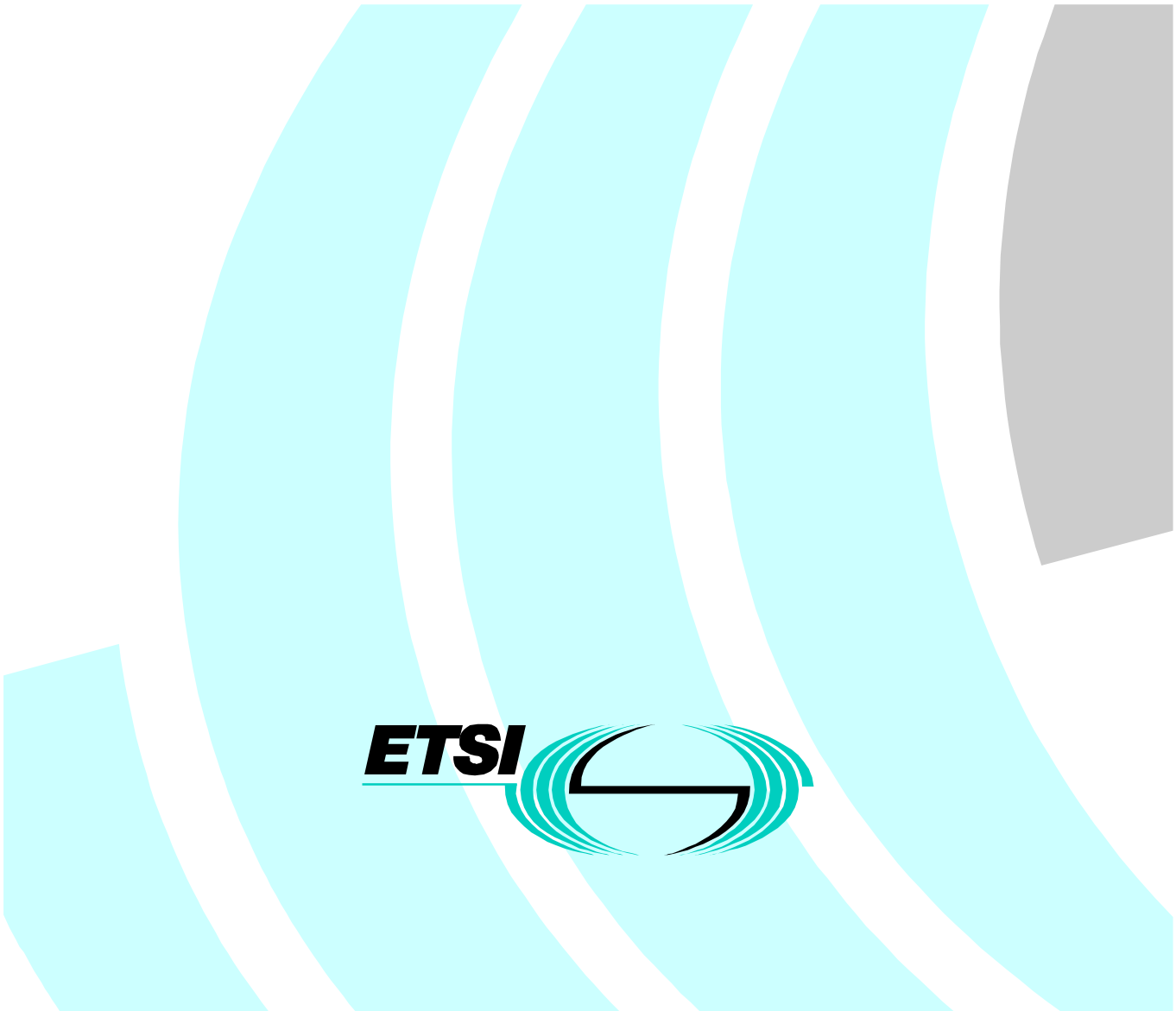
ETSI TS 101 761-2 V1.2.1 (2001-04)

Technical Specification

Broadband Radio Access Networks (BRAN);
HIPERLAN Type 2;
Data Link Control (DLC) Layer;
Part 2: Radio Link Control (RLC) sublayer



Reference
RTS/BRAN-0020004-2R1

Keywords
access, broadband, HIPERLAN, layer 2, radio

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights 8
Foreword 8
Introduction 8
1 Scope 9
2 References 9
3 Definitions, symbols and abbreviations 10
3.1 Definitions 10
3.2 Symbols 12
3.3 Abbreviations 12
4 Overview 14
4.1 BRAN HIPERLAN/2 Protocol Stack 14
4.2 Functional Entities of RLC sublayer 15
4.3 Usage of DLC logical and transport channels for RLC messages 15
4.4 Transfer Syntax of RLC 16
4.5 Sequencing of RLC messages 17
4.6 Message Sequence Charts of RLC 17
4.7 Abstract Syntax Notation one of the RLC PDUs 17
4.8 The text style of RLC PDUs and parameters 17
4.9 Mandatory and Optional functions 18
4.10 Message transmission not allowed 18
5 RLC Services 18
5.1 Services supporting ACF (Association Control Function) 18
5.1.1 Association 18
5.1.1.1 RBCH Association 20
5.1.1.2 MAC ID Assignment 22
5.1.1.3 Link Capability 23
5.1.1.4 Encryption startup 25
5.1.1.5 Authentication 26
5.1.1.5.1 General 26
5.1.1.5.2 Authentication procedures 26
5.1.1.5.3 Authentication key identifier alternatives 27
5.1.1.5.3.1 General 27
5.1.1.5.3.2 IEEE address as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1) 27
5.1.1.5.3.3 Extended IEEE address as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1) 28
5.1.1.5.3.4 Network access identifier as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1) 29
5.1.1.5.3.5 Distinguished name as on authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1) 30
5.1.1.5.3.6 Compressed type as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1) 31
5.1.1.5.3.7 Generic type as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1) 32
5.1.1.6 Authentication based on different key types 33
5.1.1.6.1 Authentication with pre-shared key 33
5.1.1.6.2 Authentication based on 512 bit RSA signature (OAP/OMT) 35
5.1.1.6.3 Authentication based on 768 bit RSA signature (OAP/OMT) 36
5.1.1.6.4 Authentication based on 1 024 bit RSA signature (OAP/OMT) 38
5.1.1.7 DM Common Key Distribution (OAP/OMT) 40
5.1.1.8 Info Transfer procedure (OAP/OMT, depends on CL and DLC) 41
5.1.2 Key Management 43
5.1.2.1 General 43
5.1.2.2 Unicast Key Refresh (OAP) 43
5.1.2.3 Common keys (OAP/OMT) 45

5.1.2.3.1 General 45
5.1.2.3.2 DM Common Key Distribution (OAP/OMT) 45
5.1.2.3.3 Common Key Refresh (OAP) 46
5.1.2.4 RBCH-Seed Transfer 47
5.1.2.5 Encryption key calculations 48
5.1.2.5.1 Diffie-Hellman Key Exchange 48
5.1.2.5.2 DES Key Calculation 48
5.1.2.5.3 Calculation of Triple DES keys (OAP/OMT) 49
5.1.2.5.4 Unicast Key Generation at Network Handover (Mandatory if Handover supported) 49
5.1.2.6 Authentication functions 50
5.1.2.6.1 Algorithms 50
5.1.2.6.2 Authentication protocols 50
5.1.2.6.3 Pre-shared key based authentication 51
5.1.2.6.4 Public key based authentication (OAP/OMT) 52
5.1.3 Disassociation 52
5.1.4 Multicast (OAP/OMT) 54
5.1.5 CL Broadcast (OAP/OMT, depends on CL) 56
5.1.6 Association Rejection 58
5.2 Services supporting RRC (Radio Resource Control) 59
5.2.1 Handover (OAP/OMT) 59
5.2.1.1 Sector Handover (OAP/OMT) 60
5.2.1.2 Radio (intra-AP) Handover (OAP/OMT) 61
5.2.1.3 Network Handover (OAP/OMT) 64
5.2.1.4 Token distribution for Network Handover 71
5.2.1.5 Handover Rejection 73
5.2.1.6 Forced Handover (AP initiated handover) (OAP/OMT) 73
5.2.2 Dynamic Frequency Selection 74
5.2.2.1 Introduction to DFS 74
5.2.2.2 DFS algorithm 75
5.2.2.3 DFS protocol 75
5.2.2.4 DFS measurements 76
5.2.2.4.1 AP Measurement Procedure 76
5.2.2.4.2 MT Measurement Procedures 77
5.2.2.4.2.1 Measurements on other frequencies 80
5.2.2.4.2.2 Measurements on used frequency 81
5.2.2.4.3 MT measurement processing 83
5.2.2.4.3.1 Measurement processing (field strength measurement) 83
5.2.2.4.3.2 Measurement processing (field strength measurement and BCH decoding) 83
5.2.2.4.3.3 Indication of measurement time 84
5.2.2.4.4 Calculation of RSS statistics (informative) 84
5.2.2.5 DFS measurement reports 85
5.2.2.6 Change Frequency 88
5.2.3 Transmission Power Control 89
5.2.3.1 Uplink power control 89
5.2.3.2 Downlink Power Control 90
5.2.3.3 Direct Link Power Control (OAP/OMT) 90
5.2.4 MT Alive 90
5.2.5 MT Absence (OAP/OMT) 92
5.2.6 Power Saving (OMT) 94
5.2.6.1 General 94
5.2.6.2 MT sleep request procedure 96
5.2.6.3 AP Procedure 98
5.2.6.3.1 AP Procedure for unicast data 98
5.2.6.3.2 AP Procedure for broadcast data 98
5.2.6.4 MT Procedure 99
5.3 Services supporting DUCC (DLC User Connection Control) 99
5.3.1 Unicast DUC Setup 100
5.3.1.1 AP Initiated DUC Setup (OAP/OMT) 100
5.3.1.2 MT initiated DUC Setup 102
5.3.2 Unicast DUC release 104
5.3.2.1 AP Initiated DUC Release 104
5.3.2.2 MT Initiated DUC release 105

5.3.3 Unicast DUC modify (OAP/OMT) 105
5.3.3.1 AP Initiated DUC modify 106
5.3.3.2 MT Initiated DUC modify 108
5.3.4 Unicast DUC Reset 109
5.3.4.1 AP Initiated DUC Reset 110
5.3.4.2 MT Initiated DUC reset 111
5.3.5 Multicast DUC 112
5.3.6 Broadcast DUC 112
5.3.7 Unicast Direct Link DUC Setup (OAP/OMT) 112
5.3.7.1 AP/CC initiated DM DUC Setup (OMT) 113
5.3.7.2 MT initiated DM DUC Setup (OMT) 117
5.3.7.3 DM DUC Relay Setup (OMT) 119
5.3.8 Unicast Direct Link DUC Release 122
5.3.8.1 AP/CC initiated DM DUC Release 122
5.3.8.2 MT initiated DM DUC Release 124
5.3.8.3 DM DUC Relay Release 125
5.3.9 Unicast Direct Link DUC Modify 127
5.3.9.1 AP/CC initiated DM DUC Modify 127
5.3.9.2 MT initiated DM DUC Modify 131
5.3.9.3 DM DUC Relay Modify 133
5.3.10 Unicast Direct Link DUC Reset 135
5.3.10.1 AP/CC initiated DM DUC Reset 135
5.3.10.2 MT initiated DM DUC Reset 136
5.3.11 Multicast Direct Link 137
5.3.12 Broadcast Direct Link 137

6 Timers and repetitions of RLC messages 137

7 PDU for unsupported messages 139

8 Primitives 139
8.1 Primitive types 139
8.2 Primitives to the Convergence Layer, DLC C-SAP 140

Annex A (normative): PDU type and Transfer Syntax Tables 141

A.1 RLC PDU type 141
A.1.1 LCH RLC PDU type 142
A.1.2 SCH RLC PDU type 144

A.2 Transfer Syntax Tables for LCH ACF messages 145
A.2.1 RLC-RBCH-ASSOCIATION encoding 145
A.2.2 RLC-LINK-CAPABILITY encoding 145
A.2.3 RLC-LINK-CAPABILITY-ACK encoding 146
A.2.4 RLC-KEY-EXCHANGE-MT-1 encoding 146
A.2.5 RLC-KEY-EXCHANGE-MT-2 encoding 146
A.2.6 RLC-KEY-EXCHANGE-AP-1 encoding 146
A.2.7 RLC-KEY-EXCHANGE-AP-2 encoding 146
A.2.8 RLC-AUTHENTICATION encoding 147
A.2.9 RLC-AUTHENTICATION-MT encoding 147
A.2.10 RLC-AUTHENTICATION-AP-1 encoding 147
A.2.11 RLC-AUTHENTICATION-AP-2 encoding 147
A.2.12 RLC-AUTHENTICATION-AP-3 encoding 148
A.2.13 RLC-AUTHENTICATION-ACK-1 encoding 148
A.2.14 RLC-AUTHENTICATION-ACK-2 encoding 148
A.2.15 RLC-AUTHENTICATION-ACK-3 encoding 148
A.2.16 RLC-DM-COMMON-KEY-DISTR encoding (OAP/OMT) 149
A.2.17 RLC-DM-COMMON-KEY-DISTR-ACK encoding (OAP/OMT) 149
A.2.18 RLC-INFO encoding (OAP/OMT) 149
A.2.19 RLC-INFO-ACK encoding (OAP/OMT) 149
A.2.20 RLC-UNICAST-KEY-REFRESH encoding (OAP) 150
A.2.21 RLC-UNICAST-KEY-REFRESH-ACK encoding (OAP) 150
A.2.22 RLC-COMMON-KEY-REFRESH encoding (OAP) 150
A.2.23 RLC-COMMON-KEY-REFRESH-ACK encoding (OAP) 150

A.2.24 RLC-GROUP-JOIN encoding (OAP/OMT) 151
A.2.25 RLC-GROUP-JOIN-ACK encoding (OAP/OMT) 151
A.2.26 RLC-GROUP-JOIN-NACK encoding (OAP/OMT) 151
A.2.27 RLC-GROUP-LEAVE encoding (OAP/OMT) 151
A.2.28 RLC-GROUP-LEAVE-ACK encoding (OAP/OMT) 152
A.2.29 RLC-CL-BROADCAST-JOIN encoding (OAP/OMT) 152
A.2.30 RLC-CL-BROADCAST-JOIN-ACK encoding (OAP/OMT) 152
A.2.31 RLC-CL-BROADCAST-LEAVE encoding (OAP/OMT) 153
A.2.32 RLC-CL-BROADCAST-LEAVE-ACK encoding (OAP/OMT) 153

A.3 Transfer Syntax Tables for LCH RRC messages 154
A.3.1 RLC-RADIO-HANDOVER-COMPLETE encoding (OAP/OMT) 154
A.3.2 RLC-HANDOVER-ASSOCIATION encoding (OAP/OMT) 155
A.3.3 RLC-HANDOVER-LINK-CAPABILITY-ACK encoding (OAP/OMT) 155
A.3.4 RLC-NW-SIGNALLING-HANDOVER encoding (OAP/OMT) 156
A.3.5 RLC-NW-SIGNALLING-HANDOVER-ACK encoding (OAP/OMT) 156
A.3.6 RLC-NETWORK-HANDOVER-COMPLETE encoding (OAP/OMT) 157
A.3.7 RLC-HO-INFO-DISTRIBUTION encoding (OAP/OMT) 158
A.3.8 RLC-DFS-MEASUREMENT-COMPLETE-REQUEST encoding 158
A.3.9 RLC-DFS-MEASUREMENT-PERCENTILES-REQUEST encoding 158
A.3.10 RLC-DFS-MEASUREMENT-SHORT-REQUEST encoding 159
A.3.11 RLC-DFS-REPORT-COMPLETE encoding 159
A.3.12 RLC-DFS-REPORT-PERCENTILES encoding 160
A.3.13 RLC-DFS-REPORT-SHORT encoding 160

A.4 Transfer Syntax Tables for LCH DUCC messages 161
A.4.1 RLC-SETUP encoding 161
A.4.2 RLC-CONNECT encoding 162
A.4.3 RLC-CONNECT-ACK encoding 163
A.4.4 RLC-RELEASE encoding 164
A.4.5 RLC-RELEASE-ACK encoding 164
A.4.6 RLC-MODIFY-REQUEST encoding (OAP/OMT) 164
A.4.7 RLC-MODIFY encoding (OAP/OMT) 166
A.4.8 RLC-MODIFY-ACK encoding (OAP/OMT) 167
A.4.9 RLC-RESET, RLC-RESET-ACK encoding 167
A.4.10 RLC-DM-SETUP encoding (OAP/OMT) 168
A.4.11 RLC-DM-CONNECT encoding (OAP/OMT) 169
A.4.12 RLC-DM-CONNECT-ACK encoding (OAP/OMT) 169
A.4.13 RLC-DM-CONNECT-COMPLETE encoding (OAP/OMT) 170
A.4.14 RLC-DM-RELAY-SETUP encoding (OAP/OMT) 170
A.4.15 RLC-DM-RELAY-SETUP-ACK encoding (OAP/OMT) 171
A.4.16 RLC-DM-MODIFY-REQ encoding (OAP/OMT) 171
A.4.17 RLC-DM-MODIFY encoding (OAP/OMT) 172
A.4.18 RLC-DM-MODIFY-ACK encoding (OAP/OMT) 172
A.4.19 RLC-DM-MODIFY-COMPLETE encoding (OAP/OMT) 172
A.4.20 RLC-DM-RELAY-MODIFY encoding (OAP/OMT) 173
A.4.21 RLC-DM-RELAY-MODIFY-ACK encoding (OAP/OMT) 173
A.4.22 RLC-DM-RELEASE encoding (OAP/OMT) 174
A.4.23 RLC-DM-RELEASE-ACK encoding (OAP/OMT) 174
A.4.24 RLC-DM-RELAY-RELEASE encoding (OAP/OMT) 174
A.4.25 RLC-DM-RELAY-RELEASE-ACK encoding (OAP/OMT) 174
A.4.26 RLC-DM-RESET, RLC-DM-RESET-ACK encoding (OAP/OMT) 175

A.5 Transfer Syntax Tables for SCH ACF messages 175
A.5.1 RLC-RBCH-ASSOCIATION-REQUEST encoding (OMT) 175
A.5.2 RLC-MAC-ID-ASSIGN encoding 175
A.5.3 RLC-MAC-ID-ASSIGN-ACK encoding 175
A.5.4 RLC-MAC-ID-ASSIGN-NACK encoding 175
A.5.5 RLC- RLC-COMMON-KEY-ACTIVATE encoding (OAP) 176
A.5.6 RLC-DISASSOCIATION encoding 176
A.5.7 RLC-DISASSOCIATION-ACK encoding 176
A.5.8 RLC-PROCEEDING encoding 176
A.5.9 RLC-UNICAST-KEY-ACTIVATE encoding (OAP) 176

A.6 Transfer Syntax Tables for SCH RRC messages 177
A.6.1 RLC-SECTOR-HANDOVER-REQUEST encoding (OAP/OMT) 177
A.6.2 RLC-SECTOR-HANDOVER-ACK encoding (OAP/OMT) 177
A.6.3 RLC-HANDOVER-NOTIFY encoding (OAP/OMT) 177
A.6.4 RLC-HANDOVER-REQUEST encoding (OAP/OMT) 177
A.6.5 RLC-HANDOVER-REQUEST-NACK encoding (OAP/OMT) 177
A.6.6 RLC-HO-INFO-DISTRIBUTION-ACK encoding (OAP/OMT) 178
A.6.7 RLC-FORCE-HANDOVER encoding (OAP/OMT) 178
A.6.8 RLC-FORCE-HANDOVER-ACK encoding (OAP/OMT) 178
A.6.9 RLC-AP-ABSENCE encoding (OAP) 178
A.6.10 RLC-DFS-MT-INIT-REPORT-REQUEST encoding (OMT) 178
A.6.11 RLC-DFS-MT-INIT-REPORT-REQUEST-ACK encoding (OMT) 179
A.6.12 RLC-CHANGE-FREQUENCY encoding 179
A.6.13 RLC-UPLINK-PC-CALIBRATION encoding 179
A.6.14 RLC-MT-ALIVE-REQUEST encoding 179
A.6.15 RLC-MT-ALIVE-REQUEST-ACK encoding 179
A.6.16 RLC-MT-ALIVE encoding 179
A.6.17 RLC-MT-ALIVE-ACK encoding 180
A.6.18 RLC-MT-ABSENCE encoding (OAP/OMT) 180
A.6.19 RLC-MT-ABSENCE-ACK encoding (OAP/OMT) 180
A.6.20 RLC-SLEEP encoding (OMT) 180
A.6.21 RLC-SLEEP-ACK encoding (OMT) 180

A.7 Transfer Syntax Tables for SCH DUCC messages 181
A.7.1 RLC-DM-MODIFY-COMPLETE-ACK encoding (OAP/OMT) 181
A.7.2 RLC-DM-CONNECT-COMPLETE-ACK encoding (OAP/OMT) 181

A.8 Transfer Syntax Tables for other RLC SCH messages 181
A.8.1 RLC-NO-SUPPORT encoding 181

Annex B (normative): Types 182

Annex C (normative): RLC TIMERS 192

Annex D (normative): SDL specification of the RLC protocol 194

D.1 The SDL Graphical form (SDL/GR) 194

D.2 The SDL Textual format (SDL/PR) 194

D.3 The SDL Common Interchange format (SDL/CIF) 194

D.4 The ASN.1 files 194

D.5 The PDF format 194

Annex E (informative): Bibliography 195

History 196

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/ipr).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI Project Broadband Radio Access Networks (BRAN).

The present document is part 2 of a multi-part deliverable covering the HIPERLAN Type 2; Data Link Control (DLC) Layer, as identified below:

Part 1: "Basic Data Transport Functions";

Part 2: "Radio Link Control (RLC) sublayer";

Part 3: "Profile for Business Environment";

Part 4: "Extension for Home Environment";

Part 5: "Profile for Home Environment".

Introduction

HIPERLAN type 2 (HIPERLAN/2) is confined to the two lowest layers of the open systems interconnection (OSI) model, the physical and the data link control layer. TR 101 683 [20] contains an overall description of the HIPERLAN type 2 (HIPERLAN/2) system. The physical layer is described in TS 101 475 [4]. The interworking with higher layers is handled by convergence layers on top of the data link control layer. The Packet based Convergence Layer is described in TS 101 493 ([17], [19], [20]) and the Cell based Convergence Layer in TS 101 763 ([16], [17]).

Separate ETSI documents provide details on the system overview, physical layer, data link control layer, convergence sub-layers and conformance testing requirements for HIPERLAN/2.

1 Scope

The present document specifies the basic Radio Link Control (RLC) sublayer of HIPERLAN/2. The RLC sublayer is used to control radio association, resources and connection.The present document does not address the requirements and technical characteristics for conformance testing.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] Federal Information Processing Standard (FIPS) Publication 46-3 (1993): "Data Encryption Standard (DES)".

[2] US National Bureau of Standards "Guidelines for Implementing and Using the Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 74, April 1981, http://www.itl.nist.gov/div897/pubs/fip74.htm.

[3] US National Bureau of Standards "DES Modes of Operation", Federal Information Processing Standard (FIPS) Publication 81, December 1980, http://www.itl.nist.gov/div897/pubs/fip81.htm.

[4] ETSI TS 101 475 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Physical (PHY) Layer".

[5] ETSI TS 101 761-1: "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Data Link Control (DLC) Layer; Part 1: Basic Data Transport Functions".

[6] RFC 1321: "The MD5 Message Digest Algorithm".

[7] RFC 2104: "HMAC: Keyed-Hashing for Message Authentication".

[8] IEEE 802.3 (2000): "Information technology - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications".

[9] IEEE Std 1394 (1995): "IEEE Standard for a High Performance Serial Bus".

[10] RFC 2486: "The Network Access Identifier".

[11] ITU-T Recommendation X.509 (1993): "Information technology - Open Systems Interconnection - The Directory: Authentication framework".

[12] CEPT ERC/DEC/(99)23: "ERC Decision of 29 November 1999 on the harmonized frequency bands to be designated for the introduction of High Performance Radio Local Area Networks (HIPERLANs)".

[13] ETSI TS 101 761-3 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Data Link Control (DLC) Layer; Part 3: Profile for Business Environment".

[14] ETSI TS 101 761-5 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Data Link Control (DLC) Layer; Part 5: Profile for Home Environment".

[15] ETSI TS 101 763-1 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Cell based Convergence Layer; Part 1: Common Part".

[16] ETSI TS 101 763-2 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Cell based Convergence Layer; Part 2: UNI Service Specific Convergence Sublayer (SSCS)".

[17] ETSI TS 101 493-1 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Packet based Convergence Layer; Part 1: Common Part".

[18] ETSI TS 101 493-2 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Packet based Convergence Layer; Part 2: Ethernet Service Specific Convergence Sublayer (SSCS)".

[19] ETSI TS 101 493-3 (V1.1.1): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Packet based Convergence Layer; Part 3: IEEE 1394 Service Specific Convergence Sublayer (SSCS)".

[20] ETSI TR 101 683 (V1.1.2): "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; System Overview".

[21] ETSI TR 101 031: "Broadband Radio Access Networks (BRAN); HIgh PErformance Radio Local Area Network (HIPERLAN) Type 2; Requirements and architectures for wireless broadband access".

[22] "New Directions in Cryptography", W. Diffie and M. E. Hellman, IEEE Transactions on Information Theory, v. IT-22, n. 6, pp. 644-654, November 1976.

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

Access Point (AP): device that is responsible for the centralized control of the resources in a radio cell
It is usually connected to a fixed network.

association: process where an MT gets a MAC-ID from an AP
A Dedicated Control CHannel (DCCH) is established. Basic link layer parameters are agreed on between an MT and an AP. Encryption start up and authentication are done, if the use of them is decided on for this association. Information between convergence layers in the MT and the AP may be transferred.

Association Control Function: group of control functions that use the services of the RLC
These functions are responsible for the handling of the association between MT and AP.

ASsociation control CHannel: logical channel in the uplink that conveys new association request messages

authentication: corroboration that a peer entity in an association is the one claimed

Broadcast CHannel: transport channel that broadcasts control information

Broadcast Control CHannel: logical channel that broadcasts control information which is relevant for the current MAC frame

Broadcast frame: MAC frame sent at regular intervals, wherein all MTs in a cell are listening, sleeping ones included

Broadcast phase: part of a MAC Frame in which the AP sends pure control signalling which can be received by any MT in the range of the AP
The Broadcast phase consists of Broadcast Control Channel, Frame Control Channel and Access Feedback CHannel.

Central Controller: provides control functionality equivalent to that of an access point but is not necessarily attached to a fixed network
This term is normally used if central controller and MT functionality are located in a single device. It mostly involves direct mode communication.

Centralized Mode: in centralized mode, all data transmitted or received by a mobile terminal shall pass the access point or the centralized controller, even if the data exchange is between mobile terminals associated to the same access point or centralized controller

DLC connection: HIPERLAN/2 DLC operates connection oriented
A DLC connection carries user or control data and is identified by a DLC connection identifier. A connection has a set of properties for the transfer of data agreed upon between the MT and the AP or between MTs and a CC.

DLC User Connection: DLC user connection is uniquely identified by the DLC connection ID and a MAC-ID

DLC User Connection Control: group of control functions that uses the services of the RLC
It is responsible for the handling of DLC user connections.

DLC TS: TS 101 761-1

Direct Mode: data exchange between MTs associated with the same AP or CC takes place without passing but under control of the access point or the central controller

Direct link phase: part of a MAC frame that only contains the data exchanged directly between MTs using direct mode communication methods

Downlink phase: part of the Downlink transmission of a MAC Frame during which user and control data is transmitted from the access point or central controller to mobile terminals
The data transmitted can be user as well as control data in unicast, broadcast and multicast modes.

Encryption Function: function that is responsible for keeping user data and part of RLC signalling secret between HIPERLAN/2 devices

Error control: error control is responsible for detection of transmission errors and, where appropriate, for the correction of errors

Forward-Handover: handover where MT may not inform the old AP/APT about its intention to change to another AP/APT

Frame CHannel: transport channel that is broadcast and which carries the frame control channel

Frame Control CHannel: logical channel that contains the information defining how the resources are allocated in the current MAC frame
Its content changes in general dynamically from frame to frame.

Logical channel: generic term for any distinct data path
A set of logical channel types is defined for different kinds of data transfer services. Each logical channel type is defined by the type of information it carries. Logical channels can be considered to operate between logical connection end points.

MAC Frame: periodical structure in time that appears on the air interface and that determines the communication of HIPERLAN/2 devices

Mobile Terminal: device that communicates with an access point or with each other via a radio link
It is typically a user terminal.

Multicast: function that makes it possible for a group of MTs associated to an AP to receive the same information

PHY mode: PHY mode corresponds to a signal constellation (Modulation alphabet) and a code rate combination

Radio cell: radio cell is the area covered by an access point or central controller
It is sometimes used as a term to describe an AP or CC and its associated terminals.

Radio Link Control sublayer: control plane of the DLC which offers transport services for the radio resource control, association control function and the DLC user connection control

Radio Resource Control: group of control functions that use the services of the RLC
It controls the handling of radio resources.

Random Access CHannel: logical channel in the uplink of the MAC frame in which the MTs can send signalling data for the DLC or the RLC
It is transported in the random channel.

Random access Feedback CHannel: logical channel where the result of the access attempts to the random channel made in the previous MAC frame is conveyed

Random CHannel: transport channel in the uplink of the MAC that carries the logical channels random access channel and association control channel
A contention scheme is applied to access it.

Random Access Phase: period of the MAC Frame where any MT can try to access the system
The access to this phase is based on a contention scheme.

Resource Grant: allocation of transmission resources by an access point or a central controller

Resource Request: message from a terminal to an access point or central controller in which the current buffer status is conveyed to request for transmission opportunities in the uplink or direct link phase

Sector antenna: term is used to describe if an access point or central controller uses one or more antenna element

Transport channel: basic element to construct PDU trains
Transport channel describes the message format.

Uplink phase: part of the MAC frame in which data is transmitted from mobile terminals to an access point or a central controller

3.2 Symbols

For the purposes of the present document, the following symbols apply:

RxBoW	Receivers BoW
TxBoW	Transmitters BoW
|	concatenation of parameters

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ACF	Association Control Function
ACH	Access feedback CHannel
AP ID	Access Point IDentifier
AP	Access Point
APC	Access Point Controller
APT	Access Point Transceiver
ARP	Antenna Reference Point
ARQ	Automatic Repeat Request
BCH	Broadcast CHannel
CC	Central Controller
CL	Convergence Layer
DCC	DLC user Connection Control
DCCH	Dedicated Control CHannel
DES	Data Encryption Standard
DFS	Dynamic Frequency Selection
DiL	Direct Link
DL	DownLink
DLC	Data Link Control
DM	Direct Mode
DUC	DLC User Connection
EC	Error Control
FCA	Fixed Capacity Agreement
FCCH	Frame Control CHannel

FCH	Frame CHannel
FSA	Fixed Slot Allocation
HL/2	HIPERLAN type 2
HE	Home Extension
HMAC	Hash based Message Authentication Code
HMSC	High level MSC
IV	Initialization Vector
LCH	Long transport CHannel
MAC ID	MAC IDentifier
MAC	Medium Access Control
MD5	Message Digest #5
MSB	Most Significant Bit
MSC	Message Sequence Chart
MT	Mobile Terminal
NAI	Network Access Identifier
NET ID	NETwork IDentifier
NOP ID	Network Operator IDentifier
NW	NetWork
OAP	Optional in the AP
OMT	Optional in the MT
PDU	Protocol Data Unit
PHY	PHYsical layer
PR	Present
RBCH	RLC Broadcast CHannel
RLC	Radio Link Control
RRC	Radio Resource Control
RSS	Received Signal Strength
SAP	Service Access Point
SCH	Short transport CHannel
SSK	Session Secret Key
TS	Technical Specification
UL	UpLink
UOA	Use Omni Antenna

4 Overview

The present document describes the basic RLC functions for the Association, Radio Resource Control and DLC User Connection Control between HIPERLAN/2 devices. It consists of functions and message formats for the AP and MT side. An overview of HIPERLAN/2 is given in TR 101 683 [20] and detailed description of the DLC basic transport functions is given in TS 101 761-1 [5]. It is recommended that TR 101 683 [20] and TS 101 761-1 [5] have been read before reading the present document.

4.1 BRAN HIPERLAN/2 Protocol Stack

The HIPERLAN/2 basic protocol stack on the AP side and its functions are shown in figure 1. It consists of the PHY layer on the bottom, the DLC layer in the middle (including RLC sublayer) and one or more convergence layers on top. The scope of HIPERLAN/2 standard end at the upper end of the CL on top of which higher layers are located.

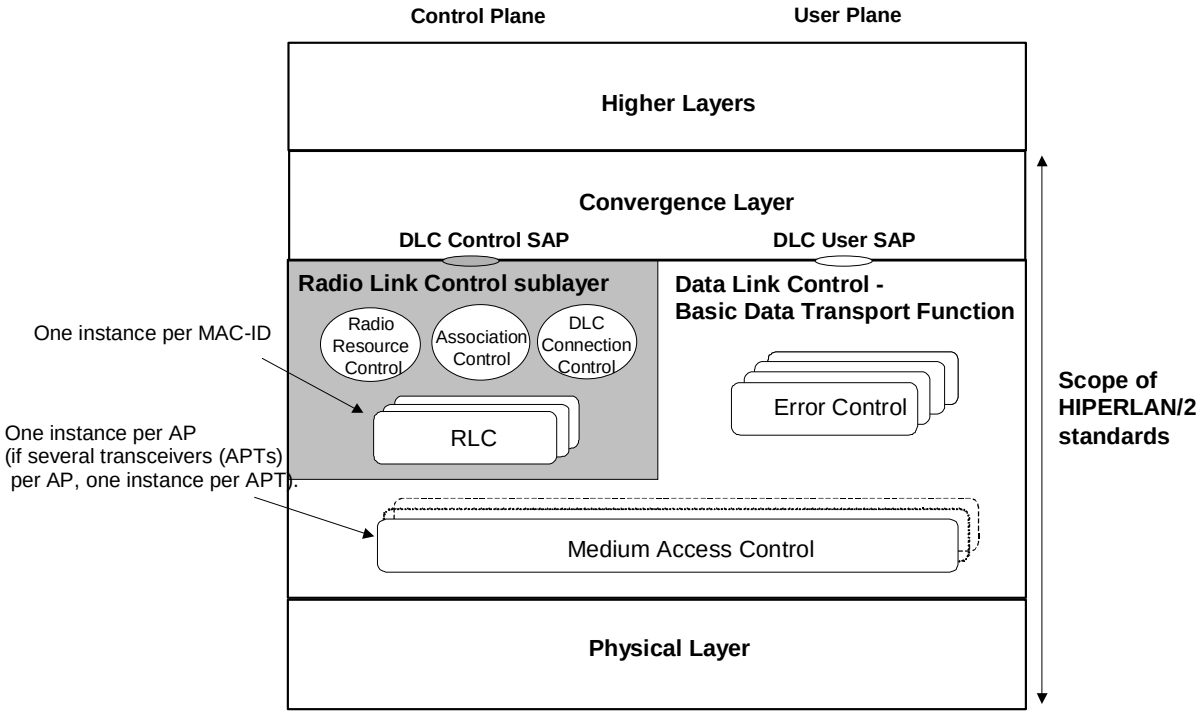


Figure 1: BRAN HIPERLAN/2 protocol stack in the AP/CC

The HIPERLAN/2 basic protocol stack on the MT side and its functions are depicted in figure 2. The difference to the model of the AP is that it contains only one RLC entity.

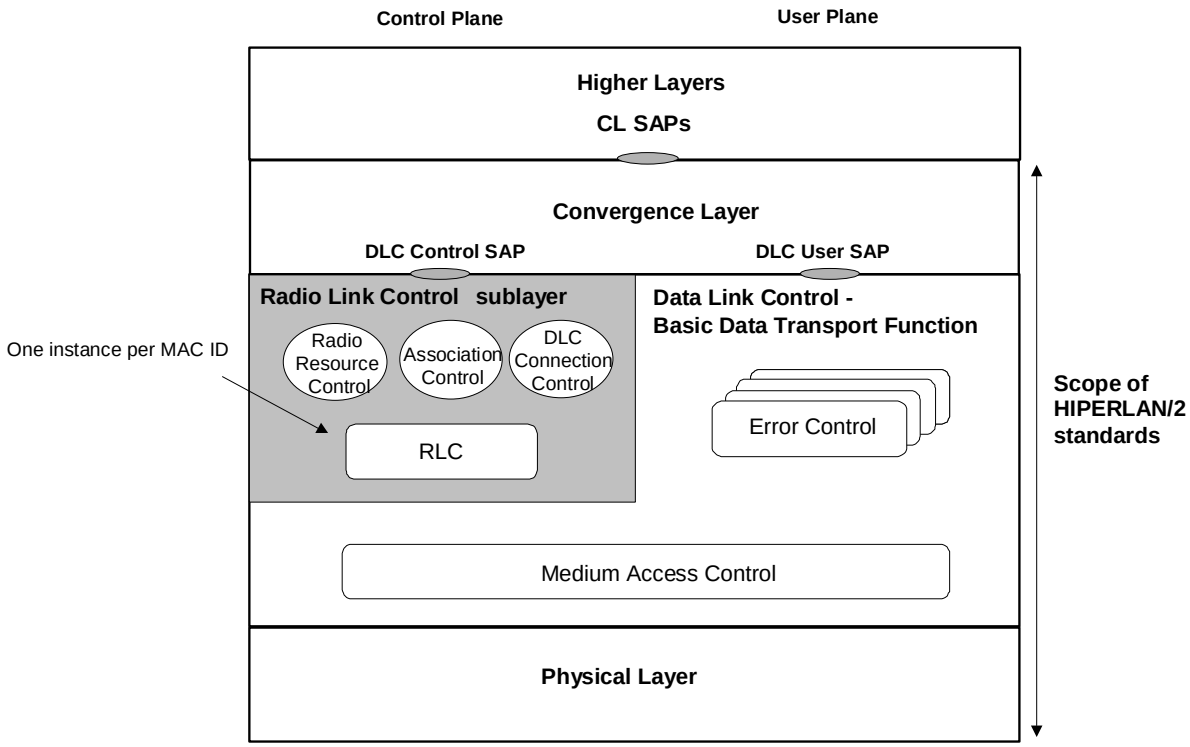


Figure 2: BRAN HIPERLAN/2 protocol stack in the MT

The present document is confined to the definition of the highlighted part shaded in grey, the RLC sublayer. It describes mainly the RLC messages and their format.

4.2 Functional Entities of RLC sublayer

The division of functional entities of RLC sublayer is informative. From the perspective of functionality they have been divided to four groups in the present document:

- Association Control;
- Radio Resource Control;
- DLC User Connection Control;
- Basic RLC transport machine.

The Association Control contains the functionality and the messages that are needed for establishing and releasing association, including Key Management and Authentication. Additionally, the broadcast and multicast group join and leave functions are included in Association Control.

The Radio Resource Control contains the functionality and the messages of Dynamic Frequency Selection, Transmission Power Control, different kind of Handovers, Power Saving, MT_Alive and MT_Absence functions. The DLC User Connection Control contains the functionality and the messages of connection Setup, Modify, Reset and Release between AP/CC and MT (centralized mode) and between MTs (direct mode).

4.3 Usage of DLC logical and transport channels for RLC messages

The logical and transport channels of HIPERLAN/2 are described in [5]. The uplink and downlink RLC messages use the logical channels DCCH or RBCH. DCCH and RBCH use the transport channels SCH or LCH for downlink communication. The logical channel DCCH use the transport channels SCH, LCH or RCH for uplink communication.

4.4 Transfer Syntax of RLC

The transfer syntax of the RLC PDUs is described in annex A. The usage of the first octet in all SCH PDUs and the three most significant bits of octet three in Uplink SCHs are described in [5]. The first octet and the four most significant bits in octet two of the LCH PDUs are defined in [5]. The Most Significant Bit (MSB) is always on the left end of the field and therefore not shown separately in the tables in the annex.

If a parameter is longer than one octet but is contained within one PDU, the most significant bit shall be placed in the octet with the lowest octet number of the octets containing the parameter and in the bit with the highest bit number. The least significant bit shall be placed in the octet with the highest octet number of the octets containing the parameter and in the bit with the lowest bit number.

If a parameter is longer than one PDU, the most significant bit shall be placed in the PDU that is sent first and follow the rules given above with the exception that it is the least significant bit of the first part of the parameter that takes the place of the least significant bit of the whole parameter. The first and subsequent PDUs except the last one are completely filled. The least significant bit of the parameter shall be placed in the PDU that is sent last and follow the rule given above with the exception that it is the most significant bit of the last part of the parameter that takes the place of the most significant part.

Table 1: Transfer syntax format of the RLC SCH in Downlink

	8	7	6	5	4	3	2	1
Octet 1	Defined in DLC TS							
Octet 2	MSB			RLC SCH PDU type				
Octet 3	MSB	EXTENSION-TYPE		MSB		RLC DATA		
Octet 4	MSB			RLC DATA				
…								
Octet 7								

Table 2: Transfer syntax format of the RLC SCH in UpLink and RCH

	8	7	6	5	4	3	2	1
Octet 1	Defined in DLC TS							
Octet 2	MSB			RLC SCH PDU type				
Octet 3	Defined in DLC TS			MSB	EXTENSION-TYPE		MSB	RLC DATA
Octet 4	MSB			RLC DATA				
…								
Octet 6								
Octet 7	MAC ID							

Table 3: Transfer syntax format of the RLC LCH in Uplink and Downlink

	8	7	6	5	4	3	2	1
Octet 1	Defined in DLC TS		MSB	Sequence number				
Octet 2	Sequence number				MSB	EXTENSION-TYPE		Future use
Octet 3	MSB			RLC LCH PDU type				
Octet 4	MSB			RLC DATA				
…								
Octet 51								

4.5 Sequencing of RLC messages

The high level MSCs, MSCs and the SDL diagrams define the sequences of the RLC messages.

The basic principle of sequencing shall be that a new message from a node shall be sent after that a reply to the last message has been received.

An important exception from the basic sequencing rule shall be the following: retransmission of the same message may be done in two ways. One way shall be according to the basic rule. The second way shall be that the (same) message is transmitted several times without waiting for a reply between the retransmission occasions.

For timer values that are longer than shortest timer value, an extra reply message shall be used supervised by an extra timer with the shortest timer value. The reason for this method is minimize the retransmission time.

4.6 Message Sequence Charts of RLC

The Message Sequence Charts (MSCs) of RLC PDU exchanges are shown in the corresponding clauses. Only the normal behaviour is defined in the MSCs.

An MSC consists of messages being exchanged between peers of the system. The messages contain parameters that are defined by ASN.1. The parts that are used in the RLC MSCs are RLC and environment instances in the MT and also RLC and environment instances in the AP. The environment is usually one of the informative function groups ACF, RRC or DUC, but may also be something else.

4.7 Abstract Syntax Notation one of the RLC PDUs

The mandatory and optional fields of the RLC PDUs are described with Abstract Syntax Notation one (ASN.1). The description is in annex B.

4.8 The text style of RLC PDUs and parameters

The RLC PDUs are always written with capital letters and an underscore between the words. They are normally called messages.

EXAMPLE 1: RLC_MAC_ID_ASSIGN, RLC_LINK_CAPABILITY_ACK

The parameters of RLC PDUs (messages) are written with small italic letters and there is a slash (-) between the words.

EXAMPLE 2: duc-ext-ind, sleep-interval

When an abbreviation, which is also used as a parameter is mentioned in text, the format of abbreviation is used (not the format of parameters).

EXAMPLE 3: MAC ID, AP ID

The message exchanges that belong to one function are often called procedures. The functions and procedures are written in the following way: the first letter is a capital letter and the rest are small letters.

EXAMPLE 4: Association, MT_Alive

4.9 Mandatory and Optional functions

The present document contains mandatory and optional functions at different levels. For functions that are stated as mandatory, optional and mandatory subfunctions, messages and parameters can be defined.

Functions, subfunctions, messages and parameters that are stated as optional in the present document may be set as mandatory in extension documents or other documents referring to the present document.

Every function, subfunction, message or parameter that is optional to implement in order to comply with the present document is marked with an "OAP" or "OMT" in the present document, depending on whether it is optional in the MT or the AP. All functions, subfunctions, messages and parameters that are not marked are mandatory to implement.

Functions, subfunctions, messages and parameters that can be selected or not at a particular occasion are negotiated at that occasion and the negotiation is specified in the present document in MSCs, ASN.1s, and SDLs.

EXAMPLE: The Encryption startup PDUs are mandatory to implement, but optional to use. The decision of the use is made by the AP.

4.10 Message transmission not allowed

It shall not be allowed to send messages to an AP that has set the traffic load indicator in the BCCH to the value 001.

5 RLC Services

5.1 Services supporting ACF (Association Control Function)

5.1.1 Association

During the Association procedure, the AP allocates a locally unique MAC ID [5] for the requesting MT.

The Association procedure consists of the following procedures:

- RBCH Association;
- MAC ID Assignment;
- Link Capability Negotiation;
- Encryption Startup;
- Authentication;
- DM Common Key Distribution (OMT/OAP);
- Info Transfer.

The order of the procedures is shown in the following high level Message Sequence Chart.

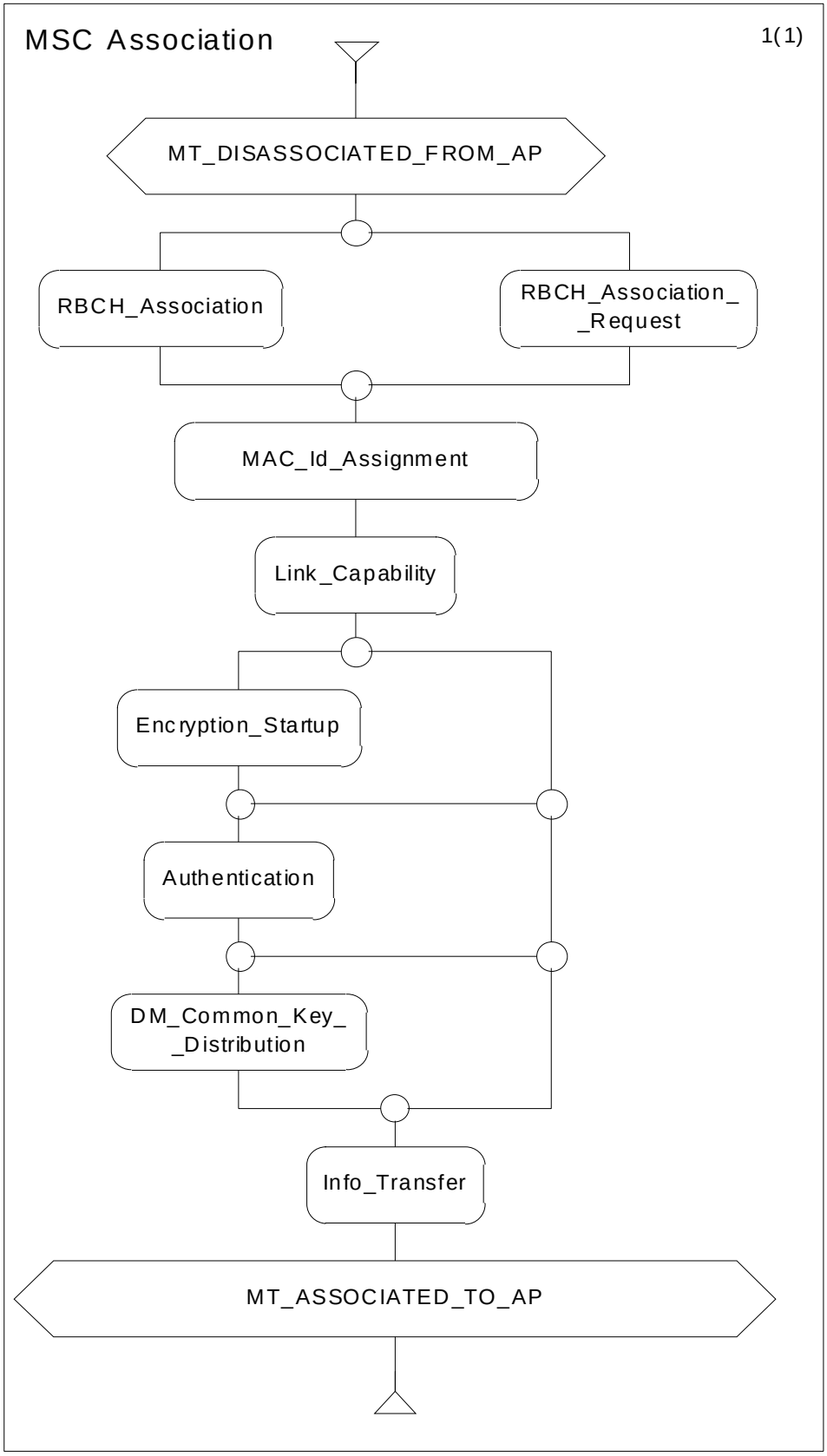


Diagram 1: HMSC of the Association procedure

5.1.1.1 RBCH Association

The AP shall send the RLC_RBCH_ASSOCIATION message periodically. The number of frames between periodic transmission is outside the scope of the standard. The AP shall send the RLC_RBCH_ASSOCIATION message, when requested by the MT with RLC_RBCH_ASSOCIATION_REQ message. The MT shall request the RLC_RBCH_ASSOCIATION message, if the MT has not received the periodically sent message.

If the *c-u-g* in the RLC_RBCH_ASSOCIATION message indicates that the group is closed, the MT should compare the Network Operator ID (NOP ID) to the preferred NOP IDs in the MT before the MT sends RLC_MAC_ID_ASSIGN message. The MT should associate to a group that is included in the MT's NOP ID list. If the *c-u-g* of the AP indicates an open group the MT may continue the Association procedure, but the MT shall compare the profile id/version first.

NOTE: The allocation of the global part of the NOP ID is managed by ETSI. The allocation of the local part of the NOP ID is outside of scope of the present document. The network may be set up without having a NOP ID for APs.

If none of the profile id:s/profile versions sent in the RLC_RBCH_ASSOCIATION message is supported by the MT, it shall not continue the Association. If one or more of the profile id:s/profile versions sent in the RLC_RBCH_ASSOCIATION message is supported by the MT, it may continue the Association.

Among parameters that are included in the profile id/version is convergence layer id/version.

If the MT decides to continue the association procedure, the MT shall send the RLC_MAC_ID_ASSIGN message to the AP.

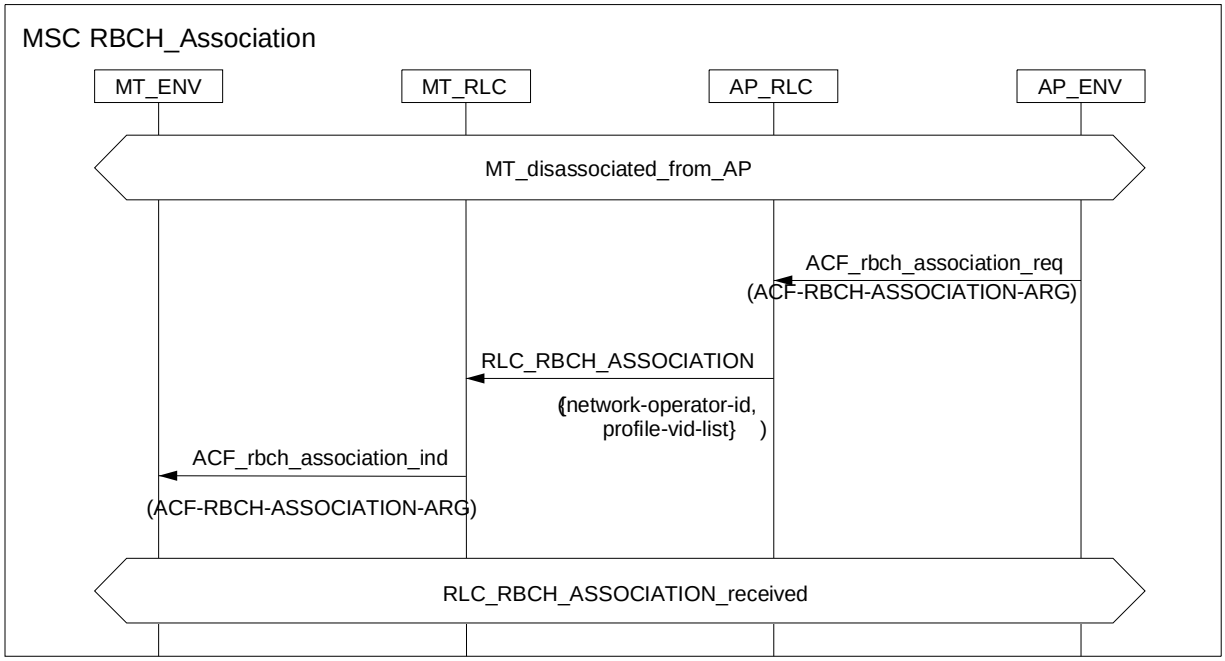


Diagram 2: RBCH Association

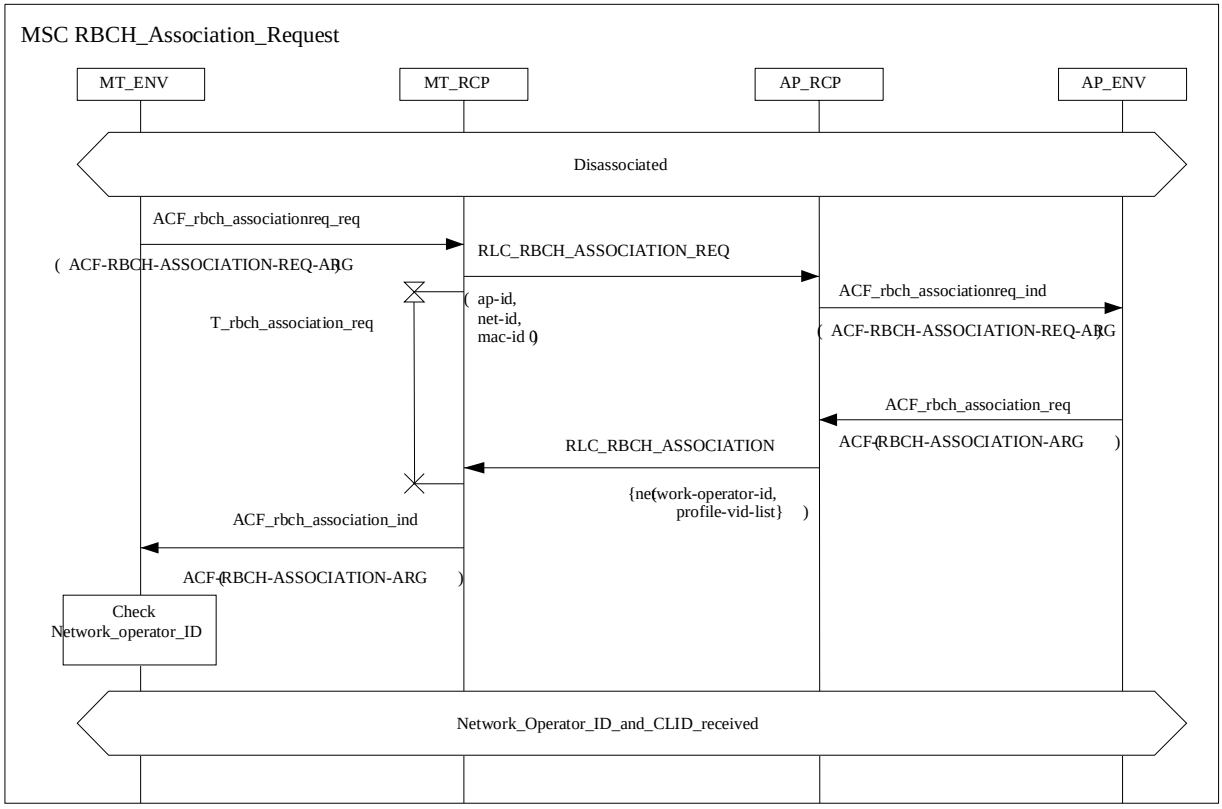


Diagram 3: RBCH Association Request

Table 4: RLC-RBCH-ASSOCIATION

RLC-RBCH-ASSOCIATION-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
network-operator-id	NETWORK-OPERATOR-ID OPTIONAL
profile-vid-list	PROFILE-VID-LIST }

Table 5: RLC-RBCH-ASSOCIATION-REQ (OMT)

RLC-RBCH-ASSOCIATION-REQ-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
ap-id	AP-ID
net-id	NET-ID
mac-id	MAC-ID0 }

5.1.1.2 MAC ID Assignment

The MT shall request a MAC ID from the AP by sending RLC_MAC_ID_ASSIGN message. The parameter *magic* is a temporary identifier for the MT and should be a random number. The parameters *mac-id* and *mac-id1* of the RLC_MAC_ID_ASSIGN message shall be set to zero. The parameter *mac-id* of the RLC_MAC_ID_ASSIGN_ACK message shall contain the MAC ID allocated by the AP. The RLC_MAC_ID_ASSIGN_ACK message shall be sent via RBCH.

NOTE: The *mac-id* in RLC_MAC_ID_ASSIGN_ACK is doubled to decrease the risk of errors in the *mac-id.*

If the AP does not allocate a MAC ID for the MT, the AP shall send the RLC_MAC_ID_ASSIGN_NACK message.

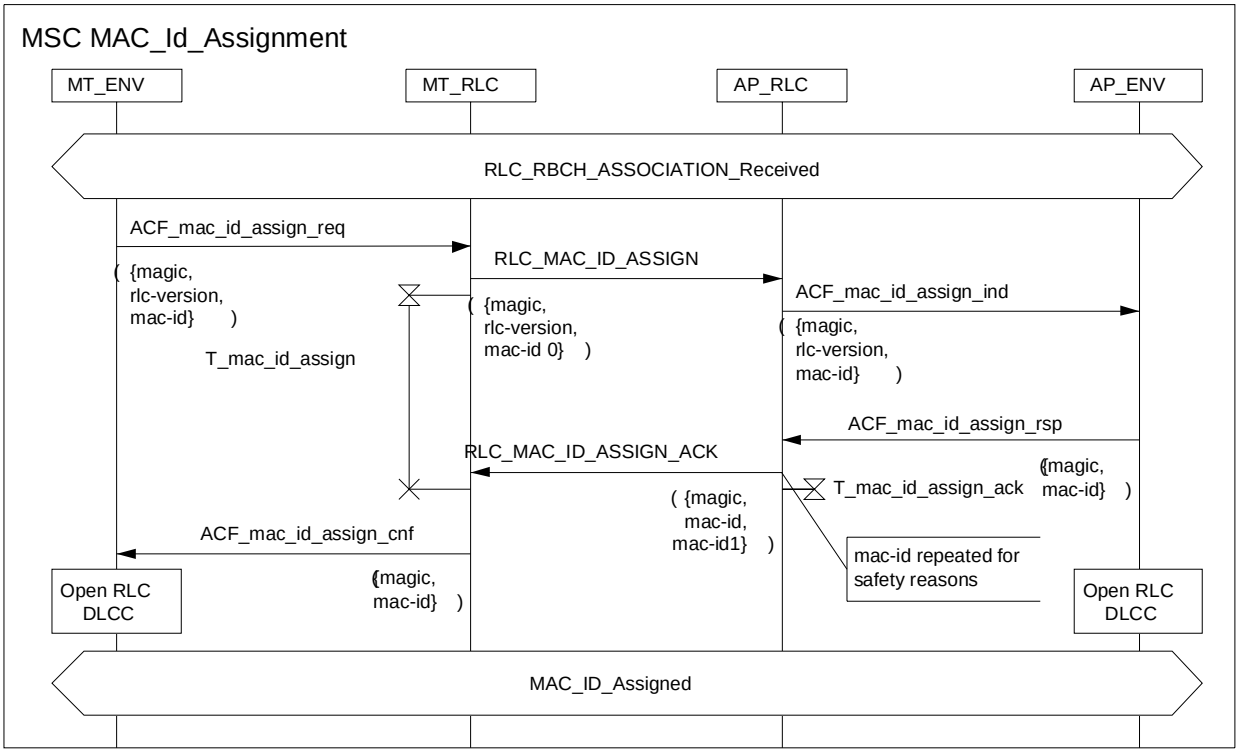


Diagram 4: MAC ID Assignment

Table 6: RLC-MAC-ID-ASSIGN

RLC-MAC-ID-ASSIGN-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
magic	MAGIC
rlc-version	RLC-VERSION
mac-id	MAC-ID0 }

Table 7: RLC-MAC-ID-ASSIGN-ACK

RLC-MAC-ID-ASSIGN-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
magic	MAGIC
mac-id	MAC-ID
mac-id1	MAC-ID }

Table 8: RLC-MAC-ID-ASSIGN-NACK

RLC-MAC-ID-ASSIGN-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
magic	MAGIC}

5.1.1.3 Link Capability

The MT shall propose link capability alternatives to the AP in the RLC_LINK_CAPABILITY message. The AP shall select from the link capability alternatives proposed by the MT and add its own link capabilities to the RLC_LINK_CAPABILITY_ACK message, which shall then be sent to the MT.

If the MT accepts the parameters in the RLC_LINK_CAPABILITY_ACK message, it shall proceed with the Association. Whether the Association continues with Encryption startup, Authentication, DM Common Key distribution, Info Transfer or goes directly to the associated state, shall be controlled by parameters in the RLC_LINK_CAPABILITY_ACK message.

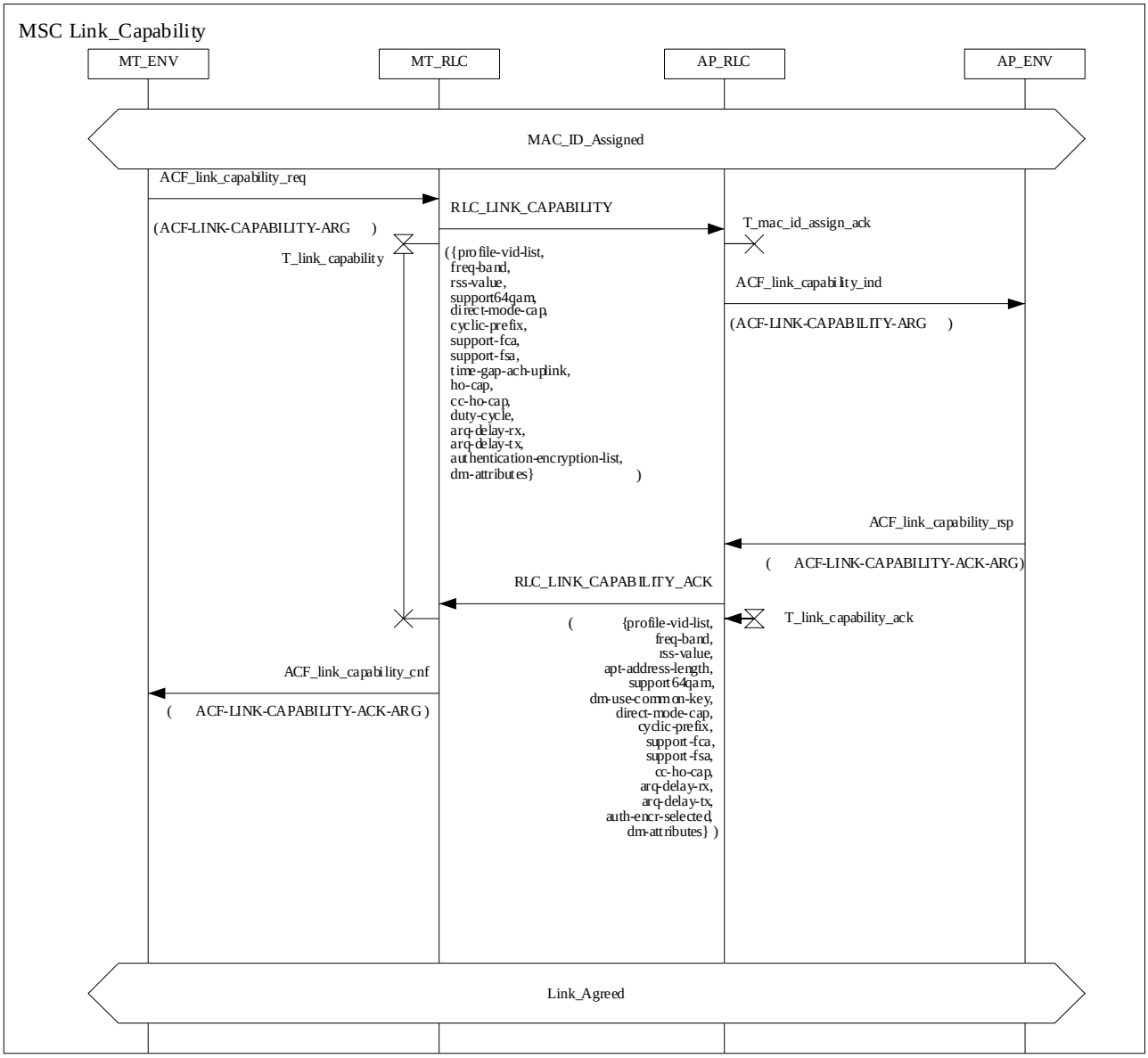


Diagram 5: Link Capability Negotiation

Table 9: RLC-LINK-CAPABILITY

RLC-LINK-CAPABILITY-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
profile-vid-list	PROFILE-VID-LIST
freq-band	FREQUENCY-BAND
rss-value	RSS-VALUE
support64QAM	SUPPORTED64QAM
direct-mode-cap	DIRECT-MODE-CAP
cyclic-prefix	CYCLIC-PREFIX
support-fca	SUPPORTED-FCA
support-fsa	SUPPORTED-FSA
time-gap-ach-uplink	TIME-GAP-ACH-UPLINK
ho-cap	HO-CAP
cc-ho-cap	CC-HO-CAP
duty-cycle	DUTY-CYCLE
arq-delay-rx	ARQ-DELAY
arq-delay-tx	ARQ-DELAY
authentication-encryption-list	AUTHENTICATION-ENCRYPTION-LIST
dm-attributes	DM-ATTRIBUTES OPTIONAL -- (OMT/OAP) -- }

Table 10: RLC-LINK-CAPABILITY-ACK

RLC-LINK-CAPABILITY-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
profile-vid-list-selected	PROFILE-VID-LIST
freq-band	FREQUENCY-BAND
rss-value	RSS-VALUE
apt-address-length	APT-ADDRESS-LENGTH
support64QAM	SUPPORTED64QAM
dm-use-common-key	DM-USE-COMMON-KEY
direct-mode-cap	DIRECT-MODE-CAP
cyclic-prefix	CYCLIC-PREFIX
support-fca	SUPPORTED-FCA
support-fsa	SUPPORTED-FSA
cc-ho-cap	CC-HO-CAP
arq-delay-rx	ARQ-DELAY
arq-delay-tx	ARQ-DELAY
auth-encr-selected	AUTH-ENCR-INFO
dm-attributes	DM-ATTRIBUTES OPTIONAL -- (OMT/OAP) -- }

5.1.1.4 Encryption startup

The MT shall calculate and send its public Diffie-Hellman value to the AP. The AP shall also calculate and send its public Diffie-Hellman value to the MT. Both MT and AP shall calculate encryption keys based on the received public Diffie-Hellman values. How the public Diffie-Hellman values and the encryption keys shall be calculated is described in clause 5.1.2.5.1.

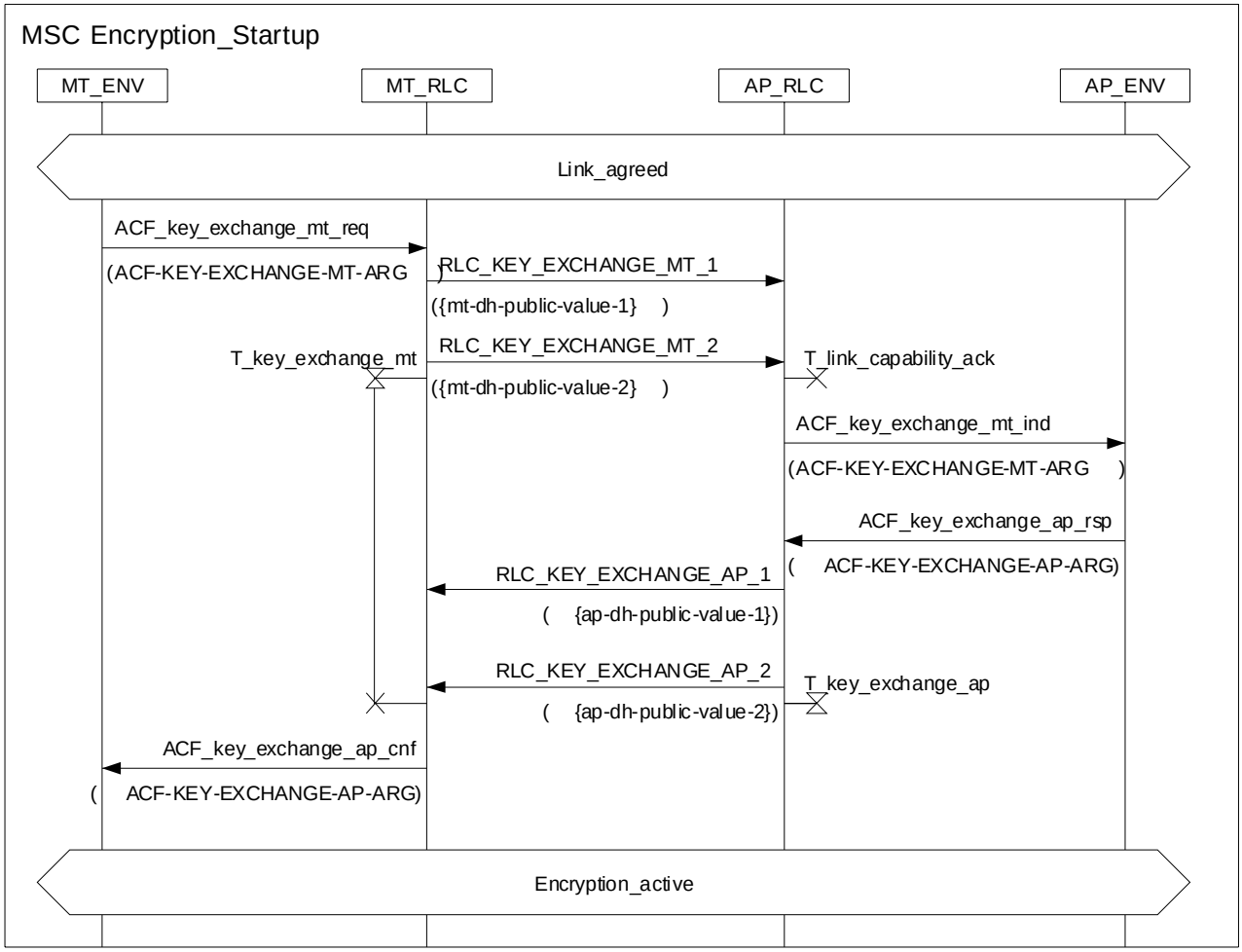


Diagram 6: Encryption Startup procedure

Table 11: RLC-KEY-EXCHANGE-MT-1

RLC-KEY-EXCHANGE-MT-1-ARG ::= SEQUENCE {
rlc-pdu-type RLC-LCH-PDU-TYPE
mt-dh-public-value-1 DH-PUBLIC-VALUE-HALF}

Table 12: RLC-KEY-EXCHANGE-MT-2

RLC-KEY-EXCHANGE-MT-2-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
mt-dh-public-value-2	DH-PUBLIC-VALUE-HALF }

Table 13: RLC-KEY-EXCHANGE-AP-1

RLC-KEY-EXCHANGE-AP-1-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
ap-dh-public-value-1	DH-PUBLIC-VALUE-HALF }

Table 14: RLC-KEY-EXCHANGE-AP-2

RLC-KEY-EXCHANGE-AP-2-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
ap-dh-public-value-2	DH-PUBLIC-VALUE-HALF }

5.1.1.5 Authentication

5.1.1.5.1 General

The scope for the authentication is limited to the local HIPERLAN/2 wireless LAN. Higher level or remote authentication, for example user authentication through the Internet to corporate network, is outside the scope of the present document.

