for PPP(Built-in server) can be changed using following procedure.

Setting of Username and Password

Table C-35 shows the default value to be able to login PPP server of Signalling tester.

Table C-35 Default value of Username and Password

Parameter	Value
Username	PPP_CLIENT
Password	MD8480A

To change these value, describe as follows in your scenario.

Example)To change "Username = UE_CLIENT, Password = UE_PASSWD", describe as follows in the top of scenario.

```
#define PPP_MS_NAME         "UE_CLIENT"
#define PPP_MS_PASSWORD   "UE_PASSWD"
```

After described above, describe followings before SimulatorStart( )

```
memcpy(CteStartPar.PppSvPar.PapPar.MsName,
  PPP_MS_NAME, sizeof(PPP_MS_NAME));
memcpy(CteStartPar.PppSvPar.PapPar.MsPasswd,
  PPP_MS_PASSWORD, sizeof(PPP_MS_PASSWORD));
```

Note)

The max length of Username, Password is 21 characters(including NULL character)

Setting of IP address

Signalling Tester sets the default IP address set by IPCP negotiation as follows.

Signalling Tester: 1.3.0.1

Mobile Station: 1.3.0.1

To change these values, describe the 2 lines as follows in your scenario.

Example) IP address Signalling Tester=1.4.0.1 Mobile Station=1.4.0.2

```
CteStartPar.PppSvPar.IpcpPar.BtsIP   = 0x01040001;
CteStartPar.PppSvPar.IpcpPar.MsIP   = 0x01040002;
```

#)Describe in HEX.

Signalling tester using the new values for IPCP negotiation.

C.6.7 RFC1877 Setting Function

About RFC1877

In ordinary PPP negotiations, IP addresses are allocated to Mobile Station and RFC1877(IPCP Extensions for Name Server Addresses) is the standards to notify each server address of DNS and WINS at this time.

An applicable TE Type

This function is available only for PPP(built-in) server. Please note that it does not apply to serial connection PPP.

How to set

You can set up before the SimulatorStart( ) function on the scenario. The correspondence of the names of server and variable are as shown in TableC-36

TableC-36 the correspondence of the names of server and variables

Server names	Variable names to set
Primary DNS server	CteStartPar.PppSvPar.IpcpPar.PrmDnsSvAddr
PrimaryWINS server	CteStartPar.PppSvPar.IpcpPar.PrmWinsSvAddr
Secondary DNS server	CteStartPar.PppSvPar.IpcpPar.SecDnsSvAddr
Secondary WINS server	CteStartPar.PppSvPar.IpcpPar.SecWinsSvAddr

Below are examples to specify;
PrimaryDNS server: 1.3.0.3
Primary WINS server: 1.3.0.4
Secondary DNS server: 1.3.0.5
Secondary WINS server: 1.3.0.6

Ex.)
```
CteStartPar.PppSvPar.IpcpPar.Rfc1877FLAG      = PPPSV_RFC1877_ENA;
CteStartPar.PppSvPar.IpcpPar.PrmDnsSvAddr    = 0x01030003;
CteStartPar.PppSvPar.IpcpPar.PrmWinsSvAddr = 0x01030004;
CteStartPar.PppSvPar.IpcpPar.SecDnsSvAddr    = 0x01030005;
CteStartPar.PppSvPar.IpcpPar.SecWinsSvAddr = 0x01030006;
```

CteStartPar.PppSvPar.IpcpPar.Rfc1877FLAG is a flag to activate this function. Be certain to substitute PPPSV_RFC1877_ENA when using this function. When you do not use this function, substitute PPPSV_RFC1877_DIS or avoid describing the formula. In case that Mobile Station requires only certain address (for example, when only DNS server addresses of primary and secondary are set for IPCP Configure-Request frame), PPP(built-in) server will return just the required address. (Only DNS server address will be added to IPCP Configure-Ack/Nak frame that is returned from PPP(Built-in) server.) Please note that PPP(built-in) server will reject it even if the Mobile Station requires the allocation of address when the variable of the server address within the scenario is set to 0. (IPCP Configure-Reject will be returned.)

Matters to notice

1. Uplink Packet will stay in MD8480A/B inside for a certain period of time when a server PC that has a specified address is not connected.
2. There is no need to re-set a server address at "Server PC IP address setup function" that has been set using this function. Therefore, you need to set IP address of server PC except DNS and WINS server using "Server PC IP Address Set-

ting Function” when using this function and “Server PC IP Address Setting Function” together.
3. If no setting in regard to this function are made, it will operate as usual.
4. This function works on PC:V1.41a and later, Firm:V1.41d and later.

Restriction matters

This function does not completely conform to RFC1877 and is simplification due to being a unscheduled function at first

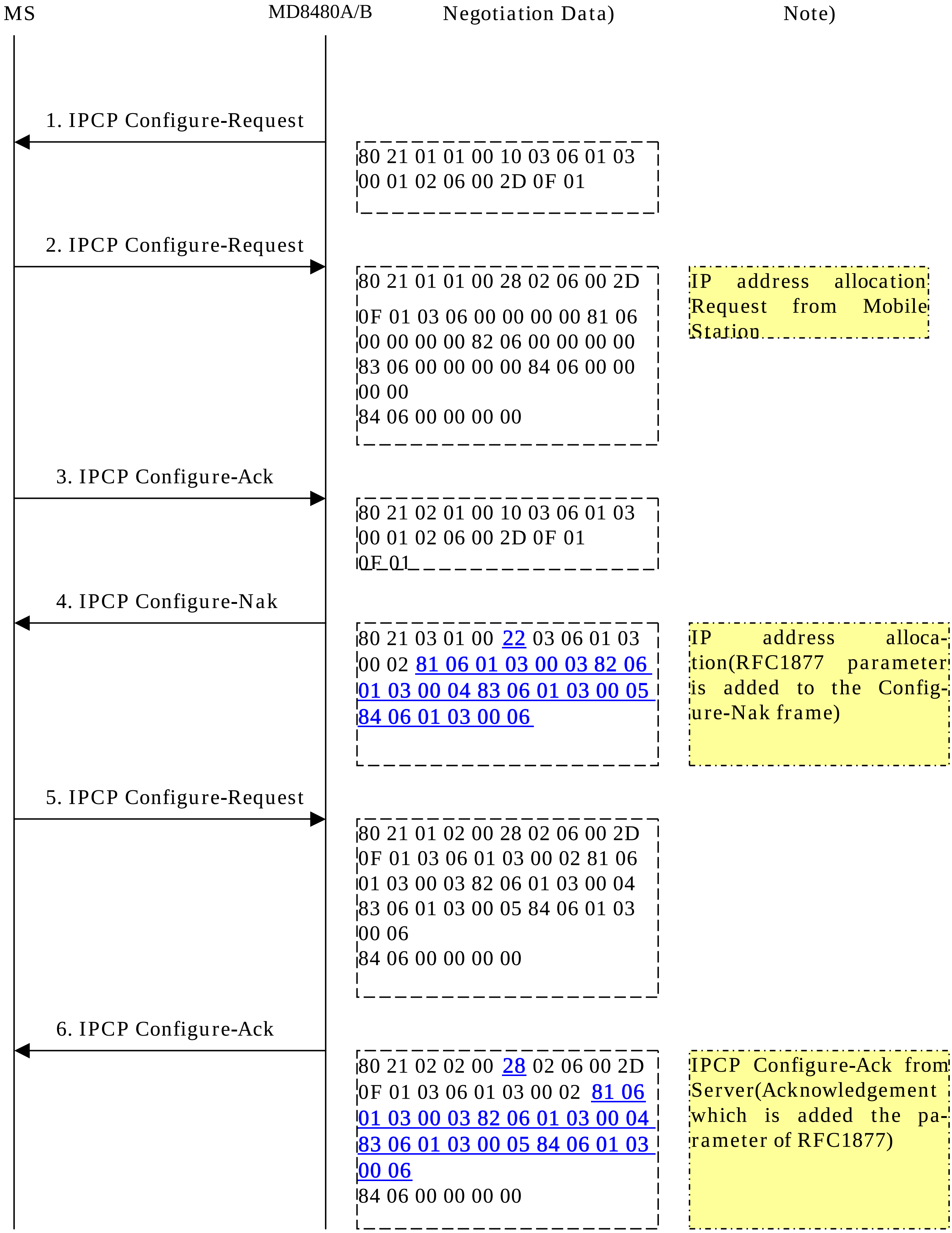
Generally PPP server performs an allocation at IPCP Configure-Nak (negative response) when PPP client requires the allocation of IP address. In this measure, we have adopted a method that parameter provided by RFC1877 is added separately to the behind of IPCP Configure-Nak frame that is sent from PPP(built-in) server.

FigureC-6 shows an actual example of IPCP negotiation, and the followings are cases to which this function corresponds.

- A case that PPP(built-in server) allocates all of IP addresses of mobile station and addresses of DNS/WINS among the options within IPCP Configure-Request frame sent from the mobile station. (a case that all of frames of 1. through 6. in the FigureC-6 exist.)
- A case that MD8480A/B accepts all of options within IPCP Configure-Request frame sent upward from the mobile station. (a case that frames of 2. and 4. in the FigureC-6 do not exist.)

Therefore, operations such as below are not warranted.

- A case that IP address of mobile station is pre-set and PPP(built-in) server allocates only DNS/WINS addresses.
- A case that PPP(built-in) server allocates IP address of mobile station and pre-set DNS/WINS address is used.



FigureC-6 IPCP negotiations supported on this function addition

Note1) letters in blue is data that is modified or added at this patch.

Note2) Negotiated value of address can be set voluntarily.

Operations in which frame except as listed above is transmitted and received are not warranted. (excluding IPCP Configure-Reject in which 0 is substituted for variable of server address.) However, operation in which frame of PPP CCP(RFC1962) is sent has no problem. (It will be discarded inside of MD8480A/B unconditionally.)

C.7 MS-TO-MS TEST

C.7.1 Sample Scenarios

The sample scenarios for the MS-to-MS are shown below.

Table C-37 Scenario list

Scenario name	Description
CteConn_MStoMS.c	Starts the function of MS-to-MS test.
CteDisc_MStoMS.c	Stops the function of MS-to-MS test.
CteConn_PCtoPC.c	Enables the data transfer between two control PCs (RRC).
SndPCtoPC.c	Performs the data transfer between two control PCs (RRC).
CteDisc_PCtoPC.c	Stops the data transfer between two control PCs (RRC).
CteConn_MStoMS2(AV32).c	Starts the function of MS-to-MS test. (TE_PORT_MSTOMS2, for AV bearer 32k service)
CteConn_MStoMS2(AV64).c	Starts the function of MS-to-MS test. (TE_PORT_MSTOMS2,for AV bearer 64k service)

C.7.2 Scenario Configuration

CteConn_MStoMS.c

This scenario executes the CteConfig( ) function to set the logical channel, the logical channel number, and the configuration parameters, and then it executes the CteConnect( ) function to enable the transmit/receive of data between two Signalling Testers. This scenario specifies the parameters to be set in CteConfig( ) as shown in Table C-38. (UDI data transmit/receive is taken as an example.)

Table C-38 Logical channel number and parameters

Logical channel number	TTI	NumOfTB	TBS*)
0	40ms	4	640 bits

*) For the TBS value, set the same value as the transport size.

Table C-39 CteConnect( ) argument list

Argument	Description
1st argument	Logical channel
2nd argument	Logical channel number
3rd argument	Downlink port
4th argument	Uplink port
5th argument	Direction
6th argument	Option
7th argument	Timeout time

The CteConnect( ) provides the function for starting the transmit/receive of data between two Signalling Testers for the data corresponding to the logical channel and the logical channel number specified in the arguments. (Therefore, it is necessary to set the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.) For the

Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_MSTOMS here.　Though "Direction" is originally to specify either Air side call origination or TE side call origination, specify CALL_FROM_AIR in this scenario.

The concept of MS-to-MS test is shown in Figure C-7 below.　Prepare mobile stations and Signalling Testers two units for each.　Let us provisionally call them MS_1, MS_2, MD8480A/B_1, and MD8480A/B_2　The execution of this scenario means enabling the "Data transfer between the Signalling Testers" shown in Figure C-7.　After the execution of the scenario, the uplink data in MS_1 enters MD8480A/B_2 through MD8480A/B_1 and reaches MS_2 after being mapped with the downlink signal in MD8480A/B_2.　The uplink data in MS_2 enters MD8480A/B_1 through MD8480A/B_2 and reaches MS_1 after being mapped with the downlink signal in MD8480A/B_1.

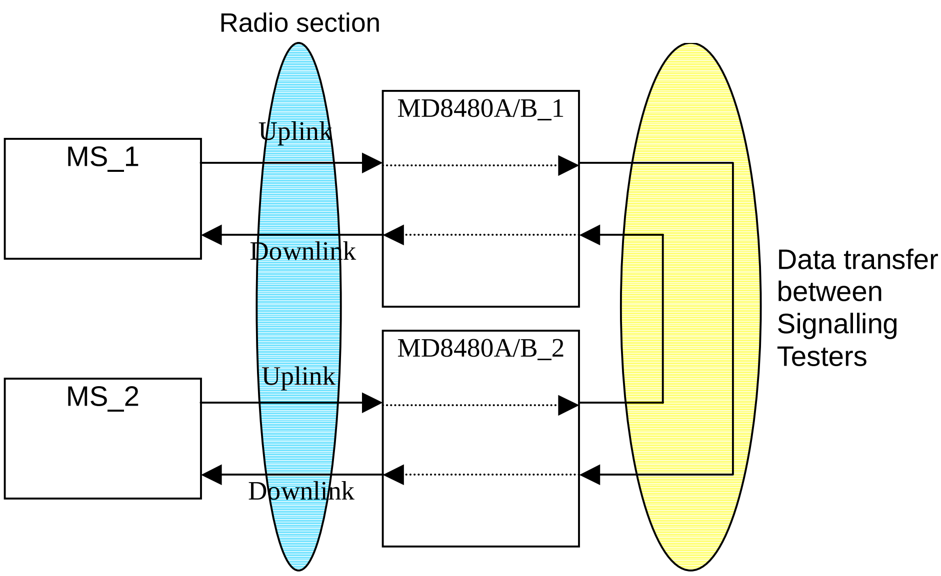


Figure C-7　Concept of MS-to-MS test

Note that this scenario needs to be executed for the logical channels and the logical channel numbers subject to the MS-to-MS test at each time of its necessity.

CteDisc_MStoMS.c

This scenario executes the CteDisconnect( ) function to stop the data transfer of the logical channel started for MS-to-MS test, and then it executes the CteRelease( ) function to release the logical channel and the logical channel number and to reset the configuration parameters.

Table C-40 CteDisconnect() argument list

Argument	Description
1st argument	Logical channel
2nd argument	Logical channel number
3rd argument	Direction
4th argument	Timeout time

CteDisconnect( ) stops the data transfer between the Signalling Testers for the data corresponding to the logical channel and the logical channel number specified in the arguments.
Though "Direction" is originally to specify either Air side call disconnection or TE side call disconnection, specify CALL_FROM_AIR in this scenario.

CteConn_PCtoPC.c

This scenario executes the CteConfig( ) function to set the logical channel, the logical channel number, and the configuration parameters, and then execute the CteConnect ( ) function to enable the transmit/receive of data between the two control PCs (RRC).
This scenario specifies the parameters to be set in CteConfig( ) as shown in Table C-41. (Transmit/receive of DCCH data is taken as an example in this scenario.)

Table C-41 Logical channel number and parameters

Logical channel number	TTI	NumOfTB	TBS*1)	Frame	Layer*2)
0	40ms	1	148 bits	TE_DATA_REQ	RRC

*1) For the TBS value, set the same value as the transport size.
*2) Layer indicates the one of the transmit destination.

Table C-42 CteConnect( ) argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Downlink port
4th argument	Specifies Uplink port
5th argument	Specifies Direction
6th argument	Specifies the option
7th argument	Specifies the timeout time

The CteConnect( ) provides the function for starting the transmit/receive of data between the Signalling Testers for the data corresponding to the logical channel and the logical channel number specified in the arguments. (Therefore, it is necessary to set the logical channel and the logical channel number, by executing the above-mentioned CteConfig( ) function before executing this function.) For the Downlink and Uplink ports that are to specify the input/output destinations of data, specify TE_PORT_MSTOMS here. Though "Direction" is originally to specify either Air side call origination or TE side call origination, specify CALL_FROM_AIR in this scenario.

SndPCtoPC.c

This scenario transfers data between two control PCs (RRC) for the channel for which CteConn_PctoPC.dll is executed.　Note that in the example of this scenario, DCCH is specified as the logical channel, 0 is specified as the logical channel number, and TE_DATA_REQ is specified as Frame.

CteDisc_PCtoPC.c

This scenario executes the CteDisconnect( ) function to stop the data transfer of the logical channel, and then it executes the CteRelease( ) function to release the logical channel and the logical channel number and to reset the configuration parameters.

Table C-43　CteDisconnect() argument list

Argument	Description
1st argument	Specifies the logical channel
2nd argument	Specifies the logical channel number
3rd argument	Specifies Direction
4th argument	Specifies the timeout time

CteDisconnect( ) stops the data transfer between two control PCs (RRC) for the data corresponding to the logical channel and the logical channel number specified in the arguments.
Though "Direction" is originally to specify either Air side call disconnection or TE side call disconnection, specify CALL_FROM_AIR in this scenario.

CteConn_MStoMS2(AV32).c

As well as CteConn_MStoMS.c, this scenario executes the CteConfig( ) function to set the logical channel, the logical channel number, and the configuration parameters, and then it executes the CteConnect( ) function to enable the transmit/receive of data between two Signalling Testers. This scenario specifies the parameters to be set in CteConfig( ) as shown in Table C-44. For CteConnect( ) argument list, refer to table C-39/42

Table C-44 Logical channel number and parameters

Logical channel number	TTI	NumOfTB	TBS*)
0	20ms	1	640 bits

*)For the TBS value, set the same value as the Transport Size.

Downlink and Uplink ports specify TE_PORT_MSTOMS2 here.
TE_PORT_MSTOMS2 is specified, then Signalling Tester transmits downlink dummy data(FF) when there is no transmitted data between Signalling Testers.

CteConn_MStoMS2(AV64).c

As well as CteConn_MStoMS2(AV32).c, this scenario executes the CteConfig( ) function to set the logical channel, the logical channel number, and the configuration parameters, and then it executes the CteConnect( ) function to enable the transmit/receive of data between two Signalling Testers. This scenario specifies the parameters to be set in CteConfig( ) as shown in Table C-45. For CteConnect( ) argument list, refer to table C-39/42.

Table C-45 Logical channel number and parameters

Logical channel number	TTI	NumOfTB	TBS*)
0	20ms	2	640 bits

*)For the TBS value, set the same value as the Transport Size.

Downlink and Uplink ports specify TE_PORT_MSTOMS2 here.
TE_PORT_MSTOMS2 is specified, then Signalling Tester transmits downlink dummy data(FF) when there is no transmitted data between Signalling Testers.
CteConn_MStoMS2(AV32).c and CteConn_MStoMS2(AV64).c are the scenarios for AV bearer, so the other of that executes tests using the scenario of CteConn_MStoMS.c

C.7.3　Executing Scenarios

Advance preparation

(1) Compile the sample scenario.

(2) Connect the 10/100Base-T connectors of the two Signalling Testers(called MD8480A/B_1 and MD8480A/B_2 here) with an Ethernet cross type cable. (Refer to Figure C-8.)

(3) Connect MD8480A/B_1(Sync Out connector) and MD8480A/B_2(Sync In connector) with BNC cable.(Refer to Figure C-9.)

(4) Connect MD8480A/B_1(10MHz Buffered Output connector) and MD8480A/B_2(10MHz Reference Input connector) with BNC cable. (Refer to Figure C-10.)

(5) Set to "Output" in Timing Sync in SETUP screen of MD8480A/B_1 GUI.

(6) Set to "Input" in Timing Sync in SETUP screen of MD8480A/B_2 GUI.

Note) Execute (2) to (6) before execute the sample scenario.

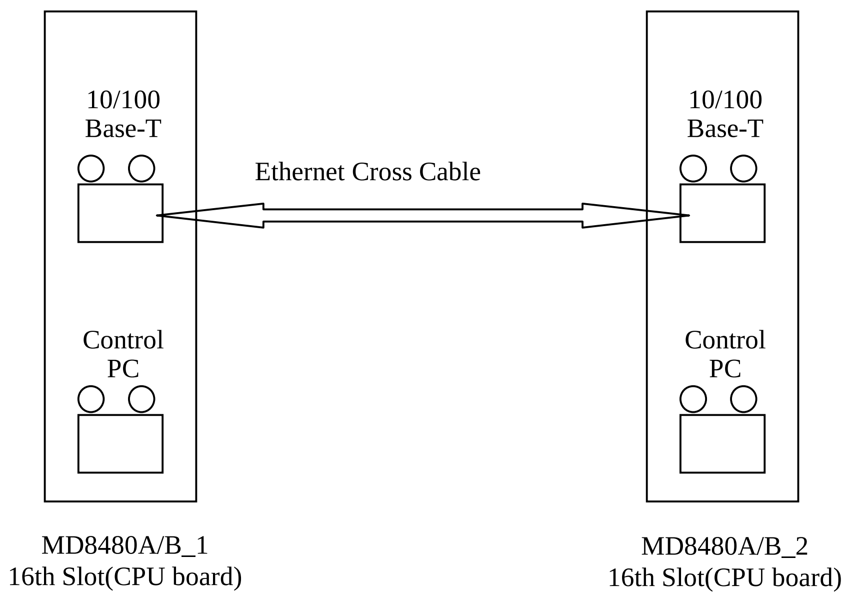


Figure C-8　Ethernet cable connection

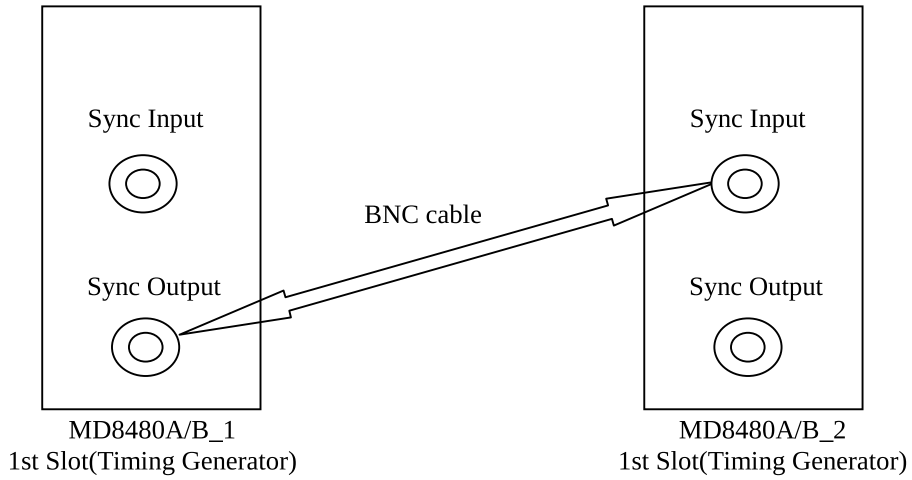


Figure C-9 BNC cable connection

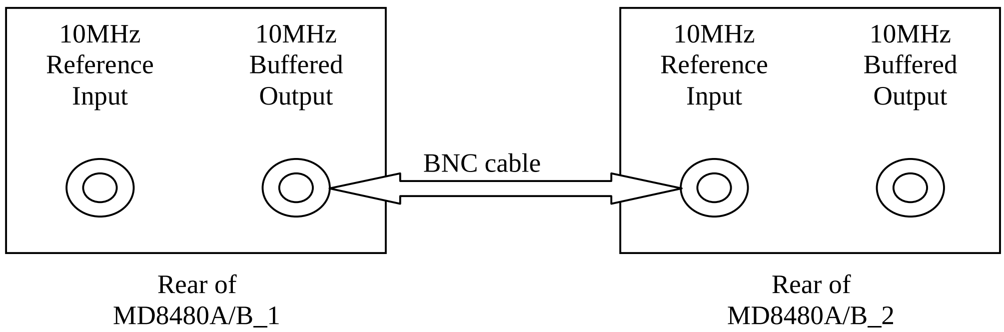


Figure C-10 Rear connection

Procedure for using the scenario for MS-to-MS

Execute the following procedure for both the two Signalling Testers.

(1) Select Start_UDI.dll and click the START button.

(2) After the window for selecting the activation appears, click the Active button.

(3) After the window for selecting the scenario appears, execute CteConn_MStoMS.dll, CteConn_MStoMS2(AV32).dll , CteConn_MStoMS2(AV64).dll or CteConn_PCtoPC.dll.

#) The completion of Step (3) starts the function of MS-to-MS test and enables MS-to-MS test through two Signalling Testers.

(4) Execute CteDisc_MStoMS.dll or CteDisc_PCtoPC.dll. The MS-to-MS function stops.

D. EXPLANATION OF SCENARIOS

This chapter explains the actual examples of a scenario by using a sample scenario.

D.1 Explanation of Sample Scenario for Harikiri (Physical Layer) Test

This section explains the sample scenarios with "Start_AMR.c" and "DCH_Data (AMR).c" taken as examples.

D.1.1 “ Start_AMR.c”

```
/***** INCLUDE *****/
#include      <windows.h>
#include      "wcdma.h"
#include      "primitive.h"
#include      "parameter.h"
#include      "scenario.h"
#include      "stdio.h"

/***** FUNCTION *****/
DLLEXPORT INT ScenarioMain(LPVOID);
```

(1) Be sure to describe it at the beginning of the scenario.

```
INT ReInitializeParameter();
```

(2) A function for changing the setting not to add the MAC Header for Harikiri (physical layer) tests. Therefore, when conducting a sequence test including upper layers, this function is not required. (The function is defined in Part (7) of this scenario.

```
CHAR *P_SCH_str;
CHAR *S_SCH_str;
CHAR *P_CCPCH_str;
CHAR *P_CPICH_str;
CHAR *S_CCPCH_str;
CHAR *PICH_str;
CHAR *RACH_str;
CHAR *AICH_str;
CHAR *DPCH_str;

INT DispChannelState();

INT ScenarioMain(LPVOID dmy)
{
  INT    BtnNo;

  P_SCH_str     = "OFF";
  S_SCH_str     = "OFF";
  P_CCPCH_str = "OFF";
  P_CPICH_str = "OFF";
  S_CCPCH_str = "OFF";
  PICH_str       = "OFF";
  RACH_str       = "OFF";
  AICH_str       = "OFF";
  DPCH_str       = "OFF";
  DispChannelState();
```

(3) Describe the functions to be executed in the Signalling Tester here.

(4) Describe it only to the scenario that is executed first.

```
  SimulatorStart(0,NO_TIMEOUT );
  ReInitializeParameter();
```

(5) See Part (2).

Starts the channel that transmits data when “P/S-SCH,P-CCPCH and P-CPICH” are activated. The initial values of the variables used in these functions have already been set in the software. The initial values are shown in C:¥MX848000¥InitializeParameter.c.

```
  /* P_SCH */
  CphyRlSetup(UNIT_BTS1,P_SCH ,0 ,&CphyRlSetupP_SCH    ,ACTIVATE_NOW,NO_TIMEOUT);

  /* S_SCH */
  CphyRlSetup(UNIT_BTS1,S_SCH ,0 ,&CphyRlSetupS_SCH    ,ACTIVATE_NOW,NO_TIMEOUT);

  /* P_CCPCH */
  CalcRMParameter(P_CCPCH,&CphyRlSetupP_CCPCH,&CphyTrchConfigP_CCPCH);
  CphyRlSetup     (UNIT_BTS1,P_CCPCH,0,&CphyRlSetupP_CCPCH,     ACTIVATE_NOW,NO_TIMEOUT);
  CphyTrchConfig(UNIT_BTS1,P_CCPCH,0,&CphyTrchConfigP_CCPCH,ACTIVATE_NOW,NO_TIMEOUT);
  CmacConfig       (UNIT_BTS1,P_CCPCH,0,&CmacConfigP_CCPCH,       ACTIVATE_NOW,NO_TIMEOUT);
  CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,BCCH,0,&CrlcConfigBCCH,TE,NO_TIMEOUT);

  /* P_CPICH Activete */
  CphyRlSetup(UNIT_BTS1,P_CPICH,0,&CphyRlSetupP_CPICH,ACTIVATE_NOW,NO_TIMEOUT);
```

```
 P_SCH_str     = "ON ";
 S_SCH_str     = "ON ";
 P_CCPCH_str = "ON ";
 P_CPICH_str = "ON ";
 DispChannelState();

 BtnNo=SequenceBtn("Voice Activate?","Active","No Active "," ",NO_TIMEOUT);
 if (BtnNo == 1) {
     /* DPCH Down Link */

     CalcRMParameter(D_DPCH,&CphyRlSetupD_DPCH,&CphyTrchConfigD_DPCH_AMR);
     CphyRlSetup     ( UNIT_BTS1, D_DPCH, 0, &CphyRlSetupD_DPCH ,           ACTIVATE_NOW,NO_TIMEOUT);
     CphyTrchConfig( UNIT_BTS1, D_DPCH, 0, &CphyTrchConfigD_DPCH_AMR, ACTIVATE_NOW,NO_TIMEOUT);
     CmacConfig      ( UNIT_BTS1, D_DPCH, 0, &CmacConfigD_DPCH_AMR,         ACTIVATE_NOW,NO_TIMEOUT);

     /* DPCH Up Link */
     CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH,&CphyTrchConfigU_DPCH_AMR);
     CphyRlSetup     ( UNIT_BTS1, U_DPCH, 0, &CphyRlSetupU_DPCH,            ACTIVATE_NOW,NO_TIMEOUT);
     CphyTrchConfig( UNIT_BTS1, U_DPCH, 0, &CphyTrchConfigU_DPCH_AMR, ACTIVATE_NOW,NO_TIMEOUT);
     CmacConfig      ( UNIT_BTS1, U_DPCH, 0, &CmacConfigU_DPCH_AMR,         ACTIVATE_NOW,NO_TIMEOUT);

     CrlcConfig( UNIT_BTS1, CRLC_TR_ESTABLISH, DTCH, 0, &CrlcConfigDTCH, TE , NO_TIMEOUT);
     CrlcConfig( UNIT_BTS1, CRLC_TR_ESTABLISH, DTCH, 1, &CrlcConfigDTCH, TE , NO_TIMEOUT);
     CrlcConfig( UNIT_BTS1, CRLC_TR_ESTABLISH, DTCH, 2, &CrlcConfigDTCH, TE , NO_TIMEOUT);
     CrlcConfig( UNIT_BTS1, CRLC_TR_ESTABLISH, DCCH, 0, &CrlcConfigDCCH, TE , NO_TIMEOUT);

     DPCH_str = "Voice ON ";
     DispChannelState();
 };

 return(0);
}
```

(6)

Starts the channel for AMRVoice. Like Part (5), the initial values of the variables used in these functions have already been set in the software. The initial values are shown in C:¥MX848000¥InitializeParameter.c.

```
INT DispChannelState()
 {
     CHAR str[100];

     sprintf(str,"P_SCH:%s S_SCH:%s P_CCPCH:%s P_CPICH:%s S_CCPCH:%s PICH:%s PRACH:%s AICH:%s DPCH:%s",
             P_SCH_str,S_SCH_str,P_CCPCH_str,P_CPICH_str,S_CCPCH_str,PICH_str,RACH_str,AICH_str,DPCH_str);

     SequenceDisp(str);
     return(0);
 };


/***********************************************
     ReInitializeParameter()
***********************************************/
INT ReInitializeParameter()
 {
     INT                  i,j,k;
     CMAC_CONFIG_PAR *CmacConfigPar;
     CMAC_CONFIG_PAR *CmacConfigParTable[]= {
                     &CmacConfigP_CCPCH,
                     &CmacConfigS_CCPCH_PCH,
                     &CmacConfigS_CCPCH_FACH,
                     &CmacConfigS_CCPCH_PCHxFACH,
                     &CmacConfigPRACH10M,
                     &CmacConfigPRACH20M,
                     &CmacConfigD_DPCH_SDCCH,
                     &CmacConfigU_DPCH_SDCCH,
                     &CmacConfigD_DPCH_AMR,
                     &CmacConfigU_DPCH_AMR,
                     &CmacConfigD_DPCH_UDI,
                     &CmacConfigU_DPCH_UDI,
                     &CmacConfigD_DPCH_AV64K,
                     &CmacConfigU_DPCH_AV64K,
                     &CmacConfigD_DPCH_AV32K,
                     &CmacConfigU_DPCH_AV32K,
                     &CmacConfigD_DPCH_P32K,
                     &CmacConfigU_DPCH_P32K,
                     &CmacConfigD_DPCH_P64K,
                     &CmacConfigU_DPCH_P64K,
                     &CmacConfigD_DPCH_P128K,
                     &CmacConfigU_DPCH_P128K,
                     &CmacConfigD_DPCH_P384K,
                     &CmacConfigU_DPCH_P384K,
                     &CmacConfigD_DPCH_AMRxP32K,
                     &CmacConfigU_DPCH_AMRxP32K,
                     &CmacConfigD_DPCH_AMRxP64K,
                     &CmacConfigU_DPCH_AMRxP64K,
```

(7)

See Part (2).

```
                        &CmacConfigD_DPCH_AMRxP128K,
                        &CmacConfigU_DPCH_AMRxP128K,
                        &CmacConfigD_DPCH_AMRxP384K,
                        &CmacConfigU_DPCH_AMRxP384K,
                        &CmacConfigD_DPCH_RMC12_2K,
                        &CmacConfigU_DPCH_RMC12_2K,
                        &CmacConfigD_DPCH_RMC64K,
                        &CmacConfigU_DPCH_RMC64K,
                        &CmacConfigD_DPCH_RMC144K,
                        &CmacConfigU_DPCH_RMC144K,
                        &CmacConfigD_DPCH_RMC384K,
                        &CmacConfigU_DPCH_RMC384K,
                        &CmacConfigD_DPCH_RMCBTFD,
                        &CmacConfigU_DPCH_RMCBTFD,
                        (CMAC_CONFIG_PAR *) 0
                };


    /**********************************/
    /******* Delete MAC Header *******/
    /**********************************/

    for(i=0;;i++) {
        CmacConfigPar = CmacConfigParTable[i];
        if (CmacConfigPar == (CMAC_CONFIG_PAR *)0) {
            break;
        };
        for(j=0;j<MAX_NO_TRANSPORT_CHANNELS;j++) {
            for(k=0;k<MAX_NO_LOGICAL_CHANNELS;k++) {
                CmacConfigPar->LochInfo[j][k].CTLength     = 0;
                CmacConfigPar->LochInfo[j][k].TCTFLength = 0;
                CmacConfigPar->LochInfo[j][k].UEIDType     = UEID_NOTUSE;
            };
        };
    };

    return(0);
};
```

D.1.2 " DCH_data(AMR).c"

```
/***** INCLUDE *****/
#include        <windows.h>
#include        "wcdma.h"
#include        "primitive.h"
#include        "parameter.h"
#include        "scenario.h"
#include        "stdio.h"

UCHAR DTCH_DATA1[] = {
          0x00,0x01,0x02,0x03,0x04,0x05,0x06,0x07,     /* TB #0 39bit or 81bit */
          0x08,0x09,0x0A
          };

UCHAR DTCH_DATA2[] = {                                  /* TB #0 103bit */
          0x00,0x01,0x02,0x03,0x04,0x05,0x06,0x07,
          0x08,0x09,0x0A,0x0B,0x0C
          };

UCHAR DTCH_DATA3[] = {                                  /* TB #0 60bit */
          0x00,0x01,0x02,0x03,0x04,0x05,0x06,0x07
          };

/***** FUNCTION *****/
DLLEXPORT INT ScenarioMain(LPVOID);

INT ScenarioMain(LPVOID dmy)
{
CTE_CONFIG_PAR     CteConfigDtch;

INT     BtnNo;
UCHAR Mode;

BtnNo=SequenceBtn("Sending Pattern?","FIX DATA","LOOP BACK","PN9",NO_TIMEOUT);
switch (BtnNo) {
     case 1:
          Mode = TE_PORT_FIXDATA;
          SequenceDisp("DTCH = Fix Data");
          break;
     case 2:
          Mode = TE_PORT_LOOPBACK;
          SequenceDisp("DTCH = Loop Back");
          break;
     case 3:
          Mode = TE_PORT_PN9;
          SequenceDisp("DTCH = PN9");
          break;
     default:
          Mode = TE_PORT_FIXDATA;
          break;
};
```

See Part (1).

(8)

These data are FIX-DATA for Class A,B and C respectively

See Part (3).

(9)

Selects user data to be transmitted as AMR Voice.
FIXDATA (set (8)), LOOPBACK of uplink data, or PN9 can be specified. When data from the handset, the user data connector, or AUDIO connector is to be used, use another scenario. For details, see "Appendix B. DETAILS OF HARIKIRI (PHYSICAL LAYER) TEST" or "Appendix C DETAILS OF TE CONNECTION."

```
BtnNo=SequenceBtn("AMR?","A","B and C","",NO_TIMEOUT);
switch (BtnNo) {
    case 1:
        BtnNo=SequenceBtn("TB Size?","39bit","81bit","",NO_TIMEOUT);
        switch (BtnNo) {
            case 1:
                CteConfigDtch.TeType    = TE_TYPE_NONE;
                CteConfigDtch.Rate      = 0;
                CteConfigDtch.TTI       = 2;
                CteConfigDtch.NumOfTB = 1;
                CteConfigDtch.TBS       = 39;
                CteConfigDtch.Frame     = RLC_TR_DATA_REQ;
                CteConfigDtch.Layer     = RLC;
                memcpy(CteConfigDtch.Data,DTCH_DATA1,sizeof(DTCH_DATA1));

                CteConfig (DTCH, 0, &CteConfigDtch, NO_TIMEOUT);
                CteConnect(DTCH, 0, Mode,TE_PORT_USERDATA,CALL_FROM_AIR,(CHAR *)0, NO_TIMEOUT);
                break;
            case 2:
                CteConfigDtch.TeType    = TE_TYPE_NONE;
                CteConfigDtch.Rate      = 0;
                CteConfigDtch.TTI       = 2;
                CteConfigDtch.NumOfTB = 1;
                CteConfigDtch.TBS       = 81;
                CteConfigDtch.Frame     = RLC_TR_DATA_REQ;
                CteConfigDtch.Layer     = RLC;
                memcpy(CteConfigDtch.Data,DTCH_DATA1,sizeof(DTCH_DATA1));

                CteConfig (DTCH, 0, &CteConfigDtch, NO_TIMEOUT);
                CteConnect(DTCH, 0, Mode,TE_PORT_USERDATA,CALL_FROM_AIR,(CHAR *)0, NO_TIMEOUT);
                break;
            default:
                break;
            };
        break;
    case 2:
        CteConfigDtch.TeType    = TE_TYPE_NONE;
        CteConfigDtch.Rate      = 0;
        CteConfigDtch.TTI       = 2;
        CteConfigDtch.NumOfTB = 1;
        CteConfigDtch.TBS       = 103;
        CteConfigDtch.Frame     = RLC_TR_DATA_REQ;
        CteConfigDtch.Layer     = RLC;
        memcpy(CteConfigDtch.Data,DTCH_DATA2,sizeof(DTCH_DATA2));

        CteConfig (DTCH, 1, &CteConfigDtch, NO_TIMEOUT);
        CteConnect(DTCH, 1, Mode,TE_PORT_USERDATA,CALL_FROM_AIR,(CHAR *)0, NO_TIMEOUT);

        CteConfigDtch.TeType    = TE_TYPE_NONE;
        CteConfigDtch.Rate      = 0;
        CteConfigDtch.TTI       = 2;
        CteConfigDtch.NumOfTB = 1;
        CteConfigDtch.TBS       = 60;
        CteConfigDtch.Frame     = RLC_TR_DATA_REQ;
        CteConfigDtch.Layer     = RLC;
        memcpy(CteConfigDtch.Data,DTCH_DATA3,sizeof(DTCH_DATA3));

        CteConfig (DTCH, 2, &CteConfigDtch, NO_TIMEOUT);
        CteConnect(DTCH, 2, Mode,TE_PORT_USERDATA,CALL_FROM_AIR,(CHAR *)0, NO_TIMEOUT);
        break;
};

return(0);
}
```

(10)

Maps each logical channel to RLC. TFC is determined by this mapping method. For details, see "Appendix B. DETAILS OF HARIKIRI (PHYSICAL LAYER) TEST."