Mutual authentication is supported. MT authentication controls the access of the MT to the connected fixed network. If the authentication of the MT fails no access shall be granted to the MT.

NOTE: AP authentication helps a MT to detect false APs.

The AP shall decide if MTs are allowed to access the network without authentication, and the MT may decide to cancel the access attempt to a network if AP authentication fails or is not supported.

5.1.1.5.2 Authentication procedures

There are six possible types of identifiers for the authentication key, the signalling related to these are shown in the HMSC for authentication.

There are two different types of authentication protocols, pre-shared key based and RSA signature based, see clause 5.1.2.6.2.

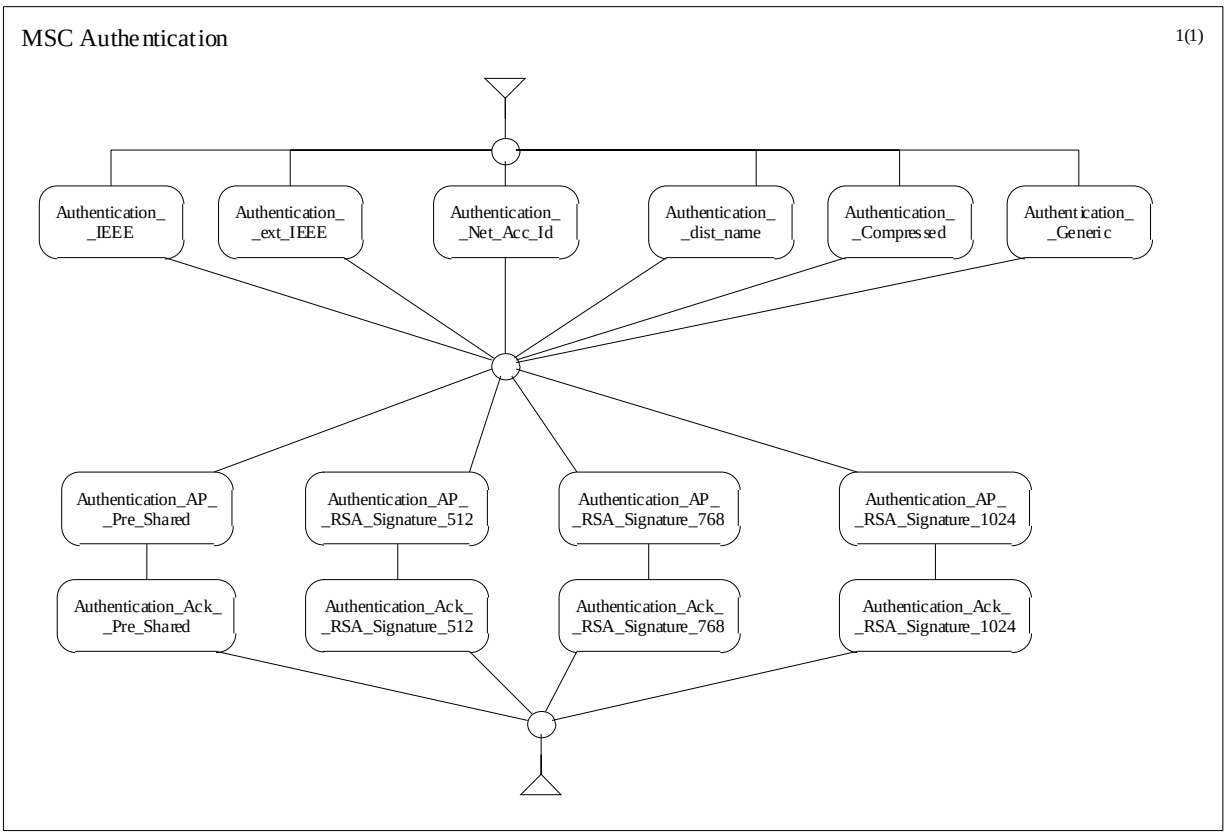


Diagram 7: HMSC of the Authentication procedure

5.1.1.5.3 Authentication key identifier alternatives

5.1.1.5.3.1 General

The MT can fetch the authentication key of the AP based on all or parts of the Network Operator (NOP ID), network (NET ID), and AP (AP ID) identities that are sent over the broadcast channels.

Each MT shall be assigned an authentication key identifier that should be presented to the connected AP at association. The AP shall use it to retrieve the key related to the access. The identifier can be of different types. One of the identifier types is mandatory to implement (free to choose), the remaining ones are optional. The MT authentication key identifier should be protected by encryption during transfer to the AP.

5.1.1.5.3.2 IEEE address as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1)

This procedure shall use the 48-bit IEEE address [8] as authentication key identifier. The AP shall send a challenge to the MT as response.

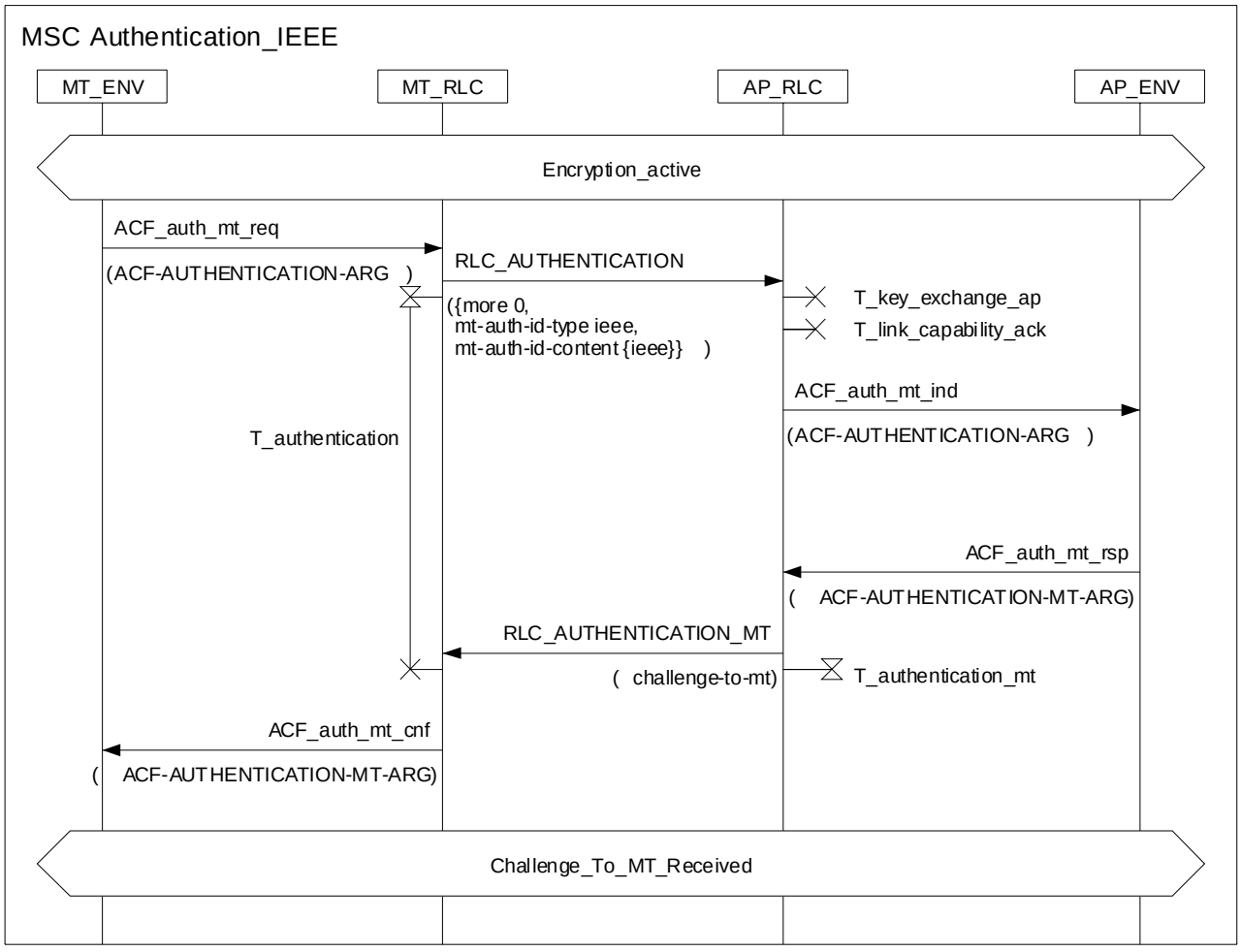


Diagram 8: Authentication IEEE

Table 15: RLC-AUTHENTICATION

RLC-AUTHENTICATION-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH_PDU-TYPE
more	MORE-AUTH
mt-auth-id-type	MT-AUTH-ID-TYPE
mt-auth-id-content	MT-AUTH-CONTENT }

Table 16: RLC-AUTHENTICATION-MT

RLC-AUTHENTICATION-MT-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
challenge-to-mt	CHALLENGE }

5.1.1.5.3.3 Extended IEEE address as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1)

This procedure shall use the 64 bit extended IEEE address [9] as authentication key identifier. The AP shall send a challenge to the MT as response.

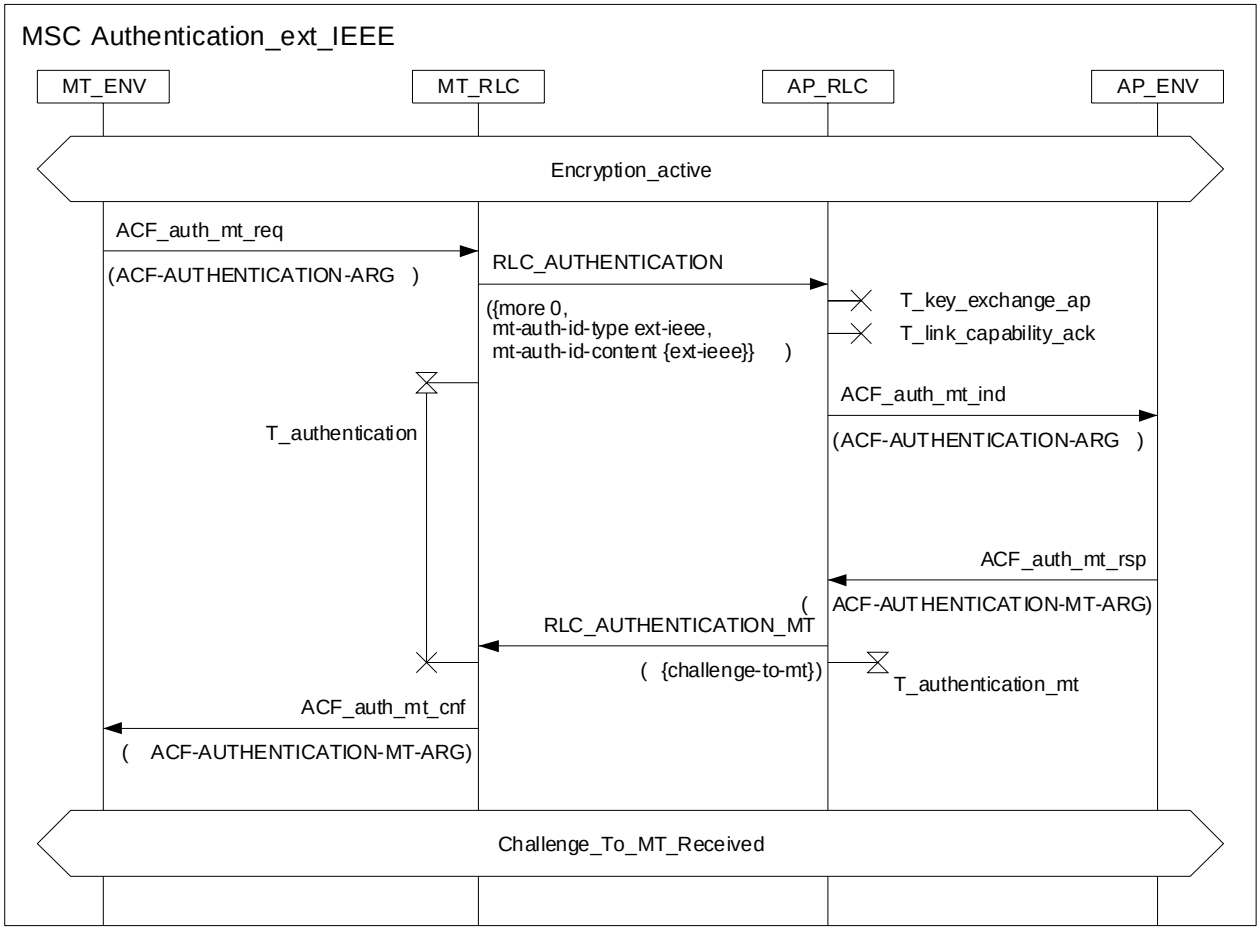


Diagram 9: Authentication EXT_IEEE

5.1.1.5.3.4 Network access identifier as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1)

This procedure shall be used when the NAI (network access identifier) [10] is used as authentication key identifier. The AP shall send a challenge to the MT as response.

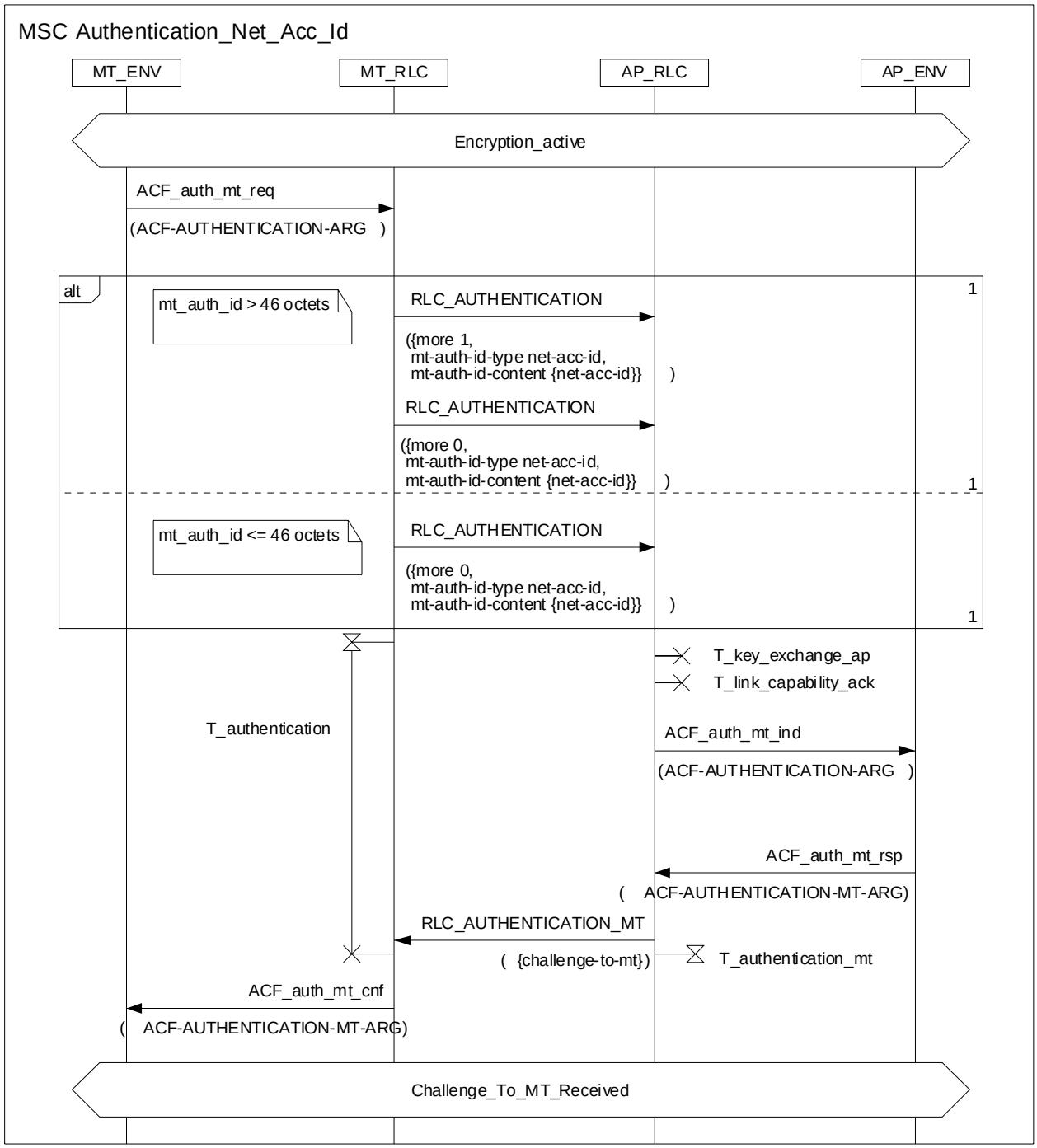


Diagram 10: Authentication NAI

5.1.1.5.3.5 Distinguished name as on authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1)

This procedure shall be used if distinguished name [11] is the MT authentication key identifier. The AP shall send a challenge to the MT as response.

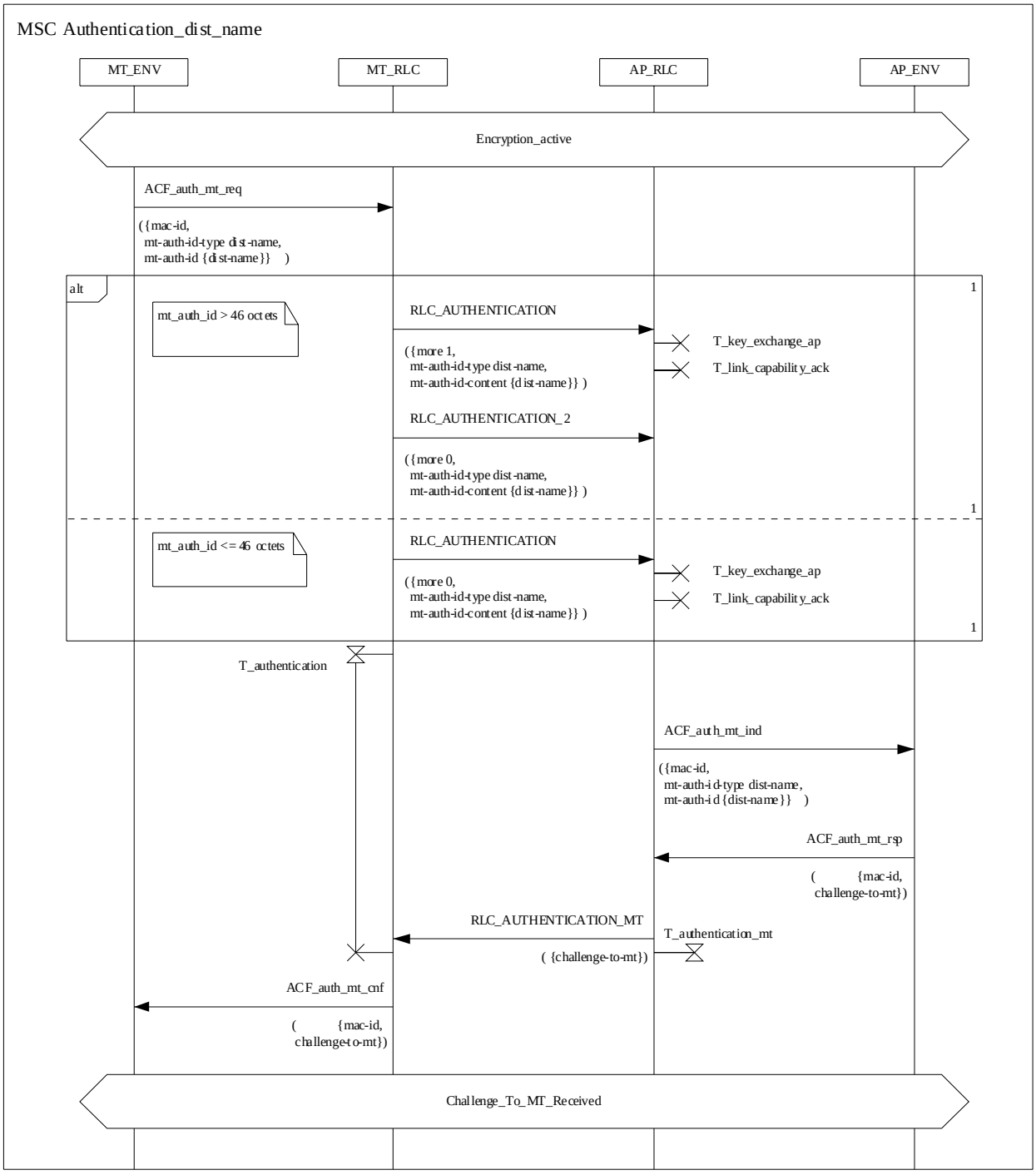


Diagram 11: Authentication Distinguished name

5.1.1.5.3.6 Compressed type as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1)

This procedure shall be used if the compressed type is the MT authentication key identifier. The compressed type can be used if the available authentication key identifier is so long that it is not possible to carry in the defined RLC messages. The compressed authentication key identifier is calculated as follows: compressed-authentication-key-identifier = MD5(available_authentication_key_identifier). The AP shall send a challenge to the MT as response.

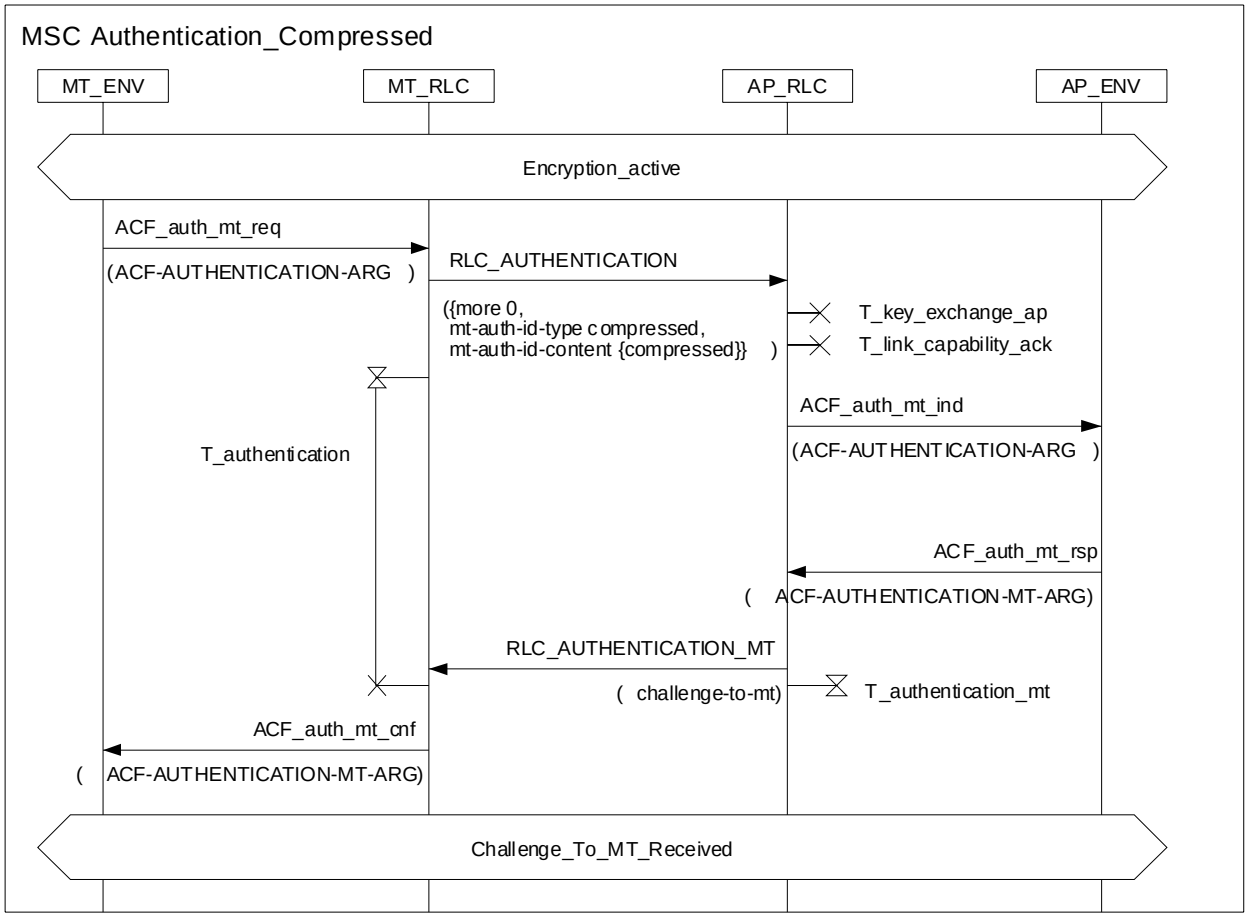


Diagram 12: Authentication Compressed

5.1.1.5.3.7 Generic type as authentication key identifier (Conditional OAP/OMT, see 5.1.1.5.3.1)

This procedure shall be used if the generic type is the MT authentication key identifier. The generic type is a non-structured octet string. The AP shall send a challenge to the MT as response.

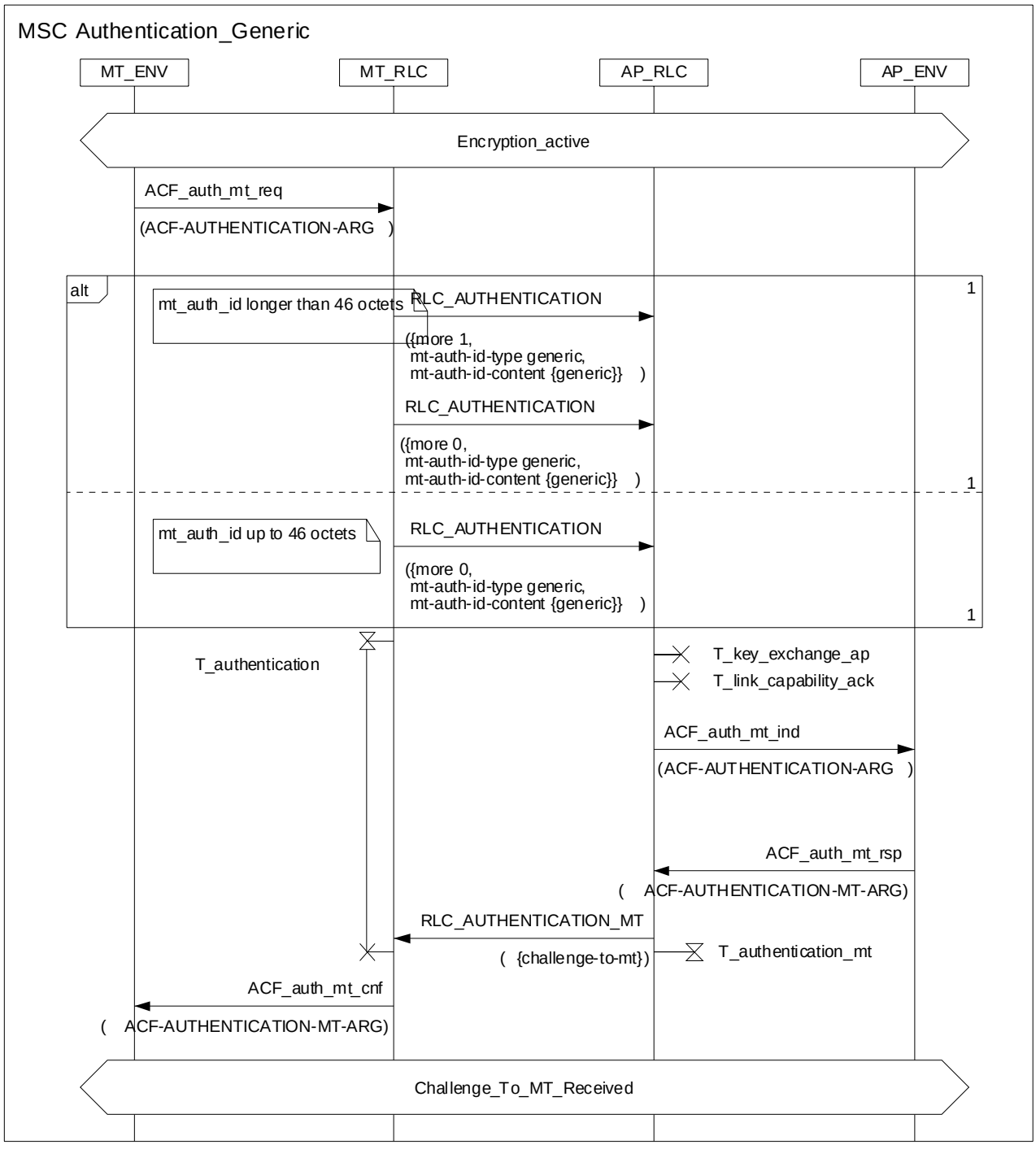


Diagram 13: Authentication Generic

5.1.1.6 Authentication based on different key types

5.1.1.6.1 Authentication with pre-shared key

The MT shall send a challenge to the AP as well as the calculated response. How the response shall be calculated is described in clause 5.1.2.6.3.

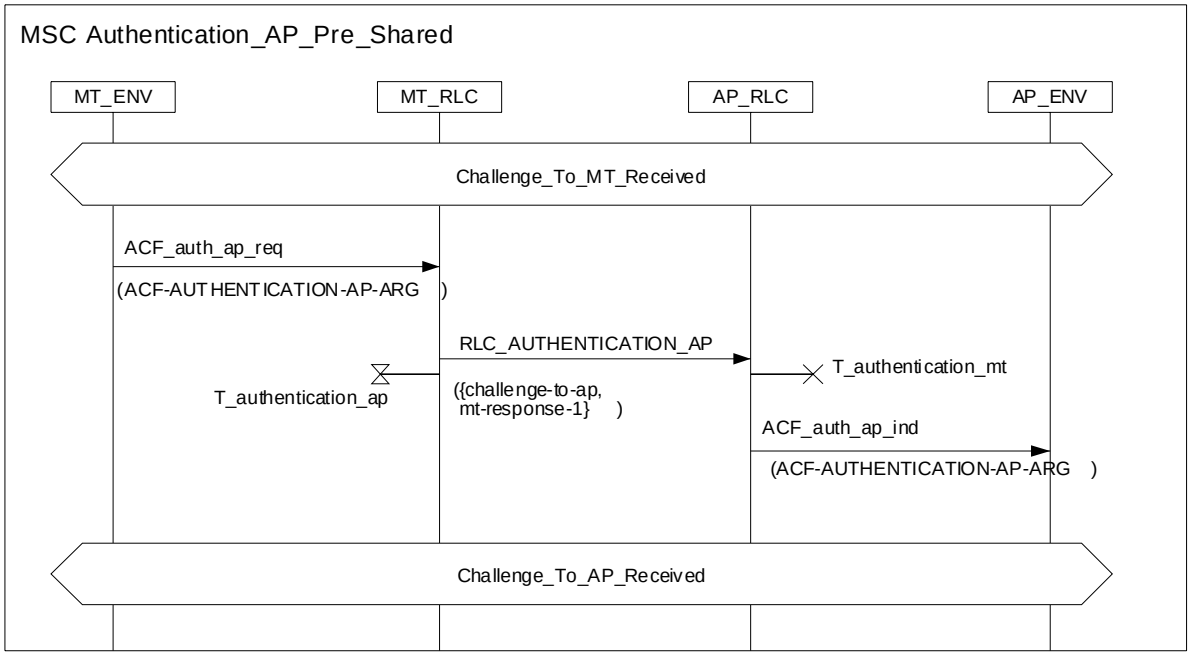


Diagram 14: Authentication AP Pre-shared

Table 17: RLC-AUTHENTICATION-AP-1

RLC-AUTHENTICATION-AP-1-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
challenge-to-ap	CHALLENGE
mt-response-1	AUTH-RESPONSE-PART1 }

Table 18: RLC-AUTHENTICATION-AP-2

RLC-AUTHENTICATION-AP-2-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
mt-response-2	AUTH-RESPONSE-PART2 }

Table 19: RLC-AUTHENTICATION-AP-3

RLC-AUTHENTICATION-AP-3-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
mt-response-2	AUTH-RESPONSE-PART2 }

The AP shall send the calculated response to the MT. How the response shall be calculated described in clause 5.1.2.6.4.

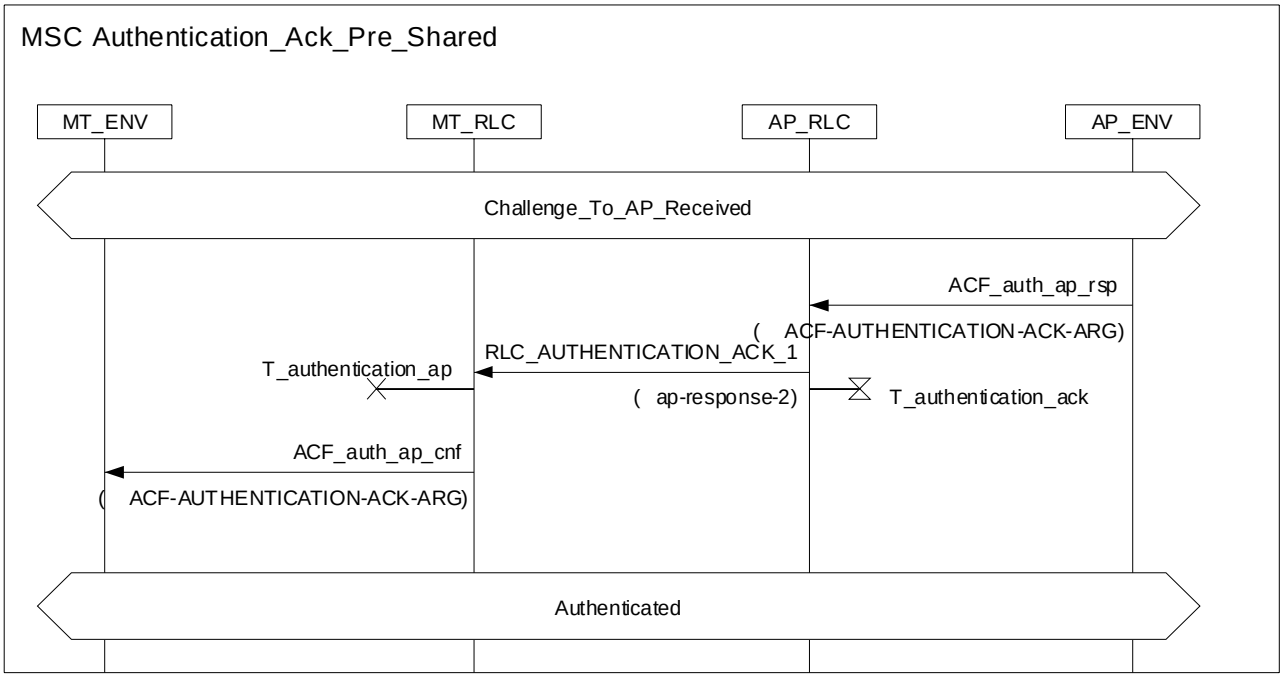


Diagram 15: Authentication Ack Pre-shared

Table 20: RLC-AUTHENTICATION-ACK-1

RLC-AUTHENTICATION-ACK-1-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
ap-response-2	AUTH-RESPONSE-PART2 }

Table 21: RLC-AUTHENTICATION-ACK-2

RLC-AUTHENTICATION-ACK-2-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
ap-response-2	AUTH-RESPONSE-PART2}

Table 22: RLC-AUTHENTICATION-ACK-3

RLC-AUTHENTICATION-ACK-3-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
ap-response-2	AUTH-RESPONSE-PART2}

5.1.1.6.2 Authentication based on 512 bit RSA signature (OAP/OMT)

The MT shall send a challenge to the AP as well as the calculated response. How the response shall be calculated is described in clause 5.1.2.6.4.

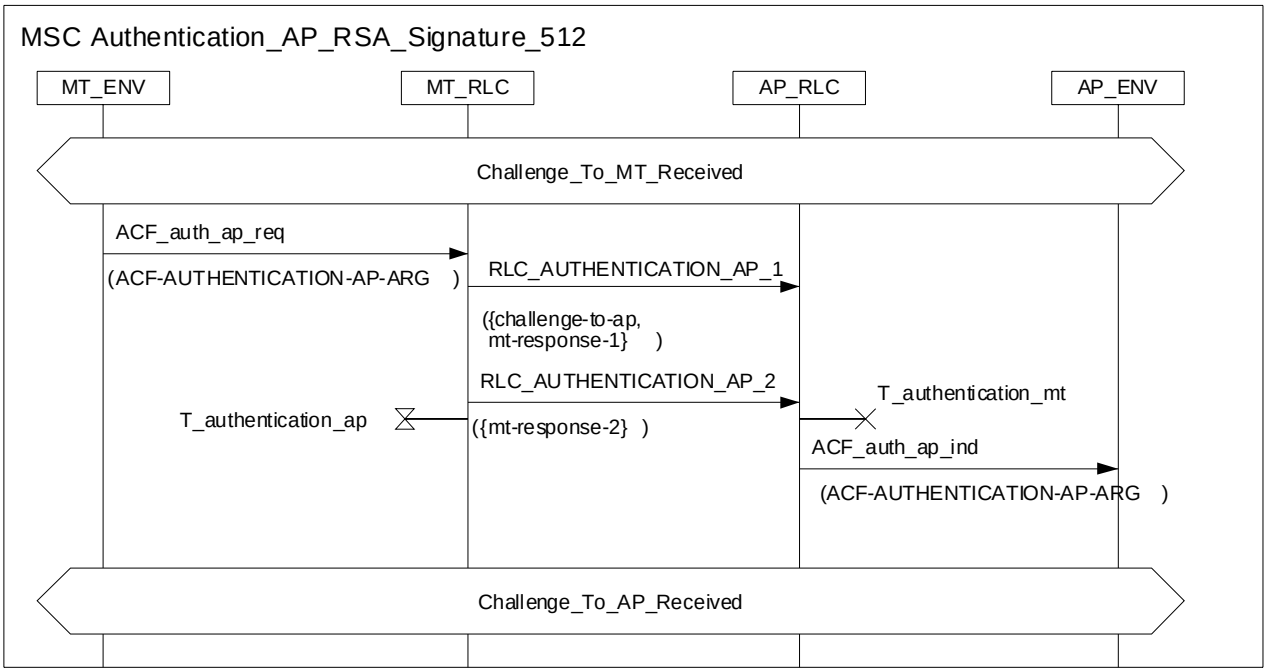


Diagram 16: Authentication AP RSA Signature 512

The RLC_AUTHENTICATION_AP_1 and RLC_AUTHENTICATION_AP_2 messages are defined in clause 5.1.1.6.1.

The AP shall send the calculated response to the MT. How the response shall be calculated is described in clause 5.1.2.6.4.

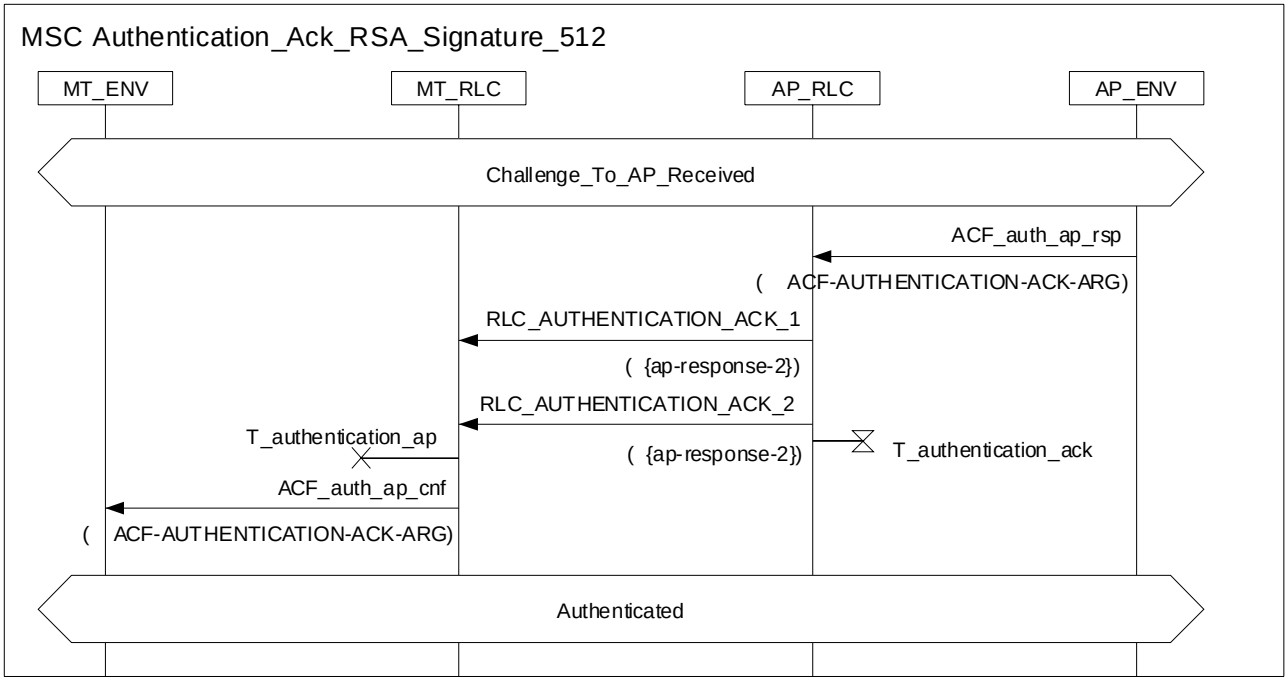


Diagram 17: Authentication Ack RSA Signature 512

The RLC_AUTHENTICATION_ACK_1 and RLC_AUTHENTICATION_ACK_2 messages are defined in clause 5.1.1.6.1.

5.1.1.6.3 Authentication based on 768 bit RSA signature (OAP/OMT)

The MT shall send a challenge to the AP as well as the calculated response. How the response shall be calculated is described in clause 5.1.2.6.4.

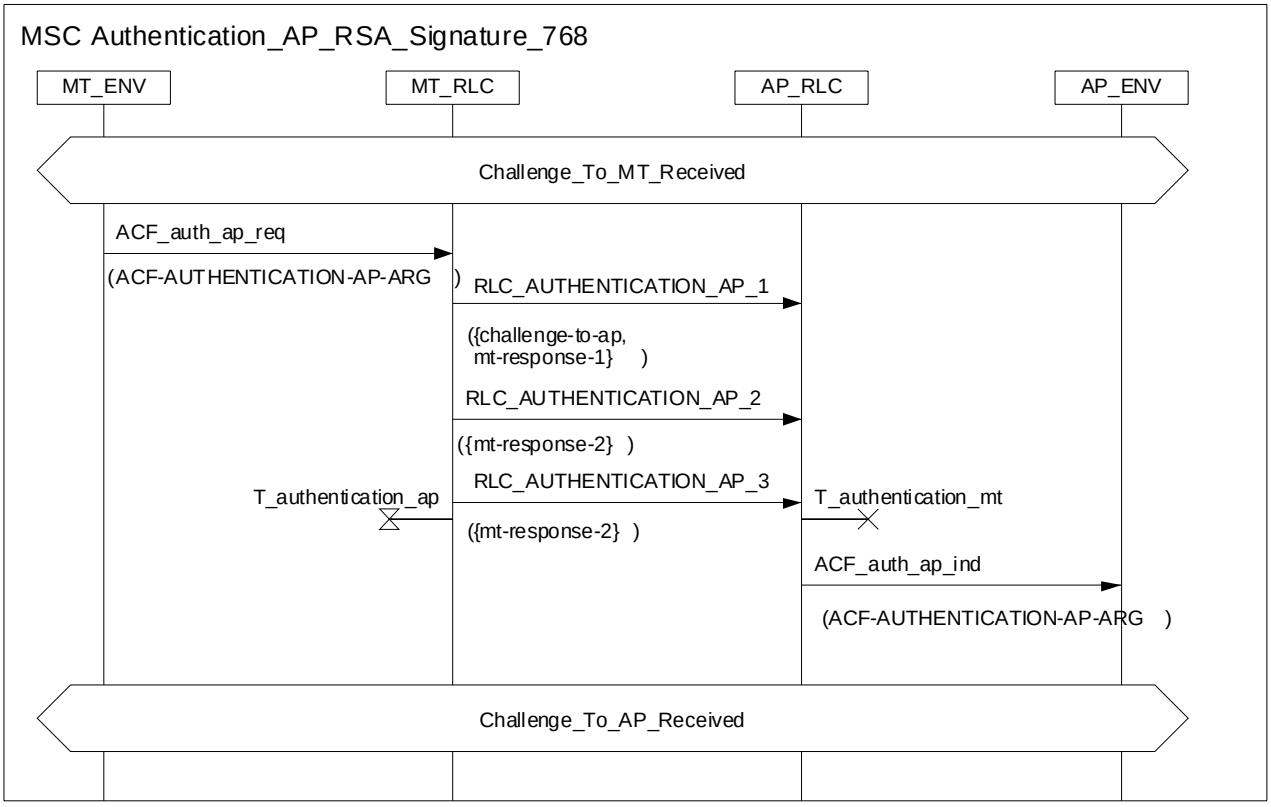


Diagram 18: Authentication AP RSA Signature 768

The RLC_AUTHENTICATION_AP_1, RLC_AUTHENTICATION_AP_2 and RLC_AUTHENTICATION_AP_3 messages are defined in clause 5.1.1.6.1.

The AP shall send the calculated response to the MT. How the response shall be calculated is described in clause 5.1.2.6.4.

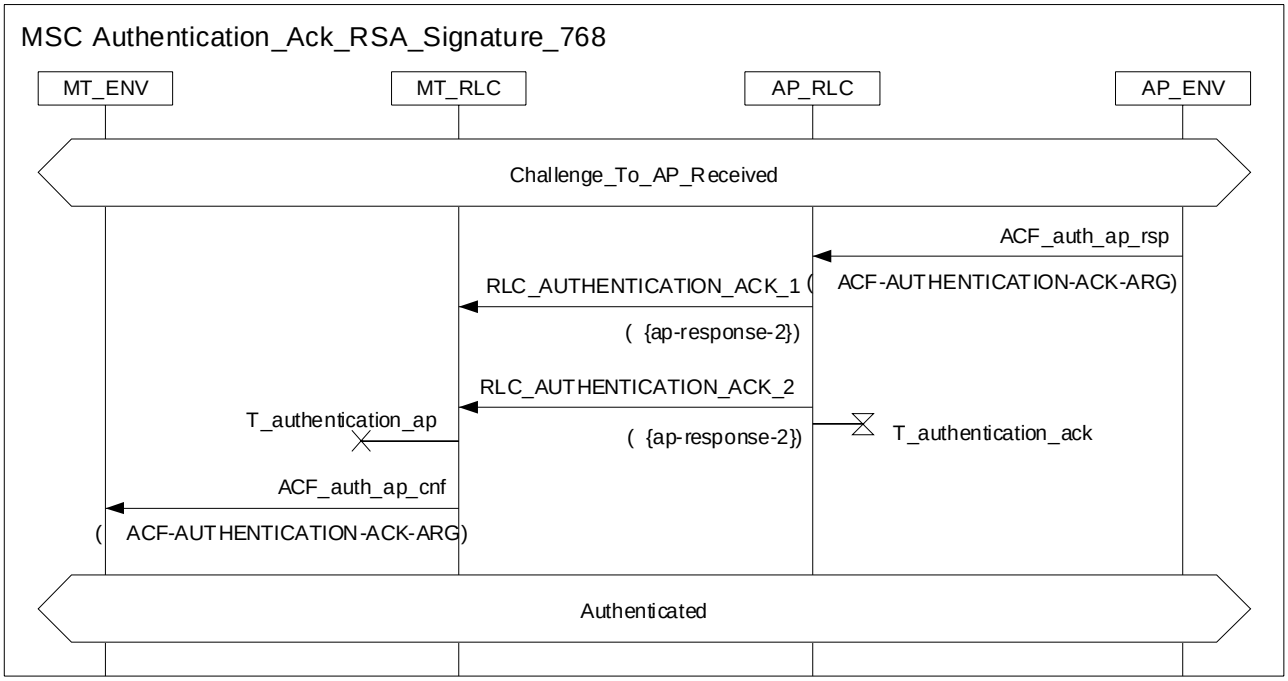


Diagram 19: Authentication Ack RSA Signature 768

The RLC_AUTHENTICATION_ACK_1 and RLC_AUTHENTICATION_ACK_2 messages are defined in clause 5.1.1.6.1.

5.1.1.6.4 Authentication based on 1 024 bit RSA signature (OAP/OMT)

The MT shall send a challenge to the AP as well as the calculated response. How the response shall be calculated is described in clause 5.1.2.6.4.

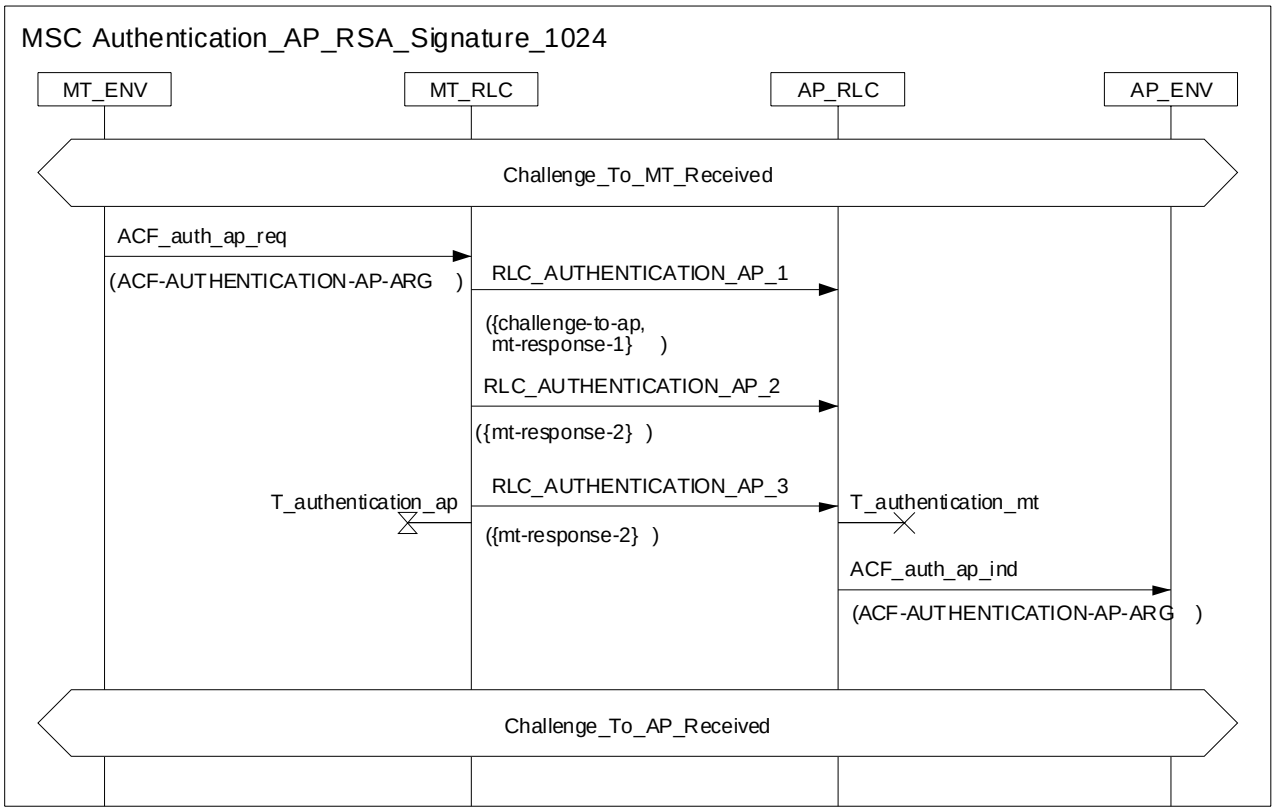


Diagram 20: Authentication AP RSA Signature 1 024

The RLC_AUTHENTICATION_AP_1, RLC_AUTHENTICATION_AP_2 and RLC_AUTHENTICATION_AP_3 messages are defined in clause 5.1.1.6.1.

The AP shall send the calculated response to the MT. How the response shall be calculated is described in clause 5.1.2.6.4.

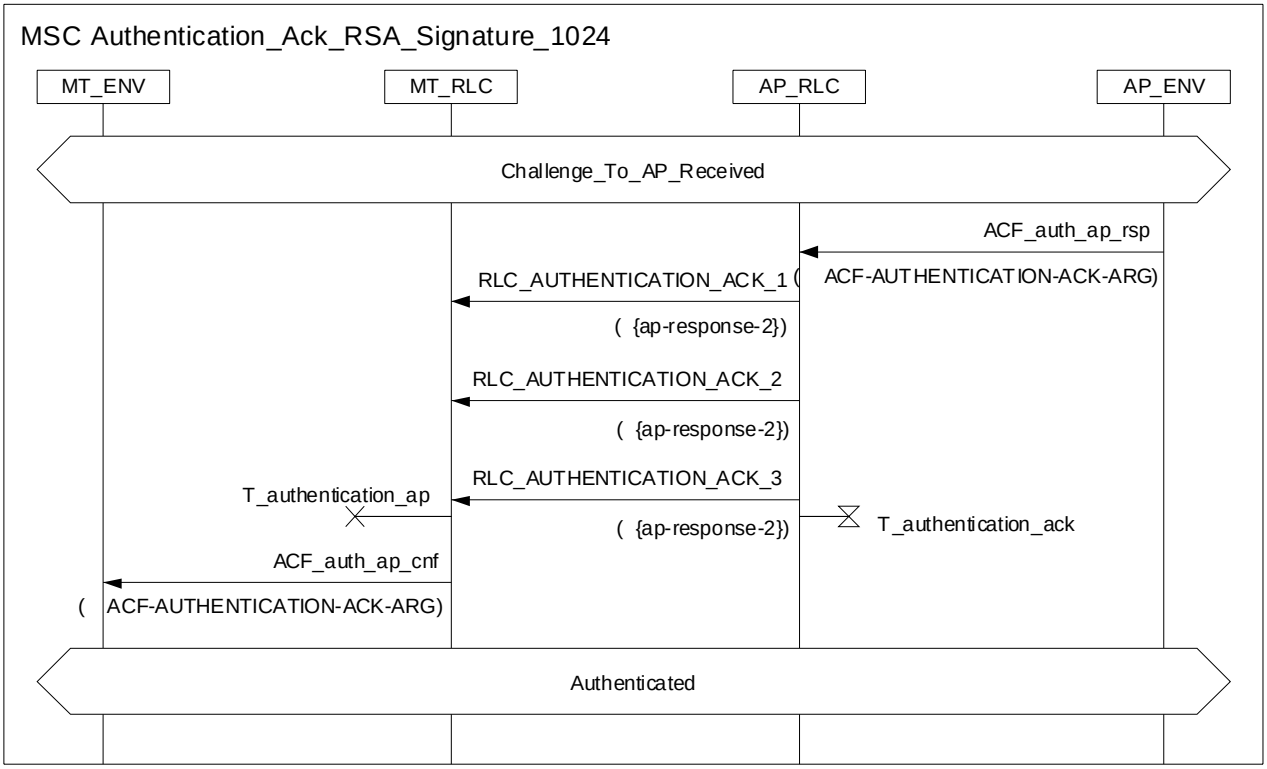


Diagram 21: Authentication Ack RSA Signature 1 024

The RLC_AUTHENTICATION_ACK_1, RLC_AUTHENTICATION_ACK_2 and RLC_AUTHENTICATION_ACK_3 messages are defined in clause 5.1.1.6.1.

5.1.1.7 DM Common Key Distribution (OAP/OMT)

The AP shall inform the MT about the encryption algorithm, the KEY ID and common key that shall be used for direct mode encryption. The MT shall confirm that the encryption algorithm and that the direct mode encryption key have been received.

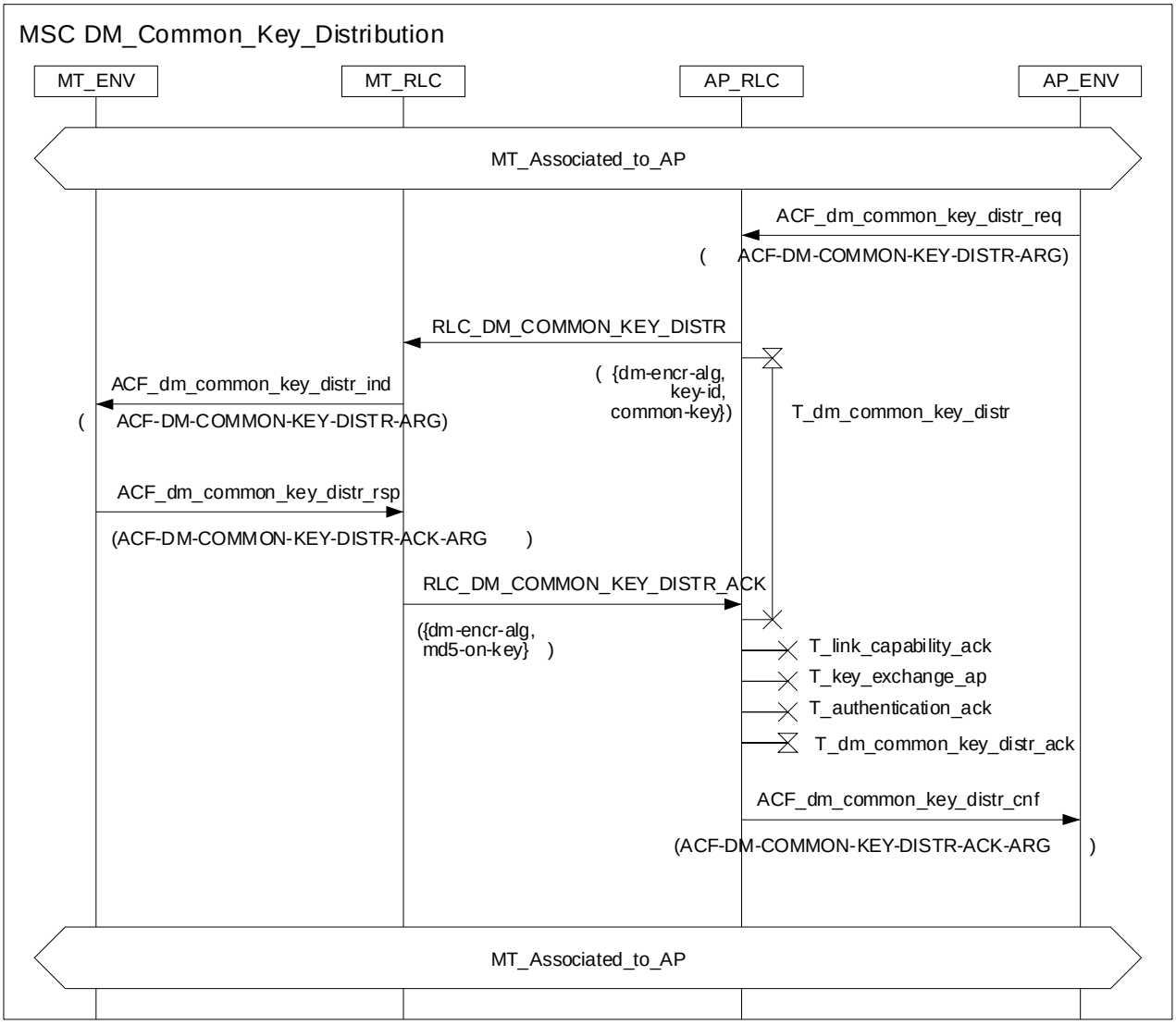


Diagram 22: DM Common Key Distribution

Table 23: RLC-DM-COMMON-KEY-DISTR

RLC-DM-COMMON-KEY-DISTR-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
dm-encr-alg	ENCR-INFO
key-id	KEY-ID
common-key	COMMON-KEY }

Table 24: RLC-DM-COMMON-KEY-DISTR-ACK

RLC-DM-COMMON-KEY-DISTR-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
dm-encr-alg	ENCR-INFO
md5-on-key	MD5-ON-KEY }

5.1.1.8 Info Transfer procedure (OAP/OMT, depends on CL and DLC)

The MTs and AP/CC may use this function to exchange convergence layer or DLC layer information.

In centralized mode (RLC messages sent over the DCCH in the uplink down link phase), the Info Transfer procedure is always initiated by the MT. Therefore, the MT shall use the RLC_INFO message and the AP/CC shall use the RLC_INFO_ACK message. The AP shall send the RLC_INFO_ACK as a response to the RLC_INFO message.

In direct mode (RLC messages sent over the DCCH in the direct link phase), the Info Transfer procedure may be initiated by either a MT or the AP/CC. The MT or AP/CC receiving a RLC_INFO message shall send the RLC_INFO_ACK message as a response.

Info Transfer procedure shall run at the end of the association phase. It may also run at any time after one MT has been associated.

In case of multiple CLs the MT or AP/CC shall check the CL-ID in the RLC_INFO_ACK to associate the respond with corresponding request.

During the association phase, several RLC_INFO messages may be sent in sequence. The number of messages depend from the supported convergence layers.

The INFO-TYPE field in the RLC_INFO message indicates whether the RLC_INFO is a new RLC_INFO message or the retransmission of a non-acknowledged, and already sent RLC_INFO message.

The INFO-COUNT field is used in different ways in the association phase and outside the association phase.

- In the association phase, the INFO-COUNT field is used to indicate to the peer entity the number of remaining RLC_INFO messages to be transmitted for the whole set of supported convergence layers during the association phase.

- Outside of the association phase, the INFO-COUNT field is used so that the RLC_INFO_ACK can be associated to the corresponding request of a single Info Transfer procedure. When a MT or AP/CC sends a RLC_INFO_ACK message, it shall use the same INFO_COUNT as found in the RLC_INFO. When a MT or AP/CC sends a RLC_INFO message, it shall:

  - use the same INFO_COUNT as the previous RLC_INFO if it is running an error recovery process (timeout expired to get the RLC_INFO_ACK);

  - decrease the INFO_COUNT by one (compared to the previous RLC_INFO) if it is starting a new Info Transfer procedure.

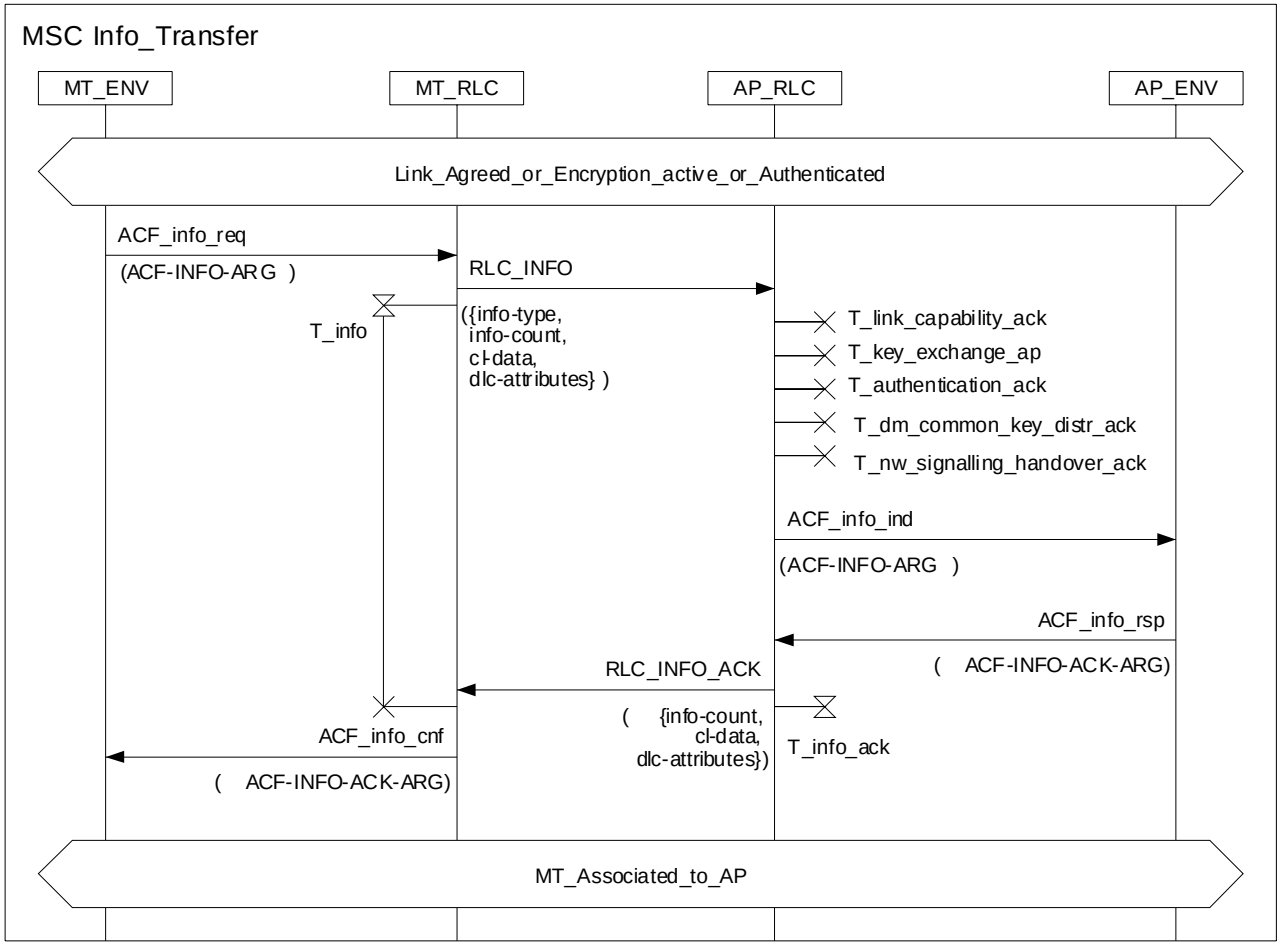


Diagram 23: Info Transfer

Table 25: RLC-INFO

RLC-INFO-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
info-type	INFO-TYPE
info-count	INFO-COUNT
cl-data	CL-DATA OPTIONAL
dlc-attributes	DLC-ATTRIBUTES OPTIONAL }

Table 26: RLC-INFO-ACK

RLC-INFO-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
info-count	INFO-COUNT
cl-data	CL-DATA OPTIONAL
dlc-attributes	DLC-ATTRIBUTES OPTIONAL }

5.1.2 Key Management

5.1.2.1 General

Key management consists of key generation and refresh as well as handling of keys to various BRAN external entities such as key databases. Only those parts of key management that are necessary for interoperability are included in the Hiperlan/2 standard.

The keys used for authentication are long-term keys. They shall be available at both MT and AP side before authentication (if desired) can take place. How to generate, configure, store or fetch the keys (and public key certificates) are outside the scope of HIPERLAN/2 standards.

Keys are needed for unicast and broadcast/multicast encryption. Both types of keys are short-term and can be refreshed. The interval for key refresh is regulated by the local security policy.

The unicast key SSK (Session Secret Key) is a secret key known only to one MT and its connected AP. As the name ("session") implies it is valid for a limited period.

Unicast key generation during initial association is described in clause 5.1.2.5.

5.1.2.2 Unicast Key Refresh (OAP)

SSK should be refreshed regularly to maintain security. When the key lifetime expires, a new SSK key should replace the old one. It is preferable to start the key refresh process before the current key expires.

When the current SSK is going to expire, the AP should send a random value *nonce* encrypted with the current SSK, that is, *SSK(nonce)*, to the MT. Upon receiving, the MT shall decrypt *nonce*, calculate the new encryption key SSK' and send a hash value of the nonce *MD5(nonce)* back to the AP as confirmation. The AP shall verify the hash value to make certain that *nonce* has been correctly received by the MT. After that the AP should send more than one RLC_UNICAST_KEY_ACTIVATE signal to the MT. In this signal the *last-mac-frame* parameter indicates the last MAC frame until which the current SSK is valid. Starting from the next frame, that is the (*last-mac-frame* 1)th frame, both parties shall use SSK'. Both the MT and AP shall compute the *SSK'* as follows:

```
KeyMat = K1 | K2 | K3 | ...
Where
K1 = HMAC-MD5(g^xy mod n, nonce)
            K2 = HMAC-MD5(g^xy mod n, K1 | nonce)
        K3 = HMAC-MD5(g^xy mod n, K2 | nonce)
etc.
K1 contains the most significant bits of the concatenated KeyMat.
```

g^{xy} mod n is the Diffie-Hellman secret obtained during the last execution of the encryption startup procedure.

The SSK' shall be derived from the KeyMat as described in clause 5.1.2.5.