D.2 Sample Scenario for Protocol Sequence

This section explains Idle.c and Orig_AMR.c as the scenario for protocol sequence.

D.2.1 " Idle.c"

```
/***** INCLUDE *****/
#include      <windows.h>
#include      "primitive.h"
#include      "parameter.h"
#include      "scenario.h"

/***** FUNCTION *****/
DLLEXPORT INT ScenarioMain(LPVOID);
```

See Part (1).

```
INT ScenarioMain(LPVOID dmy)
{
      SimulatorStart(0,NO_TIMEOUT );
```

See Part (3).

See Part (4).

See Part (5).

```
      /* P_SCH,S_SCH Activete */
      CphyRlSetup(UNIT_BTS1,P_SCH,   0,&CphyRlSetupP_SCH,   ACTIVATE_NOW,NO_TIMEOUT);
      CphyRlSetup(UNIT_BTS1,S_SCH,   0,&CphyRlSetupS_SCH,   ACTIVATE_NOW,NO_TIMEOUT);

      /* P_CPICH Activete */
      CphyRlSetup(UNIT_BTS1,P_CPICH,0,&CphyRlSetupP_CPICH,ACTIVATE_NOW,NO_TIMEOUT);

      /* P_CCPCH Activete, P_CCPCH-BCH-BCCH Setup*/
      CalcRMParameter(P_CCPCH,&CphyRlSetupP_CCPCH,&CphyTrchConfigP_CCPCH);
      CphyRlSetup     (UNIT_BTS1,P_CCPCH,0,&CphyRlSetupP_CCPCH,      ACTIVATE_NOW,NO_TIMEOUT);
      CphyTrchConfig(UNIT_BTS1,P_CCPCH,0,&CphyTrchConfigP_CCPCH,ACTIVATE_NOW,NO_TIMEOUT);
      CmacConfig      (UNIT_BTS1,P_CCPCH,0,&CmacConfigP_CCPCH,       ACTIVATE_NOW,NO_TIMEOUT);

      /* BCCH RLC Establish */
      CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,BCCH,0,&CrlcConfigBCCH,RRC,NO_TIMEOUT);

      /* S_CCPCH Activete, S_CCPCH-PCH-PCCH,S_CCPCH-FACH-CCCH Setup*/
      DRXCycleLength = 128;
      PageNp = 36;

      CalcRMParameter(S_CCPCH,&CphyRlSetupS_CCPCH,&CphyTrchConfigS_CCPCH_PCHxFACH);
      CphyRlSetup     (UNIT_BTS1,S_CCPCH,0,&CphyRlSetupS_CCPCH,                 ACTIVATE_NOW,NO_TIMEOUT);
   CphyTrchConfig(UNIT_BTS1,S_CCPCH,0,&CphyTrchConfigS_CCPCH_PCHxFACH,ACTIVATE_NOW,NO_TIMEOUT);
      CmacConfig    (UNIT_BTS1,S_CCPCH,0,&CmacConfigS_CCPCH_PCHxFACH,       ACTIVATE_NOW,NO_TIMEOUT);
```

```
      /* PRACH/AICH Activete,PRACH-RACH-CCCH Setup */
      CalcRMParameter(U_PRACH,&CphyRlSetupPRACH,&CphyTrchConfigPRACH20M);
      CphyRlSetup     (UNIT_BTS1,U_PRACH,0,&CphyRlSetupPRACH,           ACTIVATE_NOW,NO_TIMEOUT);
      CphyTrchConfig(UNIT_BTS1,U_PRACH,0,&CphyTrchConfigPRACH20M,ACTIVATE_NOW,NO_TIMEOUT);
      CmacConfig      (UNIT_BTS1,U_PRACH,0,&CmacConfigPRACH20M,       ACTIVATE_NOW,NO_TIMEOUT);
```

(11)

```
      /* CCCH RLC Establish */
      CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,PCCH,0,&CrlcConfigPCCH,RRC,NO_TIMEOUT);
      CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,CCCH,0,&CrlcConfigCCCH,RRC,NO_TIMEOUT);
      CrlcConfig(UNIT_BTS1,CRLC_UM_ESTABLISH,CCCH,1,&CrlcConfigCCCH,RRC,NO_TIMEOUT);
      CrlcConfig(UNIT_BTS1,CRLC_UM_ESTABLISH,CCCH,2,&CrlcConfigCCCH,RRC,NO_TIMEOUT);
```

(12) Configures PCCH or CCCH for RLC.

Starts the channel for Random Access.

```
    /* Sending System Indormations */
    { /* Master Information Block         Note:All SIB is padding */
        CHAR MIB[] = {
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00
        };

        SIB_POS = 0x00;
        SIB_REP = 0x10;
        SndMessage(UNIT_BTS1, RLC_TR_DATA_REQ, BCCH, 0, MIB, sizeof(MIB));
    };

    { /* System Information Block type1       Note:All SIB is padding */
        CHAR SIB1[] = {
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00
        };

        SIB_POS = 0x06;
        SIB_REP = 0x20;
        SndMessage(UNIT_BTS1, RLC_TR_DATA_REQ, BCCH, 0, SIB1, sizeof(SIB1));
    };

    { /* System Information Block type2       Note:All SIB is padding */
        CHAR SIB2_1[] = {
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00
        };

        SIB_POS = 0x12;
        SIB_REP = 0x20;
        SndMessage(UNIT_BTS1, RLC_TR_DATA_REQ, BCCH, 0, SIB2_1, sizeof(SIB2_1));
    };
    { /* SEG_COUNT = 2 */
        CHAR SIB2_2[] = {
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00
        };

        SIB_POS = 0x1A;    /* SIB_OFF(1) = 8 */
        SIB_REP = 0x20;
        SndMessage(UNIT_BTS1, RLC_TR_DATA_REQ, BCCH, 0, SIB2_2, sizeof(SIB2_2));
    };
   return(0);
}
```

(13) Transmits System Information. Unlike ordinary data transmit, set SIB_POS and SIB_REP before the SndMessage function.
Though data needs to be divided when there are two or more SEG_COUNT parameters, required division is not performed. (See System Information Block type2 in Idle.c.) Describe the divided data. Use SIB_OFF in the 3GPP standard for calculating SIB_POS. (The parameter SIB_OFF is not provided.)

D.2.2 " Orig_AMR.c"

```
/***** INCLUDE *****/
#include    <windows.h>
#include    "primitive.h"
#include    "parameter.h"
#include    "scenario.h"
#include    "stdio.h"

/***** FUNCTION *****/
DLLEXPORT    INT ScenarioMain(LPVOID);
```

See Part (1).

```
/***** Message Type *****/
/* Downlink DCCH messages */
#define DOWNLINK_DIRECT_TRANSFER          0x02
#define RADIO_BEARER_SETUP                0x0A
#define SECURITY_MODE_COMMAND             0x0E

/* Uplink DCCH messages */
#define INITIAL_DIRECT_TRANSFER           0x03
#define RADIO_BEARER_SETUP_COMPLETE       0x0C
#define RRC_CONNECTION_SETUP_COMPLETE     0x12
#define SECURITY_MODE_COMPLETE            0x14
#define UPLINK_DIRECT_TRANSFER            0x1A

/* Downlink CCCH messages */
#define RRC_CONNECTION_SETUP              0x02

/* Uplink CCCH messages */
#define RRC_CONNECTION_REQUEST            0x02
/***************************************************/
/* Origination arrow diagram in this Scenario       */
/*   MS                                   MD8480    */
/*   |----- RRC Connection Request          ---->|   */
/*   |          (TR-Mode PRACH-RACH-CCCH)        |   */
/*   |<---- RRC Connection Setup            -----|   */
/*   |          (UM-Mode S-CCPCH-FACH-CCCH)      |   */
/*   |----- RRC Connection Setup Complete   ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |----- MM CM Service Request           ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |<---- MM Authentic Request            -----|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |----- MM Authentic Response           ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |<---- Security Mode Command           -----|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |----- Security Mode Complete          ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |----- CC Setup                        ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |<---- CC Call Proceeding              -----|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |<---- Radio Bearer Setup              -----|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |----- Radio Bearer Setup Complete     ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |<---- CC Alerting                     -----|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |<---- CC Connect                      -----|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*   |----- CC Connect Acknowledge          ---->|   */
/*   |          (AM-Mode DPCH-DCH-DCCH)          |   */
/*                                               */
/***************************************************/
```

See Part (3).

```
INT ScenarioMain(LPVOID dmy)
{
    INT    Loch;    /* Logical Channel */
    Loch = DTCH;
```

```
/* Receive Message: RRC Connection Request */
{
        INT         BtsNo;
        INT         Frame;
        INT         Lo_Ch;
        INT         Lo_No;
        UCHAR       RcvData[255];

        for(;;){
                RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
                if ( (Frame == RLC_TR_DATA_IND) &&
                        (RcvData[0] == RRC_CONNECTION_REQUEST) ) {
                        break;
                };
        };
};
```

(14) Waits for RRC Connection Request (RACH).

```
/* CCCH RLC Re-Establish */
CrlcConfig(UNIT_BTS1,CRLC_UM_ESTABLISH,CCCH,0,&CrlcConfigCCCH,RRC,NO_TIMEOUT);
```

(15) Changes RLC of CCCH #0 to UM mode, to transfer the next RLC Connection Setup Complete in UM mode.

```
/* Send Message: RRC Connection Setup */
{
        UCHAR       SndData[] = {
                RRC_CONNECTION_SETUP,       /* Message Type */
                        /* UE Information Elements    */
                0x00,           /* Initial UE identity                        */
                0x00,           /* Activation time                            */
                0x00,           /* New U-RNTI                                 */
                0x00,           /* New C-RNTI                                 */
                0x00,           /* UTRAN DRX cycle length coefficient      */
                0x00,           /* Re-establishment timer                     */
                0x00,           /* Capability update requirement              */
                        /* RB Information Elements    */
                0x00,           /* Signalling RB information to setup list    */
                0x00,           /* Signalling RB information to setup         */
                        /* TrCH Information Elements      */
                0x00,           /* Uplink transport channels                                             */
                0x00,           /* UL Transport channel information common for all transport channels    */
                0x00,           /* Added or Reconfigured TrCH information list                          */
                0x00,           /* Added or Reconfigured UL TrCH information                             */
                        /* Downlink transport channels    */
                0x00,           /* DL Transport channel information common for all transport channels    */
                0x00,           /* Added or Reconfigured TrCH information list                          */
                0x00,           /* Added or Reconfigured DL TrCH information                             */
                        /* PhyCH information elements     */
                0x00,           /* Frequency info                 */
                        /* Uplink radio resources     */
                0x00,           /* Maximum allowed UL TX power    */
                0x00,           /* CHOICE channel requirement     */
                0x00,           /* Uplink DPCH info               */
                0x00,           /* PRACH Info (for RACH)          */
                        /* Downlink radio resources */
                0x00,           /* Downlink information common for all radio links    */
                0x00,           /* Downlink information per radio link list           */
                0x00            /* Downlink information for each radio link           */
        };

        SndMessage(UNIT_BTS1, RLC_UM_DATA_REQ, D_CCCH, 0, SndData,sizeof(SndData));
};
```

```
/***** Stand-Alone DCCH Activate *****/
/* Down Link Setup */
CphyRlSetupD_DPCH.SlotFormat          =   SLOT_FORMAT_4;  /* Slot Format 4              */
CphyRlSetupD_DPCH.SymbolRate          =   SYMRATE15K;     /* DPDCH Symbol Rate          */
CphyRlSetupD_DPCH.ChCode              =   30;             /* DPCCH Channelization Code */
CphyRlSetupD_DPCH.Dpdch[0].ChCode     =   30;             /* DPDCH Channelization Code */
CalcRMParameter(D_DPCH,&CphyRlSetupD_DPCH_BTS1,&CphyTrchConfigD_DPCH_SDCCH);
CphyRlSetup     (UNIT_BTS1,D_DPCH,0,&CphyRlSetupD_DPCH_BTS1,     ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,D_DPCH,0,&CphyTrchConfigD_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig      (UNIT_BTS1,D_DPCH,0,&CmacConfigD_DPCH_SDCCH,       ACTIVATE_NOW,NO_TIMEOUT);

/* Up Link Setup */
CphyRlSetupU_DPCH.SlotFormat          =   SLOT_FORMAT_2;  /* Slot Format 2              */
CphyRlSetupU_DPCH.SymbolRate          =   SYMRATE15K;     /* DPDCH Symbol Rate          */
CphyRlSetupU_DPCH.ChCode              =   0;              /* DPCCH Channelization Code */
CphyRlSetupU_DPCH.Dpdch[0].ChCode     =   64;             /* DPDCH Channelization Code */
CphyRlSetupU_DPCH.NumOfDPDCH          =   1;              /* DPDCH Channelization Code */
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH_BTS1,&CphyTrchConfigU_DPCH_SDCCH);
CphyRlSetup     (UNIT_BTS1,U_DPCH,0,&CphyRlSetupU_DPCH_BTS1,     ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_DPCH,0,&CphyTrchConfigU_DPCH_SDCCH,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig      (UNIT_BTS1,U_DPCH,0,&CmacConfigU_DPCH_SDCCH,       ACTIVATE_NOW,NO_TIMEOUT);

/* DCCH RLC Establish */
CrlcConfig(UNIT_BTS1,CRLC_AM_ESTABLISH,DCCH,0,&CrlcConfigDCCH,RRC,NO_TIMEOUT);
```

(16)

Starts the channel for Stand Alone-DCCH. For the parameters that are set in the scenario (such as Slot Format) and also set in Control Software, the settings in the scenario have higher priorities and overwrite those in the latter. Therefore, the setting in Control Software becomes invalid.

```
/* Receive Message:RRC Connection Setup Complete */
{
    INT         BtsNo;
    INT         Frame;
    INT         Lo_Ch;
    INT         Lo_No;
    UCHAR         RcvData[255];

    for(;;){
        RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
        if ( (Frame == RLC_AM_DATA_IND) &&
             (RcvData[0] == RRC_CONNECTION_SETUP_COMPLETE) ) {
            break;
        };
    };
};

/* Receive Message: MM CM Service Request */
{
    INT         BtsNo;
    INT         Frame;
    INT         Lo_Ch;
    INT         Lo_No;
    UCHAR         RcvData[255];

    for(;;){
        RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
        if ( (Frame == RLC_AM_DATA_IND) &&
             (RcvData[0] == INITIAL_DIRECT_TRANSFER) ) {
            break;
        };
    };
};
```

```
/* Send Message: MM Authentication Request           */
{
      UCHAR      SndData[] = {
                  DOWNLINK_DIRECT_TRANSFER,        /* Message Type */
                        /*UE information elements      */
                  0x00,
                        /* CN information elements */
                  0x00,                                   /* CN Domain Identity      */
                  0x00,0x01,0x02,0x03,0x04,      /* NAS message                   */
                  0x05,0x06,0x07,0x08,0x09
      };

      SndMessage(UNIT_BTS1, RLC_AM_DATA_REQ, D_DCCH, 0, SndData,sizeof(SndData));
};

/* Receive Message: MM Authentication Response    */
{
      INT            BtsNo;
      INT            Frame;
      INT            Lo_Ch;
      INT            Lo_No;
      UCHAR          RcvData[255];

      for(;;){
            RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
            if ( (Frame == RLC_AM_DATA_IND) &&
                    (RcvData[0] == UPLINK_DIRECT_TRANSFER) ) {
                  break;
            };
      };
};

/* Send Message: Security Mode Command */
{
      UCHAR       SndData[] = {
                  SECURITY_MODE_COMMAND,      /* Message Type */
                        /* UE Information Elements    */
                  0x00,
                  0x00,
                  0x00,
                  0x00,
                        /* CN Information Elements    */
                  0x00,            /* CN domain identity                              */
      };

      SndMessage(UNIT_BTS1, RLC_UM_DATA_REQ, D_CCCH, 0, SndData,sizeof(SndData));
};

/* Receive Message: Security Mode Complete */
{
      INT            BtsNo;
      INT            Frame;
      INT            Lo_Ch;
      INT            Lo_No;
      UCHAR          RcvData[255];

      for(;;){
            RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
            if ( (Frame == RLC_AM_DATA_IND) &&
                    (RcvData[0] == SECURITY_MODE_COMPLETE) ) {
                  break;
            };
      };
};
```

```
/* Receive Message: CC Setup */
{
    INT         BtsNo;
    INT         Frame;
    INT         Lo_Ch;
    INT         Lo_No;
    UCHAR       RcvData[255];

    for(;;){
        RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
        if ( (Frame == RLC_AM_DATA_IND) &&
             (RcvData[0] == INITIAL_DIRECT_TRANSFER) ) {
            break;
        };
    };
};
```

```
/***** Voice *****/
/* DPCH Down Link */
CphyRlRelease (UNIT_BTS1,D_DPCH,0,ACTIVATE_NOW,NO_TIMEOUT);
CphyRlSetupD_DPCH.SlotFormat          =   SLOT_FORMAT_8;  /* Slot Format 8              */
CphyRlSetupD_DPCH.SymbolRate          =   SYMRATE30K;     /* Symbol Rate                */
CphyRlSetupD_DPCH.ChCode              =   30;             /* DPCCH Channelization Code */
CphyRlSetupD_DPCH.Dpdch[0].ChCode     =   30;             /* DPDCH Channelization Code */
CalcRMParameter(D_DPCH,&CphyRlSetupD_DPCH_BTS1,&CphyTrchConfigD_DPCH_AMR);
CphyRlSetup     (UNIT_BTS1,D_DPCH,0,&CphyRlSetupD_DPCH_BTS1,    ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,D_DPCH,0,&CphyTrchConfigD_DPCH_AMR,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig      (UNIT_BTS1,D_DPCH,0,&CmacConfigD_DPCH_AMR,       ACTIVATE_NOW,NO_TIMEOUT);

/* DPCH Up Link */
CphyRlRelease (UNIT_BTS1,U_DPCH,0,ACTIVATE_NOW,NO_TIMEOUT);
CphyRlSetupU_DPCH.SlotFormat          =   SLOT_FORMAT_2;  /* Slot Format 2              */
CphyRlSetupU_DPCH.SymbolRate          =   SYMRATE60K;     /* DPDCH Symbol Rate          */
CphyRlSetupU_DPCH.ChCode              =   0;              /* DPCCH Channelization Code */
CphyRlSetupU_DPCH.Dpdch[0].ChCode     =   16;             /* DPDCH Channelization Code */
CalcRMParameter(U_DPCH,&CphyRlSetupU_DPCH_BTS1,&CphyTrchConfigU_DPCH_AMR);
CphyRlSetup     (UNIT_BTS1,U_DPCH,0,&CphyRlSetupU_DPCH_BTS1,    ACTIVATE_NOW,NO_TIMEOUT);
CphyTrchConfig(UNIT_BTS1,U_DPCH,0,&CphyTrchConfigU_DPCH_AMR,ACTIVATE_NOW,NO_TIMEOUT);
CmacConfig      (UNIT_BTS1,U_DPCH,0,&CmacConfigU_DPCH_AMR,       ACTIVATE_NOW,NO_TIMEOUT);


/* DTCH(Voice) RLC Establish */
CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,DCCH,0,&CrlcConfigDCCH,TE,NO_TIMEOUT);
CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,DTCH,0,&CrlcConfigDTCH,TE,NO_TIMEOUT);
CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,DTCH,1,&CrlcConfigDTCH,TE,NO_TIMEOUT);
CrlcConfig(UNIT_BTS1,CRLC_TR_ESTABLISH,DTCH,2,&CrlcConfigDTCH,TE,NO_TIMEOUT);
```

(17) Starts the channel of AMR-Voice+DCCH. Also see Part (16).

```
/* Send Message: CC Call Proceeding */
{
      UCHAR      SndData[] = {
                  DOWNLINK_DIRECT_TRANSFER,       /* Message Type */
                        /*UE information elements      */
                  0x00,
                        /* CN information elements */
                  0x00,                                  /* CN Domain Identity      */
                  0x00,0x01,0x02,0x03,0x04,      /* NAS message                  */
                  0x05,0x06,0x07,0x08,0x09,
                  0x10,0x11,0x12,0x13,0x14,
                  0x15,0x16,0x17,0x18,0x19,
                  0x20,0x21,0x22,0x23,0x24,
                  0x25,0x26,0x27,0x28,0x29,
                  0x30,0x31,0x32,0x33,0x34,
                  0x35,0x36,0x37,0x38,0x39,
                  0x40,0x41,0x42,0x43,0x44,
                  0x45,0x46,0x47,0x48,0x49,
                  0x50,0x51,0x52,0x53,0x54,
                  0x55,0x56,0x57,0x58,0x59
      };

      SndMessage(UNIT_BTS1, RLC_AM_DATA_REQ, D_DCCH, 0, SndData,sizeof(SndData));
};

/* Send Message: Radio Bearer Setup */
{
      UCHAR       SndData[] = {
                  RADIO_BEARER_SETUP,      /* Message Type */
                        /* UE Information Elements    */
                  0x00,            /* Initial check info                              */
                  0x00,
                  0x00,
                  0x00,            /* Activation time                                 */
                  0x00,            /* New U-RNTI                                      */
                  0x00,            /* New C-RNTI                                      */
                  0x00,            /* DRX indicator                                   */
                  0x00,            /* UTRAN DRX cycle length coefficient      */
                  0x00,            /* Re-establishment timer                          */
                        /* CN Information Elements    */
                  0x00,            /* CN information info                             */
                        /* RB Information Elements    */
                  0x00,            /* Signalling RB information to setup      */
                  0x00,            /* RAB information for setup                 */
                  0x00,            /* RB information to be affected             */
                        /* TrCH Information Elements (Uplink)    */
                  0x00,            /* UL TrCh info common for all TrChs        */
                  0x00,            /* Deleted UL TrCH information                  */
                  0x00,            /* Added or Reconfigured UL TrCH info      */
                  0x00,            /* CPCH set ID                                     */
                  0x00,            /* DRAC static information                         */
                        /* TrCH Information Elements (Downlink)    */
                  0x00,            /* DL TrCh info common for all TrChs        */
                  0x00,            /* Deleted UL TrCH information                  */
                  0x00,            /* Added or Reconfigured UL TrCH info      */
                        /* PhyCH Information Elements */
                  0x00,            /* Frequency info                                  */
                        /* Uplink radio resources */
                  0x00,            /* Maximum allowed UL TX power                  */
                  0x00,            /* Uplink DPCH info                                */
                  0x00,            /* PRACH Info (for RACH)                           */
                        /* Downlink radio resources */
                  0x00,            /* DL info common for all radio links      */
                  0x00,            /* Downlink PDSCH information                   */
                  0x00,            /* CPCH SET Info                                   */
      };

      SndMessage(UNIT_BTS1, RLC_UM_DATA_REQ, D_DCCH, 0, SndData,sizeof(SndData));
};
```

```
/* Receive Message: Radio Bearer Setup Complete */
{
    INT         BtsNo;
    INT         Frame;
    INT         Lo_Ch;
    INT         Lo_No;
    UCHAR       RcvData[255];

    for(;;){
        RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
        if ( (Frame == RLC_AM_DATA_IND) &&
             (RcvData[0] == RADIO_BEARER_SETUP_COMPLETE) ) {
            break;
        };
    };
};

/* Send Message: CC Alerting */
{
    UCHAR    SndData[] = {
            DOWNLINK_DIRECT_TRANSFER,       /* Message Type */
                /*UE information elements      */
            0x00,
                /* CN information elements */
            0x00,                           /* CN Domain Identity      */
            0x00,0x01,0x02,0x03,0x04,      /* NAS message             */
            0x05,0x06,0x07,0x08,0x09,
            0x10,0x11,0x12,0x13,0x14,
            0x15,0x16,0x17,0x18,0x19,
            0x20,0x21,0x22,0x23,0x24,
            0x25,0x26,0x27,0x28,0x29,
            0x30,0x31,0x32,0x33,0x34,
            0x35,0x36,0x37,0x38,0x39,
            0x40,0x41,0x42,0x43,0x44,
            0x45,0x46,0x47,0x48,0x49,
            0x50,0x51,0x52,0x53,0x54,
            0x55,0x56,0x57,0x58,0x59
    };

    SndMessage(UNIT_BTS1, RLC_AM_DATA_REQ, D_DCCH, 0, SndData,sizeof(SndData));
};

/* Send Message: CC Connect */
{
    UCHAR    SndData[] = {
            DOWNLINK_DIRECT_TRANSFER,       /* Message Type */
                /*UE information elements      */
            0x00,
                /* CN information elements */
            0x00,                           /* CN Domain Identity      */
            0x00,0x01,0x02,0x03,0x04,      /* NAS message             */
            0x05,0x06,0x07,0x08,0x09,
            0x10,0x11,0x12,0x13,0x14,
            0x15,0x16,0x17,0x18,0x19,
            0x20,0x21,0x22,0x23,0x24,
            0x25,0x26,0x27,0x28,0x29,
            0x30,0x31,0x32,0x33,0x34,
            0x35,0x36,0x37,0x38,0x39,
            0x40,0x41,0x42,0x43,0x44,
            0x45,0x46,0x47,0x48,0x49,
            0x50,0x51,0x52,0x53,0x54,
            0x55,0x56,0x57,0x58,0x59
    };

    SndMessage(UNIT_BTS1, RLC_AM_DATA_REQ, D_DCCH, 0, SndData,sizeof(SndData));
};
```

```
    /* Receive Message: CC Connect Acknowledge */
    {
        INT         BtsNo;
        INT         Frame;
        INT         Lo_Ch;
        INT         Lo_No;
        UCHAR       RcvData[255];

        for(;;){
            RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, NO_TIMEOUT );
            if ( (Frame == RLC_AM_DATA_IND) &&
                 (RcvData[0] == UPLINK_DIRECT_TRANSFER) ) {
                break;
            };
        };
    };
    /* TE Configuration */
    {
      CTE_CONFIG_PAR   CteConfigAmrA;   /* Class A Config Parameter */
      CTE_CONFIG_PAR   CteConfigAmrB;   /* Class B Config Parameter */
      CTE_CONFIG_PAR   CteConfigAmrC;   /* Class C Config Parameter */

      CteConfigAmrA.TeType    = TE_TYPE_VOICE_AMR_A;
      CteConfigAmrA.Rate      = VOICE_RATE_12_2;
      CteConfigAmrA.TTI       = 2;
      CteConfigAmrA.NumOfTB = 1;
      CteConfigAmrA.TBS       = 81;
      CteConfigAmrA.Frame     = RLC_TR_DATA_REQ;
      CteConfigAmrA.Layer     = RLC;

      CteConfigAmrB.TeType    = TE_TYPE_VOICE_AMR_B;
      CteConfigAmrB.Rate      = VOICE_RATE_12_2;
      CteConfigAmrB.TTI       = 2;
      CteConfigAmrB.NumOfTB = 1;
      CteConfigAmrB.TBS       = 103;
      CteConfigAmrB.Frame     = RLC_TR_DATA_REQ;
      CteConfigAmrB.Layer     = RLC;

      CteConfigAmrC.TeType    = TE_TYPE_VOICE_AMR_C;
      CteConfigAmrC.Rate      = VOICE_RATE_12_2;
      CteConfigAmrC.TTI       = 2;
      CteConfigAmrC.NumOfTB = 1;
      CteConfigAmrC.TBS       = 60;
      CteConfigAmrC.Frame     = RLC_TR_DATA_REQ;
      CteConfigAmrC.Layer     = RLC;

      CteConfig( Loch, 0, &CteConfigAmrA, NO_TIMEOUT );
      CteConfig( Loch, 1, &CteConfigAmrB, NO_TIMEOUT );
      CteConfig( Loch, 2, &CteConfigAmrC, NO_TIMEOUT );
    }

    /* Call Processing */
    {
      INT   DownPort;   /* DownLink Port   */
      INT   UpPort;     /* UpLink Port     */
      INT   Direction; /* Call Direction */

      DownPort   = TE_PORT_NORMAL;
      UpPort     = TE_PORT_NORMAL;
      Direction = CALL_FROM_AIR;

      CteConnect( Loch, 0, DownPort, UpPort, Direction, (UCHAR *)0, NO_TIMEOUT );
      CteConnect( Loch, 1, DownPort, UpPort, Direction, (UCHAR *)0, NO_TIMEOUT );
      CteConnect( Loch, 2, DownPort, UpPort, Direction, (UCHAR *)0, NO_TIMEOUT );
    }

    return(0);
}
```

(18)

Connects AMR-Voice data to the TE of the Signalling Tester side.
For details, see "Appendix B. DETAILS OF HARIKIRI (PHYSICAL LAYER) TEST" or "Appendix C DETAILS OF TE

E. GSM/GPRS OPTION

This appendix explains the GSM/GPRS option of Signalling Tester. The GSM/GPRS Option enables various GSM/GPRS tests including GSM-GSM Handover and GSM-W-CDMA Handover. GSM Circuit Switched Data(CSD) and Frequency Hopping are also possible if the additional software option is installed.

(Note) Please contact MD8480A-G-support@zy.anritsu.co.jp if you need the detailed information regarding the option installation information.

E.1 Outline of GSM/GPRS option

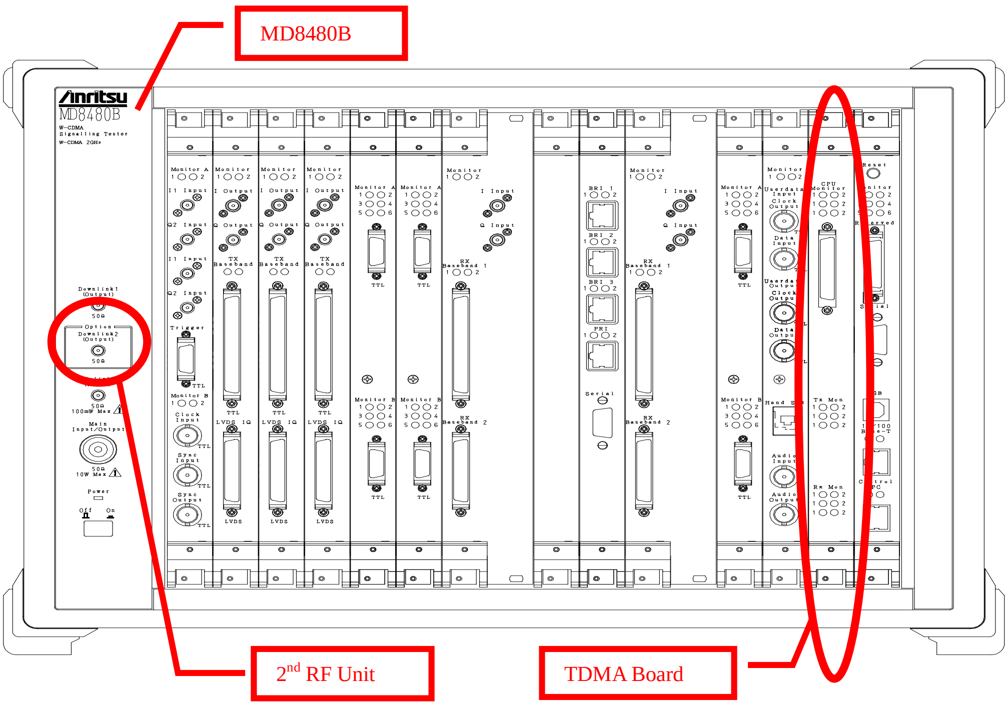
The outline of the GSM/GPRS OPTION is as follows.

* Basic function of GSM/GPRS BTS L1 and L2.
* Execution of L3 functionalities by scenario
*RF Channel of CCH, TCH, PCCH and PTCH for UL and DL are supported.
* Each RF Channel can be set to an arbitrary frequency and the transmission power.
(Note) Refer to the Release Note “MD8480ReleaseNote_VXXX_CE.pdf” of the latest version software about the current restrictions.

E.2 Option Composition needed for GSM/GPRS option

GSM/GPRS option needs the following options besides a basic composition of MD8480B Signalling Tester. Refer to the Operation Manual section 1.3.2 about the details.

- TDMA board
- Additional RF Unit



E.3 Required Software components for GSM/GPRS Option

The following Software components are necessary for the GSM/GPRS option.
*MX848000A (Control PC software)
*MX848001A (Firmware)
*MX848005B (GSM/GPRS)
*MX848002A (FPGA)

E.3.1 Software option

The following options can be added to GSM/GPRS option.
*MX848001A-04(GSM CSD)
*MX848001A-05(GSM Frequency Hopping)

You can confirm by the following method whether your Signalling Tester has the above option or not.

How to confirm the software option,

Confirm that the string below is appeared in the SYSTEM Window in the Control Software when Information Read button is clicked.

CSD	(GSM CSD Option)
GSM/GPRS F.H	(GSM Frequency Hopping Option)

E.4 GSM/GPRS Option operation

E.4.1 Connection and Operation of equipment

Connection and operation of equipment for GSM/GPRS option is almost the same as WCDMA, so refer to Section2 of this document. Operations specific to the GSM/GPRS option are described in the following.

E.4.2 Control Software GUI specific to GSM/GPRS option

This section describes the Control Software GUI specific to the GSM/GPRS Option.

E.4.2.1 Setup screen

SETUP Button

Follow the procedure below to use the GSM/GPRS option.

1) Click “SETUP” button on Control Software, it will appear “Parameter Setup” screen shown in figure E4-1.

2) Set “Base Band and RF Unit Connection” to ”W-CDMA+GSM”. It is possible to send/receive by RF for W-CDMA(Tx/Rx Unit#1) and RF for GSM/GPRS(Tx/Rx Unit#2) by this setting.

3) Select “Connector” either “Main” or “Sub”.(“Analog IQ” and “Digital IQ” cannot be used for the GSM/GPRS option).

4) Click “TX RF#2” and input appropriate Tx Attenuator level for your test environment. The actual transmission power of each GSM/GPRS RF channel is defined by the equation below..

[Actual downlink power for each GSM/GPRS channel] =
[Channel power set by GsmRfchConfig() in Scenario] - [Tx Attenuator level]

5) Click “RX RF#2” and input Reference Power appropriately for your test environment. The appropriate Reference Power is about -5dB from the input level at Signalling Tester(Refer to the Operation Manual chapter S.3 for details).

6) LVDS should be set to disable.

(Note) Parameters for GSM/GPRS and W-CDMA can be set with the “Parameter Setup” screen and also by the Scenario. Scenario has higher priority than on the Setup screen if the same parameters are set. GSM/GPRS configures with the “Parameter Setup” screen can be only Tx Attenuator and Reference level. Other settings such as frequency should be set from the scenarios.

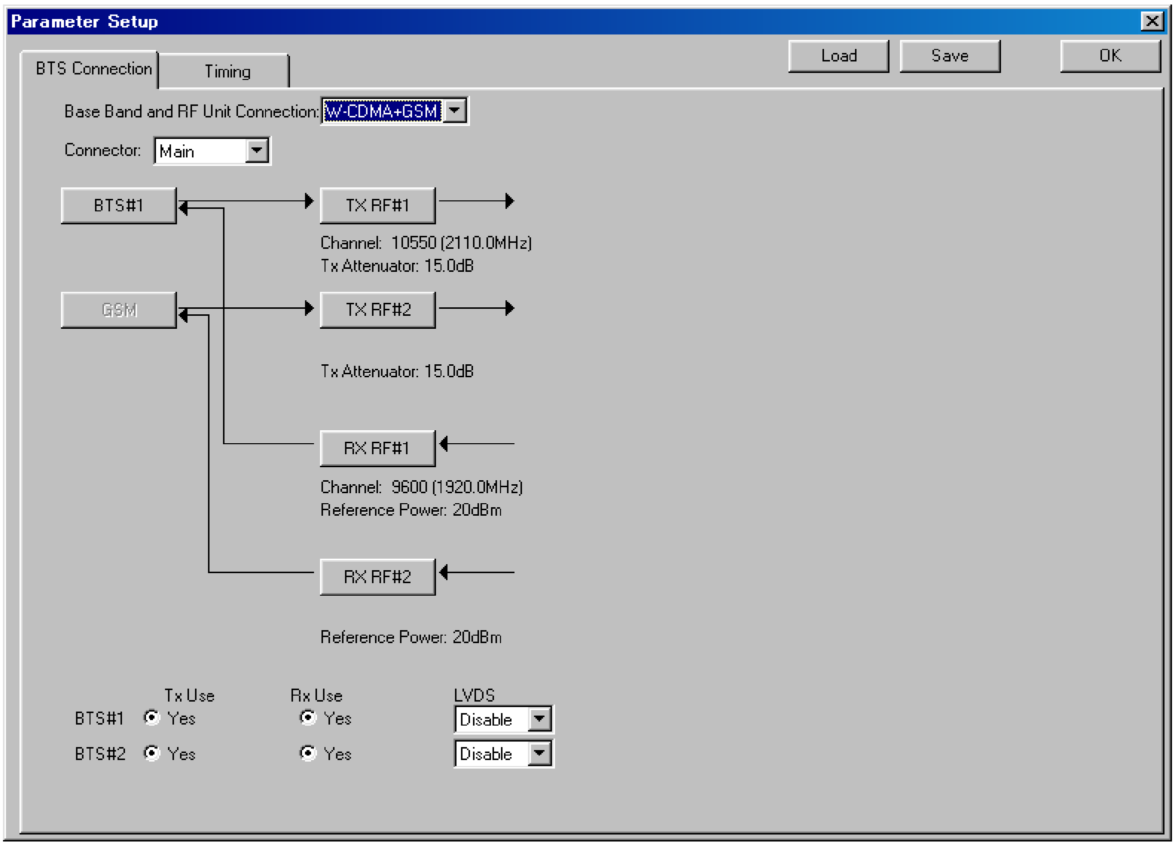


Figure E4-1 Parameter setup screen

Scenario Button
Click on “Scenario” button and select scenario that you plan to execute.