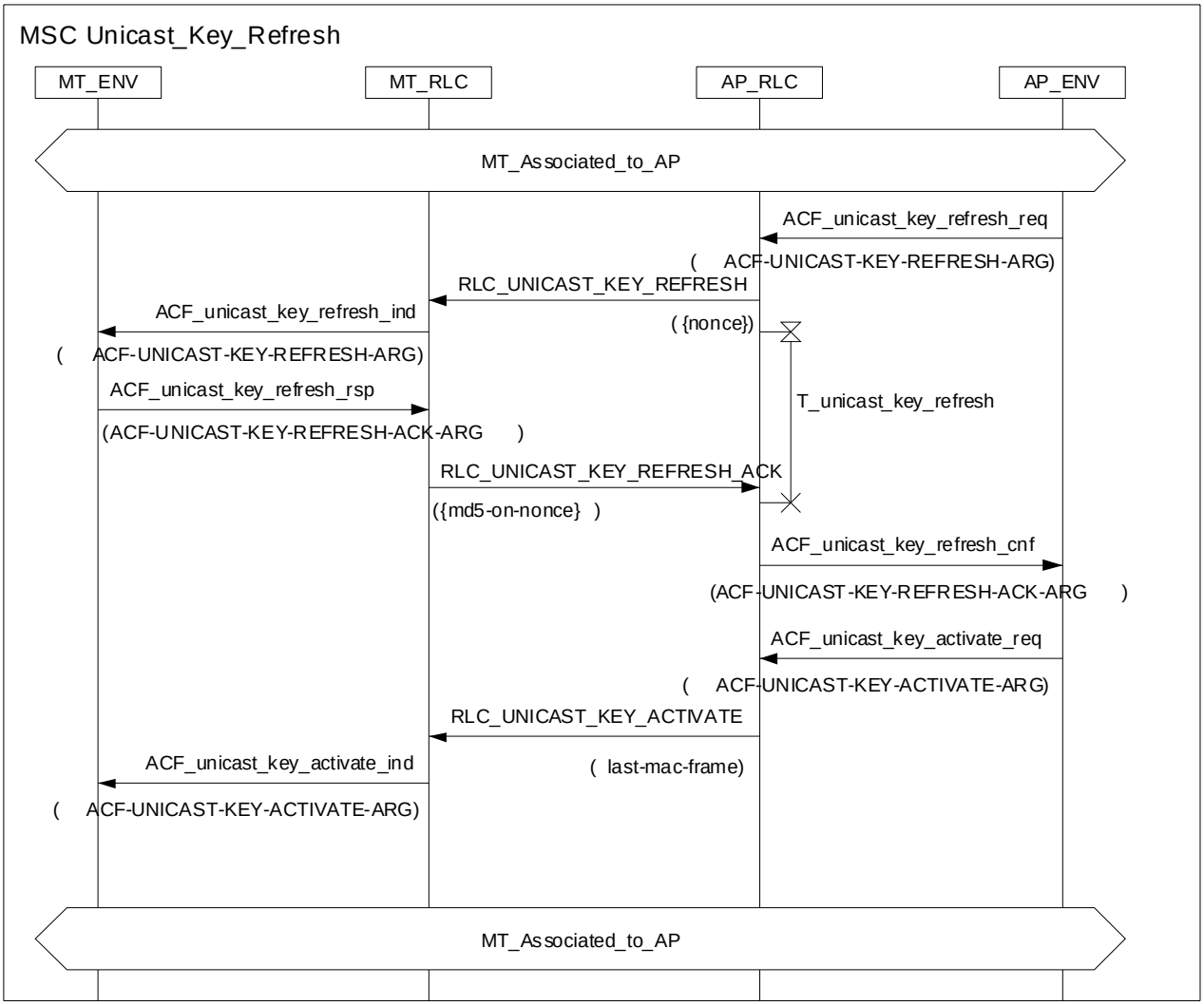


Diagram 24: Unicast Key Refresh

Table 27: RLC-UNICAST-KEY-REFRESH

RLC-UNICAST-KEY-REFRESH-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
nonce	NONCE }

Table 28: RLC-UNICAST-KEY-REFRESH-ACK

RLC-UNICAST-KEY-REFRESH-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
md5-on-nonce	MD5-ON-NONCE }

Table 29: RLC-UNICAST-KEY-ACTIVATE

RLC-UNICAST-KEY-ACTIVATE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
last-mac-frame	LAST-MAC-FRAME}

5.1.2.3 Common keys (OAP/OMT)

5.1.2.3.1 General

Common keys are used for encryption of multicast and broadcast user data and are used if encryption is chosen in the join procedure. Common keys are also used for the encryption of DiL data. Every common key shall be associated with an identifier (*key-id*).

The AP shall generate and distribute common keys to MTs. One common key may be used for broadcast only, for one multicast group only, for more than one multicast group, or for both broadcast and multicast. If the encryption is chosen to be used, the AP shall encrypt multicast/user broadcast data with a common key using a chosen algorithm. It is assumed that all MTs in the cell support this encryption algorithm. The decision for the algorithm shall be made by the AP.

Common keys shall be generated by APs on a stand-alone basis. A random number generator should be used to produce the KeyMat.

NOTE: The random number generator should produce random numbers with good characteristics to get a secure system. Implementers should strive for good random properties for the random generator output (see bibliography, "Applied cryptography Second Edition" and "Randomness Recommendations for Security" for further information about this subject).

A common key shall be derived from the KeyMat as described in clause 5.1.2.5.

5.1.2.3.2 DM Common Key Distribution (OAP/OMT)

The DM Common Key Distribution is introduced in clause 5.1.1.7.

5.1.2.3.3 Common Key Refresh (OAP)

When it is time to refresh a common key, the AP should generate a new common key as described in clause 5.1.2.3.1. The AP shall send the new key to every related MT encrypted with the respective unicast key. The key identifier shall tell MTs which common key is to be updated. Each MT shall decrypt and send back a hash value as confirmation.

After receiving confirmation from all the related MTs in the cell, the AP should send out more than one RLC_COMMON_KEY_ACTIVATE broadcast message to activate the common key. The *last-mac-frame* parameter indicates the last MAC frame until which the old common key is valid. Starting from the next frame, that is the (*last-mac-frame* + 1)th frame, the new common key shall be used. If the key identifier contained in the activation message is unknown to the MT, the MT shall ignore the message.

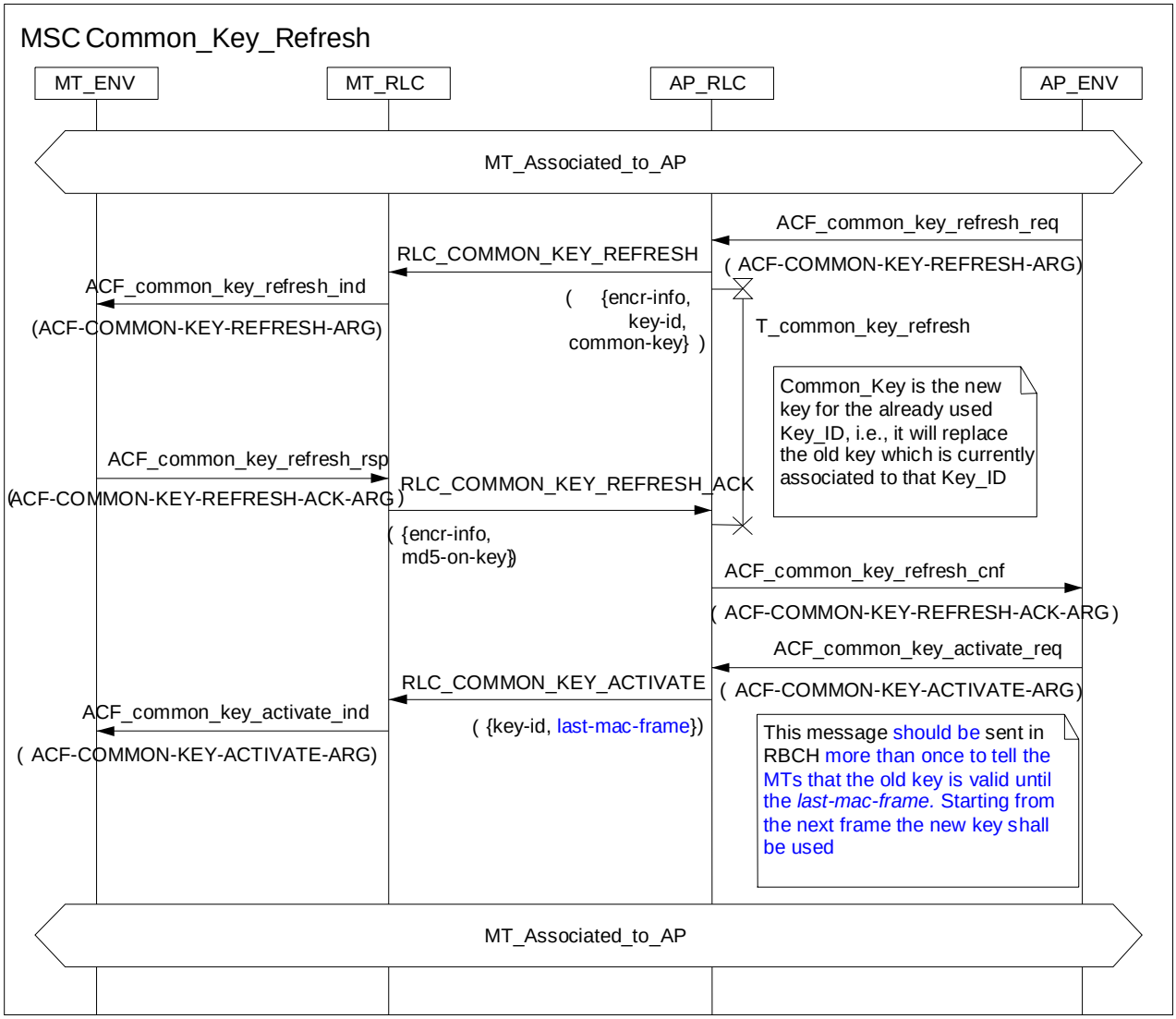


Diagram 25: Common Key Refresh

Table 30: RLC-COMMON-KEY-REFRESH

RLC-COMMON-KEY-REFRESH-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
encr-info	ENCR-INFO
key-id	KEY-ID
common-key	COMMON-KEY }

Table 31: RLC-COMMON-KEY-REFRESH-ACK

RLC-COMMON-KEY-REFRESH-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
encr-info	ENCR-INFO
md5-on-key	MD5-ON-KEY }

Table 32: RLC-COMMON-KEY-ACTIVATE

RLC-COMMON-KEY-ACTIVATE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
key-id	KEY-ID
last-mac-frame	LAST-MAC-FRAME }

5.1.2.4 RBCH-Seed Transfer

Seed shall be used to support IV generation that is needed for encryption. A seed shall be transferred to the MT in the RBCH. The following rules apply to handling of the seed transfer:

1) The seed shall be sent repeatedly in each Nth frame when the AP supports encryption. The AP may send the seed also whenever needed.

2) The seed transmission shall be synchronized with the used sleep cycles.

3) A MT that intends to use encryption shall from the start of the RLC_LINK_CAPABITY message (in both Association and NW Handover) try to update its internal seed generator. The AP shall provide at least one seed value during the Link Capability/HO Link Capability and Encryption Start-up/Token-Signalling procedures during Association and Network Handover. The value for the repetition of seed (N) is vendor specific.

4) The MT shall receive a seed before starting the procedure that follows encryption start up/Token Signalling.

NOTE: Data will be lost during N frames if a residual error occurs at reception of the seed in the MT. Care should therefore be taken to avoid too high N values.

5.1.2.5 Encryption key calculations

5.1.2.5.1 Diffie-Hellman Key Exchange

A Diffie-Hellman Key Exchange [22] shall be carried out in the association phase to generate the unicast encryption key. The two parameters required by the Diffie-Hellman (DH) key exchange, a base generator *g* and a big prime number *n*, shall be pre-configured to the following value in both MT and AP:

```
g = 2
    n = 2^768 - 2^704 - 1 + 2^64* {[2] + 149686} (Oakley group 1, see [22])
```

The hexadecimal value of *n* is:

```
FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1 29024E08 8A67CC74
020BBEA6 3B139B22 514A0879 8E3404DD EF9519B3 CD3A431B 302B0A6D F25F1437
4FE1356D 6D51C245 E485B576 625E7EC6 F44C42E9 A63A3620 FFFFFFFF FFFFFFFF
```

During the Diffie-Hellman Key Exchange, both MT and AP shall generate a random number as their DH private value, say MT has *x* and AP has *y*. The MT shall calculate its DH public value $MtDhPublicValue = g^x \bmod n$ and send it to the AP. Likewise the AP shall calculate $ApDhPublicValue = g^y \bmod n$ and send it to the MT. After that both the MT and AP shall calculate the keying material as described in the following clauses.

NOTE: The Diffie-Hellman key exchange relies on random numbers with good characteristics to get a secure system. Implementers should strive for good random properties for the random number generator output (see Bibliography, "Applied cryptography Second Edition" and "Randomness Recommendations for Security" for further information about this subject).

The MT shall calculate the relevant encryption key before sending the first message after the Encryption Startup procedure.

5.1.2.5.2 DES Key Calculation

For DES [1] encryption,
KeyMat = HMAC-MD5(g^{xy} mod n, 0 x 00), (HMAC-MD5 is explained in clause 5.1.2.6.1)

in which the first (most significant) 8 octets shall be taken out as the session key *SSK*. The least significant bit in each octet of *SSK* shall be the parity bit, and it shall be set according to [2].

The *SSK* shall be checked against all weak and semi-weak keys (keys that have a dual) [2] known to the DES algorithm. If *SSK* is weak or semi-weak, a new *KeyMat* shall be generated by increasing the second parameter in the HMAC-MD5 function, that is, compute:

```
KeyMat = HMAC-MD5(g^xy mod n, 0x 01)
KeyMat = HMAC-MD5(g^xy mod n, 0x 02)
…
```

until a non-weak and non-semi-weak key is obtained.

5.1.2.5.3 Calculation of Triple DES keys (OAP/OMT)

The key used for the first DES encryption module (that receives the IV) is called key1, the key for the second DES decryption module is called key2 and the key for the last DES encryption module is called key3.

```
KeyMat = K1 | K2
```

Where:

```
K1 = HMAC-MD5(g^xy mod n, 0x 00)
K2 = HMAC-MD5(g^xy mod n, K1 | 0x 00),
```

K1 contains the most significant bits of the concatenated *KeyMat*.

The first (most significant) 8 octets of *KeyMat* shall be taken as *key1*, the next 8 octets shall be taken as *key2* and the next 8 octets as *key3*. The least significant bit in each octet of the keys shall be the parity bit. The parity bits shall be calculated according to [2].

For the three keys (*key1, key2 and key3)*, the following checks shall be performed:

- check each of them against all weak and semi-weak keys known to the DES algorithm [2];
- check that all three keys are different.

If any of the checks above fails a new *KeyMat* is generated by increasing the second parameter in the HMAC-MD5 function, that is, compute:

```
KeyMat = K1 | K2
```

Where:

```
K1 = HMAC-MD5(g^xy mod n, 0x 01)
K2 = HMAC-MD5(g^xy mod n, K1 | 0x 01)

KeyMat = K1 | K2
```

Where:

```
K1 = HMAC-MD5(g^xy mod n, 0x 02)
K2 = HMAC-MD5(g^xy mod n, K1 | 0x 02)
…
```

until non-weak, non-semi-weak and different keys are obtained.

5.1.2.5.4 Unicast Key Generation at Network Handover (Mandatory if Handover supported)

The network handover procedure is described in clause 5.2.1.3.

A new SSK for unicast encryption shall be generated at the end of each successful network handover. Both the MT and new AP shall calculate a new KeyMat as follows:

```
KeyMat = K1 | K2 | K3 | ...
```

Where:

```
K1 = HMAC-MD5(g^xy mod n, nonce)
K2 = HMAC-MD5(g^xy mod n, K1 | nonce)
K3 = HMAC-MD5(g^xy mod n, K2 | nonce)
```

etc.

```
K1 contains the most significant bits of the concatenated KeyMat.
```

in which,

g^{xy} mod n is the earlier established Diffie-Hellman secret, nonce is a random value contained in the handover token.

The new SSK shall be derived from the KeyMat as described in clause 5.1.2.5.

Nonce + 1, nonce + 2,..., shall be used as the second parameter in the HMAC-MD5 calculation if the first key check fails. The value is increased by one until the key check procedure gives a successful result.

5.1.2.6 Authentication functions

5.1.2.6.1 Algorithms

MD5 and HMAC are mandatory to implement, since DES encryption is mandatory to implement, whereas RSA is optional to implement. All are optional to use.

MD5 (Message Digest 5), [6] is a one-way hash algorithm developed by RSA Data Security, Inc. It can be used to hash an arbitrary-length byte string into a 128-bit value.

HMAC (Message Authentication with keyed Hashing) [7] is a mechanism for message authentication using keyed hash algorithms. HMAC-MD5 works as follows:

HMAC-MD5(k, m) = MD5((k XOR opad) | MD5((k XOR ipad) | m))

in which *k* is the secret key, *m* is the message, *ipad* (inner padding) is 0 x 36 repeated 64 times, *opad* (outer padding) is 0 x 5c repeated 64 times, "XOR" is exclusive OR, and '|' is concatenation.

RSA (see bibliography, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM") is a public-key algorithm. It can be used for digital signatures. Assume a user A has a pair of RSA keys, a private key $PrivKey_A$ and a public key $PubKey_A$. A can digitally sign a message *m* by first hashing *m* and then encrypting the hash value with $PrivKey_A$. Anyone can verify A's signature by decrypting it with $PubKey_A$ and then checking the hash value.

5.1.2.6.2 Authentication protocols

In the association phase, if authentication is chosen, it shall be performed after the Diffie-Hellman key exchange (see Encryption Startup MSC).

NOTE: To detect man-in-the-middle attacks that, over the radio link, are difficult but yet potentially possible to launch with DH exchange, the DH public values exchanged are linked to the authentication procedure. Proposed and selected encryption and authentication alternatives are also included to combat attacks aiming for a lower security level than requested.

The authentication protocol shall be negotiated between the AP and the MT. Five alternatives are specified:

1) Pre-shared key based;
2) RSA512 (OAP/OMT);
3) RSA768 (OAP/OMT);
4) RSA1024 (OAP/OMT);
5) No authentication.

If authentication is chosen, either pre-shared key based or public key (RSA) based, a challenge-response scheme shall fulfil the task. The challenge should be a random number sent by the verifier to the claimant. The claimant shall calculate a response using a function with the challenge as input. The function to use depends on which alternative (1 or 2-4) is chosen.

If the MT authentication succeeds, the MT is allowed to access the network and the association procedure should continue. Otherwise the MT shall be rejected and the DLC connection between the MT and AP shall be terminated. The MT may also terminate the access attempt if the AP authentication fails.

NOTE: The challenge-response scheme needs a random number with good characteristics to get a secure system. Implementers should strive for good random properties for the random number generator output (see bibliography for further discussion on this subject).

5.1.2.6.3 Pre-shared key based authentication

In this alternative a party is authenticated if it has knowledge of the pre-shared key. In this case, the key shall be generated and distributed to the communicating parties in advance.

NOTE: Because of key management overhead, this solution is primarily applicable to business and residential environment with a limited number of users and administrative domains.

The response shall be calculated as follows when pre-shared keys are used:

- *mt-response* = HMAC-MD5(PresharedKey, MtAuthenticationString):

- if encryption is chosen, i.e. Encryption Startup has preceded the authentication procedure,

MtAuthenticationString = challenge_to_mt | mt_dh_public_value | ap_dh_public_value | authentication_encryption_list | auth_encr_selected.

Otherwise:

MtAuthenticationString = challenge_to_mt | authentication_encryption_list | auth_encr_selected

ap-response = HMAC-MD5(PresharedKey, APAuthenticationString).

If encryption is chosen, i.e. Encryption Startup has preceded the authentication procedure,

APAuthenticationString = challenge_to_ap | mt_dh_public_value | ap_dh_public_value | authentication_encryption_list | auth_encr_selected

Otherwise:

APAuthenticationString = challenge_to_ap | authentication_encryption_list | auth_encr_selected

MtAuthenticationString and APAuthenticationString are concatenations of a few items. These items shall be concatenated in the same order (from left to right) as shown above. Each item shall be appended with the most significant bit first:

- *challenge-to-mt*: the challenge sent by AP to MT, 128 bit long;

- *challenge-to-ap*: the challenge sent by MT to AP, 128 bit long;

- *mt-dh-public-value*: MTs Diffie-Hellman public value, 768 bit long;

- *ap-dh-public-value*: AP's Diffie-Hellman public value, 768 bit long;

- *authentication-encryption-list*: the list of authentication-encryption alternatives proposed by MT during the link capability phase, n x 8 bit long (n is the number of authentication-encryption alternatives in the list);

- *auth-encr-selected*: the authentication-encryption alternative selected by AP during the link capability phase, 8 bit long.

The length of the pre-shared key should be at least 128 bits, see RFC 2104 [7].

5.1.2.6.4 Public key based authentication (OAP/OMT)

To use public-key cryptography, the most important thing is to assure the authentic binding between a public key and its owner. A public-key certificate signed by some trusted authority is an effective means to distribute public keys securely. A public key infrastructure (PKI) provides mechanisms to issue, fetch, verify or revoke public-key certificates.

In this alternative a party is authenticated if it has the ability to correctly generate a digital signature. The signature and verification calculations shall be done as defined in [PKCS#1] using MD5 as hash algorithm. The response shall be calculated as follows when a public key solution is used:

- *mt-response* = RSASSA-PKCS-V1_5-SIGN(MtPrivKey,MtAuthenticationString);
- *ap-response* = RSASSA-PKCS-V1_5-SIGN(ApPrivateKey,ApAuthenticationString).

MtAuthenticationString and ApAuthenticationString are the same as described in clause 5.1.2.6.3.

Three public key lengths are supported: 512, 768 and 1 024 bits.

5.1.3 Disassociation

Disassociation is for releasing MT's association to a particular AP. There are two types of disassociation, 1) explicit and 2) implicit.

1) In the explicit disassociation either the MT or the AP initiates the disassociation. The initiating peer (either MT or AP) shall send the RLC_DISASSOCIATION message and the other peer shall respond with RLC_DISASSOCIATION_ACK message. After sending the RLC_DISASSOCIATION message or when receiving the same from MT, the AP should release the resources for the MT.

2) The Implicit disassociation should take place when the MT and the AP have lost the ability to communicate via the radio interface. In this case, the MT_Alive process shall notice that the MT or/and AP can not be reached and the resources for the MT should be released by the AP.

NOTE: In the MT Initiated Disassociation, the MT may not wait for the RLC_DISASSOCIATION_ACK.

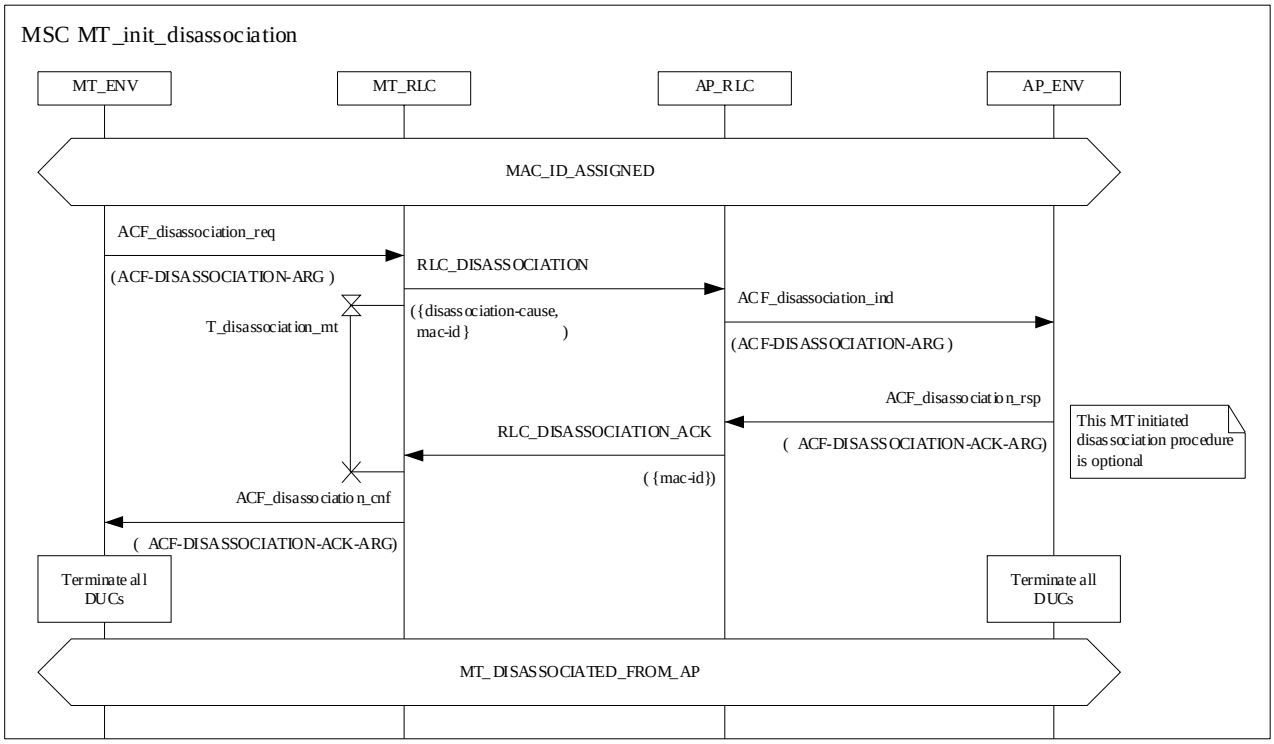


Diagram 26: MT Initiated Disassociation

Table 33: RLC-DISASSOCIATION

RLC-DISASSOCIATION-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
disassociation-cause	DISASSOCIATION-CAUSE
mac-id	MAC-ID }

Table 34: RLC-DISASSOCIATION-ACK

RLC-DISASSOCIATION-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
mac-id	MAC-ID -- (if Uplink) -- }

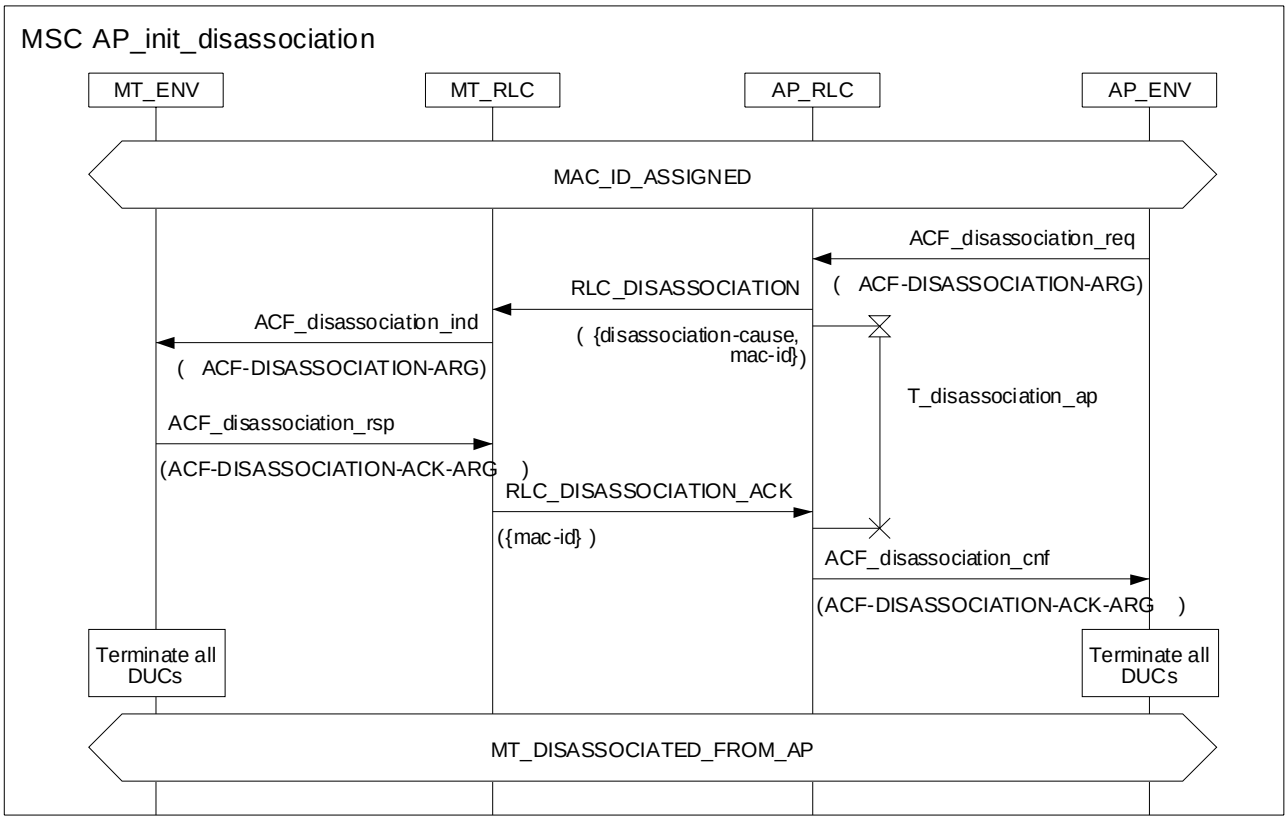


Diagram 27: AP Initiated Disassociation

5.1.4 Multicast (OAP/OMT)

In order to join multicast groups the MT shall first have to be associated to the AP as an individual and the MT should also have made connection set-ups for unicast traffic. If individual connections are not set up before the group join attempt, the AP cannot use the n x unicast method.

There shall be two ways of implementing multicast: using multicast MAC ID and transmitting the information once to a multicast group over the air and n times unicast where the information is transmitted individually to each member of the group.

The multicast MAC ID method shall use multicast MAC IDs from the range 224 to 255.

The MT shall initialize the group joining by sending the RLC_GROUP_JOIN message to the AP. The MT shall use a higher layer address as the group identifier in the request. The MT shall also propose the encryption algorithms it wants to use.

The AP shall acknowledge the request by sending MAC IDs and the selected encryption algorithm and keys for each group. Each RLC_GROUP_JOIN_ACK message shall have exactly one encryption algorithm, one common key and one key id and all HL addresses and all MAC-Ids having those security parameters. Other groups with other security parameters shall use other RLC_GROUP_JOIN_ACK messages.

The MAC ID shall be either a Multicast MAC ID from the permitted range or the individual MAC ID that the MT is already using. If it is the MAC ID that the MT is already using, then the multicast traffic shall use one of the DLCC-IDs that the MT is already using. If it is a multicast MAC ID, then the DLCC-ID 63 shall be used.

How the AP decides on the method, multicast or n x unicast, is not a part of the present document.

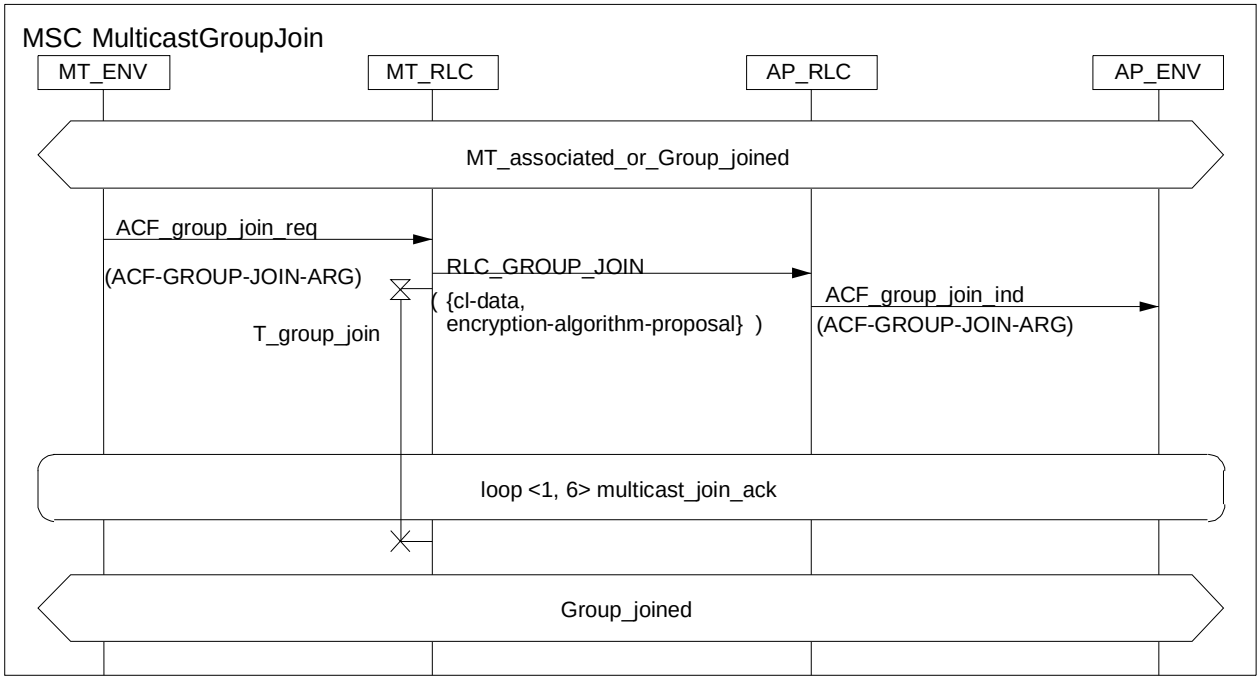


Diagram 28: Multicast Group Join

Table 35: RLC-GROUP-JOIN

RLC-GROUP-JOIN-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-data	CL-DATA
encryption-algorithm-proposal	ENCRYPTION-ALGORITHM-PROPOSAL }

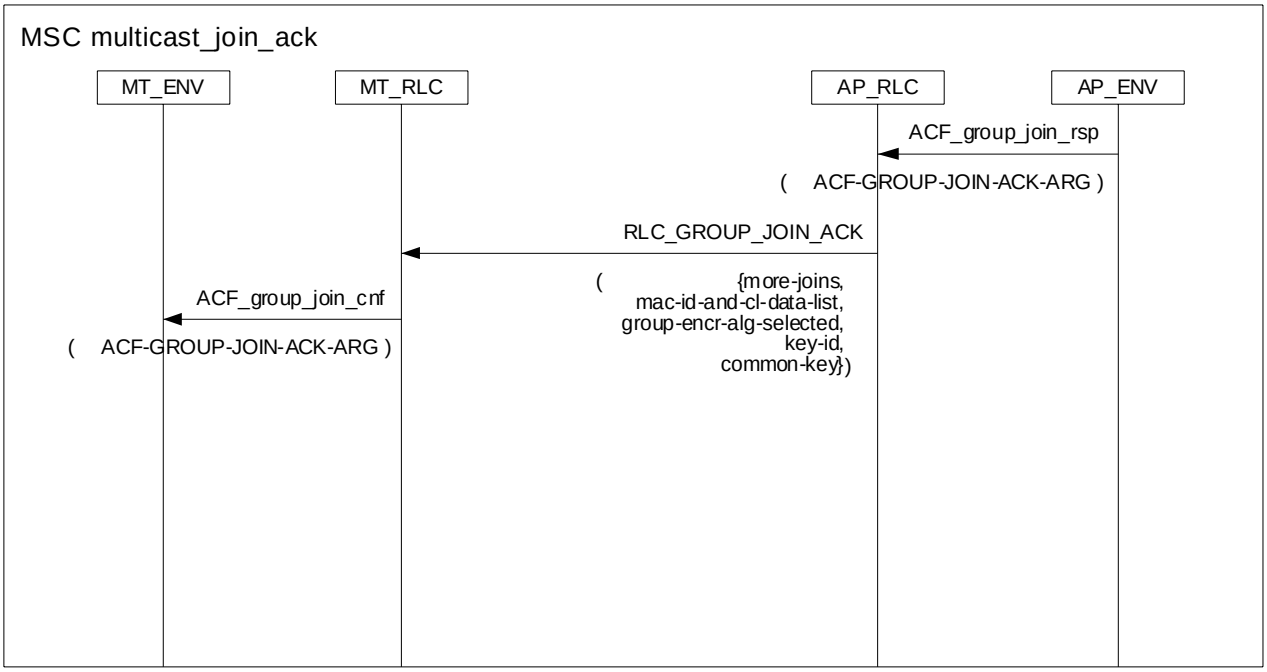


Diagram 29: Multicast Group Join Ack

Table 36: RLC-GROUP-JOIN-ACK

RLC-GROUP-JOIN-ACK-ARG::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
more-joins	MORE-JOINS
mac-id-and-cl-data-list	MAC-ID-AND-CL-DATA-LIST
encryption-algorithm-selected	ENCR-INFO
key-id	KEY-ID
common-key	COMMON-KEY }

Table 37: RLC-GROUP-JOIN-NACK

RLC-GROUP-JOIN-NACK-ARG::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
more-joins	MORE-JOINS
cl-data CL-DATA }	

When a MT leaves a group, it shall send a group-leave-request to the AP. The MT shall use the higher layer address as the group identifier. The AP shall acknowledge the request positively and again use the higher layer address as an identifier.

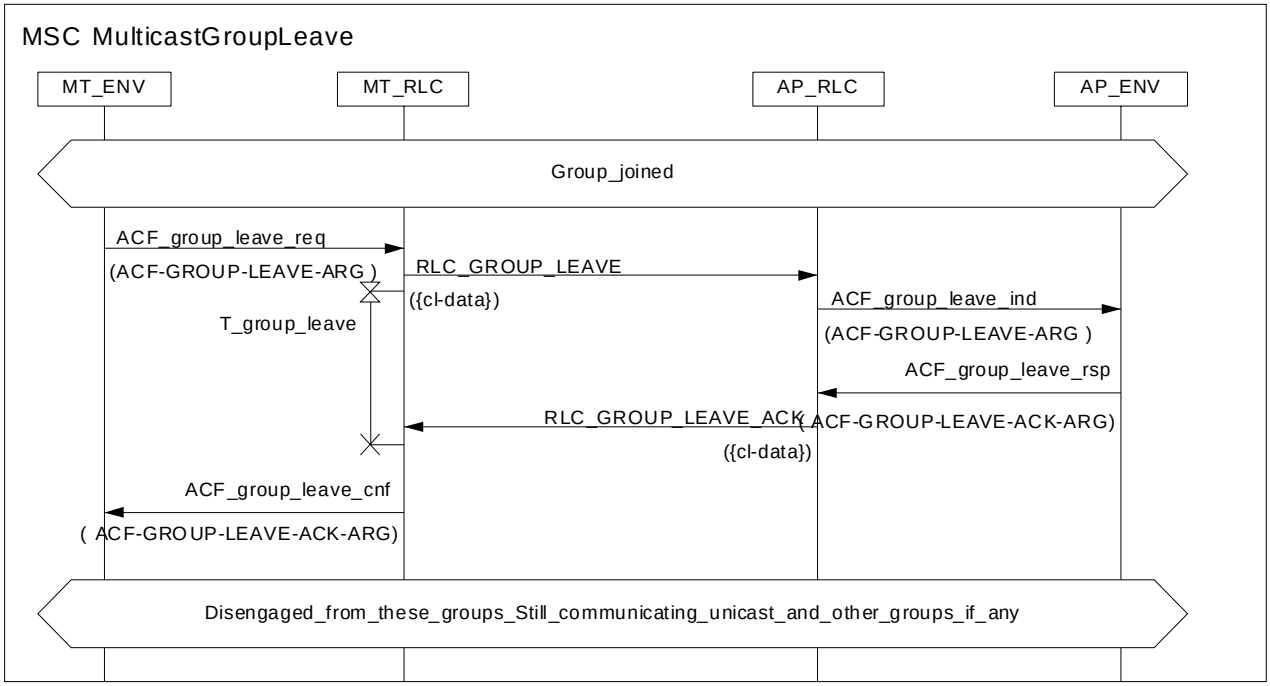


Diagram 30: Multicast Group Leave

Table 38: RLC-GROUP-LEAVE

RLC-GROUP-LEAVE-ARG::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-data	CL-DATA }

Table 39: RLC-GROUP-LEAVE-ACK

RLC-GROUP-LEAVE-ACK-ARG::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-data	CL-DATA }

5.1.5 CL Broadcast (OAP/OMT, depends on CL)

In order to join CL (user) broadcast groups the MT shall be first associated to the AP as an individual.

The MT shall initialize the CL broadcast group joining by sending the RLC_CL_BROADCAST_JOIN message to the AP. The MT shall use a higher layer address as the broadcast group identifier in the request. The MT shall also propose the encryption algorithms it wants to use.

The AP shall acknowledge the request by sending MAC IDs, HL-addresses, the selected encryption algorithm and keys for each group.

The broadcast MAC ID can be any free MAC ID except the one used for RBCH signalling (MAC ID = 0) and the multicast MAC IDs from the range 224 to 255. The DLCC-ID value 63 shall be used for all broadcast traffic. The MAC-ID selected by the AP for broadcast transmission shall not be the one that is already being used for unicast transmission.

The UBCH [5] shall be used for CL broadcast.

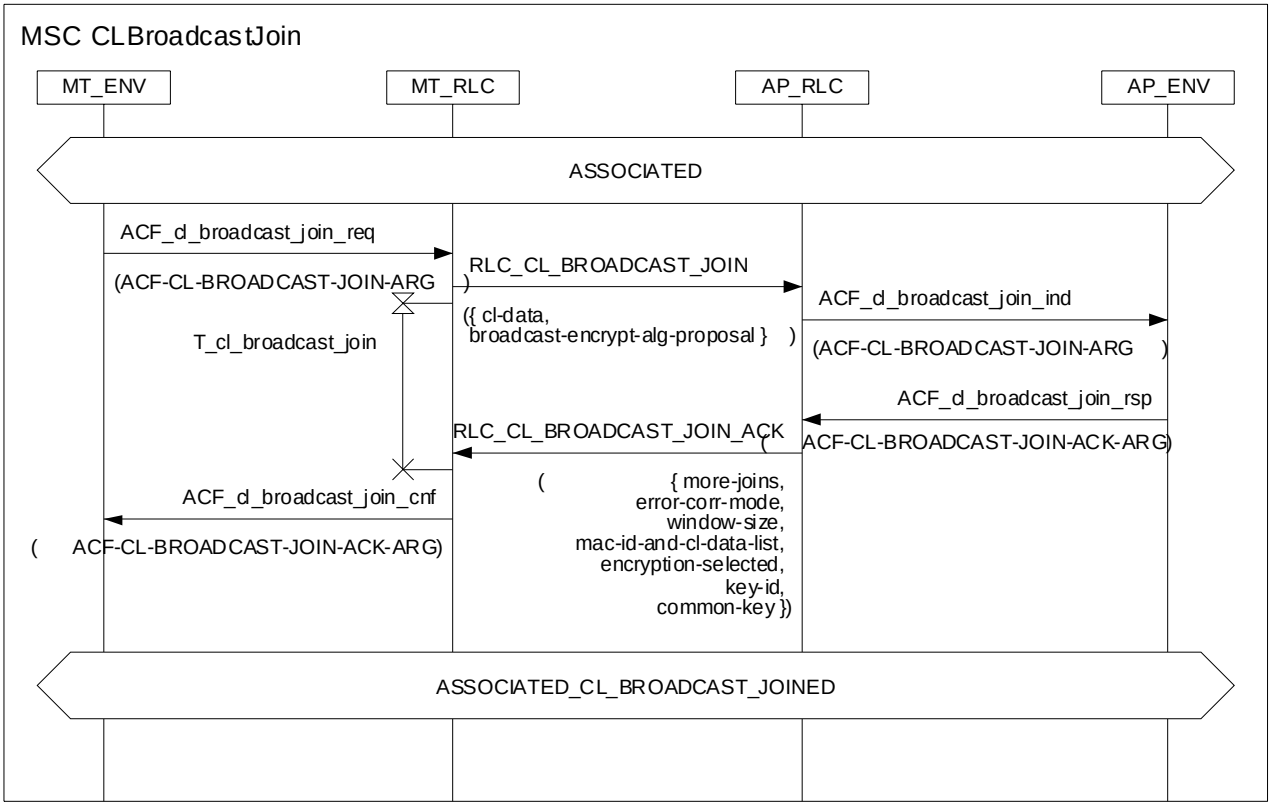


Diagram 31: CL Broadcast Group Join

Table 40: RLC-CL-BROADCAST-JOIN

RLC-CL-BROADCAST-JOIN-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-data	CL-DATA
encryption-algorithm-proposal	ENCRYPTION-ALGORITHM-PROPOSAL }

Table 41: RLC-CL-BROADCAST-JOIN-ACK

RLC-CL-BROADCAST-JOIN-ACK-ARG::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
more-joins	MORE-JOINS
error-corr-mode	ERROR-CORR-MODE
window-size	BROAD-WINDOW
mac-id-and-cl-data-list	MAC-ID-AND-CL-DATA-LIST
encryption-algorithm-selected	ENCR-INFO
key-id	KEY-ID
common-key	COMMON-KEY }

When a MT leaves a group, it shall send a group-leave-request to the AP. The MT shall use the higher layer address as the group identifier. The AP shall acknowledge the request positively and again use the higher layer address as an identifier.

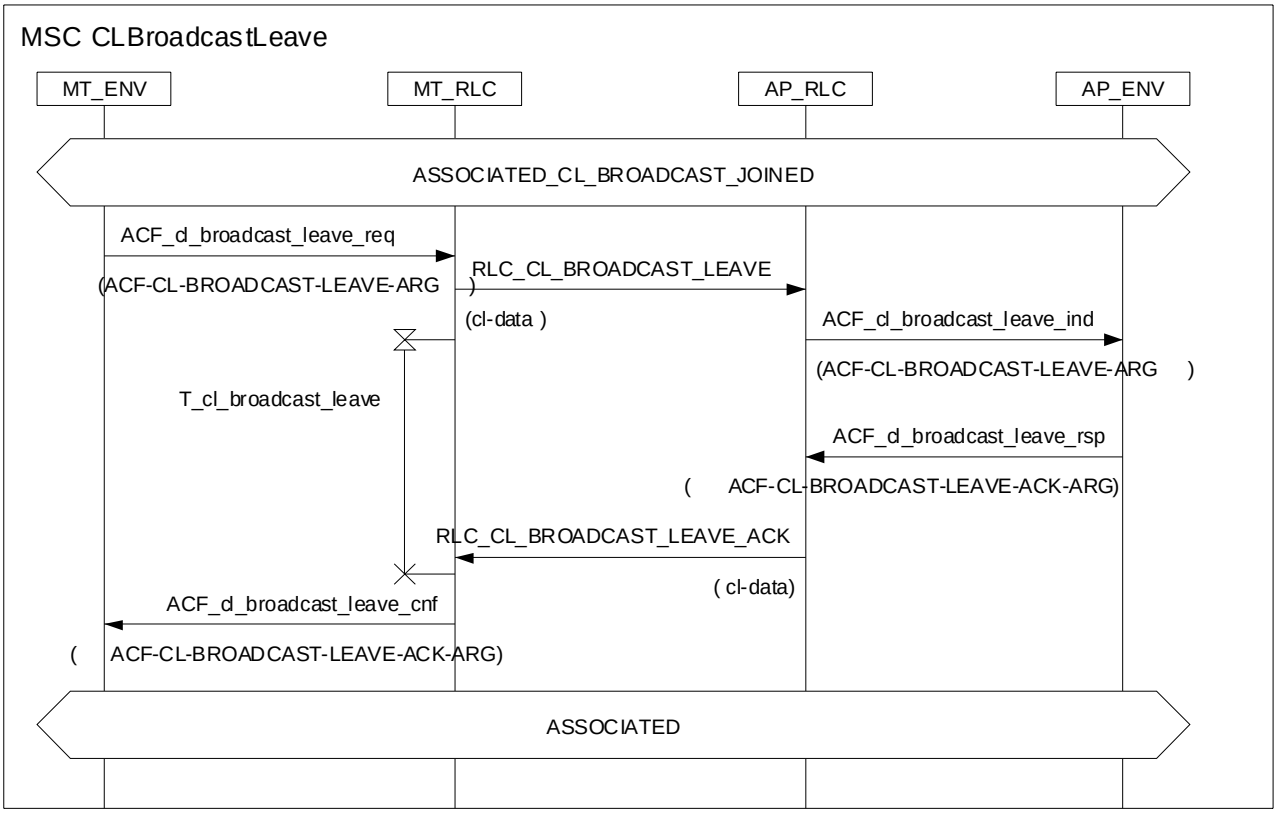


Diagram 32: CL Broadcast Leave

Table 42: RLC-CL-BROADCAST-LEAVE

RLC-CL-BROADCAST-LEAVE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-data	CL-DATA }

Table 43: RLC-CL-BROADCAST-LEAVE-ACK

RLC-CL-BROADCAST-LEAVE-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-data	CL-DATA }

5.1.6 Association Rejection

The rejection of the Association procedure shall follow the next three rules:

1) The information in RBCH Association does not fit with what is expected by the MT. The MT shall not continue the Association procedure.

2) If the AP does not accept the RLC_MAC_ID_ASSIGN message, the AP shall respond with the RLC_MAC_ID_ASSIGN_NACK message.

3) After the MT has got a MAC ID, the rejecting shall be done by sending RLC_DISASSOCIATION message. Both the MT and AP shall use this message for this purpose. The RLC_DISASSOCIATION message shall be used for every type of explicit disassociation during all states of the system, not only during the association phase.

NOTE: Because of varying radio link conditions, the Disassociation may happen implicitly, see clause 5.1.3.

5.2 Services supporting RRC (Radio Resource Control)

5.2.1 Handover (OAP/OMT)

Depending on the MT handover decision three types of handover can be performed:

- Sector Handover (Inter-Sector);
- Radio Handover (Inter-APT/Intra-AP Handover);
- Network Handover (Inter-AP/Intra-Network Handover).

A radio cell shall be controlled by one APC, whereby APCs may support several transceivers (APT) and, hence, serve several radio cells [5].

NOTE 1: The distinction between APC and APT is only relevant for radio handover, where the controlling RLC instance may remain the same, while the MAC instance changes. Therefore, for ease of description, in all sub-clauses of this clause, except 5.2.1.2, this distinction is not referred explicitly.

NOTE 2: When initiating a handover to an adjacent radio cell, the MT may not know if it initiates a radio or network handover. Therefore, a handover supporting MT should support radio as well as network handover procedures.

NOTE 3: Prior to handover execution the MTs should gather relevant measurements on the frequency used by the current AP as well as on the frequencies used by candidate APs for a handover. In order to measure neighbouring APs, the MT may use MT Absence.

NOTE 4: The reason to perform any kind of handover is out the scope of the present document.

5.2.1.1 Sector Handover (OAP/OMT)

Sector antennas are optional for APs. If the AP uses sectors, the AP shall support Sector Handover as shown in (diagram 1).

During the Sector Handover only the antenna sector of the AP shall be changed.

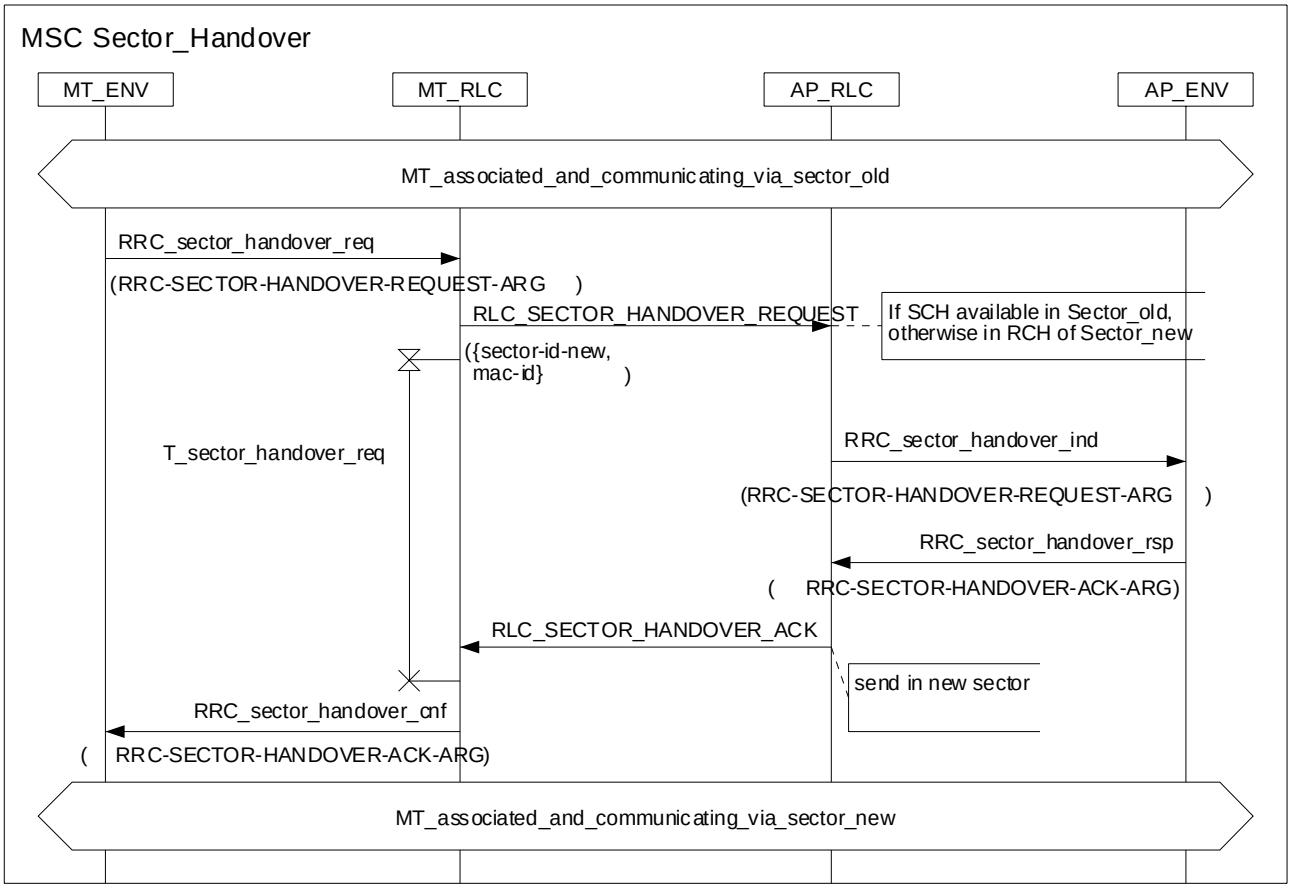


Diagram 33: Sector handover

Table 44: RLC-SECTOR-HANDOVER-REQUEST

RLC-SECTOR-HANDOVER-REQUEST-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
sector-id-new	SECTOR-ID
mac-id	MAC-ID }

Table 45: RLC-SECTOR-HANDOVER-ACK

RLC-SECTOR-HANDOVER-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE }

To perform a sector handover, the MT may send a sector handover request via the old sector, if a SCH is still available and feasible for this purpose (i.e. SCH is not needed for other purposes and communication in the old sector is still possible). If no SCH is available the MT shall send the request in the RCH of the new sector. After sending RLC_SECTOR_HANDOVER_REQUEST the MT shall change to the new sector before the next frame starts. The AP shall respond with the RLC_SECTOR_HANDOVER_ACK message via the new sector (in DCCH).

NOTE: No Reset of ARQ parameters shall be made at sector handover.

5.2.1.2 Radio (intra-AP) Handover (OAP/OMT)

Radio Handover is an optional tool for AP implementations with more than one transceiver per AP (see figure 3).

The following prerequisite is made in order to get the radio handover in the simple way it is described here. The central controller shall have control of both encryption key and seed for all APTs that are controlled by the central controller. The key and the seed shall be identical for all APTs controlled by the controller.

In a multiple transceiver configuration, for each transceiver, a unique AP ID shall be assigned, i.e. multiple AP IDs are assigned to one AP. The AP may provide an address mask during association and handover which indicates the address range of the AP ID dedicated for transceiver identification. The parameter *apt-address-length* conveyed in RLC_LINK_CAPABILITY_ACK and RLC_HANDOVER_LINK_CAPABILITY_ACK messages shall indicate the number of bits assigned for transceiver identification starting from the least significant bit of the AP ID.

NOTE 1: Using this parameter for multiple transceiver APs allows the MT to distinguish between candidate radio cells for radio and network handover, respectively, before initiating the handover procedure.

In case of single transceiver APs the *apt-address-length* shall be set to zero.

Values of *apt-address-length* greater than zero shall be indicated only, when the AP supports radio handover (copy to parameter). In a single coverage area using APs with identical *net-id* the *apt-address-length* shall be set to the same value in all APs.

NOTE 2: According to the above dependencies, the MT may not know if it initiates a radio or network handover, when the corresponding apt-address-length is set to zero, since both handover types are initiated by the MT with the same RLC message.

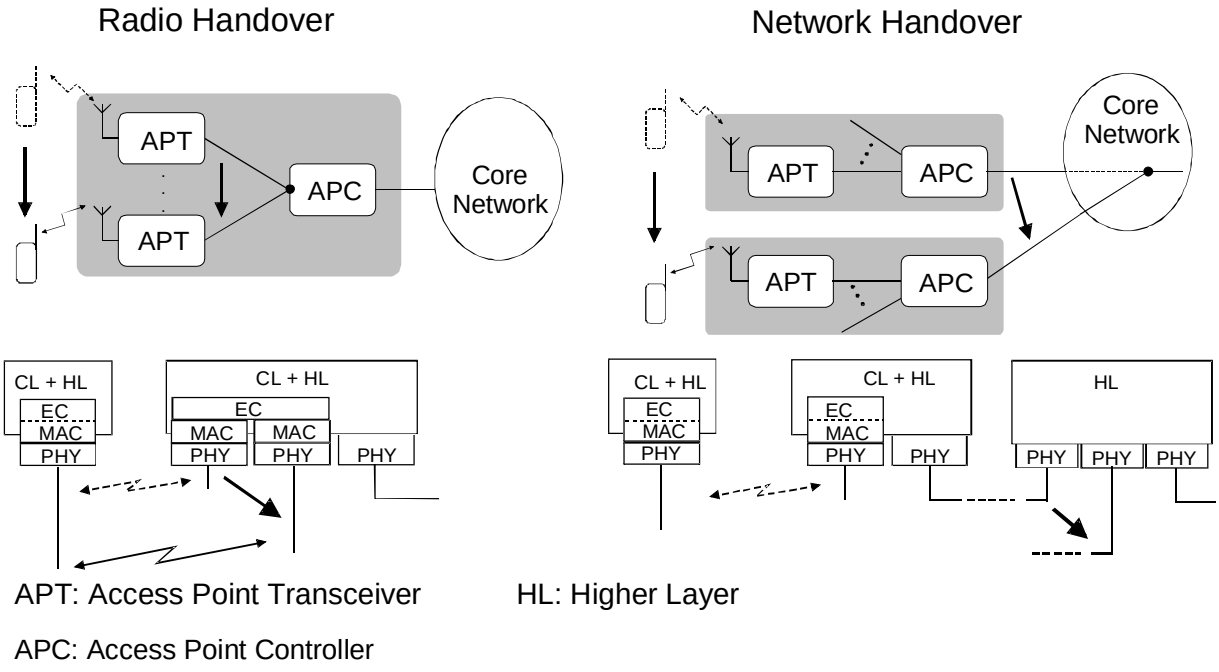


Figure 3: Informative radio and network handover scenarios

NOTE 3: The figure is an example of APT-APC division and does not make any restrictions for implementation.

NOTE 4: In such a configuration, a radio handover may be performed when an associated MT moves from the coverage area of one APT to another, which is served by the same APC. Since the handover execution may be performed within the DLC layer, the Higher Layer Protocols (HL) may not be involved.

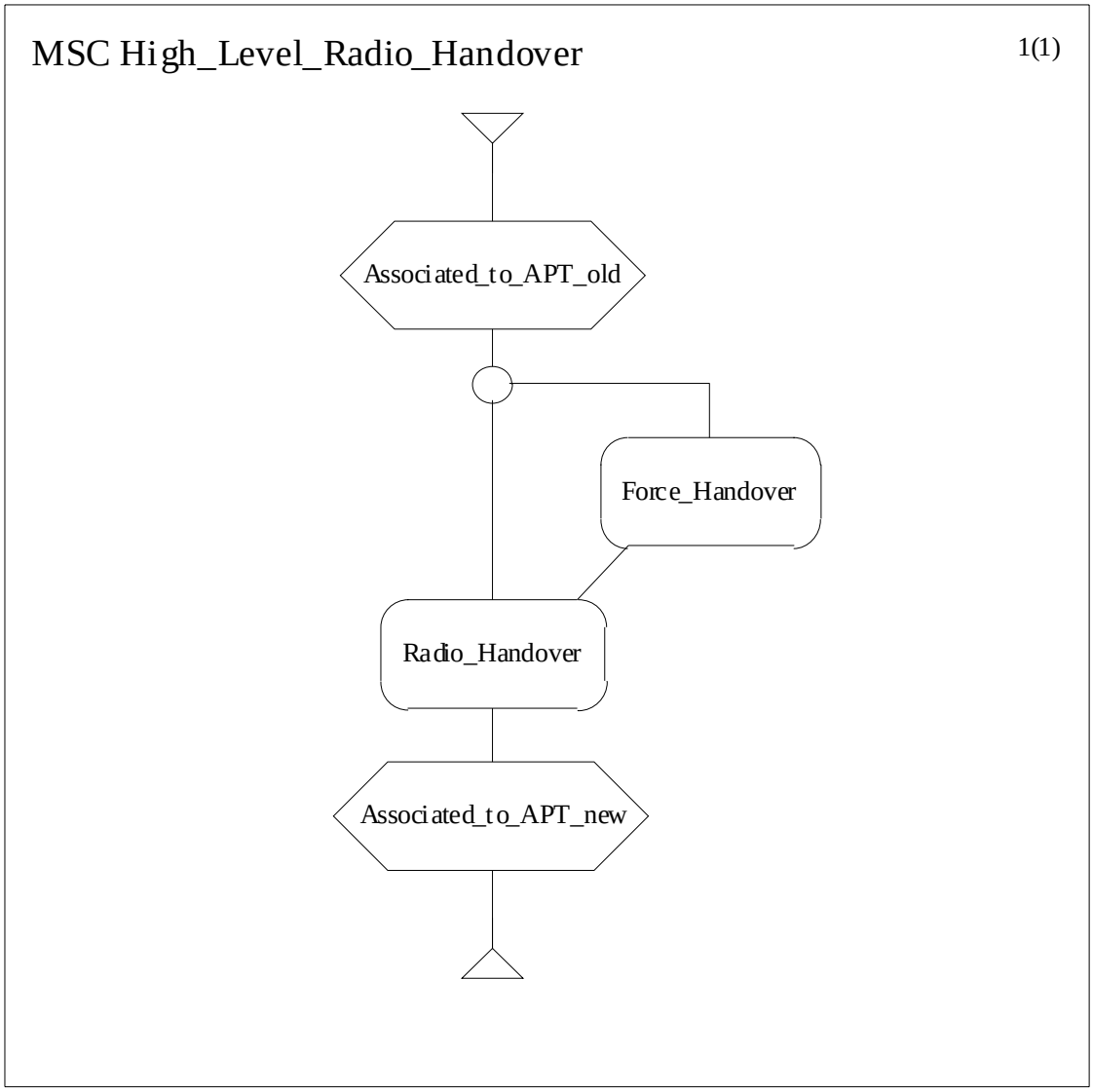


Diagram 34: Radio handover overview

If the MT is still synchronized to the current APT, when triggering a handover to another (target) APT, the MT may notify the AP via the current APT (RLC_HANDOVER_NOTIFY message), that it will perform a handover to another APT.

NOTE 5: This message is only useful in case of network handover, since the AP is implicitly informed, when the MT initiates a radio handover. However, the MT may not be aware of the type of handover it initiates.

The MT may return within 255 frames from the frame, where the RLC_HANDOVER_NOTIFY message was sent (sending frame is number 0). If the MT returns to the current APT it shall notify the AP. In case data waits for transmission in the MT, it sends Resource Requests to the AP and, hence, informs it thereby about its presence. In case no data has to be transmitted, the MT shall trigger the MT-ALIVE procedure.

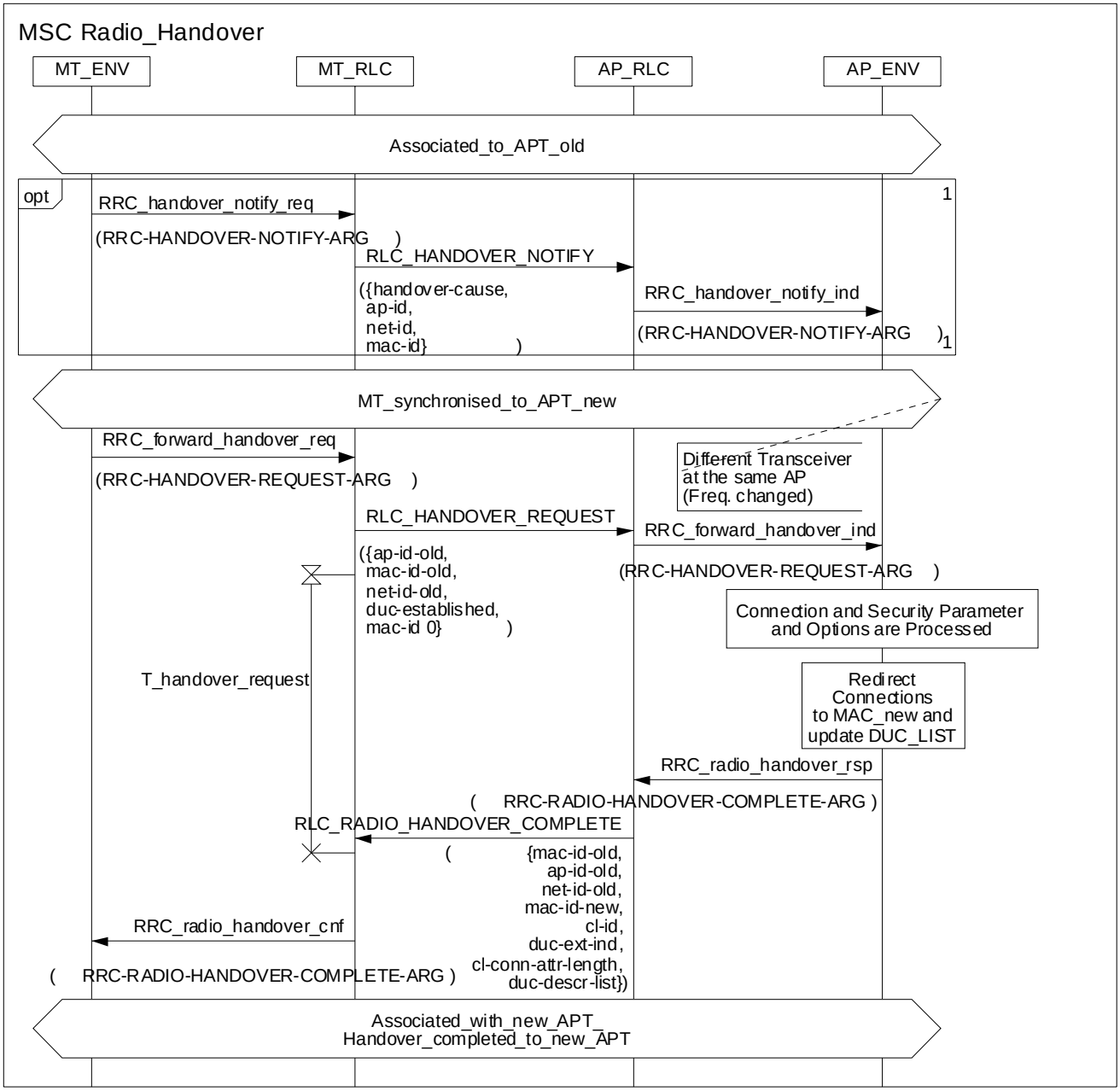


Diagram 35: Radio handover

Table 46: RLC-HANDOVER-NOTIFY (OAP/OMT)

RLC-HANDOVER-NOTIFY-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
handover-cause	HANDOVER-CAUSE
ap-id	AP-ID OPTIONAL
net-id	NET-ID OPTIONAL
mac-id	MAC-ID }

Table 47: RLC-HANDOVER-REQUEST

RLC-HANDOVER-REQUEST-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE,
ap-id-old	AP-ID, -- AP-ID of old APT
mac-id-old	MAC-ID, -- MAC-ID used in old APT
net-id-old	NET-ID
duc-established	DUC-ESTABLISHED }

Table 48: RLC-RADIO-HANDOVER-COMPLETE

RLC-RADIO-HANDOVER-COMPLETE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
mac-id-old	MAC-ID
ap-id-old	AP-ID
net-id-old	NET-ID
mac-id-new	MAC-ID
cl-id	CL-ID
duc-ext-ind	DUC-EXT-IND
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

The MT triggers handover by sending RLC_HANDOVER_REQUEST to the target AP, which shall then select the handover procedure (either radio or network handover).

In case of a radio handover the AP shall respond with RLC_RADIO_HANDOVER_COMPLETE message to approve the request. The AP shall use RBCH for sending this message. MT shall update the unicast DUCs according to RLC_RADIO_HANDOVER_COMPLETE message before sending any Resource Requests. In this message the AP may either confirm the characteristics of the DUCs established in the previous radio cell or modify them.

If not all DUCs, which the AP intends to maintain after handover, can be indicated in RLC_RADIO_HANDOVER_COMPLETE, the AP shall set the duc-ext-ind flag and address the remaining DUCs in subsequent DUC Modify.

DUCs established in a previous radio cell, but addressed neither in RLC_RADIO_HANDOVER_COMPLETE nor in subsequent DUC Modify by the AP shall be considered by the MT as released.

After radio handover AP, i.e. after sending RLC_RADIO_HANDOVER_COMPLETE, and MT, i.e. after receiving RLC_RADIO_HANDOVER_COMPLETE, shall trigger the reset actions for the on-going DUCs in acknowledged mode (reference [5]) only, when the DUCs are modified in RLC_RADIO_HANDOVER_COMPLETE or DUC Modify.

For the rejection of the Radio Handover, see clause 5.2.1.5.

5.2.1.3 Network Handover (OAP/OMT)

A network handover may be carried out when an associated MT moves from one AP to another AP (figure 3). Since the MT leaves the serving area of an RLC instance, a network handover involves also the CL and higher layers (HL). To maintain HL association and connections specific signalling via the fixed network may be needed.

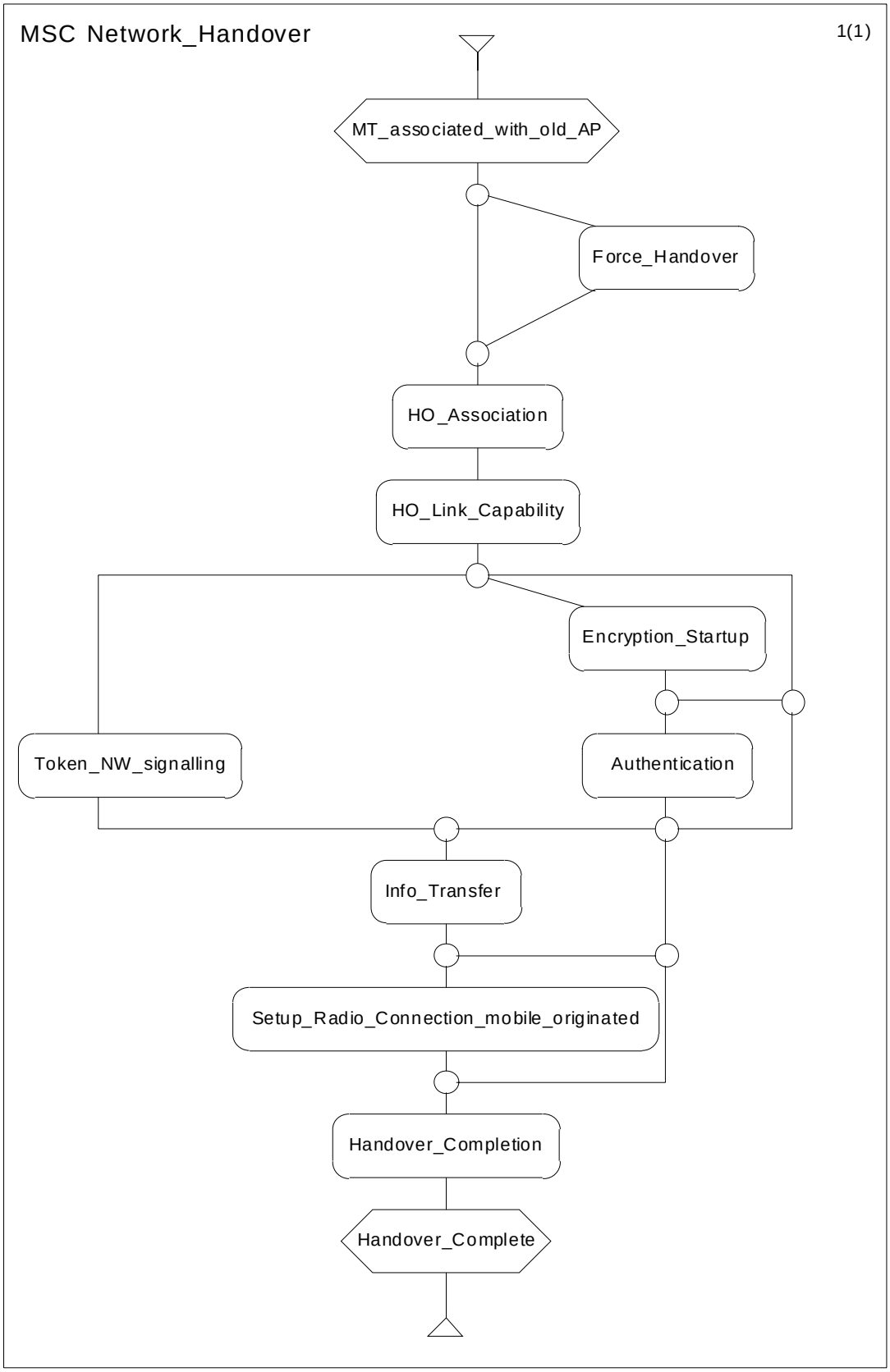


Diagram 36: Network Handover procedure

If the MT is still synchronized to the current AP, when triggering a handover to another (target) AP, it may notify the current AP by the RLC_HANDOVER_NOTIFY message, that it will perform a handover to another AP.