START Button
Start simulation by clicking “START” button.

TRACE Button
Display the trace during simulation by clicking “TRACE” button. Primitives for both of W-CDMA and GSM/GPRS will be displayed on the trace screen.

E.4.2.2 Trace screen

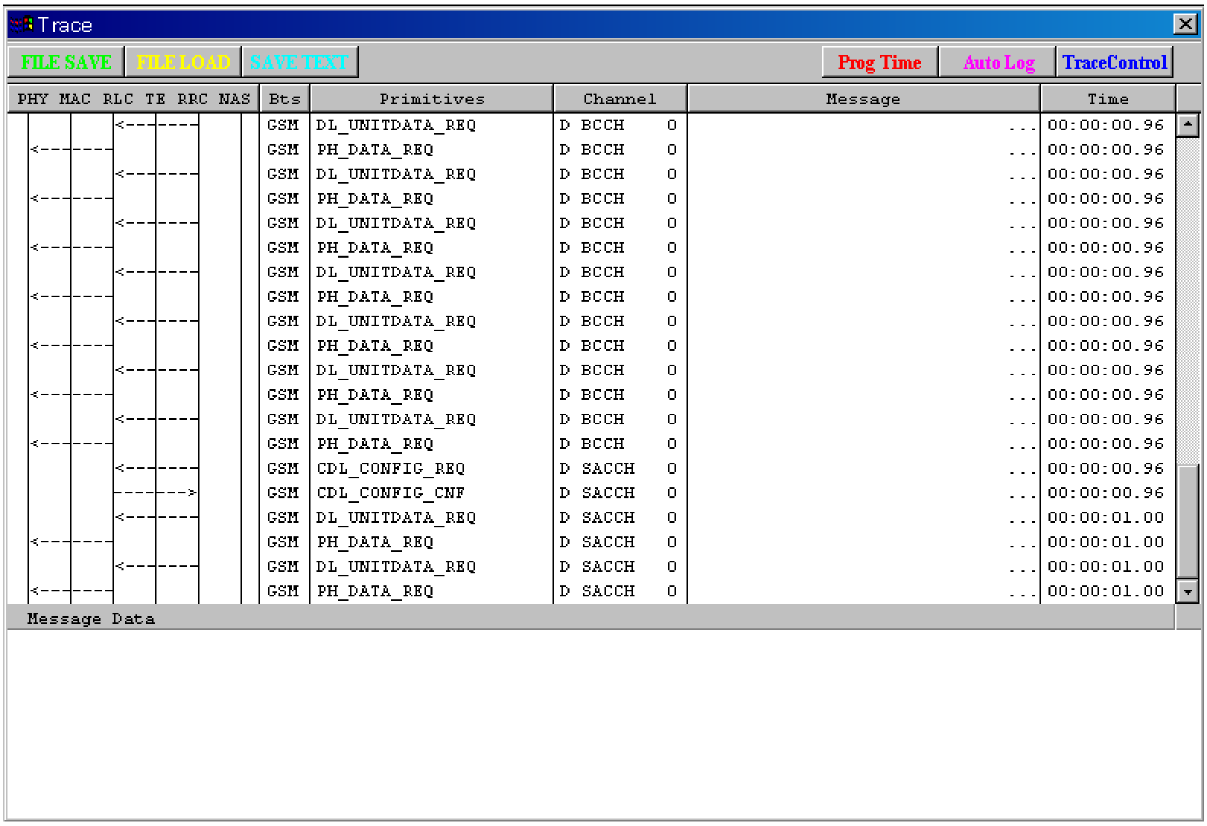
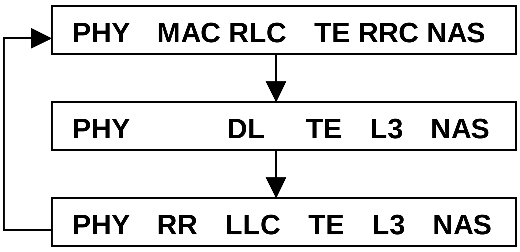


Figure E4-2 Trace screen

“PHY MAC RLC TE RRC NAS” Display

Clicking the “PHY MAC RLC TE RRC NAS”, it changes as below,



“Bts” Display

For the GSM/GPRS Primitives, ”GSM” will be displayed on “Bts” screen.

“Primitives” Display

The GSM/GPRS Primitives are defined in
C:¥Mx848000¥Scenario¥include¥primitive.h.

“Channel” Display

Channels used in GSM/GPRS are defined in
C:¥Mx848000¥Scenario¥include¥primitive.h.
If U PACCH or U PDTCH, it displays received Timeslot No.

“Message Data” Display

You can refer to specific contents of Primitive message. Type indicates define value of Primitive and Channel indicates define value of Channel.

The trace screen is processed as follows.
(1) When it cannot receive the burst for one data
  It doesn't display it on the trace screen.
(2) When it receives the burst for one data and the decode result is NG.
  It displays PH_CRCERR_IND on the trace screen.
  (Note) Only the channel that the CRC check code is included in the PHY Layer is an object.
  If the burst becomes complete, the channel that the CRC check code is not included in the PHY Layer (ex. Data TCH) is decoded and it is processed as following (3).
(3) When it receives the burst for one data and the decode result is OK.
  It displays excluding PH_CRCERR_IND (PH_DATA_IND etc.) on the trace screen.

E.4.2.3 Trace Control screen

The channel whose tick is removed will not be displayed on the Trace screen.

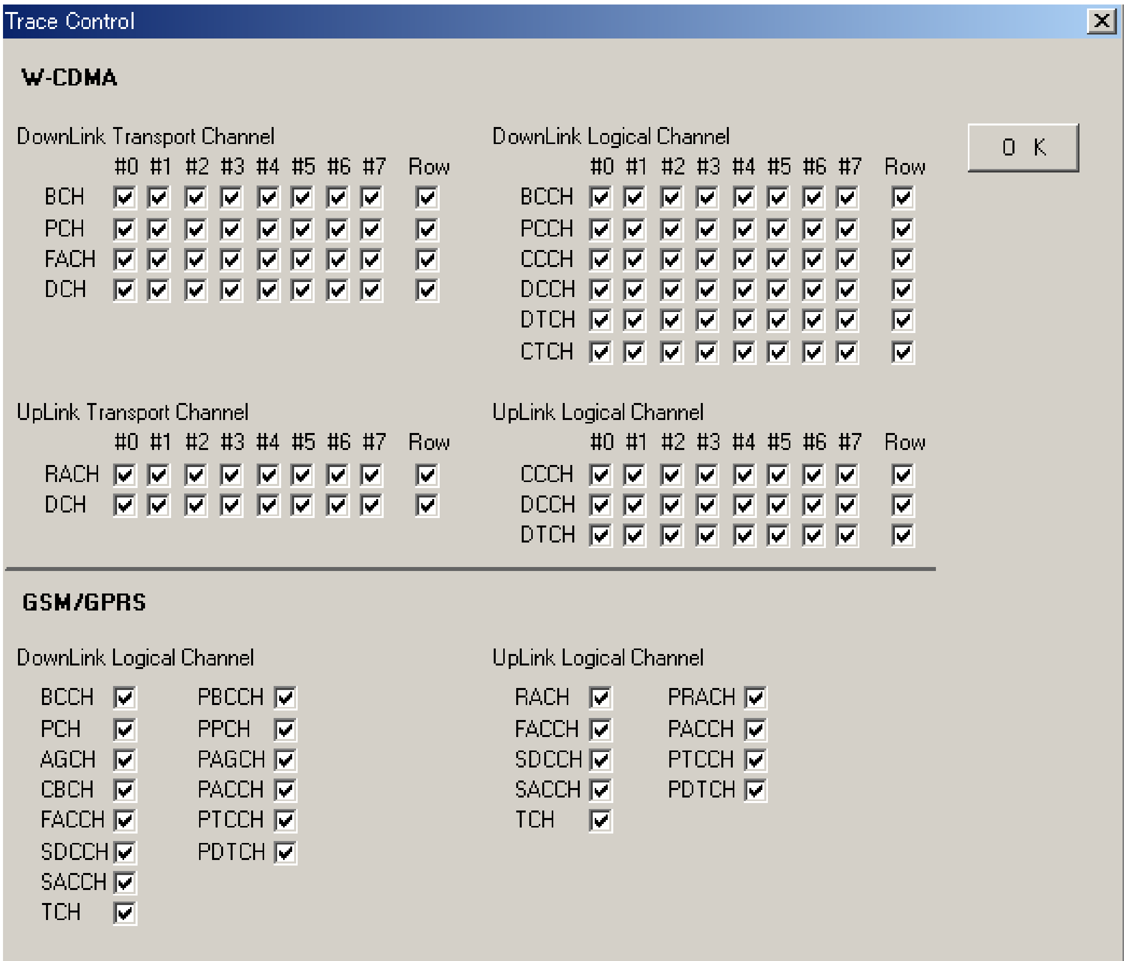


Figure E4-3 Trace Control screen

E.4.2.4 Monitor screen

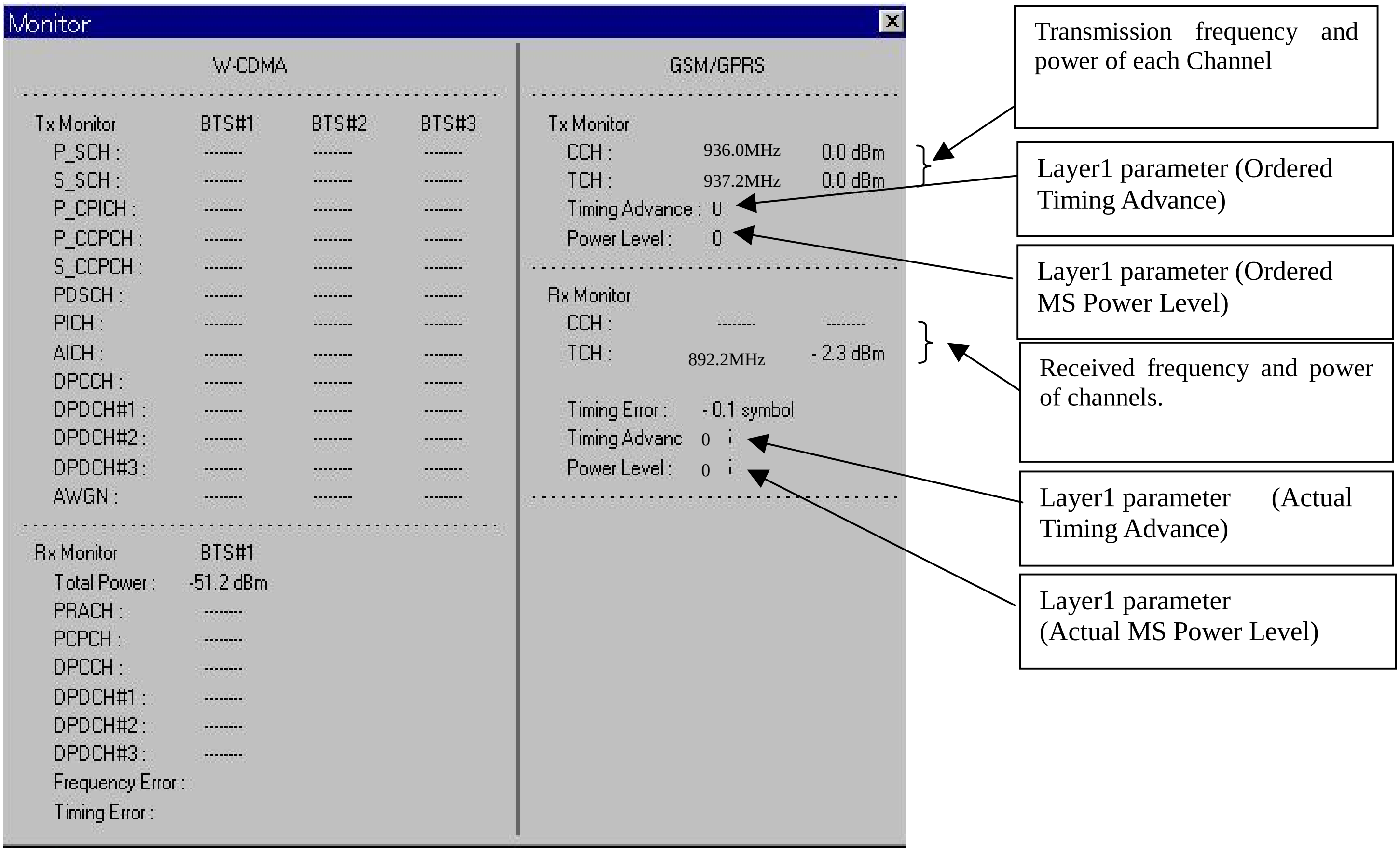


Figure E4-4 Monitor screen

”Timing Error”

“Timing Error” indicates the uplink timing error from the correct one. It is updated and calculated when Signalling Tester detects the TSC. It has the range of -63 to 63[symbol] from the center of Timing Advance of 0 to 219 which is specified by Scenario. The resolution is 0.1[symbol].

(Note) The displayed items in the Monitor screen except “Timing Error” are not confirmed. Please use the items as a rough result.

E.4.3 Monitor and LED specifications

The specifications of the Monitor terminals and LEDs of the TDMA board are as follows.
(Note) There is a case that specification of monitor is changed without notice.

Table E4-1 Monitor terminal

Signal name	Description	I/O	Remarks	Pin No.
CCH Coding	Outputs “L/H” alternately during transmitting CCH. (connected with Tx Monitor LED1)	O		1
TCH Coding	Outputs “L/H” alternately during transmitting TCH. (connected with Tx Monitor LED2)	O		2
PTCH Coding	Outputs “L/H” alternately during transmitting PTCH. (connected with Tx Monitor LED3)	O		3
BCCH Frame	Outputs “H” alternately when transmitting BCCH Frame. (connected with Tx Monitor LED4)	O		4
	Reserved	O		5
	Reserved	O		6
	Reserved	O		7
	Reserved	O		8
GND	Ground	O		9
GND	Ground	O		10
Tx CCH	Outputs “H” alternately when starting up CCH. (connected with Tx Monitor LED5)	O		11
Tx TCH	Outputs “H” alternately when starting up TCH. (connected with Tx Monitor LED6)	O		12
	Reserved	O		13
	Reserved	O		14
	Reserved	O		15
	Reserved	O		16
	Reserved	O		17
	Reserved	O		18
GND	Ground	O		19
GND	Ground	O		20
Rx CCH	Outputs “L/H “alternately when CCH CRC=OK. (connected with Rx Monitor LED1)	O		21
Rx TCH	Outputs “L/H “alternately when TCH CRC=OK. (connected with Rx Monitor LED2)	O		22
Rx PTCH	Outputs “L/H “alternately when PTCH CRC=OK. (connected with Rx Monitor LED3)	O		23
	Reserved	O		24
	Reserved	O		25
	Reserved	O		26
	Reserved	O		27
	Reserved	O		28
GND	Ground	O		29
GND	Ground	O		30
RACH	Outputs “H” when detecting RACH. (connected with Rx Monitor LED5)	O		31
TSC	Outputs “H” when detecting TSC. (connected with Rx Monitor LED6)	O		32
	Reserved	O		33
RACH	Outputs “H” when detecting RACH.	O		34
TSC	Outputs “H” when detecting TSC.	O		35

Signal name	Description	I/O	Remarks	Pin No.
Level Over	Outputs “H” when input level is -3dB more than saturate level.	O		36
	Reserved	O		37
	Reserved	O		38
TX Frame Timing	Outputs “H” by timing of Frame. (There is some constant timing offset against the RF signal timing.)	O		39
TX Slot Timing	Outputs “H” by timing of Slot. (There is some constant timing offset against the RF signal timing.)	O		40
Reserved	Reserved	O		41
Power	Outputs “H” when powered ON. (connected with CPU Monitor LED2)	O		42
	Reserved	O		43
Active	Outputs “L/H” alternately during operation. (connected with CPU Monitor LED 4)	O		44
	Reserved	O		45
Active	Outputs “L/H” alternately during operation. (connected with CPU Monitor LED6)	O		46
	Reserved	O		47
	Reserved	O		48
RX Frame Timing	Outputs “H” by timing of Frame. (There is some constant timing offset against the RF signal timing.)	O		49
RX Slot Timing	Outputs “H” by timing of Slot. (There is some constant timing offset against the RF signal timing.)	O		50

Table E4-2 LED

Input/output connector	Description	Signal level/Specifications	Connector type
CPU Monitor	Operation monitor LED 1: 2:Power 3: 4: 5: 6:	 Not used Goes on when powered on Not used Blinks during operating Not used Blinks during operating	LED
TX Monitor	Operation monitor LED 1:CCH 2:TCH 3:PTCH 4:BCCH Frame 5:CCH act 6:TCH act	 Blinks when CCH is transmitted. Blinks when TCH is transmitted. Blinks when PTCH is transmitted. Blinks when BCCH Frame is transmitted. Goes on when CCH is activated. Goes on when TCH is activated.	LED
RX Monitor	Operation monitor LED 1:CCH 2:TCCH 3:PTCH 4: 5:RACH 6:TSC	 Blinks when CCH is transmitted. Blinks when TCH is transmitted. Blinks when PTCH is transmitted. Not used Goes on when RACH is activated. Goes on when TSC is activated.	LED

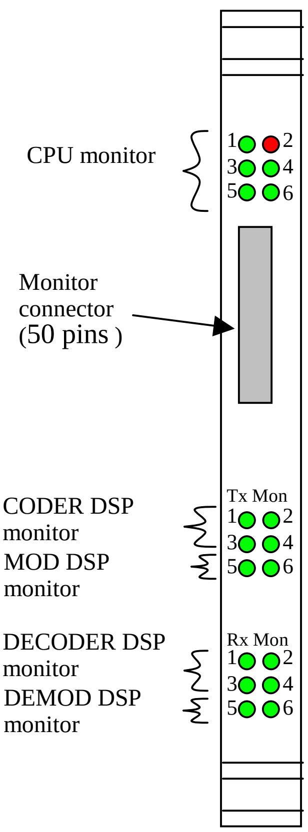


Figure E4-5 Monitor of TDMA board

E.5 GSM/GPRS tests

E.5.1 GSM call origination and termination test

This section describes a test for GSM call origination and termination.

E.5.1.1 Connection of equipment

Refer to section 2.2 about connection of the equipment.

E.5.1.2 Execution order of the sample scenarios

Compile the related scenarios in advance(refer to section 2.4.4 how to compile a scenario).

How to test GSM speech call:

1. Confirm the settings.

   Confirm that "Baseband and RF Connection" of SETUP screen in Control Software is "W-CDMA+GSM". Also confirm "Tx Attenuator" and "Reference Power" of RF#2 are set to the appropriate value(Refer to chapter E.4.2.1).
2. Execute GSM_Idle.c.

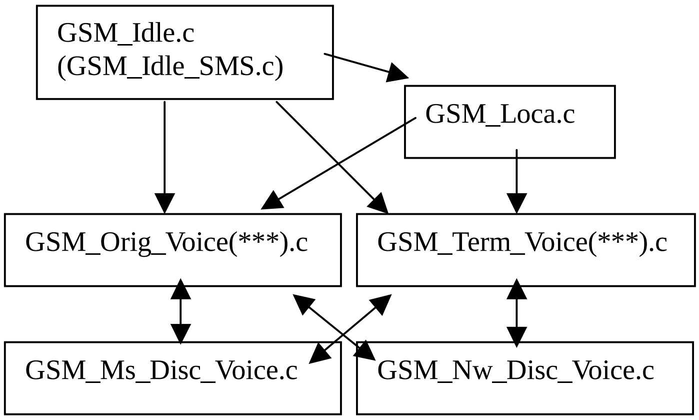
   This scenario starts CCH Rf Channel, configures the data link for BCCH, PCH, AGCH, RACH, SACCH, and set the broadcasting information.
3. Execute GSM_Loca.c if the MS has not been registered.

   This scenario executes registration(location updating) procedure after configuring a TCH RF channel(= SDCCH/8 + SACCH/8). Completing the registration, TCH Rf Channel is going to be released.
4. Execute GSM_Orig_Voice(***)c or GSM_Term_Voice(***).c.

   These scenarios execute call origination(calling from MS) and call termination (calling from NW).
5. Execute GSM_IntraSystem_HO.c if needed.

   This scenario executes Intra-System(GSM-GSM) Handover. It changes frequency and Timeslot of TCH during call.
6. Execute GSM_Ms_Disc_Voice.c or GSM_Nw_Disc_Voice.c.

   These scenarios execute call disconnection(from MS or NW).



(Note)*** is AFS,AHS,EFS,FS or HS.

*Executes GSM_Idle_SMS.c first for SMS test.

In idle status

- GSM CellReselection.c
- GSM_Orig_SMSonSDCCH.c
- GSM_Term_SMSonSDCCH.c

In call active status

- GSM_InterSystem_HO.c
- GSM_IntraSystem_HO.c
- GSM_Orig_SMSonSACCH.c
- GSM_Term_SMSonSACCH.c

How to test GSM SMS:

1. Confirm the settings.

   Confirm that "Baseband and RF Connection" of SETUP screen in Control Software is "W-CDMA+GSM". Also confirm "Tx Attenuator" and "Reference Power" of RF#2 are set to the appropriate value(Refer to chapter E.4.2.1).

2. Execute GSM_Idle_SMS.c.

   This scenario starts CCH Rf Channel, configures the data link for BCCH, PCH, AGCH, RACH, SACCH, and set the broadcasting information.

3. Execute GSM_Loca.c if the MS has not been registered.

   This scenario executes registration(location updating) procedure after configuring a TCH RF channel. Completing the registration, TCH Rf Channel is going to be released.

4. Execute GSM_Orig_SMSonSDCCH.c or GSM_Term_SMSonSDCCH.c.

   These scenarios execute SMS using SDCCH from MS(GSM_Orig_SMSon SDCCH.c) or SMS from NW(GSM_Term_SMSonSDCCH) .

5. Execute GSM_Orig_Voice(***).c or GSM_Term_Voice(***).c

   These scenarios execute call origination(calling from MS) and call termination (calling from NW).

6. Execute GSM_Orig_SMSonSACCH.c or GSM_Term_SMSonSACCH.c.

   These scenarios execute SMS using SACCH from MS(GSM_Orig_SMSon SACCH.c) or from NW(GSM_Term_SMSonSACCH) .

7. Execute GSM_Ms_Disc_Voice.c or GSM_Nw_Disc_Voice.c.

   These scenarios execute call disconnection(from MS or NW).

How to test GSM SMSCB:

GSM_SMSCBonSDCCH***(repetition).c can send the uniform CBS Messages on CBCH at the designated intervals.
GSM_SMSCBonSDCCH***(DRX).c can send one Schedule Message and one scheduled CBS Message on CBCH.
Before testing SMSCB, enable MS to accept SMSCB. MS needs to finish registration (location updating) procedure with GSM_Idle.c and GSM_Loca.c(Refer to chapter E.5.1.2).

1. GSM_SMSCBonSDCCH***(repetition).c

1-1. Confirm the settings.

Confirm that "Baseband and RF Connection" of SETUP screen in Control Software is "W-CDMA+GSM". Also confirm "Tx Attenuator" and "Reference Power" of RF#2 are set to the appropriate value(Refer to chapter E.4.2.1).

1-2. Execute GSM_SMSCBonSDCCH***(repetition).c

Set the parameters by selecting the displayed buttons. Input the character string included in CBS Message within 93 characters. After pushing " START" button, SMSCB starts.

2. GSM_SMSCBonSDCCH***(DRX).c

2-1. Confirm the settings.

Confirm that "Baseband and RF Connection" of SETUP screen in Control Software is "W-CDMA+GSM". Also confirm "Tx Attenuator" and "Reference Power" of RF#2 are set to the appropriate value(Refer to chapter E.4.2.1).

2-2. Execute GSM_SMSCBonSDCCH***(DRX).c

Set the parameters by selecting the displayed buttons. Input the character string included in CBS Message within 93 characters. After pushing " START" button, SMSCB starts.

E.5.2 GPRS data communication test(Between Server and Client PC)

This section describes a test for GPRS data communication between Server and Client PC.

E.5.2.1 Connection of equipment

Connect the equipment as figure E5-1.

(Note) GPRS data communication throughput may changed by type and setting of mobile station, server PC and Client PC.
Throughput degradation or abnormal termination may be occurred if followings are applied in.
- Protocol except TCP/IP is enabled in Server and/or Client PC
- Anti-virus software is enabled in Server and/or Client PC.

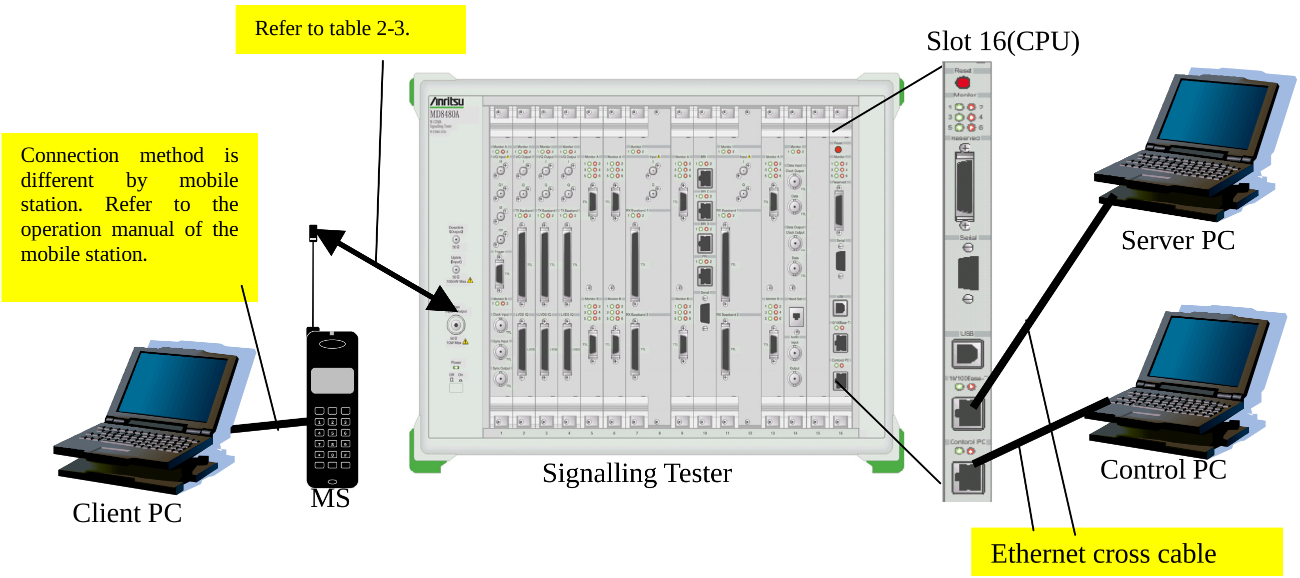


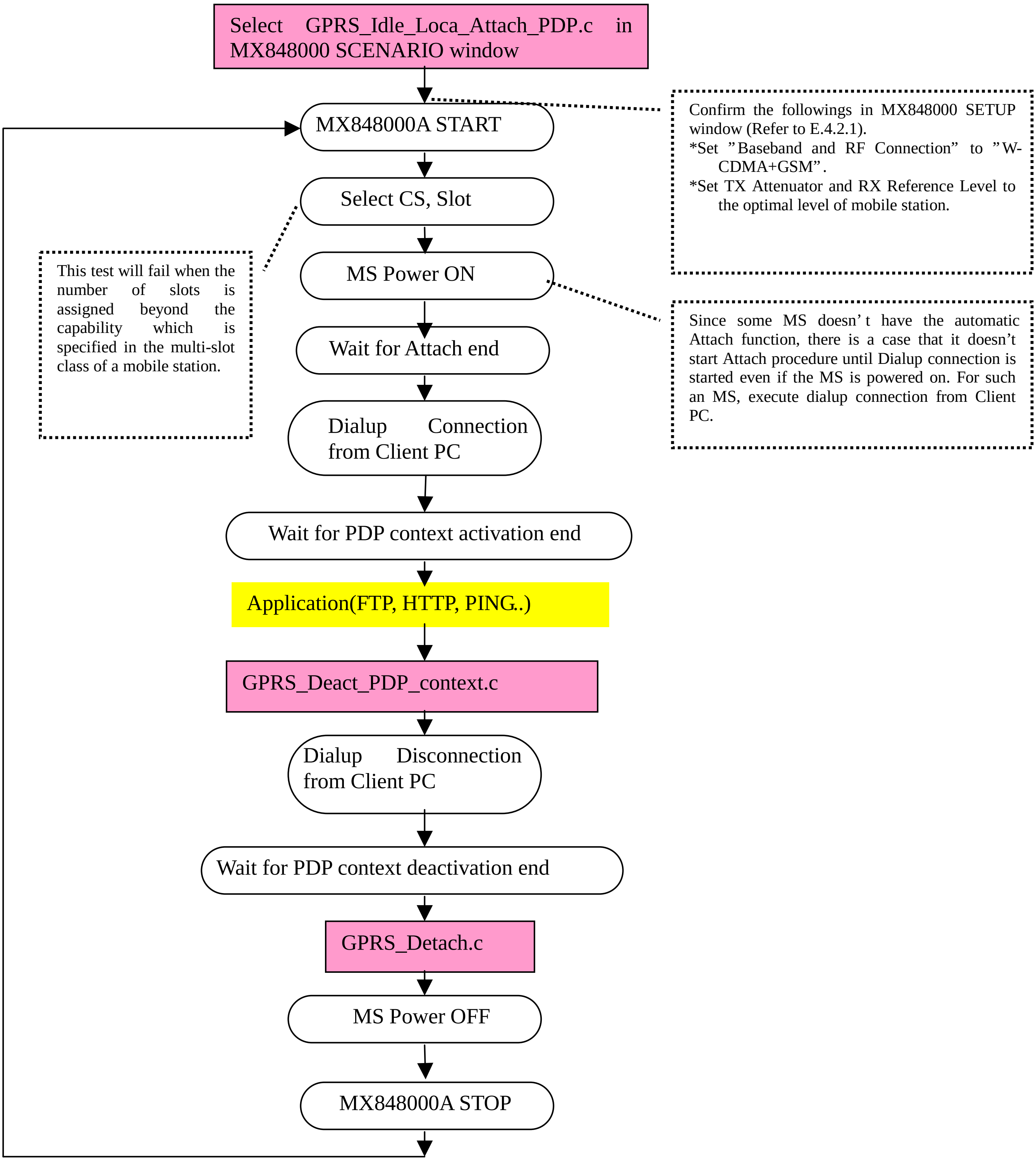
Figure E5-1 connection of equipment

E.5.2.2 Execution order of the sample scenarios

Compile the related scenarios in advance(refer to section 2.4.4 how to compile a scenario).

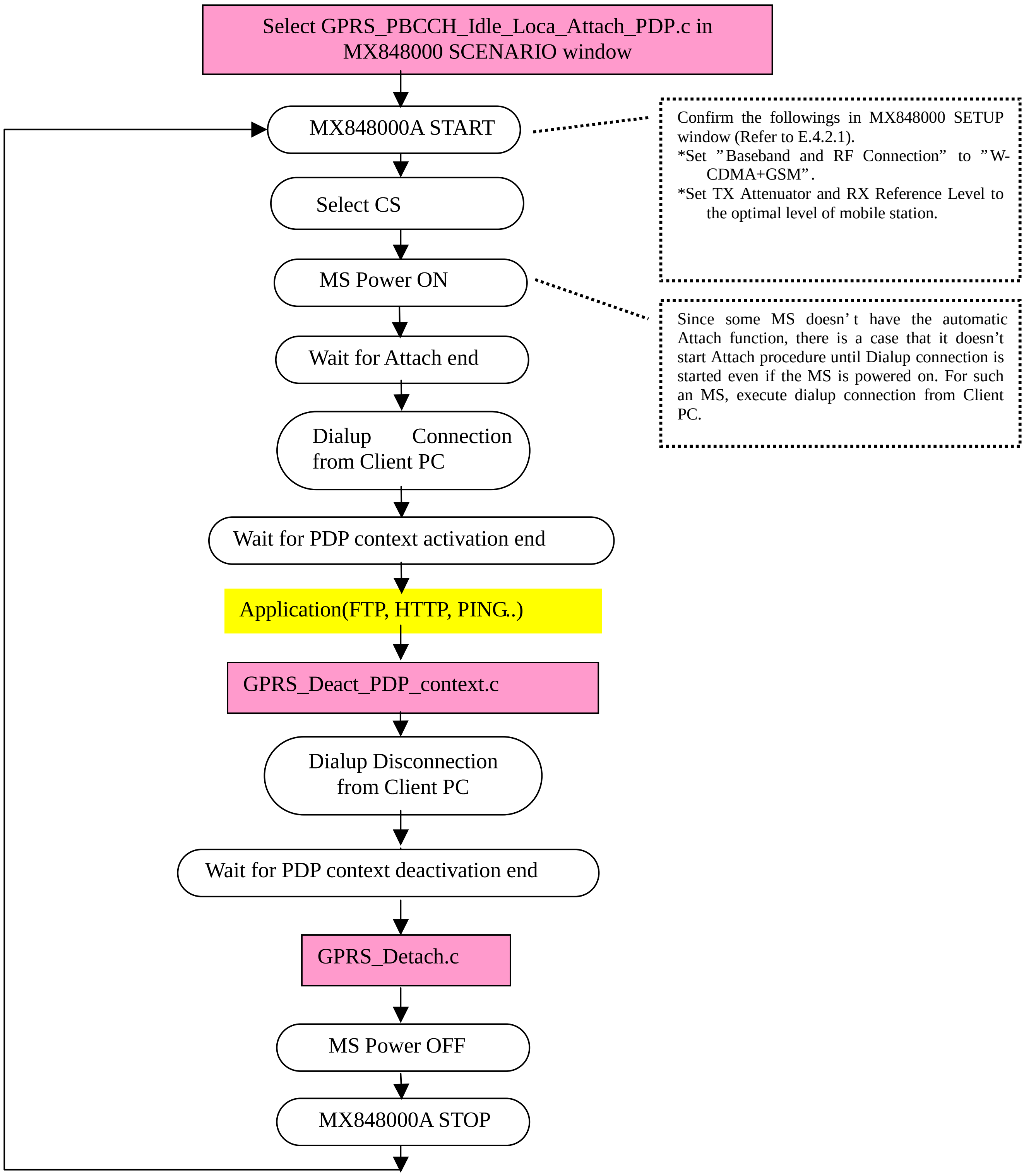
E.5.2.2.1 Data communication test using BCCH/CCCH.

A multislot test is possible with this scenario.



E.5.2.2.2 Data communication test which used PBCCH/PCCCH.

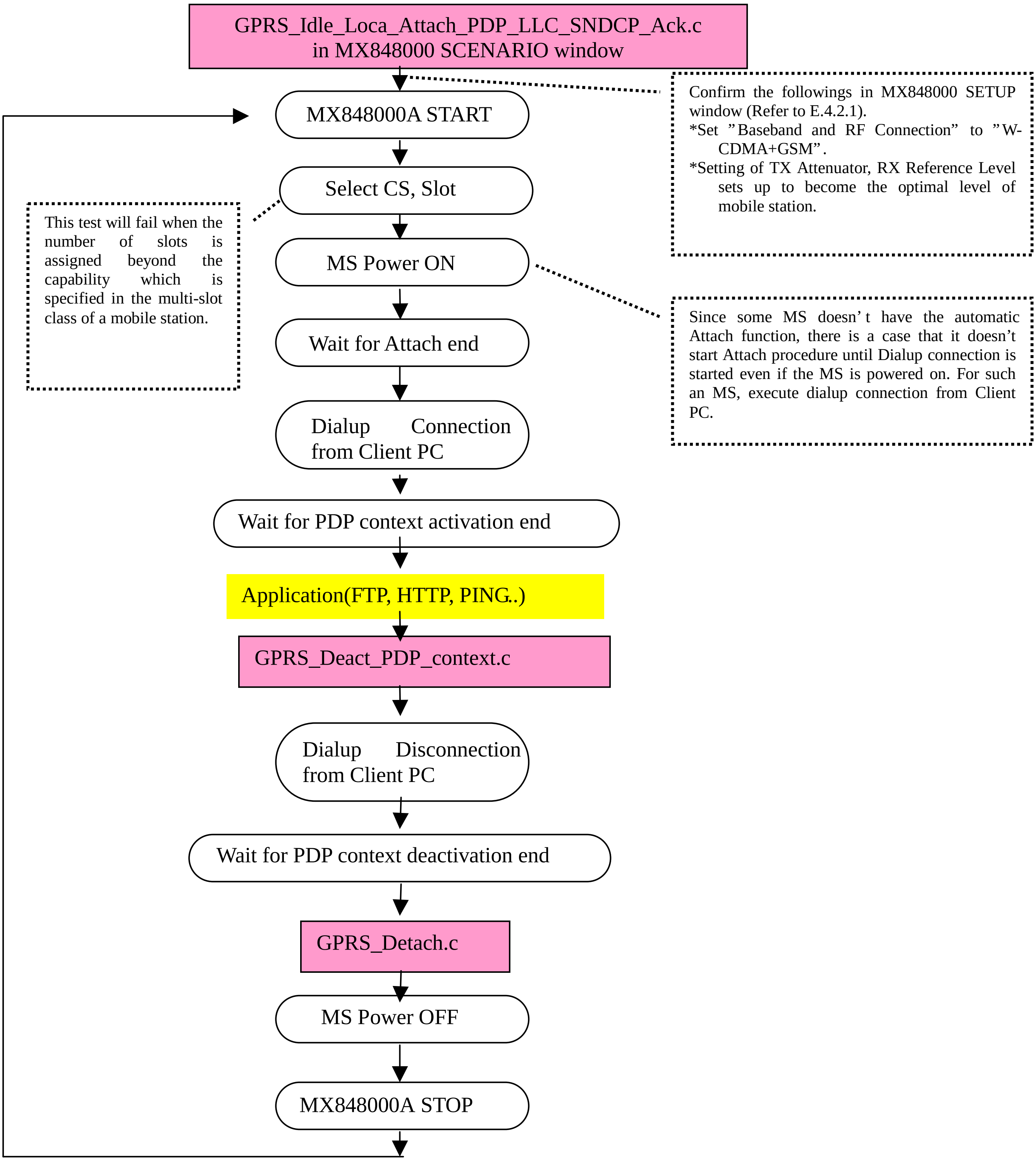
Execute scenarios by the following procedure.



E.5.2.2.3 LLC/SNDCP Ack mode test

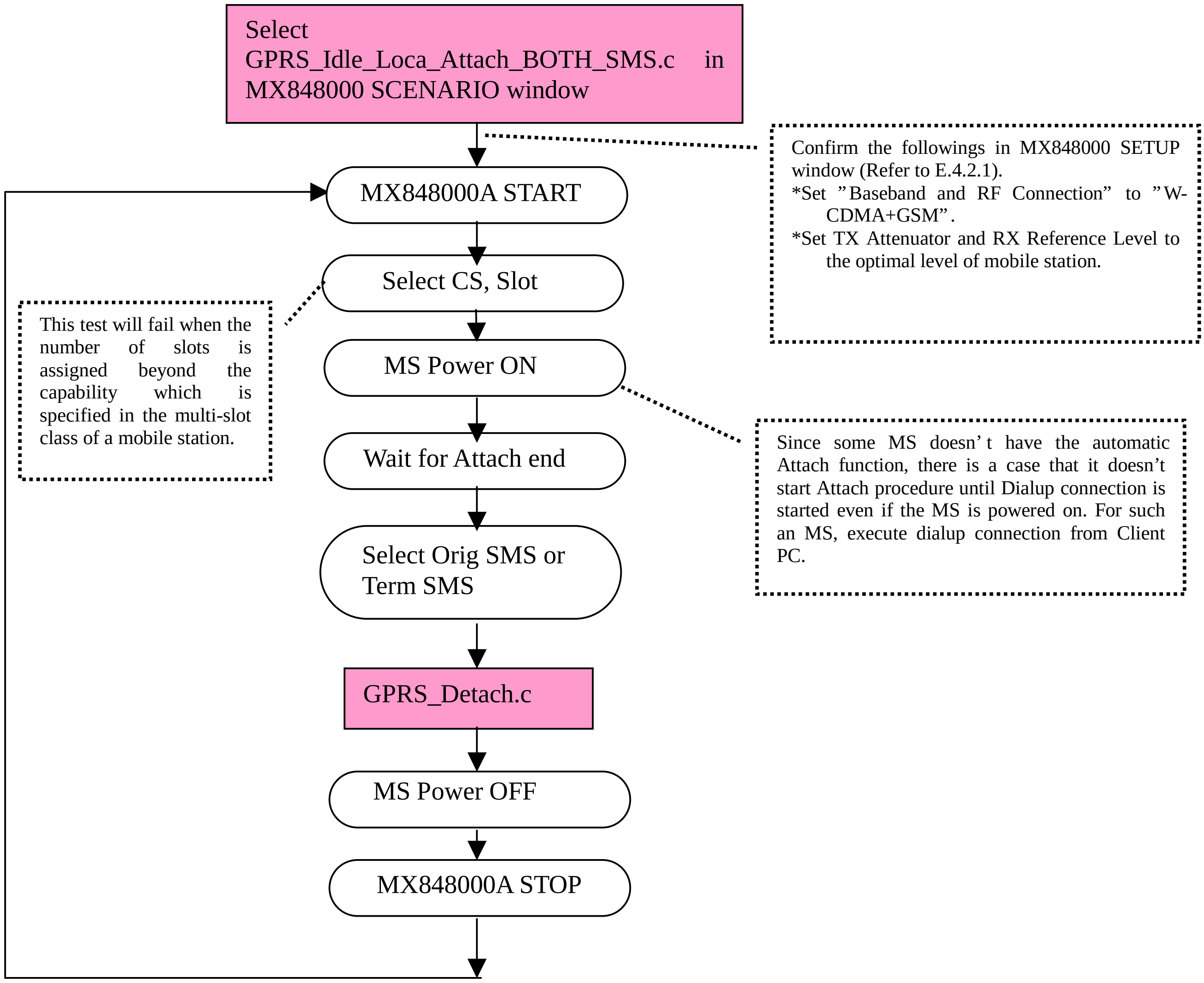
Execute the scenarios by the following procedure.

(Note) In the case of the mobile station which doesn’t correspond to LLC/SNDCP Ack mode, this scenario doesn’t work.



E.5.2.2.4 GPRS SMS test

GPRS SMS test is possible with this scenario



E.5.3 GSM CSD test

This section explains about GSM CSD /HSCSD test.

(Note) This test requires support service contract, ISDN Option and GSM CSD software option. You can find if ISDN Option is configured to check the 10th card from the left. Refer to the following how to confirm that support service contract and the GSM CSD option are installed in your Signalling Tester.

How to confirm support service contract,

Confirm that the version you are using is include in the option names that are shown in the System Window in the Control Software when Information Read button is clicked.

How to confirm the GSM CSD Option,

Confirm that the string "CSD" is appeared in the System Window in the Control Software when Information Read button is clicked.

(Note) Please contact MD8480A-G-support@zy.anritsu.co.jp if you need the detailed information regarding the GSM CSD option installation.

E.5.3.1 Connection of equipment

Connect the required equipment as figure E5-2.

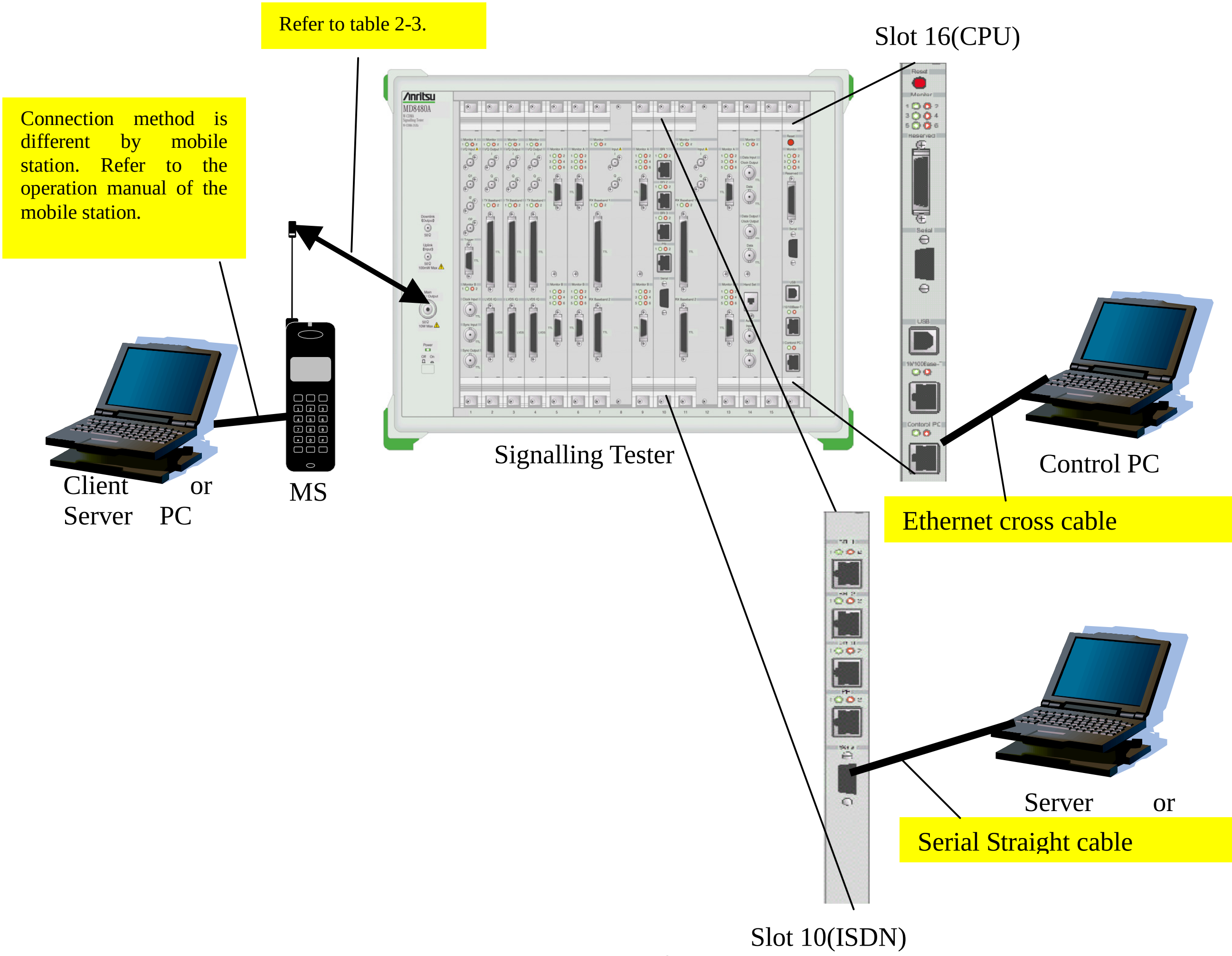


Figure E5-2 Equipment connection for GSM CSD test

E.5.3.2 Windows2000 Serial Connection Setting Procedure for GSM CSD test

This chapter describes the setting procedure for Windows 2000 PPP server connected to a Signalling Tester by a serial cable. There are too many setting patterns in Windows 2000 for a serial cable connection, so this document describes only one setting example which we have confirmed in our experiment.

For Widows 2000 Professional, follow the (1), (2), (3) and (4) in sequence. For Widows 2000 Server, refer to section (1), (2), (3), (4), and (5) in sequence. After the setting is finished, connect Windows2000 and MD8480A/B with a straight type serial cable before executing a GSM CSD test scenario.

(1) Settings for Phone and Modem Options

(01) Click ***Start***.

(02) Point to ***Settings***.

(03) Click ***Control Panel***.

(04) Double-click ***Phone and Modem Options***.

(05) Click ***Add*** in ***Modem*** tab in ***Phone and Modem Options*** window.

(06) Click ***Next*** for ***Do you want Windows to detect your Modem?***.

(07) Click ***Next*** for ***Windows was unable to detect any modems***.

(08) Select ***Standard Modem Types*** in ***Manufacturers.***

(09) Click ***Next*** after point to ***Communications cable between two computers***.

(10) Click ***Selected ports***, and then select ***COM1*** in ***Select the port(s) you want to install the modem on,*** and then click ***Next***.

(11) Click ***Finish*** for ***Modem installation is finished!***.

(12) Click ***OK*** for ***Phone and Modem Options***.

(2) Settings for Modem

(01) Click ***Start***.

(02) Point to ***Settings***.

(03) Click ***Control Panel***.

(04) Double-click ***System***.

(05) Select ***Hardware*** tab in ***System Properties*** window, and then click ***Device Manager***.

(06) Double-click ***Communications Cable between two computers*** in ***Modems.***

(07) Select ***115200*** for ***Maximum Port Speed*** in ***Modem*** tab in ***Communications cable between two computers Property*** window.

(08) Click ***OK***.

(3) Settings for Port

(01) Click ***Start***.

(02) Point to ***Settings***.

(03) Click ***Control Panel***.

(04) Double-click ***System***.

(05) Select ***Hardware*** tab in ***System Properties*** window, and then click ***Device Manager***.

(06) Double click ***Communications Port(COM1)*** in ***Ports(COM&LPT)***.

(07) Set as below for ***Port Settings*** tab in ***Communications Port(COM1) Properties*** window,

Bit per Second:	115200
Data bits:	8
Parity:	None
Stop bits:	1
Flow Control:	Hardware

(08) Click ***OK***

(4) Settings for Incoming Connections

(01) Click ***Start***.

(02) Point to ***Settings***.

(03) Click ***Control Panel***.

(04) Double-click ***Network and Dial-up Connections***.

(05) Double-click ***Make New Connection***.

(06) Click ***Next*** in ***Network Connection Wizard*** window.

(07) Click(Mark on) ***Accept incoming connections*** , and then click ***Next.***

(08) Click(Mark on) ***Communications cable between two computers(COM1)*** in ***Connection devices,***, and then click ***Next***

(09) Click(Mark on) ***Do not allow virtual private connections*** , and then click ***Next***.

(10) Click ***Add***.

(11) Set as below for ***New User*** window. and then click ***OK***.

User Name	TEST
Full Name	TEST
Password	TEST
Confirm Password	TEST

(Note) Set same user name and password as the setting for client PC dial-up.

(12) Confirm the check box is marked on for the user name set in (11), and then click ***Next***.

(13) Select(Mark on) ***Internet Protocol(TCP/IP)*** in ***Networking Components***, and then click ***Properties***.

(14) Click(Mark on) ***Specify TCP/IP addresses*** for ***TCP/IP address assignment*** in ***Incoming TCP/IP Properties*** window and set as below, and then click ***OK***.

From	1.4.0.1
To	1.4.0.100

(15) Click ***Next***.

(16) Click ***Finish***.

(5)Settings for Remote Access Server

(01) Click ***Start***.

(02) Point to ***Programs***, point to ***Administrative Tools***, and then click ***Routing and Remote Access***.

(03) Right click ***XXXXXXXX(Local)*** in ***Routing and Remote Access*** window, and then click ***Configure and Enable Routing and Remote Access***.

(Note) "XXXXXXXX (Local)" is your computer name.

(04) Click ***Next*** in ***Welcome to the Routing and Remote Access Server Setup Wizard*** window.

(05) Click(Mark on) ***Manual Configured Server***, and then click ***Next***.

(06) Click ***Finish***.

(07) The Wizard will ask you if you want to start the service. Click ***Yes*** to start the service.

(08) Double-click ***Remote Access Policy*** in ***Routing and Remote Access*** window, and confirm the message ***Allow access if dial-in permission is enabled*** is displayed.

E.5.3.3 Execution order of the sample scenarios

Compile the related scenarios in advance(refer to section 2.4.4 how to compile a scenario).

The following setups shall be completed before the test..
*”Serial cable PC-PC connection” has been set up to COM port in Server PC.
* Modem driver has been set up to COM port in Client PC.
* Network settings have been configured in Server and Client PC.
* RAS (Remote Access Service) has been set up in Server PC.
* Settings of dial up connection have been set up in Client PC.

Call Origination/ Termination Procedure

(1) Start up Control Software on Control PC.

(2) Click [SETUP] and select ”W-CDMA+GSM” from [Base Band and RF Unit Connection]. Set up the TX Attenuator and RX Reference Level to be the optimal level for the test environment (Refer to E.4.2.1).

(3) Click the [SCENARIO] button and select ”GSM_Idle_CSD.dll” from the file selection dialog. (Select” GSM_Idle_HSCSD.dll” when you test HSCSD.)

(4) Click the [START] button, then Signalling Tester will start. File selection dialog box will appear to select a next scenario after “GSM_Idle_CSD.dll” or ”GSM_Idle_HSCSD.dll” completes, .

(5) Turn the power on of the MS.

(6) Confirm that Antenna mark which shows level OK is displayed on the MS screen.

*If not confirm (6) and RACHs are transmitted from the MS repeatedly, execute the Registration(Location Updating) procedure below.

Registration(Location Updating) Procedure

This procedure is only needed when the MS was not registered to the NW yet.

(a) Turn the MS power off and Click [STOP].

(b) Execute the procedure (1) to (4).

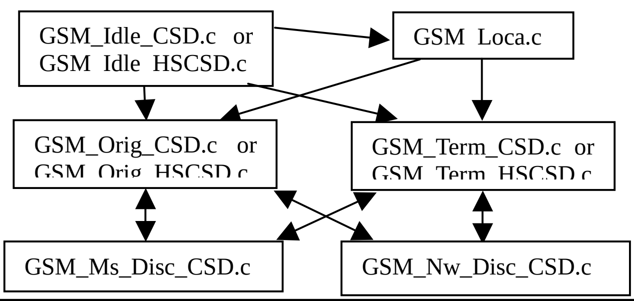
(c) Execute “GSM_Loca.dll” by selecting from the file selection dialog-box and wait for RACH after MS power on.

(d) Wait until “GSM_Loca.dll” is completed. Then file dialog is displayed to select a next scenario.

(e) Confirm that Antenna mark which shows level OK is displayed on the MS screen.

(7) Execute ”GSM_Orig_CSD.dll” (dial up from MS side Client) or ”GSM_Term_CSD.dll” (dial up from NW side client) by selecting from the scenario selection dialog. (Select ”GSM_Orig_HSCSD.dll” or ”GSM_Tirm_HSCSD.dll” when you test HSCSD.)

(8) Dialup from Client PC and login to RAS on the server PC.



E.5.3.4 Restriction of Timeslot settings

There are following restriction items,

- Only the following Timeslot combinations are effective.

Uplink 1Slot – Downlink 1Slot

Uplink 1Slot – Downlink 2Slots

Uplink 2Slots – Downlink 2Slots

Uplink 2Slots – Downlink 4Slots

(Note)This test will fail when the number of slots is assigned beyond the capability which is specified in the multi-slot class of a mobile station.

- Select "Timeslot2", "Timeslot4" or "Timeslot6" as Main Timeslot.
- Select a neighbor timeslot on main timeslot as an additional timeslot.

E.5.4 Handover test

E.5.4.1 Intra-RAT Handover

Handover test between GSM BTSs is possible.

The following three types of Handover are supported. Refer to 3GPP TS04.08 of 3.4.4 "Handover procedure" for details.

* Intra-cell Handover
* Inter-cell Handover (Synchronization)
* Inter-cell Handover (Non-Synchronization)

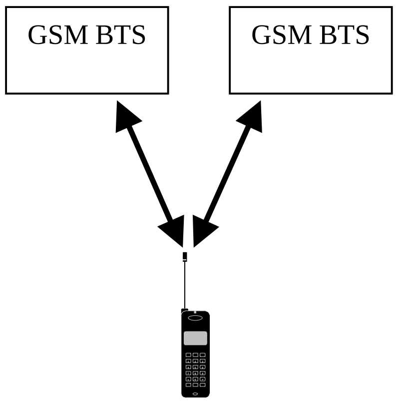


Figure E5-3 Intra-RAT Handover

E.5.4.2 Inter-RAT Handover

Inter-RAT handover between GSM and WCDMA BTS is possible.
To test the GSM-WCDMA Handover, edit layer3 message properly referring to GSM_InterSystem_HO.c (use this scenario as a sample).

Under a dial call, HO from GSM to WCDMA and HHO from WCDMA to GSM are supported.GSM side supports the following CODECs.

1) Speech EFR
   Support Voice Codec and Loopback.
2) Speech FR
   Support only Loopback.
3) Speech HR
   Support only Loopback.
4) Speech AMR (Fixed rate)
   Support Voice Codec and Loopback.

Please refer section 3.7.23 for Lossless Handover.

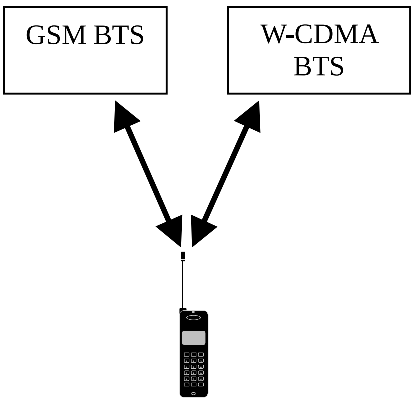


Figure E5-4 Inter-RAT Handover

E.5.5 Frequency Hopping Test

This section describes how to execute the GSM Frequency Hopping test. Regarding the definition of Frequency Hopping in the 3GPP specifications, refer to TS05.02 “Multiplexing and Multiple Access on the Radio Path”.

E.5.5.1 Frequency Hopping option

To execute the Frequency Hopping test, Signalling Tester needs to be installed the Frequency Hopping Option. Please confirm that your Signalling Tester has been installed the Frequency Hopping option by the following method.

How to confirm the Frequency Hopping Option,

Confirm that the string “GSM/GPRS F.H” is appeared in the System Window in the Control Software when Information Read button is clicked.

(Note) Please contact MD8480A-G-support@zy.anritsu.co.jp if you need the detailed information regarding the Frequency Hopping option installation.

E.5.5.2 Frequency Hopping sample scenario

The sample scenarios below for Frequency Hopping are attached to V5.11 or later Control Software.

C:¥Mx848000¥Scenario¥SRC¥Call_Sample(GSM)¥FH
GSM_Idle_FH.c
GSM_Voice(EFS)_FH.c

The specification for the sample scenarios are as follows,

- Executes Call origination, termination, disconnection with GSM-EFS voice call with Frequency Hopping(TCH).
- It is possible to select a frequency band from PGSM900, EGSM900, RGSM900, DCS1800, and PCS1900. The allocated ARFCNs of TCH for each band are as follows,

PGSM900: {1, 2, 3, 4, 5, 6, 7, 8, 117, 118, 119, 120, 121, 122, 123, 124,};
EGSM900: {0, 1, 2, 3, 121, 122, 123, 124, 975, 976, 977,978,1020,1021,1022,1023,};
RGSM900: {0, 1, 2, 3, 121, 122, 123, 124, 955, 956, 957, 958,1020,1021,1022,1023,};
DCS1800: {512, 513, 514, 515, 516, 517, 518, 519, 878, 879, 880, 881, 882, 883, 884, 885,};
PCS1900: {512, 513, 514, 515, 516, 517, 518, 519, 803, 804, 805, 806, 807, 808, 809, 810,};

E.5.5.3 Execution order of the sample scenarios for Frequency Hopping

This chapter describes how to execute the sample scenarios.
Scenarios have to be compiled before executing the procedure below,

1) Confirm the followings in MX848000 SETUP window (Refer to E.4.2.1).
   * Set "Baseband and RF Connection" to "W-CDMA+GSM".
   * Set TX Attenuator and RX Reference Level to the optimal level of mobile station.

2) Execute GSM_Idle_FH.dll

3) "BAND?" window will appear. Select the frequency band you test.
   (Note) Before selecting, confirm the frequency band capability of the MS.

4) Turn on the MS power. Wait about one minutes(the waiting time depends on a MS) and then confirm that the MS displays antenna mark on its panel. If the MS doesn't display the antenna mark and MS transmits RACHs repeatedly, click STOP and execute a location registration procedure(GSM_Idle -> GSM_Loca) in E.5.1.2. After the location registration procedure, re-execute 2) to 4) again.

5) Execute GSM_Voice(EFS)_FH.dll

6) "Connect?" window will appear. Click "Orig"(MS call origination) or "Term"(MS call termination).

7-1) If "Orig" is selected in 6).
   Make a dial call from MS. A voice call will be connected with Frequency Hopping.

7-2) If "Term" is selected in 6)
   A voice call will be terminated to MS. Connect the call by push the talk button of the MS. And then the call will be connected with Frequency Hopping.

8) "Disconnect?" window will appear. Click "MS Disc"(MS Disconnection) or "NW Disc"(NW Disconnection). If you clicked "MS Disc", push the disconnect button of the MS. After that, the call will be disconnected.

E.5.6 GPRS Harikiri (physical layer) test

This section describes GPRS Harikiri test. Refer to chapter 3.1 for the concept of Harikiri test.

E.5.6.1 Execution order of GPRS Harikiri test

1) Confirm the followings in MX848000 SETUP window (Refer to E.4.2.1).
   * Set "Baseband and RF Connection" to "W-CDMA+GSM".
   * Set TX Attenuator and RX Reference Level to the optimal level of mobile station.
2) Execute GPRS_Harikiri.dll

3) "Sending Pattern?" button will appear, then selects a value from followings,
   FIX DATA: fix data described in scenario
   USER DATA: input data from USER DATA connector
   PN9: PN9 sequence
   PN15: PN15 sequence

4) "Coding Scheme" button will appear, then selects coding scheme.

5) Downlink signal transmission starts and Uplink signal reception is ready.

   To change downlink/uplink frequency, modify frequency value in GsmRfchConfig in GPRS_Harikiri.c.

E.6 Scenario Library

This section summarizes the functions used for scenarios of the GSM/GPRS Option. Section E6.1 shows the function lists of scenario library. Section E6.2 shows feature details of each function. And Section E6.3 shows variables and the default values of each function. Refer to the sample scenarios which are attached to Control Software as examples of the Scenario Library. The sample scenarios are in the following folder in the PC which was installed the Control Software.

C:¥Mx848000¥Scenario¥SRC¥Call_Sample(GSM)
C:¥Mx848000¥Scenario¥SRC¥Attach_Sample(GPRS)

E.6.1 Scenario library function list

To control the GSM/GPRS Option, the following functions are prepared. Refer to the section which is described in this table for details.

*Functions newly prepared for the GSM/GPRS.

Table E6-1 New function for GSM/GPRS

Section number	Function name	Function outline
E.6.2.1	GsmReadFN ( )	Reads the current TDMA frame number.
E.6.2.2	GsmLochConfig()	Configures Loch setting.
E.6.2.3	GsmRfchConfig ( )	Executes a configuration for each RF channel.
E.6.2.4	GsmSetL1header()	Sets SACCH L1 Header.
E.6.2.5	GsmGetL1header()	Gets SACCH L1 Header.
E.6.2.6	GsmSetPagingGroup()	Sets up Paging Group for GPRS attached mode.
E.6.2.7	GsmDatalinkEstablish ( )	Establishes (configures) data link for each Loch.
E.6.2.8	GsmDatalinkRelease ( )	Releases Datalink for each Loch.
E.6.2.9	GsmDatalinkMsgSet()	Provides a Message setting to Datalink Layer.
E.6.2.10	GsmDatalinkSABMTest()	Configures behavier of L2 SABM failure case.
E.6.2.11	GsmRlcConfig()	Configures GPRS RLC/MAC Layer parameters.
E.6.2.12	GsmRlcRelease()	Releases GPRS RLC/MAC Layer.
E.6.2.13	GsmSndControl()	Transmits RLC/MAC Control Message from scenario.
E.6.2.14	GsmRcvControl()	Receives RLC/MAC Control Message in scenario.
E.6.2.15	GsmRrConfig()	Configures GRR Layer parameters.
E.6.2.16	GsmRrRelease()	Not support.
E.6.2.17	GsmRrInfoSet()	Sets DRX Paging information to GRR Layer.
E.6.2.18	GsmRrPagingTrig()	Transmits Paging Request Type1 or Packet Paging Request from GRR layer.
E.6.2.19	GsmRrMsgSet()	Sets Control Message to GRR Layer.
E.6.2.20	GsmTbfSTimeCalc()	Executes the calculation of TBF Starting Time.
E.6.2.21	GsmLlcConfig()	Configures LLC Layer parameters.
E.6.2.22	GsmLlcRelease()	Releases LLC Layer.
E.6.2.23	GsmLlcAssign()	Set TLLI parameter to layer of LLC.
E.6.2.24	GsmLlcReset()	Negotiates Parameter of Reset XID.
E.6.2.25	GsmLlcSuspend()	Change the state to suspension the layer of LLC, and stop sending data of downlink.
E.6.2.26	GsmSndcpConfig()	Configures SNDCP layer parameters
E.6.2.27	GsmSndcpRelease()	Releases SNDCP layer
E.6.2.28	GsmSndcpStopAssign()	Stops downlink transmission and Requires the indication of “DL

		Send N-PDU Number/UL Receive N-PDU Number" to SNDCP
E.6.2.29	GsmSndcpSequence ()	Notifies the sequence number when SNDCP layer re-opens downlink data transmission.
E.6.2.30	GsmSndcpModify()	Change the established Link of NSAPI and SAPI.
E.6.2.31	GsmSndcpRcvControl()	Receive notification that indicates all data for IntertRAT Handover has received from layer of SNCDP. Receive notification that indicates all retransmission data has sent to MS.
E.6.2.32	GsmSndcpFwd()	Transmit the data stored in SNDCP layer to PDCP layer.
E.6.2.33	GsmSndcpDeact()	The sequence number in which the downlink data transmission is restarted is notified.

*Extend library which add GSM/GPRS functionalities to WCDMA library

Table E6-2 Function for GSM/GPRS Option from W-CDMA software

Chapter number	Function name	Functional outline
E.6.2.34	SimulatorStart ( )	Starts the operation of MD8480B.
E.6.2.35	SndMessage ( )	Sends Message from scenario.
E.6.2.36	RcvMessage ( )	Receives Message at scenario.
E.6.2.37	BtsPower ( )	Increases or decreases the power of each RF channel
E.6.2.38	BtsAttenuator()	Sets the TX Attenuator and RX Reference Power.
E.6.2.39	CteConfig ( )	Configures TE layer parameters(relates TE type=Service to Logical Channel)
E.6.2.40	CteRelease ( )	Releases TE layer(deletes the relation between TE type and Logical Channel.)
E.6.2.41	CteConnect ( )	Starts data transmission and reception for the assigned TE Type.
E.6.2.42	CteDisconnect ( )	Stops data transmission and reception for the assigned TE Type.
E.6.2.43	CteSuspend ( )	Stops data transmission and reception for the assigned TE Type temporarily and received data is preserved.
E.6.2.44	CteResume ( )	Restarts data transmission and reception for the assigned TE Type.

*Library of W-CDMA which is described in Chapter A can be used.

E.6.2 Details of each function

E.6.2.1 GsmReadFN ()

Function name	INT GsmReadFN (INT Timeout)			
Feature summary	Reads the current TDMA frame number.			
	Type	Argument name	Description	I/O
Argument	INT	Timeout	Timeout value	Input
	Type	Function value	Description	
Function value	INT	> 0 < 0	TDMA frame number value Abnormal termination	
Feature details				

1) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’t complete by this time. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	There is some delay in the frame number by software processing(the delay is not constant).

E.6.2.2 GsmLochConfig ()

Function name	INT GsmLochConfig(GSM_MPH_BTSCONFIG_PAR *LochParam, INT Timeout)			
Feature summary	Configures Loch setting.			
	Type	Argument name	Description	I/O
Argument	GSM_MPH_BTSCONFIG_PAR * INT	LochParam Timeout	Loch setting parameters Timeout value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Members for LochParam

1-1) TransmitMode

Select the RF channel which Logical channel GSMD_SDCCH, GSMD_SACCH are transmitted when both RF CCH and RF TCH can carry them.

GSM_TXMODE_EITHER [default]	transmitted by the channel combination which has the earlier transmitting timing for a data at the PHY
GSM_TXMODE_TCH	transmitted only by the channel combination on RF TCH
GSM_TXMODE_CCH	transmitted only by the channel combination on RF CCH

1-2) NewChannelType

Decide whether the new Channel Types (GSM_SDCCH4, GSM_SDCCH8, GSM_SACCH4, GSM_SACCH8) can be used.

GSM_NEWCH_DISABLE [default]	not available
GSM_NEWCH_ENABLE	available

1-3) AccessBurstChannel

Decide channel type for the access burst.

GSM_AB_RACH [default]	GSMU_RACH
GSM_AB_PROPER	the proper Loch (GSMU_RACH, GSMU_SDCCH, GSMU_SDCCH4, GSMU_SDCCH8, GSMU_FACCH, GSMU_SACCH, GSMU_SACCH4, GSMU_SACCH8)

1-4) ShowupFN

For SDCCH, FACCH and SACCH, decide whether Frame Number which the Loch is sent on is showed on Trace Window.

GSM_FNTRACE_OFF [default]	OFF
GSM_FNTRACE_ON	ON

2) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this time. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.3 GsmRfchConfig ()

Function name	INT GsmRfchConfig(INT Rf_CH, GSM_RF_CONFIG_PAR*RfParam, INT ActTime, INT Timeout)			
Feature summary	Executes a configuration for each RF channel.			
	Type	Argument name	Description	I/O
Argument	INT	Rf_CH	Radio Frequency Channel	Input
	GSM_RF_CONFIG_PAR*	RfParam	Radio Frequency Channel parameter	Input
	INT	ActTime	Activation Time	Input
	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0	Normal termination	
		≠ 0	Abnormal termination (Timeout)	
Feature details				

1) Rf_CH

Select Rf_CH from the following values

```
GSMRF_D_CCH     /* setting downlink of CCH    */
GSMRF_D_TCH     /* setting downlink of TCH    */
GSMRF_D_PCCH    /* setting downlink of PCCH */
GSMRF_D_PTCH    /* setting downlink of PTCH */
GSMRF_U_CCH     /* setting uplink of CCH      */
GSMRF_U_TCH     /* setting uplink of TCH      */
GSMRF_U_PCCH    /* setting uplink of PCCH     */
GSMRF_U_PTCH    /* setting uplink of PTCH     */
```

2) Members for RfParam

Note) In case of stopping RF channel, specify (GSM_RF_CONFIG_PAR*)0 to RfParam.

2-1) TxPower

Set the transmission power for TxPower within the following ranges.

Setting value 1 dBm Unit, (-99dBm to 0dBm)

2-2) Frequency

Set the frequency within a range below for Frequency

Setting value 200000 Hz Unit, (450600000Hz – 1989800000Hz)

When Frequency Hopping is ON(refer to the following 2-13),this setting value is ignored. Refer to note item of frequency setting in Chapter E.6.7.

2-3) Timeslot

Select the timeslot configuration from the following values. Refer to Chapter E.6.7 also.

```
TIMESLOT0                 /* when the RF channel is GSMRF_D_CCH or GSMRF_U_CCH */
TIMESLOT1                 /* when the RF channel is GSMRF_D_PCCH or GSMRF_U_PCCH */
TIMESLOT2 to TIMESLOT6    /* when the RF channel is GSMRF_D_TCH or GSMRF_U_TCH*/

//When the RF channel is GSMRF_D_PTCH or GSMRF_U_PTCH
  //For Single slot GPRS PTCH,
    TIMESLOT2 to TIMESLOT6    (Note) TIMESLOT2 cannot be used when PCCH is activated.
  //For Multislot GPRS PTCHs,
    TIMESLOT3|TIMESLOT4                            /* Uplink 1slot, Downlink 2slots */
    TIMESLOT3|TIMESLOT4|TIMESLOT5                  /* Uplink 1slot, Downlink 3slots */
    TIMESLOT3|TIMESLOT4|TIMESLOT5|TIMESLOT6        /* Uplink 1slot, Downlink 4slots */
    TIMESLOT3|TIMESLOT4                            /* Uplink 2slot, Downlink 1slots */
    TIMESLOT3|TIMESLOT4                            /* Uplink 2slot, Downlink 2slots */
    TIMESLOT3|TIMESLOT4|TIMESLOT5                  /* Uplink 2slot, Downlink 3slots */
```

2-4) LochCombination

Select LochCombination from the following values,

```
COMB_D_CCCH          /* Valid in case of GSMRF_D_CCH */
COMB_D_SDCCH4        /* Valid in case of GSMRF_D_CCH */
COMB_D_SDCCH8        /* Valid in case of GSMRF_D_TCH */
COMB_D_TCHF          /* Valid in case of GSMRF_D_TCH */
COMB_D_TCHH          /* Valid in case of GSMRF_D_TCH */
COMB_D_PCCCH         /* Valid in case of GSMRF_D_PCCH */
COMB_D_PDTCH         /* Valid in case of GSMRF_D_PTCH */
COMB_D_PCCCH_PDTCH   /* Valid in case of GSMRF_D_PCCH (for Combination11) */
COMB_D_GPRSTEST      /* Valid in case of GSMRF_D_PTCH(for Harikiri test) */
COMB_U_CCCH          /* Valid in case of GSMRF_U_CCH */
COMB_U_SDCCH4        /* Valid in case of GSMRF_U_CCH */
COMB_U_SDCCH8        /* Valid in case of GSMRF_U_TCH */
COMB_U_TCHF          /* Valid in case of GSMRF_U_TCH */
COMB_U_TCHH          /* Valid in case of GSMRF_U_TCH */
COMB_U_PCCCH         /* Valid in case of GSMRF_U_PCCH */
COMB_U_PDTCH         /* Valid in case of GSMRF_U_PTCH */
COMB_U_PCCCH_PDTCH   /* Valid in case of GSMRF_U_PCCH(for Combination11) */
COMB_U_GPRSTEST      /* Valid in case of GSMRF_U_PTCH(for Harikiri test) */
```

2-5) SubChNo

Set as below, when you select COMB_D_SDCCH4, COMB_D_SDCCH8, COMB_D_TCHH, COMB_U_SDCCH4, COMB_U_SDCCH8, or COMB_U_TCHH for LochCombination.

```
Sub-channel Number[0 to 3] for SDCCH/4 and SACCH/C4    /* When COMB_D_SDCCH4 or COMB_U_SDCCH 4 */
Sub-channel Number[0 to 7] for SDCCH/8 and SACCH/C8    /* When COMB_D_SDCCH8 or COMB_U_SDCCH8*/
Sub-channel Number[0 to 1] for TCH/H, FACCH/H, SACCH/TH
                                                       /* When COMB_D_SDCCH8 or COMB_U_SDCCH8*/
```

2-6) TSC

Set the TSC value from 0 to 7.

2-7) BSIC

Set the BSIC value from 0 to 63 when Rf_CH is GSM RF_D_CCH or GSM RF_U_CCH. Set same value for the both Rf_CHs.

2-8) BS_AG_BLKS_RES, BS_PA_MFRMS

When Rf_CH is GSMRF_D_CCH, set the following parameters,

BS_AG_BLKS_RES	[0 to 7]
BS_PA_MFRMS	[2 to 9]

2-9) BS_PBCCH_BLKS, BS_PAG_BLKS_RES, PSI1_REPEAT_PERIOD, PSI_COUNT_HR, PSI_COUNT_LR

When Rf_CH is GSMRF_D_PCCH, set the following parameters,

BS_PBCCH_BLKS	[1 to 4]
BS_PAG_BLKS_RES	[0 to 11]
PSI1_REPEAT_PERIOD	[1 to 16]
PSI_COUNT_HR	[0 to 16]
PSI_COUNT_LR	[0 to 63]

2-10) CodingType

Select CodingType from the following values,

COD_EFS	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_FS	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_12_2	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_10_2	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_7_95	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_7_4	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_6_7	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_5_9	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_5_15	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_AFS_4_75	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_CSD_F14_4	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_CSD_F9_6	/* Valid in case of COMB_D_TCHF or COMB_U_TCHF */
COD_HS	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_AHS_7_95	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_AHS_7_4	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_AHS_6_7	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_AHS_5_9	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_AHS_5_15	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_AHS_4_75	/* Valid in case of COMB_D_TCHH or COMB_U_TCHH */
COD_PACKET	/* Valid in case of COMB_D_PTCH, COMB_U_PTCH, COMB_D_PCCH or COMB_U_PCCH */
0	/* The other cases */

2-11) TimingAdvance

Set the following values for Downlink or Uplink, for member TimingAdvance of RfParam.

For Downlink channels:	Ordered Timing Advance inserted to the L1 Header part on SACCH.
For Uplink channels:	Timing Advance for which Signalling Tester is actually waiting.
(Note)	Set the same values for Downlink and Uplink in the normal cases. In case that more than 1 RF channels are set (Ex. CCH and TCH), the one set last will be valid.