The AP shall stop scheduling data to this MT after receiving RLC_HANDOVER_NOTIFY message.

The MT may return within 255 frames from the frame, where the RLC_HANDOVER_NOTIFY message was sent (sending frame is number 0). If the MT returns to the old AP, the MT shall notify the old AP. In case data waits for transmission in the MT, it sends Resource Requests to the AP and, hence, informs the AP thereby about its presence. In case no data has to be transmitted, the MT shall trigger the MT Alive procedure.

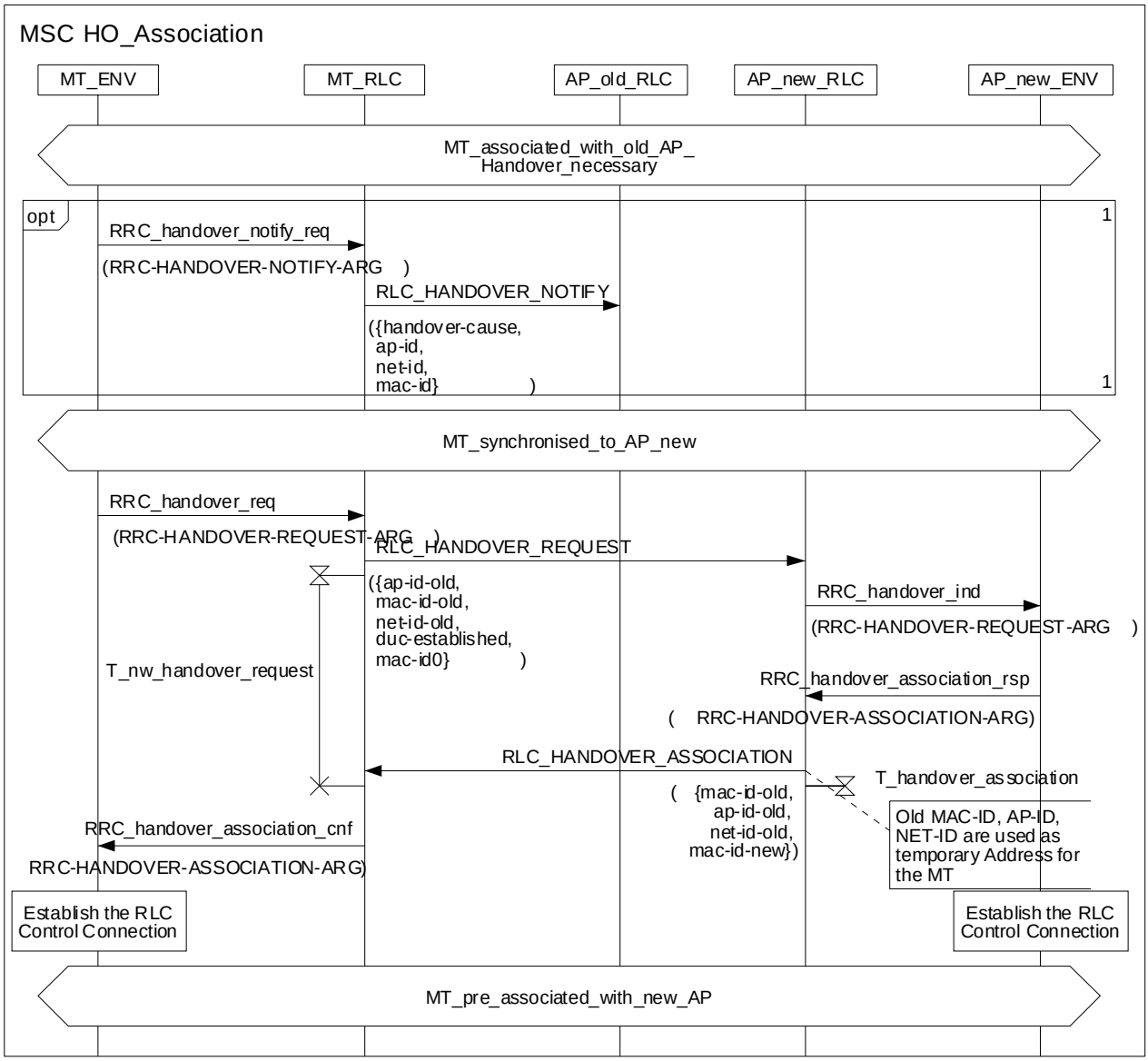


Diagram 37: Handover association procedure

Table 49: RLC-HANDOVER-ASSOCIATION

RLC-HANDOVER-ASSOCIATION-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
mac-id-old	MAC-ID
ap-id-old	AP-ID
net-id-old	NET-ID
mac-id-new	MAC-ID }

The MT shall trigger handover by sending RLC_HANDOVER_REQUEST to the target AP, which shall then select the handover procedure (either radio or network handover). The AP shall respond with the RLC_HANDOVER_ASSOCIATION message to approve the request.

In order to reject the RLC_HANDOVER_REQUEST, the AP shall respond with RCL_HANDOVER_REQUEST_NACK message. The AP shall use RBCH for sending these messages. After receiving RLC_HANDOVER_ASSOCIATION, the MT shall trigger the Handover (HO) link capability procedure.

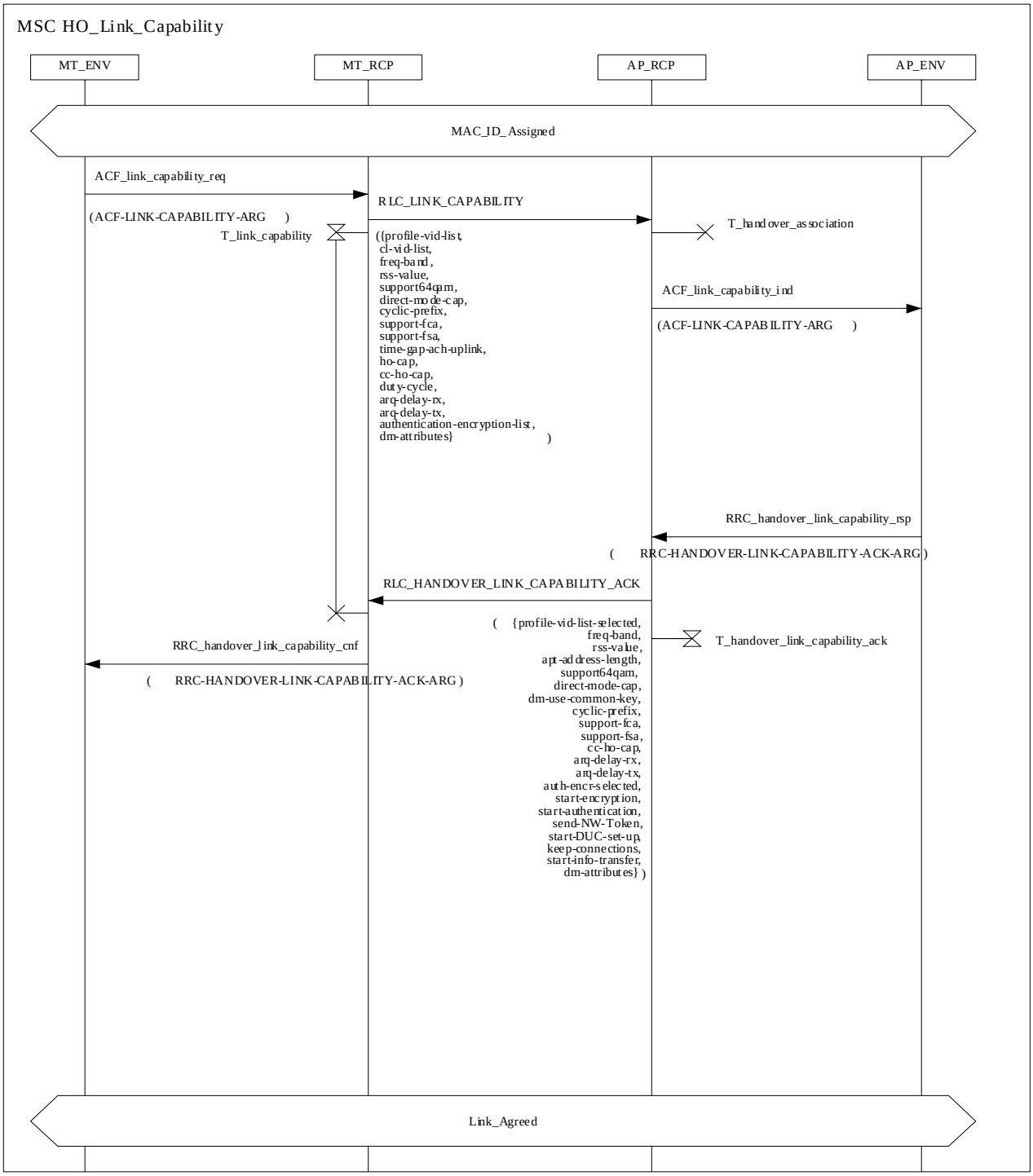


Diagram 38: Handover link capability procedure

Table 50: RLC-HANDOVER-LINK-CAPABILITY-ACK

RLC-HANDOVER-LINK-CAPABILITY-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
profile-vid-list-selected	PROFILE-VID-LIST
freq-band	FREQUENCY-BAND
rss-value	RSS-VALUE
apt-address-length	APT-ADDRESS-LENGTH
support64QAM	SUPPORTED64QAM
direct-mode-cap	DIRECT-MODE-CAP
dm-use-common-key	DM-USE-COMMON-KEY
cyclic-prefix	CYCLIC-PREFIX
support-fca	SUPPORTED-FCA
support-fsa	SUPPORTED-FSA
cc-ho-cap	CC-HO-CAP
arq-delay-rx	ARQ-DELAY
arq-delay-tx	ARQ-DELAY
auth-encr-selected	AUTH-ENCR-INFO
start-encryption	START-ENCRYPTION
start-authentication	START-AUTHENTICATION
send-NW-Token	SEND-NW-TOKEN
start-DUC-set-up	START-DUC-SET-UP
keep-connections	KEEP-CONNECTIONS
start-info-transfer	START-INFO-TRANSFER
dm-attributes	DM-ATTRIBUTES OPTIONAL -- (OAP/OMT) -- }

In the link capability procedure the MT shall provide its link capabilities to the AP using RLC_LINK_CAPABILITY message. The AP shall respond with RLC_HANDOVER_LINK_CAPABILITY_ACK. Based on the RLC_HANDOVER_LINK_CAPABILITY_ACK a subset of the following procedures shall be executed: Encryption Startup, Authentication, Token NW Signalling, Info Transfer, Setup Radio Connection mobile originated. The order of the procedures and allowed combinations are presented in diagram 36. How the *keep-connections* and *start-DUC-set-up* parameters shall be interpreted is shown in table 51.

Table 51: Definition of interpretation for *keep-connections* and *start-DUC-set-up* parameters

Parameter value for *keep-connections*	Parameter value for *start-DUC-set-up*	Comment
donot-keep-conn	start-setup	The MT shall initiate Setup Radio Connection mobile originated
keep-connections	donot-start-setup	The MT shall continue to use the DUCs established at the earlier AP.
donot-keep-conn	donot-start-setup	used if AP initiates DUC setup or DUC information is included in RLC_NETWORK_HANDOVER_COMPLETE
keep-connections	start-setup	Invalid combination

Regardless of which procedures are selected, the AP shall complete the entire network handover procedure, by transmitting the RLC_NETWORK_HANDOVER_COMPLETE message. In this message the AP may either confirm the characteristics of the DUCs established in the previous radio cell or modify them, only in the case DUCs have not already been negotiated during the Network Handover, i.e. by Setup Radio Connection mobile originated or when the *keep-connections* parameter indicates that connections shall be kept.

If not all DUCs, which the AP intends to maintain after handover, can be indicated in RLC_NETWORK_HANDOVER_COMPLETE, the AP shall set the duc-ext-ind flag and address the remaining DUCs in subsequent DUC Setup. This mechanism may also be used to establish DUCs by the AP.

DUCs established in a previous radio cell, but addressed neither in RLC_NETWORK_HANDOVER_COMPLETE nor in subsequent DUC Setup by the AP shall be considered by the MT as released.

After network handover AP and MT shall trigger the reset actions for the on-going DUCs in acknowledged mode (see [5]).

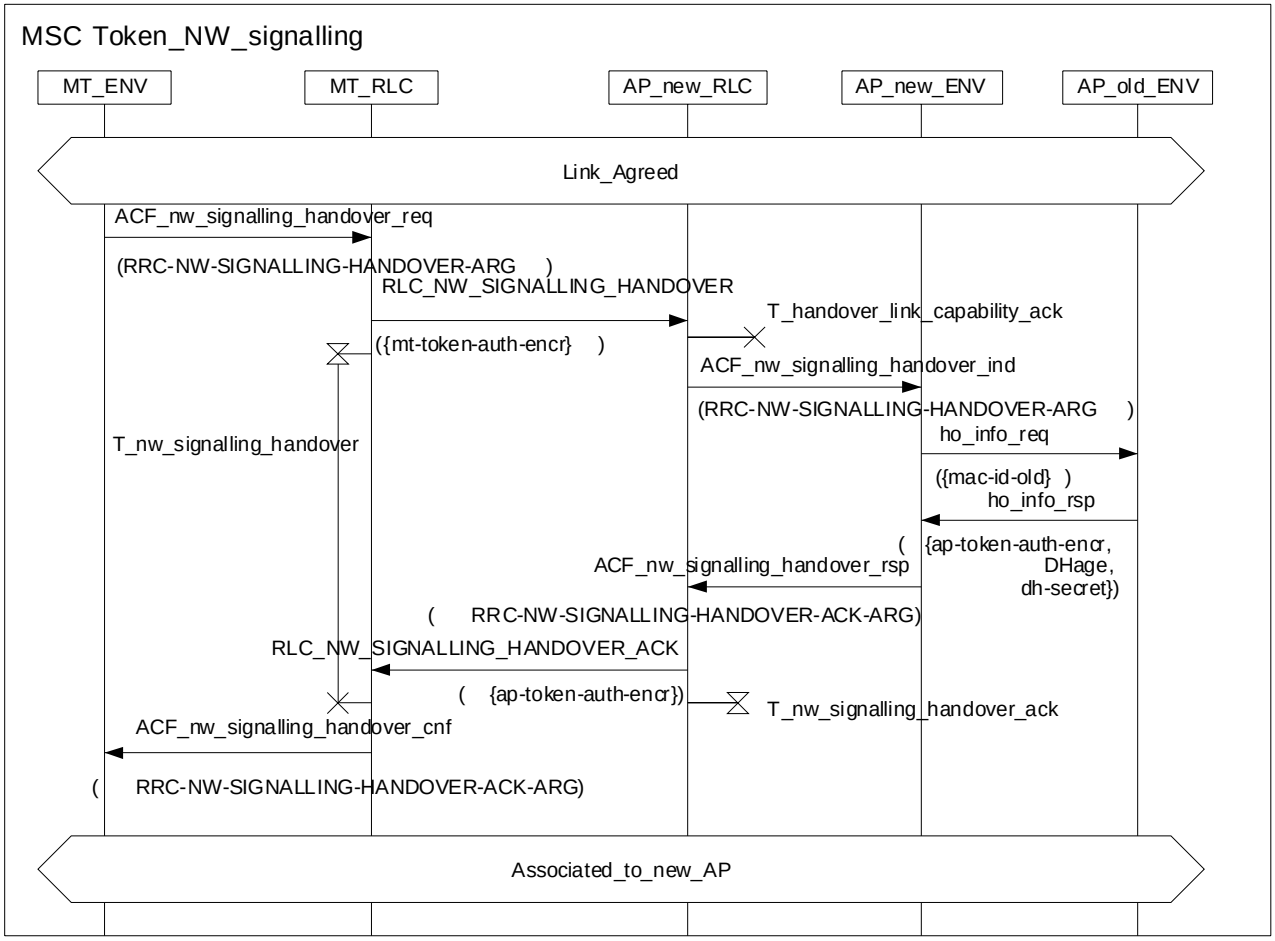


Diagram 39: Token network signalling procedure

Table 52: RLC-NW-SIGNALLING-HANDOVER

RLC-NW-SIGNALLING-HANDOVER-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
mt-token-auth-encr	MT-TOKEN-AUTH-ENCR}

Table 53: RLC-NW-SIGNALLING-HANDOVER-ACK

RLC-NW-SIGNALLING-HANDOVER-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE }
ap-token-auth-encr	**Error! Reference source not found.**

When a MT performs a handover it may send the RLC_NW_SIGNALLING_HANDOVER message, depending on send-NW-token in HANDOVER-LINK-CAPABILITY-ACK. The MT calculates MD5(token|authentication-encryption-list|auth-encr-selected) based on the earlier received token, the earlier sent *authentication-encryption-list* and the received *auth-encr-selected* and sends the result to the AP.

NOTE: With support of the fixed network, the new AP can contact the old AP directly and verify the continuity.

The token, DH age and DH secret shall be transferred from the old AP to the new AP during network handover. The procedures for the information transfer via the fixed network are out of the scope of the present document.

The new AP shall do the following for security purposes:

1) Contact the old AP, presenting the old MAC ID of the MT and fetching the following information:

   - Token, the same one was sent to the MT by the old AP in the RLC_HO_INFO_DISTRIBUTION message;

   - DHsecret, which equals g^{xy} mod n, the result of the last Diffie-Hellman exchange carried out in the encryption startup procedure;

   - DHAge, how long time the DHsecret has been used.

2) Calculate MD5(token|authentication-encryption-list|auth-encr-selected) and compare the result with the one received in RLC_NW_SIGNALLING_HANDOVER. If the two hash values are not equal, reject the MT's handover request.

3) Calculate the new unicast key SSK as described in clause 5.1.2.5.4.

4) Send back the RLC_NW_SIGNALLING_HANDOVER_ACK message, encrypted with the new SSK (if encryption is used). The AP calculates MD5(token|authentication-encryption-list|auth-encr-selected) based on the token received from the old AP, earlier received *authentication-encryption-list* and sent *auth-encr-selected* and sends the result to the MT.

The MT shall calculate the new SSK (see clause 5.1.2.5.4) and verify that the received information in the RLC_NW_SIGNALLING_HANDOVER_ACK message is the same as used for the calculation of the RLC_NW_SIGNALLING_HANDOVER message.

5.2.1.4 Token distribution for Network Handover

The transfer of the token shall be done while the MT is associated to the current AP. The current AP sends the message to the MT. It contains a token encrypted with the unicast key (RLC_HO_INFO_DISTRIBUTION). The MT decrypts the token and stores it for later use. It also sends an acknowledgement back to the old AP.

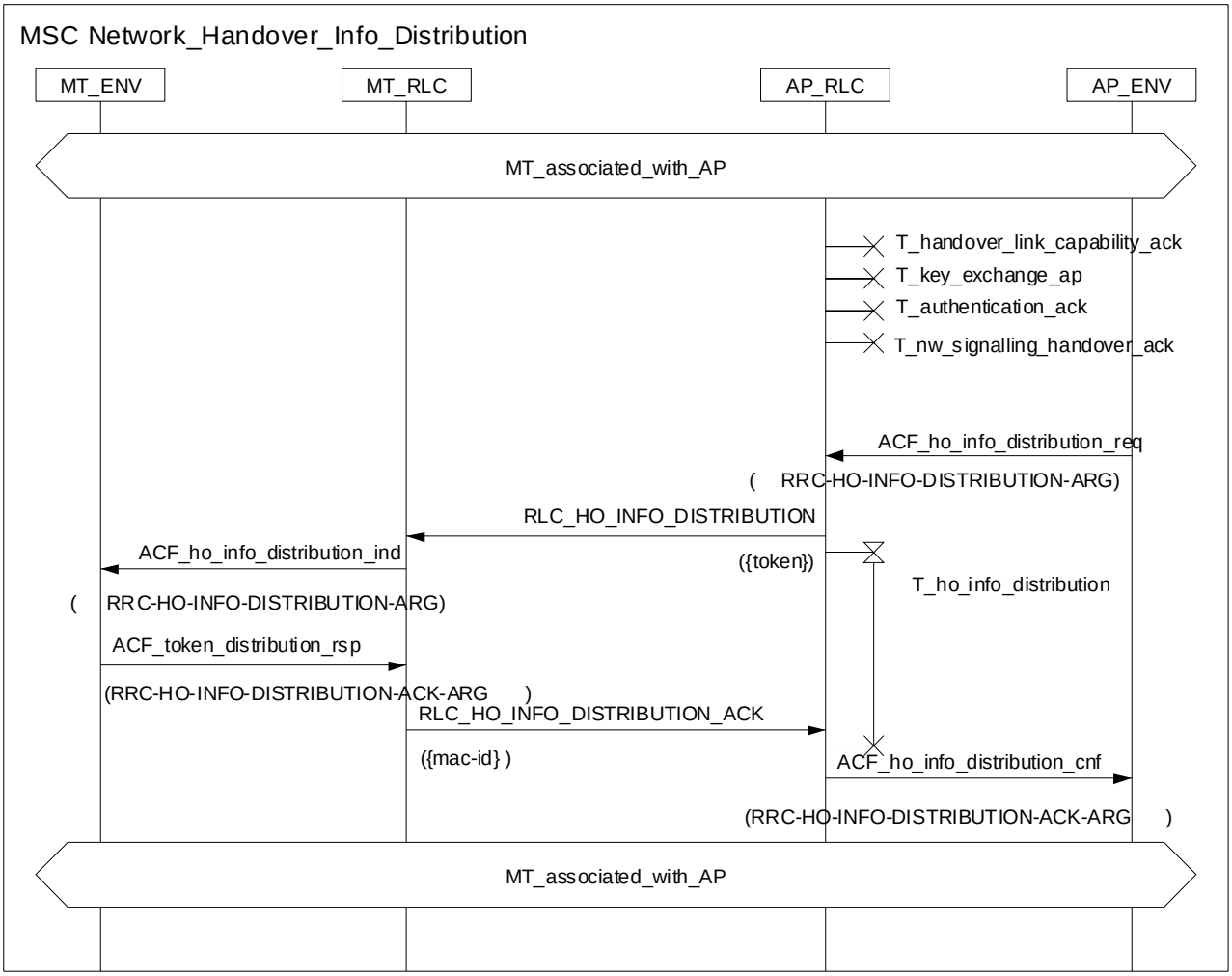


Diagram 40: Network handover info distribution

Table 54: RLC-HO-INFO-DISTRIBUTION

```
RLC-HO-INFO-DISTRIBUTION-ARG ::= SEQUENCE {
    rlc-pdu-type        RLC-LCH-PDU-TYPE
    token               TOKEN }
```

Table 55: RLC-HO-INFO-DISTRIBUTION-ACK

```
RLC-HO-INFO-DISTRIBUTION-ACK-ARG ::= SEQUENCE {
    rlc-pdu-type        RLC-SCH-PDU-TYPE
    mac-id              MAC-ID }
```

The *Token* should be a random value. The *Token* shall be used for the two following purposes:

1) to show that the MT is really the one that made a network handover from the old AP to the new AP;

2) to allow the MT and the new AP to calculate a new SSK based on the *Token*.

NOTE: The function needs a random number (see bibliography, "Applied cryptography Second Edition" and "Randomness Recommendations for Security" for further information about this subject).

NW Handover Info Distribution shall be executed after association and network handover to keep the token up-to-date.

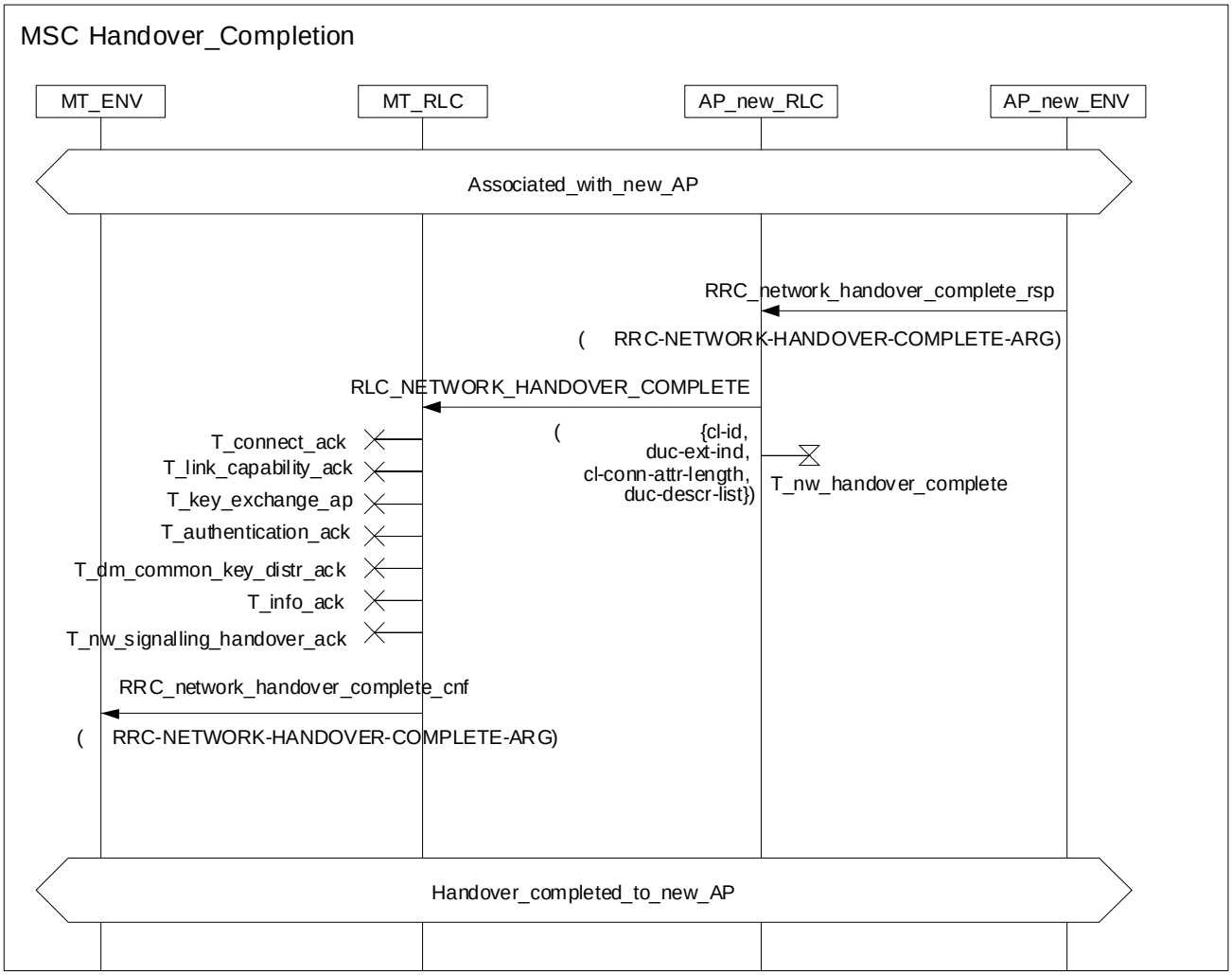


Diagram 41: Handover Completion procedure

Table 56: RLC-NETWORK-HANDOVER-COMPLETE

RLC-NETWORK-HANDOVER-COMPLETE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-id	CL-ID
duc-ext-ind	DUC-EXT-IND
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

5.2.1.5 Handover Rejection

The rejection of the Radio and Network Handover procedures shall follow the next two rules:

1. If the AP does not accept the RLC_HANDOVER_REQUEST message, the AP shall respond with the RLC_HANDOVER_REQUEST_NACK message.

2. After the MT has a MAC ID, a possible rejection shall be done by sending RLC_DISASSOCIATION message. Both the MT and AP shall use this message for this purpose.

NOTE: Because of varying radio link conditions, the Disassociation may happen implicitly, see clause 5.1.3.

Table 56a: RLC-HANDOVER-REQUEST-NACK

RLC-HANDOVER-REQUEST-NACK-ARG ::= SEQUENCE {		
rlc-pdu-type	RLC-SCH-PDU-TYPE	
ap-id-old	AP-ID	-- AP-ID of old APT
mac-id-old	MAC-ID	-- MAC-ID used in old APT
net-id-old	NET-ID}	

5.2.1.6 Forced Handover (AP initiated handover) (OAP/OMT)

APs may initiate HO capable MTs to perform forward handover by using RLC_FORCE_HANDOVER. The AP shall make sure that the MT is HO capable. The AP shall stop scheduling data to the MT after sending RLC_FORCE_HANDOVER message.

- If the *return-flag* value is set to zero, the MT shall respond with RLC_FORCE_HANDOVER_ACK and shall not return to the current AP to continue operation with the old association. When the RLC_FORCE_HANDOVER_ACK is received by the AP, it shall disassociate the MT without sending the RLC_DISASSOCIATION message.

- If *return-flag* is set to one, MTs shall respond with RLC_FORCE_HANDOVER_ACK. The MT shall not return to operate with old association after 255 frames after the frame the RLC_FORCE_HANDOVER_ACK was received by the AP (the receiving frame is number 0). When returning to the old AP, the MT shall continue operation with the AP by sending RLC_MT_ALIVE, Resource Request or RLC_HANDOVER_NOTIFY.

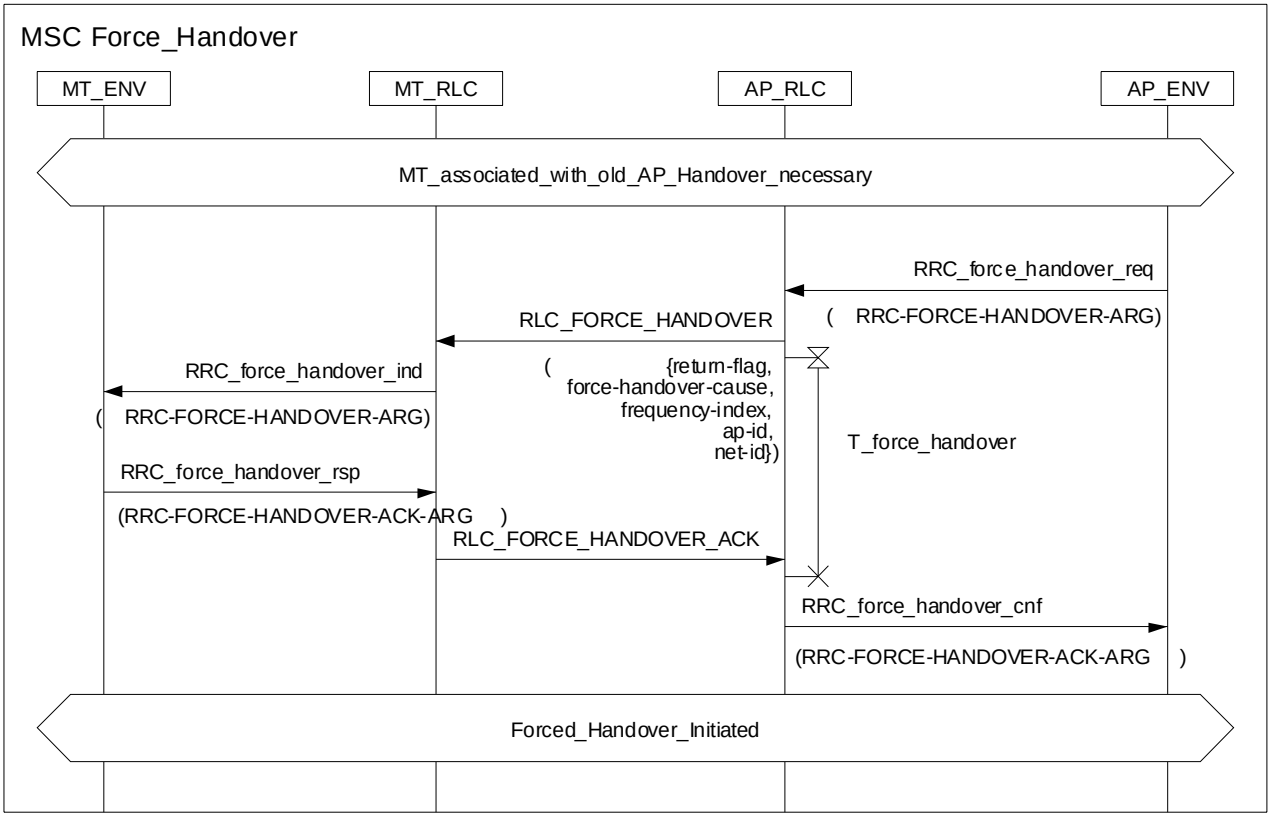


Diagram 42: Forced handover procedure

Table 57: RLC-FORCE-HANDOVER

RLC-FORCE-HANDOVER-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
return-flag	RETURN-FLAG
force-handover-cause	FORCE-HANDOVER-CAUSE
frequency-index	FREQUENCY-INDEX
ap-id	AP-ID
net-id	NET-ID }

Table 58: RLC-FORCE-HANDOVER-ACK

RLC-FORCE-HANDOVER-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
mac-id	MAC-ID }

5.2.2 Dynamic Frequency Selection

5.2.2.1 Introduction to DFS

The Dynamic Frequency Selection (DFS) in HL/2 systems shall result in equal usage of available frequencies under the consideration of avoiding the interference of other devices using the same spectrum [12]. The interference may arise from neighbouring HL/2 networks using the same frequency or non-HL/2 devices in the frequency band. Every AP should collect measurement results and choose an operating frequency based on the measurement results. The decision may be done independently of other APs.

5.2.2.2 DFS algorithm

The DFS algorithm is out of the scope of the present document.

5.2.2.3 DFS protocol

The AP may request any associated MTs to measure and report the measurements. The MT shall perform the measurements and after the measurement it shall report the results to the AP.

A MT may also do self-initiated measurements and request to report the results to the AP. The AP may then poll the MT for the result or may ignore the request. The implementation of the RLC_MT_INITIATED_REPORT_REQUEST is optional for the MT, but mandatory for the AP.

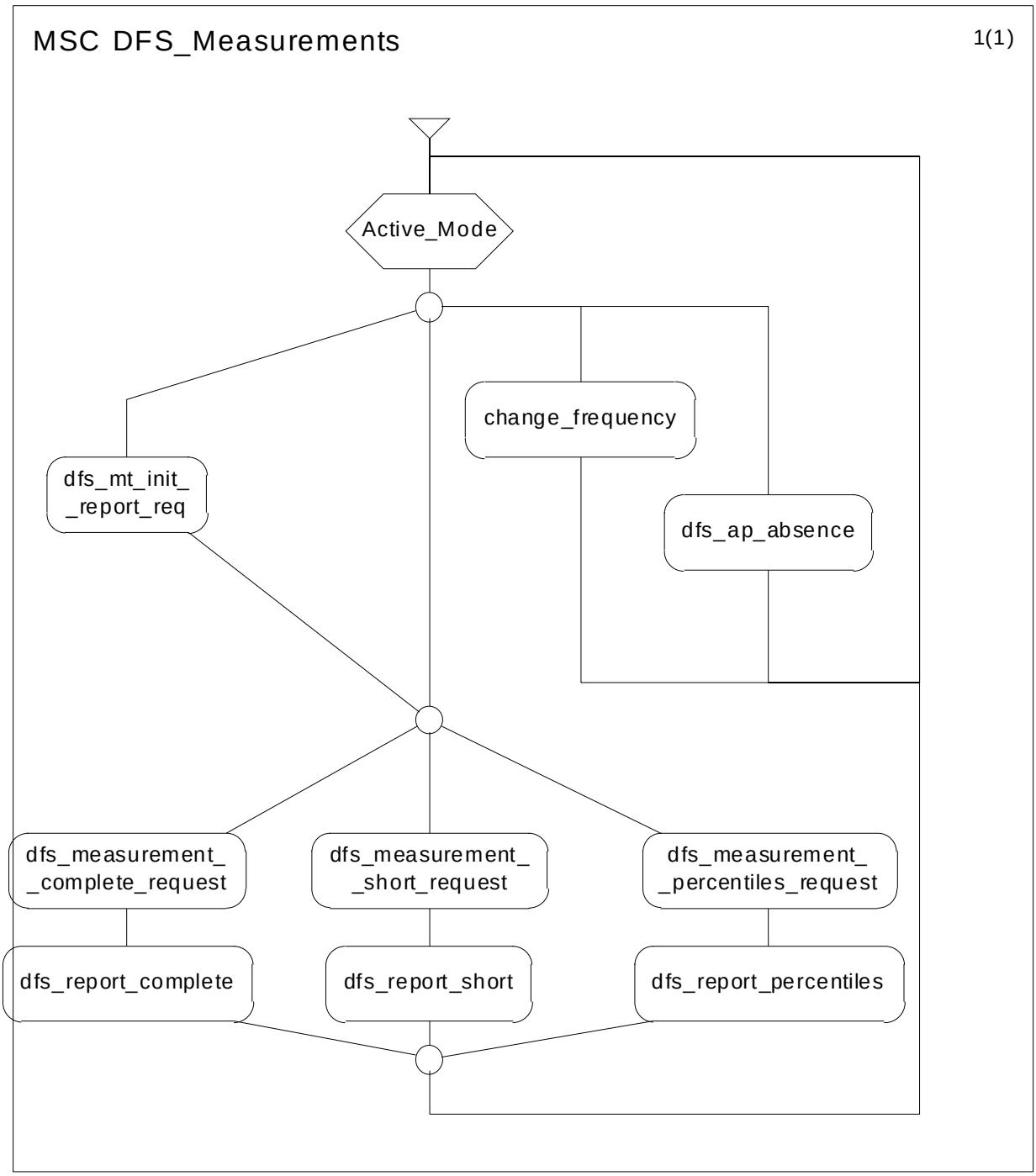


Diagram 43: DFS measurements

5.2.2.4 DFS measurements

APs and MTs shall be able to perform measurements to support DFS, namely RSS measurements at a given frequency. APs and MTs shall be able to decode possible BCH transmissions of other APs at a given frequency.

The measurements in the AP are out of scope of the specification.

When the AP is switched on, it shall measure on all frequencies it is permitted to use.

5.2.2.4.1 AP Measurement Procedure

When the AP makes measurements and is not able to transmit, it shall use AP Absence.

NOTE 1: AP should disturb sleeping MTs as little as possible. This can be handled by making the measurement between MAC broadcast sleep group frames.

The maximum AP Absence period shall be 15 frames. The AP shall keep its frame cycle during AP Absence.

NOTE 2: The three shortest sleeping periods for MTs are 2, 4 and 8 frames. The MT that has one of these short sleeping periods, may lose 1 to 8 expected BCHs during AP Absence period. The MT can, however, hear the AP_ABSENCE message during the MAC broadcast sleep group frame.

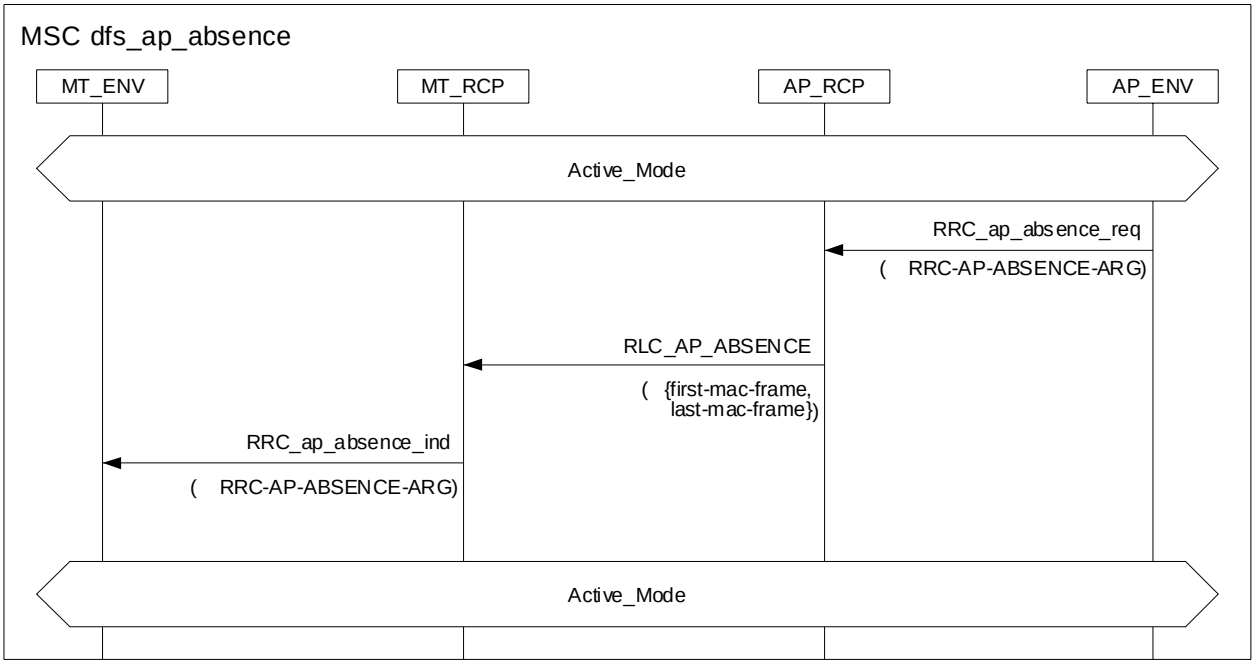


Diagram 44: DFS AP absence procedure

Table 59: RLC-AP-ABSENCE

RLC-AP-ABSENCE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
first-mac-frame	FIRST-MAC-FRAME
last-mac-frame	LAST-MAC-FRAME }

5.2.2.4.2 MT Measurement Procedures

Three measurement types are defined for a MT: short, percentiles and complete. All of the measurement requests shall be sent as unicast messages only.

Short: At the given time and frequency, MT shall scan for BCH. If a BCH is found it shall be decoded and its RSS shall be measured.

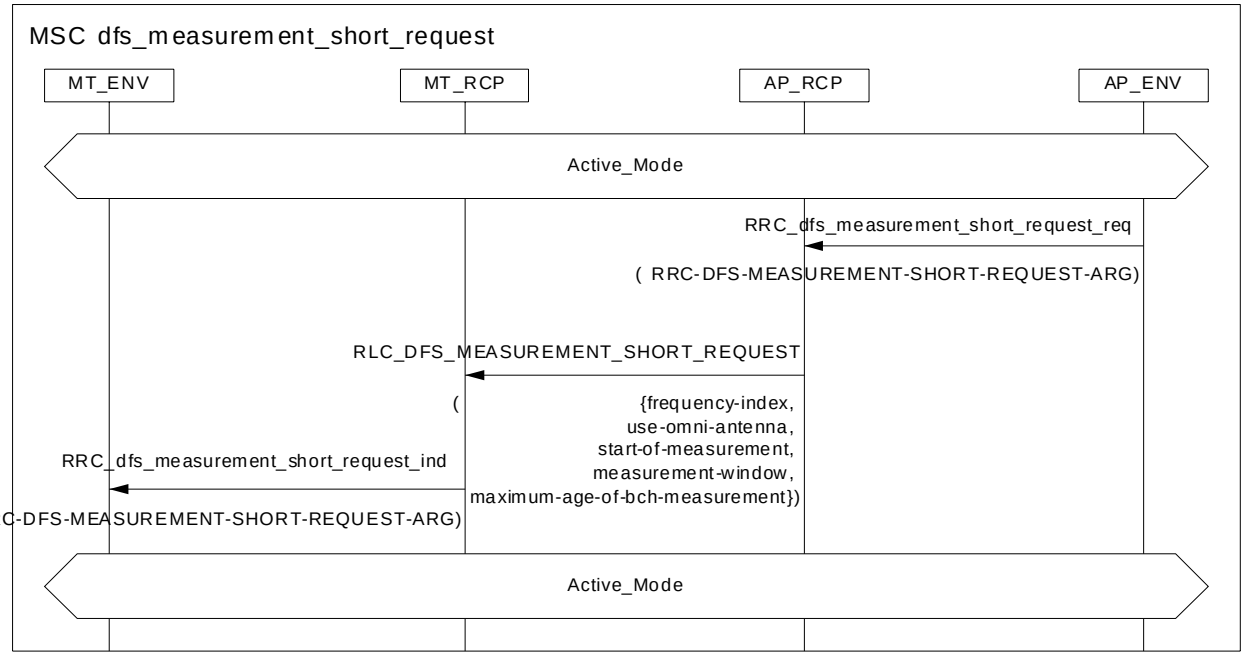


Diagram 45: DFS short measurement request

Table 60: RLC-DFS-MEASUREMENT-SHORT-REQUEST

RLC-DFS-MEASUREMENT-SHORT-REQUEST-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
frequency-index	FREQUENCY-INDEX
use-omni-antenna	USE-OMNI-ANTENNA
start-of-measurement	START-OF-MEASUREMENT
length-of-measurement	MEASUREMENT-WINDOW
maximum-age-of-bch-measurement	MAXIMUM-AGE-OF-BCH-MEASUREMENT }

Percentiles: at the given time and frequency, the MT shall collect RSS samples with equal distance 8 μs.

NOTE: No decoding of BCH is done for percentiles measurement.

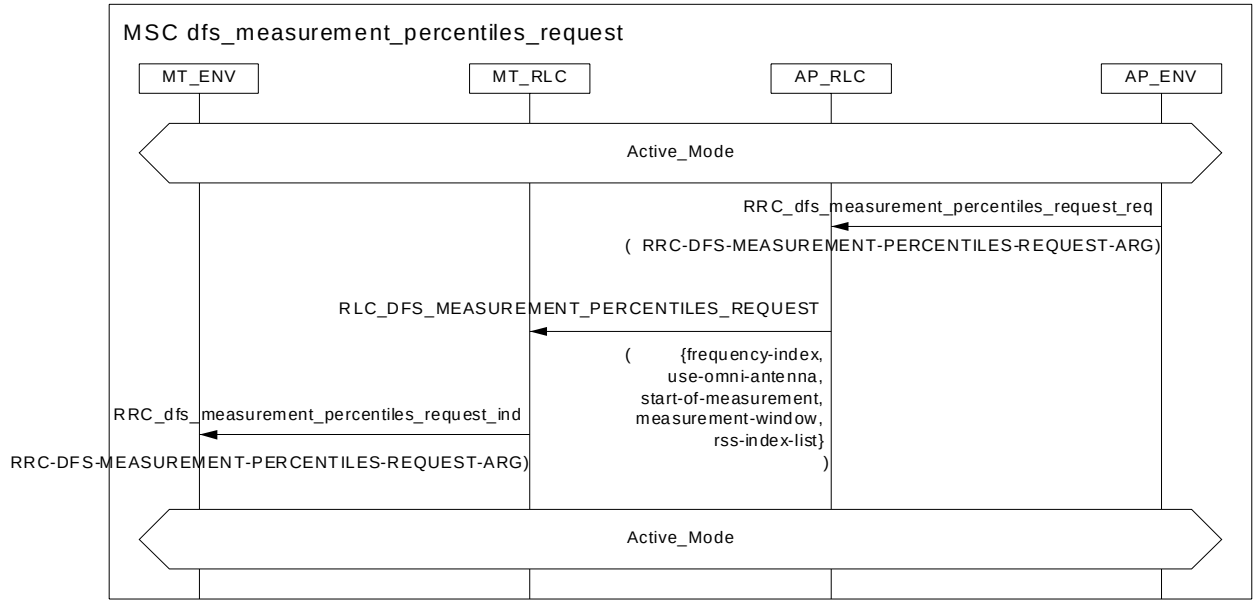


Diagram 46: DFS percentiles measurement request

Table 61: RLC-DFS-MEASUREMENT-PERCENTILES-REQUEST

RLC-DFS-MEASUREMENT-PERCENTILES-REQUEST-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
frequency-index	FREQUENCY-INDEX
use-omni-antenna	USE-OMNI-ANTENNA
start-of-measurement	START-OF-MEASUREMENT
measurement-window	MEASUREMENT-WINDOW
rss-index-list	RSS-INDEX-LIST }

Complete: a combination of the short and percentiles measurements, where at the given time and frequency BCH shall be decoded and RSS measurement on the BCH shall be performed and the RSS samples shall be collected.

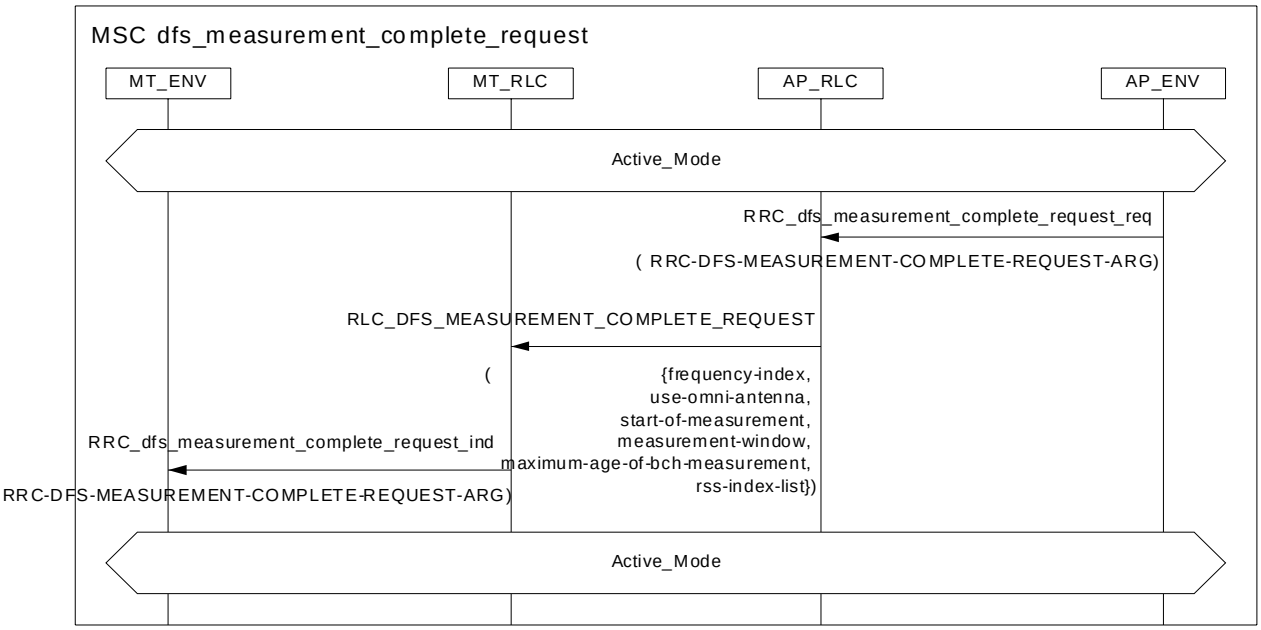


Diagram 47: DFS complete measurement request

Table 62: RLC-DFS-MEASUREMENT-COMPLETE-REQUEST

RLC-DFS-MEASUREMENT-COMPLETE-REQUEST-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
frequency-index	FREQUENCY-INDEX
use-omni-antenna	USE-OMNI-ANTENNA
start-of-measurement	START-OF-MEASUREMENT
measurement-window	MEASUREMENT-WINDOW
maximum-age-of-bch-measurement	MAXIMUM-AGE-OF-BCH-MEASUREMENT
rss-index-list	RSS-INDEX-LIST }

Each RSS sample measurement shall follow the requirements in [4].

The measurement requests described above are sent from AP to MT, but MT may also ask AP to send the request. In this case, the AP shall answer MT with an ACK. The ACK shall contain a flag stating whether the AP is going to send a request or not.

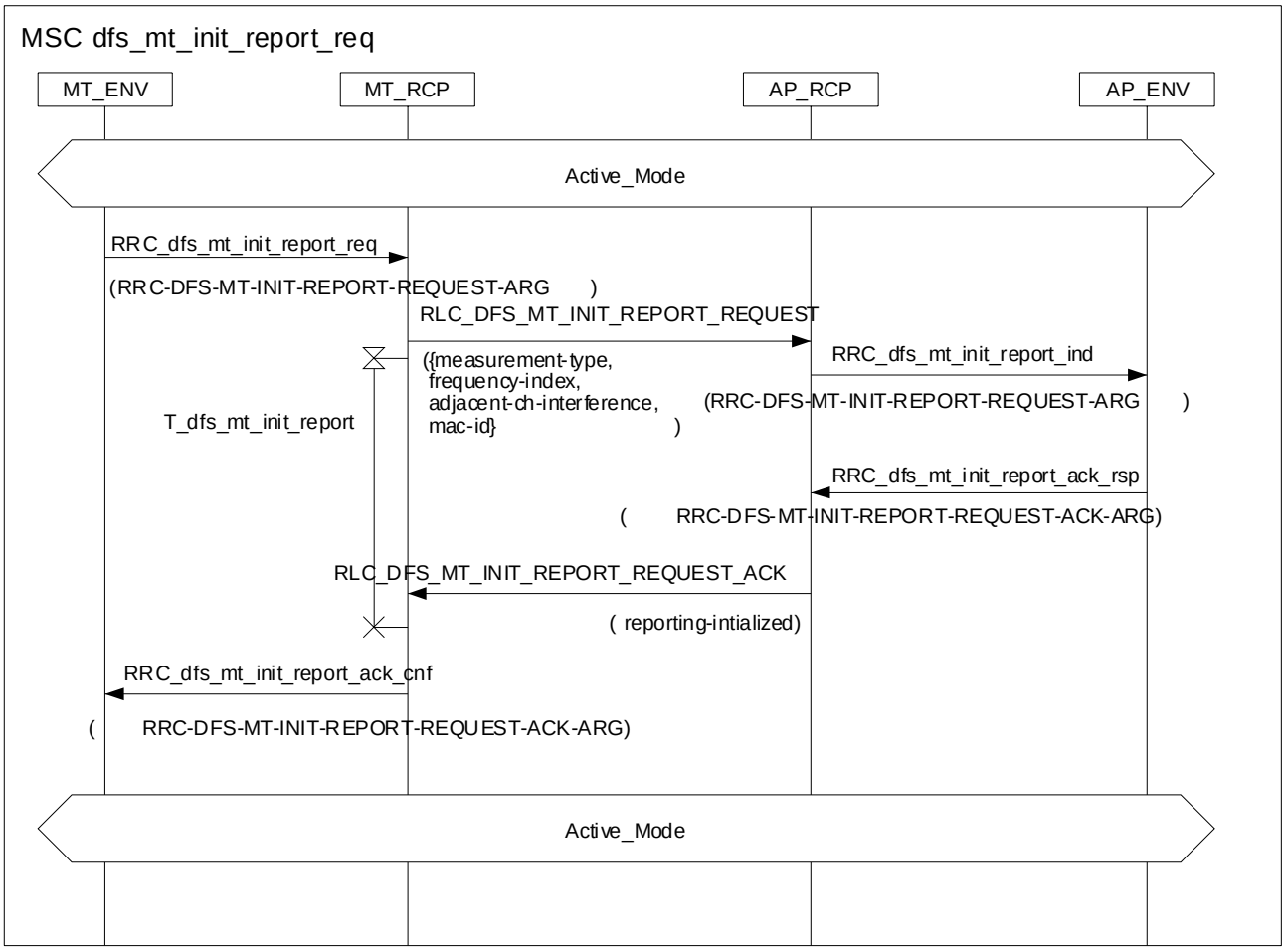


Diagram 48: DFS MT init report request procedure
(OMT Optional for the AP to send a request after receiving the message)

Table 63: RLC-DFS-MT-INIT-REPORT-REQUEST

RLC-DFS-MT-INIT-REPORT-REQUEST-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
measurement-type	MEASUREMENT-TYPE
frequency-index	FREQUENCY-INDEX
adjacent-ch-interference	ADJACENT-CH-INTERFERENCE
mac-id	MAC-ID }

Table 64: RLC-DFS-MT-INIT-REPORT-REQUEST-ACK

RLC-DFS-MT-INIT-REPORT-REQUEST-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
reporting-initialized	REPORTING-INITIALIZED }

5.2.2.4.2.1 Measurements on other frequencies

When the AP needs measurements on other frequencies than the one it is currently using, the AP may request one or more MTs to do the measurements. In order to do that, the AP shall send one of the three measurement requests.

The measurement procedure on a currently not used frequency is described below.

1) The MT shall measure RSS0 [4] for the *last_own_bch_rss_level* from the frame previous to the frame number *start-of-measurement*.

2) The MT shall start to tune to the specified frequency *f* from the beginning of the MAC frame that is announced with *start-of-measurement*.

NOTE: Tuning time is not included in the *measurement-window*.

3) Depending on the measurement type, the measurement procedure shall be the following:

   - **Short:** The MT shall search for BCH of other interfering APs for at most for 5 frames (i.e. measurement window larger than 5 is not applicable). If found the BCH shall be decoded and RSS0 [4] shall be measured on the BCH. If several BCHs are found the strongest RSS0 value should be stored for reporting. The MT also stores part of the BCH content for reporting. See description of corresponding measurement reports.

   - **Percentiles:** Consecutive RSS1 samples [4] with distance 8 µs shall be collected during the specified *measurement-window*.

   - **Complete:** The MT shall tune to the new frequency and search for BCH of other interfering APs for at most 5 frames. If found the BCH shall be decoded and RSS0 [4] shall be measured on the BCH. If several BCHs are found the strongest RSS0 value should be stored for reporting. Also parts of the decoded BCH content is stored for reporting. See description of corresponding measurement report. Also for the remaining time of the measurement window consecutive RSS1 samples [4] with distance 8 µs shall be collected. The order of these two measurement phases is out of scope of the present document.

4) The MT shall then tune to the original frequency f_0. The retuning time is not included in the *measurement-window*.

5) MT shall report the requested measurement results to the AP. The report shall be available at the latest 5 frames after the return to the current frequency.

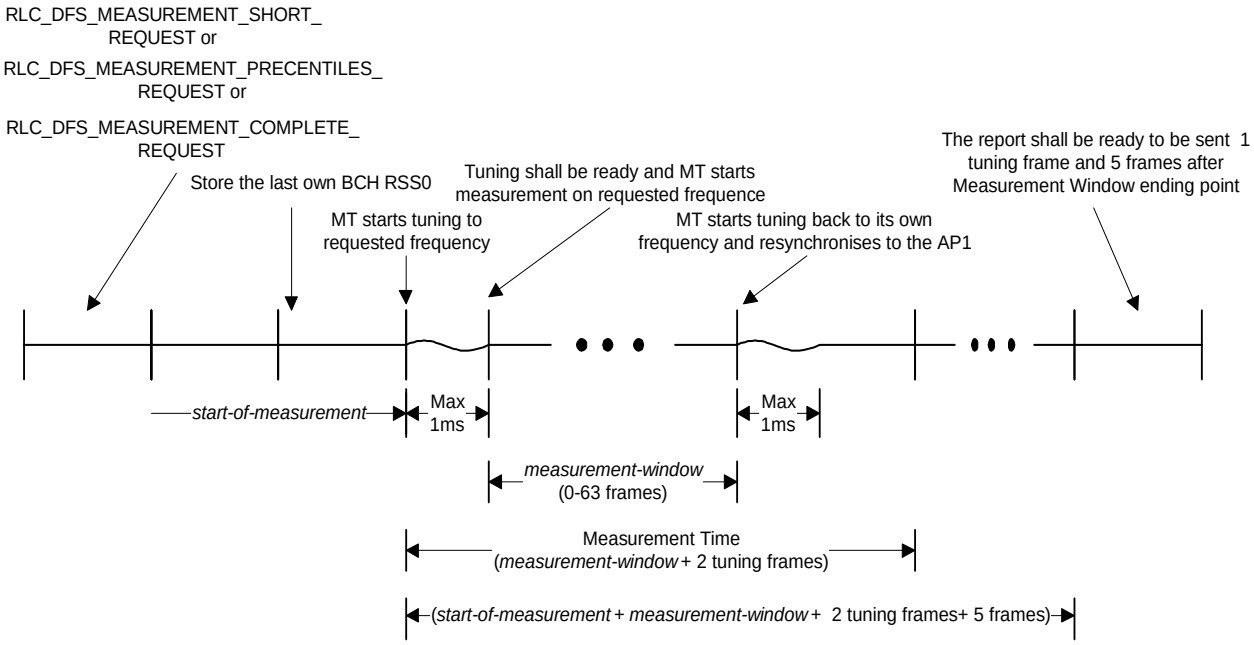


Figure 4: The measurement on other frequencies

The *start-of-measurement* value is counted from the frame, where the message was received (i.e. next frame is frame 0).

Start-of-measurement shall not be smaller than 2 MAC frames i.e. the AP can not expect the MT to start tuning to the new frequency until two frames after the measurement request was transmitted. The example above depicts the scenario when *start-of-measurement* = 2.

The AP should not schedule any data to the measuring MT during the measurement time. The measurement time shall be from the *start-of-measurement* frame (belonging to measurement):

measurement-window + 2 frames

NOTE: The 2 frames extra are for tuning and retuning time.

5.2.2.4.2.2 Measurements on used frequency

When the AP needs measurements on the used frequency to be done by MT, the AP shall send one of the three measurement requests. The procedures shall be according to A or B.

Procedure A:
If the AP sends the RLC_DFS_MEASUREMENT_COMPLETE_REQUEST or RLC_DFS_MEASUREMENT_SHORT_REQUEST, the AP shall set the empty space for measurements by transmitting the RLC_AP_ABSENCE message and entering into the AP_Absence.

The measurement-window shall be set in the request so that it defines a coarse interval during which the MT can expect to perform its measurement request. The real measurement is then triggered by the RLC_AP_ABSENCE message and the MT uses the beginning of the AP_ABSENCE interval as its start_of_measuring. The AP Absence interval should defined be within the measurement window.

The measurement procedure is described below:

1) The MT shall measure RSS0 [4] for the *last_own_bch_rss_level* from the frame previous to the frame number *start-of-measurement*.

2) Depending on the measurement type, the measurement shall be the following:

   - **Short:** The MT shall search for BCH of other interfering APs during the AP Absence. If found the BCH shall be decoded and RSS0 [4] shall be measured on the BCH. If several BCHs are found the strongest RSS0 value should be stored for reporting. The MT also stores parts of the BCH content for reporting.

   - **Complete:** The MT shall search for BCH of other interfering APs during the AP Absence. If found the BCH shall be decoded and RSS0 [4] shall be measured on the BCH. If several BCHs are found the strongest RSS0 value should be stored for reporting. The MT also stores parts of the BCH content for reporting. Also for the remaining time of the Absence period consecutive RSS1 samples [4] with distance 8 µs shall be collected. The order of these two measurement phases is out of scope of the present document.

3) MT shall report the requested measurement results to the AP. The report shall be available at the latest 5 frames after end of measurement window.

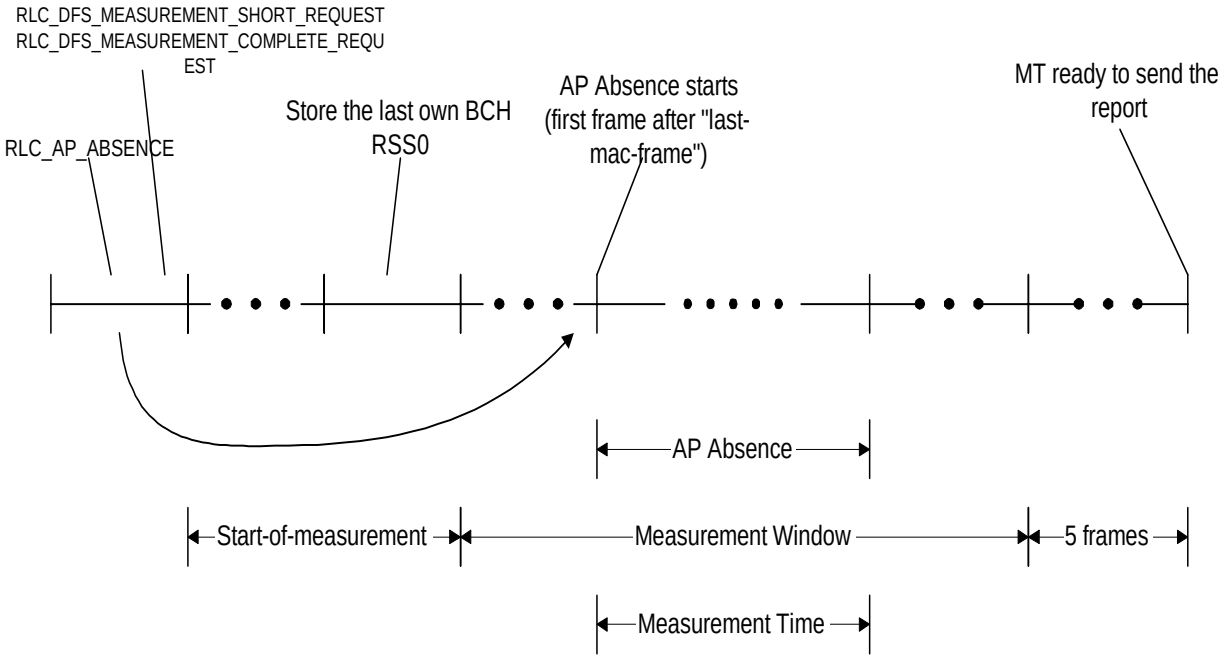


Figure 5: Short or Complete measurement on used frequency

The *start-of-measurement* value is counted from the frame, where the message was received (i.e. next frame is frame 0).

Start-of-measurement shall not be smaller than 2 MAC frames i.e. the AP can not expect the MT to start measuring until two frames after the measurement request was transmitted. The example above depicts the scenario when *start-of-measurement* = 2.

The AP shall make sure that the MT is notified of the start of the absence period at least 2 frames prior to the start of the actual measurement interval (i.e. also this notification shall also be performed at least two frames beforehand).

If the MT for some reason doesn't succeed in finding an AP Absence interval within the *measurement-window* it shall neglect the corresponding measurement request.

The RLC_AP_ABSENCE message pointing to the measurement shall be sent before or in the same frame with the measurement request.

NOTE: The AP should not schedule any data to the measuring MT during the measurement time. The measurement time shall be from the *start-of-measurement* frame (belonging to measurement): *measurement-window*

Procedure B:
If the AP sends the RLC_DFS_MEASUREMENT_PERCENTILES_REQUEST there are two possibilities (B1 or B2). Both shall be supported by the MT.

B1: If the AP requires the MT to measure only the empty spaces of the frame, the AP shall indicate them by using FCH IE for indicating empty parts in the MAC frame [5].

The FCH IE for the empty parts shall not be scheduled earlier in the MAC frame than what is given by the minimum time between ACH and first uplink phase [5].
The AP shall use the same technique as described in A. That is it only gives a coarse measurement window description in the measurement request and set the *start-of-measurement* to point out the frame, where the scheduling of the empty slots is expected to be started.

The measurement procedure is described below.

1) The MT shall measure RSS0 [4] for the *last_own_bch_rss_level* from the frame previous to the frame number *start-of-measurement.*

2) The MT shall collect the RSS1 samples [4] with distance 8 μs in all available empty spaces.

3) MT shall report the requested measurement results to the AP. The report shall be available at the latest 5 frames after end of measurement window.

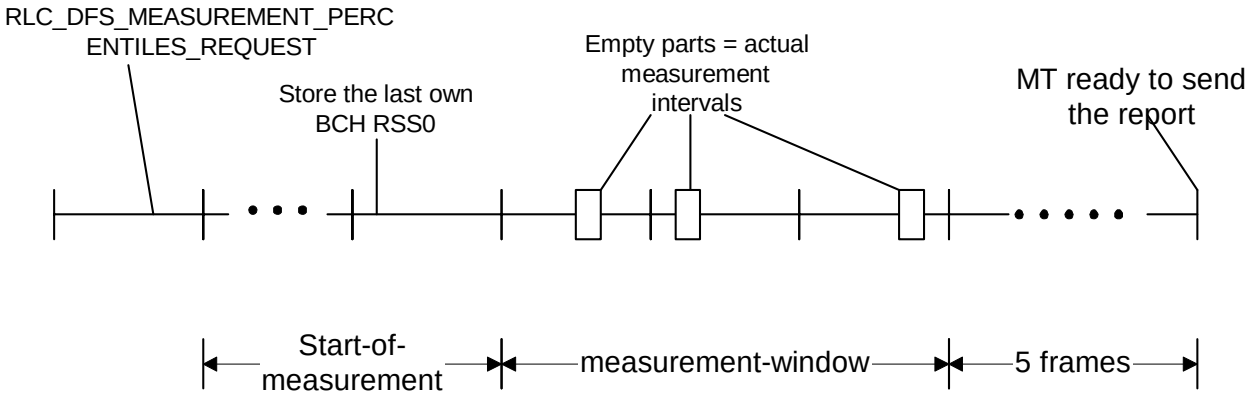


Figure 6: Percentiles measurement on used frequency

The *start-of-measurement* value is counted from the frame, where the message was received (the next frame is frame number 0).

Start-of-measurement shall not be smaller than 2 MAC frames i.e. the AP can not expect the MT to start measuring until two frames after the measurement request was transmitted.

The AP should not schedule any data to the measuring MT during the measurement time. The measurement time shall be from the *start-of-measurement* frame (belonging to measurement): *measurement-window.*

B2: The AP may also use AP Absence procedure as in case A. The AP is supposed not to use B1 and B2 at the same time for a certain percentile measurement.

5.2.2.4.3 MT measurement processing

5.2.2.4.3.1 Measurement processing (field strength measurement)

The RSS statistics specified in the measurement request shall be calculated and reported to the AP.

5.2.2.4.3.2 Measurement processing (field strength measurement and BCH decoding)

In addition to the RSS measurements described above, the MT may also be requested to decode the BCH of the measured frequency. In this case, the RSS0 [4] of the BCH and parts of the BCH content shall also be reported to the AP.