2-12) MSPowerLevel

The following value can be set up for MSPowerLevel of RfParam

When Downlink SACCH is included in LochCombination, Ordered MS Power Level is entered L1 Header part of SACCH.

2-13) Hopping

"Hopping" can set the parameters for Frequency Hopping. You can set these parameters when Rf_CH is GSMRF_D_TCH or GSMRF_U_TCH. Set these parameters same value for the both Rf_CHs. When Flag is set to HOPPING_OFF, they are ignored after it.

(Note) When your Signalling Tester doesn't have the Frequency Hopping option, these parameters are all ignored. For the information of Frequency Hopping option , refer to E.3.1.

(Note) Set GsmWcdmaOffset 0, when you test Frequency Hopping.

"Hopping" has the members below,

2-13-1) Flag

Select from the following values,

HOPPING_OFF /* When without Frequency Hopping */
HOPPING_ON /* When with Frequency Hopping */

2-13-2) FreqBand

Select from the following values,

BAND_PGSM900 /* P-GSM900 */
BAND_EGSM900 /* E-GSM900 */
BAND_RGSM900 /* R-GSM900 */
BAND_DCS1800 /* DCS1800 */
BAND_PCS1900 /* PCS1900 */
BAND_GSM450 /* GSM450 */
BAND_GSM480 /* GSM480 */
BAND_GSM850 /* GSM850 */

2-13-3) MAIO

Set the value from 0 to NumFreqList -1.

2-13-4) HSN

Set the value from 0 to 63.

2-13-5) NumFreqList

Set the number of allocated ARFCNs from 1 to 64.

2-13-6) ARFCN[64]

Set the ARFCN values from the lower ARFCN. The variables are valid from ARFCN[0] to ARFCN[NumFreqList -1]

(Note) This array have to be set from the lower ARFCN. Please be aware that the lower frequency has higher ARFCN value

2-14) Hscsd

"Hscsd" can set the parameters for HSCSD. You can set these parameters when Rf_CH is GSMRF_D_TCH or GSMRF_U_TCH. "Hscsd" has the members below,

2-14-1) MainTimeslot

Select from the following values,

TIMESLOT2 to TIMESLOT6

ex. When HSCSD is constructed by Timeslot 2, 3 and 4, and Main Timeslot is Timeslot 3, set "Timeslot" to TIMESLOT2 | TIMESLOT3 | TIMESLOT4 and "MainTimeslot" to TIMESLOT3.

2-15)Smscb

"Smscb" can set the parameters for SMSCB. You can set these parameters when LochCombination is COMB_D_SDCCH4 or COMB_D_SDCCH8. "Smscb" has the members below,

2-15-1)Flag

Select from the following values,

SMSCB_OFF /* No SMSCB(Not send CBCH) */
SMSCB_ON /* SMSCB(send CBCH) */

3)ActTime

Input the TDMA frame No value to ActTime for enabling the settings. The following values can be specified.

ACTIVATE_NOW /* when FN is not specified */
0 to 2715647 /* when FN is specified */

(Note) When setting up Uplink RF, about 1 frame would shift forward or backward from a designated FN due to hardware limitation.

4) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.4 GsmSetL1header ()

Function name	INT GsmSetL1header (UCHAR MSPowerLevel, UCHAR TimingAdvance, INT Timeout)			
Feature summary	Sets SACCH L1 Header.			
	Type	Argument name	Description	I/O
Argument	UCHAR UCHAR INT	MSPowerLevel TimingAdvance Timeout	Specified MS power level Specified Timing Advance Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	

Feature details

1) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.5 GsmGetL1header ()

Function name	INT GsmGetL1header(UCHAR *MSPowerLevel, UCHAR *TimingAdvance, INT Timeout)			
Feature summary	Gets SACCH L1 Header.			
	Type	Argument name	Description	I/O
Argument	UCHAR * UCHAR * INT	MSPowerLevel TimingAdvance Timeout	Actual MS Power Level Actual Timing Advance Time-out value	Output Output Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	

Feature details

1) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.6 GsmSetPagingGroup ()

Function name	INT GsmSetPagingGroup(USHORT *PG_Table, INT Timeout)			
Feature summary	Sets up Paging Group for GPRS attached mode.			
	Type	Argument name	Description	I/O
Argument	USHORT* INT	PG_Table Timeout	Paging Group Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

Use this function when many Paging Groups is set up for MS.

1)PG_Table

For PG_Table[x], Paging block index in multiframe of the x-th can be set up as follows.

(x = 0 - 63 for PCH, PPCH)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Bit *i* indicates Paging block index *i*.: <- don' t care ->

1 --- Transmit paging.

0 --- Not to transmit paging.

(Note) When Paging Group which is set by this function is used, it' s necessary to set next common variables as follows.

GsmPagingGroup = GSM_PG_USE_PRESET;

GsmPagingBlockIndex = GSM_PG_USE_PRESET;

2) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.7 GsmDatalinkEstablish ()

Function name	INT GsmDatalinkEstablish (INT Loch, INT Sapi,GSM_MDL_CONFIG_PAR *DlParam, INT Direction, INT UpperLayer, INT Timeout)			
Feature summary	Establishes (configures) data link for each Loch			
	Type	Argument name	Description	I/O
Argument	INT INT GSM_MDL_CONFIG_PAR* INT INT INT	Loch Sapi DlParam Direction UpperLayer Timeout	Logical Channel SAPI Loch datalink parameter Direction for establishment Going up data transmission Layer Time-out value	Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	
Feature details				

This function is used only when T200 or N200 values are changed. Otherwise this function need not be used.

1) Loch

Select Loch from the following values. (Note: There is no distinction for downlink and uplink.)

GSM_SDCCH
GSM_FACCH
GSM_SACCH
GSM_BCCH
GSM_PCH
GSM_AGCH
GSM_RACH

When establishing datalink from Network side, the following values can be selected to send SABM. It is necessary to permit use of these values by GsmLochConfig().

GSM_SDCCH8
GSM_SDCCH4
GSM_SACCH8
GSM_SACCH4

2) Sapi

When Loch is FACCH, SDCCH, SDCCH8, SDCCH4, SACCH, SACCH8 or SACCH4, set GSM_SAPI0 or GSM_SAPI3 for Sapi.

3) DlParam

Set the following parameters to each member of DlParam. When T200 and N200 values are set to 0, these values return to default value.

3-1) Mode

GSM_DATALINK_ACK /* When multiple frame acknowledged operation is used */
GSM_DATALINK_UNACK /* When unacknowledged operation is used */
GSM_DATALINK_NOHEAD /* When data-through (no head) operation is used */

3-2) T200 Timer

Set T200 value (Valid only for GSM_DATALINK_ACK)

3-3) N200

Set N200 value (Valid only for GSM_DATALINK_ACK)

If 0 is specified to T200 and N200, it becomes initialization setting value.

4)Direction

Set the following value to Direction. (Valid only GSM_DATALINK_ACK)

GSM_DATALINK_FROM_NW /* Establish datalink from Network side */
GSM_DATALINK_FROM_UE /* Establish datalink from UE side. */

5) UpperLayer

UpperLayer is the specified layer which is gave the received data. GSML3 or GSMTE can be specified.

6) Common Variables

The following common variables are used to specify the detail of function. Set these common variables before executing GsmDatalinkEstablish( ).

6-1) When Network sends SABM to establish Datalink and Loch is GSM_SDCCH8 and GSM_SACCH8:

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH7]

6-2) When Network sends SABM to establish Datalink and Loch is GSM_SDCCH4 and GSM_SACCH4:

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH3]

6-3) When Network sends SABM to establish Datalink and Loch is GSM_FACCH and GSM_SACCH:

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH1]

Note: When GsmSndSubChNo is 0 (default), Sub-channel Number set by GsmRfchConfig() is valid.
Note: After executing GsmDatalinkEstablish(), GsmSndSubChNo is initialized by 0 automatically.

7) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.8 GsmDatalinkRelease ()

Function name	INT GsmDatalinkRelease (INT Loch, INT Sapi, INT Direction, INT Mode, INT Timeout)			
Feature summary	Releases Datalink for each Loch.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT INT	Loch Sapi Direction Mode Timeout	Logical Channel SAPI Direction of Release Release Mode Time-out value	Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	=0 ≠ 0	Normal termination Abnormal termination	

Feature details

This function is used only when Datalink is established with GSM_DATALINK_ACK.

1) Loch

Select Loch from the following values, (Note: There is no distinction for downlink and uplink.)

GSM_SACCH
GSM_SDCCH
GSM_FACCH

When releasing datalink from Network side, the following values can be selected to send DISC. It is necessary to permit use of these values by GsmLochConfig().

GSM_SDCCH8
GSM_SDCCH4
GSM_SACCH8
GSM_SACCH4

2) Sapi

Select Sapi from the following values,

GSM_SAPI0
GSM_SAPI3

3) Direction

Select Direction from the following values,

GSM_DATALINK_FROM_NW /* When Datalink is released from Network side */
GSM_DATALINK_FROM_UE /* When Datalink is released from UE side */

(Note) When GSM_DATALINK_REL_LOCALEND is specified to Mode,
set GSM_DATALINK_FROM_NW for Direction.

4) Mode

Select Mode from the following values,

GSM_DATALINK_REL_NORMAL /* When Datalink is released normally */
GSM_DATALINK_REL_LOCALEND /* When Datalink is released at Local End */

5) Common Variables

The following common variables are used to specify the detail of function. Set these common variables before executing GsmDatalinkRelease( ).

5-1) When Network sends DISC to release Datalink and Loch is GSM_SDCCH8 and GSM_SACCH8:

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH7]

5-2) When Network sends DISC to release Datalink and Loch is GSM_SDCCH4 and GSM_SACCH4:

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH3]

5-3) When Network sends DISC to release Datalink and Loch is GSM_FACCH and GSM_SACCH:

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH1]

Note: When GsmSndSubChNo is 0 (default), Sub-channel Number set by GsmRfchConfig() is valid.
Note: After executing GsmDatalinkRelease(), GsmSndSubChNo is initialized by 0 automatically.

6) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.9 GsmDatalinkMsgSet ()

Function name	INT GsmDatalinkMsgSet(GSM_DL_MSGSET_PAR *DlParam, CHAR *Message, INT Length, INT Timeout)			
Feature summary	Provides a Message setting to Datalink Layer.			
	Type	Argument name	Description	I/O
Argument	GSM_DL_MSGSET_PAR*	DlParam	Setup information	Input
	CHAR*	Message	Transmit data	Input
	INT	Length	The length of transmit data	Input
	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0	Normal termination	
		≠ 0	Abnormal termination	
Feature details				

1) DlParam

1-1) MsgType

The following value can be specified for MsgType in case of setting PHYSICAL INFORMATION and SMSCB Message.

GSM_PHYSICAL_INFORMATION /* set PHYSICAL INFORMATION */
GSM_SMSCB /* set SMSCB Message */

1-2) Channel

When MsgType is GSM_PHYSICAL_INFORMATION, select Channel from the following values,

GSMD_SDCCH
GSMD_FACCH

The following values can be selected when they are available by GsmLochConfig().

GSMD_SDCCH8
GSMD_SDCCH4

When MsgType is GSM_SMSCB, select Channel from the following values,

GSMD_CBCH

1-3) SAPI

When MsgType is GSM_PHYSICAL_INFORMATION, set to GSM_SAPI0 or GSM_SAPI3.
When MsgType is GSM_SMSCB, set to 0.

1-4) Opt1

When MsgType is GSM_PHYSICAL_INFORMATION, set the number of Access Burst which should be received by DL layer.
When MsgType is GSM_SMSCB, set CBCH Type and Message Type as Opt1 = CBCH Type | Message Type.

For CBCH Type, select from the following values,

GSM_SMSCB_BASIC_SDCCH8 /* basic CBCH on SDCCH/8 */
GSM_SMSCB_EXTENDED_SDCCH8 /* extended CBCH on SDCCH/8 */
GSM_SMSCB_BASIC_SDCCH4 /* basic CBCH on SDCCH/4 */
GSM_SMSCB_EXTENDED_SDCCH4 /* extended CBCH on SDCCH/4 */

For Message Type, select from the following values,

GSM_SMSCB_CBS_MESSAGE /* CBS Message */
GSM_SMSCB_SCHEDULE_MESSAGE /* Schedule Message */
GSM_SMSCB_NULL_MESSAGE /* Null Message */

1-5) Opt2

When MsgType is GSM_PHYSICAL_INFORMATION, set the repetition number (Ny1) and the interval (T3105) [frame] as Opt2 = Ny1 << 8 | T3105.

Ny1	repetition number	[0 to 255]
T3105 [frame]	repetition interval	[0 to 255]

When Opt2 is 0, no repetition is done and the message can be sent only once.
When MsgType is GSM_SMSCB, select the repetition number (count) and the interval [sec] as Opt2 = count << 8 | interval.

count	repetition number	[0 to 100]
interval [sec]	repetition interval	[1 to 60]

When Opt2 is 0, no repetition is done and the message can be sent only once.

2) Message

Set the Message content to send.

3) Length

Set the byte length of the Message.
For SMSCB, set Length to 88 bytes when CBS Message or Schedule Message is sent and 0 byte when Null Message.

4) Common Variables

The following common variables are used to specify the detail of function. Set these common variables before executing GsmDatalinkMsgSet( ).

4-1) Loch is GSM_SDCCH8 :

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH7]

4-2) Loch is GSM_SDCCH4 :

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH3]

4-3) Loch is GSM_FACCH :

GsmSndSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH1]

Note: When GsmSndSubChNo is 0 (default), Sub-channel Number set by GsmRfchConfig() is valid.
Note: After executing GsmDatalinkMsgSet(), GsmSndSubChNo is initialized by 0 automatically.

5) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.10 GsmDatalinkSABMTest ()

Function name	INT GsmDatalinkSABMTest(INT Loch, INT Sapi, INT Action, INT Type, UCHAR* Message, INT Timeout)			
Feature summary	Configures behavier of L2 SABM failure case.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT UCHAR* INT	Loch Sapi Action Type Message Timeout	Logical channel SAPI START / STOP Test type Transmit data Time-out value	Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	

Feature details

1) Loch
Select Loch from the following values,
GSM_SDCCH
GSM_FACCH
GSM_SACCH
2) Sapi
Set GSM_SAPI0 or GSM_SAPI3.
3) Action
Set SABMTEST_START or SABMTEST_STOP.

SABMTEST_START	Start Test
SABMTEST_STOP	Stop Test

4) Type
Select Test type from the following values,

GSM_DATALINK_SEND_UA	send UA after receiving SABM
GSM_DATALINK_SEND_NON	send no L2 frame after receiving SABM
GSM_DATALINK_SEND_DM	send DM after receiving SABM
GSM_DATALINK_SEND_MES	send Message after receiving SABM

5) Message
When Type is GSM_DATALINK_SEND_MES, set 23 octets L2 frame to be sent.
6) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.11 GsmRlcConfig ()

Function name	INT GsmRlcConfig(INT Loch, CRLC_CONFIG_LPARA* ConfigData, INT Timeout)			
Feature summary	Configures GPRS RLC/MAC Layer parameters.			
	Type	Argument name	Description	I/O
Argument	INT CRLC_CONFIG_LPARA* INT	Loch ConfigData Timeout	Logical Channel Layer parameter Time-out value	Input Input Input
	Type	Function value	Description	

Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination

Feature details

1) Loch

Set 0 (Loch is not used).

2) ConfigData

Configures the RLC layer parameters by the members below.

SP_BIT	Set to SP_ON_8_FB.
UL_DATA_SIZE	Select from CS_1/CS_2/CS_3/CS_4.
DL_DATA_SIZE	Select from CS_1/CS_2/CS_3/CS_4.
RLC_MODE	Set to ACK_MODE. (UNACK mode doesn' t operate now.)
P_SEND_TIM	Set to P_ACK_8_CT. Specify Uplink ACK/NACK transmission periodic setup.
SEQ_MODE	Set to SEQMODE_AUTO.
SLOT_VALUE	Select from the following value. UL1DL1 /* Uplink 1 slot, Downlink 1 slot */ UL1DL2 /* Uplink 1 slot, Downlink 2 slot */ UL1DL3 /* Uplink 1 slot, Downlink 3 slot */ UL1DL4 /* Uplink 1 slot, Downlink 4 slot */ UL2DL1 /* Uplink 2 slot, Downlink 1 slot */ UL2DL2 /* Uplink 2 slot, Downlink 2 slot */ UL2DL3 /* Uplink 2 slot, Downlink 3 slot */ (Note) It' s necessary to set to the same values as GsmRfChConfig(). Refer to Chapter E.6.2.2 also.
USF_VALUE	Set USF_VALUE(0 to 7).
T3169	Set to 541. (It' s set about 5 Sec.)
T3191	Set to 541. (It' s set about 5 Sec.)
T3193	Set to 163. (It' s set about 1.5 Sec)
T3195	Set to 541. (It' s set about 5 Sec)
N3101	Set to 32.
N3103	Set to 2.
N3105	Set to 3. (Signalling Tester adds 3 more, so actual N3105 value is 6)

(Note) The values except above have not been confirmed the operation.

3) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout doesnot occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.12 GsmRlcRelease ()

Function name	INT GsmRlcRelease (INT Timeout)			
Feature summary	Releases GPRS RLC/MAC Layer.			
	Type	Argument name	Description	I/O
Argument	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.13 GsmSndControl ()

Function name	INT GsmSndControl (INT Loch , INT MsgType, INT TbfDirection, CGRR_MSGSET_DATA* MsgData, INT Timeout)			
Feature summary	Transmits RLC/MAC Control Message from scenario.			
	Type	Argument name	Description	I/O
Argument	INT INT INT CGRR_MSGSET_DATA* INT	Loch MsgType TbfDirection MsgData Timeout	Logical Channel Classification of message TBF Direction Message Data Time-out value	Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

(Note) In case of transmitting RLC/MAC Control message from scenario by this library, even if status of RLC/MAC layer is Idle, TBF Establishment isn’ t established automatically.

1) Loch
Select from the following value for Loch
GSMD_PACCH
GSMD_PPCH
GSMD_PAGCH

2) MsgType
Select from the following values for MsgType.
DL_ASN /* When Packet Downlink Assignment message is sent. */
UL_ASN /* When Packet Uplink Assignment message is sent.*/
TBF_UL_REL /* When Packet TBF Release message (Uplink) is sent. */
TBF_DL_REL /* When Packet TBF Release message (Downlink)is sent. */
DL_CTRLMSG_GEN /* In case of transmitting except 4 message above. */

3) TbfDirection
When Loch is GSMD_PACCH , direction of the TBF to be transmitted can be chosen. Select from the following values for TbfDirection. When other value is selected ,this value is ignored.
UL_TBFD /* When transmitting message in UL TBF. */
DL_TBFD /* When transmitting message in DL TBF. */
ANY_TBFD /* When transmitting message in UL TBF or DL TBF. */

4) MsgData
Set the pointer of message data for MsgData.

5) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.14 GsmRcvControl ()

Function name	INT GsmRcvControl (INT *Loch, INT *RcvFN, CHAR *Message, INT Timeout)			
Feature summary	Receives RLC/MAC Control Message in scenario.			
	Type	Argument name	Description	I/O

Argument	INT* INT* CHAR* INT	Loch RcvFN Message Timeout	Logical Channel Receive Frame Number Message Data Time-out value	Output Output Output Input
	Type	Function value	Description	
Function value	INT	> 0 < 0	Data length Abnormal termination (Timeout)	

Feature details

1) Loch
Logical Channel (PRACH or PACCH) returns when RLC/MAC Control Message is received by Scenario.
2) RcvFN
Frame Number returns when RLC/MAC Control Message is received by Scenario.
3) Message
Received message of RLC/MAC Control Message is returned.
4) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.15 GsmRrConfig ()

Function name	INT GsmRrConfig (CGRR_CONFIG_LPARA *ConfigData, INT Timeout)			
Feature summary	Configures GRR Layer parameters.			
	Type	Argument name	Description	I/O
Argument	CGRR_CONFIG_LPARA* INT	ConfigData Timeout	GRR Layer config Parameter Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) ConfigData
Set the members for ConfigData as follows. (CGRR_CONFIG_LPARA structure)
1-1) SEQ_MODE
Set to SEQMODE_AUTO
1-2) MSG_TYPE
Select from the following values
MSG_NORMAL /* When CCCH is used for Packet Assignment */
MSG_PACKET /* When PCCCH is used for Packet Assignment */
1-3) SLOT_TYPE
Select from the following values,
SINGLE_SLOT /* When single slot operation is applied */
MULTI_SLOT /* When multi-slot operation is applied */
2) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.17 GsmRrInfoSet ()

Function name	INT GsmRrInfoSet(CGRR_INFO_PAR* InfoData, INT Timeout)			
Feature summary	Sets DRX Paging information to GRR Layer. .			
	Type	Argument name	Description	I/O
Argument	CGRR_INFO_PAR * INT	InfoData Timeout	Information Data Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) InfoData

Set the members for InfoData as below (CGRR_INFO_PAR structure).

1-1)DRX_MODE

Select the DRX mode from the following values.

NON_DRX

DRX

1-2) PAG_GROUP

Set to acquired the GsmPagingGroup.

1-3) PAG_INDEX

Set to acquired the GsmPagingBlockIndex.

2)Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	PAGING_GROUP (GsmPagingGroup) and Paging_group_index (GsmPagingBlockIndex) are defiend in 3GPP. They are calculated by GsmCalcPagingClock() function in sample scenarios.

E.6.2.18 GsmRrPagingTrig ()

Function name	INT GsmRrPagingTrig(INT Timeout)			
Feature summary	Transmits Paging Request Type1 or Packet Paging Request from GRR layer			
	Type	Argument name	Description	I/O
Argument	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

This function is used when Paging Request Type1 or Packet Paging Request is transmitted from GRR layer.

1)Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Note)

(a) Before executing this function, it's necessary to set up Paging Request Type1 and Transmit data of Packet Paging Request to GRR layer by GsmRrMsgSet().

(b) When MsgType set by GsmRrConfig() is MSG_PACKET, GRR transmits Packet Paging to RLC/MAC layer on Request PPCH and when the MsgType is set to MSG_NORMAL , GRR transmits Paging Request Type1 to Datalink layer on PCH.

Remarks:	

E.6.2.19 GsmRrMsgSet ()

Function name	INT GsmRrMsgSet (INT MsgType, CGRR_MSGSET_DATA* MsgData, INT ReqRefPosi/Cycle_Idle, INT TlliPosi/Cycle_Trnsf, INT StrPosi, INT Timeout)			
Feature summary	Sets Control Message to GRR Layer.			
	Type	Argument name	Description	I/O
Argument	INT CGRR_MSGSET_DATA* INT INT INT INT	MsgType MsgData ReqRefPosi / Cycle_Idle TlliPosi / Cycle_Trnsf StrPosi Timeout	Message Type Control message data RequestReference setting position /Transmission cycle at Idle mode. TLLI setting position / Transmission cycle at Transfer mode. Starting Time setting position Time-out value	Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) MsgType

Select the MsgType from the following values.

P_UL_ACKNACK	/* Packet Uplink ACK/NACK message. */
P_UL_ASSIGN_R	/* Packet Uplink Assignment message for PAGCH in One Phase Assignment(It is used when MsgType of GsmRrConfig() is MSG_PACKET. */
P_UL_ASSIGN_D	/* Packet Uplink Assignment message for PACCH. (Pattern which TLLI is entered.) */
P_UL_ASSIGN_TWO_PH	/* Packet Uplink Assignment message for PAGCH in Two phase(single block) Assignment. It is used when MsgType of GsmRrConfig() is MSG_PACKET. */
P_DL_ASSIGN	/* Packet Downlink Assignment message. */
P_TBF_REL	/* Packet TBF Release message. */
P_DUMMY_CTRL	/* Packet Downlink Dummy Control Block message. */
UL_ASSIGN	/* Immediate Assignment(Packet Uplink Assignment) message. */
DL_ASSIGN	/* Immediate Assignment(Packet Downlink Assignment) message. */
UL_ASSIGN_TWO_PH	/* Immediate Assignment(Packet Uplink Assignment) message setting for two phase access */
PAGING_REQ_TYPE1	/* Paging Request Type1 message. */
P_PAGING_REQ	/* Packet Paging Request message. */
PSI_TYPE1	/* Packet System Information Type1 on PACCH. */
PSI_TYPE2	/* Packet System Information Type2 on PACCH. */
PSI_TYPE3	/* Packet System Information Type3 on PACCH. */
PSI_TYPE3BIS	/* Packet System Information Type3bis on PACCH. */
PSI_TYPE3TER	/* Packet System Information Type3ter on PACCH. */
PSI_TYPE3QUA	/* Packet System Information Type3quater on PACCH. */
PSI_TYPE4	/* Packet System Information Type4 on PACCH. */
PSI_TYPE5	/* Reserved */
PSI_TYPE6	/* Packet System Information Type6 on PACCH. */
PSI_TYPE7	/* Packet System Information Type7 on PACCH. */
PSI_TYPE8	/* Packet System Information Type8 on PACCH. */
PSI_TYPE13	/* Packet System Information Type13 on PCCCH or PACCH. */
PSI_TYPE14	/* Packet System Information Type14 on PACCH. */
PSI_TYPE15	/* Packet System Information Type15 on PACCH. */

2) MsgData

Set the message content to MsgData(23 octet length).

3) ReqRefPosi

The beginning position of Request Reference IE is set automatically. So set 0.

4) TlliPosi

The beginning position of TLLI IE is set automatically. So set 0.

5) StrPosi

The beginning position of TBF Starting Time IE is set automatically. So set 0.

6) Cycle_Idle, Cycle_Trnsf

When MsgType is PSI_TYPE1, PSI_TYPE2, PSI_TYPE3, PSI_TYPE3BIS, PSI_TYPE3TER, PSI_TYPE3QUA, PSI_TYPE4, PSI_TYPE5, PSI_TYPE6, PSI_TYPE7, PSI_TYPE8, PSI_TYPE13, PSI_TYPE14, PSI_TYPE15 set Cycle_Idle, Cycle_Trnsf. If it' s set to 0,transmission will be stopped.

Cycle_Idle	/* Transmission cycle at Idle mode [0, 1 to 255] (sec) */
Cycle_Trnsf	/* Transmission cycle at Transfer mode [0, 1 to 255] (sec) */

7) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.20 GsmTbfSTimeCalc ()

Function name	INT GsmTbfSTimeCalc (INT SndCtrl, INT Offset, USHORT *TbfSTime, INT Timeout)			
Feature summary	Executes the calculation of TBF Starting Time.			
	Type	Argument name	Description	I/O
Argument	INT INT USHORT* INT	SndCtrl Offset TbfSTime TIMEOUT	TBF Direction Offset value which wants to add. Calculation result of TBF Starting Time Time-out value	Input Input Output Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

This function can get the calculation result of Absolute Frame Number Encoding from "current frame number + offset" .The result is adapted to the head of Radio Block. So this result can be used for TBF starting time IE for the Packet Assignment messages.

1) SndCtrl

Select from the following values to SndCtrl.

GPRS_DL_ASN /* In case of calculating TBF Starting time which uses for Assignment message of downlink TBF. */

GPRS_UL_ASN /* In case of calculating TBF Starting time which uses for Assignment message of uplink TBF.*/

2) Offset

An offset value which wants to be added to TBF Starting time can be set. Unit is frame. The range is 0 to 65535 [frame]. Since it is necessary to set TBF Starting Time as frame of the top Radio Block , the value with which the minimum compensation was added to this value is added to TBF Starting Time as offset.

3) TbfSTime

Calculation result of TBF Starting Time is returned to TbfSTime by Absolute Frame Number Encoding(16bit).

4) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.21 GsmLlcConfig ()

Function name	INT GsmLlcConfig(CLL_CONFIG_LPARA *ConfigData, INT Timeout)			
Feature summary	Configures LLC Layer parameters.			
	Type	Argument name	Description	I/O
Argument	CLL_CONFIG_LPARA* INT	ConfigData Timeout	LLC layer parameter Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) ConfigData

Set the members of ConfigData(CLL_CONFIG_LPARA) as below,

Version	LLC version No (0 to 15)
TLLI	TLLI value (0x00000000 to 0x7FFFFFFF)
Protect	PM bit (0/1)
SEQ_MODE	Set to SEQMODE_AUTO
SAPI_INF[16]	Information element of each SAPI

Set the members of SAPI_INF[ ] as below,

T200	Re-transmission timer (1 to 4095)
N200	Re-transmission Counter (1 to 15)
N201_U	U, UI frame maximum information length (byte). Set as below, SAPI1 400 to 1520 SAPI2, 7, 8 270 to 1520 Other SAPIs 140 to 1520
N201_I	I frame maximum information length (140byte to 1520byte)
mD	Downlink I frame buffer size (0, 9 to 24320)
mU	Uplink I frame buffer size (0, 9 to 24320)
kD	Downlink Window size (1 to 255)
kU	Uplink Window size (1 to 255)

2) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.22 GsmLlcRelease ()

Function name	INT GsmLlcRelease (INT Timeout)			
Feature summary	Releases LLC Layer.			
	Type	Argument name	Description	I/O
Argument	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.23 GsmLlcAssign ()

Function name	INT GsmLlcAssign(GSM_CLL_ASSIGN_PAR *AssignData, INT Timeout)			
Feature summary	Set TLLI parameter to layer of LLC.			
	Type	Argument name	Description	I/O
Argument	GSM_CLL_ASSIGN_PAR * INT	AssignData Timeout	Parameter of Assignment at LLC Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) AssignData

Set the members of AssignData as below,

1-1)Tlli_old

Set the Tlli_old value in the range as below,

0 to 0xFFFFFFFF.

1-2) Tlli_new

Set the Tlli_old value in the range as below,

0 to 0xFFFFFFFF.

1-3) GEA

Set to NOT_SET_GEA.

2) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_CLL_ASSIGN_CNF from layer of LLC until timeout occur, after sent GSM_CLL_ASSIGN_REQ to layer of LLC.

E.6.2.24 GsmLlcReset ()

Function name	INT GsmLlcReset(INT Sapi, INT Timeout)			
Feature summary	Negotiates Parameter of Reset XID.			
	Type	Argument name	Description	I/O
Argument	INT INT	Sapi Timeout	SAPI between LLC and SNDCP Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Sapi

Set the SAPI that set with GsmSndcpConfig() .

2) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_CLL_RESET_CNF from layer of LLC until timeout occur, after sent GSM_CLL_RESET_REQ to layer of LLC.

E.6.2.25 GsmLlcSuspend ()

Function name	INT GsmLlcSuspend(INT Sapi, UCHAR *Tlli ,INT Timeout)			
Feature summary	Change the state to suspension the layer of LLC, and stop sending data of downlink.			
	Type	Argument name	Description	I/O
Argument	INT UCHAR * INT	Sapi Tlli Timeout	SAPI between LLC and SNDCP TLLI Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Sapi
Set the SAPI that set with GsmSndcpConfig() . 3, 5, 9, and 11 can be set.
2)Tlli
Set the Tlli value in the range as below,
0 to 0xFFFFFFFF.
3)Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_CLL_SUSPEND_CNF from layer of LLC until timeout occur, after sent GSM_CLL_SUSPEND_REQ to layer of LLC.

E.6.2.26 GsmSndcpConfig ()

Function name	INT GsmSndcpConfig(INT Nsapi, INT Sapi, GSM_CSN_CONFIG_PAR *CsnConfigPar, INT Timeout)			
Feature summary	Configures SNDCP layer parameters			
	Type	Argument name	Description	I/O
Argument	INT INT GSM_CSN_CONFIG_PAR* INT	Nsapi Sapi CsnConfigPar Timeout	NSAPI of SNDCP SAPI between LLC-SNDCP. SDNCP Layer parameter Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Nsapi
Specify NSAPI which becomes interface between SNDCP-TE in the range 5 to 15.
2) Sapi
Specify SAPI which becomes interface between LLC-SNDCP from 3,5,9,11. Set to same value of GsmLlcConfig(). (If it is not same value, data transfer is Abnormal termination.)
3) CsnConfigPar
Set the member values of CsnConfigPar(GSM_CSN_CONFIG_PAR structure) as below

Pcic	Set SN_PCIC_NOCOMP which means to let the SNDCP layer No compression.
Dcmp	Set SN_DCMP_NOCOMP which means to let the SNDCP layer No compression.
Npnsync	Please set Management of N-PDU Number as SN_NPNSYNC_OFF.
Layer	Set to GSMLLC
Frame	Select from the values below for downlink SN-PDU, GSM_LL_DATA_REQ /* for Acknowledged Mode */ GSM_LL_UNITDATA_REQ /* for Unacknowledged Mode */
SEQ_MODE	Set to SEQMODE_AUTO
TxWtime	Set to 0.

4) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	Multi PDP context cannot be used simultaneously.

E.6.2.27 GsmSndcpRelease ()

Function name	INT GsmSndcpRelease (INT Nsapi, INT Sapi, INT Timeout)			
Feature summary	Releases SNDCP layer			
	Type	Argument name	Description	I/O

Argument	INT INT INT	Nsapi Sapi Timeout	NSAPI of SNDCP SAPI between LLC-SNDCP Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Nsapi
Specify the NSAPI value which has been already set by GsmSndcpConfig( ).This value becomes the target to be released. It' s possible to specify from 5 to 15.
2) Sapi
Specify the SAPI value which has been already set by GsmSndcpConfig( ).This value becomes the target to release. It' s possible to specify 3, 5, 9, 11.
3) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.28 GsmSndcpStopAssign ()

Function name	INT GsmSndcpStopAssign (INT Nsapi, INT Sapi, UCHAR *DLSndNpduNo, UCHAR *ULRcvNpduNo, INT Timeout)			
Feature summary	Stops downlink transmission and Requires the indication of "DL Send N-PDU Number/UL Receive N-PDU Number" to SNDCP			
	Type	Argument name	Description	I/O
Argument	INT INT UCHAR* UCHAR* INT	Nsapi Sapi DLSndNpduNo ULRcvNpduNo Timeout	NSAPI of SNDCP SAPI between LLC-SNDCP DL Send N-PDU Number UL Receive N-PDU Number Time-out value	Input Input Output Output Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Nsapi
Specify NSAPI which is already set up by GsmSndcpConfig( ). It' s possible to specify from 5 to 15.
2) Sapi
Specify SAPI which is already set up by GsmSndcpConfig( ). It' s possible to specify 3, 5, 9, and 11.
3) DLSndNpduNo
N-PDU Number which transmitted by downlink from SNDCP Layer.
4) ULRcvNpduNo
N-PDU Number which received by uplink from SNDCP Layer.
5) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks: When this function is executed, SNSM_STOPASGN_IND is transmitted to SNDCP Layer. After that, SNSM_STOPASGN_RSP from SNDCP Layer is waited until Timeout.

E.6.2.29 GsmSndcpSequence ()

Function name	INT GsmSndcpSequence (INT Nsapi, INT Sapi, GSM_SNSM_SEQ_PAR *Message, INT Timeout)			
Feature summary	Notifies the sequence number when SNDCP layer re-opens downlink data transmission.			
	Type	Argument name	Description	I/O

Argument	INT INT GSN_SNSM_SEQ_PAR* INT	Nsapi Sapi Message Timeout	NSAPI of SNDCP SAPI between LLC-SNDCP. Message Data Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Nsapi
Specify NSAPI which is already set up by GsmSndcpConfig( ). It' s possible to specify from 5 to 15.
2) Sapi
Specify SAPI which is already set up by GsmSndcpConfig( ). It' s possible to specify 3, 5, 9, 11.
3) Message(GSM_SNSM_SEQ_PAR structure)
Set "DL Receive N-PDU Number" which was received from mobile station.
(Note) If input value except it, there is no guarantee for operation.
4) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	When this function is executed, SNSM_SEQ_IND to SNDCP Layer is transmitted . After that, SNSM_SEQ_RSP from SNDCP Layer is waited until Timeout.

E.6.2.30 GsmSndcpModify ()

Function name	INT GsmSndcpModify(INT Nsapi, INT Sapi, GSM_SNSM_MODIFY_PAR *Message, INT Timeout)			
Feature summary	Change the established Link of NSAPI and SAPI.			
	Type	Argument name	Description	I/O
Argument	INT INT GSM_SNSM_MODIFY_PAR * INT	Nsapi Sapi Message Timeout	NSAPI of SNDCP SAPI between LLC and SNDCP Message Data Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Nsapi
Set the NSAPI that set with GsmSndcpConfig() . [5 to 15]
2) Sapi
Set the SAPI that set with GsmSndcpConfig() . 3, 5, 9, and 11 can be set.
3) Message Data
DLSndNpduNo: Set N-PDU number that sent on DL.
Range is 0 to 255.
ULRcvNpduNo: Set N-PDU number that sent on UL.
Range is 0 to 255.
Reliability_class: Set Reliability Class
Set to 2.
Delay_class: Set Delay class.
Set to NONE.
Peak_through: Set Peak Through.
Set to NONE.
Precd_class: Set Precedence Class.
Set to NONE.
tlli: Set the TLLI value in the range as below,
0 to 0xFFFFFFFF.

(Note) If input value except it, there is no guarantee for operation.
4)Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_SNSM_MODIFY_RSP from layer of SNDCP until timeout occur, after sent GSM_SNSM_MODIFY_IND to layer of SNDCP.

E.6.2.31 GsmSndcpRcvControl ()

Function name	INT GsmSndcpRcvControl(INT *Nsapi, INT *Sapi , INT Timeout)			
Feature summary	Receive notification that indicates all data for IntertRAT Handover has received from layer of SNCDP. Receive notification that indicates all retransmission data has sent to MS.			
	Type	Argument name	Description	I/O
Argument	INT * INT * INT	Nsapi Sapi Timeout	NSAPI of SNDCP SAPI between LLC and SNDCP Time-out value	Output Output Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Nsapi
NSAPI in GSM_CSN_RCVCMP_IND / GSM_CSN_SNDCP_IND is returned.
2) Sapi
SAPI in GSM_CSN_RCVCMP_IND / GSM_CSN_SNDCP_IND is returned.
3) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_CSN_RCVCMP_IND / GSM_CSN_SNDCP_IND until timeout occur.

E.6.2.32 GsmSndcpFwd ()

Function name	INT GsmSndcpFwd (INT Nsapi, INT Sapi, INT Timeout)			
Feature summary	Force release to Layer of SNDCP.			
	Type	Argument name	Description	I/O
Argument	INT INT INT	Nsapi Sapi Timeout	NSAPI of SNDCP SAPI between LLC and SNDCP Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Nsapi
Set the NSAPI that set with GsmSndcpConfig() . [5 to 15]
2) Sapi
Set the SAPI that set with GsmSndcpConfig() . 3, 5, 9, and 11 can be set.
3)Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_CSN_FWD_CNF from layer of SNDCP until timeout occur, after sent GSM_CSN_FWD_REQ to layer of SNDCP.