5.2.2.4.3.3 Indication of measurement time

When the AP requests an MT to measure interference on the currently used frequency according to alternative B the AP shall indicate in the FCH where in the MAC frames the MT shall measure (i.e. where empty space is located). This shall be done by using a special information element (IE) type for this purpose, defined in [5].

5.2.2.4.4 Calculation of RSS statistics (informative)

After the MT has collected RSS samples, statistical values are calculated by the MT and reported to the AP. The statistical values are percentiles of the interference. The percentile calculation is described below.

The only possible RSS values are [0, -1, -2, ..., -31] dB, as defined in [4].

NOTE: The RSS1 are relative measurements that give the message outside of the BCH relative to a value RSS_REF, as described in the PHY reference mentioned.

Thus, the percentiles can be calculated in the following way:

1) each RSS sample is placed in a bin, where the bins have values [0, -1, -2, ..., -31];
2) after all samples have been collected (250 samples/MAC frame) the value *M* is found, where *M* equals the maximum value *m* (dB) such that ≤ *x* % of the samples have an RSS value ≤ *m*;
3) the value *M* equals the *x* % percentile.

EXAMPLE: Assume that there are the following RSS samples collected:

RSS value (bin)	No of samples
0	0
...	...
-28	4
-29	5
-30	7
-31	0

When 5 % percentile out of total 250 samples is needed to be calculated, the result will be M = -29, since 12 samples (4,8 %) lie in the bins -31 to -29 and 16 samples (64 %) lie in the bins -31 to -28.

5.2.2.5 DFS measurement reports

MT shall report the RSS distribution at a given frequency relative to the latest performed RSS0 measurement on the currently used frequency, which is also included in the message. If the MT decodes the BCH channel of another AP, it shall also include the following fields of the BCH, if it is requested by the AP; AP ID, Net ID, AP Tx power level, Traffic load. In addition the MT shall report RSS0 of the decoded BCH.

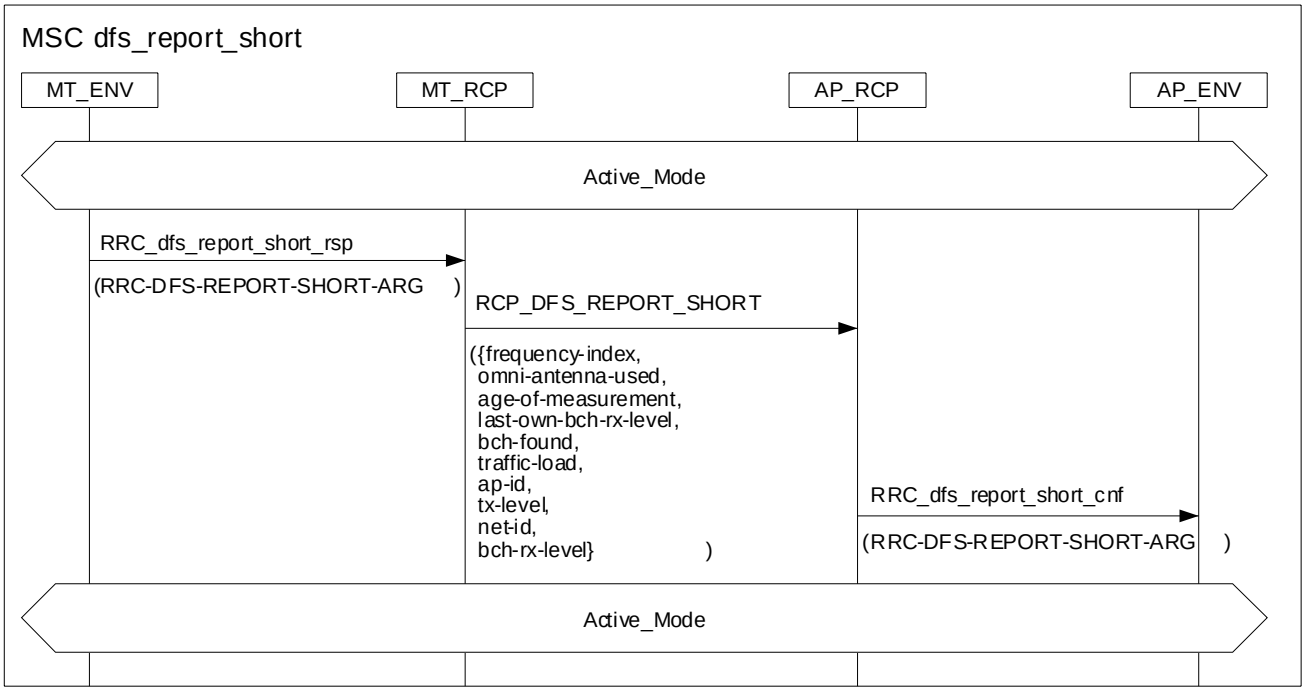


Diagram 49: DFS short report

Table 65: RLC-DFS-REPORT-SHORT

RLC-DFS-REPORT-SHORT-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
frequency-index	FREQUENCY-INDEX
omni-antenna-used	OMNI-ANTENNA-USED
age-of-measurement	AGE-OF-MEASUREMENT
last-own-bch-rx-level	LAST-OWN-BCH-RX-LEVEL
bch-found	BCH-FOUND
traffic-load	TRAFFIC-LOAD
ap-id	AP-ID
tx-level	TX-LEVEL
net-id	NET-ID
bch-rx-level	BCH-RX-LEVEL }

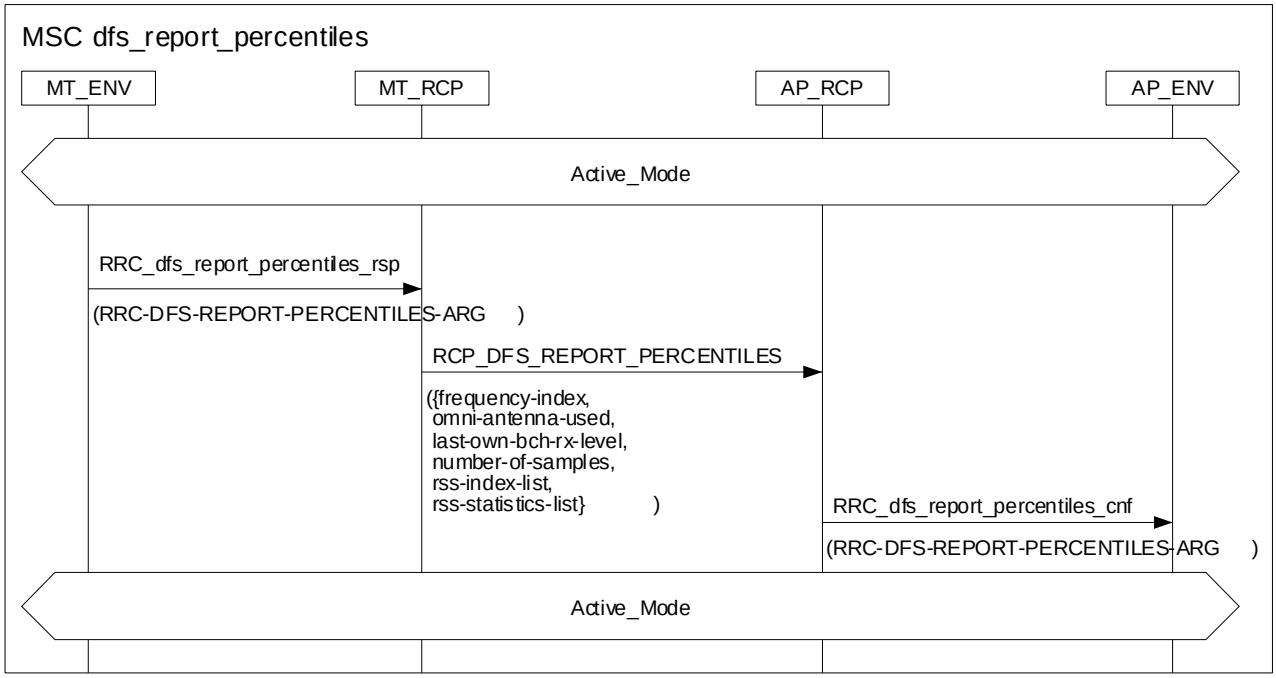


Diagram 50: DFS percentiles report

Table 66: RLC-DFS-REPORT-PERCENTILES

RLC-DFS-REPORT-PERCENTILES-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE,
frequency-index	FREQUENCY-INDEX
omni-antenna-used	OMNI-ANTENNA-USED
last-own-bch-rx-level	LAST-OWN-BCH-RX-LEVEL
number-of-samples	NUMBER-OF-SAMPLES
rss-index-list	RSS-INDEX-LIST
rss-statistics-list	RSS-STATISTICS-LIST }

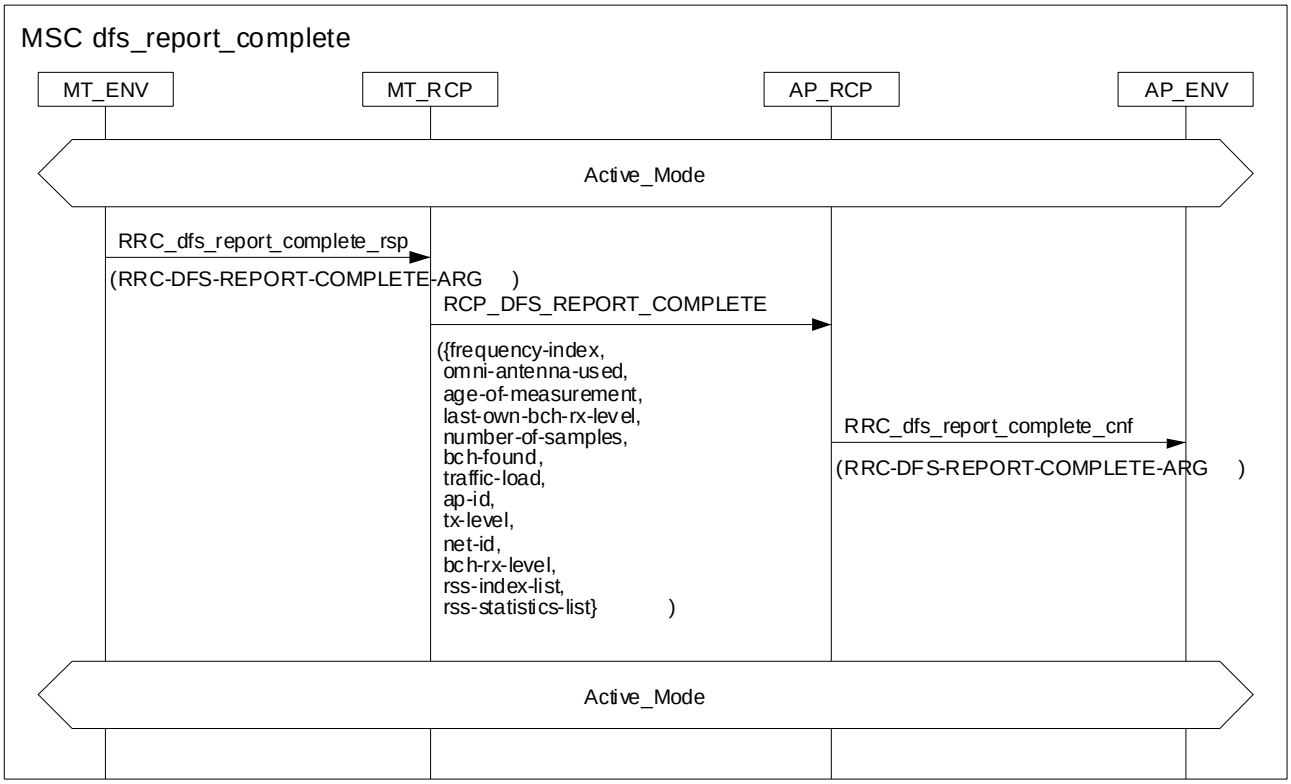


Diagram 51: DFS complete report

Table 67: RLC-DFS-REPORT-COMPLETE

RLC-DFS-REPORT-COMPLETE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
frequency-index	FREQUENCY-INDEX
omni-antenna-used	OMNI-ANTENNA-USED
age-of-measurement	AGE-OF-MEASUREMENT
last-own-bch-rx-level	LAST-OWN-BCH-RX-LEVEL
number-of-samples	NUMBER-OF-SAMPLES
bch-found	BCH-FOUND
traffic-load	TRAFFIC-LOAD
ap-id	AP-ID
tx-level	TX-LEVEL
net-id	NET-ID
bch-rx-level	BCH-RX-LEVEL
rss-index-list	RSS-INDEX-LIST
rss-statistics-list	RSS-STATISTICS-LIST }

5.2.2.6 Change Frequency

The AP shall broadcast the RLC_CHANGE_FREQUENCY message, before changing the operating frequency. The AP should use the MAC broadcast sleep group frames for broadcasting the RLC_CHANGE_FREQUENCY message.

The AP shall transmit the RLC_CHANGE_FREQUENCY message in RBCH more than once before the change of frequency.

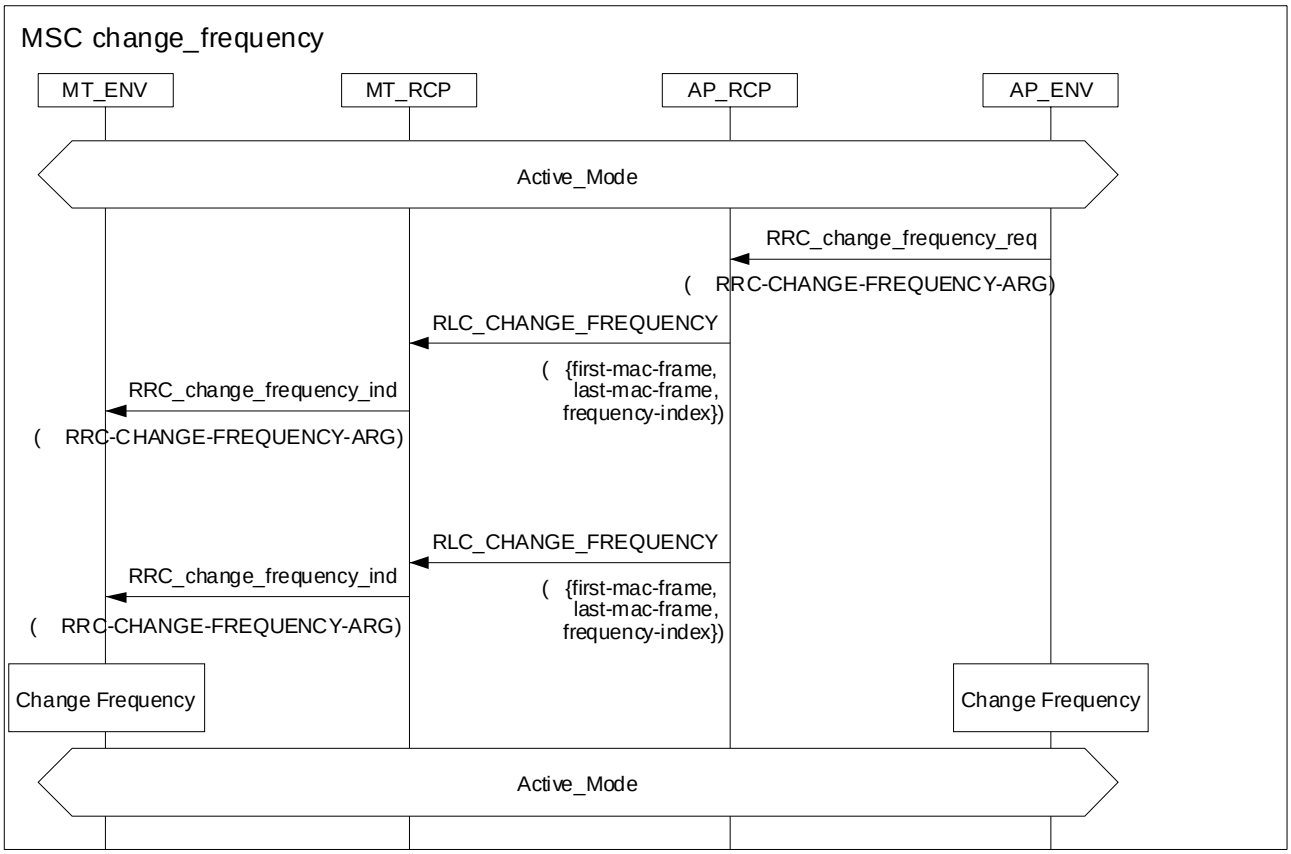


Diagram 52: Change Frequency procedure

Table 68: RLC-CHANGE-FREQUENCY

```
RLC-CHANGE-FREQUENCY-ARG ::= SEQUENCE {
    rlc-pdu-type        RLC-SCH-PDU-TYPE
    first-mac-frame     FIRST-MAC-FRAME
    last-mac-frame      LAST-MAC-FRAME

    frequency-index     FREQUENCY-INDEX }
```

5.2.3 Transmission Power Control

5.2.3.1 Uplink power control

The MT shall operate with a transmission power ≥ –15 dBm. The maximum output power is an arbitrary power level within the regulatory requirements. The transmission power range shall be composed of power steps equal to or smaller than 3 dB, and the transmitting MT shall ensure that the power levels shall provide monotonic transmission power. The MT shall define its' transmission power level at the ARP as:

> min (AP_Tx_Level – MT_received_power_level + AP_Rx_UL_Level + Σ(PC_Offset), AP_Tx_Level, maximum output power of MT)

Where AP_Tx_Level denotes access point transmit power level and AP_Rx_UL_Level stands for the power level the access point is expecting to receive for all uplink signals. These values are transmitted as part of the BCH information [3]. MT_received_power _level is the estimated power level of the signal received by the MT.

∑(PC_Offset) is the sum of the received *PC_Offset* values from the AP. It is optional for AP to send *PC_Offset* values. The algorithm when to send the RLC_UPLINK_PC_CALIBRATION message is vendor specific issue. To send the *PC_Offset* value, AP shall use the RLC_UPLINK_PC_CALIBRATION message dedicated to a single MT. When no *PC_Offset* value has been received from the AP, the value *∑(PC_Offset)* = 0 shall be used. The maximum minimum values for *∑(PC_Offset) shall be +15 and –20 dB, respectively*. The RLC_UPLINK_PC_CALIBRATION message shown in table 69 may be transmitted from the AP to the MT at any time after association. When a handover to another AP is performed by the MT, a PC-reset is performed, i.e. *∑(PC_Offset)* = 0. The AP may also force a PC-reset at any time. No reset is performed at sector handover. The MT shall apply the updated *∑(PC_Offset)* value no later than 2 MAC frames after the frame where the RLC_UPLINK_PC_CALIBRATION message was received. When the AP has transmitted an RLC_UPLINK_PC_CALIBRATION message to a MT, AP shall wait at least 10 MAC frames until the next calibration message may be transmitted to the MT.

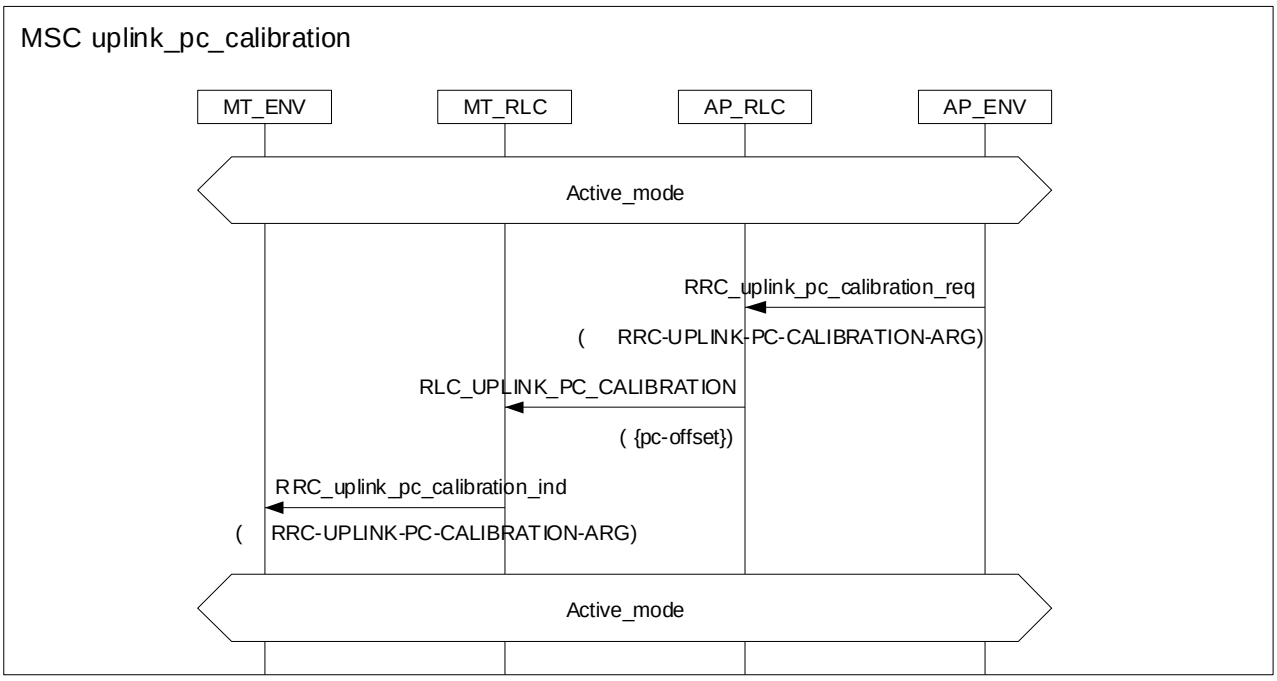


Diagram 53: Uplink power control calibration

Table 69: RLC-UPLINK-PC-CALIBRATION

```
RLC-UPLINK-PC-CALIBRATION-ARG ::= SEQUENCE {
    rlc-pdu-type   RLC-SCH-PDU-TYPE
    pc-offset      PC-OFFSET }
```

5.2.3.2 Downlink Power Control

Downlink power control is an implementation specific issue. In order to avoid interoperability problems the following restrictions apply to the default/normal downlink power control operation:

- The AP shall use the power levels and accuracy specified in [4].
- The AP power can be decreased rapidly (3 dB/frame) without limitation on the dynamic range.
- The AP transmitter power shall not be changed more than 3 dB between two consecutive MAC frames.
- The AP shall not increase the transmitter power more than three steps (9 dB) during any 5 minute interval.
- The AP shall ensure that it is compliant with the maximum allowed transmitted power for the center frequency where it is operating.

Spectrum regulatory requirements states that the interference to other radio systems than Hiperlan/2 shall be reduced as far as possible, but at least with 3 dB in average compared to transmission at full power by all APs and all MTs. This requirement implies that the AP should posses some degree of DL power control functionality.

5.2.3.3 Direct Link Power Control (OAP/OMT)

The MT shall be able to operate with a transmission power ≥ -15 dBm. The accuracies are given in [4].

A fixed power control shall be supported by all MTs and the AP/CC in DM. This fixed power control requires that each DM capable device shall set its transmission power level 3 dB below the maximum allowed transmitted power for the centre frequency where it is operating (see [4] clause 5.8.1.1). The usage of this fixed power control is only mandatory, if DiL power control using any other algorithm is not possible. Business or Home Environment Profiles (see [13] [14]) may specify a more accurate DiL power control scheme.

5.2.4 MT Alive

This function is used for checking that an MT and an AP can communicate with each other. In turn, this shall be used for deciding on whether the AP and the MT are still associated to each other and on whether a MAC ID is free to use or occupied.

If the AP sends RLC_MT_ALIVE_REQUEST message, the MT shall respond with the RLC_MT_ALIVE_REQUEST_ACK.

The AP should set *mt-alive-interval* time for the MT in the RLC_MT_ALIVE_REQUEST message. The MT shall respond to this message by sending RLC_MT_ALIVE_REQUEST_ACK message. Within the set interval, the MT shall send the RLC_MT_ALIVE message periodically to remain associated.

NOTE 1: The AP may also send the RLC_MT_ALIVE_REQUEST periodically, which should be the same period that the AP has set to the MT.

NOTE 2: If the mt-alive-interval is set to zero, the MT is not expected to send the RLC_MT_ALIVE message periodically, but the AP is expected to take care of the MT Alive function.

If the MT does not respond within the given period with the RLC_MT_ALIVE message, the AP should send the RLC_MT_ALIVE_REQUEST message to the MT. If the MT does not respond with RLC_MT_ALIVE_REQUEST_ACK when requested, the AP should remove the association of the particular MT (implicit disassociation procedure, no Disassociation message exchange).

There is a possibility to increase the number of failed MT Alive procedures (retransmissions included) before Disassociation takes place. The number can be from one to four failures.

NOTE 3: When the MT sends MT_ALIVE_REQUEST it becomes active.

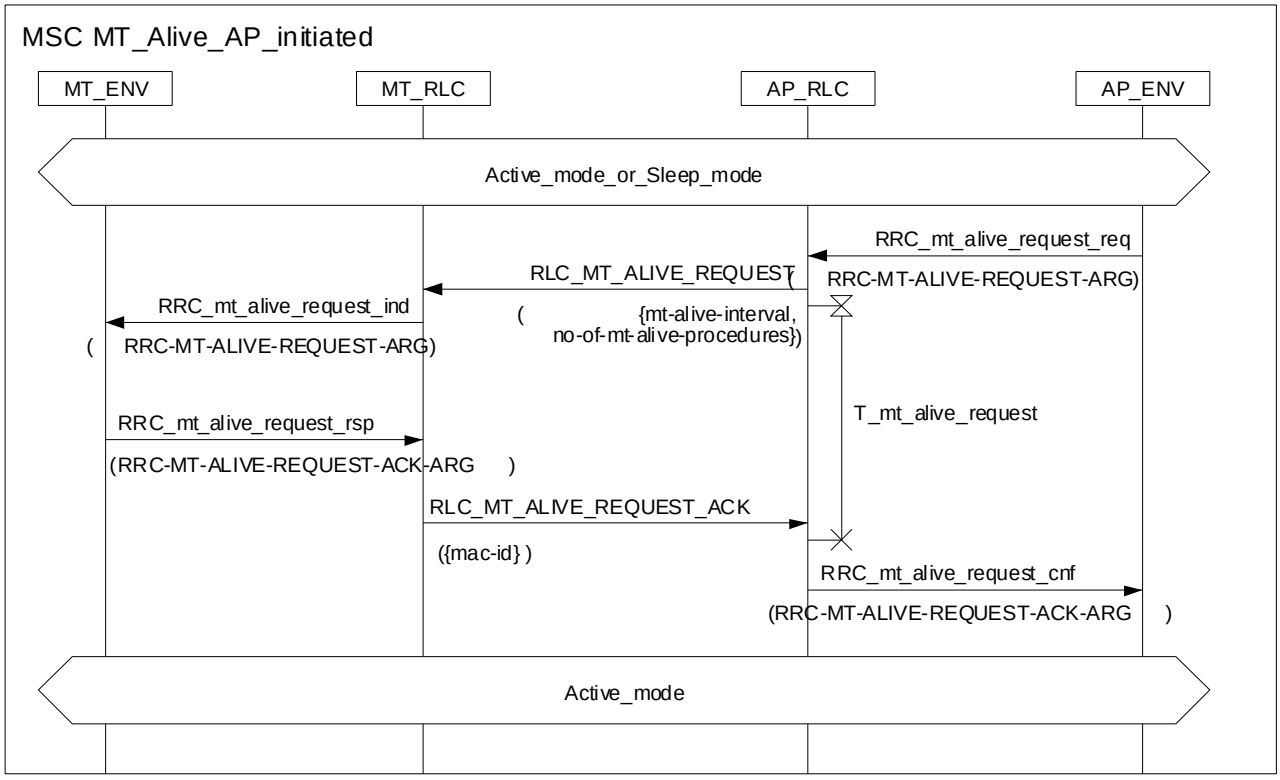


Diagram 54: MT alive procedure - AP initiated

Table 70: RLC-MT-ALIVE-REQUEST

```
RLC-MT-ALIVE-REQUEST-ARG ::= SEQUENCE {
    rlc-pdu-type        RLC-SCH-PDU-TYPE
    no-of-mt-procedures  NO-OF-MT-ALIVE-PROCEDURES
    mt-alive-interval   MT-ALIVE-INTERVAL    OPTIONAL }
```

Table 71: RLC-MT-ALIVE-REQUEST-ACK

```
RLC-MT-ALIVE-REQUEST-ACK-ARG ::= SEQUENCE {
    rlc-pdu-type   RLC-SCH-PDU-TYPE
    mac-id         MAC-ID }
```

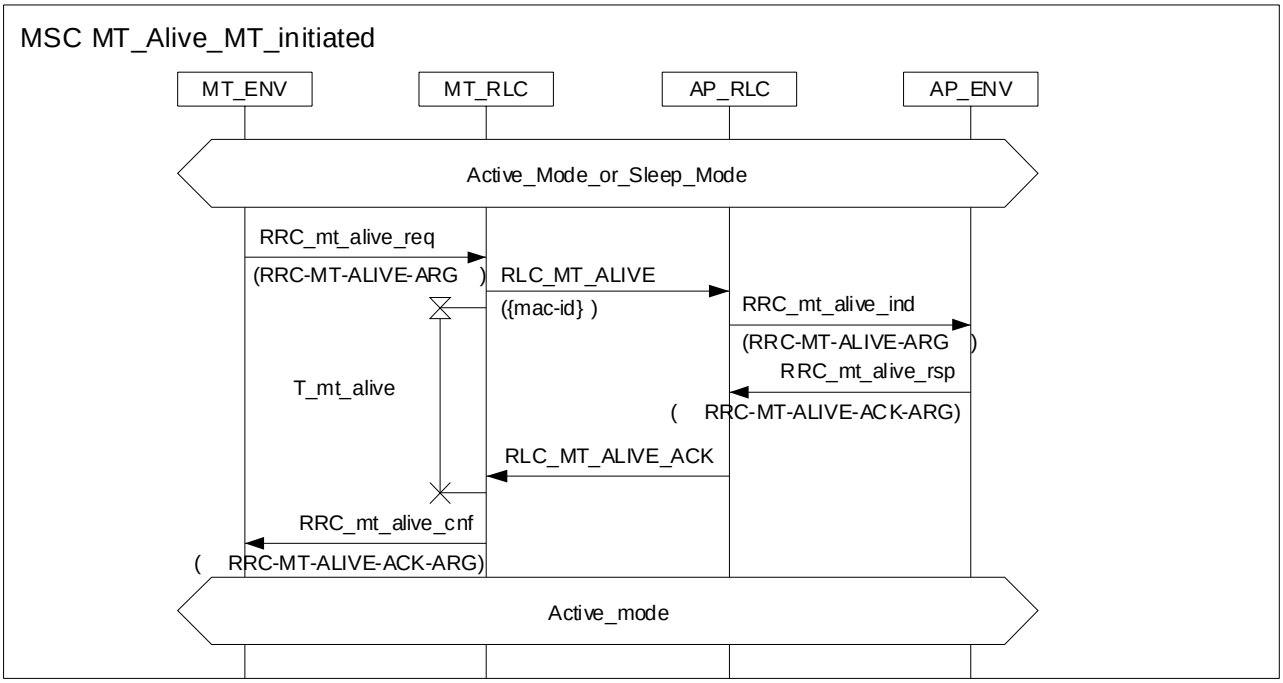


Diagram 55: MT alive procedure - MT initiated

Table 72: RLC-MT-ALIVE

```
RLC-MT-ALIVE-ARG ::= SEQUENCE {
    rlc-pdu-type   RLC-SCH-PDU-TYPE
    mac-id         MAC-ID }
```

Table 73: RLC-MT-ALIVE-ACK

```
RLC-MT-ALIVE-ACK-ARG ::= SEQUENCE {
    rlc-pdu-type   RLC-SCH-PDU-TYPE }
```

5.2.5 MT Absence (OAP/OMT)

This procedure is used by the MT to announce that it is temporary unavailable for the current AP, e.g. in order to perform measurements on neighbouring APs. During this time, no communication between MT and the current AP is possible.

The MT should inform the AP that it would be absent with the RLC_MT_ABSENCE message and tell the absence time. The AP shall respond with the RLC_MT_ABSENCE_ACK message. The absence time is started from the frame the RLC_MT_ABSENCE_ACK is received (the next frame is number 1, etc.). During this time the MT and the AP cannot communicate.. After the absence period (*n* frames) the MT and AP shall immediately continue with normal operation. After this period the MT shall execute the MT Alive sequence, if the MT has no data is to be transmitted. In all cases the AP may send the RLC_MT_ALIVE_REQUEST message after the absent period to check if the MT is available.

NOTE 1: The AP should take care of its own processing delays, when counting the returning moment of the MT.

NOTE 2: The MT should take care of its own processing delays, when counting the returning moment to the AP.

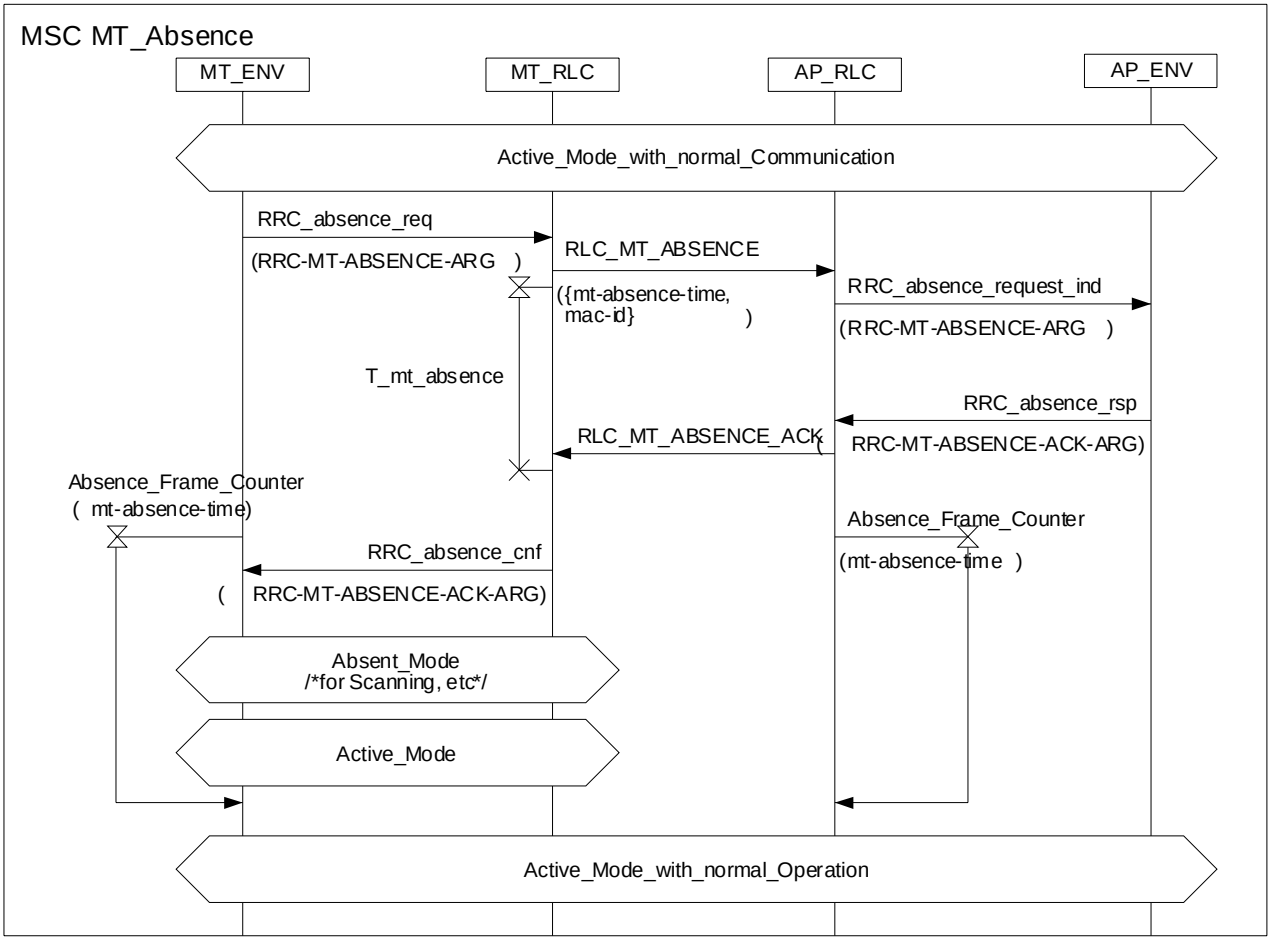


Diagram 56: MT absence procedure

Table 74: RLC-MT-ABSENCE

```
RLC-MT-ABSENCE-ARG ::= SEQUENCE {
    rlc-pdu-type        RLC-SCH-PDU-TYPE
    mt-absence-time MT-ABSENCE-TIME
    mac-id              MAC-ID }
```

Table 75: RLC-MT-ABSENCE-ACK

```
RLC-ABSENCE-ACK-ARG ::= SEQUENCE {
    rlc-pdu-type        RLC-SCH-PDU-TYPE }
```

5.2.6 Power Saving (OMT)

5.2.6.1 General

To allow reduced power consumption for mobile terminals the power save function can be used.

A mobile can enter one out of sixteen different sleep groups to reduce its power consumption. The term sleep mode is used to refer to a mobile that has joined one of the sleep groups.

A mobile terminal in sleep mode shall monitor the BCCH content periodically, as opposed to active mode where mobile terminal shall monitor the BCCH in every frame. The term wake-up frame refers to the frame where the mobile terminal in sleep mode monitors the BCCH.

The AP/CC should maintain an internal sleep state for each MT being either *active* or *sleep*. The MT should maintain an internal sleep state being either *active* or *sleep*.

In order to enter sleep mode the MT shall request the AP by sending a RLC_SLEEP message and receive a positive acknowledgement in RLC_SLEEP_ACK message.

The sleep state shall be changed to *sleep* at the AP at the transmission of the RLC_SLEEP_ACK message. The sleep state should be changed to *sleep* at the MT at the reception of the RLC_SLEEP_ACK message indicating an acceptance to enter sleep mode.

The AP should change the sleep state to *active* at arbitrary message reception from an MT with current sleep state *active*. The MT shall change its sleep state to *active* at decoding of the FCCH IE with a matching MAC ID that corresponds to the MT's MAC ID if current sleep state is *sleep*. The MT shall change its sleep state to active at arbitrary message transmission.

NOTE 1: As is given by the rules above for *sleep* → *active* transitions, the MT will always be in *active* state if it receives an arbitrary unicasted message.

NOTE 2: According to the rules above will broadcasted or multicasted messages, i.e. UBCH, UMCH and RBCH, not cause any change of sleep state for an MT.

Sixteen sleep groups exists, where the sleep mode periodicity is given by:

2^n with ($1 \leq n \leq 16$) with the unit frames.

The AP shall coordinate the sleep groups such that the periodicity for all sleep groups coincide with the periodicity for sleep group with n = 1. The latter will guarantee that for arbitrary sleep group m (m > 1), all sleep groups with shorter periodicity will at regular interval have their wake up frames equal to wake up frame for sleep group m.

An MT in sleep mode shall monitor the BCCH at the wake-up frames that corresponds to the sleep group. Upon such a wake-up frame, if the BCCH DST [5] indication is active, the MT shall proceed to decode the FCCH. The FCCH may indicate a change of sleep state to *active*, or may indicate the presence of granted UBCH or UMCH in the frame. The DST indication also requires the MT to decode the following frame and apply the same decoding rules as for the current wake up frame. The BCCH may also contain an DL RBCH Indicator, that when set active requires the MT to decode the FCCH. The wakeup will in that case contain a granted RBCH.

The AP shall internally select a MAC broadcast sleep group, denoted n_{AP}, out of a subset from the sixteen sleep groups. An exemplatory wake-up frame for the MAC broadcast sleep group is denoted MAC broadcast frame.

The periodicity of the MAC broadcast sleep group is given by:

2^{nAP} with ($5 \leq n \leq 16$) with the unit frames.

MAC broadcast frames can be used by the AP to transmit multicast and broadcast data since the wake-up frame for all MTs in sleep mode, with sleep group $n \leq n_{AP}$, will coincide with MAC broadcast frames.

The AP may transmit multicast and broadcast data in the following frames after a MAC broadcast frame by using the DST indication in the BCCH set active until the transmission is completed. The MTs will then revert to their active/sleep state they had prior to the MAC broadcast frame. This allows for a scalable broadcast and multicast bandwidth for sleeping MTs.

For MTs with a sleep group $n < n_{AP}$, the wake-up frames in between the MAC broadcast frames are denoted subBroadcast frames.

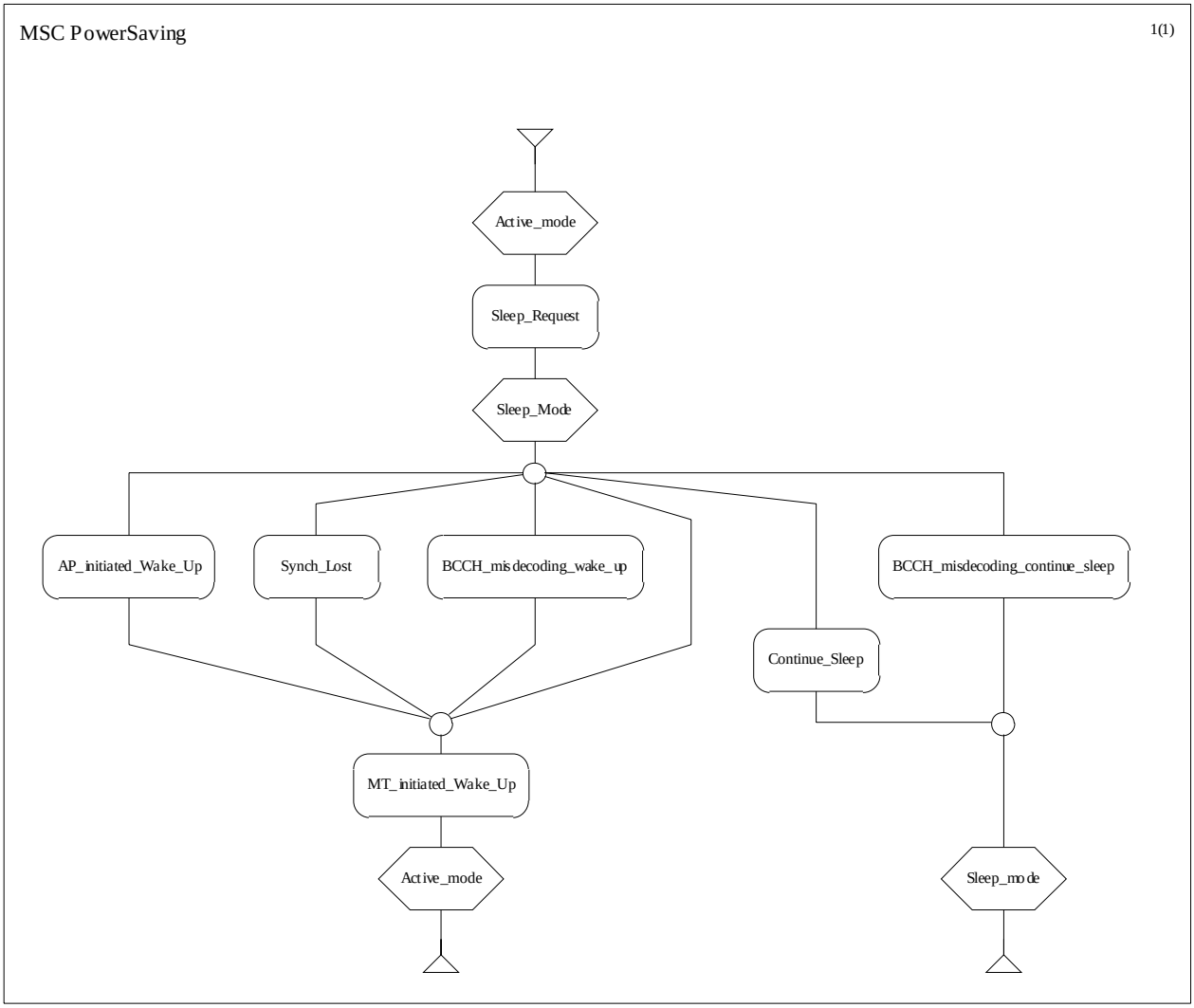


Diagram 57: Power saving procedures

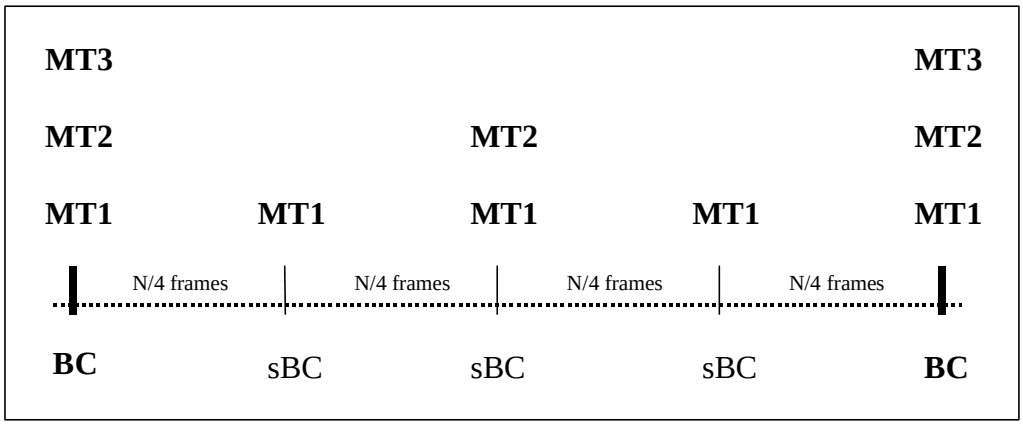


Figure 7: Example of the MAC Broadcast frames and subBroadcast frames

NOTE 3: In the example the MT3 listens only the MAC broadcast frames (repeated once per N frames), the MT2 has a sleep mode period of N/2 frames and the MT1 has a sleep mode period of N/4 frames. Thus, the MT2 listens to MAC broadcast frames and always one subBroadcast frame between MAC broadcast frames. The MT1 listens to three subBroadcast frames between MAC broadcast frames. The interval of subBroadcast frames is the result of allocating different, but synchronized sleep mode periods for MTs.

5.2.6.2 MT sleep request procedure

The MT shall not send RLC_SLEEP message (enter the sleep mode) during the association and handover procedures, see clauses 5.1.1 and 5.2.1.

Apart from the exception above, the MT may at any time request to enter sleep mode, by sending the RLC_SLEEP message. The message shall contain a proposed *sleep-group*. The *sleep-group* shall be:

$$1 \leq \text{Sleep-group} \leq 16$$

NOTE: The corresponding sleep mode periodicity is $2^{\text{sleep-group}}$ with the unit frames.

Based upon the internally selected MAC broadcast sleep group, n_{AP}, the AP shall determine the sleep group according to the following formulas:

- If *sleep-group* $\leq n_{AP}$, then the sleep group for the MT shall be equal to the proposed *sleep-group*.
- If *sleep-group* $> n_{AP}$, then the sleep group for the MT shall be equal to n_{AP}.

Since the MT may request for sleep mode at any frame the AP shall set an *offset* parameter to align the MT to coincide with the periodicity that the AP has for the selected *sleep-group*.

The *offset* parameter shall be the number of frames after the current frame, in which the RLC_SLEEP_ACK is transmitted, that shall elapse until the first wake-up frame appears for the selected *sleep-group*.

$$0 \leq \mathit{Offset} \leq (2^{\mathit{sleep\text{-}group}} - 1) \text{ with the unit frames}$$

EXAMPLE: The AP has determined the *sleep-group* for an MT to be 2, with a sleep mode periodicity of 4 frames.
If the RLC_SLEEP_ACK is transmitted in the wake-up frame for the sleep group, the *offset* parameter will be equal to 3.
If the RLC_SLEEP_ACK is transmitted in the frame prior to the wake-up frame for the sleep group, the *offset* parameter will be equal to 0.

If the AP sets the *sleep-group* to zero, the MT shall not enter sleep mode.

The response to a RLC_SLEEP shall be sent in RLC_SLEEP_ACK.

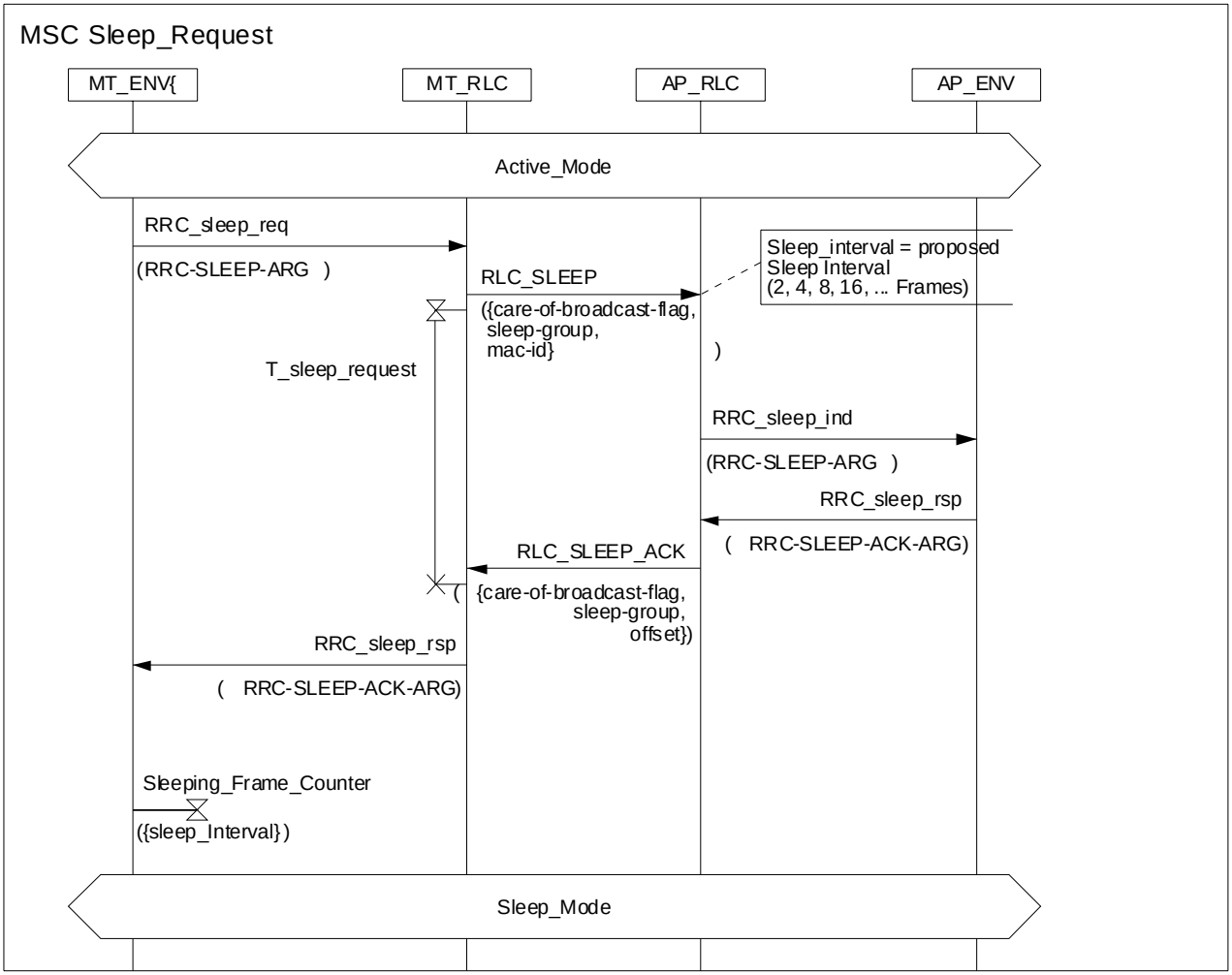


Diagram 58: Sleep request procedure

Table 76: RLC-SLEEP

RLC-SLEEP-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE,
care-of-broadcast	CARE-OF-BROADCAST
sleep-group	SLEEP-GROUP
mac-id	MAC-ID }

Table 77: RLC-SLEEP-ACK

RLC-SLEEP-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
care-of-broadcast	CARE-OF-BROADCAST
sleep-group	SLEEP-GROUP
offset	FRAME-NUM }

5.2.6.3 AP Procedure

5.2.6.3.1 AP Procedure for unicast data

At the occurrence of pending unicast data to an MT with sleep state equal *sleep*, the AP shall initiate an AP wake-up procedure of the MT prior to transmission of the unicast data. The AP wake-up procedure shall occur at arbitrary wake-up frame for the corresponding *sleep-group* for the MT.

At the wake-up frame, the AP shall:

1) set the DST indication active in the BCCH [5];

2) allocate one uplink RG for DCCH (DLCC ID = 0) with one granted SCH for the corresponding MT.

In order to improve the response probability from a MT in *sleep*, the AP may repeat the AP wake-up procedure at the subsequent frame after the wake-up frame. More generally, since the MT wake-up procedure is repeated until the DST indication is inactive, the AP may repeat the AP wake-up procedure at arbitrary frame until the DST indication is inactive.

At the reception of arbitrary data from an MT with sleep state *sleep* the AP should change the sleep state to *active*.

5.2.6.3.2 AP Procedure for broadcast data

At the occurrence of pending broadcast data (i.e. UMCH and/or UBCH), with exception for data transmitted in RBCH, the AP may defer transmission until the MAC broadcast frame. If used, the AP shall set the DST indication in the BCCH to indicate the possible presence of either UMCH or UBCH data in the frame.

For broadcast data that spans over multiple MAC frames, or due to the use of repetition mode, the AP may use the bandwidth scalability option to send UMCH and/or UBCH data at subsequent frames after the MAC broadcast frame. The DST indication shall be set in all subsequent frames after the MAC broadcast frame until the UMCH and/or UMCH transmission is completed.

NOTE: Due to the MT wake-up procedure, the AP does not necessarily have to transmit UBCH and/or UMCH data in a frame where the DST indication is active.

If the AP does not use the MAC broadcast frame for the transmission of broadcast data (i.e. UMCH and/or UBCH), with exception for data transmitted in RBCH, the DST indication shall be inactive.

At the occurrence of pending broadcast data transmitted in RBCH, the AP may defer transmission until a MAC broadcast frame, or arbitrary wake-up frame.
The DL RBCH Indication in BCCH [5] shall be active for all frames where RBCH is transmitted.

5.2.6.4 MT Procedure

For an MT in sleep mode, the MT shall initiate the MT wake-up procedure at the wake-up frames for the corresponding sleep group. At the wake-up frame the MT shall decode the BCCH for the occurrence of the DST or DL RBCH indication [5].

If the DST indication is active and/or DL RBCH Indicator is active, the MT shall:

1) Decode the FCCH:

   - At the occurrence of an IE with MAC ID that corresponds to the MAC ID of the MT, the sleep state shall be set to *active*.

NOTE 1: An MT that changes its sleep state from *sleep* to *active* shall proceed according to the normal procedure for decoding FCCH.

   - At the occurrence of UBCH and/or UMCH, the MT receives UBCH and/or UMCH in the frame.

   - At the occurrence of RBCH, the MT receives the RBCH in the frame.

2) Decode the BCCH in the following frame and restart the MT wake-up procedure:

   - If the DST indication is inactive and DL RBCH Indication is inactive, and the MT sleep state is *sleep*, the MT should revert to its low power mode until the expiration of the next wake-up frame.

At the occurrence of pending data transmission for an MT in sleep state *sleep*, the sleep state should be changed to *active*.

NOTE 2: An MT with sleep state *active* will continue by transmitting a resource request, which will change the sleep state from *sleep* to *active* at the AP. Normal data transmission will follow.

Since it is likely that the MT from time to time will have failure in the CRC check of the BCH, or due to an internal MT error initiate the MT wake-up procedure at an incorrect frame. To mitigate the effect of the latter erroneous conditions the following rules shall be supported:

- If the CRC check of the BCH fails for a frame where MT wake-up procedure is started, the sleep state of the mobile shall be unchanged, and the MT shall re-start the wake-up procedure in the following frame.

  - If the CRC check of the BCH for the following frame is successful, the MT wake-up procedure shall continue.

  - If the CRC check of the BCH for the following frame fails, the MT sleep state should be set to *active*.

- If the frame counter in the BCCH differs from the frame counter in MT, the MT should align its own counter value according to the frame counter of the BCCH.

NOTE 3: The MT has to take into account in which frame after the wake-up frame the MT wake-up procedure is started.

5.3 Services supporting DUCC (DLC User Connection Control)

This control function is responsible for setting up, maintaining, re-negotiating and closing a DLC user connection at the DLC layer. Its main functions are:

- Allocating the DLCC-ID for a new specific connection.

- Setting up the connection at DLC layer.

The MT is not allowed to send any DUCC message before being associated to the AP.

5.3.1 Unicast DUC Setup

Unicast radio connection can be requested both by the AP and by the MT using the RLC_SETUP message. In the message the selected DLCC IDs and corresponding characteristics shall be included.

The RLC_SETUP message may be sent at multiple occasions since the maximal numbers of DUCs are limited in the message. The *duc-ext-ind* shall be set inactive for the first RLC_SETUP message and active for all subsequent RLC_SETUP messages.

NOTE 1: The *duc-ext-ind* mechanism is especially relevant during the network handover procedure, when the number of DUCs that have to be re-established via the radio interface are more than what can be sent in one RLC_SETUP message.

NOTE 2: DLCC IDs for mobile originating RLC_SETUP messages does not have to be set. The *dlcc-id* values proposed by an MT can be ignored by the AP.

The RLC_SETUP message shall not be used to modify existing DUCs.

If the receiver of the RLC_SETUP message being either the AP or the MT is not able to accept the proposal made by the sender, it shall send an RLC_CONNECT message containing the receiver's proposal. To accept the proposal made by the sender, the receiver shall repeat the proposal of the sender in the RLC_CONNECT message. In order to reject the RLC_SETUP the receiver shall send RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

If the sender accepts the receivers proposal sent in the RLC_CONNECT message, the sender shall respond with RLC_CONNECT_ACK message. Otherwise the sender shall send RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

5.3.1.1 AP Initiated DUC Setup (OAP/OMT)

The AP can send RLC_SETUP message in order to establish unicast radio connections. In the message the selected DLCC IDs and corresponding characteristics shall be included. The AP sender shall not transmit downlink traffic to the DUCs that are included in the RLC_SETUP message until RLC_CONNECT message is received.

If the AP accepts the MT's proposal of the DUC's included in the RLC_CONNECT message, the AP shall respond with RLC_CONNECT_ACK message.

At the reception of the RLC_CONNECT message, for the DUCs included in the RLC_CONNECT message, the AP shall:

- Discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific).
- Set TxBoW and RxBoW to 0.
- If the sender and/or receiver are in flow control state, exit the flow control state.
- Clear all other ARQ state information, i.e. acknowledgement bitmaps.
- Set the DUC characteristics as included in the RLC_CONNECT message.

If the AP does not accept the MT's proposal sent in the RLC_CONNECT message, the AP shall send the RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

If the MT is not able to accept the proposal made by the AP in the RLC_SETUP message, it shall send RLC_CONNECT message containing the MT's proposal. To accept the proposal made by the AP, the MT shall repeat the proposal of the AP in the RLC_CONNECT message. The MT sender shall not transmit uplink traffic to the DUCs that are included in the RLC_CONNECT message until RLC_CONNECT_ACK message is received.

At the transmission of the of the RLC_CONNECT message, for the DUCs included in the RLC_CONNECT message, the MT shall:

- Discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific).
- Set TxBoW and RxBoW to 0.
- If the sender and/or receiver are in flow control state, exit the flow control state.
- Clear all other ARQ state information, i.e. acknowledgement bitmaps and rtt awareness for resource request.
- Set the DUC characteristics as included in the RLC_CONNECT message.

In order to reject the RLC_SETUP the MT shall send RLC_RELEASE message and continue with the Release procedure, as described in 5.3.2.

NOTE: DLCC IDs for mobile originating RLC_SETUP messages does not have to be set. The *dlcc-id* values proposed by an MT can be ignored by the AP.

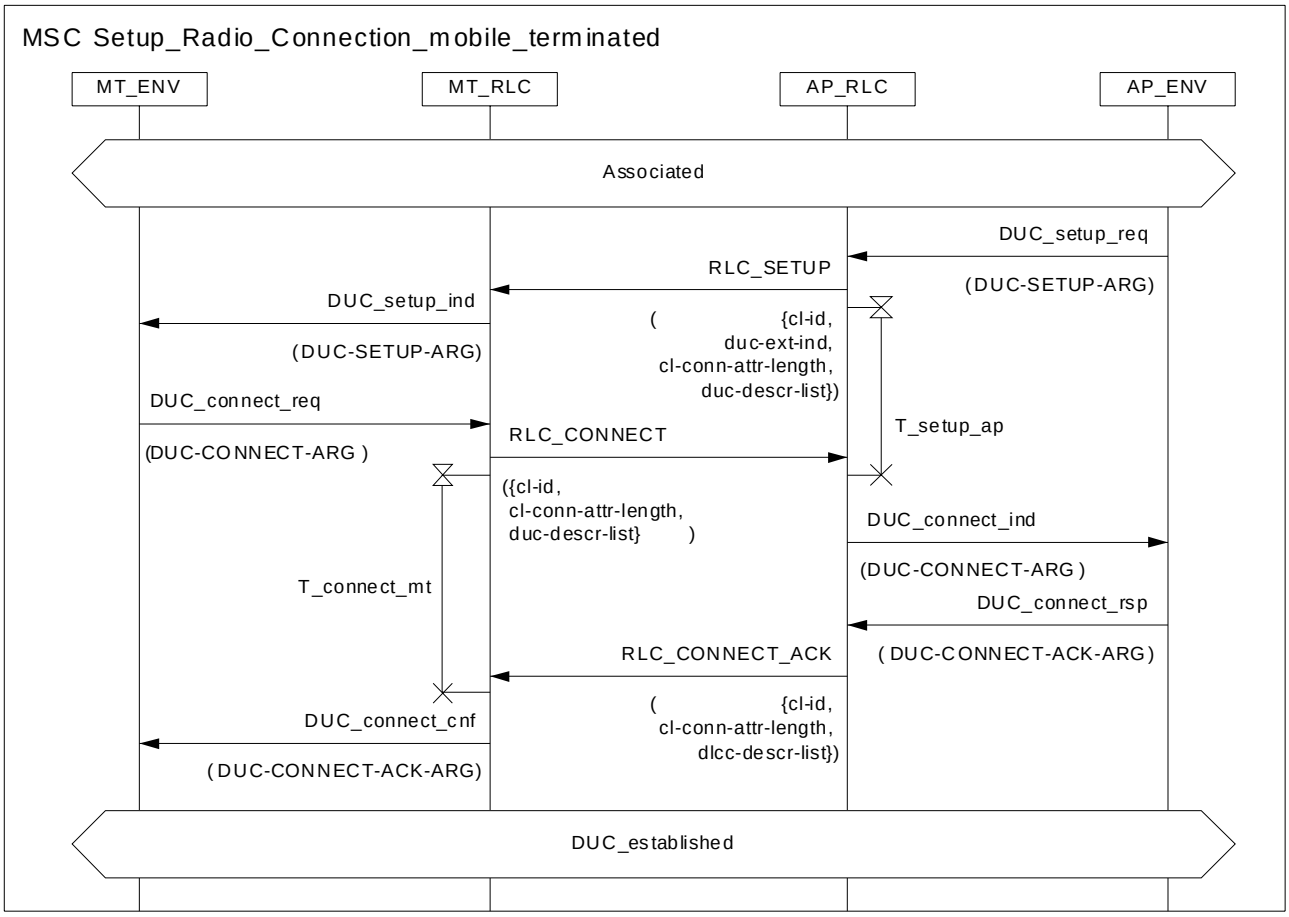


Diagram 59: Mobile terminated connection Setup procedure

Table 78: RLC-SETUP

RLC-SETUP-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-id	CL-ID
duc-ext-ind	DUC-EXT-IND
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 79: RLC-CONNECT

RLC-CONNECT-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-id	CL-ID
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 80: RLC-CONNECT-ACK

RLC-CONNECT-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-id	CL-ID
cl-conn-attr-length	INTEGER(0..31)
dlcc-descr-list	DLCC-DESCR-LIST }

5.3.1.2 MT initiated DUC Setup

The MT can send RLC_SETUP message in order to establish unicast radio connections. In the message the selected DLCC IDs and corresponding characteristics shall be included. The MT sender shall not transmit uplink traffic to the DUCs that are included in the RLC_SETUP message until RLC_CONNECT message is received.

If the MT accepts the AP's proposal of the DUC's included in the RLC_CONNECT message, the MT shall respond with RLC_CONNECT_ACK message.

At the reception of the RLC_CONNECT message, for the DUCs included in the RLC_CONNECT message, the MT shall:

- Discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific).
- Set TxBoW and RxBoW to 0.
- If the sender and/or receiver are in flow control state, exit the flow control state.
- Clear all other ARQ state information, i.e. acknowledgement bitmaps and rtt awareness for resource request.
- Set the DUC characteristics as included in the RLC_CONNECT message.

If the AP is able to accept the proposal made by the MT in the RLC_SETUP message completely, it shall send RLC_CONNECT message containing the MT's proposal. To accept the proposal made by the MT, the AP shall repeat the proposal of the MT in the RLC_CONNECT message. The AP sender shall not transmit downlink traffic to the DUCs that are included in the RLC_CONNECT message until RLC_CONNECT_ACK message is received.

If the AP does not accept the proposal made by the MT completely, it shall send an RLC_CONNECT message with the AP's new proposal.

If the MT does not accept the AP's proposal sent in the RLC_CONNECT message, the MT shall send the RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

If the AP is not able to accept the proposal made by the MT and can not send an alternative to the MT, the AP shall send an RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

At the transmission of the of the RLC_CONNECT message, for the DUCs included in the RLC_CONNECT message, the AP shall:

- Discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific).
- Set TxBoW and RxBoW to 0.
- If the sender and/or receiver are in flow control state, exit the flow control state.
- Clear all other ARQ state information, i.e. acknowledgement bitmaps.
- Set the DUC characteristics as included in the RLC_CONNECT message.

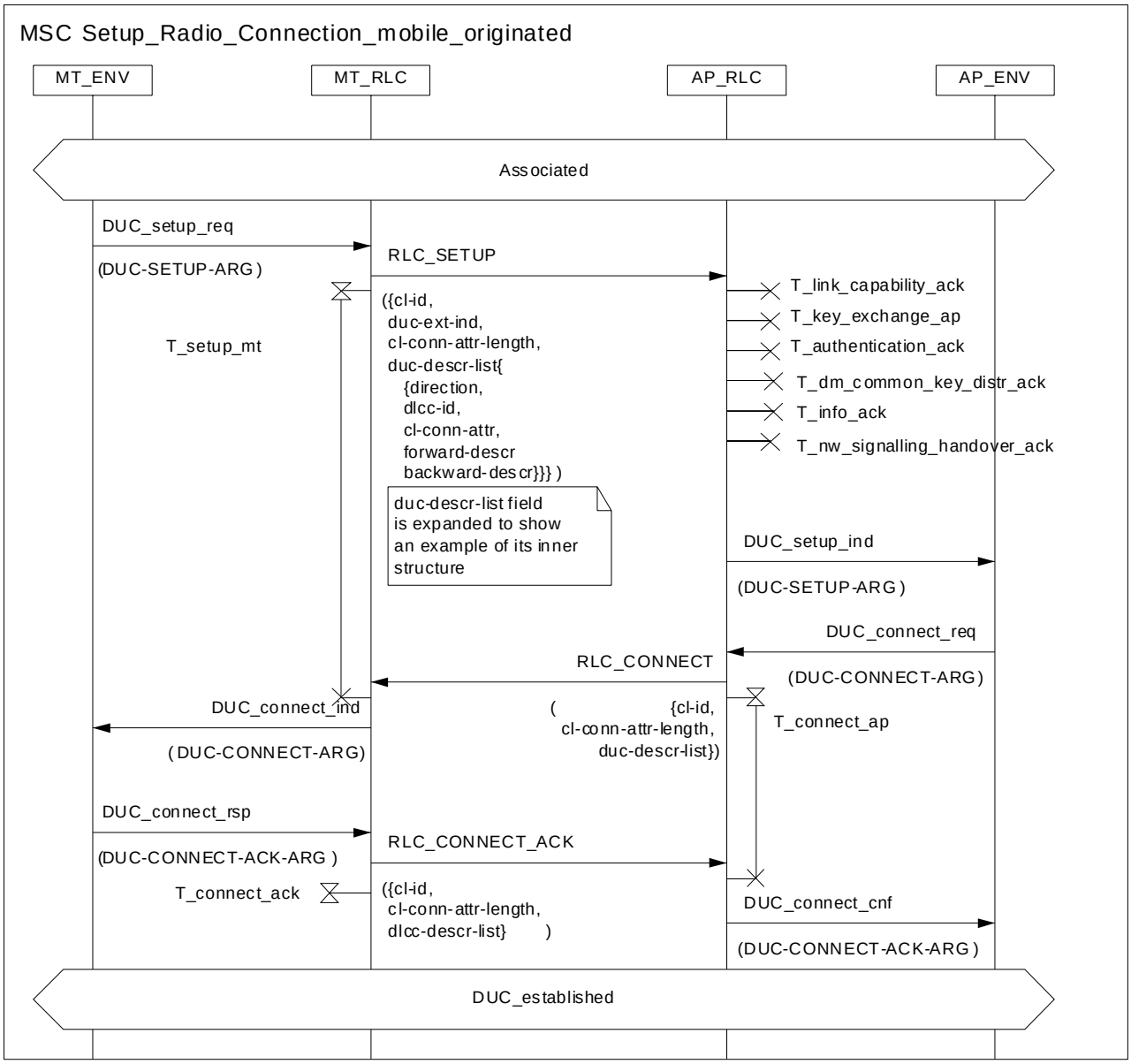


Diagram 60: Mobile originated connection Setup procedure

If AP is not able to accept the proposal made by MT, it shall send RLC_CONNECT message containing AP's proposal. To accept the proposal made by the MT, the AP shall repeat the proposal of the MT in the RLC_CONNECT message. In order to reject the SETUP the AP shall send RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

If the MT accepts AP's proposal sent in the RLC_CONNECT message, the MT shall respond with RLC_CONNECT_ACK message. Otherwise the MT shall send RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

5.3.2 Unicast DUC release

5.3.2.1 AP Initiated DUC Release

In case of an AP Initiated Connection Release the AP shall send an RLC_RELEASE message commanding the MT to release one or multiple unicast DUCs. In this message the AP shall indicate to the MT the selected DLCC IDs of the respective DUCs.

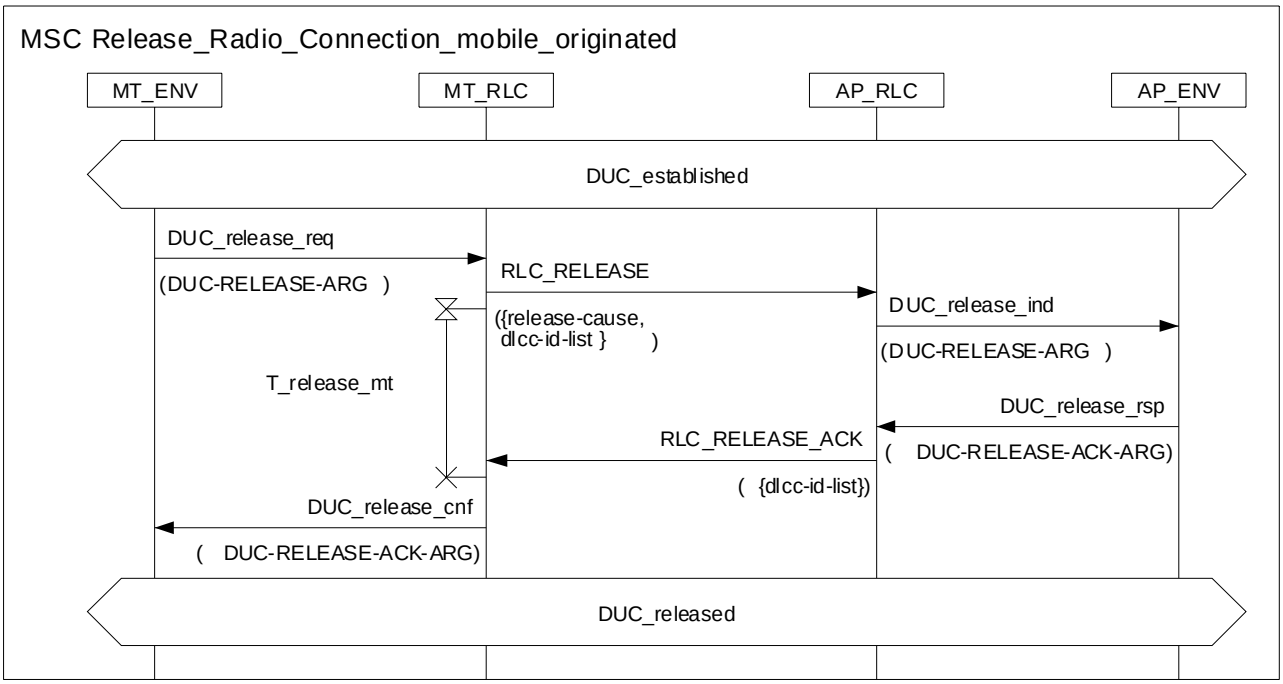


Diagram 61: Mobile terminated connection release procedure

Table 81: RLC-RELEASE

RLC-RELEASE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
release-cause	RELEASE-CAUSE
dlcc-id-list	DLCC-ID-LIST }

Table 82: RLC-RELEASE-ACK

RLC-RELEASE-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
dlcc-id-list	DLCC-ID-LIST }

The MT shall respond with RLC_RELEASE_ACK indicating that the relevant DLCC IDs are released.

5.3.2.2 MT Initiated DUC release

In case of an MT Initiated Connection Release the MT shall send an RLC_RELEASE message requesting the AP to release one or multiple unicast DUCs. In this message the MT shall indicate to the AP the selected DLCC IDs of the respective DUCs.

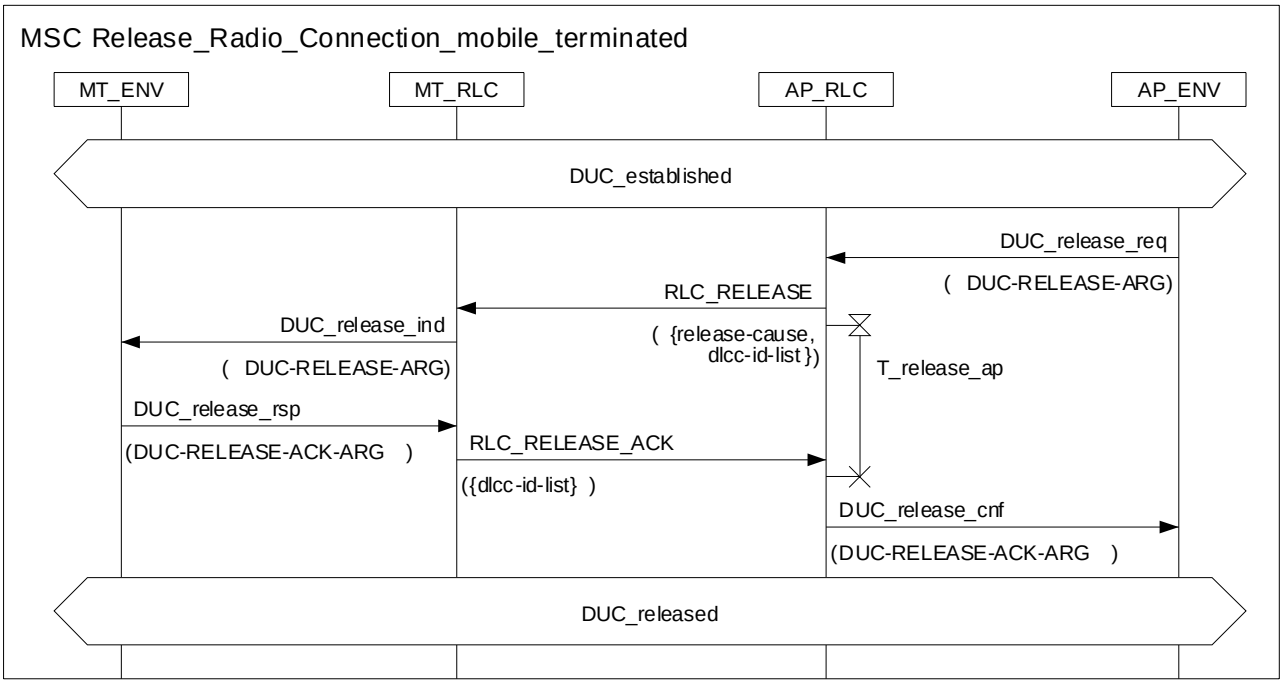


Diagram 62: Mobile originated connection release procedure

The AP shall respond with RLC_RELEASE_ACK indicating that the relevant DLCC IDs are released.

5.3.3 Unicast DUC modify (OAP/OMT)

Existing unicast radio connection can be modified by both the AP and by the MT using the RLC_MODIFY_REQ message. In the message the selected DLCC IDs of the respective unicast DUCs and their new characteristics shall be included.

The RLC_SETUP message may be sent at multiple occasions changing the characteristics for the same DUC.

The RLC_MODIFY_REQ message shall be used to modify existing DUCs.

If the receiver of the RLC_MODIFY_REQ message being either the AP or the MT is not able to accept the proposal made by the sender, it shall send an RLC_MODIFY message containing the receiver's proposal. To accept the proposal made by the sender, the receiver shall repeat the proposal of the sender in the RLC_MODIFY message.

If the sender accepts the receivers proposal sent in the RLC_MODIFY message, the sender shall respond with RLC_MODIFY_ACK message. Otherwise the sender shall send RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

5.3.3.1 AP Initiated DUC modify

The AP can send RLC_MODIFY_REQ message in order to modify existing unicast radio connections. In the message the selected DLCC IDs of the respective unicast DUCs and their new characteristics shall be included. The AP sender shall either stop transmitting downlink traffic to the DUCs that are included in the RLC_MODIFY_REQ message, or transmit dummy PDUs, until RLC_MODIFY message is received.

If the AP accepts the MT's proposal of the DUC's included in the RLC_MODIFY message, the AP shall respond with RLC_MODIFY_ACK message:

At the reception of the RLC_MODIFY message, for the DUCs included in the RLC_MODIFY message, the AP shall:

- Discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific).
- Set TxBoW and RxBoW to 0.
- If the sender and/or receiver are in flow control state, exit the flow control state.
- Clear all other ARQ state information, i.e. acknowledgement bitmaps.
- Set the DUC characteristics as included in the RLC_MODIFY message.

If the AP does not accept the MT's proposal sent in the RLC_MODIFY message, the AP shall send the RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

If the MT is not able to accept the proposal made by the AP in the RLC_MODIFY_REQ message, it shall send RLC_MODIFY message containing the MT's proposal. To accept the proposal made by the AP, the MT shall repeat the proposal of the AP in the RLC_MODIFY message. The MT sender shall either stop transmitting uplink traffic to the DUCs that are included in the RLC_CONNECT message, or transmit dummy PDUs, until RLC_MODIFY_ACK message is received.

At the transmission of the of the RLC_MODIFY message, for the DUCs included in the RLC_MODIFY message, the MT shall:

- Discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific).
- Set TxBoW and RxBoW to 0.
- If the sender and/or receiver are in flow control state, exit the flow control state.
- Clear all other ARQ state information, i.e. acknowledgement bitmaps and rtt awareness for resource request.
- Set the DUC characteristics as included in the RLC_MODIFY message.

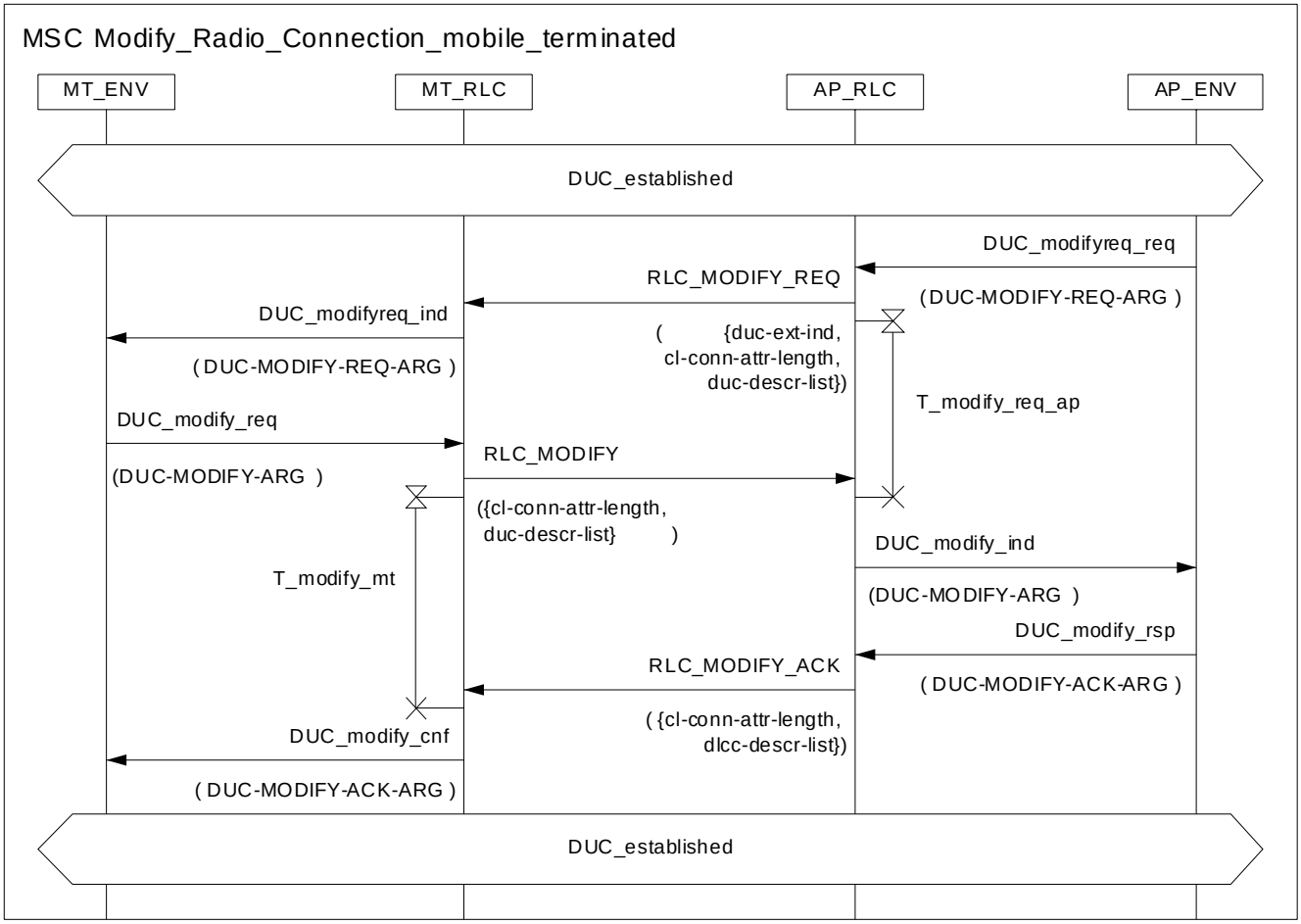


Diagram 63: Mobile terminated connection modify procedure

Table 83: RLC-MODIFY-REQ

RLC-MODIFY-REQ-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
duc-ext-ind	DUC-EXT-IND
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 84: RLC-MODIFY

RLC-MODIFY-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 85: RLC-MODIFY-ACK

RLC-MODIFY-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
cl-conn-attr-length	INTEGER(0..31)
dlcc-descr-list	DLCC-DESCR-LIST }

5.3.3.2 MT Initiated DUC modify

The MT can send RLC_MODIFY_REQ message in order to modify existing unicast radio connections. In the message the selected DLCC IDs of the respective unicast DUCs and their new characteristics shall be included. The MT sender shall either stop transmitting downlink traffic to the DUCs that are included in the RLC_MODIFY_REQ message, or transmit dummy PDUs, until RLC_MODIFY message is received.

If the MT accepts the MT's proposal of the DUC's included in the RLC_MODIFY message, the MT shall respond with RLC_MODIFY_ACK message.

At the reception of the RLC_MODIFY message, for the DUCs included in the RLC_MODIFY message, the MT shall:

- discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific);
- set TxBoW and RxBoW to 0;
- if the sender and/or receiver are in flow control state, exit the flow control state;
- clear all other ARQ state information, i.e. acknowledgement bitmaps and rtt awareness for resource request;
- set the DUC characteristics as included in the RLC_MODIFY message.

If the MT does not accept the MT's proposal sent in the RLC_MODIFY message, the MT shall send the RLC_RELEASE message and continue with the Release procedure, as described in clause 5.3.2.

If the AP is not able to accept the proposal made by the MT in the RLC_MODIFY_REQ message, it shall send RLC_MODIFY message containing the AP's proposal. To accept the proposal made by the MT, the AP shall repeat the proposal of the MT in the RLC_MODIFY message. The AP sender shall either stop transmitting uplink traffic to the DUCs that are included in the RLC_CONNECT message, or transmit dummy PDUs, until RLC_MODIFY_ACK message is received.

At the transmission of the of the RLC_MODIFY message, for the DUCs included in the RLC_MODIFY message, the AP shall:

- discard all data in its reception buffer and optionally discard all data in its transmission buffer (the decision is vendor specific);
- set TxBoW and RxBoW to 0;
- if the sender and/or receiver are in flow control state, exit the flow control state;
- clear all other ARQ state information, i.e. acknowledgement bitmaps.

Set the DUC characteristics as included in the RLC_MODIFY message.

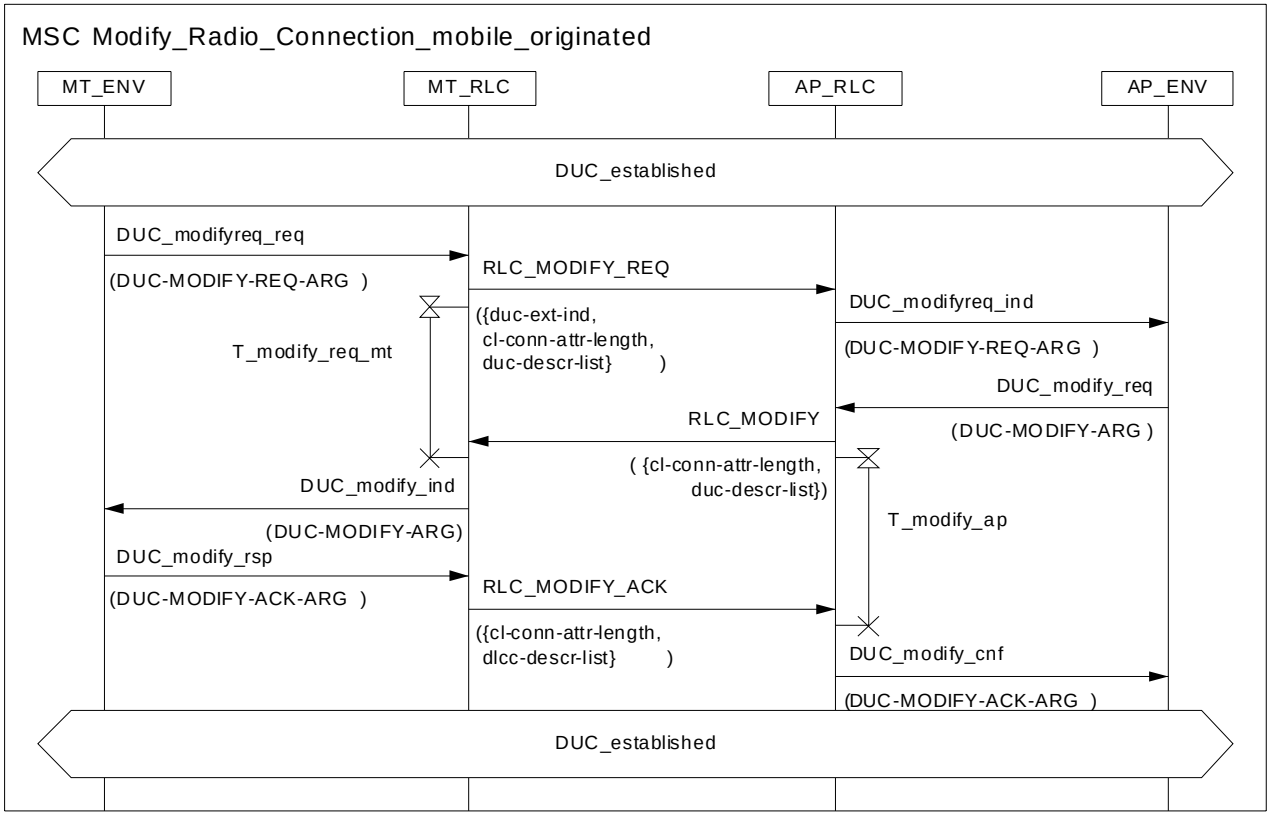


Diagram 64: Mobile originated connection modify procedure

The DUCs shall be reset, see clause 5.3.4.

5.3.4 Unicast DUC Reset

With the reset procedure the ARQ instances and related timers of one or more unicast DUCs shall be reset to their initial state. The DUC characteristics as agreed upon Setup or latest modification shall be maintained. The reset procedure shall be initiated by sending the RLC_RESET message including the DLCC ID(s) of the DUCs. The receiving entity (either MT or AP) shall acknowledge the Reset by responding with RLC_RESET_ACK indicating the corresponding DLCC ID(s).

The ARQ sequence number of the established DUCs shall be reset to zero both in the MT and the AP. At the receiving entity (either MT or AP) RxBoW shall be reset to zero at the reception of the RLC_RESET message. At the sending entity (either MT or AP) TxBoW shall be reset to zero at the reception of the RLC_RESET_ACK message.

A sender shall not send a duplicate RLC_RESET until RLC_RESET_ACK for this particular DUC is received or until the retransmit timer (*T_reset_ap* or *T_reset_mt*) expires.

NOTE: The reset does only affect either uplink or downlink direction.

5.3.4.1 AP Initiated DUC Reset

The AP sender shall either stop transmitting downlink traffic related to the DUCs that are included in the RLC_RESET message after sending the RLC_RESET (including the current frame), or transmit dummy PDUs. The AP should not start transmitting downlink traffic related to the DUCs that are included in the RLC_RESET message before RLC_RESET_ACK is received. The AP shall reset the ARQ instance immediately after receiving RLC_RESET_ACK. The MT shall reset the ARQ instance immediately after receiving RLC_RESET.

At the reception of RLC_RESET, for the DUCs that are included in the RLC_RESET message, the MT shall:

- discard all data in its reception buffer;
- set RxBoW to 0;
- if the receiver is in flow control state, exit the flow control state;
- clear all other ARQ state information, i.e. acknowledgement bitmap;
- retain all DUC characteristics as agreed upon Setup or latest modification.

At the reception of RLC_RESET_ACK, for the DUCs that are included in the RLC_RESET message, the AP shall:

- optionally discard all data in its transmission buffer (the decision is vendor specific);
- set TxBoW to 0;
- if the sender is in flow control state, exit the flow control state;
- clear all other ARQ state information, i.e. rtt awareness for resource requests;
- retain all DUC characteristics as agreed upon Setup or latest modification.

The AP shall not retransmit a duplicate RLC_RESET message, until either the retransmit timer expires or until the RLC_RESET_ACK is received.

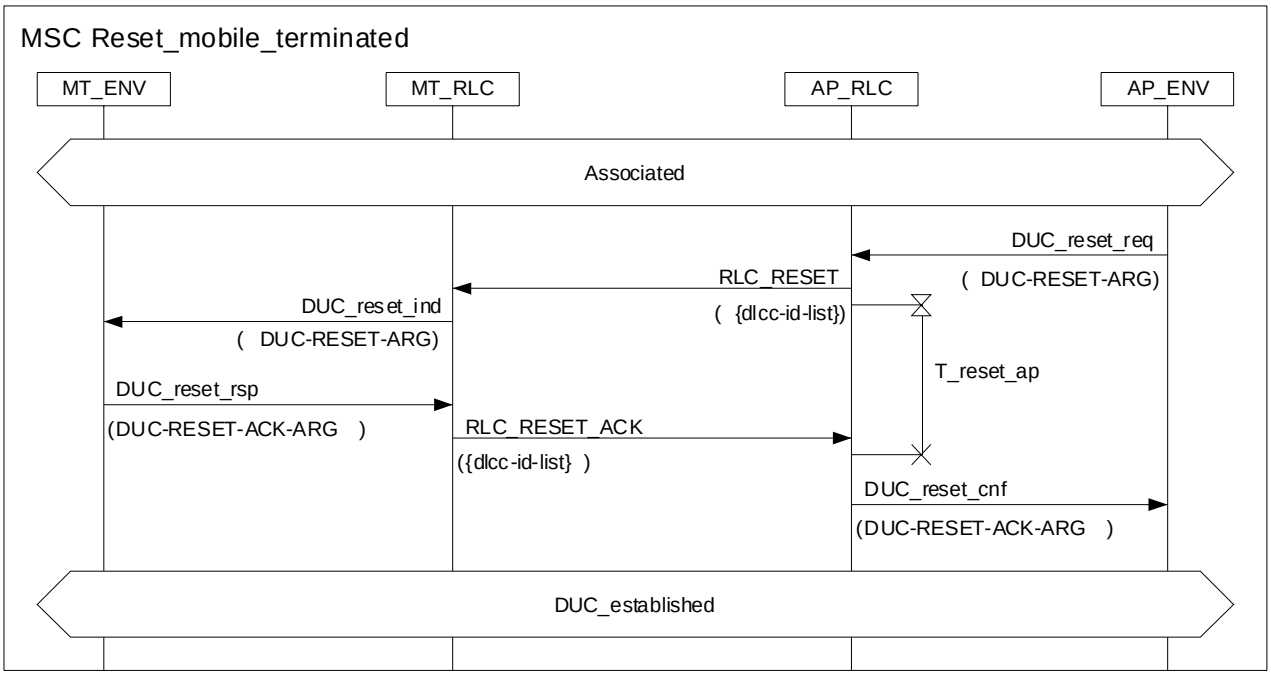


Diagram 65: Mobile terminated reset procedure

Table 86: RLC-RESET

RLC-RESET-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
dlcc-id-list	DLCC-ID-LIST }

Table 87: RLC-RESET-ACK

RLC-RESET-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
dlcc-id-list	DLCC-ID-LIST }

5.3.4.2 MT Initiated DUC reset

The MT sender shall either stop transmitting uplink traffic related to the DUCs that are included in the RLC_RESET message after sending the RLC_RESET, or transmit dummy PDUs.

The MT should not start transmitting uplink traffic related to the DUCs that are included in the RLC_RESET message before RLC_RESET_ACK is received. If the AP allocates uplink capacity for this particular DUC, the MT shall send either the dummy LCH PDU or leave the LCH unused if possible.

The MT shall reset the ARQ instance immediately after receiving RLC_RESET_ACK. The AP shall reset the ARQ instance immediately after receiving RLC_RESET.

At the reception of RLC_RESET, for the DUCs that are included in the RLC_RESET message, the AP shall:

- discard all data in its reception buffer;
- set RxBoW to 0;
- if the receiver is in flow control state, exit the flow control state;
- clear all other ARQ state information, i.e. acknowledgement bitmap;
- retain all DUC characteristics as agreed upon Setup or latest modification.

At the reception of RLC_RESET_ACK, for the DUCs that are included in the RLC_RESET message, the MT shall:

- optionally discard all data in its transmission buffer (the decision is vendor specific);
- set TxBoW to 0;
- if the sender is in flow control state, exit the flow control state;
- clear all other ARQ state information, i.e. rtt awareness for resource requests;
- retain all DUC characteristics as agreed upon Setup or latest modification.

The MT should not transmit a duplicate RLC_RESET message until either the retransmit timer expires or until the RLC_RESET_ACK is received.

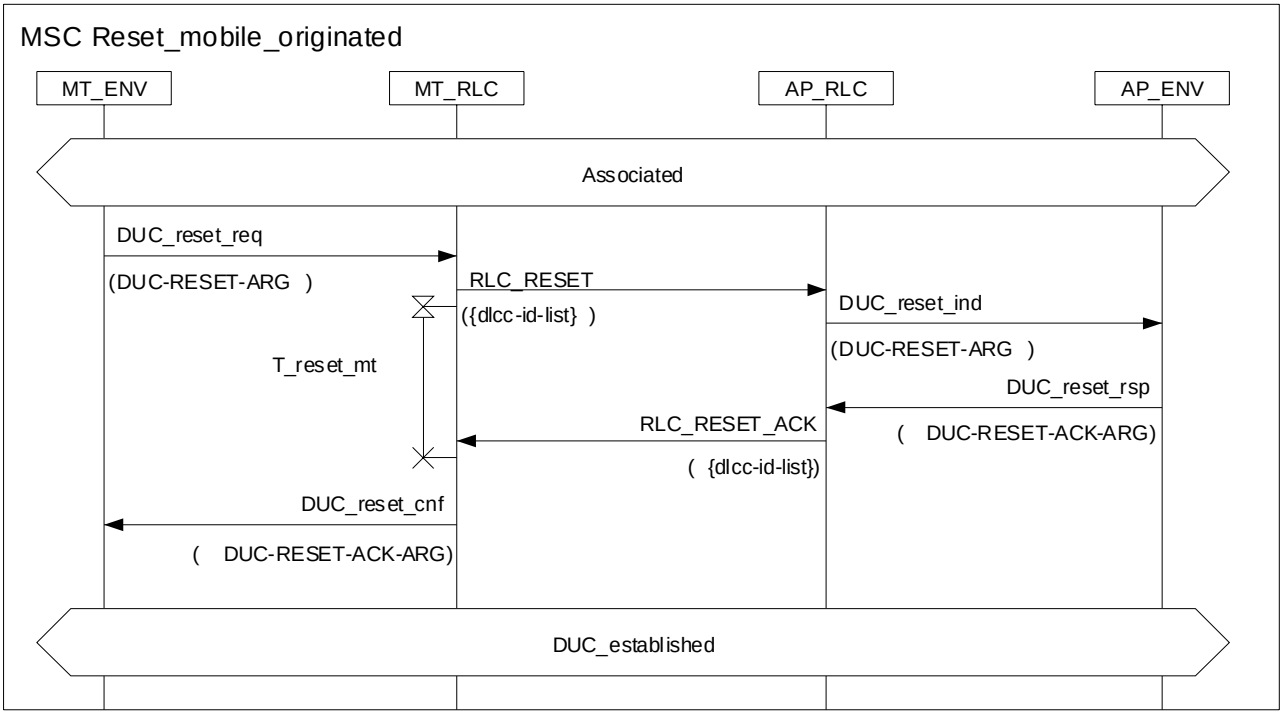


Diagram 66: Mobile originated reset procedure

5.3.5 Multicast DUC

Multicast DUCs are implicitly set up by Group Join during the association procedure, as defined in clause 5.1.4. The MAC IDs and DLCC IDs reserved for multicast connections are defined in [5].

5.3.6 Broadcast DUC

Broadcast DUCs are implicitly set up by Broadcast Join during the association procedure, as defined in clause 5.1.5. The MAC IDs and DLCC IDs reserved for broadcast connections are defined in [5].

5.3.7 Unicast Direct Link DUC Setup (OAP/OMT)

The direct link (DiL), which is used in Direct mode (DM), allows direct connections between two or more MTs. The corresponding control functions are quite similar to those of the centralized mode (AP/CC and MT). But in this case there are one AP/CC and two MTs involved for a DiL unicast connection, as shown as in figure 8.

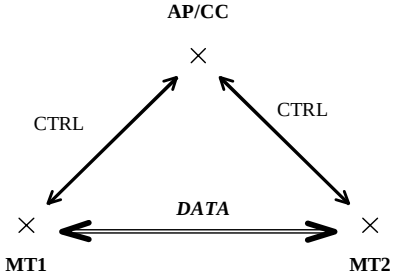


Figure 8: Direct Link connection between an AP/CC and two MTs

The following basic characteristics are valid for the DiL:

- the access point or central controller shall still control DUC establishment and modification;
- a calibration mechanism may be used in the control plane, so that:
  - any MT can know which other MTs it has radio contact with;
  - the AP/CC can know the radio map of the network (which MT is in radio contact with which other MT);
- if no means are available to check the connectivity to the peer DM device before DiL DUC Setup, the DiL DUC Setup shall be performed anyway. In this case the receiving MT shall monitor whether it is able to receive LCHs and/or SCHs of the related DiL DUC. This can be done by monitoring the RG for this DiL DUC. If for a number of consecutive RGs for this DiL DUC no LCH and/or SCH has been received successfully the receiving MT shall release this DiL DUC by performing the MT initiated DiL DUC release (clause 5.3.8.2) procedure. The *release-cause* parameter of the RLC_DM_RELEASE message is set to *low-qos*. The number of consecutive RGs is out of the scope of the present document;
- after receiving the RLC_DM_RELEASE message with *release-cause* equal to *low-qos*, the transmitting device may initiate a DiL DUC relay Setup (clause 5.3.7.3), in order to establish a DiL DUC to the peer device via the AP/CC.

The DUC procedures may request several connections, but all these connections shall belong to the same convergence layer.

It may be possible for the higher layers to establish a connection between two MTs (MT1 and MT2) even if there is no connectivity. In that case, the AP/CC shall act as a Relay by establishing two DiL connections between MT1 and AP/CC and between AP/CC and MT2.

The DUC descriptor field contains one direction descriptor for unidirectional or symmetric duplex connections and two direction descriptors for asymmetric duplex connections. These direction descriptors are called *forward* and *backward descriptor*. For unidirectional connection, *duc-direction* field in DUC descriptor shall be set to simplex *forward* for the sender and simplex *backward* for the receiver. If it is set to simplex forward, then only the forward descriptor shall be present, and if it is set to simplex backward, then only the backward descriptor shall be present. For symmetric duplex connections the *duc-direction* field in DUC descriptor shall be set to symmetric duplex, then the forward and backward descriptor would be identical, therefore only the forward descriptor shall be provided. For asymmetric duplex connections the forward descriptor is related to the sender and the backward descriptor is related to the receiver. The AP/CC is responsible for inverting the contents of the two descriptors for the two MTs involved in the DiL connection.

All DiL DUC connection control messages shall be transmitted over uplink and downlink DCCH. The MAC ID contained in the messages are use to describe the peer MT.

5.3.7.1 AP/CC initiated DM DUC Setup (OMT)

If the Direct Mode is supported, the Relay Setup shall be implemented for the AP/CC.

In case of an AP/CC initiated DiL connection set-up, the RLC shall send a RLC_DM_SETUP message, requesting the MT to establish one or multiple unicast DUCs. In this message the AP/CC shall indicate to the MT the selected DLCC IDs and their connection characteristics (i.e. direction, ARQ parameters, etc.). Referring to the following MSC, the *peer-mac-id* in the DiL DUC Setup messages from/to MT1 shall be set to the MAC ID of the MT2 and the *peer-mac-id* in the DiL DUC Setup messages from/to MT2 shall be set to the MAC ID of the MT1.

When the AP/CC requires to establish additional DUCs in a subsequent Setup procedure, the AP/CC should indicate this by setting the *duc-ext-ind* flag. The AP/CC shall use the forward direction for the MT that is the sender and the backward direction for the receiver MT.

The MT shall respond with RLC_DM_CONNECT message. The connection characteristics shall not be modified by the MT. In order to reject the DiL Setup the MT shall send RLC_DM_RELEASE message and continue with the DiL Release procedure. The AP/CC shall send the RLC_DM_CONNECT_ACK message before continuing with the other MT.

The AP/CC shall run the same procedure for the second MT to negotiate the connection. If the second MT can not support the characteristics, it shall initiate the DiL Release procedure.

The AP/CC can also act as one of the MTs, in this case the messages between this MT and the AP/CC are exchanged internally in the AP/CC.

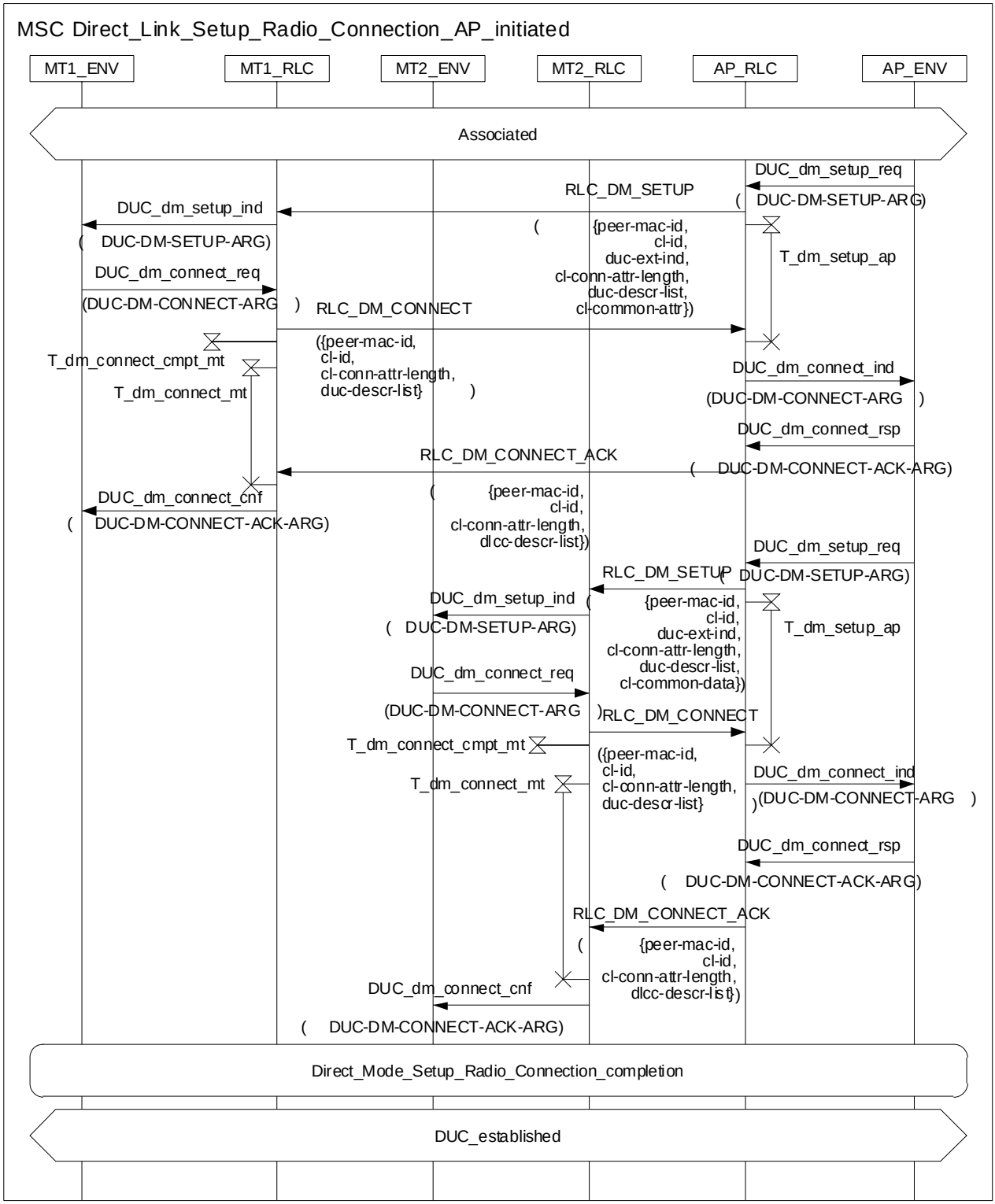


Diagram 67: Direct Link Setup procedure – AP/CC initiated

The message RLC_DM_CONNECT_COMPLETE shall be used to synchronize the two MTs after the connection phase. These messages may be sent in parallel to both MTs.

If the MT does not receive the RLC_DM_CONNECT_COMPLETE message before the timer runs out, it shall not establish the connections and shall initiated the DiL Release procedure. After receiving this message, the MT shall respond with RLC_DM_CONNECT_COMPLETE_ACK message.

In case of the connections use ARQ, the ARQ sequence number of the established DUCs shall be reset to zero in both MTs. TxBoW and RxBoW of the corresponding DUCs shall be reset to zero in both MTs.

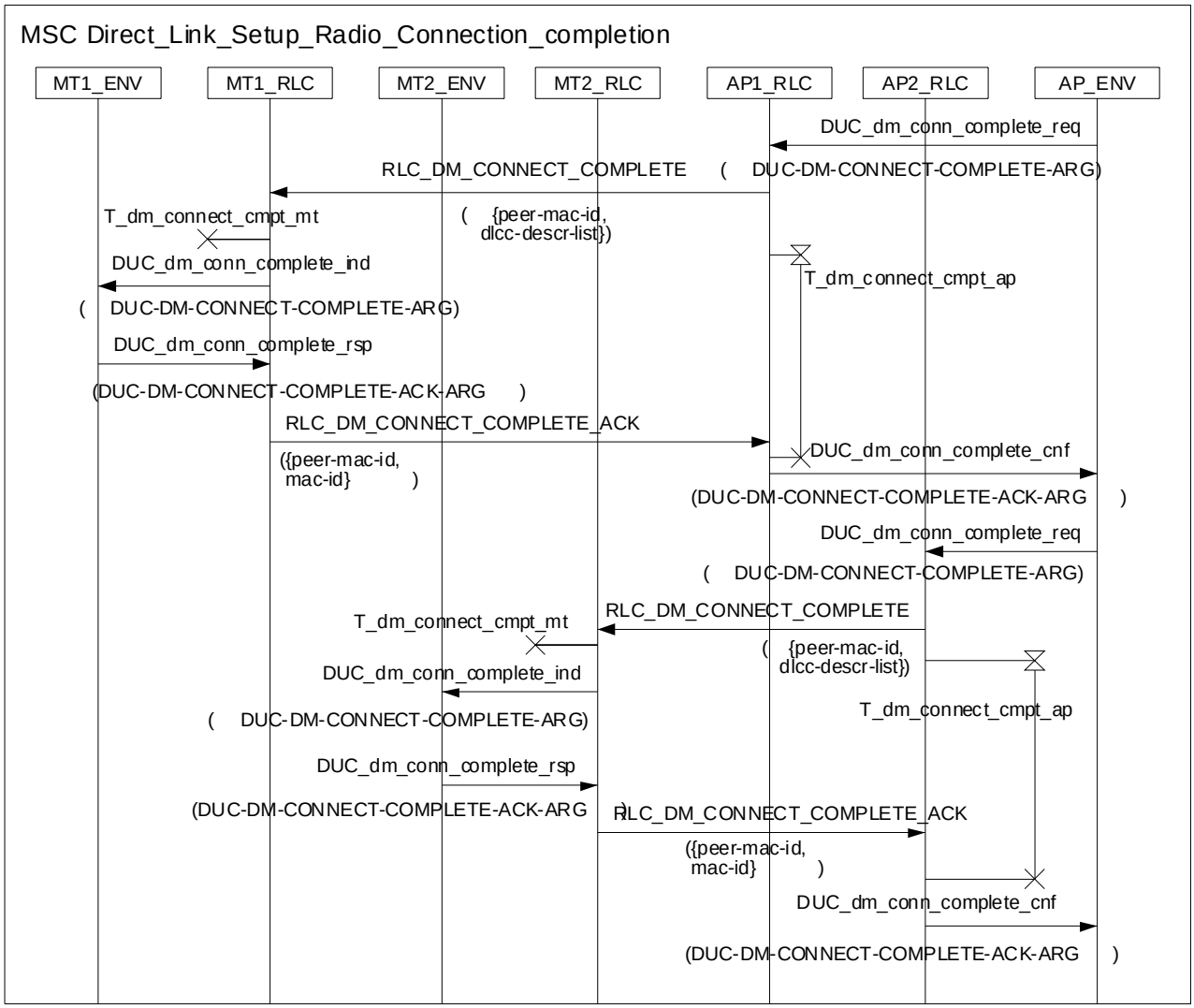


Diagram 68: Completion of direct link Setup procedure

Table 88: RLC-DM-SETUP

RLC-DM-SETUP-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-id	CL-ID
duc-ext-ind	DUC-EXT-IND
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST
cl-common-attr	CL-COMMON-ATTR }

Table 89: RLC-DM-CONNECT

RLC-DM-CONNECT-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-id	CL-ID
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 90: RLC-DM-CONNECT-ACK

RLC-DM-CONNECT-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-id	CL-ID
cl-conn-attr-length	INTEGER(0..31)
dlcc-descr-list	DLCC-DESCR-LIST }

Table 91: RLC-DM-CONNECT-COMPLETE

RLC-DM-CONNECT-COMPLETE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
dlcc-id-list	DLCC-ID-LIST }

Table 92: RLC-DM-CONNECT-COMPLETE-ACK

RLC-DM-CONNECT-COMPLETE-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
peer-mac-id	MAC-ID
mac-id	MAC-ID }

NOTE: The AP/CC may be one of the MTs, i.e. MT1 or MT2. In this case, the messages shall be sent and received internally in the AP/CC.

5.3.7.2 MT initiated DM DUC Setup (OMT)

If the Direct Mode is supported, the Relay Modify shall be implemented for the AP/CC.

A MT can also initiate one or several direct link connections by sending the RLC_DM_SETUP message. The MT shall make a proposal for the characteristics of the DLC user connections. In that case the DLCC IDs shall be set to zero. When the AP/CC receives this RLC_DM_SETUP message from MT, the AP/CC shall respond with RLC_DM_CONNECT with DLCC IDs selected by the AP/CC. The AP/CC may modify the connection characteristics. If the new characteristics are not acceptable for the initiating MT, it should reject the Setup by initiating a DiL Release procedure.

If the MT has accepted the characteristics in the RLC_DM_CONNECT, the AP/CC shall connect the other MT with the same procedure as describe in the AP/CC initiated RLC_DM_ SETUP.

NOTE: Messages RLC_DM_SETUP, RLC_DM_CONNECT and RLC_DM_CONNECT_ACK are used either in uplink or in downlink.

Once parameters negotiated with the two MTs, the AP/CC shall send the RLC_DM_CONNECT_COMPLETE message to both MTs to indicate which of the connections shall be established. Then after receiving these messages, both MTs shall respond with RLC_DM_CONNECT_COMPLETE_ACK message.

The ARQ sequence number of the established DUCs shall be reset to zero in both MTs. TxBoW and RxBoW of the corresponding DUCs shall be reset to zero in both MTs.

Referring to the following MSC, the peer MAC ID in the DiL Duc Setup messages from/to MT1 shall be set to the MAC ID of the MT2 and the peer-mac-id in the DiL Duc Setup messages from/to MT2 shall be set to the MAC ID of the MT1. If the peer-mac-id is not known to RLC of the initiating MT, then it shall be set to its own value. In that case the peer MT is identified in convergence layer container and the AP/CC shall do the mapping.

The AP/CC can also act as one of the MTs, in this case the messages between this MT and the AP/CC are exchanged internally in the AP/CC.

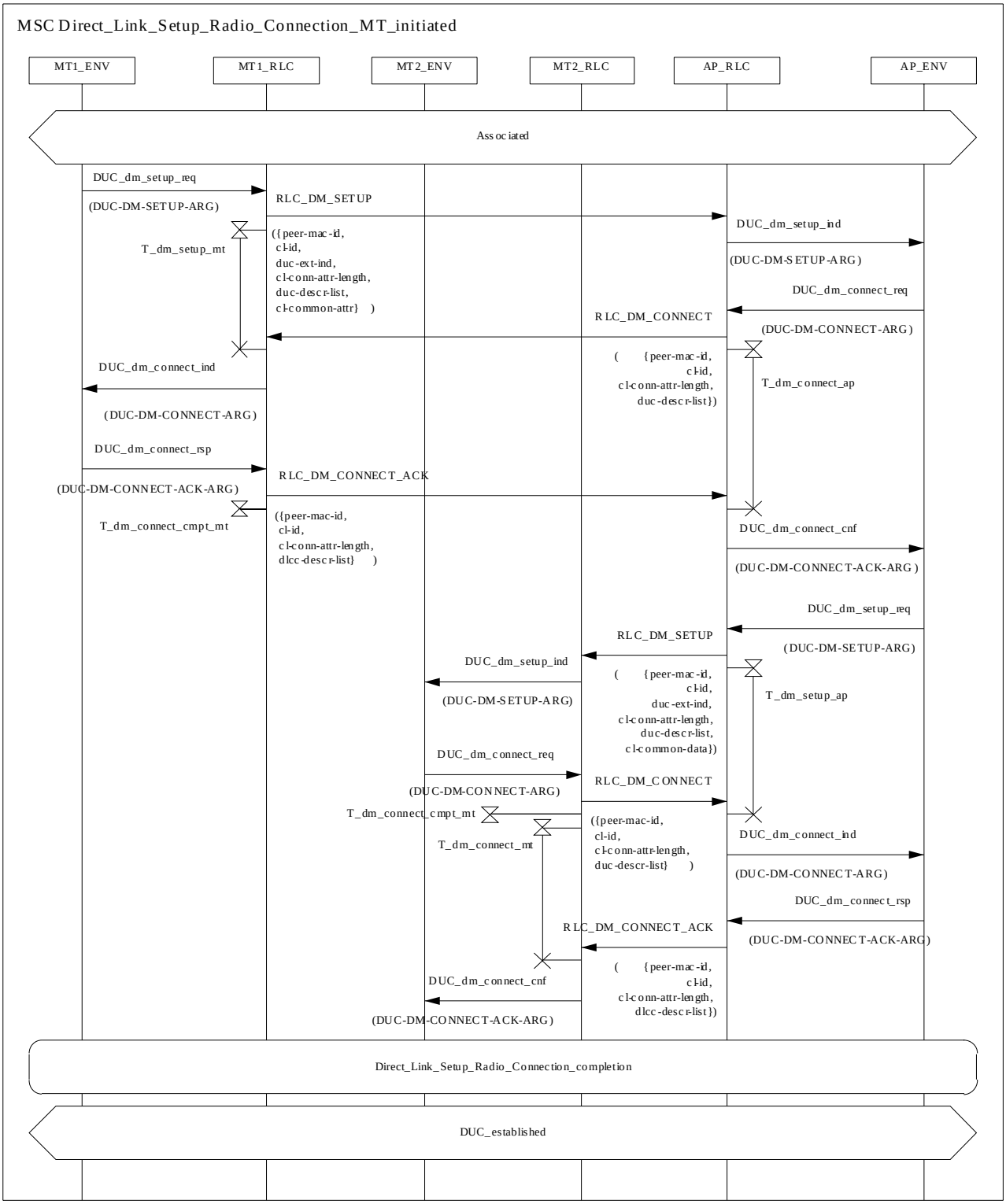


Diagram 69: Direct Link Setup procedure - MT initiated

5.3.7.3 DM DUC Relay Setup (OMT)

If the Direct Mode is supported, the Relay Release shall be implemented for the AP/CC.

Even if there is no connectivity between two MTs some facilities might exist for the upper layers to establish a connection. In this case, the AP/CC shall relay the data transmission by opening two DiL DUCs for both MTs. The RLC_DM_RELAY_SETUP shall be sent by the MT to the AP/CC. The message contains the connection characteristics that shall be used by the AP/CC to initiate two DiL Setup procedures. RLC_DM_RELAY_SETUP message shall only be initiated by a MT.

The AP/CC shall open two direct link DUCs, these connections shall be opened between the MTs and the AP/CC, as shown in this figure. In the case MT1 has sent the RLC_DM_RELAY_SETUP message, the AP/CC shall respond with the RLC_DM_RELAY_SETUP_ACK to MT1. This message shall carry the DLCC_IDs and characteristics of the DUCs opened between MT1 and the AP/CC. If the negotiated characteristics are not acceptable for the MT, it shall release the connection using the RLC_DM_RELAY_RELEASE message.

Referring to the following MSC, the *peer-mac-id* of the Relay messages from and to MT1 shall be set to the MAC ID of the MT2.

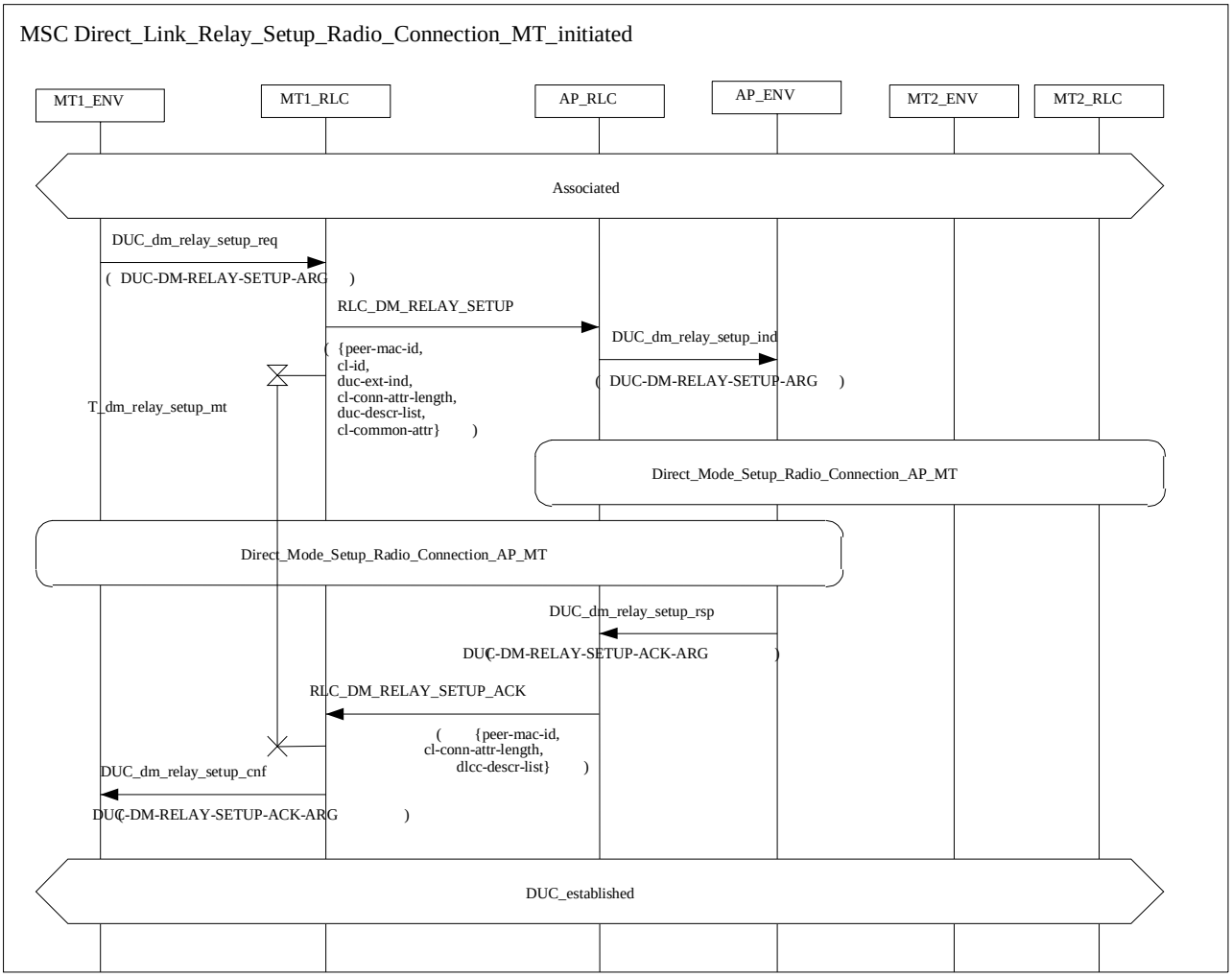


Diagram 70: Relay Setup - MT originated

Table 93: RLC-DM-RELAY-SETUP

RLC-DM-RELAY-SETUP-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-id	CL-ID
duc-ext-ind	DUC-EXT-IND
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST
cl-common-attr	CL-COMMON-ATTR }

Table 94: RLC-DM-RELAY-SETUP-ACK

RLC-DM-RELAY-SETUP-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-conn-attr-length	INTEGER(0..31)
dlcc-descr-list	DLCC-DESCR-LIST }

The connections between the AP/CC and both MTs shall be established by the AP/CC initiated DiL DUC Setup procedure, where AP/CC acts as one of the MT.

The following MSC shows this special case.

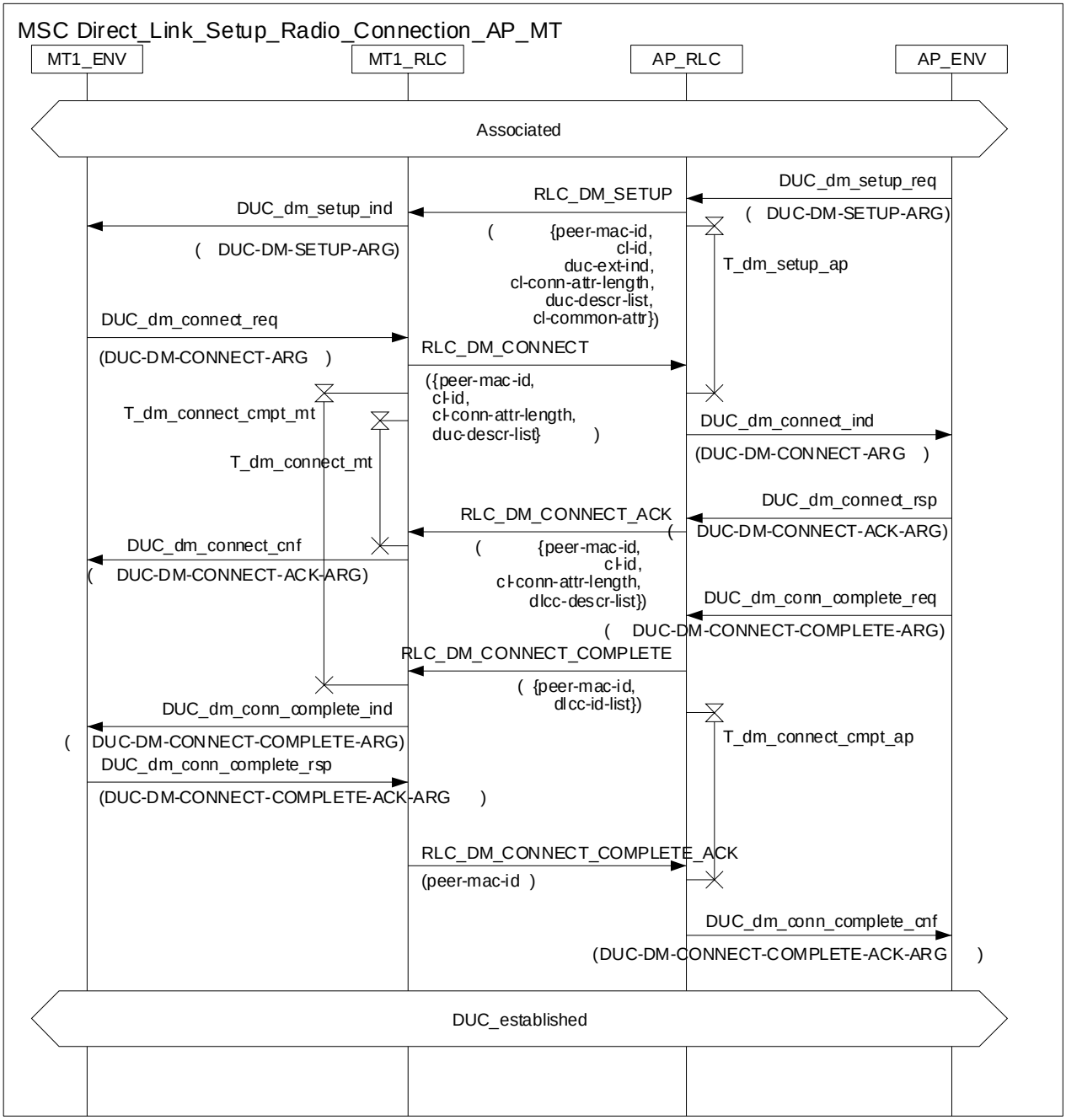


Diagram 71: Direct Link Setup connection procedure between AP/CC and MT, AP/CC initiated

NOTE: The AP/CC can use AP/CC initiated DiL DUC set up procedure to set up a connection between MT1 and AP/CC and another one between AP/CC and MT2, then realize the relay between the MT1 and MT2.

5.3.8 Unicast Direct Link DUC Release

5.3.8.1 AP/CC initiated DM DUC Release

This procedure is used to release one or more unicast DiL connections in direct mode.

The AP/CC shall first indicate to a MT to release the indicated DUCs by a RLC_DM_RELEASE message. The MT shall send the RLC_DM_RELEASE_ACK to indicate that the relevant DLCC IDs are released. The AP/CC shall not schedule any data to the particular MTs for the particular DLCC IDs after sending the first RLC_DM_RELEASE message.

Referring to the following MSC, the peer-mac-id in the DiL DUC Setup messages from/to MT1 shall be set to the MAC ID of the MT2 and the peer-mac-id in the DiL DUC Setup messages from/to MT2 shall be set to the MAC ID of the MT1.

The AP/CC shall do the same procedure with the other MT to complete the release of the direct link connections.

The AP/CC can also act as one of the MTs, in that case the messages between this MT and the AP/CC are exchanged internally in the AP/CC.

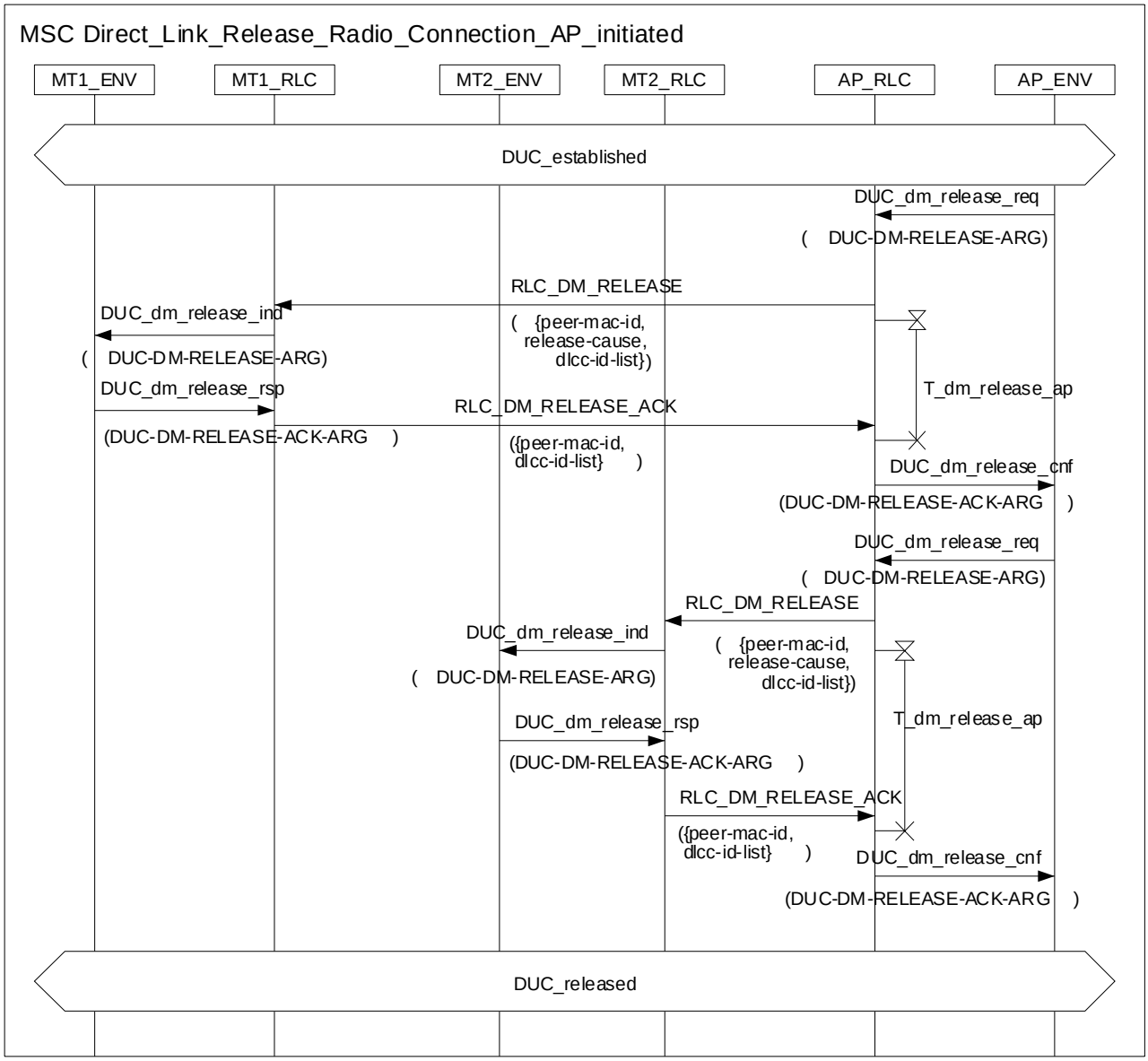


Diagram 72: Direct Link Release connection procedure – AP/CC initiated

Table 95: RLC-DM-RELEASE

RLC-DM-RELEASE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
release-cause	RELEASE-CAUSE
dlcc-id-list	DLCC-ID-LIST }

Table 96: RLC-DM-RELEASE-ACK

RLC-DM-RELEASE-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
dlcc-id-list	DLCC-ID-LIST }

5.3.8.2 MT initiated DM DUC Release

A MT may release one or few DiL connections by sending the RLC_DM_RELEASE message to the AP/CC. When the AP/CC receives this message, it shall release the connections with the other MT before sending back the acknowledgement. In case that the second MT shall send RLC_DM_RELEASE_ACK as a positive acknowledgement, the AP/CC shall send the RLC_DM_RELEASE_ACK to the MT that initiated the Release procedure.

The AP/CC shall not schedule any data to the particular MTs for particular DLCC IDs after receiving the RLC_DM_RELEASE message.

The AP/CC may also act as one of the MTs, in this case the messages between this MT and the AP/CC are exchanged internally in the AP/CC.

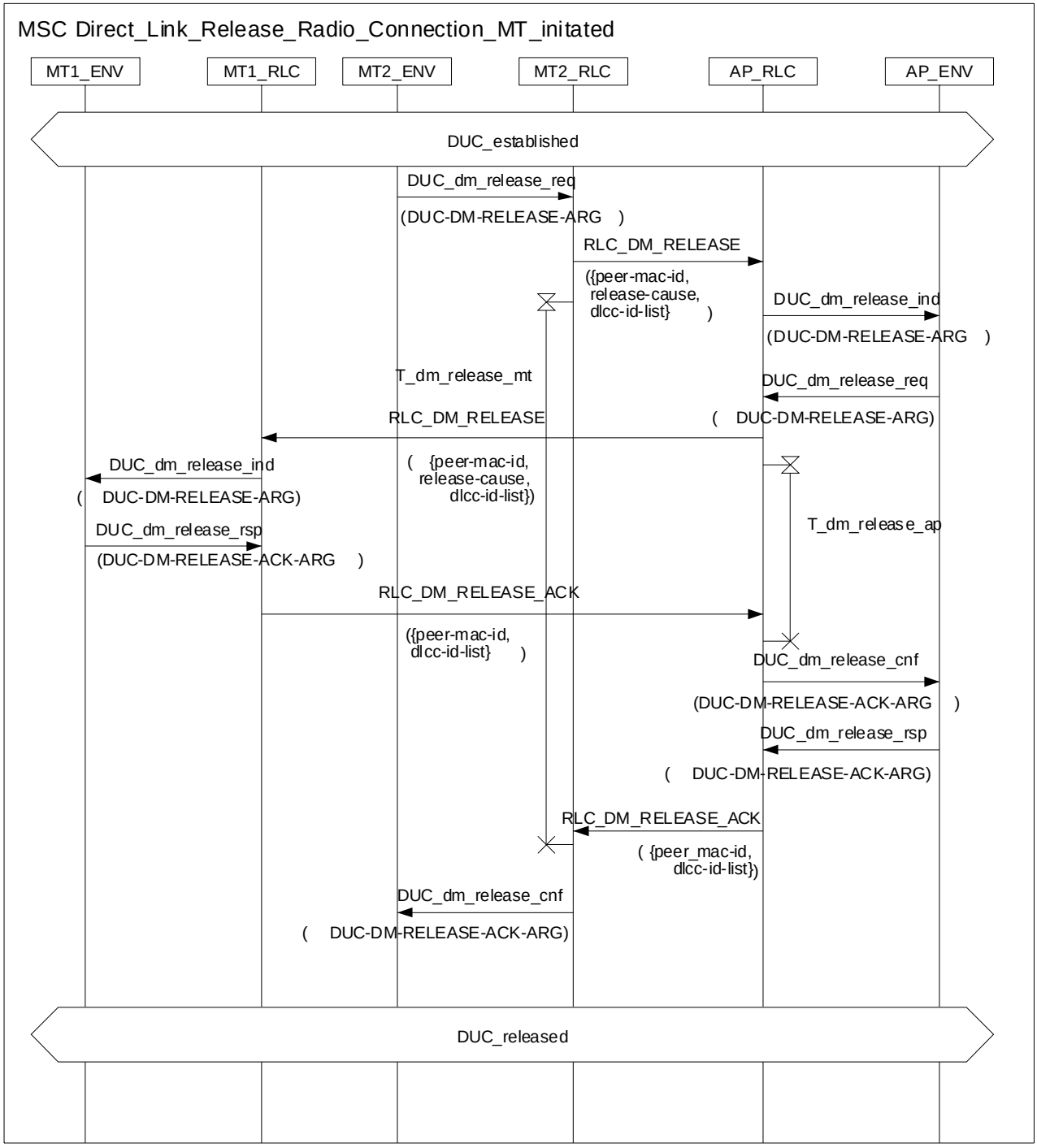


Diagram 73: Direct Link Release connection procedure - MT initiated

5.3.8.3 DM DUC Relay Release

Based on the same principle than in the Relay Setup, the MT shall use the RLC_DM_RELAY_RELEASE message to send the DLCC IDs to the AP/CC for the Release. The AP/CC shall release the two direct link connections and sent back the RLC_DM_RELAY_RELEASE_ACK message as an acknowledgement to the MT. RLC_RELAY_RELEASE message shall only be initiated by a MT.

The AP/CC shall not schedule any data to the particular MTs for these particular DLCC IDs after receiving the RLC_DM_RELAY_RELEASE message.

Referring to the following MSC, the peer-mac-id of the Relay messages from and to MT1 shall be set to the MAC ID of the MT2.

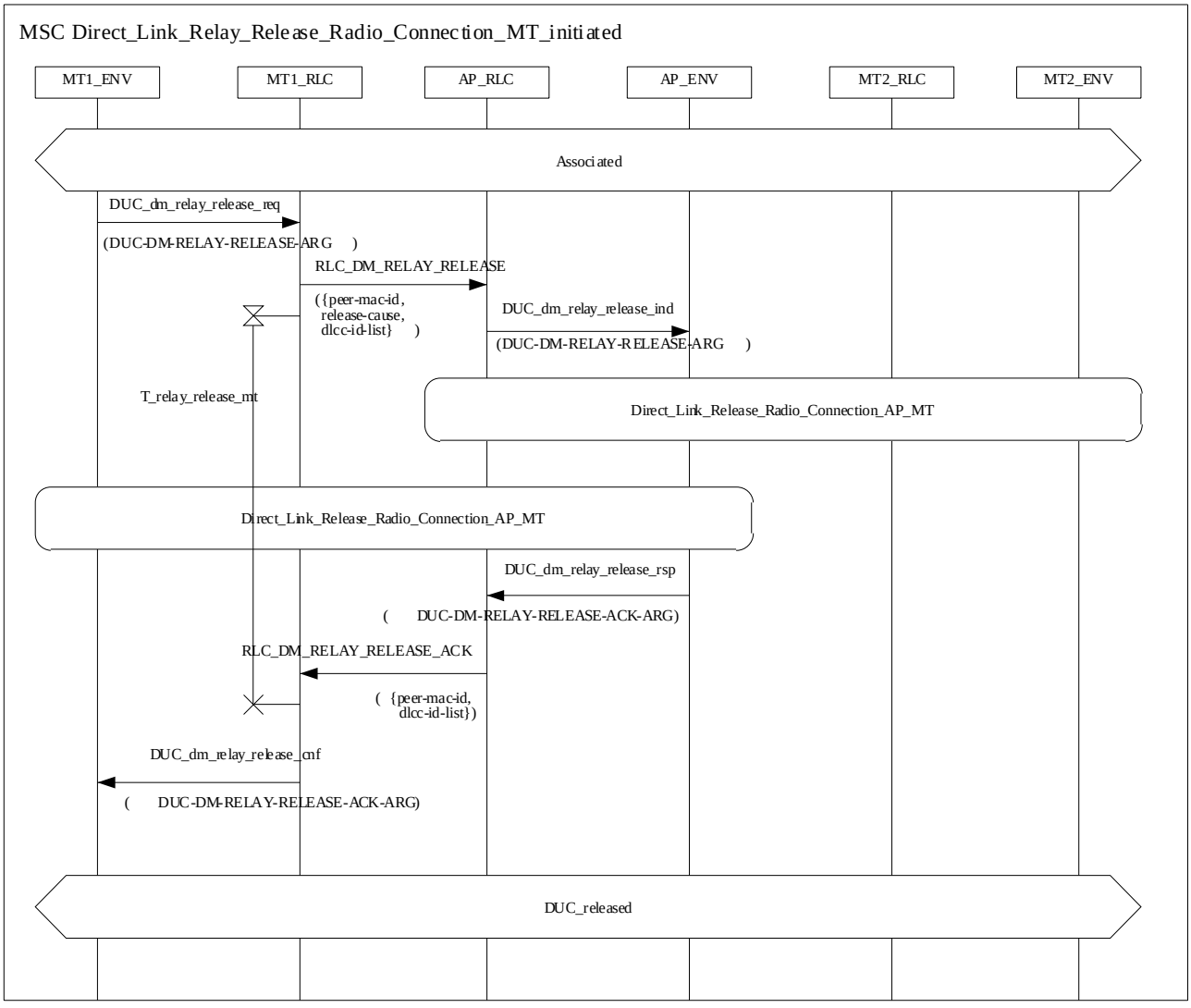


Diagram 74: Relay Release - MT originated

Table 97: RLC-DM-RELAY-RELEASE

RLC-DM-RELAY-RELEASE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
release-cause	RELEASE-CAUSE
dlcc-id-list	DLCC-ID-LIST }

Table 98: RLC-DM-RELAY-RELEASE-ACK

RLC-DM-RELAY-RELEASE-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
dlcc-id-list	DLCC-ID-LIST }

The connections between the AP/CC and both MTs shall be released by AP/CC initiated DiL DUC Release, where AP/CC acts as one of the MT.