E.6.2.33 GsmSndcpDeact ()

Function name	INT GsmSndcpDeact(INT Nsapi, INT Sapi, GSM_SNSM_DEACT_PAR *Message, INT Timeout)			
Feature summary	The sequence number in which the downlink data transmission is restarted is notified.			
	Type	Argument name	Description	I/O
Argument	INT INT GSM_SNSM_DEACT_PAR* INT	Nsapi Sapi Message Timeout	NSAPI of SNDCP SAPI between LLC and SNDCP Message Data Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Nsapi
Set the NSAPI that set with GsmSndcpConfig() . [5 to 15]
2) Sapi
Set the SAPI that set with GsmSndcpConfig() . 3, 5, 9, and 11 can be set.
3) Message Data
Llc_Release_Ind: Set Release Indicator.
REL_NORMAL Release procedure with MS is done.
REL_LOCALEND Release procedure with MS is not done, specified NSAPI / SAPI is released.
Tlli: Set TLLI [0 to 0xFFFFFFFF]
(Note) If input value except it, there is no guarantee for operation.
4) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	This function waits GSM_SNSM_DEACT_RSP from layer of SNDCP until timeout occur, after sent GSM_SNSM_DEACT_IND to layer of SNDCP.

E.6.2.34 SimulatorStart ()

Function name	INT SimulatorStart (INT Mode, INT Timeout)			
Feature summary	Starts the operation of MD8480B.			
	Type	Argument name	Description	I/O
Argument	INT INT	Mode Timeout	Operation mode(Set 0) Time-out value	Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	
Feature details				

The following common variable can set the GSM frame offset (Offset to the W-CDMA timing) . This common variable needs to be described before executing SimulatorStart( ).

GsmWcdmaOffset 0 to 1249 [symbols] by Unit of 1 symbol.

1) Mode

Set this value to 0.

2) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.35 SndMessage ()

Function name	INT SndMessage (INT BtsNo, INT Frame, INT Loch, INT LochNo, CHAR*message, INT Length)			
Feature summary	Sends Message from scenario.			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT CHAR* INT	BtsNo Frame Loch LochNo Message Length	BTS No Primitive which is transmitted. Logical Channel Logical Channel No / SAPI No Data(less than 1600) which is transmitted. Data length which is transmitted.	Input Input Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	
Feature details				

1) BtsNo

Set UNIT_GSM to this parameter.

2) Frame

Select Frame from the following values,

GSM_DL_DATA_REQ /* When Datalink I frame is sent */
GSM_DL_UNITDATA_REQ /* When Datalink UI frame is sent */
GSM_LL_UNITDATA_REQ /* When LLC frame is sent */
GSM_RLC_DATA_REQ /* When RLC message is sent (We support only PBCCH. When you send RLC Control Message except PBCCH, please use GsmSndControl())*/

3) Loch

Select from the following values for Loch.

GSMD_BCCH
GSMD_PCH
GSMD_AGCH
GSMD_SACCH
GSMD_SACCH8
GSMD_SACCH4
GSMD_FACCH
GSMD_SDCCH
GSMD_SDCCH8
GSMD_SDCCH4
GSMD_PDTCH
GSMD_PBCCH

4) LochNo

LochNo needs to be set the SAPI value only when Loch is GSMD_PDTCH, GSMD_FACCH, GSMD_SDCCH, GSMD_SDCCH8, GSMD_SDCCH4, GSMD_SACCH, GSMD_SACCH8 and GSMD_SACCH4. Set 0 for the other Lochs.

5) message

Set the pointer of the sent message . The message has to be less than 1600 bytes (octets).

6) Length

Set the length of the message.

7) Common Variables

The following common variables are used to specify the details functionalities. Set these common variables before executing SndMessage( ).

7-1) When Loch is GSMD_BCCH and GSMD_SACCH:

GsmSysInfo_TC	TC value [BCCH: 0 to 7, SACCH: 0 to 2]
GsmSysInfo_Alloc	TC Allocation Mask [0x0000 to 0xFFFF]
GsmSysInfo_Carry	Select from following values for the BCCH Carrier Type (Only GSMD_BCCH) GSM_BCCH_NORM GSM_BCCH_EXT
GsmL2Header	Select from following values for the L2 Header Type (Only GSMD_SACCH) GSM_DATALINK_NORMAL_HEADER GSM_DATALINK_SHORT_L2_HEADER

7-2) When Loch is GSMD_PCH:

GsmPagingGroup	PAGING_GROUP
GsmPagingBlockIndex	Paging block index

7-3) When Loch is GSMD_PBCCH:

GsmPSI_Category	Select from the following values for the PSI classification CATEGORY_PSI1 CATEGORY_HR CATEGORY_LR
GsmPSI_Position	Set PSI position [PSI1: N/A, HR: 0 to 16, LR: 0 to 63]

7-4) When Loch is GSMD_SDCCH8 and GSMD_SACCH8:

GsmSndSubChNo — Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH7]

7-5) When Loch is GSMD_SDCCH4 and GSMD_SACCH4:

GsmSndSubChNo — Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH3]

7-6) When Loch is GSMD_FACCH and GSMD_SACCH:

GsmSndSubChNo — Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH1]

Note: When GsmSndSubChNo is 0 (default), Sub-channel Number set by GsmRfchConfig() is valid.

Note: After executing GsmDatalinkEstablish(), GsmSndSubChNo is initialized by 0 automatically.

Remarks:	

E.6.2.36 RcvMessage ()

Function name	INT RcvMessage (INT *BtsNo, INT *Frame, INT *Loch, INT LochNo, CHAR *message, INT Timeout)			
Feature summary	Receive Message.			
	Type	Argument name	Description	I/O
Argument	INT*	BtsNo	BTS No	Output
	INT*	Frame	Primitive which is received.	Output
	INT*	Loch	Logical Channel	Output
	INT*	LochNo	Logical Channel Number / SAPI	Output
	CHAR*	Message	Data which is received.	Output
	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	> 0	Normal termination (Reception data length)	
		< 0	Abnormal termination (Timeout)	

Feature details

1) BtsNo

BtsNo is not used.

2) Frame

One of the following values returns as Frame.

GSM_DL_DATA_IND
GSM_DL_UNITDATA_IND
GSM_DL_RANDACCESS_IND
GSM_DL_ESTABLISH_IND
GSM_LL_UNITDATA_IND

3) Loch

One of The following values returns as Loch.

GSMU_RACH
GSMU_SACCH
GSMU_SACCH8
GSMU_SACCH4
GSMU_FACCH
GSMU_SDCCH
GSMU_SDCCH8
GSMU_SDCCH4
GSMU_PDTCH

4) LochNo

When the Frame is GSM_DL_DATA_IND, GSM_DL_UNITDATA_IND, GSM_DL_ESTABLISH_IND or GSM_LL_UNITDATA_IND, the SAPI value returns as LochNo.

5) message

Received message returns to this pointer.

6) Common Variables

The following common variables are used to specify the details functionalities. Refer these common variables after executing RcvMessage( ).

6-1) When Frame is GSM_DL_DATA_IND, GSM_DL_UNITDATA_IND, GSM_DL_ESTABLISH_IND and GSM_DL_RANDACCESS_IND:

GsmRcvSubChNo Sub-channel Number [GSM_SUBCH0 to GSM_SUBCH7]

7) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn't complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.37 BtsPower ()

Function name	INT BtsPower (INT BtsNo, INT Rf_CH, INT UpDown, INT Timeout)			
Feature summary	Increases or decreases the power of each RF channel			
	Type	Argument name	Description	I/O
Argument	INT INT INT INT	BtsNo Rf_Ch UpDown Timeout	BTS No RF Channel Increase and decrease of Tx Power Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	

Feature details

1) BtsNo

Set UNIT_GSM to BtsNo.

2) Rf_CH

Select Rf_CH from the following values,

GSMRF_D_CCH
GSMRF_D_TCH
GSMRF_D_PCCH
GSMRF_D_PTCH

3) UpDown

Set the value of increase and decrease for downlink Tx Power by the following range.

Setting value: -99 to 99[dBm] by 1dBm Unit.

4) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’ t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.38 BtsAttenuator ()

Function name	INT BtsAttenuator(INT BtsNo, INT TxAtt, INT RxRefPower, INT Timeout)			
Feature summary	Sets the TX Attenuator and RX Reference Power.			
	Type	Argument name	Description	Output
Argument	INT INT INT INT	BtsNo TxAtt RxRefPower Timeout	BTS No Tx Attenuator Rx reference Power Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination	

Feature details

1) BtsNo

BtsNo can be specified from UNIT_BTS1,UNIT_BTS2,UNIT_BTS3. In case of setting Tx Attenuator, Reference Level of GSM, set UNIT_BTS2 for BtsNo.

2) TxAtt

Set the TxAtt value in the range as below,

0 to 1200(0 to 120.0dB) 0.1dB Unit 0.1dB step

(Note) If Main connector is specified in SETUP screen, 15dB is added to the setting of TxAtt in execution of this function.

3) RxRefPower

Set the RxRefPower in the range as below,

-400 to +50(-40.0dB to +5dB) 0.1dB Unit, 1dB step

(Note) If Main connector is specified in SETUP screen, 20dB is added to the setting of RxRefPower in execution of this function.

4)Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn’t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.39 CteConfig ()

Function name	INT CteConfig (INT Loch, INT LochNo, CTE_CONFIG_PAR*CteParam, INT Timeout)			
Feature summary	Configures TE layer parameters(relates TE type=Service to Logical Channel)			
	Type	Argument name	Description	I/O
Argument	INT INT CTE_CONFIG_PAR* INT	Loch LochNo CteParam Timeout	Logical Channel Logical Channel No TE Type and parameter Time-out value	Input Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	
Feature details				

1) Loch
Select Loch from the following values, (Note: There is no distinction for downlink and uplink.)
GSM_PCH
GSM_AGCH
GSM_SACCH
GSM_SACCH8
GSM_SACCH4
GSM_FACCH
GSM_SDCCH
GSM_SDCCH8
GSM_SDCCH4
GSM_TCH
GSM_PDTCH

2) LochNo
When Loch is GSM_TCH, Sub-channel Number can be set. [GSM_SUBCH0 or GSM_SUBCH1]
When Sub-channel need not be set or Loch is not GSM_TCH, set 0.

3) CteParam
Setting of variable of structure object is as follows.

3-1) TeType
Select TE Type from the following values,
```
TE_TYPE_NONE                /* NO TE */
TE_TYPE_VOICE_GSM_EFR       /* EFR */
TE_TYPE_VOICE_GSM_FR        /* FR */
TE_TYPE_VOICE_GSM_HR        /* HR */
TE_TYPE_VOICE_GSM_AMR       /* AMR */
TE_TYPE_IPPACKET            /* IP PACKET */
TE_TYPE_CSD                 /* CSD */
```

3-2) Rate
Set as below,
When TeType is TE_TYPE_CSD or TE_TYPE_IPPACKET or TE_TYPE_NONE,
Set to 0.
When TeType is except TE_TYPE_CSD and TE_TYPE_VOICE_GSM_AMR, TE_TYPE_IPPACKET, TE_TYPE_NONE,
```
VOICE_RATE_12_2             /* 12.2 kbps */
```
When TeType is TE_TYPE_VOICE_GSM_AMR, Select Rate from the following value,
```
VOICE_RATE_4_75             /* 4.75 kbps */
VOICE_RATE_5_15             /* 5.15 kbps */
VOICE_RATE_5_90             /* 5.90 kbps */
VOICE_RATE_6_70             /* 6.70 kbps */
VOICE_RATE_7_40             /* 7.40 kbps */
VOICE_RATE_7_95             /* 7.95 kbps */
VOICE_RATE_10_2             /* 10.2 kbps */
VOICE_RATE_12_2             /* 12.2 kbps */
```

3-3) TTI
Set TTI in the range from 1 to 255 frames by frame Unit as “Repeat length” specified in 3GPP spec .

3-4) NumOfTB
Set NumOfTB in the range from 1 to 255 as “Interleaved Block Number” specified in 3GPP spec.

3-5) TBS
Set TBS by bit Unit as “Block length” specified in 3GPP spec.

3-6) Frame
Select from the following values,
GSM_DL_UNITDATA_REQ
GSM_DL_DATA_REQ
GSM_PH_DATA_REQ

3-7) Layer
Select Layer from the following values for the downlink data transmission.
```
GSML1           /* When Loch is GSM_TCH */
GSMSNDCP        /* When Loch is GSM_PDTCH */
```

3-8) Data [1024]

Set the data for FIX data transmission. Since it' s necessary to set up as word alignment, in case of setting data of odd number byte, add dummy data to the last of data.(When transmitting from TE, dummy data is ignored.)

3-9) CsdConfPar

It' s used by CSD test. Refer to "E.6.3.3 common variable initialization value and range" for details.

4) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.40 CteRelease ()

Function name	INT CteRelease (INT Loch, INT LochNo, INT Timeout)			
Feature summary	Releases TE layer(deletes the relation between TE type and Logical Channel.)			
	Type	Argument name	Description	I/O
Argument	INT INT INT	Loch LochNo Timeout	Logical Channel Logical Channel No Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1) Loch

Select Loch from the following values, (Note: There is no distinction for downlink and uplink.)

GSM_PCH
GSM_AGCH
GSM_SACCH
GSM_SACCH8
GSM_SACCH4
GSM_FACCH
GSM_SDCCH
GSM_SDCCH8
GSM_SDCCH4
GSM_TCH
GSM_PDTCH

2) LochNo

When Loch is GSM_TCH, Sub-channel Number can be set. [GSM_SUBCH0 or GSM_SUBCH1]
When Sub-channel need not be set or Loch is not GSM_TCH, set 0.

3) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.41 CteConnect ()

Function name	INT CteConnect (INT Loch, INT LochNo, INT DownPort, INT UpPort, INT Direction, UCHAR*Opt, INT Timeout)			
Feature summary	Starts data transmission and reception for the assigned TE Type.			
	Type	Argument name	Description	I/O

Argument	INT	Loch	Logical Channel	Input
	INT	LochNo	Logical Channel No	Input
	INT	DownPort	Downlink port	Input
	INT	UpPort	Uplink port	Input
	INT	Direction	Call source	Input
	UCHAR*	Opt	Option	Input
	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0	Normal termination	
		≠ 0	Abnormal termination (Timeout)	
Feature details				

1) Loch

The following value can be specified for Loch. (Note: There is no distinction for downlink and uplink.)

GSM_PCH
GSM_AGCH
GSM_SACCH
GSM_SACCH8
GSM_SACCH4
GSM_FACCH
GSM_SDCCH
GSM_SDCCH8
GSM_SDCCH4
GSM_TCH
GSM_PDTCH

2) LochNo

When Loch is GSM_TCH, Sub-channel Number can be set. [GSM_SUBCH0 or GSM_SUBCH1]
When Sub-channel need not be set or Loch is not GSM_TCH, set 0.

3) DownPort

Select DownPort from the following values.

TE_PORT_NORMAL	/* Data from TE is transmitted to downlink. It can' t use when TeType of CteConfig() is TE_TYPE_VOICE_GSM_FR or TE_TYPE_VOICE_GSM_HR. */
TE_PORT_FIXDATA	/* Fixed pattern which was specified by CteConfig( ) is transmitted to downlink. */
TE_PORT_LOOPBACK	/* Uplink data is transmitted to downlink again. (Set up Uplink port similarly) Delay occurs about 1 Sec only when Speech data is lookbacked */
TE_PORT_PN9	/* PN9 is transmitted to downlink. */
TE_PORT_USERDATA	/* Input from Userdata terminal of Voice Codec is transmitted to downlink */
TE_PORT_AUDIO	/* Input from Audio terminal of Voice Codec is transmitted to downlink. */
TE_PORT_MSTOMS	/* Used in MS to MS test. The uplink data from another MD8480B is transmitted as downlink data . */

4) UpPort

Select UpPort from the following values.

TE_PORT_NORMAL	/* Uplink data is transmitted to TE. It can' t use when TeType of CteConfig() is TE_TYPE_VOICE_GSM_FR or TE_TYPE_VOICE_GSM_HR.*/
TE_PORT_LOOPBACK	/* Uplink data is transmitted to downlink again. (Set up Downlink port similarly.)*/
TE_PORT_USERDATA	/* Uplink data is output from Userdata terminal of Voice Codec. */
TE_PORT_AUDIO	/* Uplink data is output from Audio terminal. */
TE_PORT_MSTOMS	/* Used in MS to MS test. The received data on uplink is transmitted to another Signalling Tester. */

5) Direction

Parameter which can specify in Direction is as follows.

CALL_FROM_AIR	/* Calling from Air side */
CALL_FROM_TE	/* Calling from TE side */

6) Opt is not used.

7) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:	

E.6.2.42 CteDisconnect ()

Function name	INT CteDisconnect (INT Loch, INT LochNo, INT Direction, INT Timeout)			
Feature summary	Stops data transmission and reception for the assigned TE Type.			
	Type	Argument name	Description	I/O

Argument	INT	Loch	Logical Channel	Input
	INT	LochNo	Not used	Input
	INT	Direction	Disconnect source	Input
	INT	Timeout	Time-out value	Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1)Loch

Select Loch from the following values

GSM_PCH
GSM_AGCH
GSM_SACCH
GSM_SACCH8
GSM_SACCH4
GSM_FACCH
GSM_SDCCH
GSM_SDCCH8
GSM_SDCCH4
GSM_TCH
GSM_PDTCH

2) LochNo

When Loch is GSM_TCH, Sub-channel Number can be set. [GSM_SUBCH0 or GSM_SUBCH1]
When Sub-channel need not be set or Loch is not GSM_TCH, set 0.

3) Direction

Select Direction from the following values.

CALL_FROM_AIR /* Disconnection from Air side */
CALL_FROM_TE /* Disconnection from TE side */

4) Timeout

Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.43 CteSuspend ()

Function name	INT CteSuspend (INT Loch, INT LochNo, INT Timeout)			
Feature summary	Stops data transmission and reception for the assigned TE Type temporarily and received data is preserved.			
	Type	Argument name	Description	I/O
Argument	INT INT INT	Loch LochNo Timeout	Logical Channel Logical Channel No Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1)Loch
Select Loch from the following values. (Note: There are no distinctions in Uplilnk and Downlink.)
GSM_PDTCH

2) LochNo
Set to 0.

3) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.2.44 CteResume ()

Function name	INT CteResume (INT Loch, INT LochNo, INT Timeout)			
Feature summary	Restarts data transmission and reception for the assigned TE Type.			
	Type	Argument name	Description	I/O
Argument	INT INT INT	Loch LochNo Timeout	Logical Channel Logical Channel No Time-out value	Input Input Input
	Type	Function value	Description	
Function value	INT	= 0 ≠ 0	Normal termination Abnormal termination (Timeout)	

Feature details

1)Loch
Select Loch from the following values. (Note: There are no distinctions in Uplilnk and Downlink.)
GSM_PDTCH

2) LochNo
Set to 0.

3) Timeout
Specify the waiting time in unit of msec and 100msec step. Abnormal termination occurs if this functionality doesn' t complete by this value. Timeout does not occur if the value is set to NO_TIMEOUT.

Remarks:

E.6.3 Variable used in each function

E.6.3.1 Structure

The following structures are used.

* Setting parameters of Radio Frequency Channel (It' s used by GsmRfchConfig( ).

```
typedef  struct      {
        ULONG           Frequency;                    /* Frequency                   */
        SHORT           TxPower;                      /* Power                       */
        USHORT          Timeslot;                     /* Timeslot                    */
        USHORT          LochCombination;              /* LoCH Combination            */
        USHORT          SubChNo;                      /* Sub-Channel Number          */
        USHORT          TSC;                          /* TSC                         */
        USHORT          BSIC;                         /* BSIC                        */
        USHORT          BS_AG_BLKS_RES;               /* BS_AG_BLKS_RES              */
        USHORT          BS_PA_MFRMS;                  /* BS_PA_MFRMS                 */
        USHORT          BS_PBCCH_BLKS;                /* BS_PBCCH_BLKS               */
        USHORT          BS_PAG_BLKS_RES;              /* BS_PAG_BLKS_RES             */
        USHORT          Reserved1;                    /* reserved                    */
        USHORT          PSI1_REPEAT_PERIOD;           /* PSI1_REPEAT_PERIOD          */
        USHORT          PSI_COUNT_HR;                 /* PSI_COUNT_HR                */
        USHORT          PSI_COUNT_LR;                 /* PSI_COUNT_LR                */
        USHORT          CodingType;                   /* Coding Type                 */
        USHORT          USF;                          /* Not Used */
        UCHAR           MSPowerLevel;                 /* SACCH MS Power Level        */
        UCHAR           TimingAdvance;                /* SACCH Timing Advance        */
        GSM_HOPPING_PAR
                        Hopping;                      /* Frequency Hopping           */
        GSM_HSCSD_PAR
                        Hscsd;                        /* HSCSD                       */
        GSM_SMSCB_PAR
                        Smscb;                        /* SMSCB                       */
  } GSM_RF_CONFIG_PAR;

  typedef struct    {
        USHORT          Flag;                         /* Hopping ON/OFF              */
        USHORT          FreqBand;                     /* Frequency Band              */
        USHORT          MAIO;                         /* MAIO (0 to m-1)             */
        USHORT          HSN;                          /* HSN (0 to 63)               */
        USHORT          NumFreqList;                  /* m (1 to 64)                 */
        USHORT          Reserved;                     /* reserved                    */
        USHORT          ARFCN[MAX_FH_ARFCN];          /* ARFCN[0]-ARFCN[m-1]         */
  } GSM_HOPPING_PAR;

  typedef struct    {
        USHORT          MainTimeslot;                 /* Main Timeslot               */
        USHORT          Reserved[3];                  /* reserved                    */
  } GSM_HSCSD_PAR;

  typedef struct    {
        USHORT          Flag;                         /* SMSCB ON/OFF                */
        USHORT          Reserved[3];                  /* reserved                    */
  } GSM_SMSCB_PAR;
```

* Set parameter of each Logical Channel in Datalink Layer (It's used by GsmDatalinkEstablish( ) and GsmDatalinkMsgSet( )).

```
typedef struct {
    USHORT    Mode;       /*Mode[Ack/Unack/Nohead]   * /
    USHORT    T200;       /*Timer T200               * /
    USHORT    N200;       /*Retransmission N200      * /
    USHORT    Reserved;
} GSM_MDL_CONFIG_PAR;

typedef struct {
    UCHAR     MsgType;    /* Message Type   */
    UCHAR     Reserve;    /* Reserve        */
    UCHAR     Channel;    /* Channel        */
    UCHAR     Sapi;       /* SAPI           */
    USHORT    Opt1;       /* Option         */
    USHORT    Opt2;       /* Option         */
} GSM_DL_MSGSET_PAR;
```

* RLC/MAC layer CONFIG setting parameters (It's used by GsmRlcConfig( ).

```
typedef struct {
    USHORT  SP_BIT;         /* S/P bit setting */
    USHORT  UL_DATA_SIZE;   /* Channel Coding (Uplink) */
    USHORT  DL_DATA_SIZE;   /* Channel Coding (Downlink) */
    USHORT  RLC_MODE;       /* ACK/UNACK Mode */
    USHORT  P_SEND_TIM;     /* Packet Uplink ACK/NACK transmission timing */
    USHORT  SEQ_MODE;       /* message sequence mode */
    USHORT  SLOT_VALUE;     /* Slot value */
    USHORT  USF_VALUE;      /* USF value */
    USHORT  T3169;          /* T3169 Timer value */
    USHORT  T3191;          /* T3191 Timer value */
    USHORT  T3193;          /* T3193 Timer value */
    USHORT  T3195;          /* T3195 Timer value */
    USHORT  N3101;          /* N3101 counter value */
    USHORT  N3103;          /* N3103 counter value */
    USHORT  N3105;          /* N3105 counter value */
} CRLC_CONFIG_LPARA;
```

* GRR layer CONFIG setting parameters. (It's used by GsmRrConfig( )).

```
typedef struct {
    USHORT  SEQ_MODE;     /* message sequence mode */
    USHORT  MSG_TYPE;     /* specification of message type   */
    USHORT  SLOT_TYPE;    /* Specification of slot type      */
    USHORT  TT02;         /* TT02 Timer value   */
} CGRR_CONFIG_LPARA;
```

* GRR layer Information setting parameters (It's used by GsmRrInfoSet( )).

```
typedef struct {
    USHORT  DRX_MODE;     /* DRX mode    */
    USHORT  PAG_GRUOP;    /* Paging group   */
    USHORT  PAG_INDEX;    /* Paging block index */
} CGRR_INFO_PAR;
```

* LLC layer CONFIG setting parameter (It' s used by GsmLlcConfig( )).

```
typedef struct {
        USHORT   T200;        /* T200 Timer value */
        USHORT   N200;        /* N200 counter value */
        USHORT   N201_U;      /* U, UI frame information Field maximum length /
        USHORT   N201_I;      /* I frame information Field maximum length */
        USHORT   mD;          /* I frame buffer size(Downlink) */
        USHORT   mU;          /* I frame buffer size(Uplink) */
        USHORT   kD;          /* Window size(Downlink) */
        USHORT   kU;          /* Window size(Uplink) */
} UNIT_SAPI;

typedef struct {
    USHORT      Version;          /* LLC version number */
    USHORT      Protect;          /* Protect Mode bit setting */
    ULONG       TLLI;             /* TLLI value */
    USHORT      SEQ_MODE;         /* message sequence mode*/
    USHORT      Reserve;          /* Reserve */
    UNIT_SAPI   SAPI_INF[16];     /* LLC layer parameter for each SAPI */
} CLL_CONFIG_LPARA;

/***** GSM_CLL_ASSIGN_REQ parameter    *****/
typedef struct{
    UCHAR    Tlli_old[4];
    UCHAR    Tlli_new[4];
    UCHAR    GEA;
} GSM_CLL_ASSIGN_PAR;
```

* SNDCP layer setting parameters (It' s used by GsmSndcpConfig( ), GsmSndcpSequence( ), and GsmSndcpStopAssign( )).

```
typedef struct{
    CHAR     Pcic;         /* Protocol control information compression */
    CHAR     Dcmp;         /* Data compression                   */
    UCHAR    Npnsync;      /* N-PDU Number Synchronization */
    UCHAR    Layer;        /* Destination layer to be sent */
    USHORT   Frame;        /* Primitive type                     */
    USHORT   SEQ_MODE; /* message sequence mode      */
    USHORT   TxWtime       /* Tx Wait Time                       */
} GSM_CSN_CONFIG_PAR;

/***** GSM_SNSM_MODIFY_IND parameter *****/
typedef struct{
    UCHAR    DLSndNpduNo;    /* Downlink Send N-PDU Number       */
    UCHAR    ULRcvNpduNo;    /* Uplink Receive N-PDU Number      */
    UCHAR      Reliability_class;
    UCHAR      Delay_class;
    UCHAR      Peak_through;
    UCHAR      Prece_class;
    UCHAR      tlli[4];
} GSM_SNSM_MODIFY_PAR;

/***** GSM_SNSM_SEQ_IND parameter *****/
typedef struct{
```

```
    UCHAR    DLSndNpduNo;    /* Downlink Send N-PDU Number     */
    UCHAR    ULRcvNpduNo;    /* Uplink Receive N-PDU Number    */
    UCHAR    DLRcvNpduNo;    /* Downlink Receive N-PDU Number */
} GSM_SNSM_SEQ_PAR;

/***** GSM_SNSM_STOPASGN_RSP parameter *****/
typedef struct{
    UCHAR    DLSndNpduNo;    /* Downlink Send N-PDU Number     */
    UCHAR    ULRcvNpduNo;    /* Uplink Receive N-PDU Number    */
} GSM_SNSM_STOPASGN_PAR;

/***** GSM_SNSM_DEACT_IND parameter   *****/
typedef struct{
    UCHAR    Llc_Release_Ind;
    UCHAR    tlli[4];
} GSM_SNSM_DEACT_PAR;
```

* TE layer setting parameter (It’ s used by CteConfig()).

```
typedef struct{
    USHORT   TTI;                 /* TTI */
    USHORT   ProtoType;           /* Protocol Type */
    USHORT   ChCoding;            /* Channel Coding */
    USHORT   Version;             /* Version */
    USHORT   MsIwfKv01;           /* MS->IWF(K) Ver.0/1 */
    USHORT   MsIwfKv2;            /* MS->IWF(K) Ver.2 */
    USHORT   IwfMsKv01;           /* IWF(K)->MS Ver.0/1*/
    USHORT   IwfMsKv2;            /* IWF(K)->MS Ver.2*/
    USHORT   T1;                  /* Acknowledge Timer */
    USHORT   T2;                  /* Reply delay */
    USHORT   N2;                  /* Maximum number of retransmission */
    USHORT   Pt;                  /* Type of data copression */
    USHORT   P0;                  /* V.42bis data compression request */
    USHORT   P1;                  /* V.42bis number of possible codewords in the algorithm */
    USHORT   P2;                  /* V.42bis maximum encodable data string length */
    USHORT   T4;                  /* Re-sequencing period */
    USHORT   UpSignal;            /* Up Signalling */
    USHORT   SREJtoSend;          /* SREJ to Send */
} RLP_PAR;

typedef struct{
    RLP_PAR  RlpConfPar;
    UCHAR    ConElement;          /* Connection Element */
    UCHAR    SyncMode;            /* Synchronous Mode */
    UCHAR    InfoTransCap;        /* Information Transfer Capability */
    UCHAR    UserInfoL2P;         /* User Information Layer2 Protocol */
    USHORT   SerialRxBuff;        /* Serial Receive Buffer */
    UCHAR    Waiur;               /* Wanted Air Interface User Rate */
    UCHAR    AcptChCoding;        /* Acceptable Channel Coding */
    UCHAR    UpMaxTchNo;          /* Uplink Max TCH/F Number */
    UCHAR    DownMaxTchNo;        /* Downlink Max TCH/F Number */
    UCHAR    Uimi;                /* UIMI */
    UCHAR    Reserve1;            /* Reserve field */
} CSD_PAR;
```

```
typedef struct{
    UCHAR       TeType;       /* TE type */
    UCHAR       Rate;         /* Voice rate */
    UCHAR       TTI;          /* Repeat length */
    UCHAR       NumOfTB;      /* Interleaved Block Number */
    USHORT      TBS;          /* Block length */
    USHORT      Frame;        /* Primitive type */
    UCHAR       Layer;        /* Destination layer in downlink */
    UCHAR       Reserve2;     /* Reserve */
    UCHAR       Data[1024];   /* Fixed pattern data buffer */
    USHORT      Reserve3;     /* Reserve */
    CSD_PAR     CsdConfPar;   /* CSD Config Parameter */
} CTE_CONFIG_PAR;
```

E.6.3.2 Common variables

The following common variables are used.

Table E5-4 Common variables

Type	Variable name	Content
INT	GsmWcdmaOffset	Tx offset against W-CDMA
INT	GsmSysInfo_TC	TC
USHORT	GsmSysInfo_Alloc	TC Allocation Mask
INT	GsmSysInfo_Carry	BCCH Carrier (BCCH Norm/BCCH Ext)
UCHAR	GsmL2Header	L2 Header Type
INT	GsmPagingGroup	PAGING_GROUP
INT	GsmPagingBlockIndex	Paging Block Index
INT	GsmBS_PA_MFRMS	BS_PA_MFRMS
INT	GsmBS_AG_BLKS_RES	BS_AG_BLKS_RES
INT	GsmCCCH_CONF	CCCH_CONF
INT	GsmPSI_Category	PSI Category (PSI1/HR/LR)
INT	GsmPSI_Position	PSI Position
INT	GsmBS_PBCCH_BLKS	BS_PBCCH_BLKS
INT	GsmBS_PAG_BLKS_RES	BS_PAG_BLKS_RES
INT	GsmBS_PCC_CHANS	BS_PCC_CHANS
INT	GsmAssignInfo	Uplink or Downlink information for sending Immediate Assignment from scenario.
INT	GsmTIValue	Transaction Identifier
UCHAR	GsmSndSubChNo	Downlink Sub-channel Number
UCHAR	GsmRcvSubChNo	Uplink Sub-channel Number
GSM_RF_CONFIG_PAR	GsmRfConfigD_CCH	Downlink CCH Information
GSM_RF_CONFIG_PAR	GsmRfConfigD_TCH	Downlink TCH Information
GSM_RF_CONFIG_PAR	GsmRfConfigU_CCH	Uplink CCH Information
GSM_RF_CONFIG_PAR	GsmRfConfigU_TCH	Uplink TCH Information
GSM_RF_CONFIG_PAR	GsmRfConfigD_PCCH	Downlink PCCH(GPRS) Information
GSM_RF_CONFIG_PAR	GsmRfConfigD_PTCH	Downlink PTCH(GPRS) Information
GSM_RF_CONFIG_PAR	GsmRfConfigU_PCCH	Uplink PCCH(GPRS) Information
GSM_RF_CONFIG_PAR	GsmRfConfigU_PTCH	Uplink PTCH(GPRS) Information
GSM_MDL_CONFIG_PAR	GsmDlConfigBCCH	Datalink Config of BCCH
GSM_MDL_CONFIG_PAR	GsmDlConfigPCH	Datalink Config of PCH
GSM_MDL_CONFIG_PAR	GsmDlConfigAGCH	Datalink Config of AGCH
GSM_MDL_CONFIG_PAR	GsmDlConfigRACH	RACH Datalink Config
GSM_MDL_CONFIG_PAR	GsmDlConfigFACCH	FACCH Datalink Config
GSM_MDL_CONFIG_PAR	GsmDlConfigSACCH	SACCH Datalink Config
GSM_MDL_CONFIG_PAR	GsmDlConfigSDCCH	SDCCH Datalink Config
CRLC_CONFIG_LPARA	GsmRlcConfigPar	RLC/MAC Layer config parameter
CGRR_CONFIG_LPARA	GsmRrConfigPar	GRR Layer config parameter
CGRR_INFO_PAR	GsmRrInfoPar	GRR Layer information parameter
CLL_CONFIG_LPARA	GsmLlcConfigPar	LLC Layer config parameter
GSM_CSN_CONFIG_PAR	GsmSndcpConfigPar	SNDCP Layer config parameter
RLP_PAR	CteConfigCSD .CsdConfPar . RlpConfPar	RLP config parameter

CSD_PAR	CteConfigCSD .CsdConfPar	CSD config parameter
CTE_CONFIG_PAR	CteConfigCSD	TE Layer config parameter

E.6.3.3 Initial values of common variables and ranges

INT	Range	Initial value
GsmWcdmaOffset	0 – 1249 [symbol]	0
GsmSysInfo_TC	0 – 7	0
GsmSysInfo_Alloc	0x0000 – 0xFFFF	0x0000
GsmSysInfo_Carry	GSM_BCCH_NORM GSM_BCCH_EXT	GSM_BCCH_NORM
GsmPagingGroup	0 – 80	0
GsmPagingBlockIndex	0 – 8	0
GsmBS_PA_MFRMS	2 – 9	2
GsmBS_AG_BLKS_RES	0 – 7	0
GsmCCCH_CONF	0, 1, 2, 4 or 6	0
GsmPSI_Category	CATEGORY_PSI1 CATEGORY_HR CATEGORY_LR	CATEGORY_PSI1
GsmPSI_Position	0 – 63	0
GsmBS_PBCCH_BLKS	1 – 4	1
GsmBS_PAG_BLKS_RES	0 – 11	0
GsmBS_PCC_CHANS	0 – 1	0
GsmAssignInfo	IMMASSIGN_GSM IMMASSIGN_GPRS_UL IMMASSIGN_GPRS_DL	IMMASSIGN_GSM
GsmTIValue	0x00 – 0xF0	0x00

UCHAR	Range	Initial value
GsmL2Header	GSM_DATALINK_NORMAL_HEADER GSM_DATALINK_SHORT_L2_HEADER	GSM_DATALINK_NORMAL_HEADER
GsmSndSubChNo	0, GSM_SUBCH0 – GSM_SUBCH7	0
GsmRcvSubChNo	0, GSM_SUBCH0 – GSM_SUBCH7	0

GSM_RF_CONFIG_PAR	Range	Initial value			
variable / member		GsmRfConfigD_CCH	GsmRfConfigD_TCH	GsmRfConfigU_CCH	GsmRfConfigU_TCH
Frequency	450600000 – 1989800000 [Hz]	936000000	936000000	936000000	936000000
TxPower	-99 – 0 [dBm]	0	0	-	-
Timeslot	TIMESLOT0 – TIMESLOT7	TIMESLOT0	TIMESLOT4	TIMESLOT0	TIMESLOT4
LochCombination	COMB_***	COMB_D_CCCH	COMB_D_TCHF	COMB_U_CCCH	COMB_U_TCHF
SubChNo	0 – 7	0	0	0	0
TSC	0 – 7	0	0	0	0
BSIC	0 – 63	0	-	0	-
BS_AG_BLKS_RES	0 – 7	0	-	-	-
BS_PA_MFRMS	2 – 9	2	-	-	-
BS_PBCCH_BLKS	1 – 4	-	-	-	-
BS_PAG_BLKS_RES	0 – 11	-	-	-	-
PSI1_REPEAT_PERIOD	1 – 16	-	-	-	-
PSI_COUNT_HR	0 – 16	-	-	-	-
PSI_COUNT_LR	0 – 63	-	-	-	-
CodingType	COD_***	-	COD_EFS	-	COD_EFS
USF(Not Used)	-	-	-	-	-
MSPowerLevel	0 – 255	0	0	-	-

Anritsu

TimingAdvance		0 – 255	0	0	0	0
Hopping	Flag	ON/OFF	-	HOPPING_OFF	-	HOPPING_OFF
	FreqBand	BAND_***	-	BAND_PGSM900	-	BAND_PGSM900
	MAIO	0 – m-1	-	0	-	0
	HSN	0 – 63	-	0	-	0
	NumFreqList	m (1 – 64)	-	1	-	1
	ARFCN[64]	0 - 1023	-	1	-	1
Hscsd	MainTimeslot	TIMESLOT1 – TIMESLOT7	-	0	-	0
Smscb	Flag	SMSCB_ON SMSCB_OFF	SMSCB_OFF	SMSCB_OFF	-	-