The following MSC shows this special case.

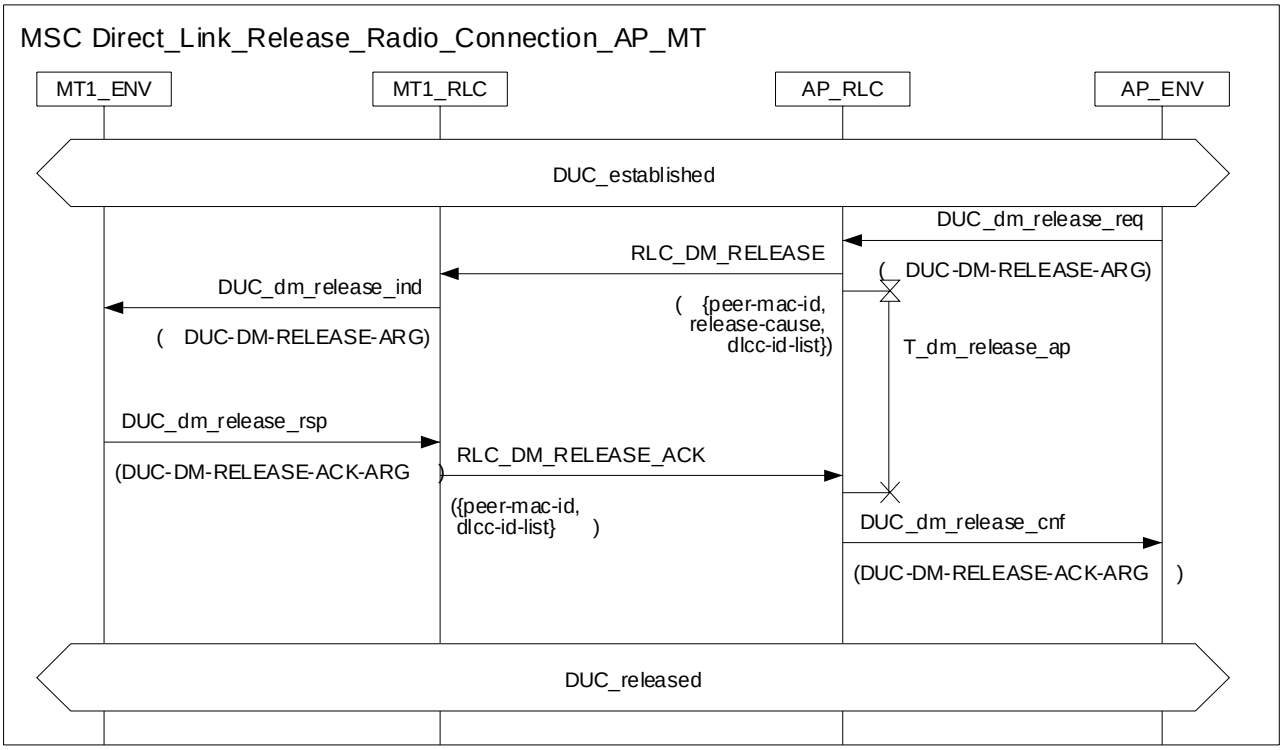


Diagram 75: Direct Link Release connection AP/CC – MT, AP/CC initiated

NOTE: The AP/CC can use AP/CC initiated DiL DUC Release procedure to release connections between MT1 and AP/CC and between AP/CC and MT2.

5.3.9 Unicast Direct Link DUC Modify

5.3.9.1 AP/CC initiated DM DUC Modify

This procedure shall be used to modify one or multiple unicast DiL connections.

The AP/CC shall send a RLC_DM_MODIFY_REQ message to one of the two MTs with the new parameters for the selected connections. The MT shall accept the modification by sending the RLC_DM_MODIFY, otherwise, if it does not accept the modification, it shall start a DiL Release procedure. The MT shall not modify parameters. The AP/CC shall respond to the MT with the RLC_DM_MODIFY_ACK as an acknowledgement to indicate, which DLCC IDs are concerned by the change. Referring to the following MSC, the *peer_mac_id* in the DiL DUC Modify messages from and to MT1 shall be set to the MAC ID of the MT2 and the *peer_mac_id* in the DiL DUC Modify messages from and to MT2 shall be set to the MAC ID of the MT1.

The AP/CC can also act as one of the MTs, in this case the messages between this MT and the AP/CC are exchanged internally in the AP/CC.

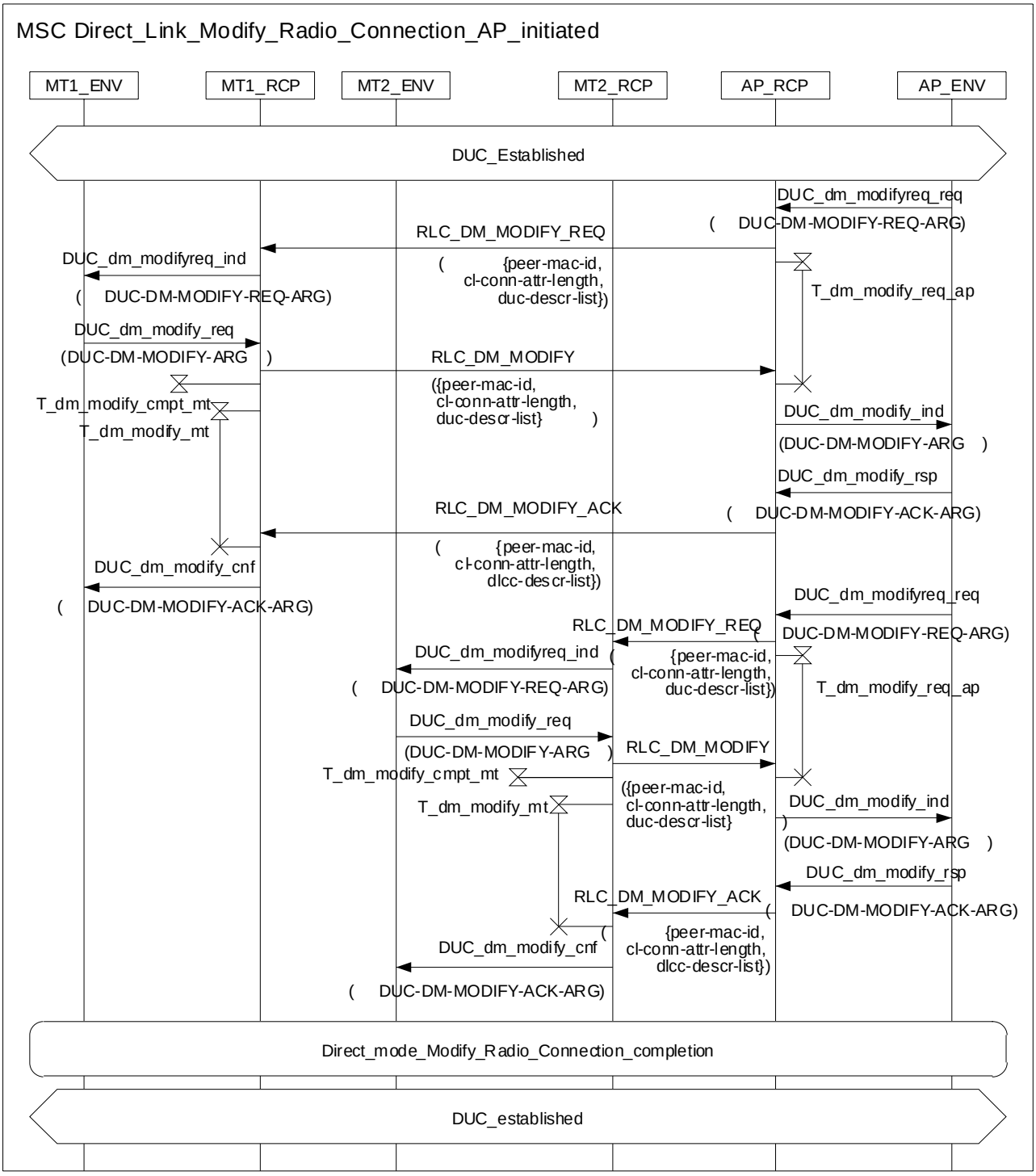


Diagram 76: Direct Link Modify procedure – AP/CC initiated

Table 99: RLC-DM-MODIFY-REQ

RLC-DM-MODIFY-REQ-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 100: RLC-DM-MODIFY

RLC-DM-MODIFY-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 101: RLC-DM-MODIFY-ACK

RLC-DM-MODIFY-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-conn-attr-length	INTEGER(0..31)
dlcc-descr-list	DLCC-DESCR-LIST }

The AP/CC shall use the same procedure to modify parameters of the other MT. For a secure connection, the modifications shall be the same for the two MTs. The AP/CC shall send the RLC_DM_MODIFY_COMPLETE message to both MTs to indicate that both MTs are able to perform the modifications. The message shall be used to synchronize the two MTs after the modification phase. These messages should be sent in parallel to both MTs.

The MTs shall perform modifications on DUCs after having received the RLC_DM_MODIFY_COMPLETE message from the AP/CC, and they shall send the RLC_DM_MODIFY_COMPLETE_ACK message.

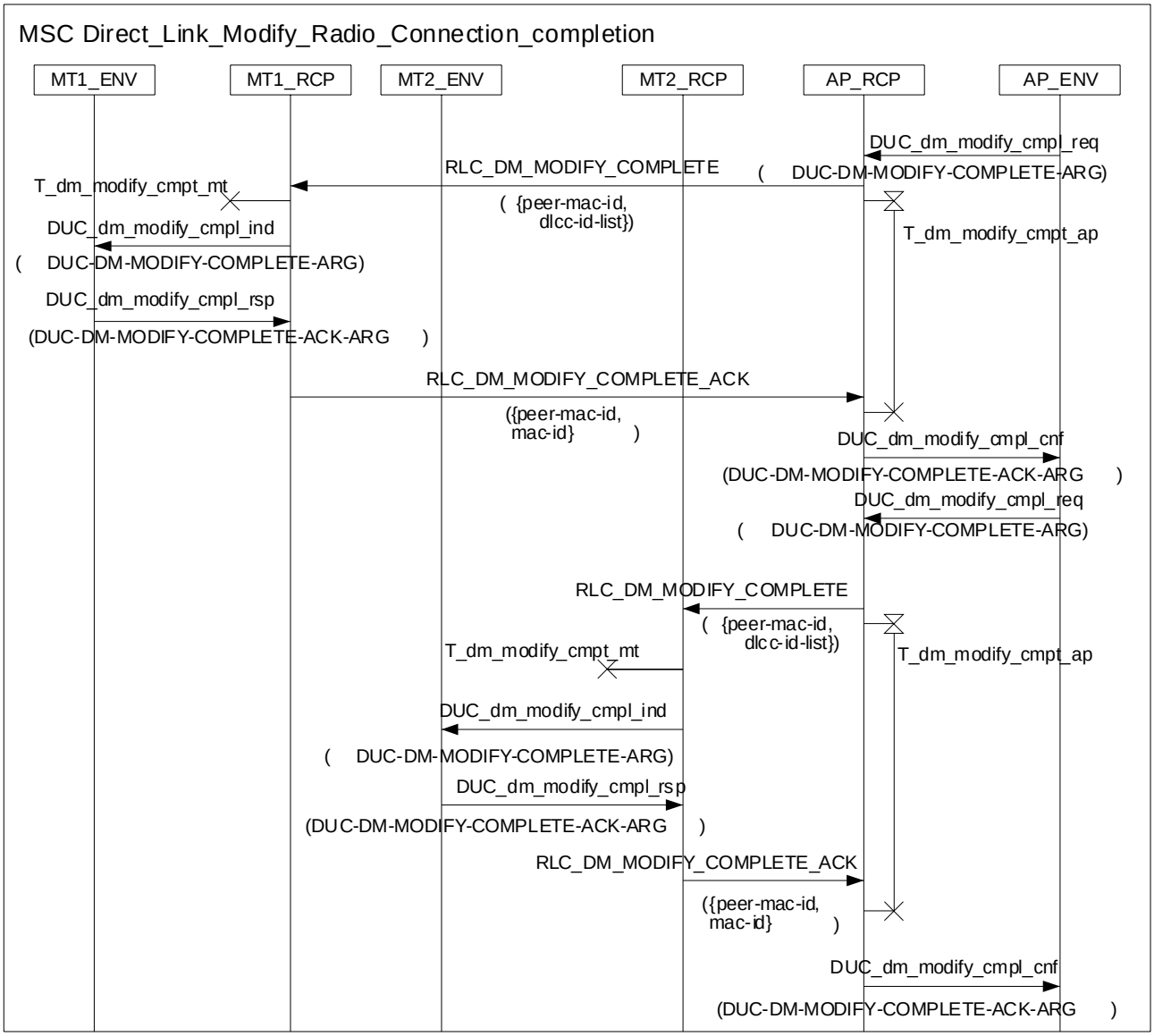


Diagram 77: Direct Link Modify connection completion procedure

Table 102: RLC-DM-MODIFY-COMPLETE

RLC-DM-MODIFY-COMPLETE-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
dlcc-id-list	DLCC-ID-LIST }

Table 103: RLC-DM-MODIFY-COMPLETE-ACK

RLC-DM-MODIFY-COMPLETE-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
peer-mac-id	MAC-ID
mac-id	MAC-ID }

NOTE: The AP/CC may be one of the MTs, i.e. MT1 or MT2. In this case, the messages shall be sent and received internally in the AP/CC.

5.3.9.2 MT initiated DM DUC Modify

A MT shall initiate a DUC Modify by sending the RLC_DM_MODIFY_REQ message to the AP/CC. This message shall contain the new parameters proposed for the modification of one or more connections. The AP/CC shall use the RLC_DM_MODIFY message to modify parameters if it is not able to accept those that are proposed. If the MT accepts the parameters in the RLC_DM_MODIFY message, the MT shall accept it by sending the RLC_DM_MODIFY_ACK message with accepted DLCC IDs. Both MT and AP/CC may reject modifications required by initiating the DiL Release procedure.

The AP/CC shall use the same procedure to modify the parameters of the other MT involved in this direct link.

Referring to the following MSC, the peer MAC ID in the DiL Duc Setup messages from/to MT1 shall be set to the MAC ID of the MT2 and the peer MAC ID in the DiL Duc Setup messages from/to MT2 shall be set to the MAC ID of the MT1. If the peer MAC ID is not known to RLC of the initiating MT, then it shall be set to its own value. In that case the peer MT is identified in convergence layer container and the AP/CC shall do the mapping.

The AP/CC can also act as one of the MTs, in this case the messages between this MT and the AP/CC are exchanged internally in the AP/CC.

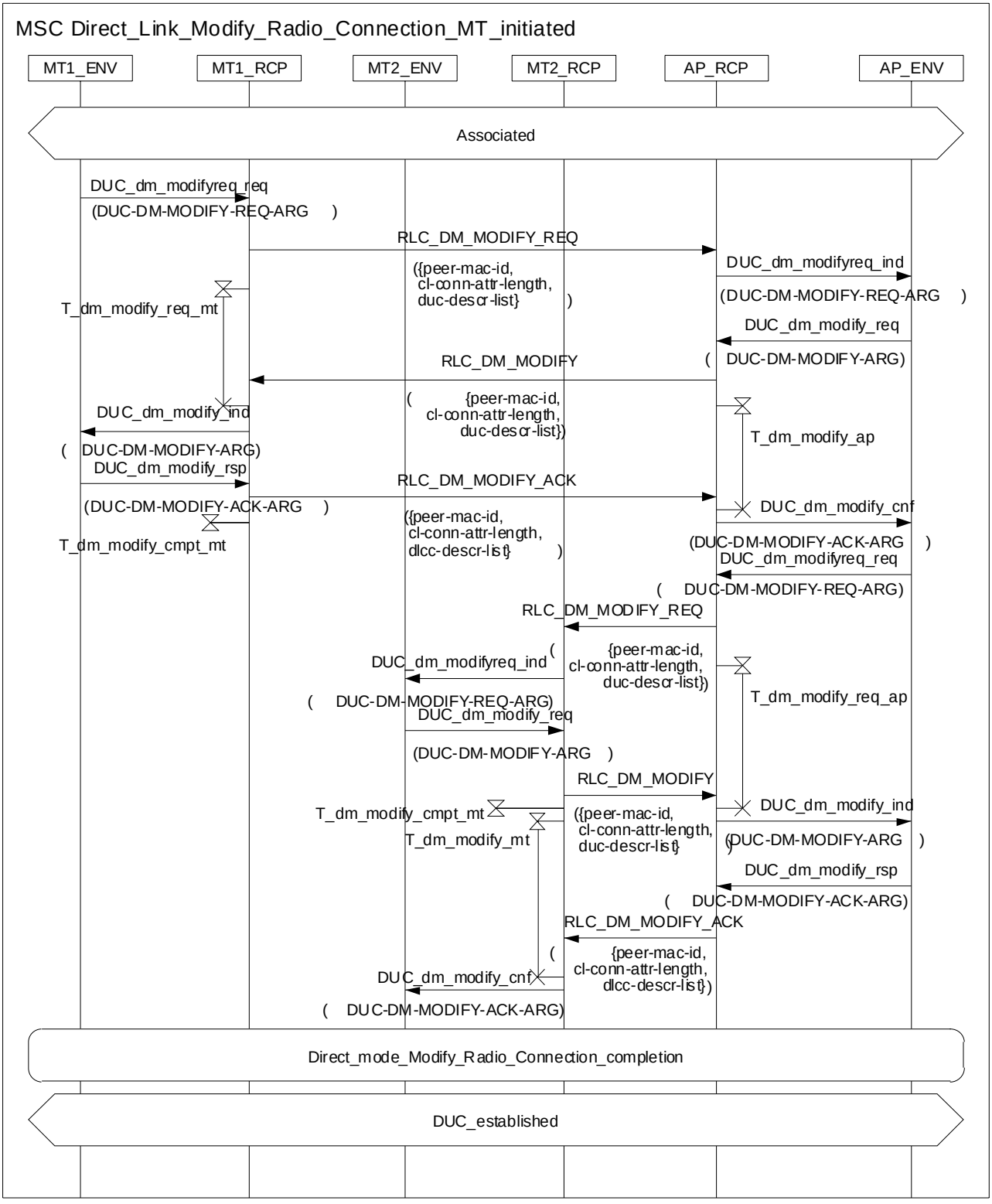


Diagram 78: Direct Link Modify procedure - MT initiated

After receiving RLC_DM_MODIFY_ACK the AP/CC shall send the RLC_DM_MODIFY_COMPLETE message to synchronize both MTs and also to indicate, which connections will change their characteristics. If both MTs do not accept the same parameters, the AP/CC shall release the connections. Then after receiving these messages, both MTs shall respond with RLC_DM_CONNECT_COMPLETE_ACK message.

5.3.9.3 DM DUC Relay Modify

The principle of this procedure is the same than in the Relay Setup, the MT will ask the AP/CC to modify the two direct link connections previously opened between AP/CC and MT1, and between MT2 and AP/CC. The DLCC IDs used shall be those of the connections between the AP/CC and the MT that sent the RLC_DM_RELAY_MODIFY message. The AP/CC shall respond with the RLC_DM_RELAY_MODIFY_ACK containing the modified DLCC_IDs as an acknowledgement. This list shall be empty, if the AP/CC or the other MT have not accepted the requested characteristics.

The RLC_DM_RELAY_MODIFY message shall only be initiated by an MT. Referring to the following MSC, the peer-mac-id of the Relay messages from and to MT1 shall be set to the MAC ID of the MT2.

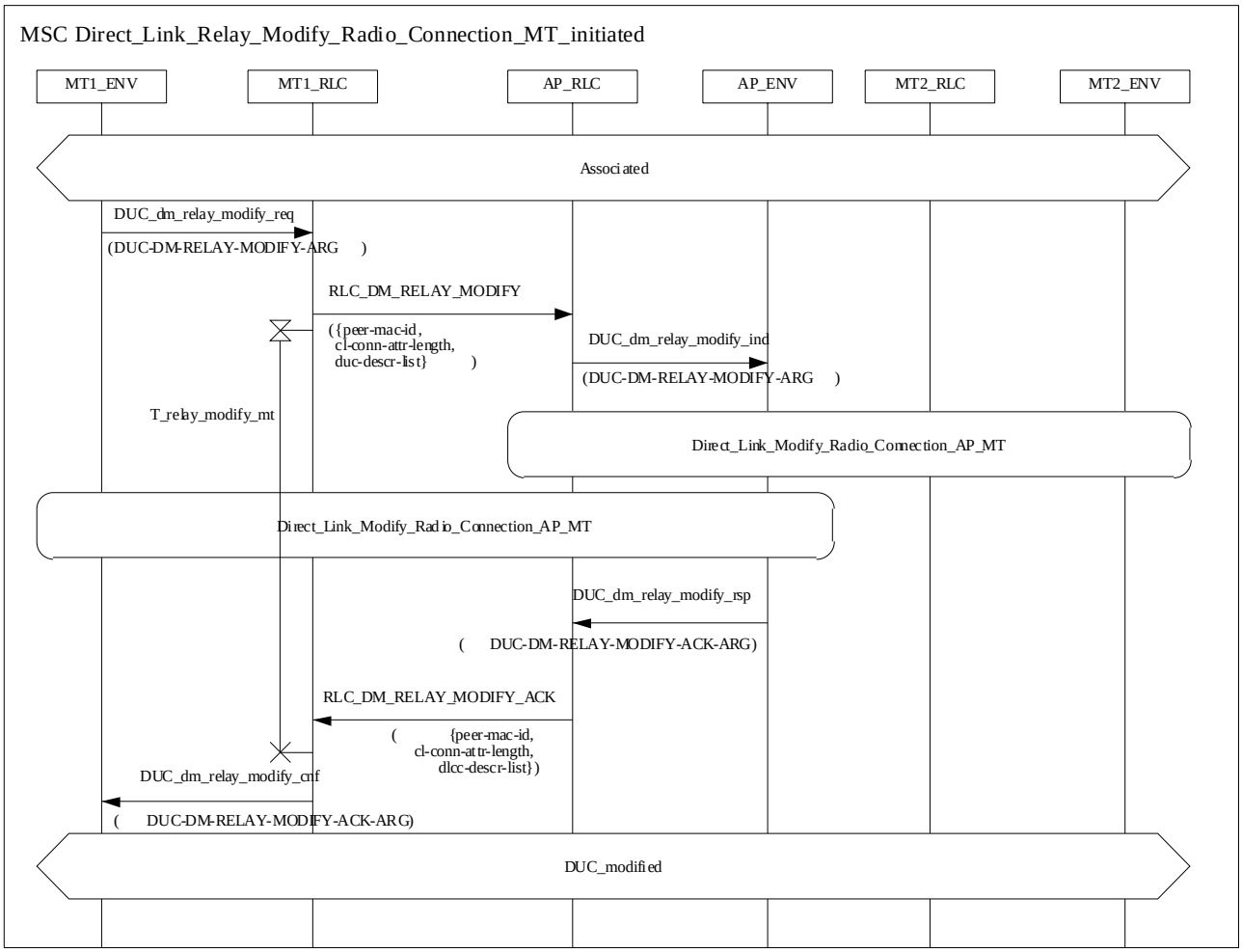


Diagram 79: Direct Link Relay Modify procedure - MT initiated

Table 104: RLC-DM-RELAY-MODIFY

RLC-DM-RELAY-MODIFY-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-conn-attr-length	INTEGER(0..31)
duc-descr-list	DUC-DESCR-LIST }

Table 105: RLC-DM-RELAY-MODIFY-ACK

RLC-DM-RELAY-MODIFY-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
cl-conn-attr-length	INTEGER(0..31)
dlcc-descr-list	DLCC-DESCR-LIST }

The connections between the AP/CC and both MTs shall be modified by AP/CC initiated DiL DUC Modify procedure, where AP/CC acts as one of the MT.

The following MSC shows this special case.

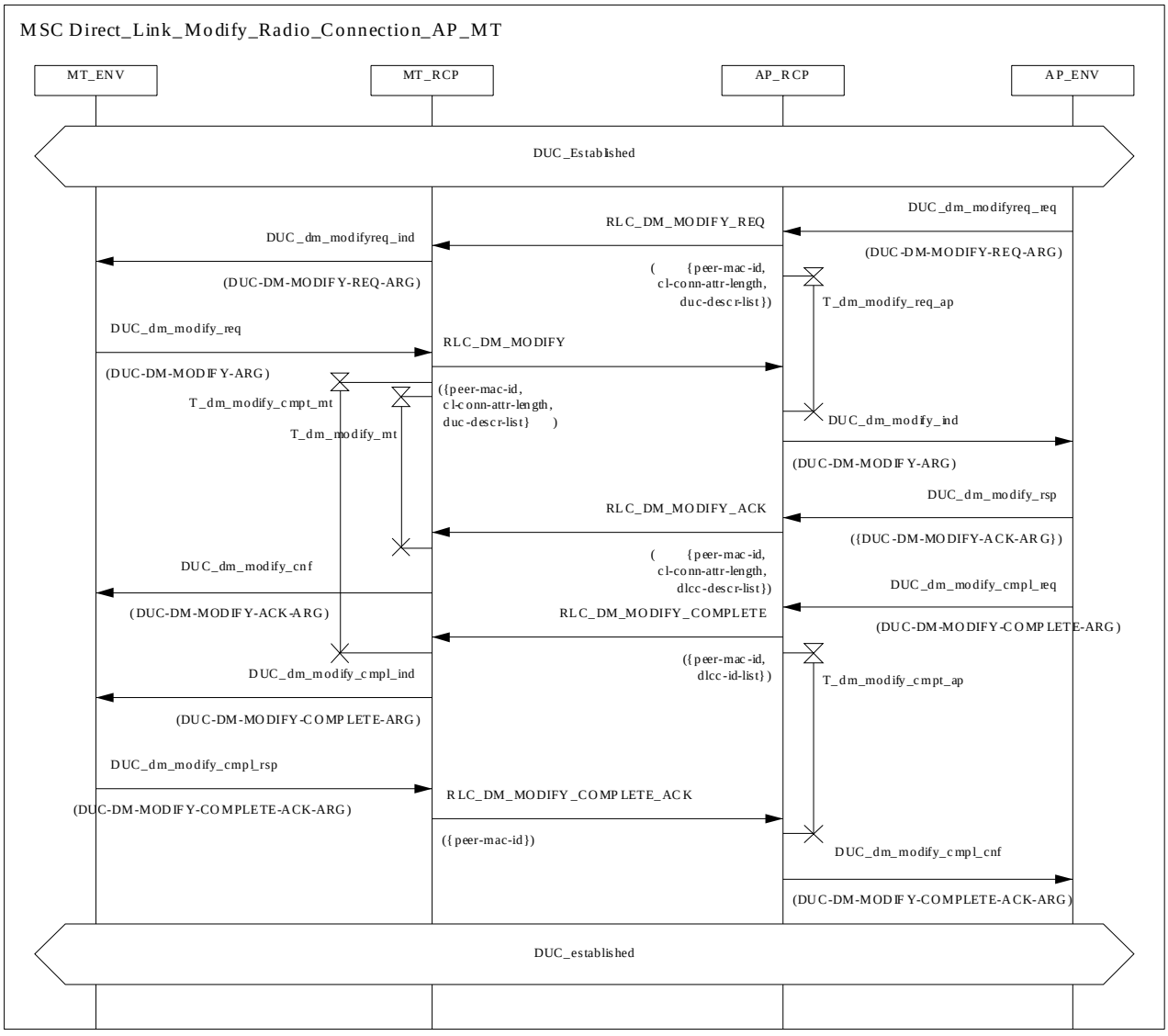


Diagram 80: Direct Link Modify connection AP/CC – MT, AP/CC initiated

5.3.10 Unicast Direct Link DUC Reset

With the reset procedure the ARQ instance and related timers of one or more unicast DiL DUCs shall be reset to their initial state. The DUC characteristics as agreed on Setup or latest modifications will be maintained. The direct link Reset procedure shall be initiated by sending the RLC_DM_RESET message including the DLCC ID(s) of the DUCs. The receiving entity (MT and/or AP/CC) shall acknowledge the Reset by responding with RLC_DM_RESET_ACK indicating the corresponding DLCC ID(s).

NOTE: These messages may not be used for connections using FCA or FSA.

5.3.10.1 AP/CC initiated DM DUC Reset

This procedure allows to reset one or more unicast DiL connections. The AP/CC should stop scheduling data to the particular MT after sending the RLC_DM_RESET message.

To reset the DUCs in the MT, the AP/CC shall send the RLC_DM_RESET message. The MT shall send the RLC_DM_RESET_ACK as positive acknowledgement, if the relevant DLCC_ID has been reseted.

The AP/CC shall do the same procedure with the other MT to complete the Reset of the direct mode connections.

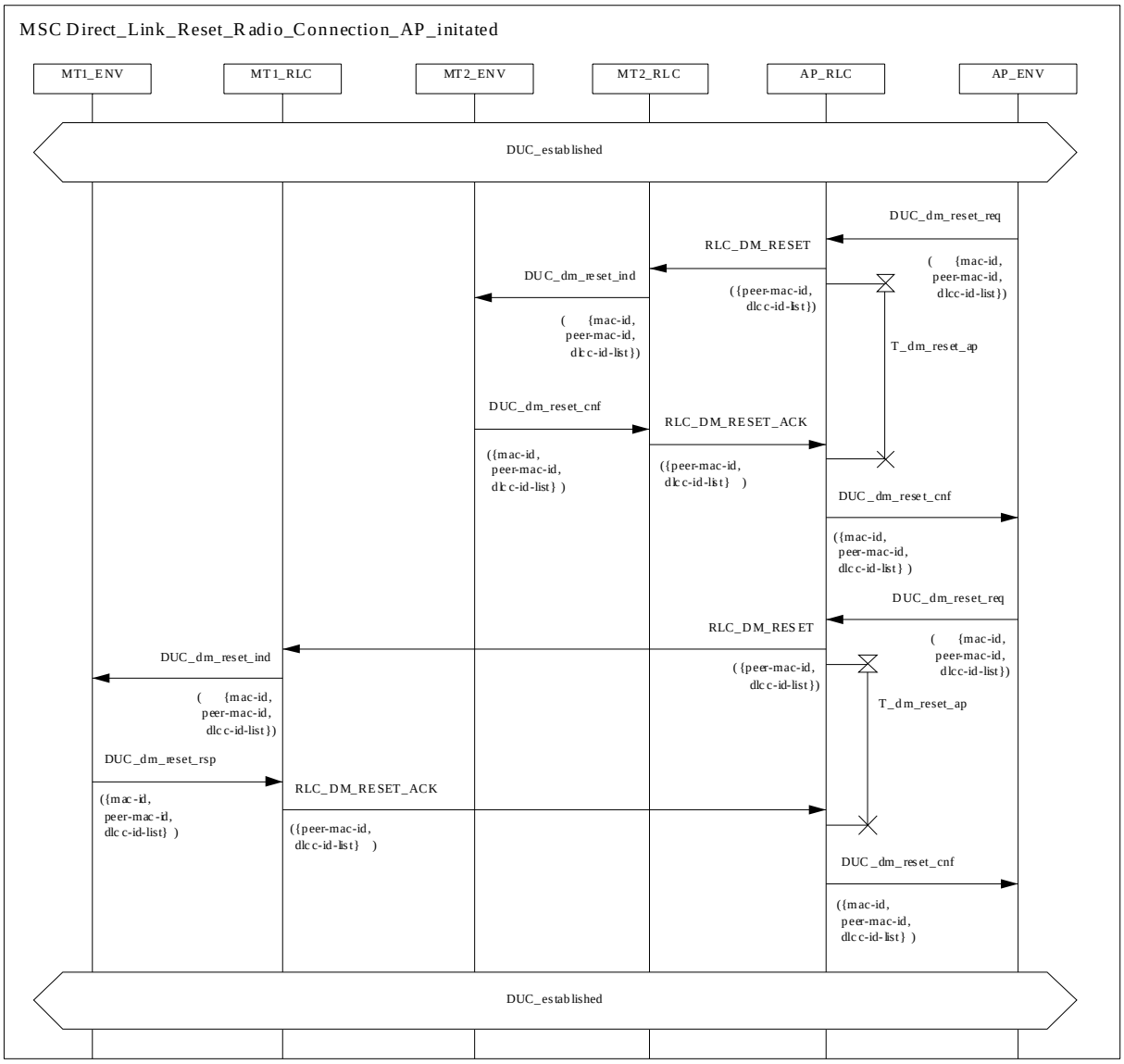


Diagram 81: Direct Link Reset connection procedure – AP/CC initiated

Table 106: RLC-DM-RESET

RLC-RESET-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
dlcc-id-list	DLCC-ID-LIST }

Table 107: RLC-DM-RESET-ACK

RLC-RESET-ACK-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-LCH-PDU-TYPE
peer-mac-id	MAC-ID
dlcc-id-list	DLCC-ID-LIST }

5.3.10.2 MT initiated DM DUC Reset

A MT shall use the RLC_DM_RESET message to reset a connection. When the AP/CC receives this message, it shall reset the connection with the other MT before sending back the RLC_DM_RESET_ACK. In case the second MT sends the RLC_DM_RESET_ACK as a positive acknowledgement, the AP/CC shall acknowledge to the MT that has initiated the reset procedure.

The AP/CC shall reset the ARQ instance just after receiving RLC_DM_RESET message. TxBoW and RxBoW of the corresponding DUCs shall be reset to zero both in the MT and the AP/CC.

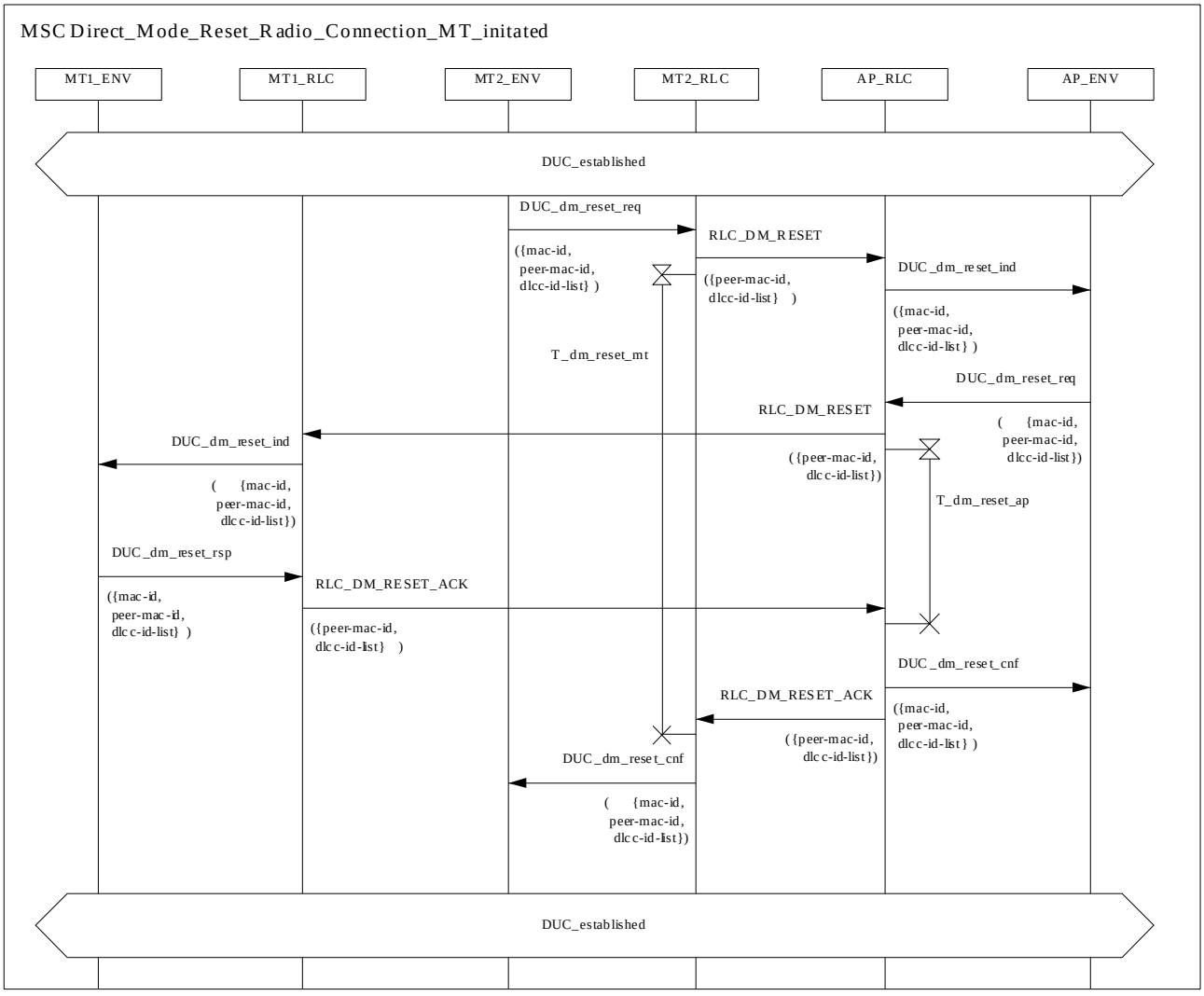


Diagram 82: Direct Link Reset connection procedure - MT initiated

5.3.11 Multicast Direct Link

Multicast DUCs are implicitly set up by Group Join procedure after the Association procedure, as defined in clause 5.3.11. The MAC IDs and DLCC IDs reserved for multicast connections are defined in [5].

5.3.12 Broadcast Direct Link

Broadcast DUCs are implicitly set up by Broadcast Join during the association procedure, as defined in clause 5.1.5. The MAC IDs and DLCC IDs reserved for broadcast connections are defined in [5].

6 Timers and repetitions of RLC messages

RLC messages use DLC unacknowledged mode and they use the most robust PHY mode. Retransmission at the RLC level shall be used to ensure the receiving of the messages. When a message is sent, a timer function shall be activated at the sender. If a reply to the sent message is not received within the time set for the timer function, the sender of the message shall send the message again (retransmit). The maximum number of retransmissions shall be 4, that is, the maximum number of transmissions shall be 5. The receiver of the message that is sent by the sender, shall respond within the time set for the timer function at the sender. The exception for this scheme are the unacknowledged broadcast messages (e.g. RLC_CHANGE_FREQUENCY), which may be sent more often and without specific limits in retransmission timers.

The timer values differ within wide ranges, both for implementation reasons and, for certain messages, due to delays in the fixed network. A message should not be retransmitted until the timer function has expired. For the case of large timer function values, this would mean that retransmission could take a long time (the number of retransmissions times the timer value). To avoid that to happen, the timer function shall work as follows. For all timer values, an ordinary timer shall be started at the sending of a message, supervising the arrival of the ordinary acknowledgement to the sent message. For long and medium timer values, an extra timer (short timer value) may be started. The function of the extra timer shall be to supervise the retransmission procedure of the sender. A reply message, RLC_PROCEEDING, shall be used as acknowledgement and to stop the extra (short) timer. The total retransmission time is then the number of retransmissions times the short time instead of the number of retransmission times the medium or long time.

The use of the extra timer should be used for time critical functions like association/handover.

The RLC_PROCEEDING message with the short extra timer shall also be used when a message has no answer to secure that messages will be retransmitted when needed.

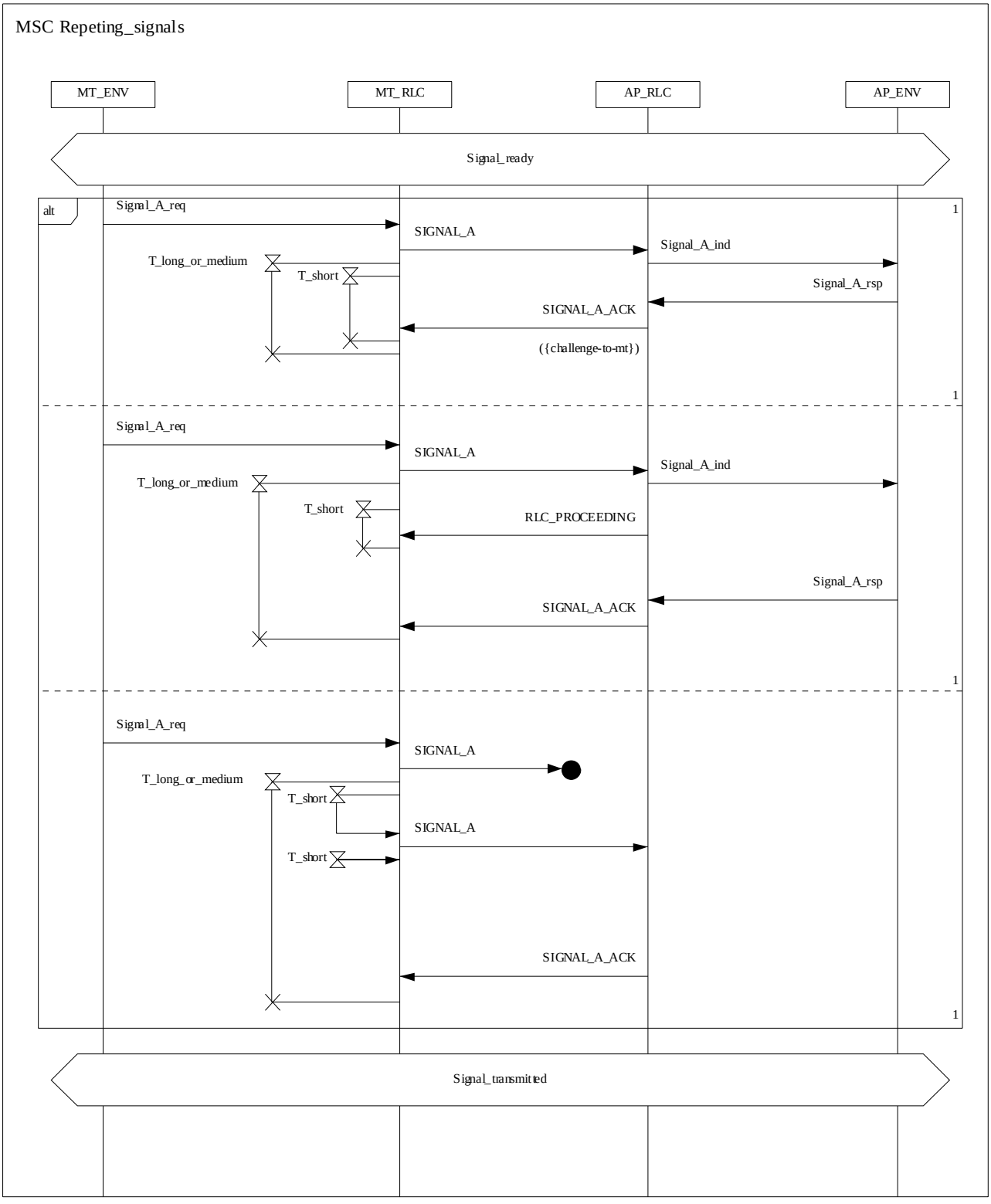


Diagram 83: Repetition and proceeding procedure

Table 108: RLC-PROCEEDING

RLC-PROCEEDING-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
sch-lch	SCH-LCH
no-support-pdu-type	PDU-TYPE-CHOICE
extension-type	EXTENSION-TYPE
mac-id	MAC-ID }

7 PDU for unsupported messages

The presentdocument contains several optional functions and there may be new versions and extensions in future. Therefore, there is no confirmation that both the MT and AP support all the messages they receive. In the case that the MT or the AP receives an RLC message that it does not support, it shall send the RLC_NO_SUPPORT message.

Table 109: RLC-NO-SUPPORT

RLC-NO-SUPPORT-ARG ::= SEQUENCE {	
rlc-pdu-type	RLC-SCH-PDU-TYPE
sch-lch	SCH-LCH
no-support-pdu-type	PDU-TYPE-CHOICE
extension-type	EXTENSION-TYPE
mac-id	MAC-ID }

8 Primitives

8.1 Primitive types

Four primitive types may be used:

- req (request), for a higher layer to request service from a lower layer;
- cnf (confirm), for the layer providing the service to confirm that the activity has been completed;
- ind (indication), for a layer providing a service to notify the next higher layer of any specific service related activity;
- rsp (response), for a layer to acknowledge receipt of an indication primitive from the next lower layer.

The defined types for each category of primitive are shown as a list in curly brackets.

NOTE: These primitives are defined only for the purpose of describing layer-to-layer interactions. The primitives are defined as an abstract list of parameters, and their concrete realization may vary between implementations. No formal testing of primitives is intended. The following primitive definitions have no normative significance.

8.2 Primitives to the Convergence Layer, DLC C-SAP

This clause summarizes the primitives between the convergence layer and the RLC layer.

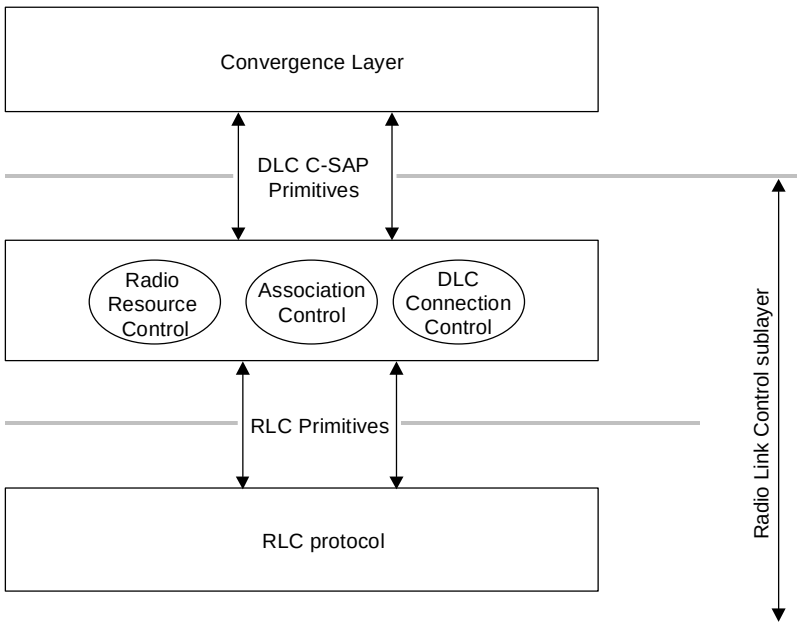


Figure 9

The primitives at the DLC C-SAP have a correspondence to a subset of the RLC primitives. The parameters used in the primitives at the DLC C-SAP are equal to or a subset of the RLC parameter.

At the AP the MAC ID is used to distinguish between different RLC instances. In the MT only one RLC instance exist.

The following DLC C-SAP primitives with their corresponding RLC primitives exist:

DLC C-SAP primitive	RLC primitive
DLC_SETUP - { req, ind };	DUC_SETUP - { req, ind }; DM_SETUP - { req, ind };
DLC_CONNECT - { req, cnf, ind, rsp };	DUC_CONN - { req, cnf, ind, rsp }; DM_CONN - { req, cnf, ind, rsp };
DLC_RELEASE - { req, cnf, ind, rsp };	DUC_RELEASE - { req, cnf, ind, rsp }; DM_RELEASE - { req, cnf, ind, rsp };
DLC_MODIFY - { req, cnf, ind, rsp };	DUC_MODIFY - { req, cnf, ind, rsp }; DM_MODIFY - { req, cnf, ind, rsp };
DLC_MULTICAST_JOIN - { req, cnf, ind, rsp };	ACF_GROUP_JOIN - { req, cnf, ind, rsp };
DLC_MULTICAST_LEAVE - { req, ind };	ACF_GROUP_LEAVE - { req, cnf, ind, rsp };
DLC_CL_BROADCAST_JOIN - { req, cnf, ind, rsp };	ACF_CL_BROADCAST_JOIN - {req, cnf, ind, rsp};
DLC_INFO_TRANSFER - { req, cnf, ind, rsp };	ACF_INFO - { req, cnf, ind, rsp };

NOTE: One DLC C-SAP primitive may correspond to one of possibly several RLC primitives. It is the task of the RLC functions (ACF and DCC) to control the relation between DLC C-SAP primitives and the RLC primitives. The RLC functions invoke the RLC primitives with appropriate parameters. (E.g. a request from the CL to setup 8 connections may result in 2 subsequent connection setup sequences at the RLC level, each setting up 4 connections.)

Annex A (normative): PDU type and Transfer Syntax Tables

A.1 RLC PDU type

The MT and AP columns indicate, whether the PDU is Mandatory or Optional to implement in the MT and AP. The PDU can be mandatory for both the sender and the receiver or the other peer only. If a PDU is mandatory for the sender, the sender shall be able to encode the PDU and send it at the correct time according to the present document. If the PDU is mandatory for the receiver, it shall be able to decode the PDU and perform the requested actions according to the present document. If the PDU is optional, the sender may not be able to encode or use it. If the PDU is optional the receiver may not be able to decode the PDU, but the receiver shall be able send corresponding RLC_NO_SUPPORT message.

A.1.1 LCH RLC PDU type

Table A.1: RLC ACF LCH PDU messages

LCH PDU type	RLC message name	MT	AP
(0000 0001) 1	RLC_RBCH_ASSOCIATION	M	M
2	RLC_LINK_CAPABILITY	M	M
3	RLC_LINK_CAPABILITY_ACK	M	M
4	RLC_KEY_EXCHANGE_MT_1	M	M
5	RLC_KEY_EXCHANGE_MT_2	M	M
6	RLC_KEY_EXCHANGE_AP_1	M	M
7	RLC_KEY_EXCHANGE_AP_2	M	M
8	RLC_AUTHENTICATION	M	M
9	RLC_AUTHENTICATION_MT	M	M
10	RLC_AUTHENTICATION_AP_1	M	M
11	RLC_AUTHENTICATION_AP_2	M	M
12	RLC_AUTHENTICATION_AP_3	M	M
13	RLC_AUTHENTICATION_ACK_1	M	M
14	RLC_AUTHENTICATION_ACK_2	M	M
15	RLC_AUTHENTICATION_ACK_3	M	M
16	RLC_DM_COMMON_KEY_DISTR	O	O
17	RLC_DM_COMMON_KEY_DISTR_ACK	O	O
18	RLC_INFO	O	O
19	RLC_INFO_ACK	O	O
20	RLC_UNICAST_KEY_REFRESH	M	O
21	RLC_UNICAST_KEY_REFRESH_ACK	M	O
22	RLC_COMMON_KEY_REFRESH	M	O
23	RLC_COMMON_KEY_REFRESH_ACK	M	O
24	RLC_GROUP_JOIN	O	O
25	RLC_GROUP_JOIN_ACK	O	O
26	RLC_GROUP_JOIN_NACK	O	O
27	RLC_GROUP_LEAVE	O	O
28	RLC_GROUP_LEAVE_ACK	O	O
29	RLC_CL_BROADCAST_JOIN	O	O
30	RLC_CL_BROADCAST_JOIN_ACK	O	O
31	RLC_CL_BROADCAST_LEAVE	O	O
32	RLC_CL_BROADCAST_LEAVE_ACK	O	O

Table A.2: RLC RRC LCH PDU messages

LCH PDU type	RLC message name	MT	AP
64	RLC_RADIO_HANDOVER_COMPLETE	O	O
65	RLC_HANDOVER_ASSOCIATION	O	O
66	RLC_HANDOVER_LINK_CAPABILITY_ACK	O	O
67	RLC_NW_SIGNALLING_HANDOVER	O	O
68	RLC_NW_SIGNALLING_HANDOVER_ACK	O	O
69	RLC_NETWORK_HANDOVER_COMPLETE	O	O
70	RLC_HO_INFO_DISTRIBUTION	O	O
71	RLC_DFS_MEASUREMENT_COMPLETE_REQUEST MENT_COMPLETE_REQUEST	M	M
72	RLC_DFS_MEASUREMENT_PERCENTILES_REQU EST	M	M
73	RLC_DFS_MEASUREMENT_SHORT_REQUEST	M	M
74	RLC_DFS_REPORT_COMPLETE	M	M
75	RLC_DFS_REPORT_PERCENTILES	M	M
76	RLC_DFS_REPORT_SHORT	M	M

Table A.3: RLC DUCC LCH PDU messages

LCH PDU type	RLC message name	MT	AP
128	RLC_SETUP	M	M
129	RLC_CONNECT	M	M
130	RLC_CONNECT_ACK	M	M
131	RLC_RELEASE	M	M
132	RLC_RELEASE_ACK	M	M
133	RLC_MODIFY_REQ	O	O
134	RLC_MODIFY	O	O
135	RLC_MODIFY_ACK	O	O
136	RLC_RESET	M	M
137	RLC_RESET_ACK	M	M
138	RLC_DM_SETUP	O	O
139	RLC_DM_CONNECT	O	O
140	RLC_DM_CONNECT_ACK	O	O
141	RLC_DM_CONNECT_COMPLETE	O	O
143	RLC_DM_RELAY_SETUP	O	O
144	RLC_DM_RELAY_SETUP_ACK	O	O
145	RLC_DM_MODIFY_REQ	O	O
146	RLC_DM_MODIFY	O	O
147	RLC_DM_MODIFY_ACK	O	O
148	RLC_DM_MODIFY_COMPLETE	O	O
149	RLC_DM_RELAY_MODIFY	O	O
150	RLC_DM_RELAY_MODIFY_ACK	O	O
151	RLC_DM_RELEASE	O	O
152	RLC_DM_RELEASE_ACK	O	O
153	RLC_DM_RELAY_RELEASE	O	O
154	RLC_DM_RELAY_RELEASE_ACK	O	O
155	RLC_DM_RESET	O	O
156	RLC_DM_RESET_ACK	O	O

A.1.2 SCH RLC PDU type

Table A.4: RLC ACF SCH PDU messages

SCH PDU type	RLC message name	MT	AP
(0000 0001) 1	RLC_RBCH_ASSOCIATION_REQ	O	M
2	RLC_MAC_ID_ASSIGN	M	M
3	RLC_MAC_ID_ASSIGN_ACK	M	M
4	RLC_MAC_ID_ASSIGN_NACK	M	M
5	RLC_COMMON_KEY_ACTIVATE	M	O
6	RLC_DISASSOCIATION	M	M
7	RLC_DISASSOCIATION_ACK	M	M
8	RLC_PROCEEDING	M	M
9	RLC_UNICAST_KEY_ACTIVATE	M	O

Table A.5: RLC RRC SCH PDU messages

SCH PDU type	RLC message name	MT	AP
64	RLC_SECTOR_HANDOVER_REQUEST	O	O
65	RLC_SECTOR_HANDOVER_ACK	O	O
66	RLC_HANDOVER_NOTIFY	O	O
67	RLC_HANDOVER_REQUEST	O	O
68	RLC_HANDOVER_REQUEST_NACK	O	O
70	RLC_HO_INFO_DISTRIBUTION_ACK	O	O
71	RLC_FORCE_HANDOVER	O	O
72	RLC_FORCE_HANDOVER_ACK	O	O
73	RLC_AP_ABSENCE	M	O
74	RLC_MT_INIT_REPORT_REQUEST	O	MO
75	RLC_MT_INIT_REPORT_REQUEST_ACK	O	MO
76	RLC_CHANGE_FREQUENCY	M	M
77	RLC_UPLINK_PC_CALIBRATION	M	M
78	RLC_MT_ALIVE_REQUEST	M	M
79	RLC_MT_ALIVE_REQUEST_ACK	M	M
80	RLC_MT_ALIVE	M	M
81	RLC_MT_ALIVE_ACK	M	M
82	RLC_MT_ABSENCE	O	O
83	RLC_MT_ABSENCE_ACK	O	O
84	RLC_SLEEP	O	M
85	RLC_SLEEP_ACK	O	M

Table A.6: RLC DUCC SCH PDU messages

SCH PDU type	RLC message name	MT	AP
128	RLC_DM_CONNECT_COMPLETE_ACK	O	O
129	RLC_DM_MODIFY_COMPLETE_ACK	O	O

Table A.7: OTHER RLC SCH PDU messages

SCH PDU type	RLC message name	MT	AP
255	RLC_NO_SUPPORT	M	M

A.2 Transfer Syntax Tables for LCH ACF messages

A.2.1 RLC-RBCH-ASSOCIATION encoding

	8	7	6	5	4	3	2	1
Octet 4	NOP-ID Local Part Length (= L)							
Octet 5	IDENTIFIER-FORMAT			NOP-ID Globally Unique Part Length (= UL)				
Octet 6	Future use							C-U-G
Octet 7	H2 Network Operator Identifier string – local part							
	H2 Network Operator Identifier string – globally unique part							
Octet	Future use					# PROFILE-VID (K)		
Octet	Profile-ID no.1					Profile-Version no. 1		
Octet	Profile-Version no. 1			Profile-ID no.2				
	Profile-Version no. 2				Profile-ID no.3			
		Profile-Version no. 3						
Octet	Other profile-id and profile version							
	Not used							
Octet 51								

Coding rule for profiles: Independent of the number of profiles, the whole of the last octet of the profile field that is not filled with profile information is filled with zeroes and the next information field is placed in the next octet. If the number of profiles happens to be 4, 5 whole octets are filled with profile information.

A.2.2 RLC-LINK-CAPABILITY encoding

	8	7	6	5	4	3	2	1
Octet 4	Future use					# PROFILE-VID (L)		
Octet 5	Profile-ID no.1					Profile-Version no. 1		
	Profile-Version no. 1		Profile-ID no.2					
	Profile-Version no. 2				Profile-ID no.3			
		Profile-Version no. 3						
	Other profile-id and profile version							
Octet	Freq-band-MT		RSS value					
Octet 8 + (2 x N)	64QAM?	DM-cap	Cyclic prefix	FCA?	FSA?	Time-gap-ACH-UL		
Octet	Future use		ho-cap	cc-ho-cap	Future use	Duty-cycle		
Octet 9 + (2 x N)	ARQ-DELAY-rx		ARQ-DELAY-tx		Auth/Encr-No-of-Proposals (K)			
Octet 10 + (2 x N)	Authentication-Proposal-#1				Encryption-Proposal-#1			
Octet 11 + (2 x N)	Authentication-Proposal-#2				Encryption-Proposal-#2			
…	Possibly used for up to 15 proposals (for one CL)							
Octet	DIL-power-control		TX-ARQ-WIN-SIZE			RX-ARQ-WIN-SIZE		
	Not used (size depends on #CL_VID, DM-cap, #of proposals)							
…								
Octet 51								

A.2.3 RLC-LINK-CAPABILITY-ACK encoding

	8	7	6	5	4	3	2	1
Octet 4	Future use					# PROFILE-VID (L)		
Octet 5	Profile-ID no.1					Profile-Version no. 1		
	Profile-Version no. 1		Profile-ID no.2					
	Profile-Version no. 2				Profile-ID no.3			
		Profile-Version no. 3						
	Other profile-id and profile version							
Octet	Freq-band-sel		RSS-value					
Octet	APT-ADDRESS-LENGTH				64QAM	DMCkey	DM-cap	Cyclic prefix
Octet	FCA	FSA	Future use	cc-ho-cap	ARQ-DELAY-rx		ARQ-DELAY-tx	
Octet	Authentication-Selected				Encryption-Selected			
Octet	DIL-power-control		OUT-ARQ-WIN-SIZE		IN-ARQ-WIN-SIZE			
Octet	Not used							
Octet …								
Octet 51								

A.2.4 RLC-KEY-EXCHANGE-MT-1 encoding

	8	7	6	5	4	3	2	1
Octet 4	MT_DH_PUBLIC_VALUE_PART1							
…								
Octet 51								

A.2.5 RLC-KEY-EXCHANGE-MT-2 encoding

	8	7	6	5	4	3	2	1
Octet 4	MT_DH_PUBLIC_VALUE_PART2							
…								
Octet 51								

A.2.6 RLC-KEY-EXCHANGE-AP-1 encoding

	8	7	6	5	4	3	2	1
Octet 4	AP_DH_PUBLIC_VALUE_PART1							
…								
Octet 51								

A.2.7 RLC-KEY-EXCHANGE-AP-2 encoding

	8	7	6	5	4	3	2	1
Octet 4	AP_DH_PUBLIC_VALUE_PART2							
…								
Octet 51								

A.2.8 RLC-AUTHENTICATION encoding

	8	7	6	5	4	3	2	1
Octet 4	More	Future use	Length of MT-AUTH-ID in this PDU (L)					
Octet 5	Future use				MT-AUTH-ID-TYPE			
Octet 6	MT-AUTH-ID-CONTENT (L)							
Octet L + 5								
Octet L + 6	Not used							
Octet 51								

A.2.9 RLC-AUTHENTICATION-MT encoding

	8	7	6	5	4	3	2	1
Octet 4	CHALLENGE_TO_MT							
…								
Octet 19								
Octet 20	Not used							
…								
Octet 51								

A.2.10 RLC-AUTHENTICATION-AP-1 encoding

	8	7	6	5	4	3	2	1
Octet 4	CHALLENGE-TO-AP							
…								
Octet 19								
Octet 20	MT_RESPONSE (Possible total length over several PDUs: 16, 64, 96, 128 octets. Which one is given by the authentication procedure negotiated during the link capability phase.)							
…								
Octet …	Not used if MT_RESPONSE total length = 16							
Octet …								
Octet 51								

A.2.11 RLC-AUTHENTICATION-AP-2 encoding

	8	7	6	5	4	3	2	1
Octet 4	MT_RESPONSE (Possible total length over several PDUs: 64, 96, 128 octets. Which one is given by the authentication procedure negotiated during the link capability phase.)							
…								
Octet …	Not used if MT_RESPONSE = 64							
Octet …								
Octet 51								

A.2.12 RLC-AUTHENTICATION-AP-3 encoding

	8	7	6	5	4	3	2	1
Octet 4	MT_RESPONSE (Possible total length over several PDUs: 96, 128 octets. Which one is given by the authentication procedure negotiated during the link capability phase.)							
…								
Octet …	Not used if MT_RESPONSE = 96							
Octet …								
Octet 51								

A.2.13 RLC-AUTHENTICATION-ACK-1 encoding

	8	7	6	5	4	3	2	1
Octet 4	AP_RESPONSE (Possible total length over several PDUs: 16, 64, 96, 128 octets. Which one is given by the authentication procedure negotiated during the link capability phase.)							
…								
Octet …	Not used (if not by AP_RESPONSE)							
Octet …								
Octet 51								

A.2.14 RLC-AUTHENTICATION-ACK-2 encoding

	8	7	6	5	4	3	2	1
Octet 4	AP_RESPONSE (Possible total length over several PDUs: 16, 64, 96, 128 octets. Which one is given by the authentication procedure negotiated during the link capability phase.)							
…								
Octet …	Not used (if not by AP_RESPONSE)							
Octet …								
Octet 51								

A.2.15 RLC-AUTHENTICATION-ACK-3 encoding

	8	7	6	5	4	3	2	1
Octet 4	AP_RESPONSE (Possible total length over several PDUs: 16, 64, 96, 128 octets. Which one is given by the authentication procedure negotiated during the link capability phase.)							
…								
Octet …	Not used (if not by AP_RESPONSE)							
Octet …								
Octet 51								

A.2.16 RLC-DM-COMMON-KEY-DISTR encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use				Encryption algorithm			
Octet 5	KEY-ID							
Octet 6	KEY. Length according to encryption algorithm (either 0, 8, or 24 octets).							
Octet ...								
Octet ...								
Octet ...								
Octet ...	Not used							
Octet ...								
Octet 51								

A.2.17 RLC-DM-COMMON-KEY-DISTR-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use				Encryption algorithm			
Octet 5	MD5-ON-KEY							
Octet ...								
Octet 20								
Octet 21	Not used							
Octet ...								
Octet 54								

A.2.18 RLC-INFO encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use		Info-type	INFO-COUNT			DLC-attr-pr	CL-attr-pr
Octet 5	Future use		CL Attribute Length (= L1) (in octets)					
Octet 6	CL-ID							
Octet 7	CL attributes (L1)							
…								
…								
Octet 7 + L1	Future use		DLC Attribute Length (= L2) (in octets)					
…	DLC-Attributes							
7 + L1 + L2								
Octet …	Not used							
Octet …								
Octet 51								

A.2.19 RLC-INFO-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use			INFO-COUNT			DLC-attr-pr	CL-attr-pr
Octet 5	Future use		CL Attribute Length (= L1) (in octets)					
Octet 6	CL-ID							
Octet 7	CL attributes (L1)							
…								
…								
Octet 7 + L1	Future use		DLC Attribute Length (= L2) (in octets)					
…	DLC-Attributes							
7 + L1 + L2								
Octet …	Not used							
Octet …								
Octet 51								

A.2.20 RLC-UNICAST-KEY-REFRESH encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 4	NONCE							
Octet ...								
Octet ...								
Octet 19								
Octet 20	Not used							
Octet ...								
Octet 51								

A.2.21 RLC-UNICAST-KEY-REFRESH-ACK encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 4	MD5-ON-NONCE							
Octet ...								
Octet ...								
Octet 19								
Octet 20	Not used							
Octet ...								
Octet 51								

A.2.22 RLC-COMMON-KEY-REFRESH encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 4	Future use				Encr Info			
Octet 5	KEY-ID							
Octet 6	KEY. Length according to encryption algorithm.							
Octet ...								
Octet ...								
Octet ...								
Octet ...	Not used							
Octet ...								
Octet 51								

A.2.23 RLC-COMMON-KEY-REFRESH-ACK encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 4	Future use				Encr Info			
Octet 5	MD5-ON-KEY							
Octet 6								
Octet ...								
Octet ...								
Octet 20								
Octet 21	Not used							
Octet ...								
Octet 51								

A.2.24 RLC-GROUP-JOIN encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute in octets (L)				Number of CL attributes (n)			
Octet 5	CL-ID							
Octet 6	CL-attributes (Higher layer group addresses)							
Octet ...								
Octet (L x n + 5)								
Octet (L x n + 6)	Future use				No. of encryption proposals (k)			
Octet (L x n + 7)	Encryption proposal no. 1				Encryption proposal no. 2			
...	...				...			
...	...				...			
Octet (L x n + 6 + k/2)	Encryption proposal no. k-1				Encryption proposal no. k			
Octet ...	Not used							
Octet ...								
Octet 51								

A.2.25 RLC-GROUP-JOIN-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	more-joins	Future use			Number of MAC-ID:s And CL-data (N)			
Octet 5					Length-of-CL-data in octets (L)			
Octet 6	MAC-ID no. 1							
Octet 7	CL-data no.1							
Octet 7 + L								
Octet	MAC-ID no. N							
	CL-data no.N							
Octet N x (L + 1) + 5								
Octet N x (L + 1) + 6	Future use				Encryption algorithm selected			
Octet N x (L + 1) + 7	Key-ID							
	Common Key. Length given by Encryption algorithm selected (K). K is 0 octets for no-encr, 8 octets for DES, and 24 octets for tripleDES.							
Octet N x (L + 1) + 7 + K								
	Not used							
Octet ...								
Octet 51								

A.2.26 RLC-GROUP-JOIN-NACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute(L)				Number of CL attributes (N)			
Octet 5	CL-ID							
	CL Attributes (Higher layer group addresses)							
Octet 5 + L x N								
	Not used							
Octet 51								

A.2.27 RLC-GROUP-LEAVE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute (L)				Number of CL attributes (N)			
Octet 5	CL-ID							
	CL Attributes (Higher layer group addresses)							
Octet 5 + L x N								
	Not used							
Octet 51								

A.2.28 RLC-GROUP-LEAVE-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute (L)				Number of CL attributes (N)			
Octet 5	CL-ID							
	CL Attributes (Higher layer group addresses)							
Octet 5 + L x N								
	Not used							
Octet 51								

A.2.29 RLC-CL-BROADCAST-JOIN encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute (L)				Number of CL attributes (N)			
Octet 5	CL-ID							
Octet 6	CL attributes (Higher layer broadcast addresses)							
Octet 7								
Octet 8								
Octet L x N + 5)								
Octet (L x n + 6)	Future use				No. of encryption proposals (k)			
Octet (L x n + 7)	Encryption proposal no. 1				Encryption proposal no. 2			
Octet ...	...				...			
Oct (L x n + 6 + k/2)	Encryption proposal no. (k-1)				Encryption proposal no. k			
	Not used							
Octet 51								

A.2.30 RLC-CL-BROADCAST-JOIN-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	more-joins	rep/unack	Window size		No. of MAC-ID:s and CL-data (N)			
Octet 5					Length of CL-data in octets (L)			
Octet 6	MAC-ID no. 1							
	CL-data no. 1 (L)							
	MAC-ID no. 2							
	CL-data no.2 (L)							
	MAC-ID no. N							
	CL-data no. N (L)							
Octet 5 + (N x L)								
	Future use				Encryption algorithm selected			
Octet 5 + (N x L) + 2	Key-ID							
	Key. Length given by Encryption algorithm selected (K) K is 0 octets for no-encr, 8 octets for DES and 24 octets for tripleDES							
Octet 5 + (N x L) + 2 + K								
	Not used							
Octet 51								

A.2.31 RLC-CL-BROADCAST-LEAVE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute (L)				Number of CL attributes (N)			
Octet 5	CL-ID							
Octet 6	CL attributes (Higher layer or peer layer broadcast addresses)							
Octet (L x n + 5)								
	Not used							
Octet 51								

A.2.32 RLC-CL-BROADCAST-LEAVE-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Length of each CL attribute (L)				Number of CL attributes (N)			
Octet 5	CL-ID							
Octet 6	CL attributes (Higher layer or peer layer broadcast addresses)							
Octet L x N + 5								
	Not used							
Octet 51								

A.3 Transfer Syntax Tables for LCH RRC messages

A.3.1 RLC-RADIO-HANDOVER-COMPLETE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use				MAC-ID-OLD			
Octet 5	MAC-ID-OLD				AP-ID-OLD			
Octet 6	AP-ID-OLD						NET-ID-OLD	
Octet 7	NET-ID-OLD							
Octet 8	MAC-ID-NEW							
Octet 9	CL-ID							
Octet 10	EXT-IND	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 11	# of DUC:s (N)	Future use						
Octet12	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet (12 + L)								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION	Future use			CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use		DUC1-FW-ARQ-WIN-SIZE	
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-FW-TYPE-OF-ALLOCATION	Future use			CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use		DUC1-FW-ARQ-WIN-SIZE	
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

Total length = 12 + L + 14 x N if asymmetric duplex with fixed capacity agreement and with FEC

Total length = 12 + L + 6 x N if asymmetric duplex with basic allocation and with FEC

Total length = 12 + L + 7 x N if symmetric duplex with fixed capacity agreement and with FEC

Total length = 12 + L + 3 x N if symmetric duplex with basic allocation and with FEC

Total length = 12 + L + 3 x N if simplex with basic allocation and with FEC

Total length = 12 + L + 7 x N if simplex with fixed capacity agreement and with FEC

Total length = 12 + L + 12 x N if asymmetric duplex with fixed capacity agreement and without FEC

Total length = 12 + L + 4 x N if asymmetric duplex with basic allocation and without FEC

Total length = 12 + L + 6 x N if symmetric duplex with fixed capacity agreement and without FEC

Total length = 12 + L + 2 x N if symmetric duplex with basic allocation and without FEC

Total length = 12 + L + 2 x N if simplex with basic allocation and without FEC

Total length = 12 + L + 6 x N if simplex with fixed capacity agreement and without FEC

Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In the following the total length of the PDU is given for the unacknowledged mode:

Total length = 12 + L + 18 x N if asymmetric duplex with FSA and with FEC

Total length = 12 + L + 9 x N if symmetric duplex with FSA and with FEC

Total length = 12 + L + 9 x N if simplex with FSA and with FEC

Total length = 12 + L + 16 x N if asymmetric duplex with FSA and without FEC

Total length = 12 + L + 8 x N if symmetric duplex with FSA and without FEC

Total length = 12 + L + 8 x N if simplex with FSA and without FEC

A.3.2 RLC-HANDOVER-ASSOCIATION encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID-OLD							
Octet 5	Future use				AP-ID-OLD			
Octet 6	AP-ID-OLD						NET-ID-OLD	
Octet 7	NET-ID-OLD							
Octet 8	MAC-ID-NEW							
Octet …	Future use							
Octet 51								

A.3.3 RLC-HANDOVER-LINK-CAPABILITY-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use					# PROFILE-VID (L)		
Octet 5	Profile-ID no.1					Profile-Version no. 1		
Octet ...	Profile-Version no. 1		Profile-ID no.2					
Octet ...	Profile-Version no. 2				Profile-ID no.3			
Octet ...		Profile-Version no. 3						
Octet ...	Other profile-id and profile version							
Octet ...	Freq-band-sel		RSS-value					
Octet ...	APT-ADDRESS-LENGTH				64QAM ?	APDMcap	DMCkey	Cyclic prefix
Octet ...	FCA?	FSA?	Future use	cc-ho-cap	ARQ-DELAY-rx		ARQ-DELAY-tx	
Octet ...	Authentication-Selected				Encryption-Selected			
Octet ...	Encrypt?	Authent?	NWTkn?	DUCSu?	Future use		connections	Info-transfer
Octet ...	DIL-power-control		TX-ARQ-WIN-SIZE		RX-ARQ-WIN-SIZE			
Octet ...	Not used							
Octet ...								
Octet 51								

A.3.4 RLC-NW-SIGNALLING-HANDOVER encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MD5-ON-MT-TOKEN-AUTH-ENCR							
...								
…								
Octet 19								
Octet 20	Not used							
Octet ...								
Octet 51								

A.3.5 RLC-NW-SIGNALLING-HANDOVER-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	AP-token							
…								
Octet 19								
Octet 20					Auth/Encr-No-of-Proposals (K)			
Octet 21	Authentication-Proposal-#1				Encryption-Proposal-#1			
…	…				…			
Octet 20 + K	Authentication-Proposal-#K				Encryption-Proposal-#K			
Octet 21 + K	Authentication-Proposal-Selected				Encryption-Proposal-Selected			
Octet ...	Future use							
Octet ...								
Octet 51								

A.3.6 RLC-NETWORK-HANDOVER-COMPLETE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	CL-ID							
Octet 5	EXT-IND	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 6	# of DUC:s (N)		Future use					
Octet 7	DUC1-DIRECTION		DLCC-ID					
	CL-CONN-ATTR							
Octet (7 + L)								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	Cyclic-pref	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	Cyclic-pref	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

Total length = 7 + L + 14 x N if asymmetric duplex with fixed capacity agreement and with FEC

Total length = 7 + L + 6 x N if asymmetric duplex with basic allocation and with FEC

Total length = 7 + L + 7 x N if symmetric duplex with fixed capacity agreement and with FEC

Total length = 7 + L + 3 x N if symmetric duplex with basic allocation and with FEC

Total length = 7 + L + 3 x N if simplex with basic allocation and with FEC

Total length = 7 + L + 7 x N if simplex with fixed capacity agreement and with FEC

Total length = 7 + L + 12 x N if asymmetric duplex with fixed capacity agreement and without FEC

Total length = 7 + L + 4 x N if asymmetric duplex with basic allocation and without FEC

Total length = 7 + L + 6 x N if symmetric duplex with fixed capacity agreement and without FEC

Total length = 7 + L + 2 x N if symmetric duplex with basic allocation and without FEC

Total length = 7 + L + 2 x N if simplex with basic allocation and without FEC

Total length = 7 + L + 6 x N if simplex with fixed capacity agreement and without FEC

Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In the following the total length of the PDU is given for the unacknowledged mode:

Total length = 7 + L + 18 x N if asymmetric duplex with FSA and with FEC

Total length = 7 + L + 9 x N if symmetric duplex with FSA and with FEC

Total length = 7 + L + 9 x N if simplex with FSA and with FEC

Total length = 7 + L + 16 x N if asymmetric duplex with FSA and without FEC

Total length = 7 + L + 8 x N if symmetric duplex with FSA and without FEC

Total length = 7 + L + 8 x N if simplex with FSA and without FEC

A.3.7 RLC-HO-INFO-DISTRIBUTION encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	TOKEN							
Octet ...								
Octet ...								
Octet 19								
Octet 20	Not used							
Octet ...								
Octet 51								

A.3.8 RLC-DFS-MEASUREMENT-COMPLETE-REQUEST encoding

	8	7	6	5	4	3	2	1
Octet 4	FREQUENCY-INDEX							
Octet 5	Future use			UOA	START-OF-MEASUREMENT			
Octet 6	Future use		MEASUREMENT-WINDOW					
Octet 7	Future use				MAXIMUM-AGE-OF-MEASUREMENT			
Octet 8	MAXIMUM-AGE-OF-MEASUREMENT							
Octet 9	Future use				RSS-INDEX 1			
Octet 10	RSS-INDEX 2				RSS-INDEX 3			
Octet 11	RSS-INDEX 4				RSS-INDEX 5			
Octet ...	Not used							
Octet ...								
Octet 51								

A.3.9 RLC-DFS-MEASUREMENT-PERCENTILES-REQUEST encoding

	8	7	6	5	4	3	2	1
Octet 4	FREQUENCY-INDEX							
Octet 5	Future use			UOA	START-OF-MEASUREMENT			
Octet 6	Future use		MEASUREMENT-WINDOW					
Octet 7	Future use							
Octet 8								
Octet 9	Future use				RSS-INDEX 1			
Octet 10	RSS-INDEX 2				RSS-INDEX 3			
Octet 11	RSS-INDEX 4				RSS-INDEX 5			
Octet ...	Not used							
Octet ...								
Octet 51								

A.3.10 RLC-DFS-MEASUREMENT-SHORT-REQUEST encoding

	8	7	6	5	4	3	2	1
Octet 4	FREQUENCY-INDEX FREQUENCY-INDEX							
Octet 5	Future use			UOA	START-OF-MEASUREMENT			
Octet 6	Future use		MEASUREMENT-WINDOW					
Octet 7	Future use				MAXIMUM-AGE-OF-MEASUREMENT			
Octet 8	MAXIMUM-AGE-OF-MEASUREMENT							
Octet …	Not used							
Octet …								
Octet 51								

A.3.11 RLC-DFS-REPORT-COMPLETE encoding

	8	7	6	5	4	3	2	1
Octet 4	FREQUENCY-INDEX							
Octet 5	Future use			OAU	AGE-OF-MEASUREMENT			
Octet 6	AGE-OF-MEASUREMENT							
Octet 7	Future use		LAST-OWN-BCH-RX-LEVEL					
Octet 8	Future use		NUMBER-OF-SAMPLES					
Octet 9	NUMBER-OF-SAMPLES							
Octet 10	BCH-FOUND	Future use		TRAFFIC-LOAD			AP-ID	
Octet 11	AP-ID							
Octet 12	Future use		TX-LEVEL				NET-ID	
Octet 13	NET-ID							
Octet 14	Future use		BCH-RX-LEVEL					
Octet 15	Future use				RSS-INDEX 1			
Octet 16	RSS-INDEX 2				RSS-INDEX 3			
Octet 17	RSS-INDEX 4				RSS-INDEX 5			
Octet 18	Future use		RSS-STATISTICS 1					
Octet 19	Future use		RSS-STATISTICS 2					
Octet 20	Future use		RSS-STATISTICS 3					
Octet 21	Future use		RSS-STATISTICS 4					
Octet 22	Future use		RSS-STATISTICS 5					
Octet 23	Not used							
Octet …								
Octet 51								

A.3.12 RLC-DFS-REPORT-PERCENTILES encoding

	8	7	6	5	4	3	2	1
Octet 4	FREQUENCY-INDEX							
Octet 5	Future use			OAU	Future use			
Octet 6	Future use							
Octet 7	Future use		LAST-OWN-BCH-RX-LEVEL					
Octet 8	Future use		NUMBER-OF-SAMPLES					
Octet 9	NUMBER-OF-SAMPLES							
Octet 10	Future use							
Octet 11								
Octet 12								
Octet 13								
Octet 14								
Octet 15	Future use				RSS-INDEX 1			
Octet 16	RSS-INDEX 2				RSS-INDEX 3			
Octet 17	RSS-INDEX 4				RSS-INDEX 5			
Octet 18	Future use		RSS-STATISTICS 1					
Octet 19	Future use		RSS-STATISTICS 2					
Octet 20	Future use		RSS-STATISTICS 3					
Octet 21	Future use		RSS-STATISTICS 4					
Octet 22	Future use		RSS-STATISTICS 5					
Octet 23	Not used							
Octet …								
Octet 51								

A.3.13 RLC-DFS-REPORT-SHORT encoding

	8	7	6	5	4	3	2	1
Octet 4	FREQUENCY-INDEX							
Octet 5	Future use			OAU	AGE-OF-MEASUREMENT			
Octet 6	AGE-OF-MEASUREMENT							
Octet 7	Future use		LAST-OWN-BCH-RX-LEVEL					
Octet 8	Future use							
Octet 9								
Octet 10	BCH-FOUND	Future use		TRAFFIC-LOAD			AP-ID	
Octet 11	AP-ID							
Octet 12	Future use		TX-LEVEL				NET-ID	
Octet 13	NET-ID							
Octet 14	Future use		BCH-RX-LEVEL					
Octet 15	Not used							
Octet …								
Octet 51								

A.4 Transfer Syntax Tables for LCH DUCC messages

A.4.1 RLC-SETUP encoding

	8	7	6	5	4	3	2	1
Octet 4	CL-ID							
Octet 5	duc-ext	CL-CONN-ATTR-LENGTH(L)					# of DUC:s	
Octet 6	# of DUC:s(N)		Future use					
Octet 7	DUC1-DIRECTION		DLCC-ID					
	CL-CONN-ATTR							
Octet 7 + L								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME(SCH)				PER-#-MAC-FRAME(LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME(SCH)				PER-#-MAC-FRAME(LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

Total length = 7 + L + 14 x N if asymmetric duplex with fixed capacity agreement and with FEC

Total length = 7 + L + 6 x N if asymmetric duplex with basic allocation and with FEC

Total length = 7 + L + 7 x N if symmetric duplex with fixed capacity agreement and with FEC

Total length = 7 + L + 3 x N if symmetric duplex with basic allocation and with FEC

Total length = 7 + L + 3 x N if simplex with basic allocation and with FEC

Total length = 7 + L + 7 x N if simplex with fixed capacity agreement and with FEC

Total length = 7 + L + 12 x N if asymmetric duplex with fixed capacity agreement and without FEC

Total length = 7 + L + 4 x N if asymmetric duplex with basic allocation and without FEC

Total length = 7 + L + 6 x N if symmetric duplex with fixed capacity agreement and without FEC

Total length = 7 + L + 2 x N if symmetric duplex with basic allocation and without FEC

Total length = 7 + L + 2 x N if simplex with basic allocation and without FEC

Total length = 7 + L + 6 x N if simplex with fixed capacity agreement and without FEC

Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In the following the total length of the PDU is given for the unacknowledged mode:

Total length = 7 + L + 18 x N if asymmetric duplex with FSA and with FEC

Total length = 7 + L + 9 x N if symmetric duplex with FSA and with FEC

Total length = 7 + L + 9 x N if simplex with FSA and with FEC

Total length = 7 + L + 16 x N if asymmetric duplex with FSA and without FEC

Total length = 7 + L + 8 x N if symmetric duplex with FSA and without FEC

Total length = 7 + L + 8 x N if simplex with FSA and without FEC

A.4.2 RLC-CONNECT encoding

	8	7	6	5	4	3	2	1
Octet 4	CL-ID							
Octet 5	Future use	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 6	# of DUC:s (N)	Future use						
Octet 7	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet 7 + L								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...	Not used							
Octet 51								

Total length = 7 + L + 14 x N if asymmetric duplex with fixed capacity agreement and with FEC

Total length = 7 + L + 6 x N if asymmetric duplex with basic allocation and with FEC

Total length = 7 + L + 7 x N if symmetric duplex with fixed capacity agreement and with FEC

Total length = 7 + L + 3 x N if symmetric duplex with basic allocation and with FEC

Total length = 7 + L + 3 x N if simplex with basic allocation and with FEC

Total length = 7 + L + 7 x N if simplex with fixed capacity agreement and with FEC

Total length = 7 + L + 12 x N if asymmetric duplex with fixed capacity agreement and without FEC

Total length = 7 + L + 4 x N if asymmetric duplex with basic allocation and without FEC

Total length = 7 + L + 6 x N if symmetric duplex with fixed capacity agreement and without FEC

Total length = 7 + L + 2 x N if symmetric duplex with basic allocation and without FEC

Total length = 7 + L + 2 x N if simplex with basic allocation and without FEC

Total length = 7 + L + 6 x N if simplex with fixed capacity agreement and without FEC

Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In the following the total length of the PDU is given for the unacknowledged mode:

Total length = 7 + L + 18 x N if asymmetric duplex with FSA and with FEC

Total length = 7 + L + 9 x N if symmetric duplex with FSA and with FEC

Total length = 7 + L + 9 x N if simplex with FSA and with FEC

Total length = 7 + L + 16 x N if asymmetric duplex with FSA and without FEC

Total length = 7 + L + 8 x N if symmetric duplex with FSA and without FEC

Total length = 7 + L + 8 x N if simplex with FSA and without FEC

A.4.3 RLC-CONNECT-ACK encoding

<table>
<tr><th></th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th></tr>
<tr><td>Octet 4</td><td colspan="8">CL-ID (filled with same contents as setup message)</td></tr>
<tr><td>Octet 5</td><td>Future use</td><td colspan="5">CL-CONN-ATTR-LENGTH (L)</td><td colspan="2"># of DLCC+CL-CON-ATT</td></tr>
<tr><td>Octet 6</td><td colspan="2"># of DLCC+CL-CON-ATT(N)</td><td colspan="6">Future use</td></tr>
<tr><td>Octet ...</td><td colspan="2">Future use</td><td colspan="6">DLCC-ID-1</td></tr>
<tr><td>Octet ...</td><td colspan="8" rowspan="2">CL-CONN-ATTR-1</td></tr>
<tr><td>Octet ...</td></tr>
<tr><td>Octet ...</td><td colspan="2">Future use</td><td colspan="6">DLCC-ID-2</td></tr>
<tr><td>Octet ...</td><td colspan="8" rowspan="2">CL-CONN-ATTR-2</td></tr>
<tr><td>Octet … 6 + (L + 1) x N</td></tr>
<tr><td>Octet ...</td><td colspan="8" rowspan="3">Not used</td></tr>
<tr><td>Octet ...</td></tr>
<tr><td>Octet 51</td></tr>
</table>

A.4.4 RLC-RELEASE encoding

	8	7	6	5	4	3	2	1
Octet 4	Future use				RELEASE-CAUSE			
Octet 5	Future use				# of DLCC-ID (N)			
Octet ...	Future use		DLCC-ID#1					
...	Future use		DLCC-ID…					
Octet 5 + N	Future use		DLCC-ID#N					
Octet ...	Not used							
Octet ...								
Octet 51								

A.4.5 RLC-RELEASE-ACK encoding

	8	7	6	5	4	3	2	1
Octet 4	Future use				# of DLCC-ID (N)			
Octet ...	Future use		DLCC-ID#1					
...	Future use		DLCC-ID…					
Octet 4 + N	Future use		DLCC-ID#N					
Octet ...	Not used							
Octet ...								
Octet 51								

A.4.6 RLC-MODIFY-REQUEST encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	duc-ext-ind	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 5	# of DUC:s (N)	Future use						
Octet 6	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet 6 + L								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

Total length = 6 + L + 14 x N if asymmetric duplex with fixed capacity agreement and with FEC

Total length = 6 + L + 6 x N if asymmetric duplex with basic allocation and with FEC

Total length = 6 + L + 7 x N if symmetric duplex with fixed capacity agreement and with FEC

Total length = 6 + L + 3 x N if symmetric duplex with basic allocation and with FEC

Total length = 6 + L + 3 x N if simplex with basic allocation and with FEC

Total length = 6 + L + 7 x N if simplex with fixed capacity agreement and with FEC

Total length = 6 + L + 12 x N if asymmetric duplex with fixed capacity agreement and without FEC

Total length = 6 + L + 4 x N if asymmetric duplex with basic allocation and without FEC

Total length = 6 + L + 6 x N if symmetric duplex with fixed capacity agreement and without FEC

Total length = 6 + L + 2 x N if symmetric duplex with basic allocation and without FEC

Total length = 6 + L + 2 x N if simplex with basic allocation and without FEC

Total length = 6 + L + 6 x N if simplex with fixed capacity agreement and without FEC

Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In the following the total length of the PDU is given for the unacknowledged mode:

Total length = 6 + L + 18 x N if asymmetric duplex with FSA and with FEC

Total length = 6 + L + 9 x N if symmetric duplex with FSA and with FEC

Total length = 6 + L + 9 x N if simplex with FSA and with FEC

Total length = 6 + L + 16 x N if asymmetric duplex with FSA and without FEC

Total length = 6 + L + 8 x N if symmetric duplex with FSA and without FEC

Total length = 6 + L + 8 x N if simplex with FSA and without FEC

A.4.7 RLC-MODIFY encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use	CL-CONN-ATTR-LENGTH L)					# of DUC:s	
Octet 5	# of DUC:s (N)	Future use						
Octet 6	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet 6 + L								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

Total length = 6 + L + 14 x N if asymmetric duplex with fixed capacity agreement and with FEC

Total length = 6 + L + 6 x N if asymmetric duplex with basic allocation and with FEC

Total length = 6 + L + 7 x N if symmetric duplex with fixed capacity agreement and with FEC

Total length = 6 + L + 3 x N if symmetric duplex with basic allocation and with FEC

Total length = 6 + L + 3 x N if simplex with basic allocation and with FEC

Total length = 6 + L + 7 x N if simplex with fixed capacity agreement and with FEC

Total length = 6 + L + 12 x N if asymmetric duplex with fixed capacity agreement and without FEC

Total length = 6 + L + 4 x N if asymmetric duplex with basic allocation and without FEC

Total length = 6 + L + 6 x N if symmetric duplex with fixed capacity agreement and without FEC

Total length = 6 + L + 2 x N if symmetric duplex with basic allocation and without FEC

Total length = 6 + L + 2 x N if simplex with basic allocation and without FEC

Total length = 6 + L + 6 x N if simplex with fixed capacity agreement and without FEC

Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In the following the total length of the PDU is given for the unacknowledged mode:

Total length = 6 + L + 18 x N if asymmetric duplex with FSA and with FEC

Total length = 6 + L + 9 x N if symmetric duplex with FSA and with FEC

Total length = 6 + L + 9 x N if simplex with FSA and with FEC

Total length = 6 + L + 16 x N if asymmetric duplex with FSA and without FEC

Total length = 6 + L + 8 x N if symmetric duplex with FSA and without FEC

Total length = 6 + L + 8 x N if simplex with FSA and without FEC

A.4.8 RLC-MODIFY-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	Future use	CL-CONN-ATTR-LENGTH (L)					# of DLCC+CL-CON-ATT	
Octet 5	# of DLCC+CL-CON-ATT (N)		Future use					
Octet ...	Future use		DLCC-ID-1					
Octet ...	CL-CONN-ATTR-1							
Octet ...								
Octet ...	Future use		DLCC-ID-2					
Octet ...	CL-CONN-ATTR-2							
Octet 5 + (L + 1) x N								
Octet …	Not used							
Octet …								
Octet 51								

Total length = 5 + (L + 1) x N

A.4.9 RLC-RESET, RLC-RESET-ACK encoding

	8	7	6	5	4	3	2	1
Octet 4	Future use				# of DLCC:s (N)			
Octet 5	Future use		DLCC-ID-1					
Octet 6	Future use		DLCC-ID…					
Octet 4 + N	Future use		...					
Octet …	Not used							
Octet …								
Octet 51								

A.4.10 RLC-DM-SETUP encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CL-ID							
Octet 6	DUC-EXT-IND	CL-CONN-ATTR-LENGTH (L)					CL-COMMON-ATTR-LENGTH	
Octet 7	CL-COMMON-ATTR-LENGTH			Future use	# of DUC:s(N)			
Octet 8	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME(LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	CL-COMMON-ATTR							
Octet ...								
Octet ...								
Octet ...	Not used							
Octet ...								
Octet ...								
Octet 51								

Y = 8 + L

X = 8 + L + 6 if asymmetric duplex with polling and ARQ or FEC

A.4.11 RLC-DM-CONNECT encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CL-ID							
Octet 6	Future use	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 7	# of DUC:s (N)			Future use				
Octet 8	DUC1-DIRECTION			DLCC-ID				
	CL-CONN-ATTR							
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

A.4.12 RLC-DM-CONNECT-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CL-ID							
Octet 6	Future use	CL-CONN-ATTR-LENGTH (L)					# of DLCC:s+CL-ATTR	
Octet 7	# of DLCCs (N)		Future use					
Octet 8	Future use		DLCC-ID-1					
Octet …	CL-CONN-ATTR-1							
Octet …	Future use		DLCC-ID-N					
Octet …	CL-CONN-ATTR-N							
Octet 8 + L x N								
Octet …	Not used							
Octet …								
Octet 51								

A.4.13 RLC-DM-CONNECT-COMPLETE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use				# of DLCC:s (N)			
Octet 6	Future use		DLCC-ID-1					
Octet …	Future use		...					
Octet 5 + N	Future use		DLCC-ID-N					
Octet 6 + N	Not used							
Octet …								
Octet 51								

A.4.14 RLC-DM-RELAY-SETUP encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CL-ID							
Octet 6	DUC-EXT-IND	CL-CONN-ATTR-LENGTH (L)					CL-COMMON-ATTR-LENGTH	
Octet 7	CL-COMMON-ATTR-LENGTH			future	# of DUC:s(N)			
Octet 8	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
...	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	CL-COMMON-ATTR							
Octet ...								
Octet ...								
Octet ...	Not used							
Octet ...								
Octet 51								

A.4.15 RLC-DM-RELAY-SETUP-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use	CL-CONN-ATTR-LENGTH					# of DLCC:s	
Octet 6	# of DLCCs		Future use					
Octet 7	Future use		DLCC-ID-1					
Octet …	CL-CONN-ATTR-1							
Octet …	Future use		DLCC-ID					
Octet …	CL-CONN-ATTR							
Octet …	Not used							
Octet …								
Octet 51								

A.4.16 RLC-DM-MODIFY-REQ encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 6	# of DUC:s (N)		Future use					
Octet 7	DUC1-DIRECTION		DLCC-ID					
	CL-CONN-ATTR							
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet …	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet …	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet …	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet …	REQUESTED-NUM-OF-LCH							
Octet …	MINUMUM-NUM-OF-LCH							
Octet …	Future use				PHY-MODE-LCH			
Octet …	NB-OF-LCH							
Octet …	MIN-NB-OF-LCH							
Octet …	start-pointer							
Octet …	start-pointer					Future use		
Octet …	repetition-counter							
Octet …	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet …	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet …	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet …	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet …	REQUESTED-NUM-OF-LCH							
Octet …	MINUMUM-NUM-OF-LCH							
Octet …	Future use				PHY-MODE-LCH			
Octet …	NB-OF-LCH							
Octet …	MIN-NB-OF-LCH							
Octet …	start-pointer							
Octet …	start-pointer					Future use		
Octet …	repetition-counter							
Octet …	repetition-counter				frame-count			
Octet …	Not used							
Octet …								
Octet 51								

A.4.17 RLC-DM-MODIFY encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 6	# of DUC:s (N)	Future use						
Octet 7	DUC1-DIRECTION	DLCC-ID						
	CL-CONN-ATTR							
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME (SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

A.4.18 RLC-DM-MODIFY-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CL-CONN-ATTR-LENGTH					# of DLCC:s+CL-CONN-ATTR		
Octet 6	# of DLCCs	Future use						
Octet 7	Future use	DLCC-ID-1						
Octet …	CL-CONN-ATTR-1							
…	Future use	DLCC-ID…						
…	CL-CONN-ATTR …							
…	Not used							
…								
Octet 51								

A.4.19 RLC-DM-MODIFY-COMPLETE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use				# of DLCC:s			
Octet 6	Future use	DLCC-ID-1						
Octet 7	Future use	DLCC-ID…						
Octet …	Not used							
Octet …								
Octet 51								

A.4.20 RLC-DM-RELAY-MODIFY encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use	CL-CONN-ATTR-LENGTH (L)					# of DUC:s	
Octet 6	# of DUC:s (N)	Future use						
Octet 7	DUC1-DIRECTION	DLCC-ID						
...	CL-CONN-ATTR							
...								
Octet Y	DUC1-FW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-FW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-FW-ARQ-WIN-SIZE		
...	FEC-FW				Future Use			
Octet ...	PER-#-MAC-FRAME(SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet X	DUC1-BW-TYPE-OF-ALLOCATION			Future use	CYCLIC-PREFIX	FEC-USED	EC-MODE	
Octet ...	DUC1-BW-NUM-OF-RETRANSMISSIONS				Future use	DUC1-BW-ARQ-WIN-SIZE		
...	FEC-BW				Future Use			
Octet ...	PER-#-MAC-FRAME(SCH)				PER-#-MAC-FRAME (LCH)			
Octet ...	REQ SCH	PHY-MODE-SCH			PHY-MODE-LCH			
Octet ...	REQUESTED-NUM-OF-LCH							
Octet ...	MINUMUM-NUM-OF-LCH							
Octet ...	Future use				PHY-MODE-LCH			
Octet ...	NB-OF-LCH							
Octet ...	MIN-NB-OF-LCH							
Octet ...	start-pointer							
Octet ...	start-pointer					Future use		
Octet ...	repetition-counter							
Octet ...	repetition-counter				frame-count			
Octet ...	Not used							
Octet ...								
Octet 51								

A.4.21 RLC-DM-RELAY-MODIFY-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use	CL-CONN-ATTR-LENGTH					# of DLCC:s	
Octet 6	# of DLCCs	Future use						
Octet 7	Future use	DLCC-ID-1						
Octet …	CL-CONN-ATTR-1							
Octet …	Future use	DLCC-ID-2						
Octet …	CL-CONN-ATTR-2							
Octet …	Not used							
Octet …								
Octet 51								

A.4.22 RLC-DM-RELEASE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CAUSE				# of DLCC:s			
Octet 6	Future use		DLCC-ID-1					
Octet 7	Future use		…					
Octet …	Future use		DLCC-IDN					
Octet …	Not used							
Octet …								
Octet 51								

A.4.23 RLC-DM-RELEASE-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use				# of DLCC:s			
Octet 6	Future use		DLCC-ID-1					
Octet 7	Future use		…					
Octet …	Future use		DLCC-ID-N					
Octet …	Not used							
Octet …								
Octet 51								

A.4.24 RLC-DM-RELAY-RELEASE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	CAUSE				# of DLCC:s			
Octet 6	Future use		DLCC-ID-1					
Octet 7	Future use		…					
Octet …	Future use		DLCC-ID-N					
Octet …	Not used							
Octet …								
Octet 51								

A.4.25 RLC-DM-RELAY-RELEASE-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use				# of DLCC:s			
Octet 6	Future use		DLCC-ID-1					
Octet 7	Future use		…					
Octet …	Future use		DLCC-IDN					
Octet …	Not used							
Octet …								
Octet 51								

A.4.26 RLC-DM-RESET, RLC-DM-RESET-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 4	MAC-ID							
Octet 5	Future use				# of DLCC:s			
Octet 6	Future use		DLCC-ID-1					
Octet 7	Future use		DLCC-ID…					
Octet …	Future use		...					
Octet …	Not used							
Octet …								
Octet 53								

A.5 Transfer Syntax Tables for SCH ACF messages

A.5.1 RLC-RBCH-ASSOCIATION-REQUEST encoding (OMT)

	8	7	6	5	4	3	2	1
Octet 3							AP-ID	
Octet 4	AP-ID							
Octet 5	Future use						NET-ID	
Octet 6	NET-ID							
Octet 7	MAC-ID							

A.5.2 RLC-MAC-ID-ASSIGN encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	MAGIC							
Octet 5								
Octet 6	RLC-VERSION							
Octet 7	MAC-ID							

A.5.3 RLC-MAC-ID-ASSIGN-ACK encoding

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet 4	MAGIC							
Octet 5								
Octet 6	MAC-ID							
Octet 7	MAC-ID1							

A.5.4 RLC-MAC-ID-ASSIGN-NACK encoding

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet 4	MAGIC							
Octet 5								
Octet 6	Future use							
Octet 7								

A.5.5 RLC- RLC-COMMON-KEY-ACTIVATE encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet 4	KEY-ID							
Octet 5	LAST-MAC-FRAME							
Octet 6								
Octet 7	Not used							

A.5.6 RLC-DISASSOCIATION encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use				DISASSOCIATION-CAUSE			
Octet 5	Future use							
Octet 6								
Octet 7	MAC-ID (if sent in uplink)							

A.5.7 RLC-DISASSOCIATION-ACK encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use							
Octet 5								
Octet 6								
Octet 7	MAC-ID (if sent in uplink)							

A.5.8 RLC-PROCEEDING encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	SCH-LCH
Octet 4	Proceeded-PDU-type							
Octet 5	Future use					EXTENSION-TYPE		
Octet 6	Future use							
Octet 7	MAC-ID (if sent in uplink)							

A.5.9 RLC-UNICAST-KEY-ACTIVATE encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet 4	LAST-MAC-FRAME							
Octet 5								
Octet 6	Not used							
Octet 7								

A.6 Transfer Syntax Tables for SCH RRC messages

A.6.1 RLC-SECTOR-HANDOVER-REQUEST encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future Use	
Octet 4	Future use					SECTOR-ID		
Octet 5	Future use							
Octet 6								
Octet 7	MAC-ID							

A.6.2 RLC-SECTOR-HANDOVER-ACK encoding (OAP/OMT)

Empty PDU

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet ...	Future use							
Octet 7								

A.6.3 RLC-HANDOVER-NOTIFY encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							AP-IDpr	NET-IDpr
Octet 4	HANDOVER-CAUSE				AP-ID			
Octet 5	AP-ID						NET-ID	
Octet 6	NET-ID							
Octet 7	MAC-ID							

A.6.4 RLC-HANDOVER-REQUEST encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							AP-ID-OLD	
Octet 4	AP-ID-OLD AP-ID-OLD							
Octet 5	MAC-ID-OLD NET-ID-OLD							
Octet 6	NET-ID-OLD NET-ID-OLD							
Octet 7	NET-ID-OLD		DUC-EST	Future use				

A.6.5 RLC-HANDOVER-REQUEST-NACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3				Future use AP-ID-OLD				
Octet 4	MAC-ID-OLD							
Octet 5	Future use				AP-ID-OLD			
Octet 6	AP-ID-OLD						NET-ID-OLD	
Octet 7	NET-ID-OLD							

A.6.6 RLC-HO-INFO-DISTRIBUTION-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use							
Octet 5								
Octet 6								
Octet 7	MAC-ID							

A.6.7 RLC-FORCE-HANDOVER encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3				Return	Future use	Cause:TrafLoad/Badlink/Operator Badlink Operator		
Octet 4	FREQUENCY-INDEX NET-ID							
Octet 5	AP-ID AP-ID							
Octet 6	AP-ID		Future use				NET-ID	
Octet 7	NET-ID Future use							

A.6.8 RLC-FORCE-HANDOVER-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use							
Octet 5								
Octet 6								
Octet 7	MAC-ID							

A.6.9 RLC-AP-ABSENCE encoding (OAP)

	8	7	6	5	4	3	2	1
Octet 3				Future use	FIRST-MAC-FRAME			
Octet 4	LAST-MAC-FRAME							
Octet 5								
Octet 6	Future use							
Octet 7								

A.6.10 RLC-DFS-MT-INIT-REPORT-REQUEST encoding (OMT)

	8	7	6	5	4	3	2	1
Octet 3							MEASUREMENT-TYPE	
Octet 4	FREQUENCY-INDEX FREQUENCY-INDEX							
Octet 5	Future use						ADJ-CH-INT	
Octet 6	Future use							
Octet 7	MAC-ID							

A.6.11 RLC-DFS-MT-INIT-REPORT-REQUEST-ACK encoding (OMT)

	8	7	6	5	4	3	2	1
Octet 3				Future use				REP-INI
Octet 4	Future use							
Octet …								
Octet 7								

A.6.12 RLC-CHANGE-FREQUENCY encoding

	8	7	6	5	4	3	2	1
Octet 3				Future use	FIRST-MAC-FRAME			
Octet 4	LAST-MAC FRAME							
Octet 5								
Octet 6	FREQUENCY-INDEX							
Octet 7	Future use							

A.6.13 RLC-UPLINK-PC-CALIBRATION encoding

	8	7	6	5	4	3	2	1
Octet 3				Future use		PC-OFFSET		
Octet ...	Future use							
Octet 7								

A.6.14 RLC-MT-ALIVE-REQUEST encoding

	8	7	6	5	4	3	2	1
Octet 3				Future use		NO-OF-MT-ALIVE-PROCEDURES		
Octet 4	MT-ALIVE-INTERVAL							
Octet 5								
Octet 6								
Octet 7	Future use							

A.6.15 RLC-MT-ALIVE-REQUEST-ACK encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use							
Octet 5								
Octet 6								
Octet 7	MAC-ID							

A.6.16 RLC-MT-ALIVE encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use							
Octet 5								
Octet 6								
Octet 7	MAC-ID							

A.6.17 RLC-MT-ALIVE-ACK encoding

Empty PDU

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet ...	Future use							
Octet 7								

A.6.18 RLC-MT-ABSENCE encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	Future use		MT-ABSENCE-TIME					
Octet 5	Future use							
Octet 6								
Octet 7	MAC-ID							

A.6.19 RLC-MT-ABSENCE-ACK encoding (OAP/OMT)

Empty PDU

	8	7	6	5	4	3	2	1
Octet 3				Future use				
Octet ...	Future use							
Octet 7								

A.6.20 RLC-SLEEP encoding (OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future use	care-of-bc
Octet 4	Future use				SLEEP-GROUP			
Octet 5	Future use							
Octet 6								
Octet 7	MAC-ID							

A.6.21 RLC-SLEEP-ACK encoding (OMT)

	8	7	6	5	4	3	2	1
Octet 3				Future use				care-of-bc
Octet 4	Future use				SLEEP-GROUP			
Octet 5	OFFSET							
Octet 6								
Octet 7	Future use							

A.7 Transfer Syntax Tables for SCH DUCC messages

A.7.1 RLC-DM-MODIFY-COMPLETE-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	MAC-ID							
Octet 5	Future use							
Octet 6								
Octet 7	MAC-ID							

A.7.2 RLC-DM-CONNECT-COMPLETE-ACK encoding (OAP/OMT)

	8	7	6	5	4	3	2	1
Octet 3							Future use	
Octet 4	MAC-ID							
Octet 5	Future use							
Octet 6								
Octet 7	MAC-ID							

A.8 Transfer Syntax Tables for other RLC SCH messages

A.8.1 RLC-NO-SUPPORT encoding

	8	7	6	5	4	3	2	1
Octet 3							Future use	SCH-LCH
Octet 4	Not-supported-PDU-type							
Octet 5	Future use					EXTENSION-TYPE		
Octet 6	Future use Future use							
Octet 7	MAC-ID (if Uplink)							

Annex B (normative): Types

The implementations shall follow the numbering (0, 1, …) in the enumerated lists.

Table B.1: Data description

ADJACENT-CH-INTERFERENCE ::= ENUMERATED { channel-inteference-0 (0) }	field size 2 bits.
AGE-OF-MEASUREMENT ::= INTEGER(0..4095)	In request: The maximum age for the last BCH measurement. If the BCH on the requested frequency has been measured within this time for other reasons, e.g. handover, the MT is not required to perform the BCH decoding. Instead, the MT may use stored values of the BCH content and the RSS measured on the BCH. In report: Indicates the time since the measurement was performed. This is important if the AP polls the MT for measurement results. *Age* is the number of MAC frames since the measurement was finished. *Age* = 0 means that the measurement was performed in the previous MAC frame. The accuracy ± 2 MAC frames is allowed for the value. The backoff value due to collisions in the RACH shall not be taken into account. 12 bit field.
ALLOCATION-TYPE ::= ENUMERATED { basic (0) fca (1) fsa (2) }	3 bits. Fixed Capacity Agreement. Fixed Slot Allocation.
AP-ID ::= INTEGER (0 .. 1023)	Access Point Identifier [5]. AP-ID value 0 for future use. 10 bit field.
APT-ADDRESS-LENGTH ::= INTEGER (0..10)	4 bits field. Indicates the number of bits assigned for transceiver identification starting from the least significant bit of the AP ID. In case of single transceiver Aps the *apt-address-length* shall be set to zero. Values of *apt-address-length* greater than zero shall be indicated only, when the AP supports radio handover. In a single coverage area using Aps with identical *net-id* the *apt-address-length* shall be set to the same value in all Aps.
AP-TOKEN-AUTH-ENCR::= SEQUENCE { token TOKEN authentication-encryption-list AUTHENTICATION-ENCRYPTION-LIST auth-encr-selected AUTH-ENCR-INFO }	
ARQ-DATA ::= SEQUENCE { arq-nr-of-retr INTEGER (0..15) arq-window-size WINDOW-SIZE }	 4 bit field. 3 bit field.
ARQ-DELAY ::= INTEGER (0..3)	ARQ Delay Class. See [5]. 3 bit field.
AUTH-ENCR-INFO ::= SEQUENCE { auth-info AUTH-INFO encr-info ENCR-INFO }	
AUTHENTICATION-ENCRYPTION-LIST ::= SEQUENCE (SIZE(1..15)) OF AUTH-ENCR-INFO	The list is ordered by preference. The first item has the highest preference.

AUTH-INFO ::= ENUMERATED { no-authentication (0) pre-shared-key-based (1) signature-based-512 (2) signature-based-768 (3) signature-based-1 024 (4)}	4 bits field size.
AUTH-RESPONSE-PART1 ::= OCTET STRING (SIZE(16..32))	Only the values 16 and 32 are used. See CHALLENGE.
AUTH-RESPONSE-PART2 ::= OCTET STRING (SIZE(16..48))	Only the values 16, 32 and 48 are used. See CHALLENGE.
BCH-FOUND ::= ENUMERATED { bch-not-found (0) bch-found (1) }	field size 1 bit.
BCH-RX-LEVEL ::= INTEGER(0..63)	Measured signal strength of the BCH on frequency *f*. It is an index to a signal strength [4]. 6 bit field.
BROAD-WINDOW ::= ENUMERATED { bw-size32 (0) bw-size64 (1) bw-size128 (2) bw-size256 (3)}	2 bits. Window size for repeated broadcast PDUs, see [5].
CARE-OF-BROADCAST ::= ENUMERATED { dont-care-of-broadcast (0) care-of-broadcast (1)}	field size 1 bit.
CC-HO-CAP ::= ENUMERATED { cc-ho-not-supported (0) cc-ho-supported (1) }	field size 1 bit.
CHALLENGE ::= OCTET STRING (SIZE (16))	Used in the authentication procedure. A random number sent to the other party that calculates a response according to the authentication procedure, with the challenge as an input.
CL-ATTRIBUTES ::= OCTET STRING (SIZE(0..44))	The convergence layers exchange information between themselves, transparent to RLC.
CL-COMMON-ATTR ::= OCTET STRING (SIZE(0..31))	
CL-CONN-ATTR ::= OCTET STRING (SIZE(0..31))	
CL-DATA ::= SEQUENCE { cl-id CL-ID cl-attributes CL-ATTRIBUTES}	
CL-ID ::= ENUMERATED { }	8 bit field size. To be defined.
CL-VERSION ::= INTEGER(0..255)	Both MT and AP send their own version in Link Capability procedure. 8 bit field.
cMAX-DESCR-LIST INTEGER ::= 16	
cMAX-ID-LIST INTEGER ::= 16	
CODER-TYPE ::= ENUMERATED { reed-solomon-216-200 (0)}	8.2.1.1.1 Field size: 2 bits Reed-Solomon code.
COMMON-KEY ::= CHOICE { no-encr [0] NULL des-encr [1] OCTET STRING(SIZE(8)) tripledes [2] OCTET STRING(SIZE(24))}	A key used to encrypt multicast and/or broadcast traffic.
C-U-G ::= ENUMERATED { open-user-group (0) closed-user-group (1) }	Open group: All MTs allowed to attempt association. Closed group: Only MTs with a matching network operator identifier allowed to attempt association. 1 bit field.
CYCLIC-PREFIX ::= ENUMERATED { t400ns (0) t800ns (1)}	800 ns in mandatory, 400 ns is optional [4]. 1 bit field.
DH-PUBLIC-VALUE-HALF ::= OCTET STRING (SIZE(48))	DH = Diffie-Hellman. Used to create encryption key.

DIRECTION ::= ENUMERATED { simplex-forward (0) simplex-backward (1), duplex (2), duplex-symetric (3)}	2 bit field. simplex connection, forward direction. simplex connection, backward direction. duplex connection. duplex-symetric - the same data used for both directions.
DIRECT-MODE-CAP ::= ENUMERATED { no-dm-capabilities (0) dm-capabilities (1)}	1 bit field size.
DISASSOCIATION-CAUSE ::= ENUMERATED { unknown-dis-cause (0) user-disassociation (1) operator-disassociation (2) low-qos-dis (3) traffic-overload-dis (4) authentication-failed (5) mt-powerdown (6) ap-powerdown (7) mismatched-resources (8)}	4 bits. A "user" can be any "user" of the system, both at the MT and the AP.
DIL-POWER-CONTROL ::= ENUMERATED { dil-fixed-pc (0) dil-dynamic-pc (1) }	- 2 bits. - Fixed power = Max power- 3dB. - Dynamic dil power control as defined in HE.
DLC-ATTRIBUTES ::= OCTET STRING (SIZE(0..44))	The DLC layers exchange information between themselves, transparent to RLC.
DLC-ATTR-PR ::= ENUMERATED { dlc-attr-not-present (0) dlc-attr-present (1)}	field size 1 bit.
DLCC-DESCR ::= SEQUENCE { dlcc-id DLCC-ID cl-conn-attr CL-CONN-ATTR}	
DLCC-DESCR-LIST ::= SEQUENCE (SIZE(1..16)) OF DLCC-DESCR	The maximum number of elements is limited by the LCH-PDU length and the number of other parameters used in the same RLC PDU.
DLCC-ID ::= INTEGER (0 .. 63)	DLC Connection Identifier [5]. 6 bit field.
DLCC-ID-LIST ::= SEQUENCE (SIZE(1..cMAX-ID-LIST)) OF DLCC-ID	The maximum number of elements is limited by the LCH-PDU length and the number of other parameters used in the same RLC PDU.
DM-ATTRIBUTES ::= SEQUENCE { dil-power-control DIL-POWER-CONTROL tx-arq-win-size WINDOW-SIZE rx-arq-win-size WINDOW-SIZE }	A minimum ARQ window size shall be negotiated here. The DM-attribute is used by home extension to negotiate some specific DM parameters.
DM-USE-COMMON-KEY ::= ENUMERATED { no-common-key (0) use-common-key (1)}	1 bit.
DUC-DESCR ::= SEQUENCE { direction [0] DIRECTION dlcc-id [1] DLCC-ID cl-conn-attr [2] CL-CONN-ATTR forward-descr [3] DUC-DIRECTION-DESCR-FW OPTIONAL backward-descr [4] DUC-DIRECTION-DESCR-BW OPTIONAL}	*forward-descr* shall be used, when *direction* indicates *simplex_forward*, *duplex* or *duplex_symmetric*. *backward-descr* shall used, when *direction* indicates *simplex_backward*, *duplex*.
DUC-DESCR-LIST ::= SEQUENCE (SIZE(1..cMAX-DESCR-LIST)) OF DUC-DESCR	The maximum number of elements is limited by the LCH-PDU length and the number of other parameters used in the same RLC PDU.
DUC-DIRECTION-DESCR ::= SEQUENCE { allocation-type [0] ALLOCATION-TYPE cyclic-prefix [1] CYCLIC-PREFIX fec-used [2] FEC-USED ec-mode [3] EC-MODE arq-data [4] ARQ-DATA OPTIONAL fec [5] FEC-DESCR OPTIONAL fca-descr [6] FCA-DESCR OPTIONAL fsa-descr [7] FSA-DESCR OPTIONAL }	*fec* shall be present, *when fec-used* is set. *arq-data* shall be present, *when ec-mode* is set to *acknowledged-mode*. *fca-descr* shall be present, when *allocation-type* indicates *fca, fsa-descr* shall be present, *when* allocation-type indicates *fsa*.
DUC-DIRECTION-DESCR-BW ::= DUC-DIRECTION-DESCR	DUC description to be used for the backward direction.

DUC-DIRECTION-DESCR-FW ::= DUC-DIRECTION-DESCR	DUC description to be used for the forward direction.
DUC-ESTABLISHED ::= ENUMERATED { no-duc-established (0) ducs-established (1)}	field size 1 bit. Indicates, if the MT maintains on-going unicast DUCs. This parameter shall be considered by the target AP during network handover when re-establishing on-going DUCs.
DUC-EXT-IND ::= ENUMERATED { no-duc-ext (0) duc-ext (1)}	field size 1 bit.
DUTY-CYCLE ::= ENUMERATED { fiveper (0) tenper (1) twentyper (2) thirtyper (3) fortyper (4) sixtyper (5) eightyper (6) hundredper (7)}	Percent of the MAC frame that the MT can use for uplink transmission. Upper limit, which AP may take into account. 5 % 10 % 20 % 30 % 40 % 60 % 80 % 100 % 3 bit field
EC-MODE ::= ENUMERATED { arq-not-used (0) arq-used (1) repetition-mode (2)}	field size 2 bits.
ENCR-INFO ::= ENUMERATED { no-encryption (0) des (1) tripleDES (2)}	4 bits field size.
ENCRYPTION-ALGORITHM-PROPOSAL ::= SEQUENCE (SIZE(1..15)) OF ENCR-INFO	The list is ordered by preference. The first item has the highest preference.
ERROR-CORR-MODE ::= ENUMERATED { repetition-mode (0) unacknowledged-mode (1)}	1 bit field size, used in CL-BROADCAST-JOIN-ACK message.
EXTENSION-TYPE ::= ENUMERATED { basic-rlc (0) home-extension (1) business-extension (2) }	3 bits field size. This field is set to every RLC PDU. The usage of the field allows re-usage of the RLC PDU TYPE field for different extensions. This field shall be encoded to 0 to indicate the PDUs defined in the present document.
FCA-DESCR ::= SEQUENCE { nb-of-sch INTEGER (0..1) sch-per-nb-frames INTEGER (1..15) lch-per-nb-frames INTEGER (1..15) phy-mode-sch PHY-MODE-SCH phy-mode-lch PHY-MODE-LCH nb-of-lch INTEGER (0..255) min-nb-of-lch INTEGER (0..255)}	Nb = number. 1 bit field. 4 bit field. 4 bit field. 8 bit field. 8 bit field.
FEC-DESCR ::= SEQUENCE { coder-type CODER-TYPE, interleaver-type INTERLEAVER-TYPE}	
FEC-USED ::= ENUMERATED { fec-not-used (0) fec-used (1)}	field size 1 bit.
FIRST-MAC-FRAME ::= INTEGER(0..15)	RLC_CHANGE_FREQUENCY: Index to the first frame transmitted on new frequency. RLC_AP_ABSENCE: The first frame where AP will transmit again after AP absence. The reference for the number is the MAC frame where the AP stops transmitting. 0 denotes the MAC frame immediately after the MAC frame where the AP stops transmitting. 4 bit field.

FORCE-HANDOVER-CAUSE ::= ENUMERATED { unspec-fho-cause (0) traffic-overload (1) bad-link (2) operator-action (3) cell-closure (4) mt-behaviour (5) qos-not-achived (6)}	3 bits field size.
FRAME-COUNTER ::= INTEGER (0..15)	4-bit Frame counter.
FRAME-NUM ::= INTEGER (0..65535)	Integer value with unit FRAMES (16 bit).
FREQUENCY-BAND ::= ENUMERATED { lower-band-only (0) upper-band-only (1) lower-and-upper-band (2)}	Field size 2 bits. A node declares which frequency band that it can use. Low band: 5 180-5 320 MHz. High band: 5 500-5 700 MHz.
FREQUENCY-INDEX ::= INTEGER (1..255)	Field size 8 bits. The value points to the nominal carrier, see [4].
FSA-DESCR ::= SEQUENCE { phy-mode-lch PHY-MODE-LCH nb-of-lch INTEGER (0..255), --8 bit field min-nb-of-lch INTEGER (0..255), --8 bit field start-pointer START-POINTER, start-mac-frame START-MAC-FRAME }	Fixed Slot Allocation (FSA) may be used in any EC mode and with or without FEC. When using unacknowledged mode, a MT does not have to decode the FCCH once the connection is set up. When using acknowledged mode, a MT shall decode the FCCH, because SCHs for acknowledgements and discards are granted in basic allocation mode. Only LCHs can be allocated with FSA. In case of an FSA-RG the AP/CC shall set the min-nb-of-lch to the same value as nb-of-lch.
GROUP-MAC-ID ::= MAC-ID (224..255)	MAC ID used for a multicast group. The value 255 shall be used for overflow multicast traffic, that is, when the values 224-254 are used up.
HANDOVER-CAUSE ::= ENUMERATED { unspec-ho-cause (0) link-quality (1) traffic-related (2) network-related (3) }	field size 4 bits. Defines the reason why an MT performs a handover.
IDENTIFIER-FORMAT ::= ENUMERATED { network-id-available (0) network-id-unavailable (1) }	3 bits field size. Two values are used at present. One value for network operator available and the other value for no network operator available.
IDENTIFYER ::= CHOICE { empty NULL full OP-ID}	
INFO-COUNT ::= INTEGER (0..7)	
INFO-TYPE ::= ENUMERATED { new-info (0), retrans-info (1)}	field size 1 bit.
INTERLEAVER-TYPE ::= ENUMERATED { no-interleaver (0) three-branch-conv (1)}	field size 2 bits.
KEEP-CONNECTIONS ::= ENUMERATED { donot-keep-conn (0) keep-connections (1) }	field size 1 bit.
KEY-ID ::= INTEGER (0..255)	Identifier for a common encryption key. A key can be used in different places and to save resources a short identifier is used instead of the key itself. 8 bit field.
LAST-MAC-FRAME ::= INTEGER(0..65535)	16-bit field size. RLC_CHANGE_FREQUENCY: Index to the last transmitted frame on old frequency RLC_AP_ABSENCE: The last frame that the MT transmits at before AP Absence. Reference is the MAC frame in which the message from AP is received by the MT. The value 0 shall mean the first MAC frame after the one in which the message was received by the MT.

LAST-OWN-BCH-RX-LEVEL ::= BCH-RX-LEVEL	Measurement result: RSS on the used frequency BCH.
MAC-ID ::= INTEGER (0 .. 255)	8 bits Identifier used in communication between MT and AP/CC or another MT [5].
MAC-ID0 ::= MAC-ID(0)	A subtype of MAC-ID. A fixed value 0 used for upstream communication before an MT has got a MAC-ID of its own (RLC_MAC_ID_ASSIGN and RLC_RBCH_ASSOCIATION_REQUEST).
MAC-ID-AND-CL-DATA ::= SEQUENCE { mac-id-choice MAC-ID-CHOICE cl-data CL-DATA }	
MAC-ID-AND-CL-DATA-LIST ::= SEQUENCE (SIZE(1..7)) OF MAC-ID-AND-CL-DATA	
MAC-ID-CHOICE ::= CHOICE { group-mac-id [0] GROUP-MAC-ID, unicast-mac-id [1] MAC-ID, broadcast-mac-id [2] MAC-ID}	
MAGIC ::= INTEGER (0..63535)	Random number, 16 bits. The same magic number shall be kept during the retransmissions of the messages that use it. It is used as a temporary identifier until MT has got its own MAC-ID.
MAXIMUM-AGE-OF-BCH-MEASUREMENT ::= AGE-OF-MEASUREMENT	Sort: Number of MAC frames.
MD5-ON-KEY ::= OCTET STRING (SIZE(16))	The MD5 algorithm operating on key.
MD5-ON-NONCE ::= OCTET STRING (SIZE(16))	The MD5 algorithm operating on nonce.
MEASUREMENT-TYPE ::= ENUMERATED { type-a (0) type-b (1) type-c (2) type-u (3)}	field size 2 bits. The measurement types a, b, c are described in MEASUREMENT_REQUEST_MESSAGEs. Measurement type u (undefined) means that the MT has performed a measurement on the specified frequency, which does not follow any of the predefined measurement types. In this case, it is up to the AP to request necessary measurements from the MT. The adjacent channel flag indicates if the measurement concerns adjacent channel interference or not.
MEASUREMENT-WINDOW ::= INTEGER (0..63)	6 bit field size. On other frequency measurement window defines how many MAC-frames time units shall be spent on measurements. If the measurement type is percentile this time is spent on RSS statistics measurements. If measurement type is short the measurement window is 5 frames and the RSS from the strongest BCH obtained is reported together with the BCH-content. If the type is complete 5 frames of the window is used for BCH-synch and decode and the rest spent on RSS statistics measurements. On used frequency measurement-window gives a coarse description of the measurement interval and the final description is given by the AP-absence message or the FCH- empty-part of frame information.
MORE-AUTH ::= ENUMERATED { more-auth-pdu (1) last-auth-pdu (0)}	field size 1 bit. Indicates whether more PDUs are to follow or not. Used when authentication information is longer than what can be contained in one PDU.
MORE-JOINS ::= ENUMERATED { no-more-joins (0) more-joins (1) }	field size 1 bit.
MT-ABSENCE-TIME ::= INTEGER (0..63)	Defines the absence period of the MT in MAC frames. 6 bit field.
MT-ALIVE-INTERVAL ::= INTEGER (0..1677215)	Period (in number of frames) that MT Alive procedure is commanded to be triggered in.
MT-AUTH-CONTENT ::= CHOICE {; ieee [0] OCTET STRING (SIZE(6)) ext-ieee [1] OCTET STRING (SIZE(8)) net-acc-id [2] OCTET STRING (SIZE(1..46)) dist-name [3] OCTET STRING (SIZE(1..46)) compressed [4] OCTET STRING (SIZE(16)), generic [5] OCTET STRING (SIZE(1..46))}	Type of MT authentication identifier. The compressed type can be used if the available authentication key identifier is so long that it is not possible to carry in the defined RLC messages. The compressed authentication key identifier is calculated as follows: compressed-authentication-key-identifier =

} 	MD5(available_authentication_key_identifier). The generic type is a non-structured octet string.

MT-AUTH-ID-TYPE ::= ENUMERATED {
ieee (0)
ext-ieee (1)
net-acc-id (2)
dist-name (3)
compressed (4)
generic (5) }
}
	Type of MT authentication identifier.
	4 bit field.
	The compressed type can be used if the available authentication key identifier is so long that it is not possible to carry in the defined RLC messages. The compressed authentication key identifier is calculated as follows: compressed-authentication-key-identifier = MD5(available_authentication_key_identifier).
	
	The generic type is a non-structured octet string.

MT-TOKEN-AUTH-ENCR ::= OCTET STRING (SIZE(16))	The MD5 algorithm operating on token.
NET-ID ::= INTEGER (0 .. 1023)	10 bit Identifier for network on DLC- level [5]. Value 0 is for future use. Certain other numbers are reserved for the standardized use by public network operators.
NETW-OP-ID-GLOBAL ::= IA5String (SIZE(0..31))	ASCII string of up to 31 characters, that is up to 31 octets. This part is globally unique.
NETW-OP-ID-LOCAL ::= IA5String (SIZE(0..31))	ASCII string of up to 31 bytes, that is 31 characters/digits.
NETWORK-OPERATOR-ID ::= SEQUENCE { identifier-format IDENTIFIER-FORMAT, identifyer IDENTIFYER }	
NONCE ::= OCTET STRING (SIZE(16))	A random value used during authentication.
NO-OF-MT-ALIVE-PROCEDURES ::= INTEGER (0..4)	A 3 bit integer stating how many times the MT alive procedure shall fail before disassociation takes place. Sent from AP to MT.
NUMBER-OF-SAMPLES ::= INTEGER(0..16383)	14 bit field size. In percentile and complete report: The measurement length, given in number of 8 us samples taken in RSS statistics measurements.
OMNI-ANTENNA-USED ::= ENUMERATED { omni-antenna-not-used (0) omni-antenna-used (1)}	1 bit field size. Omni antenna definition: If the maximum antenna gain, measured in the horizontal plane, is 6 dB greater than the average antenna gain in the horizontal plane, the antenna is considered as a directional antenna. Otherwise the antenna is considered as non-directional. Note: the calculation of the average gain should be performed in linear scale, not dB scale.
OP-ID ::= SEQUENCE { unique-length UNIQUE-LENGTH c-u-g C-U-G netw-op-id-local NETW-OP-ID-LOCAL netw-op-id-global NETW-OP-ID-GLOBAL }	
PC-OFFSET ::= ENUMERATED { future-use0 (0) plus6db (1) plus3db (2) minus3db (3) minus6db (4) reset-tx-level (7)}	3 bit field size. See [4].
PDU-TYPE-CHOICE ::= CHOICE (lch RLC-LCH-PDU-TYPE schRLC-SCH-PDU-TYPE }	
PHY-MODE-LCH ::= ENUMERATED { nophy-mode-proposal (0) cpBPSK1-2 (1) cpBPSK3-4 (2) cpQPSK1-2 (3) cpQPSK1-3 (4) cp16QAM9-16 (5) cp16QAM3-4 (6) cp64QAM3-4 (7)}	4 bits field size. no proposal. BPSK, code rate ½. BPSK, code rate ¾. QPSK, code rate ½. QPSK, code rate ¾. 16QAM, code rate 9/16. 16QAM, code rate ¾. 64QAM, code rate ¾.

PHY-MODE-SCH ::= ENUMERATED { nophy-mode-propos (0) cBPSK1-2 (1) cBPSK3-4 (2) cQPSK1-2 (3) cQPSK1-3 (4) c16QAM9-16 (5) c16QAM3-4 (6) c64QAM3-4 (7)}	3 bits field size. no proposal. BPSK, code rate ½. BPSK, code rate ¾. QPSK, code rate ½. QPSK, code rate ¾. 16QAM, code rate 9/16. 16QAM, code rate ¾. 64QAM, code rate ¾.
PROFILE-VID ::= SEQUENCE { Profile-id PROFILE-ID, Profile version PROFILE-VERSION}	
PROFILE-VID-LIST ::= SEQUENCE (SIZE (0..5)) OF PROFILE-VID	
PROFILE-ID::= INTEGER (0..31)	5 bit profile id administered by BRAN. Profiles and values to be defined.
PROFILE-VERSION ::= INTEGER (0..31)	5 bit version per profile. Administered by BRAN.
RELEASE-CAUSE ::= ENUMERATED { unknown-release-cause (0) normal-release (1) low-qos (2) timed-out (3) lack-of-resources (4) network-operator-release (5)}	field size 4 bits.
REPETITION-COUNTER ::= INTEGER (0..4095)	Gives the number, the MAC Frame Counter value repeats in BCCH after the reception of the *start-mac-frame* parameter (1 repetition corresponds to 16 MAC frames). The frame, in which the frame counter of the BCCH takes the value *frame-count* for the first time after reception of the *start-mac-frame* parameter*,* corresponds to a *repetition-counter* value of 0. 12 bit field size.
REPORTING-INITIALIZED ::= ENUMERATED { reporting-not-initialized (0), reporting-initialized (1) }	field size 1 bit.
RETURN-FLAG ::= ENUMERATED { return-not-allowed (0), return-allowed (1)}	field size 1 bit.
RLC-VERSION ::= INTEGER (0..255)	Both MT and AP send their own version in Link Capability procedure. The present document is RLC version 1. 8 bit field.
RSS-INDEX ::= ENUMERATED { rss-minimum (0) rss-5-percent (1) rss-10-percent (2) rss-20-percent (3) rss-30-percent (4) rss-40-percent (5) rss-50-percent (6) rss-60-percent (7) rss-70-percent (8) rss-80-percent (9) rss-90-percent (10) rss-95-percent (11) rss-maximum (12)}	field size 4 bits. Percentage of the maximum value.
RSS-INDEX-LIST ::= SEQUENCE (SIZE (5)) OF RSS-INDEX	
RSS-STATISTICS ::= INTEGER(0..63)	Measurement result: RSS statistics on frequency *f.* Number of samples in the different percentiles, minimum or maximum. 6 bit field.
RSS-STATISTICS-LIST ::= SEQUENCE (SIZE (5)) OF RSS-STATISTICS	
RSS-VALUE ::= INTEGER(0..63)	6 bits. When used in the Link Capability messages, it is the latest Received-Signal-Strength value measured by the MT before the signal was sent. When used in the RLC_LINK_CAPABILITY_ACK or RLC_HANDOVER_LINK_CAPABILITY_ACK message, it is

	the same value that was sent in the RLC_LINK_CAPABILITY message. It is an index to an RSS value given in dBm, see [4].
SCH-LCH ::= ENUMERATED { sch (0) lch (1) }	1 bit field size.
SECTOR-ID ::= INTEGER (0..7)	3 bits field size.
SEND-NW-TOKEN ::= ENUMERATED { dont-send-nw-token (0) send-nw-token (1) }	1 bit field size.
SLEEP-GROUP ::= INTEGER (0..15)	Sleeptime = 2^SLEEP-GROUP (4 Bit). The value 0 means, that no sleeping is allowed.
START-AUTHENTICATION ::= ENUMERATED { dont-start-auth (0) start-auth (1)}	1 bit field size.
START-DUC-SET-UP ::= ENUMERATED { donot-start-setup (0) start-setup (1) }	1 bit field size.
START-ENCRYPTION ::= ENUMERATED { dont-start-encr (0) start-encr (1)}	1 bit field size.
START-INFO-TRANSFER ::= ENUMERATED { dont-start-info-transfer (0) start-info-transfer (1) }	1 bit field size.
START-MAC-FRAME ::= SEQUENCE { repetition-counter REPETITION-COUNTER frame-count FRAME-COUNTER}	Gives the exact MAC Frame to start FSA. *frame-count* gives the frame counter value in the BCCH of the starting MAC frame of FSA.
START-OF-MEASUREMENT ::= INTEGER(2..15)	The start of the measurement interval is given in number of MAC frames. The starting point for counting the start-of-measurement shall be counted from the frame the *start-of-measurement* parameter was received (number 0). The usage of the parameter is described in the DFS clauses. 4 bit field.
START-POINTER ::= INTEGER (0..8191)	Pointer to the position of the *Fixed Slot Allocation* in the MAC Frame. Same meaning and coding as for *Resource Grants,* 13 bit field size.
SUPPORTED64QAM ::= ENUMERATED { suppot64QAM (1) no-support64QAM (0) }	field size 1 bit.
SUPPORTED-FCA ::= ENUMERATED { support-fca (1) no-support-fca (0)}	field size 1 bit.
SUPPORTED-FSA ::= ENUMERATED { support-fsa (1) no-support-fsa (0)}	field size 1 bit.
TIME-GAP-ACH-UPLINK ::= INTEGER (0..7)	The minimum time, in µs, between the end of the ACH and the first uplink transmission of each individual MT [5]. 0 = 0, 1 = 16, 2 = 32, 3 = 64, 4 = 128, 5 = 256, 6 = 512, 7 = 1 024. 3 bit field.
TOKEN ::= OCTET STRING (SIZE(16))	Used for network handover authentication.
TRAFFIC-LOAD ::= ENUMERATED { notused0 (0) notused7 (7)}	3 bits, not used in phase 1.

TX-LEVEL ::= ENUMERATED { dbm30 (0) dbm27 (1) dbm24 (2) dbm21 (3) dbm18 (4) dbm15 (5) dbm12 (6) dbm9 (7) dbm6 (8) dbm3 (9) dbm0 (10) dbmm3 (11) dbmm6 (12) dbmm9 (13) dbmm12 (14) dbmm15 (15)}	4 bits. The AP transmit level that the MT has read from the field in the BCH of the measured AP.
UNIQUE-LENGTH ::= INTEGER(0..31)	The unique length field indicates the length of the globally unique part of the network operator identifier in bytes. 5 bit field.
USE-OMNI-ANTENNA ::= ENUMERATED { dont-use-omni-antenna (0) use-omni-antenna (1)}	1 bit field size.
WINDOW-SIZE ::= ENUMERATED { arq-not-used (0) w-size32 (1) w-size64 (2) w-size128 (3) w-size256 (4) w-size512 (5)}	3 bits. See [5].

Annex C (normative): RLC TIMERS

T_short = 16 frames = 32 ms

T_medium = 8 x T_short = 128 frames = 256 ms

T_long = 8 x T_medium = 1 024 frames = 2 048 ms

T_dfs = *start-of-measurement* + *measurement-window* + 5 frames

Table C.1: MT Timers

MT (testing AP)	Value
T_rbch_association_req	T_short
T_mac_id_assign	T_short
T_link_capability	T_short
T_key_exchange_mt	T_long
T_authentication	T_medium
T_authentication_ap	T_long
T_authentication-ap	T_medium
T_dm_common_key_distr_ack	T_short
T_info	T_short
T_group_join	T_short
T_group_leave	T_short
T_cl_broadcast_join	T_short
T_cl_broadcast_leave	T_short
T_disassociation_mt	T_short
T_connect_ack	T_short
T_setup_mt	T_short
T_connect_mt	T_short
T_release_mt	T_short
T_modify_req_mt	T_medium
T_modify_mt	T_medium
T_reset_mt	T_short
T_dfs_mt_init_report	T_short
T_sector_handover_req	T_short
T_handover_request	T_short
T_handover_notify	256 frames
T_nw_signalling_handover	T_medium
T_force_handover_return	256 frames
T_sleep_request	T_short
T_mt_alive	T_short
T_dm_setup_mt	T_short
T_dm_connect_mt	T_short
T_dm_connect_cmpt_mt	T_medium
T_relay_setup_mt	T_medium
T_dm_release_mt	T_medium
T_relay_release_mt	T_medium
T_dm_modify_req_mt	T_short
T_dm_modify_mt	T_short
T_dm_modify_cmpt_mt	T_medium
T_relay_modify_mt	T_medium
T_dm_reset_mt	T_medium

Table C.2: AP Timers

AP (testing MT)	Value
T_mac_id_assign_ack	T_short
T_link_capability_ack	T_short
T_key_exchange_ap	T_long
T_authentication_mt	T_long
T_authentication_ack	T_long
T_dm_common_key_distr	T_short
T_nw_signalling_handover_ack	T_short
T_info_ack	T_short
T_disassociation_ap	T_short
T_unicast_key_refresh	T_medium
T_common_key_refresh	T_medium
T_connect_ap	T_short
T_setup_ap	T_short
T_release_ap	T_short
T_modify_ap	T_medium
T_modify_req_ap	T_medium
T_reset_ap	T_short
T_force_handover	T_short
T_force_handover_return	256 frames
T_handover_association	T_short
T_handover_link_capability_ack	T_short
T_handover_notify	256 frames
T_nw_signalling_handover_ack	T_short
T_nw_handover_complete	T_short
T_ho_info_distribution	T_short
T_mt_alive_request	T_short
T_mt_absence	T_short
T_dm_setup_ap	T_short
T_dm_connect_ap	T_short
T_dm_connect_cmpt_ap	T_short
T_dm_release_ap	T_short
T_dm_modify_req_ap	T_short
T_dm_modify_ap	T_short
T_dm_modify_cmpt_ap	T_short
T_dm_reset_ap	T_short

Annex D (normative): SDL specification of the RLC protocol

The present document has been produced using Specification and Description Language - SDL and Abstract Syntax Notation No 1 - ASN.1.

The archive containing all available formats of the specification is ts_10176102v010201p0.zip.

D.1 The SDL Graphical form (SDL/GR)

The graphical form of SDL specification is available in tool specific format. All relevant files are contained in the archive rlcsdl_v03r02.zip included in the archive ts_10176102v010201p0.zip. The achive also contains the ASN.1 files that are part of the model.

D.2 The SDL Textual format (SDL/PR)

The SDL textual format is tool independent. It preserves the meaning of the specification but does not preserve the graphical layout. SDL/PR is available in the archive rlcsdl_v03r02_pr.zip included in the archive ts_10176102v010201p0.zip.

D.3 The SDL Common Interchange format (SDL/CIF)

The SDL Common Interchange format is tool independent. It preserves the meaning of the specification and the graphical layout. SDL/CIF is available in the file archive rlcsdl_v03r02_cif.zip included in the archive ts_10176102v010201p0.zip.

D.4 The ASN.1 files

The ASN.1 files that are specifying the abstract message structure are included with the SDL/GR but are also available in the archive rlcsdl_v03r02_asn.zip included in the archive ts_10176102v010101p0.zip.

D.5 The PDF format

The SDL specification including the ASN.1 part is available for viewing/printing in PDF format in the file rlcsdl_v03r02.pdf in ts_10176102v010201p0.zip.

Annex E (informative): Bibliography

- R. L. Rivest, A. Shamir, and L.M. Adleman: "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM", v. 21, n. 2, pp. 120-126, February. 1978.
- D. Harkins and D. Carrel: "The Internet Key Exchange (IKE)", November 1998.
- Bruce Schneier: "Applied cryptography Second Edition", John Wiley & Sons, New York NY, 1996 ISBN 0-471-12845-7.
- S. Crocker (Cybercash) and J. Schiller (MIT): "Randomness Recommendations for Security", December 1994.

History

Document history		
V1.1.1	April 2000	Publication
V1.2.1	April 2001	Publication