GSM_RF_CONFIG_PAR		Range	Initial value			
member \ variable			GsmRfConfigD_PCCH	GsmRfConfigD_PTCH	GsmRfConfigU_PCCH	GsmRfConfigU_PTCH
Frequency		450600000 – 1989800000 [Hz]	936000000	936000000	936000000	936000000
TxPower		-99 – 0[dBm]	0	0	-	-
Timeslot		TIMESLOT0 – TIMESLOT7	TIMESLOT1	TIMESLOT4	TIMESLOT1	TIMESLOT4
LochCombination		COMB_***	COMB_D_PCCCH	COMB_D_PDTCH	COMB_U_PCCCH	COMB_U_PDTCH
SubChNo		0 – 7	-	-	-	-
TSC		0 – 7	0	0	0	0
BSIC		0 – 63	-	-	-	-
BS_AG_BLKS_RES		0 – 7	-	-	-	-
BS_PA_MFRMS		2 – 9	-	-	-	-
BS_PBCCH_BLKS		1 – 4	1	-	-	-
BS_PAG_BLKS_RES		0 – 11	0	-	-	-
PSI1_REPEAT_PERIOD		1 – 16	1	-	-	-
PSI_COUNT_HR		0 – 16	0	-	-	-
PSI_COUNT_LR		0 – 63	0	-	-	-
CodingType		COD_***	COD_PACKET	COD_PACKET	COD_PACKET	COD_PACKET
USF(Not Used)		0 – 7	0	0	0	0
MSPowerLevel		0 – 255	0	0	-	-
TimingAdvance		0 – 255	0	0	0	0
Hopping	Flag	ON/OFF	-	-	-	-
	FreqBand	BAND_***	-	-	-	-
	MAIO	0 – m-1	-	-	-	-
	HSN	0 – 63	-	-	-	-
	NumFreqList	m (1 – 64)	-	-	-	-
	ARFCN[64]	0 - 1023	-	-	-	-
Hscsd	MainTimeslot	-	-	-	-	-
Smscb	Flag	-	-	-	-	-

COMB_***	
COMB_D_CCCH	/* FCCH + SCH + BCCH + CCCH */
COMB_D_SDCCH4	/* FCCH + SCH + BCCH + CCCH + SDCCH/4 + SACCH/C4 */
COMB_D_SDCCH8	/* SDCCH/8 + SACCH/C8 */
COMB_D_TCHF	/* TCH/F + FACCH/F + SACCH/TF */
COMB_D_TCHH	/* TCH/H + FACCH/H + SACCH/TH */
COMB_D_PCCCH	/* PBCCH + PCCCH + PDTCH/F + PACCH/F + PTCCH/F */
COMB_D_PDTCH	/* PDTCH/F + PACCH/F + PTCCH/F */
COMB_D_PCCCH_PDTCH	/* PBCCH + PCCCH + PDTCH/F + PACCH/F + PTCCH/F */
COMB_D_GPRSTEST	/* GPRS Harikiri test */

COMB_U_CCCH	/* RACH */
COMB_U_SDCCH4	/* RACH + SDCCH/4 + SACCH/C4 */
COMB_U_SDCCH8	/* SDCCH/8 + SACCH/C8 */
COMB_U_TCHF	/* TCH/F + FACCH/F + SACCH/TF */
COMB_U_TCHH	/* TCH/H + FACCH/H + SACCH/TH */
COMB_U_PCCCH	/* PRACH + PDTCH/F + PACCH/F + PTCCH/F */
COMB_U_PDTCH	/* PDTCH/F + PACCH/F + PTCCH/F */
COMB_U_PCCCH_PDTCH	/* PBCCH + PCCCH + PDTCH/F + PACCH/F + PTCCH/F */
COMB_U_GPRSTEST	/* GPRS Harikiri test */

COD_***	
COD_EFS	/* EFS */
COD_FS	/* FS */
COD_HS	/* HS */
COD_AFS_12_2	/* AFS (12.2k) */
COD_AFS_10_2	/* AFS (10.2k) */
COD_AFS_7_95	/* AFS (7.95k) */
COD_AFS_7_4	/* AFS (7.4k) */
COD_AFS_6_7	/* AFS (6.7k) */
COD_AFS_5_9	/* AFS (5.9k) */
COD_AFS_5_15	/* AFS (5.15k) */
COD_AFS_4_75	/* AFS (4.75k) */
COD_AHS_7_95	/* AHS (7.95k) */
COD_AHS_7_4	/* AHS (7.4k) */
COD_AHS_6_7	/* AHS (6.7k) */
COD_AHS_5_9	/* AHS (5.9k) */
COD_AHS_5_15	/* AHS (5.15k) */
COD_AHS_4_75	/* AHS (4.75k) */
COD_CSD_F14_4	/* CSD (Full 14.4k) */
COD_CSD_F9_6	/* CSD (Full 9.6k) */
COD_PACKET	/* Packet */

BAND_***	
BAND_PGSM900	/* P-GSM900 */
BAND_EGSM900	/* E-GSM900 */
BAND_RGSM900	/* R-GSM900 */
BAND_DCS1800	/* DCS1800 */
BAND_PCS1900	/* PCS1900 */
BAND_GSM450	/* GSM450 */
BAND_GSM480	/* GSM480 */
BAND_GSM850	/* GSM850 */

GSM_MDL_CONFIG_PAR	Range	Initial value			
variable / member		GsmDlConfigBCCH	GsmDlConfigPCH	GsmDlConfigAGCH	GsmDlConfigRACH
Mode	ACK/UNACK/NOHEAD	GSM_DATALINK_UNACK	GSM_DATALINK_UNACK	GSM_DATALINK_UNACK	GSM_DATALINK_UNACK
T200	1 – 2047 [frame]	-	-	-	-
N200	1 – 2047	-	-	-	-

GSM_MDL_CONFIG_PAR	Range	Initial value		
variable member		GsmDlConfigFACCH	GsmDlConfigSACCH	GsmDlConfigSDCCH
Mode	ACK/UNACK/NOHEAD	GSM_DATALINK_ACK	GSM_DATALINK_UNACK	GSM_DATALINK_ACK
T200	1 – 2047 [frame]	400	1800	500
N200	1 – 2047	5	5	5

CRLC_CONFIG_LPARA	Range	Initial value
variable member		GsmRlcConfigPar
SP_BIT	-	SP_ON_8_FB
UL_DATA_SIZE	CS_1/CS_2/CS_3/CS_4	CS_1
DL_DATA_SIZE	CS_1/CS_2/CS_3/CS_4	CS_1
RLC_MODE	ACK_MODE	ACK_MODE
P_SEND_TIM	-	P_ACK_8_CT
SEQ_MODE	SEQMODE_AUTO	SEQMODE_AUTO
SLOT_VALUE	UL1DL1/ UL1DL2/ UL1DL3/ UL1DL4/ UL2DL1/ UL2DL2/ UL2DL3/	UL1DL1
USF_VALUE	0-7	0
T3169	-	541[5 sec]
T3191	-	541[5 sec]
T3193	-	163[1.5 sec]
T3195	-	541[5 sec]
N3101	-	32
N3103	-	2
N3105	-	3

(Note) Value which is indicated in the above Range is only confirmed operation

CGRR_CONFIG_LPARA	Range	Initial value
variable member		GsmRrConfigPar
SEQ_MODE	SEQMODE_AUTO	SEQMODE_AUTO
MSG_TYPE	MSG_NORMAL MSG_PACKET	MSG_NORMAL
SLOT_TYPE	SINGLE_SLOT MULTI_SLOT	SINGLE_SLOT

CGRR_INFO_PAR	Range	Initial value
variable / member		GsmRrInfoPar
DRX_MODE	NON_DRX DRX	NON_DRX
PAG_GROUP	0 – 80	0
PAG_INDEX	0 – 8	0

CLL_CONFIG_LPARA		Range	Initial value
variable / member			GsmLlcConfigPar
Version		0	0
Protect		0 – 1	1
TLLI		0x00000000 – 0x7FFFFFFF	0xC8765432
SEQ_MODE		SEQMODE_AUTO	SEQMODE_AUTO
SAPI_INF[0 – 15]	T200	1 – 4095	50
	N200	1 – 15	3
	N201_U	140 – 1520	500
	N201_I	140 – 1520	1503
	mD	0, 9 – 24320	1520
	mU	0, 9 – 24320	1520
	kD	1 – 255	16
	kU	1 – 255	16

GSM_CSN_CONFIG_PAR	Range	Initial value
variable / member		GsmSndcpConfigPar
Pcic	SN_PCIC_NOCOMP SN_PCIC_RFC1144 SN_PCIC_RFC2507	SN_PCIC_NOCOMP
Dcmp	SN_DCMP_NOCOMP	SN_DCMP_NOCOMP
Npnsync	SN_NPNSYNC_OFF SN_NPNSYNC_ON	SN_NPNSYNC_OFF
Layer	GSMLLC	GSMLLC
Frame	GSM_LL_DATA_REQ GSM_LL_UNITDATA_REQ	GSM_LL_UNITDATA_REQ
SEQ_MODE	SEQMODE_AUTO	SEQMODE_AUTO
TxWtime	0-65535	11

RLP_PAR	Range	Initial value
variable / member		CteConfigCSD. CsdConfPar. RlpConfPar
TTI	1 - 65535	2
ProtoType	RLP_PROTO_UMTS, RLP_PROTO_GSM	RLP_PROTO_GSM
ChCoding	RLP_CHCODE_TCHF_4_8, RLP_CHCODE_TCHF_9_6, RLP_CHCODE_TCHF_14_4, RLP_CHCODE_TCHF_28_8, RLP_CHCODE_TCHF_43_2	RLP_CHCODE_TCHF_4_8
Version	RLP_VERSION_0, RLP_VERSION_1, RLP_VERSION_2	RLP_VERSION_0
MsIwfKv01	0 - 61	61
MsIwfKv2	0 – Kmax Kmax is maximum of WindowSize	240
IwfMsKv01	0 - 61	61
IwfMsKv2	0 – Kmax Kmax is maximum of WindowSize	240
T1	38 - 65535	100
T2	0 - 65535	0
N2	1 - 65535	6
Pt	0	0
P0	0 - 3	0
P1	512 - 65535	0
P2	6 - 250	0
T4	3 - 65535	0
UpSignal	0 - 1	0
SREJtoSend	RLP_SRJTOSND_NO, RLP_SRJTOSND_YES	RLP_SRJTOSND_NO

CSD_PAR	Range	Initial value
variable / member		CteConfigCSD. CsdConfPar
RlpConfPar	-	“as above described”
ConElement	CSD_CE_TR, CSD_CE_NT	CSD_CE_NT
SyncMode	CSD_SYNCMD_SYNC, CSD_SYNCMD_ASYNC	CSD_SYNCMD_ASYNC
InfoTransCap	CSD_ITC_UDI, CSD_ITC_AUDIO	CSD_ITC_AUDIO
UserInfoL2P	CSD_UIL2P_OUTBAND, CSD_UIL2P_COPNFLCT	CSD_UIL2P_OUTBAND
Waiur	CSD_WAIUR_9_6, CSD_WAIUR_14_4	CSD_WAIUR_9_6
AcptChCoding	CSD_ACC_9_6, CSD_ACC_14_4	CSD_ACC_9_6
UpMaxTchNo	1 - 4	1
DownMaxTchNo	1 - 4	1
Uimi	1 - 4	1

CTE_CONFIG_PAR	Range	Initial value
variable / member		CteConfigCSD
TeType	TE_TYPE_NONE TE_TYPE_VOICE_GSM_EFR TE_TYPE_VOICE_GSM_FR TE_TYPE_VOICE_GSM_HR TE_TYPE_VOICE_GSM_AMR TE_TYPE_IPPACKET TE_TYPE_CSD	TE_TYPE_CSD
Rate	-	0
TTI	1 - 255	13
NumOfTB	1 - 255	3
TBS	0 - 65535	240
Frame	GSM_DL_UNITDATA_REQ GSM_DL_DATA_REQ GSM_PH_DATA_REQ	GSM_PH_DATA_REQ
Layer	GSML1 GSMSNDCP	GSML1
Data[1024]	-	all 0
CsdConfPar	-	“as below described”

(Note) In case of using function of GSM CSD, Set up RLP_PAR, CSD_PAR, CTE_CONFIG_PAR to value of Initial Value. Even if setting value except it, I don’t offer a guarantee of operation. It’s possible to modify for the following member of value with modification of Rate.

CteConfigCSD.CsdConfPar.RlpConfPar.TTI	2 : 9.6kbps 4 : 14.4kbps
CteConfigCSD.CsdConfPar.Waiur	CSD_WAIUR_9_6 : 9.6kbps CSD_WAIUR_14_4 : 14.4kbps
CteConfigCSD.CsdConfPar.AcptChCoding	CSD_ACC_9_6 : 9.6kbps CSD_ACC_14_4 : 14.4kbps
CteConfigCSD.TBS	240 : 9.6kbps 290 : 14.4kbps

E.6.4 Scheduling of BCCH and SACCH

In case of transmitting BCCH and SACCH data by SndMessage() function, it's necessary to set up the following common variables in advance:

GsmSysInfo_TC
GsmSysInfo_Alloc
GsmSysInfo_Carry (Only BCCH)
GsmL2Header (Only SACCH)

GsmSysInfo_TC is Transmission Cycle (TC). About TC, refer to 3GPP TS05.02 6.3.1.3. The range of the value which GsmSysInfo_TC can take is 0 – 7 for BCCH and 0 – 2 for SACCH. GsmSysInfo_Alloc is the assignment which should be transmitted for the TC which was set by GsmSysInfo_TC. That is, when many messages which want to be transmit on the one TC exists, these scheduling of messages are set by GsmSysInfo_Alloc.

(LSB) (MSB)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SI9	SI13	SI9	SI13	SI9	SI13	SI9	SI13	SI9	SI13	SI9	SI13	SI9	SI13	SI9	SI13

As example above, in case of transmitting System Information Type9 and System Information Type13 to TC=4 on BCCH alternately, describe the scenario as follows.

```
//System Information Type9 and Type13
        UCHAR SI_Type9[23] = {....};
        UCHAR SI_Type13[23] = {....};
        GsmSysInfo_TC = 4;
        GsmSysInfo_Alloc = 0x5555;
        SndMessage(UNIT_GSM, GSM_DL_UNITDATA_REQ,
                        GSMD_BCCH, 0, SI_Type9, sizeof(SI_Type9));
        GsmSysInfo_TC = 4;
        GsmSysInfo_Alloc = 0xAAAA;
        SndMessage(UNIT_GSM, GSM_DL_UNITDATA_REQ,
                        GSMD_BCCH, 0, SI_Type13, sizeof(SI_Type13));
//End
```

After SndMessage() is executed, GsmSysInfo_Alloc returns to 0x0000 automatically. In case of setting it to 0x0000 and SndMessage() is executed, it's the same operation as setting 0xFFFF, that is, the same message is transmitted repeatedly on the TC.

When system information messages are sent on BCCH Ext, set GsmSysInfo_Carry to GSM_BCCH_EXT. After SndMessage() is executed, GsmSysInfo_Carry returns to GSM_BCCH_NORM automatically and system information messages shall be sent on BCCH NORM. As described in 3GPP TS05.02 6.5.1, BS_AG_BLKS_RES shall be set to a value greater than zero when BCCH Ext is used. PCH and AGCH may share the resource on where the message is not sent on BCCH Ext by GsmSysInfo_Alloc.

```
//System Information Type2 quater
                UCHAR SI_Type2q[23] = {....};
                GsmSysInfo_TC = 5;
                GsmSysInfo_Alloc = 0x5555;
                GsmSysInfo_Carry = GSM_BCCH_EXT;
                SndMessage(UNIT_GSM, GSM_DL_UNITDATA_REQ,
                                GSMD_BCCH, 0, SI_Type2q, sizeof(SI_Type2q));
//End
```

When the message on SACCH contains the short L2 header, set GsmL2Header to GSM_DATALINK_SHORT_L2_HEADER. After SndMessage() is executed, GsmL2Header returns to GSM_DATALINK_NORMAL_HEADER automatically and the message shall be sent with the normal format (Format B4).The SACCH data as the argument of SndMessage() takes 21 octets containing the short L2 header which consists of two bits.

```
//Measurement Information
                UCHAR SACCH_DATA[21] = {....};
                GsmSysInfo_TC = 2;
                GsmSysInfo_Alloc = 0xFFFF;
                GsmL2Header = GSM_DATALINK_SHORT_L2_HEADER;
                SndMessage(UNIT_GSM, GSM_DL_UNITDATA_REQ, GSMD_SACCH,
                                GSM_SAPI0, SACCH_DATA, sizeof(SACCH_DATA));
//End
```

E.6.5 Explanation regarding sample scenarios

The following sample scenarios are attached in
C:¥Mx848000¥Scenario¥SRC¥Call_Sample(GSM)
C:¥Mx848000¥Scenario¥SRC¥Attach_Sample(GPRS)

Scenario name	Outline of operation
GSM_Idle.c	Start GSM simulation, set up FCCH+SCH+BCCH+CCH.
GSM_Loca.c	GSM registration(Location updating)
GSM_Orig_Voice(EFS).c	Call from MS. (EFS Voice)-By using a handset, a telephone call is possible.
GSM_Orig_Voice(FS).c	Call from MS. (FS Voice)-Speech data from MS is loopbacked by MD8480.
GSM_Orig_Voice(HS).c	Call from MS. (HS Voice) -Speech data from MS is loopbacked by MD8480.
GSM_Orig_Voice(AFS).c	Call from MS. (AFS Voice) -By using a handset, a telephone call is possible.
GSM_Orig_Voice(AHS).c	Call from MS. (AHS Voice) -By using a handset, a telephone call is possible.
GSM_Term_Voice(EFS).c	Call to MS.(EFS Voice) -By using a handset, a telephone call is possible.
GSM_Term_Voice(FS).c	Call to MS. (FS Voice) -Speech data from MS is loopbacked by MD8480.
GSM_Term_Voice(HS).c	Call to MS. (HS Voice) -Speech data from MS is loopbacked by MD8480.
GSM_Term_Voice(AFS).c	Call to MS. (AFS Voice) -By using a handset, a telephone call is possible.
GSM_Term_Voice(AHS).c	Call to MS. (AHS Voice) -By using a handset, a telephone call is possible.
GSM_Ms_Disc_Voice.c	Call disconnection from MS.(Voice)
GSM_Nw_Disc_Voice.c	Call disconnection from Network.(Voice)
GSM_IntraSystem_HO.c	Intra-RAT Handover
GSM_InterSystem_HO.c	Inter-RAT Handover
GSM_CellReselection.c	Cell Reselection from GSM to GSM.
GSM_Idle_SMS.c	Start GSM SMS(Short Message Service) simulation.
GSM_Orig_SMSonSACCH.c	SMS from MS during calling.
GSM_Orig_SMSonSDCCH.c	SMS from MS during Idle.
GSM_Term_SMSonSACCH.c	SMS to MS during calling.
GSM_Term_SMSonSDCCH.c	SMS to MS during Idle.
GSM_Idle_CSD.c	Start GSM CSD(Circuit Switched Data) simulation.
GSM_Orig_CSD.c	Calling from MS. (CSD)
GSM_Term_CSD.c	Call to MS. (CSD)
GSM_Idle_HSCSD.c	Start GSM HSCSD (High Speed Circuit Switched Data) simulation.
GSM_Orig_HSCSD.c	Calling from MS. (HSCSD)
GSM_Term_HSCSD.c	Call to MS. (HSCSD)
GSM_Ms_Disc_CSD.c	Call disconnection from MS.(CSD/HSCSD)
GSM_Nw_Disc_CSD.c	Call disconnection from Network. (CSD/HSCSD)
GSM_Idle_FH.c	Start GSM Frequency Hopping simulation
GSM_Voice(EFS)_FH.c	Call connection and disconnection with MS (EFS with Frequency Hopping)
GSM_SMSCBonSDCCH4(repetition).c	Start GSM SMSCB(Short Message Service Cell Broadcast) simulation. send the uniform CBS Messages consecutively on CBCH (SDCCH/4)
GSM_SMSCBonSDCCH8(repetition).c	Start GSM SMSCB(Short Message Service Cell Broadcast) simulation. send the uniform CBS Messages consecutively on CBCH (SDCCH/8)
GSM_SMSCBonSDCCH4(DRX).c	Start GSM SMSCB(Short Message Service Cell Broadcast) simulation. send one Schedule Message and one scheduled CBS Message on CBCH (SDCCH/4)
GSM_SMSCBonSDCCH8(DRX).c	Start GSM SMSCB(Short Message Service Cell Broadcast) simulation. send one Schedule Message and one scheduled CBS Message on CBCH (SDCCH/8)
GPRS_Idle.c(*1)	Start GPRS simulation, set up FCCH+SCH+BCCH+CCH.

GPRS_Idle_Loca_Attach_PDP.c	Start GPRS simulation. Set up CCH(FCCH+SCH+BCCH+CCH). Procedure for registration(location updating) GPRS Attach procedure and PTCH setting GPRS Activate PDP context procedure
GPRS_Loca.c(*1)	Procedure for registration location updating
GPRS_PBCCH_Idle_Loca_Attach_PDP.c	Start GPRS PBCCH simulation. Set up CCH(FCCH+SCH+BCCH+CCH). Set up PCCH. Procedure for GPRS registry. GPRS PBCCH Attach procedure and PTCH setting. GPRS PBCCH Activate PDP context procedure
GPRS_Attach.c(*1)	For LLC/SNDCP Layer Acknowledge Mode
GPRS_Idle_Loca_Attach_PDP _LLC_SNDCP_Ack.c	GPRS Attach procedure and PTCH setting Start GPRS simulation, set up FCCH+SCH+BCCH+CCH. Procedure for GPRS registry. GPRS Attach procedure and PTCH setting. GPRS Activate PDP context procedure.
GPRS_Detach.c	GPRS Detach procedure and PTCH setting
GPRS_Deact_PDP_context.c	GPRS Deactivate PDP context procedure
GPRS_Inter_SGSN_Routing_	Use this scenario as a sample to create a scenario that tests Inter SGSN. This scenario is considered Routing area update procedure with MS. /MD8480 can not transmit two signals of GPRS, this scenario is created by the transmitting Routing Area Update Request message form MS premise by changing Location Area Code . (Note)It is not confirmed using MS.
GPRS_PBCCH_comb11_PRACH11bit.c	Start GPRS PBCCH simulation. Set up CCH(FCCH+SCH+BCCH+CCH). Set up PCCH_PTCH. Set to PRACH 11bits. Procedure for GPRS registry. GPRS PBCCH Attach procedure and PTCH setting. GPRS PBCCH Activate PDP context procedure
GPRS_Harikiri.c	GPRS Harikiri test(refer to chapter E.5.6)
GPRS_Idle_Loca_Attach_BOTH_SMS.c	Start GPRS simulation. Set up CCH(FCCH+SCH+BCCH+CCH). Procedure for registration(location updating) GPRS Attach procedure and PTCH setting. GPRS SMS(Orig:SMS from MS / Term:SMS to MS)

(*1) These scenarios aren’t attached in after Ver.3.30 or after Ver.5.00.

E.6.6 Description of each sample scenario

E.6.6.1 GSM_Idle.c

```
/***** INCLUDE *****/
#include        <windows.h>
#include        "primitive.h"
#include        "parameter.h"
#include        "scenario.h"
#include        "stdio.h"
```

Input these header files definitely.

```
#define     MCC     0     /* 0 - 999     */
#define     MNC     0     /* 0 - 99     */
#define     LAC     0     /* 0 - 65535     */
```

It's necessary for setting up "Location Area Identification".

```
/***** FUNCTION *****/
DLLEXPORT INT ScenarioMain(LPVOID);

INT ScenarioMain(LPVOID dmy)
{

    CteStartPar.VoicePar.DtxMode=VOICE_DTX_DIS;
    GsmWcdmaOffset = 0;     /* 0 - 1249 [symbol]     */
    SimulatorStart(0, NO_TIMEOUT );
```

Set up GSM voice to DTX off.

Setting for "Timing Offset" of W-CDMA (Set up before SimulatorStart().)

Start Simulation

```
    //BtsAttenuator(UNIT_BTS2, 20, 0, NO_TIMEOUT);
```

Set up "Tx Attenuator" and "Reference power".

```
    GsmBS_PA_MFRMS      = 2;     /* 2 - 9          */
    GsmBS_AG_BLKS_RES = 0;     /* 0 - 7          */
    GsmCCCH_CONF        = 0;     /* 0, 1, 2, 4, 6     */
```

These are necessary for setting up "Control Channel Description".

```
    /* Downlink CCH */
    memset(&GsmRfConfigD_CCH, 0, sizeof(GSM_RF_CONFIG_PAR));
    GsmRfConfigD_CCH.TxPower             =    0;          /* [dBm]     */
    GsmRfConfigD_CCH.Frequency          = 936000000;     /* [Hz]     */
    GsmRfConfigD_CCH.Timeslot           = TIMESLOT0;
    GsmRfConfigD_CCH.LochCombination = COMB_D_CCCH;
    GsmRfConfigD_CCH.TSC                  = 0;
    GsmRfConfigD_CCH.BSIC                 = 0;
    GsmRfConfigD_CCH.BS_AG_BLKS_RES    = GsmBS_AG_BLKS_RES;
    GsmRfConfigD_CCH.BS_PA_MFRMS        = GsmBS_PA_MFRMS;
    GsmRfchConfig(GSMRF_D_CCH, &GsmRfConfigD_CCH, ACTIVATE_NOW, NO_TIMEOUT);
```

Set up each "RF Channel"

```
    /* Uplink CCH */
    memset(&GsmRfConfigU_CCH, 0, sizeof(GSM_RF_CONFIG_PAR));
    GsmRfConfigU_CCH.Frequency          = 891000000;     /* [Hz]     */
    GsmRfConfigU_CCH.Timeslot           = TIMESLOT0;
    GsmRfConfigU_CCH.LochCombination = COMB_U_CCCH;
    GsmRfConfigU_CCH.TSC                  = 0;
    GsmRfConfigU_CCH.BSIC                 = 0;
    GsmRfchConfig(GSMRF_U_CCH, &GsmRfConfigU_CCH, ACTIVATE_NOW, NO_TIMEOUT);

    /* BCCH */
    GsmDatalinkEstablish(GSM_BCCH, GSM_SAPI0, &GsmDlConfigBCCH, 0, GSML3, NO_TIMEOUT);

    /* PCH */
    GsmDatalinkEstablish(GSM_PCH,    GSM_SAPI0, &GsmDlConfigPCH,    0, GSML3, NO_TIMEOUT);

    /* AGCH */
    GsmDatalinkEstablish(GSM_AGCH, GSM_SAPI0, &GsmDlConfigAGCH, 0, GSML3, NO_TIMEOUT);
```

```
/* RACH */
GsmDatalinkEstablish(GSM_RACH, GSM_SAPI0, &GsmDlConfigRACH, 0, GSML3, NO_TIMEOUT);

/****************************************************/
/* GSM SystemInformation
/****************************************************/
```

Set up each "Loch"

```
// System Information Type1
{
    UCHAR BCCH_DATA[23] = {
        0x55,0x06,0x19,0x00,0x00,0x00,0x00,0x00,
        0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
        0x00,0x00,0x00,0x01,0x00,0x00,0x2B
    };
    GsmSysInfo_TC = 0;
   SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA, sizeof(BCCH_DATA));
    GsmSysInfo_TC = 4;
   SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
}
```

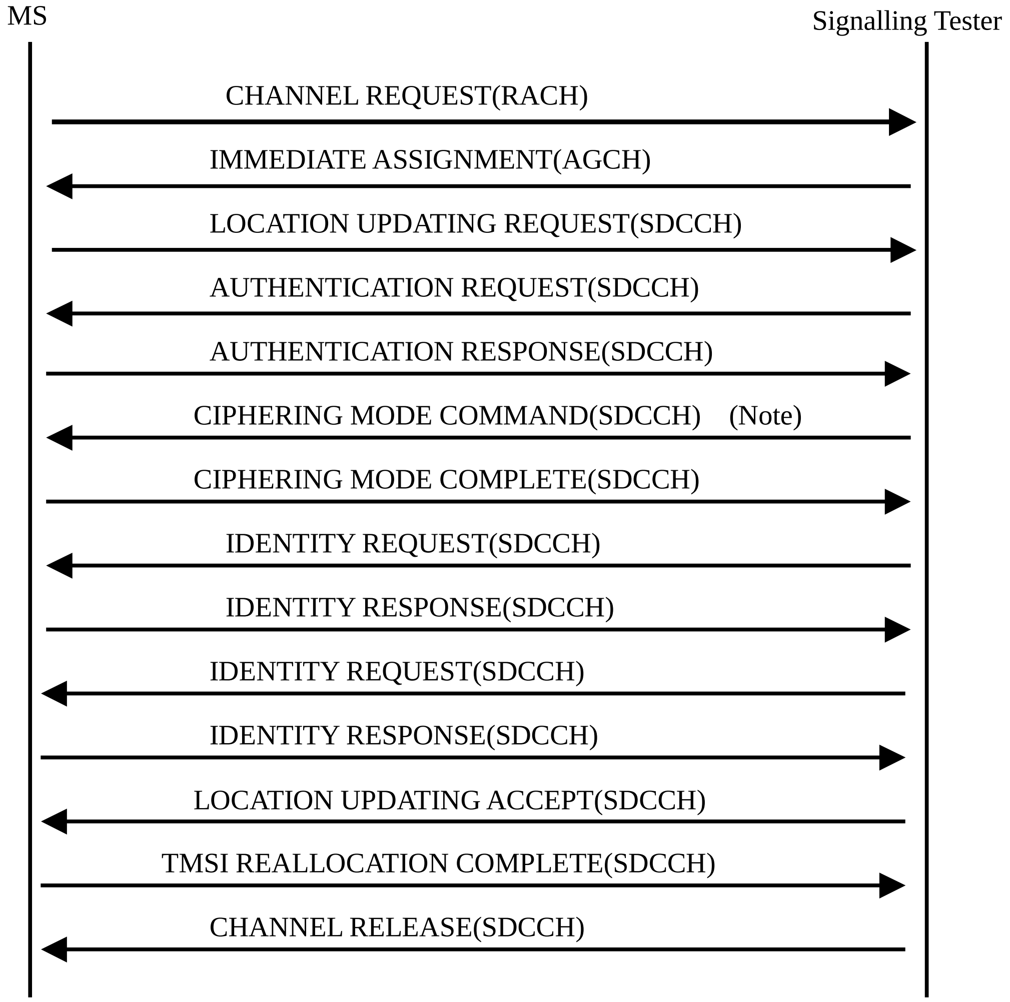
System information transmitted on the BCCH

```
// System Information Type2
{
    UCHAR BCCH_DATA[23] = {
        0x59,0x06,0x1A,0x00,0x00,0x00,0x00,0x00,
        0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
        0x08,0x00,0x00,0xFF,0x01,0x00,0x00
    };
    GsmSysInfo_TC = 1;
   SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
    GsmSysInfo_TC = 5;
   SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
}
// System Information Type3
{
    UCHAR BCCH_DATA[23] = {
        0x49,0x06,0x1B,0x00,0x01,0x00,0xF1,0x10,
        0x00,0x02,0x01,0x00,0x00,0x63,0x08,0x00,
        0x01,0x00,0x00,0x2B,0x2B,0x2B,0x2B
    };

    /*** Location Area Identification     ***/
    BCCH_DATA[5]   = (((MCC / 10) % 10) << 4) | ((MCC / 100) % 10);
    BCCH_DATA[6]   = 0xF0                     | (MCC % 10);
    BCCH_DATA[7]   = ((MNC % 10) << 4)        | ((MNC / 10) % 10);
    BCCH_DATA[8]   = LAC >> 8;
    BCCH_DATA[9]   = LAC & 0xFF;
    /*** Control Channel Description      ***/
    BCCH_DATA[10] = (GsmBS_AG_BLKS_RES << 3) | GsmCCCH_CONF;
    BCCH_DATA[11] = GsmBS_PA_MFRMS - 2;

    GsmSysInfo_TC = 2;
   SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
    GsmSysInfo_TC = 6;
   SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
}
// System Information Type4
{
    UCHAR BCCH_DATA[23] = {
        0x31,0x06,0x1C,0x00,0xF1,0x10,0x00,0x02,
        0x08,0x00,0x01,0x00,0x00,0x2B,0x2B,0x2B,
        0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B
    };
```

```
        /*** Location Area Identification        ***/
        BCCH_DATA[3]    = (((MCC / 10) % 10) << 4) | ((MCC / 100) % 10);
        BCCH_DATA[4]    = 0xF0                     | (MCC % 10);
        BCCH_DATA[5]    = ((MNC % 10) << 4)        | ((MNC / 10) % 10);
        BCCH_DATA[6]    = LAC >> 8;
        BCCH_DATA[7]    = LAC & 0xFF;

        GsmSysInfo_TC = 3;
       SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
        GsmSysInfo_TC = 7;
       SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_BCCH,0,BCCH_DATA,sizeof(BCCH_DATA));
    }

    /* SACCH */
    GsmDatalinkEstablish(GSM_SACCH, GSM_SAPI0, &GsmDlConfigSACCH, 0, GSML3, NO_TIMEOUT);

    // System Information Type5
    {
        UCHAR SACCH_DATA[19] = {
            0x49,0x06,0x1D,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
            0x00,0x00,0x01
        };
        GsmSysInfo_TC = 0;
        SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_SACCH,GSM_SAPI0,
                                                    SACCH_DATA,size of(SACCH_DATA));
    }
    // System Information Type6
    {
        UCHAR SACCH_DATA[19] = {
            0x2D,0x06,0x1E,0x00,0x01,0x00,0xF1,0x10,
            0x00,0x02,0x63,0xFF,0x2B,0x2B,0x2B,0x2B,
            0x2B,0x2B,0x2B
        };

        /*** Location Area Identification            ***/
        SACCH_DATA[5]    = (((MCC / 10) % 10) << 4) | ((MCC / 100) % 10);
        SACCH_DATA[6]    = 0xF0                     | (MCC % 10);
        SACCH_DATA[7]    = ((MNC % 10) << 4)        | ((MNC / 10) % 10);
        SACCH_DATA[8]    = LAC >> 8;
        SACCH_DATA[9]    = LAC & 0xFF;

        GsmSysInfo_TC = 1;
        SndMessage(UNIT_GSM,GSM_DL_UNITDATA_REQ,GSMD_SACCH,GSM_SAPI0,
                                                    SACCH_DATA, size of(SACCH_DATA));
    }

    return(0);
}
```

E.6.6.2 GSM_Loca.c

This scenario executes the following location updating procedure.

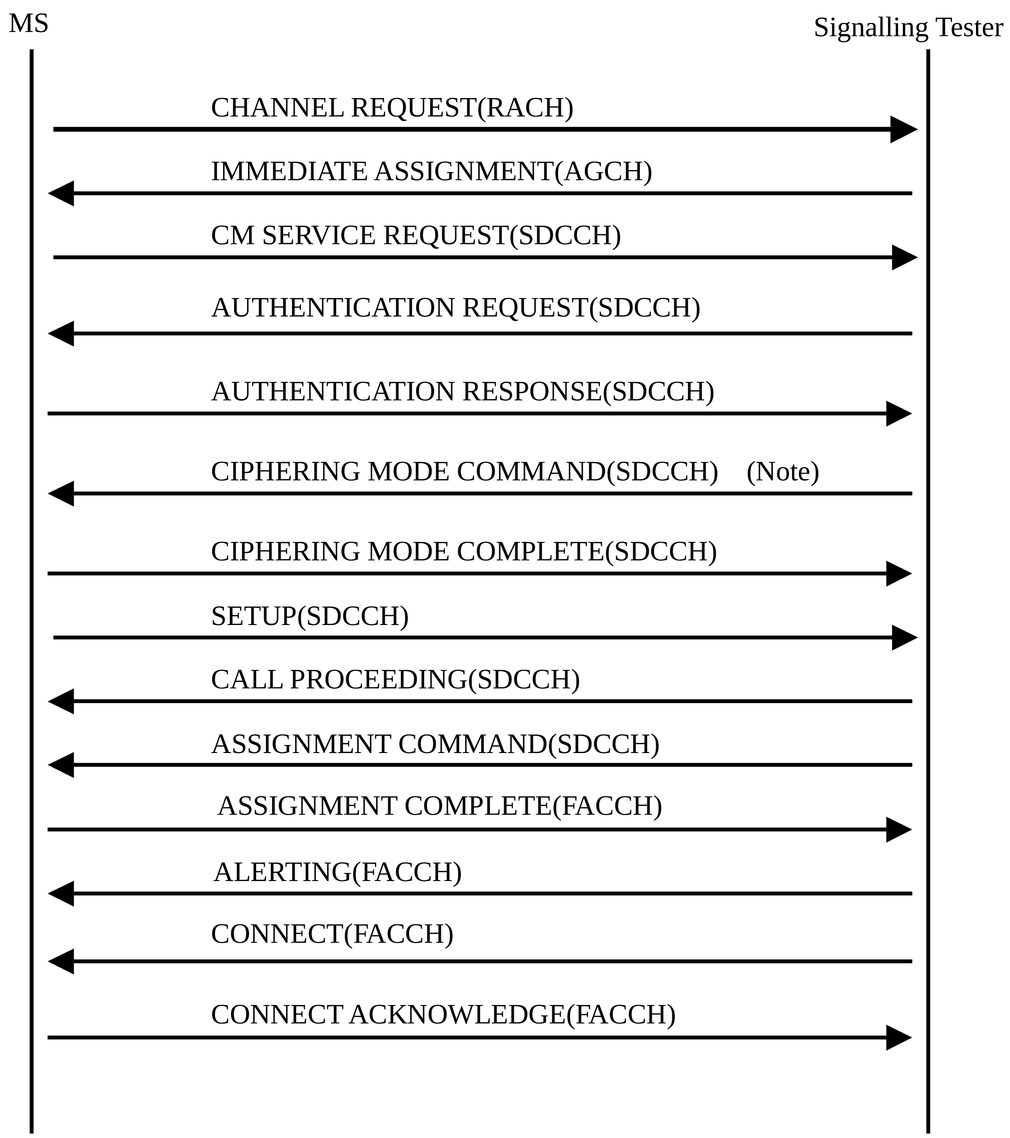


(Note) CIPHERING MODE is set to OFF.

E.6.6.3 GSM_Orig_Voice(***).c

(Note)*** is AFS,AHS,EFS,FS or HS.

This scenario executes the following call origination procedure.

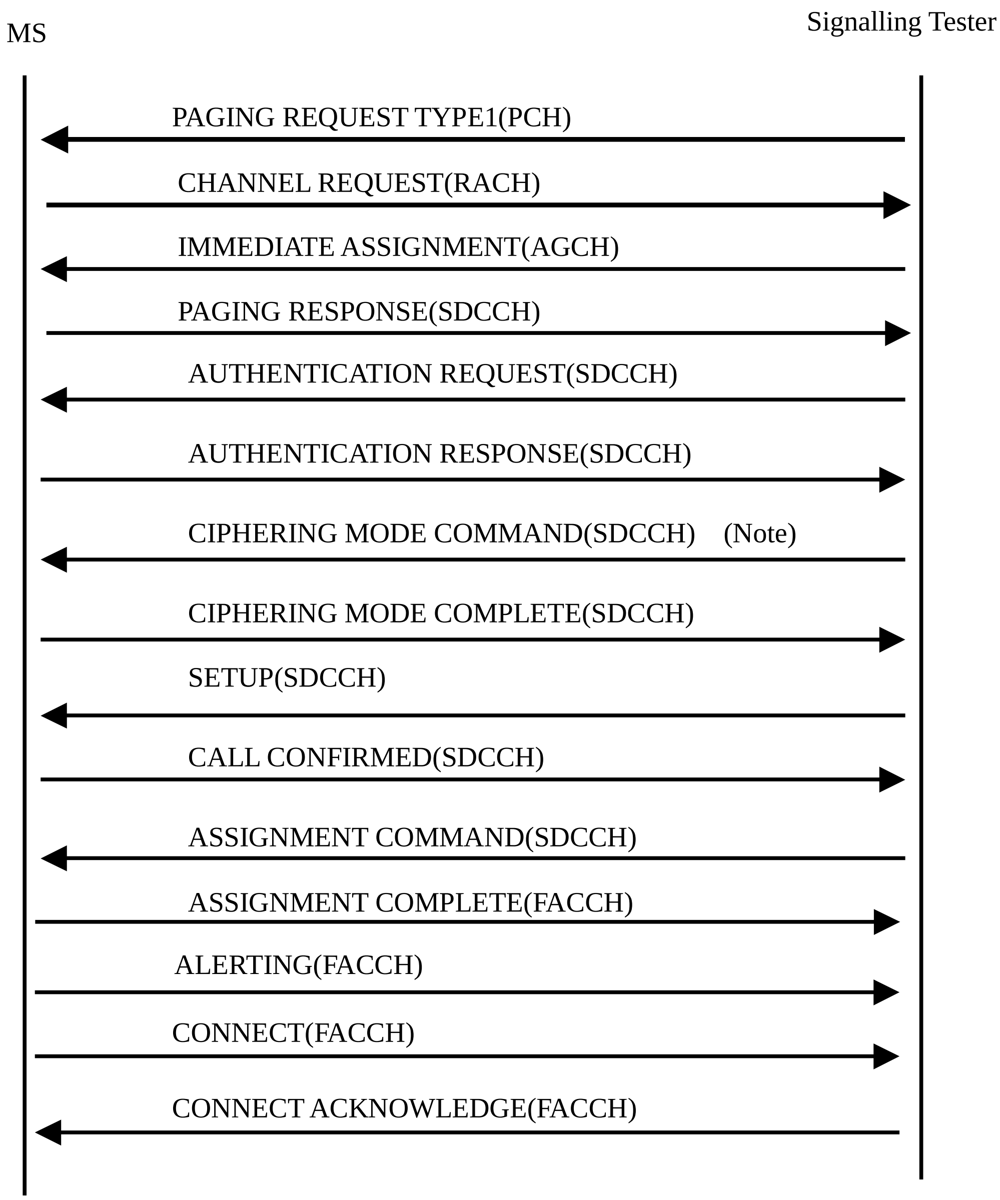


(Note) CIPHERING MODE is set to OFF.

E.6.6.4 GSM_Term_Voice.c

(Note)*** is AFS,AHS,EFS,FS or HS.

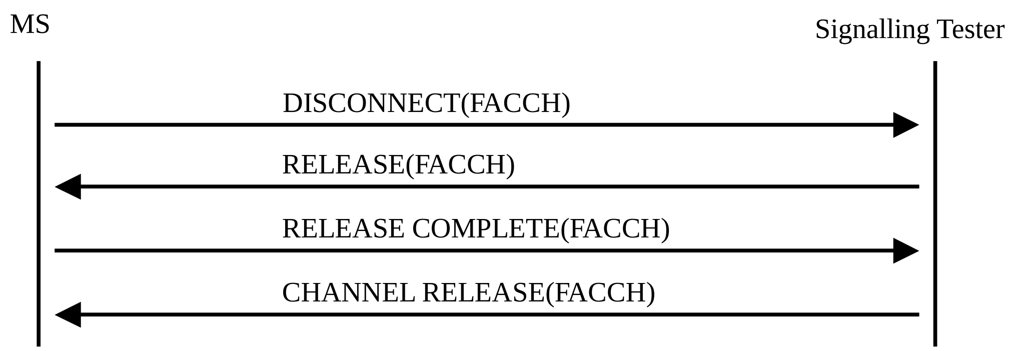
This scenario executes the following call termination procedure.



(Note) CIPHERING MODE is set to OFF.

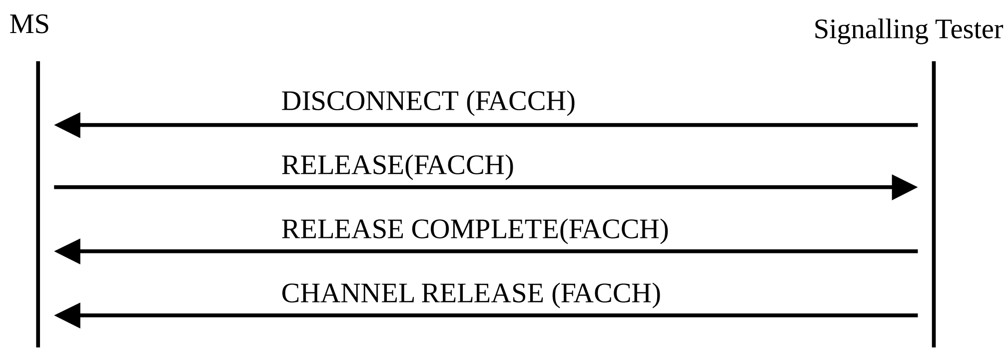
E.6.6.5 GSM_Ms_Disc_Voice.c

This scenario executes the following call disconnection from MS procedure.



E.6.6.6 GSM_Nw_Disc_Voice.c

This scenario executes the following call disconnection from NW procedure.



E.6.6.7 GSM_IntraSystem_HO.c

This scenario executes the following functionalities.

*Intra-cell Handover --- IntraCellHO( )
* Inter-cell Handover (Synchronization) --- InterCellSyncHO( )
* Inter-cell Handover(Non-Synchronization) --- InterCellAsyncHO( )

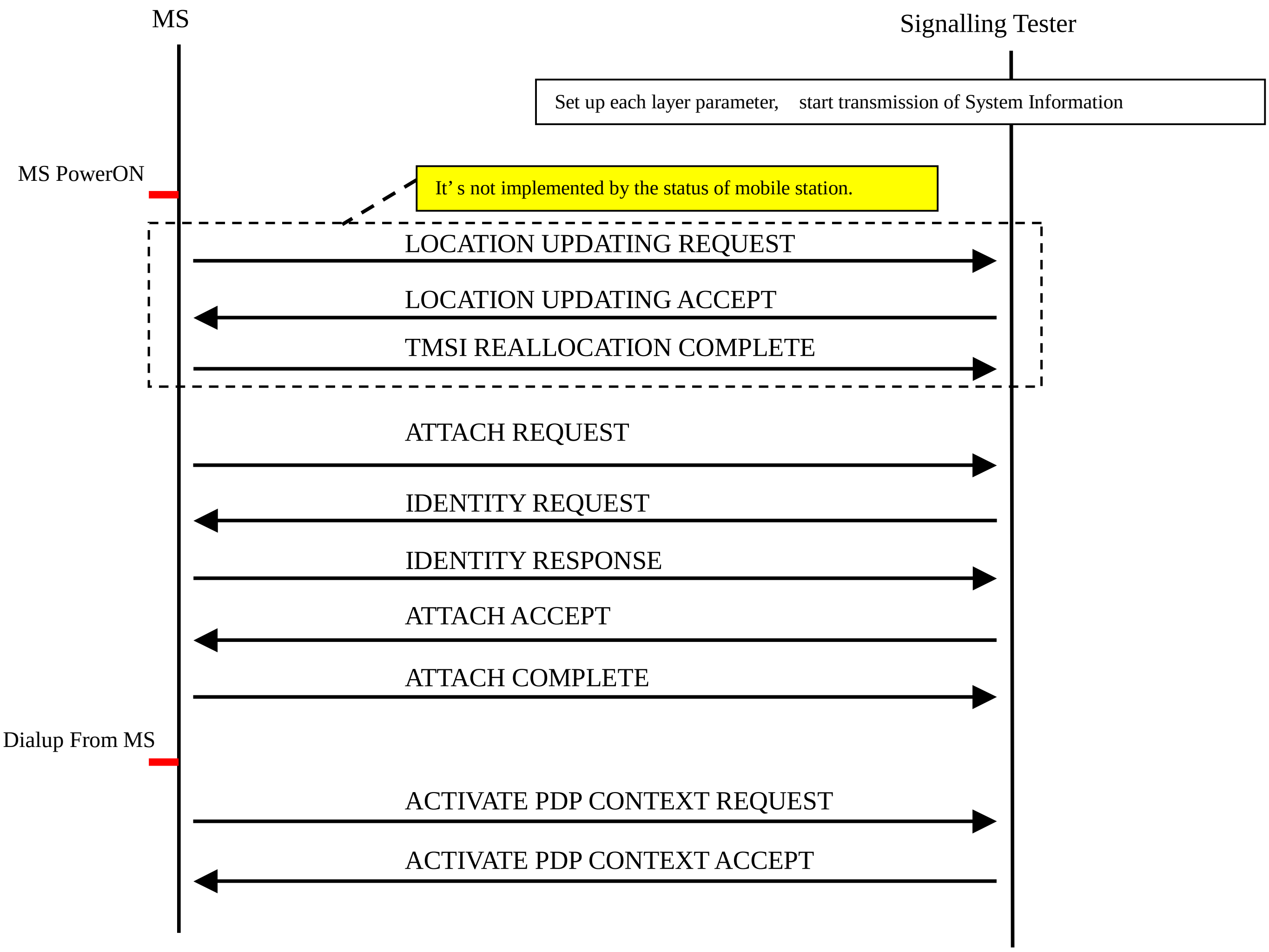
E.6.6.8 GSM_InterSystem_HO.c

This scenario executes the following functionalities.

*Inter-system Handover from GSM to UTRAN --- HO_GsmToUtran( )
*Inter-system Handover from UTRAN to GSM --- HO_UtranToGsm( )

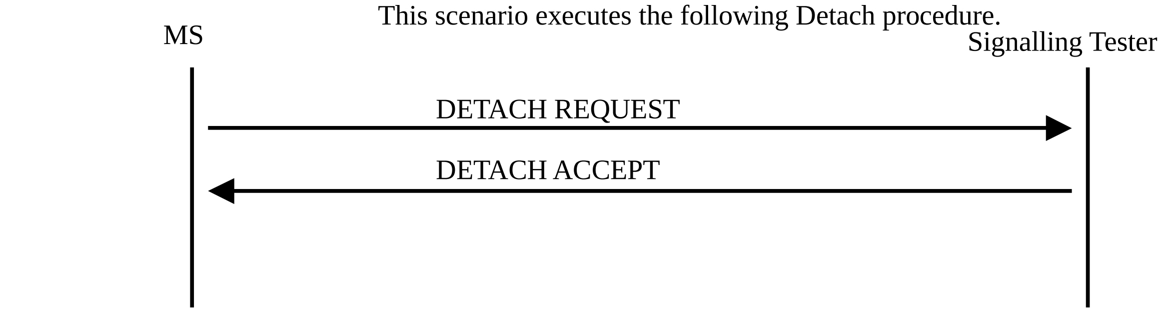
E.6.6.9 GPRS_Idle_Loca_Attach_PDP.c, GPRS_PBCCH_Idle_Loca_Attach_PDP.c,

These scenarios executes the following Attach procedure.



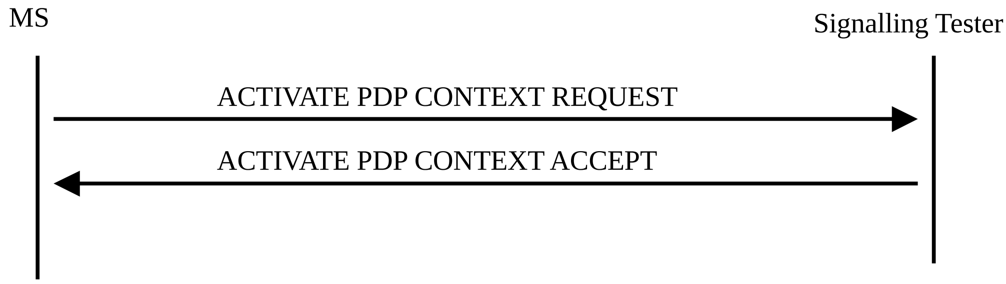
E.6.6.10 GPRS_Detach.c

This scenario executes the following Detach procedure.



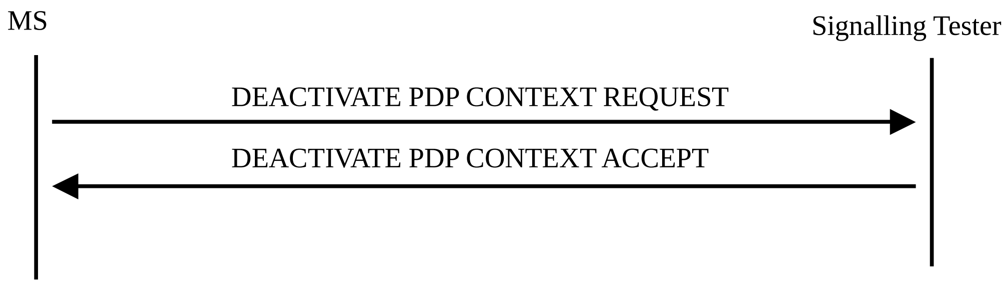
E.6.6.11 GPRS_ Act_PDP_context.c

This scenario executes the following PDP context Activate procedure



E.6.6.12 GPRS_Deact_PDP_context.c

This scenario executes the following PDP context Deactivate procedure



E.6.7 Restriction of Frequency and Timeslot settings for RF channels

This chapter describes the restrictions as regarding the settings of frequency and timeslot for RF channels.

(Note) Settings for the frequency and the timeslot for RF channels uses the library function GsmRfchConfig. (Refer to E.6.2.2)

There are following restriction items,

- When PCCH, TCH or PTCH is configured, it' s necessary to configure CCH in advance. (Common with UL/DL)
- CCH can be used only Timeslot0.(Common with UL/DL)
- PCCH can be used only Timeslot1. (Common with UL/DL)
  If you want to use PCCH except TimeSlot1, please contact MD8480A-support@zy.anritsu.co.jp .
- Set the same frequency for CCH and PCCH.(Common with UL/DL)
- Do not activate TCH and PTCH simultaneously. (Common with UL/DL)
- In case of setting the uplink frequency, it' s possible to receive CCH (or CCH and PCCH) by the frequency the all timeslots. It' s possible to receive TCH (or PTCH) by the specified frequency and the specified timeslot when CCH and TCH(or PTCH) are configured.
- In case of receiving the uplink signal, If Signalling Tester has Frequency Hopping option, the Signalling Tester can receive CCH and TCH simultaneously with different frequencies.
- If downlink frequencies are same between CCH and TCH(or PTCH), there is the following restriction about timeslots which can be used,
  - * When PCCH is not used with CCH, set the timeslot of TCH(or PTCH) in the range of 1 to 7.
  - * When PCCH is used with CCH, set the timeslot of TCH(or PTCH) in the range of 2 to 7.
- If downlink frequencies are different between CCH and TCH(or PTCH), there is the following restriction about the timeslot which can be used:
  - * When PCCH is not used, set TCH(or PTCH) timeslot in the range of Timeslot 2 to 6.
  - * When PCCH is used, set TCH(or PTCH) timeslot in the range of Timeslot 3 to 6.

The following figures show examples when TCH(PTCH) is set to Timeslot 4.

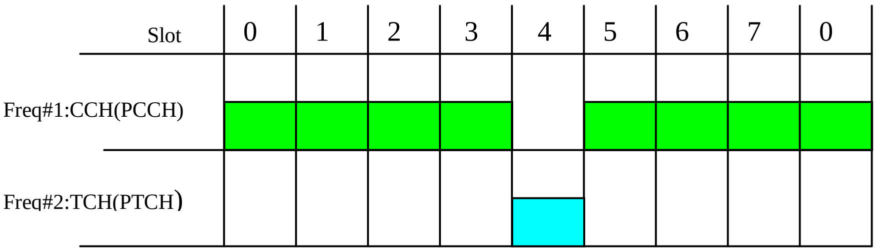


Figure E6-1　Case 1: The same frequency is set for CCH(PCCH) and TCH(PTCH)

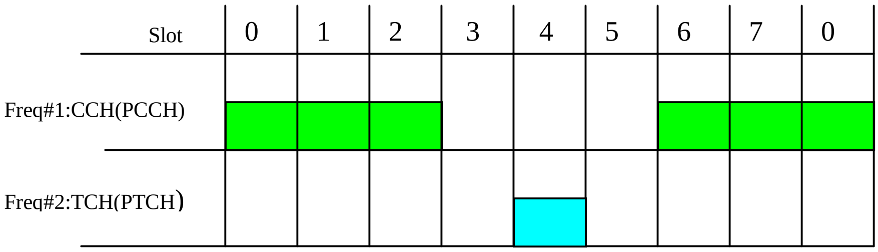


Figure E6-2　Case 2: Different frequencies are set for CCH(PCCH) and TCH(PTCH)

E.7 Modification method for GPRS Scenarios

Modification method for GPRS Sample Scenarios attached to V3.20 or before and V5.11 or before.

Scenario attached to V3.20 or before modification method is referred to section E7.2. Scenario attached to V5.11 or before modification method is referred to section E7.3.

E.7.1 About this manual

In MD8480B TDMA Firmware V3.30 and V5.00 or later, multi-slot and PBCCH/PCCCH functionalities are added. Owing to this and other modifications for RLC/MAC layer, there are cases that the sample scenarios attached to the old version(v3.20 or before) don' t work on V3.30 and V5.00 or later firmwares if the modification method described on this document aren' t done to the old sample scenarios. And there are cases that the sample scenarios attached to the old version(v5.11 or before) don' t work on V5.20 or later firmwares if the modification method described on this document aren' t done to the old sample scenarios.So please modify the old sample scenarios as described in this document for using with V3.30 , V5.00 and V5.20 or later firmwares.

(Note) Sample scenarios attached to V3.20 don' t support multi-slot and PBCCH/PCCCH functionalities. In case of these functionalities, please use the sample scenarios attached to V3.30, V5.00 and V5.20 or later.

(Note) GPRS sample scenarios are in the following folder of PC which was installed Control Software (MX848000A).

C:¥Mx848000¥Scenario¥SRC¥Attach_Sample(GPRS)

E.7.2 Scenario modification method(Sample scenarios attached to V3.20 or before)

Sample scenarios attached to V3.20 or before are required to be modified the following two scenarios

:GPRS_Idle.c, GPRS_Attach.c

E.7.2.1 Modification of GPRS_Idle.c

E.7.2.1.1 Modification of RLC config

Please modify configuration setting parts of RLC as follows. It' s the 64th to 79th line of GPRS_Idle.c attached to V3.20.

Before modification

```
GsmRlcConfigPar.SP_BIT          = SP_ON_4_FB;
GsmRlcConfigPar.UL_DATA_SIZE = CS_INF;
GsmRlcConfigPar.DL_DATA_SIZE = CS_INF;
GsmRlcConfigPar.RLC_MODE        = ACK_MODE;
GsmRlcConfigPar.P_SEND_TIM      = P_ACK_4_CT;
GsmRlcConfigPar.SEQ_MODE        = SEQMODE_AUTO;
GsmRlcConfigPar.SLOT_VALUE      = 1;
GsmRlcConfigPar.USF_VALUE       = 0;
GsmRlcConfigPar.T3169 = 500;
GsmRlcConfigPar.T3191 = 500;
GsmRlcConfigPar.T3193 = 1;
GsmRlcConfigPar.T3195 = 500;
GsmRlcConfigPar.N3101 = 30;
GsmRlcConfigPar.N3103 = 8;
GsmRlcConfigPar.N3105 = 15;
```

Please modify as follows.

```
GsmRlcConfigPar.SP_BIT          = SP_ON_8_FB;
GsmRlcConfigPar.UL_DATA_SIZE = CS_INF;
GsmRlcConfigPar.DL_DATA_SIZE = CS_INF;
GsmRlcConfigPar.RLC_MODE        = ACK_MODE;
GsmRlcConfigPar.P_SEND_TIM      = P_ACK_8_CT;
```

```
GsmRlcConfigPar.SEQ_MODE        = SEQMODE_AUTO;
GsmRlcConfigPar.SLOT_VALUE      = 1;
GsmRlcConfigPar.USF_VALUE       = 0;

GsmRlcConfigPar.T3169 = 541;
GsmRlcConfigPar.T3191 = 541;
GsmRlcConfigPar.T3193 = 163;
GsmRlcConfigPar.T3195 = 541;
GsmRlcConfigPar.N3101 = 32;
GsmRlcConfigPar.N3103 = 2;
GsmRlcConfigPar.N3105 = 3;
```

E.7.2.1.2 Replace for message contents

It' s about contents which is set up by GsmRrMsgSet(). Please replace message contents which is indicated the following.

/* Packet Uplink Assignment */ (Transmission contents of P_UL_ASSIGN_R)
It' s the 113th to 115th line of GPRS_Idle.c attached to V3.20.
Before modification

```
{0x40,0x28,0x59,0x0E,0xCA,0x86,0x43,0x01,0x00,0x00,0x88
,0x09,0x90,0x80,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
```

Please modify as follows.

```
{0x40,0x28,0x59,0x0E,0xCA,0x86,0x43,0x01,0x00,0x00,0x88
,0x08,0x10,0x80,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B,};
```

/* Packet Uplink Assignment 2 */ (Transmission contents of P_UL_ASSIGN_D)
It' s the 146th to 148th line of GPRS_Idle.c attached to V3.20.
Before modification

```
{0x40,0x28,0x59,0x0E,0xCA,0x86,0x43,0x01,0x00,0x00,0x88
,0x09,0x90,0x80,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
```

Please modify as follows.

```
{0x40,0x28,0x59,0x0E,0xCA,0x86,0x43,0x01,0x00,0x00,0x88
,0x08,0x10,0x80,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B,};
```

/* Packet Downlink Assignment */ (Transmission contents of P_DL_ASSIGN)
It' s the 167th to 179th line of GPRS_Idle.c attached to V3.20.
Before modification

```
{0x40,0x08,0x59,0x0E,0xCA,0x86,0x40,0x20,0x82,0x80,0x82,
0x05,0x00,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
        INT        Ref_posi       = 0;
        INT        Tlli_posi      = 0;
        INT        Str_posi       = 0;
```

Please modify as follows.

```
{0x40,0x08,0x59,0x0E,0xCA,0x86,0x40,0x20,0x80,0x40,0x00
,0x41,0x02,0x80,0x0B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B,};
        INT        Ref_posi       = 0;
        INT        Tlli_posi      = 0;
        INT        Str_posi       = 0;
```

Msg .CTRL_MSG [9] |= (MD_TSC << 3);
Msg .CTRL_MSG [9] |= ((MD_ARFCN>>9)&0x01);
Msg .CTRL_MSG [10] |= (MD_ARFCN>>1);
Msg .CTRL_MSG [11] |= (MD_ARFCN<<7);

/* Immediate Assignment(Packet Uplink Assignment) */
(Transmission contents of UL_ASSIGN)
It' s the 201st to 203th line of GPRS_Idle.c attached to V3.20.
Before modification
{0x2D,0x06,0x3F,0x10,0x08,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xC8,0x00,0x80,0xC1,0x03,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
Please modify as follows.
{0x2D,0x06,0x3F,0x10,0x08,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xC8,0x00,0x18,0x20,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B,};

/* Immediate Assignment(Packet Downlink Assignment) Message Setup */
(Transmission contents of DL_ASSIGN)
It' s the 225th to 227th line of GPRS_Idle.c attached to V3.20.
Before modification
{0x2D,0x06,0x3F,0x30,0x08,0x00,0x00,0x00,0x00,0x00,0x00
,0x00,0xD0,0x00,0x00,0x00,0x08,0x10,0x42,0x41,0x2B,0x2B
,0x2B};
Please modify as follows.
{0x2D,0x06,0x3F,0x30,0x08,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xD0,0x00,0x00,0x00,0x08,0x04,0x23,0x2B,0x2B,0x2B,
0x2B,};
/* Immediate Assignment(Packet Uplink Assignment) */
(Transmission contents of UL_ASSIGN_TWO_PH)
It' s the 259th to 264th line of GPRS_Idle.c attached to V3.20.
Before modification
{0x2D,0x06,0x3F,0x10,0x08,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xC5,0x50,0x80,0x00,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
INT Ref_posi = 7;
INT Tlli_posi = 0;
INT Str_posi = 14;
Please modify as follows.
{0x2D,0x06,0x3F,0x10,0x08,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0xC1,0x08,0x00,0x03,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
INT Ref_posi = 7;
INT Tlli_posi = 0;
INT Str_posi = 109;

E.7.2.2 GPRS_Attach.c modification method

E.7.2.2.1 Modification for contents of System Information Type13 and Attach Accept.

Please modify message contents as follows.
/* System Information Type13 for GPRS */
Before modification
It' s the 82nd to 84th line of GPRS_Attach.c attached to V3.20.
{0x01,0x06,0x00,0x80,0x00,0xD9,0x44,0x0C,0x00,0x00,0x87,
0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};

Please modify as bellows.
{0x01,0x06,0x00,0x80,0x00,0xD8,0x5A,0x0C,0x00,0x00,0x87,
0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2
B,0x2B};
/* GPRS Attach accept */
Before modification
It' s the 151st to 153rd line of GPRS_Attach.c attached to V3.20.
{0x08,0x02,0x01,0xE0,0x01,0x00,0xF1,0x10,0x00,0x01,0x01,
0x18,0x05,0xF4,0x08,0x76,0x54,0x32};
Please modify as bellows.
{0x08,0x02,0x01,0xE0,0x01,0x00,0xF1,0x10,0x00,0x01,0x03,
0x18,0x05,0xF4,0x08,0x76,0x54,0x32};

E.7.2.2.2 Add description of PSI13 transmission setting.

Please describe transmission setting of Packet System Information Type13 after System Information Type13 transmission. It's the 89th line of GPRS_Attach.c attached to V3.20.

Please add the following description.

```
/* Packet System Information Type13 Message Setup */
 {
      /* For PACCH */
         CGRR_MSGSET_DATA Msg =
         {0x70,0xDC,0x00,0x01,0xB0,0xB4,0x18,0x00,0x01,0x0B,
         0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,0x2B,
         0x2B,0x2B,0x2B};
         INT        idle_cycle    = 0;
         INT        tran_cycle    = 14; /* transmit every 14s in TBF TS04.60 5.5.2.1.3*/
         INT        reserve        = 0;

         GsmRrMsgSet(PSI_TYPE13, (CGRR_MSGSET_DATA *)Msg.CTRL_MSG,
         idle_cycle, tran_cycle, reserve, NO_TIMEOUT);
}
```

E.7.3 Scenario modification method(Sample scenarios attached to V5.11 or before)

Sample scenarios attached to V5.11 or before are required to be modified the following one scenarios

:GPRS_PBCCH_Idle_Loca_Attach_PDP.c

E7.3.1 Modification of GPRS_PBCCH_Idle_Loca_Attach_PDP.c

E.7.3.1.1 Modification of PRACH received function

Please modify to receive PRACH. It's the 610th to 630th line of GPRS_PBCCH_Idle_Loca_Attach_PDP.c attached to V5.11 or before.

Before modification

```
/* Receive Message: CHANNEL REQUEST        */
{
      INT       BtsNo;
      INT       Frame;
      INT       Lo_Ch;
      INT       Lo_No;
      UCHAR       RcvData[255];
      UCHAR     ReqRef[3];

      for       (;;){
      RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData,
      NO_TIMEOUT );
            if ((Lo_Ch == GSMU_RACH) &&
               (Frame == GSM_DL_RANDACCESS_IND) &&
                ((RcvData[0] & 0xE0) == 0x00)){      /* Location updating */
                 memcpy(ReqRef, RcvData, 3);
                 Loca(ReqRef);/* go to Location Updating procedure */
                 break;
            }
            if ((Lo_Ch == GSMU_PRACH) &&
                (Frame == GSM_RLC_RANDACCESS_IND)){
                 break; /* go to Attach procedure */
            }
      }
}
```

Please modify as bellows.

```
/* Receive Message: CHANNEL REQUEST        */
{
      INT         BtsNo;
      INT         Frame;
      INT         Lo_Ch;
      INT         Lo_No;
      INT         RcvFN;
      UCHAR     RcvData[255];
      UCHAR     ReqRef[3];

      for       (;;){
            RcvMessage( &BtsNo, &Frame, &Lo_Ch, &Lo_No, RcvData, 100 );
            if((Lo_Ch == GSMU_RACH) &&
                (Frame == GSM_DL_RANDACCESS_IND) &&
                 ((RcvData[0] & 0xE0) == 0x00)){      /* Location updating */
                 memcpy(ReqRef, RcvData, 3);
                 Loca(ReqRef);/* go to Location Updating procedure */
                 break;
            }
            GsmRcvControl( &Lo_Ch, &RcvFN, RcvData, 100 );
            if (Lo_Ch == GSMU_PRACH){
                 break; /* go to Attach procedure */
```

```
            }
        }
    }
```

F. ROUTER CONNECTION

This appendix is description about the function for connecting the Ethernet port (TE interface) of MD8480 A/B to a router by the environment of Fixation IP.

F.1 Outline of specification

In old MD8480 A/B, when a mobile station accessed the server PC on a real network (LAN) in Packet, it was able to access only the server PC belonging to the same subnetwork (segment) as a mobile station. (Figure 2-1) Since it becomes connectable with a router by adding this function, the communication which exceeded the subnetwork in real network environment is realizable. (Figure 2-2)

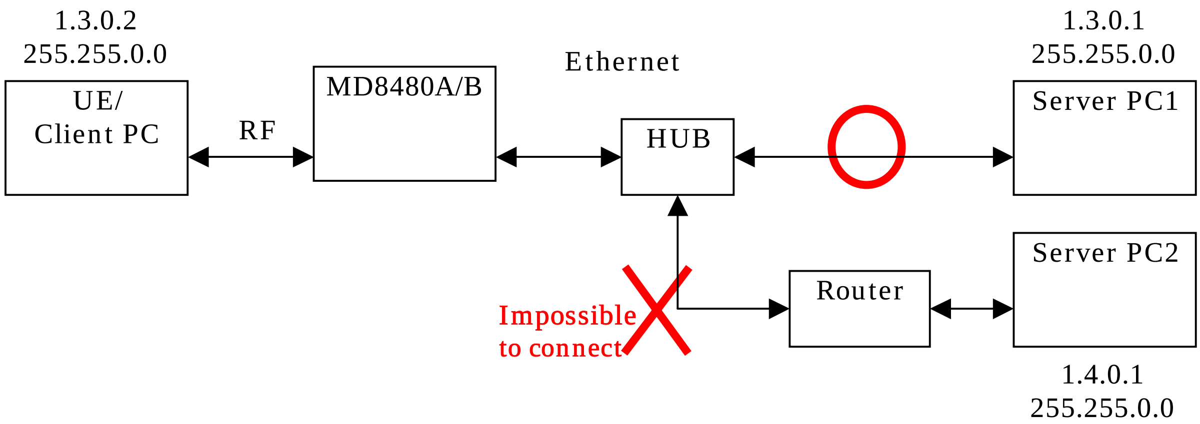


Figure 2-1 Packet communication outline figure before adding function

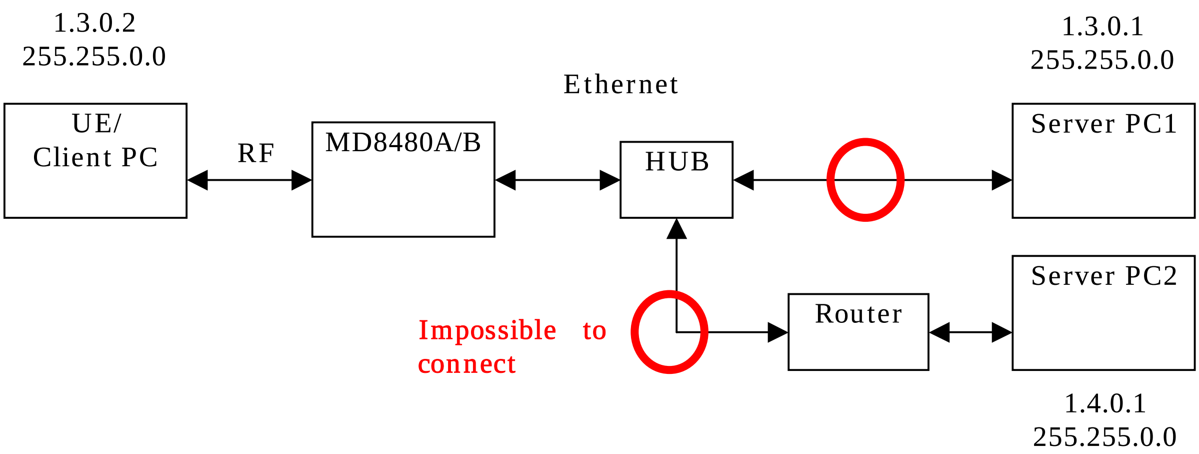


Figure 2-2) Packet communication outline figure after adding function

F.2 Setting method

When using this function, it' s necessary to set up IP address of router and subnet mask.

The above-mentioned address is set up in the portion before executing the SimulatorStart() function in a scenario.

This setting becomes effective when the following TE Type was specified with CteConfig() function in sequence after executing SimulatorStart( ) function.

(1) TE_TYPE_PPPSERVER (Built-in PPP server)

(2) TE_TYPE_IPPACKET (IP Packet)

Example of scenario description indicates the following.

EX 1) If TE Type is TE_TYPE_PPPSERVER

PPP server IP address	: 1.4.0.1
PPP client IP address	: 1.4.0.2
Router IP address	: 1.4.0.3

Subnet mask of the segment containing mobile station : 255.255.0.0

In case of setting as above, it' s mentioned in scenario as follows.

Flag is effective this function.

```
CteStartPar.RtConnPar.RtAddrType = RTADDR_TYPE_FIXED_IP;

CteStartPar.RtConnPar.RouterAddr = 0x01040003;      /* Router IP Address      */
CteStartPar.RtConnPar.SubnetMask = 0xFFFF0000;      /* Subnet Mask            */
CteStartPar.PppSvPar.IpcpPar.BtsIP = 0x01040001;    /* PPP server IP Address */
CteStartPar.PppSvPar.IpcpPar.MsIP = 0x01040002;     /* PPP client IP Address  */

SimulatorStart(0,NO_TIMEOUT);
```

Router IP address
Subnet mask
PPP server IP address
PPP client IP address
are set up.
Setting of address value is described by sexadecimal number.

*) Setting of PPP server IP address and client IP address is also described in the signaling tester which understands well C.6.6 PPP option setting.

EX2) If TE Type is TE_TYPE_IPPACKET

Mobile station side IP address :1.3.0.2

Router IP address :1.3.0.3

Subnet mask of the segment including mobile station: 255.255.0.0

In case of setting above, it' s mentioned in scenario as follows.

Sets up mobile station IP address.
Setting of address value is described by sexadecimal number..

Flag is effective this function.

```
CteStartPar.IppacketPar.MsIpv4Addr = 0x01030002;
CteStartPar.RtConnPar.RtAddrType = RTADDR_TYPE_FIXED_IP;
CteStartPar.RtConnPar.RouterAddr = 0x01030003;      /* Router IP Address */
CteStartPar.RtConnPar.SubnetMask = 0xFFFF0000;      /* Subnet Mask        */

SimulatorStart(0,NO_TIMEOUT);
```

Sets up router address and subnet mask.

Setting of address value is described by sexadecimal number.

F.3 Note item and Restriction item

F.3.1 Note item

1) This function corresponds when using radio system either W-CDMA or GPRS(OPTION).
2) When using this function, please use after connecting router.
3) In case of implementing Web access from mobile station side with using this function, please use it after setting up each setting (setup of a proxy etc.) for implementing Internet access according to a customer’ s LAN environment.

F.3.2 Restriction item

1) Address of router and subnet mask which can set up by one MD8480A/B is only one kind.
2) It does not correspond from LAN of the DHCP environment to address automatic acquisition.
3) Address of setup router isn’ t released as long as you push STOP button in PC software.

Support

If you have a question or some unknown points, please ask to the MD8480 support address [MD8480A-G-support@zy.anritsu.co.jp].

If you have the contract of MD8480 Support service option, Please ask to the address for contracted customer.

* When an inquiry is send to us, Please let us know the following information.

(1) Agreement No (If you have the contract of Support Service option)

(2) Model (MD8480A, MD8480B, MD8480C)

(3) Version (PC, FW, FPGA, ISDN, TDMA)

If you are using later version then v5.40, the information can be outputted as a text file. (Information.txt) Please send us the file.(the detailed procedure is explained in [How to output the Information file] below.)

(4) System (W-CDMA, GSM, HSDPA)

(5) Inquiry Type (Question, report of Bug, requirement of modification,

requirement to investigate whether UE or MD8480, etc.)

(6) Log files, Scenarios (If they are necessary)

How to output the Information file

If the version later than v5.40 is used, the information (Control PC, FW, FPGA, ISDN, TDMA and software options) can be outputted as a text file. (Information.txt)

Procedure

(1) Open the System window by pushing [SYSTEM] button on the main window.

(2) Enter the serial No of the unit, then push [Information Read] button.

(3) The file: information.txt is generated in C:¥Mx848000¥.

About expansion of Option key

Anritsu has been supplied various options of MD8480A/B signalling tester. This time, Anritsu expands option key because of increase of options. Anritsu has started to ship expanded option key in order from June 2005.

When you receive new option key, please remind points below.

1) In case of the option key has version 5.50 or later:

* Character number of option key is 19. (Including hyphen)

(example)(X:alphabet) XXXX-XXXX-XXXX-XXXX

* When downloading option key, please use option key download software that is included in FPGA version V5.30 or later. The file name is "OptionKey0**.zip"("**" is 11 or larger.)

* When confirming download of option key completes correctly, please download firmware V5.30 or later in MD8480A/B and use PC V5.30 or later. (If you use former version firmware, downloaded options aren't displayed correctly.)

2) In case of the option key doesn't have version 5.50 or later:

* Character number of option key is 11. There is no change. (Including hyphen)

(example)(X:alphabet) XXXXX-XXXXX

* When downloading option key, you can use option key download software of any version.

* When confirming download of option key completes correctly, you can use firmware and control software of any version.

3) In support service of MD8480A/B, option key is issued, which has 2 more versions than latest version. (Example : If latest version is V5.30, the option key has all of from V5.00 to V5.50.)